OPERATIONS RESEARCH
Vol. 55, No. 2, March–April 2007, pp. 378–394
ISSN 0030-364X | EISSN 1526-5463 | 07 | 5502 | 0378

DOI 10.1287/opre.1060.0356

An Efficient Trajectory Method for Probabilistic Production-Inventory-Distribution Problems

Miguel A. Lejeune
Tepper School of Business, Carnegie Mellon University, 5000 Forbes Avenue,
Pittsburgh, Pennsylvania 15213, mlejeune@andrew.cmu.edu

Andrzej Ruszczyński
Department of Management Science and Information Systems, Rutgers University, 94 Rockefeller Road,
Piscataway, New Jersey 08854, rusz@business.rutgers.edu

We consider a supply chain operating in an uncertain environment: The customers' demand is characterized by a discrete probability distribution. A probabilistic programming approach is adopted for constructing an inventory-production-distribution plan over a multiperiod planning horizon. The plan does not allow the backlogging of the unsatisfied demand, and minimizes the costs of the supply chain while enabling it to reach a prescribed nonstockout service level. It is a strategic plan that hedges against undesirable outcomes, and that can be adjusted to account for possible favorable realizations of uncertain quantities. A modular, integrated, and computationally tractable method is proposed for the solution of the associated stochastic mixed-integer optimization problems containing joint probabilistic constraints with dependent right-hand side variables. The concept of p-efficiency is used to construct a finite number of demand trajectories, which in turn are employed to solve problems with joint probabilistic constraints. We complement this idea by designing a preordered set-based preprocessing algorithm that selects a subset of promising p-efficient demand trajectories. Finally, to solve the resulting disjunctive mixed-integer programming problem, we implement a special column-generation algorithm that limits the risk of congestion in the resources of the supply chain. The methodology is validated on an industrial problem faced by a large chemical supply chain and turns out to be very efficient: it finds a solution with a minimal integrality gap and provides substantial cost savings.

Subject classifications: inventory/production: multistage supply chain; programming: integer, stochastic; transportation: scheduling.
Area of review: Optimization.
History: Received February 2005; revision received December 2005; accepted March 2006.

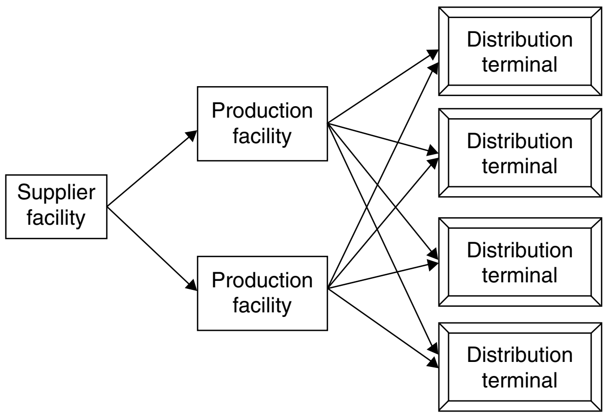
1. Introduction

Nowadays, companies no longer compete as independent entities, but rather as integrated parts of a supply chain; sustainable competitive advantages very often come from an adequate interfunctional and interorganizational collaboration and integration. Despite the abundant literature on the analysis and optimization of production-distribution systems, this remains a wide-open research area (Sarmiento and Nagi 1999, Swaminathan and Tayur 2003). Models involving the integrated management of the inventory, production, and distribution functions often take the form of very complex mixed-integer programming problems. Moreover, they often resort to decoupled or hierarchical solving methodologies: The decision variables linked to one of the functions are optimized and fixed before considering and optimizing the decision variables of the other functions. Such an approach does not guarantee the finding of very efficient solutions, and we thus adopt an integrated perspective, simultaneously considering the inventory, production, and distribution functions.

A number of deterministic models have been proposed for constructing replenishment plans (see, inter alia, Fumero and Vercellis 1999 and Dhaenens-Flipo and Finke 2001). Deterministic approaches assume perfect knowledge of the information impacting the construction of such replenishment plans or, alternatively, are based on buffering mechanisms against uncertainty.

However, the prevalence of multiple sources of uncertainty (Santoso et al. 2005), as well as the drawbacks of the buffering mechanisms, makes the use of deterministic models inappropriate. Therefore, we use a probabilistic programming approach (see, inter alia, Prékopa 1995, Ruszczyński and Shapiro 2003, and Prékopa 2003) for the construction of a reliable inventory-production-distribution plan over a multiple-period planning horizon for a three-stage supply chain involving suppliers, manufacturers, and distributors (see Figure 1). The demand is assumed to be stochastic and to have a discrete probability distribution. A constrained optimization perspective is adopted, in which nonstockout cycle service-level requirements are to be reached by the supply chain, enforcing the probability of stockout over the entire planning horizon to be limited from above. As in Bitran and Yanasse (1984), Holmberg and Tuy (1999), Kleywegt et al. (2004), Sox and Muckstadt (1996),

Figure 1. Three-stage supply chain.



and Yildirim et al. (2005) the demand is the only source of randomness considered. However, in many applications, lead times, reliability of carriers, or production levels can also be uncertain. Such sources of randomness are not considered in this paper.

Stochastic planning problems for multistage supply chains are the objects of intensive research. Graves and Willems (2000) propose a model for strategic safety-stock placement in a multiechelon supply chain subject to a bounded random demand. They focus on service times, rather than service levels. Bitran and Yanasse (1984), as well as Yildirim et al. (2005), formulate a stochastic dynamic programming problem that allows for the building of a production and sourcing plan over a multiperiod horizon, in which the demand is the only source of randomness and transportation decisions are out of scope. Both papers integrate service-level constraints in the formulation of the dynamic problem; they are individual service-level constraints ensuring that the probability of not having a stockout at a single period will be below a certain probability. An approximate method is suggested, and error bounds derived. The difficulty of integrating service-level constraints is also the reason why Sox and Muckstadt (1996) include a linear backorder component in the cost function instead of using service-level constraints. Kleywegt et al. (2004) formulate the stochastic inventory-routing problem as a discrete-time Markov process and derive dynamic programming approximations. They assume that the demand is random, and is independent and identically distributed across time periods. They do not enforce service-level requirements, but include a shortage penalty term in the objective function and consider the unsatisfied part of the demand as lost. Paschalidis et al. (2004) consider a multistage supply chain in which the demand is random and its unsatisfied part is backlogged. They use a base-stock production policy such that a facility produces if inventory falls below a certain level and idles otherwise to minimize the expected inventory costs at all stages, subject to keeping one-period stockout probability below a prescribed level. Holmberg and Tuy (1999) propose a methodology to construct a one-period production-transportation plan with multiple suppliers and demand nodes and with random demand. No service-level constraints are present, but a shortage cost is assessed at demand nodes.

The model proposed in this paper uses nonstockout cycle service-level constraints, which are joint probabilistic constraints (Charnes et al. 1958; Dentcheva et al. 2000; Prékopa 1970, 2003). A very complicated stochastic mixed-integer programming problem results, for which a modular solution method, seen as juxtaposition of cooperative ideas, is proposed. In the first module, we propose two different methods for approximating joint probabilistic constraints. The first one is based on the concept of p-efficiency (Prékopa 1990), which, adapted to the stochastic supply chain management setting, leads to the construction of p-efficient demand trajectories. After having identified the p-efficient demand trajectories, whose number is unknown, and possibly very large, we preprocess them and select a subset of "promising" ones. An algorithm relying on the preordered set concept is developed. Its application gives a high-dimensional disjunctive mixed-integer programming problem. In the second module, we develop a column-generation algorithm to supplement the branch-and-bound algorithm. Instead of simultaneously considering all the selected promising supply chain p-efficient demand trajectories and solving the corresponding disjunctive mixed-integer programming problem (Balas 1979), we design a column-generation algorithm, referred to thereafter as the *congestion relief* column-generation algorithm, that selects the p-efficient demand trajectories that limit the risk of congestion in the resources of the supply chain.

To the best of our knowledge, the proposed approach is the first one enforcing a nonstockout service level holding over the entire multiperiod planning horizon. The attainment of such a service level is crucial for the chemical company that motivated this study, and, more generally, for supply chains operating in highly competitive or malevolent environments—for example, military supply chains (Avital 2005, Kress et al. 2005, Kress 2002). Such a cycle service level corresponds to the probability of full responsiveness of the supply chain (Avital 2005, Kress 2002) and is exactly what such organizations are after.

This paper also provides a computationally tractable and very efficient solution method for handling the complex stochastic programming problem associated with the construction of such a probabilistic inventory-production-distribution plan. The outcome is a robust and sustainable plan that hedges against undesirable outcomes, and is such that it can be adjusted to account for possible favorable realizations of uncertain quantities. Finally, this paper validates the solution method on a real-life industrial problem faced by one of the major North American chemical supply chains.

The remainder of the manuscript is organized as follows. Section 2 presents the model, its notation and assumptions, and describes and analyzes alternative formulations for the considered problem. Section 3 details the modular solution methodology proposed. Section 4 reports the numerical results. Section 5 provides concluding remarks.

2. Model

2.1. Main Issues

As mentioned previously, we consider a three-stage supply chain in which suppliers hold stocks for raw materials that are delivered to manufacturers, or transformed into semifinished or end products and delivered to distributors.

Manufacturers use the raw materials provided by suppliers and transform them into finished products delivered to end customers. Suppliers and manufacturers have a limited production and storage capacity. The transportation is carried out with a fleet of heterogeneous carriers, differing in their loading capacity, speed, and ownership: They are either part of the internal fleet of the supply chain, or belong to some third-party logistics service providers. Carriers' lead times are known, and equal to the sum of the transportation, loading, and unloading times. Distributors face a stochastic demand at each period of the planning horizon.

We are interested in the situation in which it is extremely difficult or unrealistic to estimate the cost of not being able to satisfy demand on time. Instead, we enforce a *cycle service-level constraint* on the probability of the full responsiveness of the chain. The chain is not expected to be able to handle all demand levels; this would be equivalent to establishing extremely large safety stocks with excessive holding costs, which would eventually harm the profitability of the supply chain. The requirement to attain a prescribed cycle service level p is modeled as a probabilistic constraint: The probability of stockout (van der Heijden 2000) over the entire planning horizon is lower than $(1-p)$. Because we are focusing on the case when lagged satisfaction of the customer demand is not perceived as acceptable by highly demanding customers or those having a greater negotiating power, we do not allow backlogs.

The robustness of the probabilistic inventory-production-distribution model is evaluated with respect to:

- the cycle character of the service level: the prescribed nonstockout service level must be attained over a multiperiod planning horizon, and not at a collection of periods considered independently (stagewise service level); and
- the sustainability of the plan: it must be reproducible over the next planning horizon.

The probabilistic inventory-production-distribution model takes the form of a stochastic mixed-integer programming problem, whose solution provides the supply chain with an inventory-production-distribution plan enabling the supply chain to attain a prescribed nonstockout cycle service level, while minimizing its costs. More precisely, the plan must determine:

- the production scheme, i.e., the quantity of products and raw materials to be produced at each production facility at each period;
- the supply scheme, i.e., the quantity of products and raw materials to be sent from the supplier to the manufacturers, and from the manufacturers to the distributors at each time period;
- the ending inventory level required at each node in the supply chain and at each period;
- the scheduling and routing scheme for each carrier;
- the maintenance schedule for each carrier.

In the next subsections, we discuss the key elements of the probabilistic inventory-production-distribution model. We introduce and define random quantities and decision variables as well as the deterministic and probabilistic constraints that must mainly be focused on for developing a computationally tractable model. The reader is referred to the appendix for a description of all the parameters, decision variables, and deterministic constraints used in this model.

2.2. Random Variables

The random variables are:

- the end-product demand $d[j,t]$ at time t and terminal j, where $j \in J$ and $t \in T$; and
- the cumulative demand at node j up to time t:

$$\zeta[j,t] = \sum_{r=1}^{t} d[j,r], \quad j \in J,\ t \in T.$$

The cumulative demand variables are random variables that are not independent across time periods, i.e., the cumulative demand at time t is affected by the cumulative demand at earlier times. In this paper, we assume that the random variables have a discrete probability distribution: the demand $d[j,t]$ can take $l_{\max}$ different levels at each period t and node j.

Let L be a finite ordered set whose components $d^l[j,t]$ are the possible realizations of the random demand $d[j,t]$:

$$L = \{d^1[j,t], d^2[j,t], \ldots, d^l[j,t], \ldots, d^{l_{\max}}[j,t]\}, \quad j \in J,\ t \in T;$$

each component of L has probability $P^l[j,t] = P(d[j,t] = d^l[j,t])$, with $\sum_{l\in L} P^l[j,t] = 1$, $j \in J$, $t \in T$.

Denoting by $\zeta^{l'}[j,t]$ the realization of the cumulative demand $\zeta[j,t]$,

$$\zeta^{l'}[j,t] = d^{v_1}[j,1] + d^{v_2}[j,2] + \cdots + d^{v_t}[j,t], \quad 1 \leqslant v_1, v_2, \ldots, v_t \leqslant l_{\max},\ t \in T,$$

it can be seen that the number of realizations of the cumulative demand ζ at t depends on the number of realizations that the stagewise demand d can take, and is an exponential function of the number of levels that the stagewise demand can take at each period t. The probability of $\zeta[j,t]$ being realized as α is given by

$$P(\zeta[j,t] = \alpha) = \sum_{(v_1, v_2, \ldots, v_t) \in V_j(t,\alpha)} \prod_{r=1}^{t} P^{v_r}[j,r],$$

where $V_j(t,\alpha) = \{(v_1, v_2, \ldots, v_t)\colon d^{v_1}[j,1] + d^{v_2}[j,2] + \cdots + d^{v_t}[j,t] = \alpha\}$ is the set of all demand paths in the first t stages totaling α.

2.3. Objective Function

The objective function (to be minimized) is linear and represents the sum of the distribution, production, and holding costs supported by the supply chain. The distribution costs are defined with respect to the carrier used. More precisely, they depend on the cost per time unit, the lead time, and the number of shipments performed. The production costs vary with the production level at the facilities. The holding costs are proportional to the stock levels of products and raw material at all nodes in the supply chain.

An alternative to a cost-minimization function is to maximize the profits of the supply chain. However, the formulation of such an objective function entails some drawbacks. First, the optimal solution obtained with a profit-maximization objective function is very much impacted by the unit selling price, and may result in a very low service level. Second, using a profit-maximization objective function requires the introduction in the objective function of a penalty cost representing the cost of losing customers or market share. The quantification of the latter is very difficult, and would further compound the complexity of the problem.

The model proposed here does not attempt to quantify the cost of losing customers, but instead imposes, through the use of probabilistic constraints, the minimization of the costs subject to the attainment of a prescribed service level, thus effectively limiting the risk of loss of customers and of market share.

2.4. Deterministic Constraints

The following constraints play an important role in the formulation of the inventory-production-distribution problem.

Flow balance for products at distributors. Denoting by $z[j,t]$ the product inventory level at j and time t, and by $q[i,j,v,t]$ the amount of products delivered to node j at t with carrier v leaving from facility i, the flow-balance constraints at distributors

$$z[j,t] = z[j,t-1] + \sum_{i\in I}\sum_{v\in V} r[i,t]\cdot q[i,j,v,t] - d[j,t], \quad j\in J,\ t\in T, \quad (1)$$

enforce that the current random demand be met from the on-hand inventory and the supply received at the current period.

Storage capacity of products at distributors. Denoting by $z^{\max}[j]$ the maximum storage capacity at node j, the constraints

$$0 \leqslant z[j,t] \leqslant z^{\max}[j], \quad j\in J,\ t\in T, \quad (2)$$

limit the quantity of products that can be stocked, and also that can be delivered ahead of time for satisfying the random demand at node j.

Shipment indivisibility. Denoting by $x[i,j,v,t]$ the number of direct shipments from facility i to node j made at time t with carrier v, the constraints

$$x[i,j,v,t] \in \mathbb{Z}_+, \quad i\in I,\ j\in J,\ v\in V,\ t\in T \quad (3)$$

enforce the carriers' indivisibility requirement. Preliminary tests showed that attempting to construct a distribution scheme with carriers delivering diverse destinations within a single trip would lead to a computationally intractable problem. As a result, the distribution scheme allows direct shipments only.

Transportation capacity. Denoting by $C[v,t]$ the loading capacity of carrier v at time t, the constraints

$$0 \leqslant q[i,k,v,t] = C[v,t]\cdot x[i,k,v,t], \quad i\in I,\ k\in K,\ v\in V,\ t\in T \quad (4)$$

account for the limited capacity of the transportation carriers, and enforce "full load" shipments.

Time availability of carriers. The constraints below represent the limited time availability $a[v,t]$ of the carriers, and account for the lead time, equal to the sum of the loading time $l[i,v]$, the unloading time $u[k,v]$, and the traveling time $r[i,k,v]$, of a shipment made with carrier v between i and k:

$$\sum_{i\in I}\sum_{k\in K}(l[i,v]+2\cdot r[i,k,v]+u[k,v])\cdot x[i,k,v,t] \leqslant a[v,t], \quad v\in V,\ t\in T. \quad (5)$$

Maintenance of carriers. The constraints

$$x[i,j,v,t] \leqslant M\cdot\delta[v,t], \quad v\in V,\ t\in T, \quad (6)$$

in which M is the maximum number of shipments that can be done at time t with carrier v between nodes i and j, introduce a binary variable $\delta[v,t]$, which is equal to one if the carrier v is set up at time t, and is equal to zero otherwise. To enforce that carriers not be used in at least one period, allowing for their maintenance, we add the constraint

$$\sum_{t=1}^{|T|}\delta[v,t] = |T|-1, \quad v\in V. \quad (7)$$

Sustainability. Denoting by τ the last period in the planning horizon, we enforce that the ending inventory levels of products and raw materials be at least equal to the initial ones:

$$z[j,\tau] \geqslant z[j,0], \quad j\in J, \quad (8)$$

$$s[i,\tau] \geqslant s[i,0], \quad i\in I, \quad (9)$$

$$w[i,\tau] \geqslant w[i,0], \quad i\in I. \quad (10)$$

The other deterministic constraints are described in the appendix.

2.5. Probabilistic Constraints

The probabilistic constraints enforce the requirement that the supply be larger than the demand with a certain probability level p, at each node j and over the whole planning horizon, thus enabling the supply chain to reach the prescribed nonstockout cycle service level p. Stated differently, we want the probability of nonnegative inventory level $z[j,t]$ over the entire planning horizon to be at least p.

Denoting by $m[j,t] = \sum_{i\in I}\sum_{v\in V} q[i,j,v,t]$ the total quantity of product supplied to j at time t, and by $\omega[j,t] = \sum_{t'=1}^{t} m[j,t']$ the cumulative supply at node j by the end of period t, the value of the inventory level $z[j,t]$ is given by

$$z[j,t] = [z[j,t-1] + m[j,t] - d[j,t]]^+, \quad j \in J,\ t \in T. \quad (11)$$

If no stockout occurred at earlier stages, we can also write

$$z[j,t] = [z[j,0] + \omega[j,t] - \zeta[j,t]]^+, \quad j \in J,\ t \in T. \quad (12)$$

It is crucial to note that in (11), the inventory level $z[j,t]$ at time t and node j is defined in terms of the inventory level $z[j,t-1]$ at the preceding time period $(t-1)$, which is itself a random variable whose value is a function of the realization of the stochastic demand at the preceding periods $t' = 1,\ldots,t-1$:

$$z[j,t-1](d[j,t'],\ t'=1,\ldots,t-1), \quad t \in T.$$

With the inventory level $z[j,t]$ defined in terms of the stagewise demand $d[j,t]$, the stagewise supply $m[j,t]$ and the inventory level $z[j,t-1]$ at the preceding time period as in (11), it is not possible to derive a static formulation for the stochastic programming problem. Instead, we would obtain a dynamic formulation, taking the form of a multistage stochastic programming problem assuming that the quantity supplied, and thus the production-distribution scheme, is flexible and can be modified at each period. Considering a one-year planning horizon decomposed into monthly time periods, we would obtain a 12-stage stochastic programming problem for which a huge number of scenarios should be accounted for, making the problem intractable.

As a consequence of the discussion above, the inventory level at time t and node j is formulated using formula (12). Thus, the probabilistic constraints are formulated in terms of the cumulative demand and supply at each node j up to the successive periods $t \in T$, allowing for the construction of a tractable static stochastic programming problem. Therefore, the probabilistic constraints enforce that the cumulative supply be larger than the cumulative demand with a certain probability level p at each time period and each node.

We discuss below the formulation of chance programming problems containing individual probabilistic constraints (Charnes et al. 1958). For simplicity, we denote by x all problem variables, by $c^T x$ the objective function, and by $Ax \leqslant b$ all deterministic constraints. The problem

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \leqslant b, \\ & P(\omega[j,t] + z[j,0] \geqslant \zeta[j,t]) \geqslant p_{jt}, \quad t \in T,\ j \in J, \\ & x \geqslant 0, \end{aligned} \quad (13)$$

guarantees that the prescribed stagewise service level p_{jt} is attained at each period and each terminal separately. A formulation involving stagewise service level is used, for example, in Chen and Krass (2001) for determining an optimal replenishment policy. Such a service level guarantees that, on average, $100p_i\%$ of customers are satisfied all the time. It provides an expected value measure that reflects the steady-state nature of the supply chain. Its main virtue is its computational simplicity. However, even if the prescribed stagewise service level is high, formulation (13) results in an unknown, possibly very low, cycle service level, especially if a large number of periods is considered. Formulation (13) does not capture the reliability of the system as a whole, and is not always suitable in the supply chain management context. On the other hand, the service level constraint considered in this paper, $P(x_i \geqslant \xi_i,\ i=1,\ldots,\tau) \geqslant p$, constrains the probability of full responsiveness of the supply chain to be above a prescribed threshold, and is the service level privileged by military supply chains (Kress 2002, Kress et al. 2005, Avital 2005) or those operating in very competitive markets.

Because the use of individual probabilistic constraints is not suitable for the current problem and the use of joint probabilistic constraints with independent random variables located in the right-hand side of the constraints is not correct (the value of $\zeta[j,t]$ is not independent of that of $\zeta[j,t']$, $t'=1,\ldots,t-1$), we now consider a formulation with joint probabilistic constraints containing dependent random right-hand side variables. The joint probabilistic constraints can be written for

- each terminal j:

$$P(z[j,0] + \omega[j,t] \geqslant \zeta[j,t],\ t \in T) \geqslant p_j, \quad j \in J, \quad (14)$$

- the entire supply chain:

$$P(z[j,0] + \omega[j,t] \geqslant \zeta[j,t],\ j \in J,\ t \in T) \geqslant p. \quad (15)$$

Formulation (15) guarantees that the overall service level p be provided to all customers or terminals, which are viewed as one giant customer.

In contrast to (15), formulation (14) allows enforcing a differentiated service level, at least equal to p_j, because there is a possibility that $p_j \neq p_{j'}$, $j, j' \in J$, $j \neq j'$. Because the customers are independent of each other and are not interested in the service level provided to others, we opt for formulation (14), which suits the objectives of the supply chain. It allows the supply chain to customize the service level provided to its customers, reflecting their respective importance for the supply chain or accounting for the specific requirements of some of them.

Constraint (14) enforces that the inventory of each terminal be nonnegative over the whole planning horizon, this with probability p or, in other words, that the probability of shortage over the whole planning horizon is lower than or equal to $1-p$. We thereafter always refer to formulation (14) for joint probabilistic constraints.

2.6. Compact Model Formulation

In this section, we give a compact formulation of the stochastic mixed-integer programming problem corresponding to the probabilistic inventory-production-distribution model ensuring a nonstockout cycle service level:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & P(T_{jt}x \geqslant z_{jt},\, t=1,\ldots,\tau) \geqslant p_j, \quad j \in J, \\ & x = [x' \quad x''] \in \mathcal{R}_+ \times \mathcal{Z}_+. \end{aligned} \tag{16}$$

The expression $Ax \geqslant b$ represents all nonprobabilistic constraints, while the set of joint probabilistic constraints is denoted by $P(T_{jt}x \geqslant z_{jt},\, t = 1,\ldots,\tau) \geqslant p_j$, $j \in J$, in which the dependent random right-hand side is defined with respect to the cumulative demand and supply. The stochastic mixed-integer programming problem requires constraint $T_{jt}x \geqslant z_{jt}$ to hold, with some probability p_j, for all periods t.

The symbol $\mathcal{R}_+$ refers to the appropriate multidimensional spaces of nonnegative real vectors, while the symbol $\mathcal{Z}_+$ refers to the appropriate multidimensional space of nonnegative integer vectors.

2.7. Alternative Formulations

Another possibility is to formulate the problem as a *multistage stochastic programming problem*. Its general structure is as follows (see Ruszczyński and Shapiro 2003):

$$\begin{aligned} \min \quad & c_1^T x_1 + c_2^T x_2 + c_3^T x_3 + \qquad\qquad c_\tau^T x_\tau \\ \text{subject to} \quad & A_{11}x_1 = D_1 \\ & A_{21}x_1 + A_{22}x_2 = D_2 \\ & A_{23}x_2 + A_{33}x_3 = D_3 \\ & \cdots \\ & A_{\tau,\tau-1}x_{\tau-1} + A_{\tau,\tau}x_\tau = D_\tau \\ & x_1 \in X_1,\ x_2 \in X_2,\ x_3 \in X_3, \quad x_{\tau-1} \in X_{\tau-1},\ x_\tau \in X_\tau. \end{aligned}$$

Here, D_t represents the demand vector at stage t, and x_t denotes all variables associated with stage t, where $t = 1,\ldots,\tau$. The matrix A and all vectors c_t are known. The sets X_t are defined by all constraints associated with stage t and not involving demands. As the vector x_t may depend on the demand values $d_1,\ldots,d_{t-1}$ realized up to time $t-1$, it is, in fact, a *function* of these observations. An established way to approach such problems is to introduce a scenario tree of possible demand trajectories, and to represent x_t as a function of the node of the tree at level t. In our case, given that we have 12 periods and a 13-dimensional demand vector at each period, even a coarse approximation of the demand distribution by a scenario tree leads to a tree of astronomical dimensions. The resulting multistage stochastic programming problem is far beyond the capacity of the existing approaches. An additional drawback of the multistage formulation is that the enforcement of a service level for the entire planning horizon is not feasible; instead, we must introduce a penalty in the objective function that reflects the cost of losing a customer or market share.

Yet another classical approach to sequential decision problems under uncertainty is *dynamic programming*. It is easiest to discuss in our case under the additional assumption that the demand vectors at different stages are *independent*. The complete state vector in our system, denoted S_t, contains all inventory levels at production facilities and distribution centers, as well as the maintenance state of each carrier. Its dimension is usually very large. The vector of production and distribution decisions at stage t is denoted U_t. Dynamic programming aims at establishing a policy $\pi = (\pi_1,\ldots,\pi_\tau)$ such that the decisions can be calculated as functions of the state,

$$U_t = \pi_t(S_t), \quad t = 1,\ldots,\tau.$$

Because the general form of the optimal policy is not known, a discretization of the state space has to be employed. In this case, the policy is sought in the form of a table: what to do at each (discrete) state value. The problem here lies in the high dimension of the state vector, which results in a combinatorial explosion of the number of discretization points. A simplification may be achieved by restricting the class of policies considered, but we are not aware of a suitable approach in the case of limited transportation resources. The difficulties are compounded by stochastic dependence of the demand vector across the stages. This, in the best case, can be resolved by augmenting the state vector, which adds complexity to the discretization procedure. Overall, the computational burden associated with the dynamic programming approach in our case appears to be unsurmountable.

A significant drawback of both formulations is that they do not allow the enforcement of service level for the entire planning horizon. Only an indirect mechanism is possible, by a penalty term in the objective function. This requires the evaluation of loss of customer goodwill or market share, the quantification of which is artificial (Bitran and Yanasse 1984).

3. Modular Solving Methodology

3.1. Overview

Discussing the challenges linked with mixed-integer programming problems, Bixby et al. (1999) insist on the necessity to combine several approaches, calling this a "barrage of different but cooperating ideas." The complexity of the mixed-integer programming problem on hand is, moreover, further compounded by its stochastic character. That is why we develop a modular solving methodology, which is integrated and computationally tractable. It involves the joint use of two modules:

- the conversion of the joint probabilistic constraints, and
- the solution of the resulting complex (disjunctive) mixed-integer programming problem for which we design a congestion-relief column-generation algorithm.

The first challenge is to find computationally tractable approximations for the joint probabilistic constraints, allowing the resulting model to be handled by commercial mixed-integer solvers. It is clear that the replacement of the random variables by their average values is not a suitable option. It would result in a service level close to 50%, which is not acceptable. To approximate probabilistic constraints, we propose and test two methods based on

- the scenario analysis approach, and
- the concept of a p-efficient point for a discrete distribution (Prékopa 1990).

It will be shown that the concept of p-efficiency is the most appropriate for approximating the joint probabilistic constraints. The reliance on this concept requires developing an enumerative algorithm for generating the p-efficient demand trajectories associated with each joint probabilistic constraint. Although the generation of p-efficient demand trajectories significantly reduces the number of demand trajectory candidates, this number remains large, not allowing us to rely on the brute force approach that involves solving the mixed-integer programming problem associated with each specific p-efficient demand trajectory. Because all p-efficient demand trajectories are not equal, the focus must be selecting promising p-efficient demand trajectories. We propose a preordered set algorithm for preprocessing the demand trajectories. The outcome is a disjunctive mixed-integer programming problem, in which the number of candidates is significantly lower than initially.

The second challenge is associated with the difficulty of solving a complex mixed-integer programming problem, for which the sole reliance on the branch-and-bound algorithm embedded in the CPLEX solver is not sufficient to reach an optimal or close-to-optimal solution. We therefore implement a congestion-relief column-generation algorithm; it consists of selecting new p-efficient demand trajectories that are such that the congestion in the resources of the supply chain is relieved, thus reducing the risk of bottleneck and allowing the design of a better replenishment plan.

3.2. Module I: Approximation of Joint Probabilistic Constraints

3.2.1. Scenario Analysis. Scenario analysis approaches induce the decomposition of a stochastic programming problem into a finite number of many deterministic programming problems or scenarios that may be solved through suitable heuristics or branch-and-bound algorithms. The scenario-analysis-based method proposed in this paper for converting the probabilistic constraints involves two steps. In the first step, a set S of scenarios is generated using a simulation code, in which the likelihood of a scenario to be incorporated in the set S varies, and reflects the possibility of its occurrence. In the second step, we construct an inventory-production-distribution plan such that all the demand requirements are satisfied for a percentage at least equal to p of the number of the scenarios considered.

The probabilistic constraints

$$P(z[j,0]+\omega[j,t]\geqslant\zeta[j,t],\ t\in T)\geqslant p_j,\quad j\in J,$$

are replaced by the requirement that the inequality

$$z[j,0]+\omega[j,t]\geqslant\zeta[j,t,s],\quad t\in T,\ j\in J,\ s\in S,$$

where the notation $\zeta[j,t,s]$ refers to the cumulative demand at terminal j and period t in the sth scenario, must hold for a sufficiently large number of scenarios, having total probability at least equal to p.

Further assuming that all the scenarios included in the subset S are equally weighed, two approaches are possible. In the first approach, the supply chain is viewed as having a single giant customer, and scenarios are generated for this customer. The problem is then formulated as follows:

$$\min\quad c^T x \tag{17}$$

$$\text{subject to}\quad Ax\geqslant b, \tag{18}$$

$$z[j,0]+\omega[j,t]\geqslant\zeta[j,t,s]\cdot(1-\theta[s]),\quad j\in J,\ t\in T,\ s\in S, \tag{19}$$

$$\sum_{s\in S}\theta[s]\leqslant(1-p)\cdot|S|, \tag{20}$$

$$\theta[s]\in\{0,1\},\quad s\in S, \tag{21}$$

$$x=[x'\quad x'']\in\mathscr{R}_+\times\mathscr{Z}_+, \tag{22}$$

where $\theta[s]$ is a binary variable defined as follows:

$$\theta[s]=\begin{cases}0 & \text{if all constraints in scenario } s \text{ are satisfied,}\\ 1 & \text{otherwise.}\end{cases} \tag{23}$$

In the second approach, a set S_j of scenarios s_j is generated for each single customer j. Constraints (20) and (21) in problem (17)–(22) are replaced by

$$\sum_{s\in S_j}\theta[s,j]\leqslant(1-p)\cdot|S_j|,\quad j\in J, \tag{24}$$

and

$$\theta[s,j]\in\{0,1\},\quad s\in S_j,\ j\in J. \tag{25}$$

Here, $\theta[s, j]$ is a binary variable defined as follows:

$$\theta[s,j] = \begin{cases} 0 & \text{if all constraints in scenario } s \text{ for customer } j \text{ are satisfied,} \\ 1 & \text{otherwise.} \end{cases} \quad j \in J. \quad (26)$$

Constraint (19) remains almost the same:

$$z[j,0] + \omega[j,t] \geqslant \zeta[j,t,s] \cdot (1 - \theta[s,j]), \quad j \in J,\ t \in T,\ s \in S_j.$$

The second approach is more appropriate when the customers' demands are independent.

Instead of assuming that all scenarios s are equally likely, we could also assign a certain weight or probability $\beta[s]$ to each scenario s. Constraint (20) would then be replaced by

$$\sum_{s \in S} \beta[s] \cdot \theta[s] \leqslant 1 - p,$$

with

$$\sum_{s \in S} \beta[s] = 1, \quad 0 \leqslant \beta[s] \leqslant 1,\ s \in S.$$

3.3. The Concept of p-Efficiency

3.3.1. Definition. In this section, we introduce the concept of a p-efficient point for a discrete probability distribution, the definition of which was given by Prékopa (1990).

Let $z \in \mathcal{Z}_+^n$ be an n-dimensional integer random variable, and let F be its probability distribution function, $F(v) = P(z \leqslant v)$, $v \in \mathcal{R}^n$. Fix $p \in (0, 1)$.

Definition 3.1. A point $v \in \mathcal{R}^n$ is called a p-efficient point of the probability distribution function F if

$$F(v) \geqslant p \quad \text{and} \quad (27)$$

$$\text{there is no } v' \leqslant v,\ v' \neq v \quad \text{such that } F(v') \geqslant p. \quad (28)$$

If we consider a one-dimensional probability distribution, there is a unique p-efficient point, called the pth-quantile of the probability distribution, and is defined by

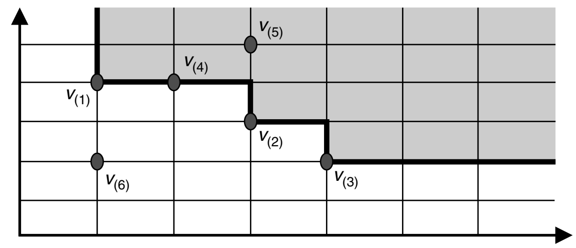
$$F^{-1}(p) = \min\{v\colon F(v) \geqslant p\}. \quad (29)$$

Finding the pth-quantile is straightforward. However, for multidimensional probability distributions, finding the set of p-efficient points, which is nonempty, finite (Dentcheva et al. 2000), but unknown, is not an easy task.

From (27) and (28), it follows that for every $y \in \mathcal{R}^n$ such that $F(y) \geqslant p$,

$$y \geqslant v = (v_1, \ldots, v_i, \ldots, v_n), \quad v \in \mathcal{R}^n, \quad (30)$$

Figure 2. p-efficient points $v_{(1)}$, $v_{(2)}$, $v_{(3)}$ for $z = (z_1, z_2)$.



where v_i denotes the p-efficient point of the one-dimensional marginal distribution $F_i(\cdot)$, $i = 1, \ldots, n$.

An illustration of the concept of p-efficiency for a two-dimensional probability distribution is given in Figure 2. All points located on the bold line or above it, in the grey-shaded region, have a value of the cumulative probability distribution $F(v)$ of at least p, which satisfies condition (27). The point $v_{(6)}$ does not satisfy condition (27) and is not p-efficient. Considering (28), it is apparent that the points $v_{(4)}$ and $v_{(5)}$ are not p-efficient either: $v_{(1)} \leqslant v_{(4)}$, $v_{(2)} \leqslant v_{(5)}$, and $v_{(1)} < v_{(5)}$. The only p-efficient points are thus $v_{(1)}$, $v_{(2)}$, and $v_{(3)}$.

3.3.2. p-Efficiency in a Probabilistic Supply Chain Management Setting. In this section, we propose an adaptation of the concept of p-efficiency for the probabilistic inventory-production-distribution setting. Based on the probability distribution of the demand, we construct the terminal p-efficient demand trajectories DT_j^p for each terminal j.

Definition 3.2. A terminal p-efficient demand trajectory DT_j^p is a τ-dimensional vector, which is a p-efficient point of the τ-dimensional random vector of cumulative demand trajectories at terminal j.

We denote terminal p-efficient trajectories as

$$DT_j^p = [\zeta^p[j,1], \ldots, \zeta^p[j,t], \ldots, \zeta^p[j,\tau]], \quad j \in J, \quad (31)$$

and we use S_j^p to denote the set that contains all p-efficient demand trajectories for terminal j.

A supply chain deals with a multitude of customers, and we also introduce the concept of supply chain p-efficient demand trajectory.

Definition 3.3. A supply chain p-efficient demand trajectory DT^p is defined as the combination of γ terminal p-efficient demand trajectories,

$$DT^p \in S^p\colon\ DT^p = [DT_1^p, \ldots, DT_j^p, \ldots, DT_\gamma^p], \quad (32)$$

where S^p denotes the set of supply chain p-efficient demand trajectories.

It has been shown by Dentcheva et al. (2000) that for any discrete probability distribution, and for any $0 < p < 1$, the set of all p-efficient points is nonempty and finite. This implies that for each terminal j, we can find at least one terminal p-efficient demand trajectory and therefore, from (32), at least one supply chain p-efficient demand trajectory.

After having generated the terminal p-efficient demand trajectories, we can substitute the constraints

$$z[j,0]+\omega[j,t] \geqslant \zeta^p[j,t], \quad t \in T,\ j \in J, \tag{33}$$

for the joint probabilistic constraints

$$P(z[j,0]+\omega[j,t] \geqslant \zeta[j,t],\ t \in T) \geqslant p_j, \quad j \in J. \tag{34}$$

The right-hand side of constraint (33) is the cumulative demand that characterizes a p-efficient demand trajectory (31): If the requirements imposed by *any* p-efficient demand trajectory are satisfied, the supply chain is then able to attain the prescribed cycle nonstockout service level.

The substitution of (33) for (34) transforms the stochastic mixed-integer programming problem (16) into the disjunctive mixed-integer programming problem given below:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & Tx \geqslant DT^p, \\ & DT^p \in S^p, \\ & x = [x' \quad x''] \in \mathcal{R}_+ \times \mathcal{Z}_+, \end{aligned} \tag{35}$$

where the expression $Tx \geqslant DT^p$ stands for (33) and must hold for at least one p-efficient demand trajectory. The supply chain p-efficient demand trajectory DT^p is a multidimensional unknown vector that must be found prior to the optimization of (35) with respect to x.

Constructing for each supply chain p-efficient demand trajectory $DT^p_{(\psi)}$ the cone $K_{(\psi)}$,

$$K_{(\psi)} = DT^p_{(\psi)} + \mathcal{R}_+, \quad DT^p_{(\psi)} \in S^p,$$

where $DT^p_{(\psi)}$, a combination of γ terminal p-efficient demand trajectories of dimension τ, refers to the ψth supply chain p-efficient demand trajectory, we obtain the disjunctive formulation below:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & Tx \in \bigcup_{\psi \in S^p} K_\psi, \\ & x = [x' \quad x''] \in \mathcal{R}_+ \times \mathcal{Z}_+. \end{aligned}$$

The definition of p-efficiency makes it rational to use a level search approach to identify p-efficient points. In Beraldi and Ruszczyński (2002), the random variables are binary and the authors propose a forward (i.e., to start from the lowest level and move from one level to the next until no more p-efficient point can be found) and a backward enumeration approach that take advantage of the binary nature of the variables. We deal here with general integer random variables and use a forward algorithm close to the enumerative algorithm proposed in Prékopa et al. (1998). Our algorithm differs in its enumerative scheme: It relies on (30) and the knowledge of the pth quantile of the cumulative demand at the last period of the planning horizon to eliminate levels from the start and so to reduce the enumerative burden.

As the number of terminal and supply chain p-efficient demand trajectories may be very large, the associated disjunctive mixed-integer programming problem may be very complex. The brute force approach that involves the solution of the mixed-integer programming problem associated with every supply chain p-efficient demand trajectory becomes intractable. It is also likely that the p-efficient demand trajectories are not all equally difficult to satisfy by the supply chain. We cannot simply use and consider a random sample of them. We should concentrate on the most promising p-efficient demand trajectories, and we therefore propose a preprocessing algorithm based on the concept of a preordered set for that purpose. The idea underlying this algorithm is based on the notion of congestion in the resources of the supply chain, and its impact on the profitability of the supply chain.

The preordered set algorithm involves the identification of the periods at which the demand is overproportional, as compared to the demand arising at other periods in the planning horizon. These periods are assigned to the set of critical periods T^b. At these periods, the risk of congestion and bottleneck in the resources of the supply chain, leading to suboptimal delivery schedule and additional costs, is higher. The preordered algorithm selects the p-efficient demand trajectories that have a comparatively low level of demand over the set of critical periods.

Definition 3.4. We say that a terminal p-efficient demand trajectory $DT_{j,(m)}$ dominates one of its equivalent $DT_{j,(n)}$, if $DT_{j,(m)}$ has a lower cumulative demand than $DT_{j,(n)}$ at each period in the set T^b of critical periods:

$$DT_{j,(m)} \succeq DT_{j,(n)} \quad \text{if } \zeta_{(m)}[j,t] \leqslant \zeta_{(n)}[j,t], \quad DT_{j,(m)}, DT_{j,(n)} \in S^p_j,\ t \in T^b. \tag{36}$$

Conversely, $DT_{j,(n)}$ is said to be dominated by $DT_{j,(m)}$.

We note that the relation defined above is defined over the set of critical periods T^b only.

All p-efficient demand trajectories that are dominated by at least one of their counterparts according to (36) are no longer considered. The nondominated p-efficient demand trajectories are included in the set S^{po}_j, the set of the preprocessed p-efficient demand trajectories, which is a subset of S_j^p. The selection of the nondominated trajectories can be easily accomplished by a simple sorting method.

Denoting by DT_j^{po} a terminal p-efficient trajectory selected with the preordered set algorithm by

$$DT^{po} = [DT_1^{po}, \ldots, DT_j^{po}, \ldots, DT_\gamma^{po}] \in S^{po},$$

a supply chain p-efficient trajectory selected with the preordered set algorithm, and by S^{po} the set of supply chain p-efficient demand trajectories selected with the preordered set algorithm, the programming problem (35) is transformed:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & Tx \geqslant DT^{po}, \\ & DT^{po} \in S^{po}, \\ & x = [x' \quad x''] \in \mathscr{R}_+ \times \mathscr{Z}_+. \end{aligned} \tag{37}$$

3.4. Module II: Enhanced Branch-and-Bound Algorithm

The disjunctive mixed-integer programming problem (37) is solved with CPLEX, and we design a specific column-generation algorithm to complement the CPLEX standard branch-and-bound algorithm.

In our method, the master and auxiliary problems, respectively, consist of optimizing the production-inventory-distribution scheme for a given p-efficient demand trajectory and the selection of another more appropriate p-efficient demand trajectory. More precisely, our column generation is an algorithmic procedure selecting a p-efficient demand trajectory that relieves congestion in the planning horizon: The selection of an appropriate p-efficient demand trajectory could alleviate the bottleneck in the resources of the supply chain and lead to a less costly solution. The algorithm proposed is called the *congestion-relief column-generation algorithm*, and is grounded in the widely accepted idea that congestion negatively affects production efficiency (see, inter alia, Färe and Grosskopf 2000 and Cooper et al. 2004).

In this paper, we define the bottleneck with respect to the period at which the use of the distribution resources is maximal. Bottleneck of resources is very likely in the case of demand seasonality. It is assumed that distribution requirements for the period(s) at which the demand is the highest will create a bottleneck in the resources, thus leading to inefficiency and to a suboptimal inventory-production-distribution scheme, and that the supply chain profitability can be increased by modifying the distribution schedule (i.e., anticipative distribution).

Prior to the description of the algorithm, we introduce the slack $\mu[v, t] \geqslant 0$ of the distribution constraint

$$\begin{aligned} \mu[v, t] = a[v, t] - \sum_{i \in I} \sum_{k \in K} (l[i, v] + 2r[i, k, v] + u[k, v]) \\ \cdot x[i, k, v, t], \quad v \in V,\ t \in T. \end{aligned} \tag{38}$$

The slack $\mu[v, t]$ is interpreted as the amount of unused time of carrier v at time t.

Step-by-Step Decomposition.

Step 0. Select a preprocessed p-efficient demand trajectory

$$DT_{(m)}^{po} \in S^{po}.$$

Step 1. Solve the resulting master problem:

$$\begin{aligned} \min \quad & = c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & Tx \geqslant DT_{(m)}^{po}, \\ & x = [x' \quad x''] \in \mathscr{R}_+ \times \mathscr{Z}_+. \end{aligned} \tag{39}$$

Step 2. Identify the *critical period* $t^b \in T$. It is the period at which the consumption of the distribution resources is the highest, and is thus the period in the planning horizon that has the lowest value for the slack (38) that quantifies the amount of unused distribution resources:

$$t^b = \arg\min \left\{ \mu[t] = \sum_{v \in V} \mu[v, t],\ t \in T \right\}.$$

Step 3. Identify the set J^b of terminals to which shipments are carried out at time t^b. These terminals, called *binding terminals*, are those consuming distribution resources at the critical period

$$J^b = \left\{ j \in J\colon \sum_{i \in I} \sum_{v \in V} x[i, j, v, t^b] > 0 \right\}. \tag{40}$$

Step 4. Select a terminal $j' \in J^b$, and check whether the selection of a demand trajectory $DT_{j', (m+1)}^{po}$ other than the current one can lead to a reduction in the distribution costs for j'. If this is the case, we proceed to a demand trajectory substitution for j'. We go to Step 5 and solve the new master problem resulting from the above substitution.

If there is no distribution costs reduction possible for terminal j', we consider another binding terminal in J^b. If there is no possibility of reducing the traveling costs for any of the binding terminals, we stop.

Step 5. Check whether the reduction in the distribution costs associated with j' can be obtained without affecting any of the other terminals j, $j \in J \backslash \{j'\}$. This is done by solving the updated master problem. If a reduction in the distribution costs of j' can be reached without increasing the distribution costs of the others, we go to Step 2. Otherwise, we go back to Step 5 and consider another binding terminal. We stop after having considered all the binding terminals and not having found any possible improvement.

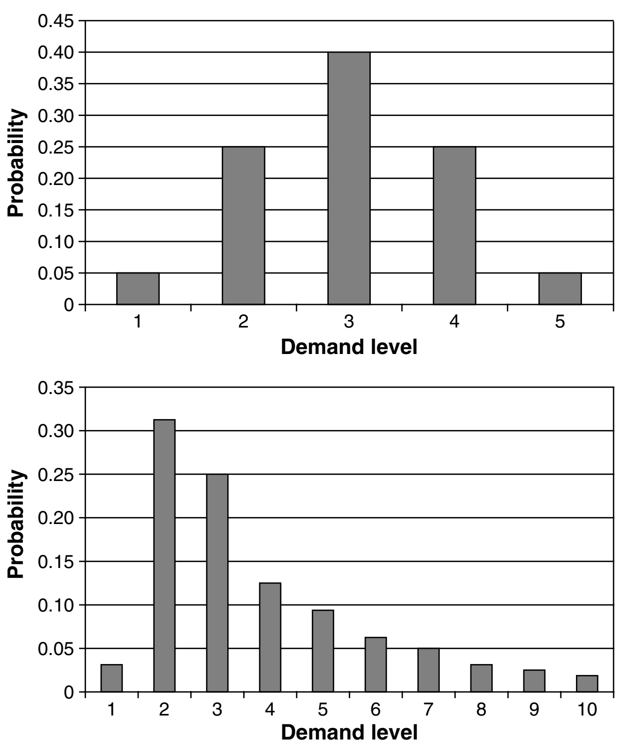
The algorithm is designed in such a way that, at each iteration, it answers the following question: Considering j', which consumes distribution resources at t^b, is it possible to find another p-efficient demand trajectory for j' that reduces the distribution costs for j', while leaving unchanged the distribution costs for the other $j \in J\backslash\{j'\}$? Using the definition of Cooper et al. (2004), saying that "efficiency is present if it is not possible to improve some of the inputs or outputs of the decision-making unit without worsening any of its other inputs or outputs," the stopping criterion can be restated as stopping when no further efficiency gain in the distribution scheme can be identified.

4. Numerical Results

4.1. Test Laboratory

As a test bed for the proposed solution techniques, we use the data provided by one of the three largest North American chemical companies, the General Chemical Group, which produces soda ash and calcium chloride. A one-year planning horizon decomposed into monthly time periods is considered. Based on historical data, we derive two probability distributions for the random demand. In the first one, the demand $d[j, t]$ can take five different levels at each time period, and the probability distribution resembles the probability density function of the standard normal probability density function (see the left-hand side of Figure 3). In the second one, $d[j, t]$ can take 10 different levels, and the probability distribution resembles the standard lognormal probability density function (see the right-hand side of Figure 3).

Figure 3. Probability distribution.



4.2. Evaluation of Results

4.2.1. Results with Scenario Analysis. As explained in §3.2.1, the scenario analysis approach transforms the original stochastic mixed-integer programming problem into a mixed-integer one requiring the construction of replenishment plans satisfying all the requirements for at least $100p\%$ of the considered scenarios.

The number of scenarios is a critical parameter for that approach: For the results to be meaningful, a sufficiently large number $|S|$ of scenarios must be generated. However, the difficulty of solving the resulting problem increases with the number of scenarios because a binary variable is added into the optimization problem for each additional scenario. Although the preliminary tests reported below were conducted with an arbitrary number (40 and 400) of scenarios, we believe that the results are clear enough to draw conclusions about the scenario analysis approach.

First, including 400 scenarios in S, the corresponding mixed-integer programming problem proves impossible to solve. While attempting to solve the programming problem using AMPL and the CPLEX solver, the computer runs out of memory. The reason lies in the large number of binary variables to be accounted for, which significantly raises the size of the arborescence in the branch-and-bound tree. Eventually, the solving process is terminated with an unrecoverable failure message due to limited memory.

Second, including 40 scenarios in S, the corresponding mixed-integer programming problem is tractable by the CPLEX solver. Two comments emerge from examination of the solution obtained when considering 40 scenarios:

- a single-integer feasible solution is found, and is obtained at the very early stages—i.e., within the first 1,000 nodes—of the algorithmic process, and is never improved afterward: There is no convergence toward a better feasible integer solution;
- the best solution obtained induces a very high cost of \$6,853,900,[1] which is more than 20% higher than the cost obtained with the method described in the next section. It can also be seen that the ending inventory level is, for about 70% of the terminals considered, twice as high as the required safety stock level, thus indicating that the replenishment is excessive. The impossibility of reaching a better solution using the scenario analysis approach is due to the presence of symmetry (see Barnhart et al. 1998 for a discussion of the effects of symmetry). Indeed, a shipment from i to j processed with v results in the same quantity supplied, the same total costs, and the same traveling times regardless of the period the shipment is carried out. By swapping some of the shipments planned in the delivery scheme—i.e., a shipment to be processed at time t with v departing from i and heading to j is cancelled and replaced by one processed at another time period $t' \neq t$—some very similar solutions can be obtained. Consequently, a fractional solution excluded at a given node in the branch-and-bound tree may reappear, with slightly different values

Table 1. Expected demand distribution.

t	$\sum_{j\in J} d[j,t]$	$\sum_{j\in J} \zeta[j,t]$	$\sum_{j\in J} d[j,t] / \sum_{j\in J} \zeta[j,\tau]$
1	2,418	2,418	0.025
2	20,620	23,038	0.211
3	31,102	54,140	0.319
4	12,991	67,131	0.133
5	7,788	74,919	0.080
6	3,448	78,367	0.035
7	2,350	80,717	0.024
8	2,654	83,371	0.027
9	4,230	87,601	0.043
10	4,403	92,004	0.045
11	3,163	95,167	0.032
12	2,443	97,610	0.025

of the variables, at another node of the tree. The problem considered thus has a very large number of alternate optima dispersed over the branching tree, which makes the usefulness of the pruning process by bounds questionable and stresses the need to supplement the branch-and-bound algorithm.

4.3. Results with p-Efficiency and Congestion-Relief Column-Generation Algorithm

In this section, we evaluate the solution methodology that jointly relies on the concept of p-efficiency and the congestion-relief column-generation algorithm. Results are provided for p equal to 95% and 97% for the two discrete distributions described above.

4.3.1. p-Efficiency and Preprocessing Algorithm. The identification of the p-efficient demand trajectories significantly lessens the number of demand trajectory candidates. It is clear, however, that the number of supply chain p-efficient demand trajectories, which are all combinations of terminal p-efficient demand trajectories, remains high, and does not allow the recourse to the brute force approach. Some preliminary tests showed that the p-efficient demand trajectories do not impact the supply chain the same way: While the satisfaction of the requirements imposed by different p-efficient demand trajectories enables reaching of the prescribed nonstockout service level, they lead to very different cost levels for the supply chain.

In the problem handled, it can be seen that the demand is characterized by a very marked seasonality. The first and second columns of Table 1, respectively, report the expected total demand $\sum_{j\in J} d[j,t]$ at, and the expected total cumulative demand $\sum_{j\in J} \zeta[j,t]$, up to each period t. The proportion of the expected total demand arising at each period is given in the last column of Table 1.

It can be seen that the demand is very strong at Periods 2, 3, and 4: The demand at these periods is overproportional as compared to the demand arising at the other periods. Indeed, the sum of the expected demands arising at these three periods, which we include in the set T^b of critical periods,

$$T^b = \{2,3,4\}, \quad T^b \subset T,$$

amounts to more than two-thirds of the total expected demand over the entire planning horizon.

The application of the preordered set algorithm described in §3 for preprocessing the p-efficient demand trajectories results in a very significant reduction in the number of the retained p-efficient demand trajectories (Table 2). Columns 2 and 7 indicate the total number of demand trajectories $|S_j^0|$ for the two considered distributions. Columns 3, 5, 8, and 10 report the number of p-efficient demand trajectories $|S_j^p|$, while Columns 4, 6, 9, and 11 report the number of preprocessed p-efficient demand trajectories $|S_j^{po}|$. The reported results are obtained for p equal to 95% and 97%.

4.3.2. Congestion-Relief Column-Generation Algorithm. The congestion-relief column-generation algorithm is now applied to find the best demand trajectory among the preprocessed p-efficient demand trajectories.

To evaluate the quality of the integer solution found, we compute a *global lower bound LB* for the disjunctive programming problem, and we use it as a benchmark. The global lower bound is obtained by constructing the convexification of the continuous relaxation of (35), which takes the following form:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax \geqslant b, \\ & T_j x \geqslant \sum_{m=1}^{|S_j^p|} \lambda_{j,m} DT_{j,(m)}^p, \quad j \in J, \\ & \sum_{m=1}^{|S_j^p|} \lambda_{j,m} = 1, \quad j \in J, \\ & \lambda_{j,m} \geqslant 0, \quad j \in J,\ m = 1, \ldots, |S_j^p|, \\ & x \in \mathcal{R}_+. \end{aligned} \tag{41}$$

The global lower bound LB is the optimal value of (41). It must be noted that the global lower bound is obtained through the convexification of the constraints $Tx_j \geqslant DT_{j,(m)}^p$ associated with each set S_j^p, $j \in J$, of terminal p-efficient demand trajectories, and by considering the combination of the convex hulls associated with each terminal $j \in J$. The intersection of these $|J|$ convex hulls does not coincide with the convex hull obtained with the following convexification:

$$Tx \geqslant \sum_{m=1}^{|S^p|} \lambda_m DT_{(m)},$$

$$\sum_{m=1}^{|S^p|} \lambda_m = 1,$$

where S^p is the set of all supply chain p-efficient trajectories that are combinations of terminal p-efficient demand trajectories. This would allow obtaining a tighter lower

Table 2. Preprocessing of p-efficient demand trajectories with the preordered set algorithm.

	$l_{\max}=5$					$l_{\max}=10$				
	$p=0.95\%$			$p=0.97\%$		$p=0.95\%$			$p=0.97\%$	
	$\lvert S_j^0\rvert$	$\lvert S_j^p\rvert$	$\lvert S_j^{po}\rvert$	$\lvert S_j^p\rvert$	$\lvert S_j^{po}\rvert$	$\lvert S_j^0\rvert$	$\lvert S_j^p\rvert$	$\lvert S_j^{po}\rvert$	$\lvert S_j^p\rvert$	$\lvert S_j^{po}\rvert$
Cleveland	5^{12}	13	1	8	2	10^{12}	21	6	10	4
Dartmouth	5^{12}	23	3	14	4	10^{12}	24	7	11	4
Little	5^{10}	26	4	18	3	10^{10}	21	4	8	3
Montreal	5^{12}	86	8	40	8	10^{12}	99	15	44	3
Morrisburg	5^{12}	35	5	16	4	10^{12}	28	10	14	6
Newington	5^{12}	33	8	8	4	10^{12}	20	3	15	4
Oshawa	5^{12}	104	4	53	6	10^{12}	121	12	64	12
Oswego	5^{12}	32	11	16	3	10^{12}	52	10	25	8
Owen	5^{7}	1	1	1	1	10^{7}	6	3	4	1
Quebec	5^{12}	68	5	38	8	10^{12}	58	7	28	10
Sept	5^{8}	6	1	4	1	10^{8}	6	2	3	1
Stephenville	5^{9}	13	4	3	2	10^{9}	10	4	5	3
Thunder	5^{12}	130	11	61	10	10^{12}	141	21	68	11

bound, but would require solving a very large problem containing a huge number of columns corresponding to all possible combinations of terminal p-efficient demand trajectories.

We compute the *global integrality gap* $I^{(\cdot)}$ as follows:

$$I^{(m)}=\frac{\mathrm{UB}^{(m)}-\mathrm{LB}}{\mathrm{LB}},\tag{42}$$

where $\mathrm{UB}^{(m)}$ is the value of the best feasible integer solution for the mixed-integer programming problem (39) associated with the mth p-efficient demand trajectory. It is reported in Table 3 for several stages of the method.

Table 3 details the solution for the case when the demand can take up to five different levels at each period and p is equal to 95%. The first conclusion is the high contribution of the preordered set algorithm used for preprocessing the p-efficient demand trajectories. We have randomly selected 20 p-efficient demand trajectories, and constructed and solved their associated mixed-integer programming problem of form (39). The first row of Table 3 reports the average value of the objective function and the average global integrality gap for these 20 trajectories. The comparison of the first two rows of Table 3 shows the significant reduction in the value of the objective function (Column 2 in Table 3) and in the average global integrality gap (Column 3 of Table 3) brought about by the use of the preordered set algorithm.

Column 4 reports the critical time period t^b. It can be seen that the modification in the delivery schedule for the Montreal terminal relieves the congestion at Period 2 in such a way that, at the next iteration, the critical time period is Period 3. The following changes in the delivery schedule allow the progressive reduction in congestion at Period 3, but leave it as the most congested one. Column 5 reports the terminal j', whose distribution costs can be reduced through the selection of another p-efficient demand trajectory. Column 6 reports the changes in the distribution schedule of that terminal j'. The distribution costs for Montreal, Little, and Oshawa can be successively reduced, through the selection of a more appropriate demand trajectory, without affecting the distribution costs associated

Table 3. Operation of the column-generation method.

	Value of objective function ($)	$I^{(\cdot)}$ (%)	t^b	Terminal considered	Delivery changes
Before preprocessing	5,967,655	7.94			
After preprocessing					
Iteration 1	5,761,665	3.69	2	Montreal	3 trips with ship 1 instead of 4 trips with ship 2
Iteration 2	5,727,380	2.97	3	Little	2 trips with ship 2 instead of 1 with ship 2 and 1 with ship 1
Iteration 3	5,726,610	2.93	3	Oshawa	3 trips with ship 1 and 2 with ship 2 instead of 4 with ship 1 and 1 with ship 2
Iteration 4	5,716,070	2.87	3	Thunder	None

with the other terminals. The recursive improvement in the objective value stops when it is observed that a reduction in the distribution costs for Thunder can only be attained by increasing the distribution costs of the other terminals.

Similar results are obtained for the probability distribution in which the random variables can take 10 different levels. The application of the partially ordered set preprocessing algorithm brings the integrality gap to 4.48%. The algorithm then stops after nine iterations, the integrality gap drops to 3.33%, and the critical time period t^b is the third one at each iteration.

It is evident that the proposed solution methodology performs well for constructing a probabilistic inventory-production-distribution model subject to the satisfaction of a nonstockout cycle service level. It eventually finds feasible integer solutions with a low integrality gap.

4.3.3. Simulation Study. In this section, we perform a simulation study in which, using the solution method described above, we design a new plan every four months. The first plan covers Months 1 to 12, and we implement the decisions related to the first four months. We then construct a new one-year plan covering Months 5 to 12, and we implement its decisions for Months 5 to 8, finally constructing a plan covering Months 9 to 12. We repeat the experiment 100 times to assess the service level attained by the method. We describe below the results obtained for the probability distribution in which the random variables can take five levels and for a nonstockout level of 95%.

Because the plans constructed hedge against most (95%) possibilities of shortfall, the updates allow us to take advantage of favorable realizations of the random demands. This translates into a slightly lower average number of shipments for certain locations (lower distribution costs), into lower average production costs, and into lower average total costs. However, because on one hand the level of the simulated demand is in most cases lower than that against which the plan hedges, and on the other hand this reduction is not necessarily sufficient to reduce the number (and thus quantity) of the deliveries (full truck load shipments) or to switch to a carrier with lower loading capacity, the average ending inventory levels do not always decrease. Table 4 reports the ratio (R_s) of the average number of shipments in the simulation study to the number of shipments calculated initially (at Stage 1), and the ratio (R_e) of the average ending inventory level in the simulation study to the ending inventory level initially planned by the method.

The estimates of the true service levels obtained with our approach are given in Figure 4. The estimates of the expected cost and of the standard deviation of the cost are, respectively, equal to 5,674,750 and 55,966.

Table 4. Terminal shipments and inventory levels in the simulation study, relative to initially planned.

	R_s	R_e
Cleveland	1	0.98
Dartmouth	1	0.98
Little	1	0.99
Montreal	0.96	0.97
Morrisburg	1	1
Newington	1	1.02
Oshawa	1	1
Oswego	0.95	0.99
Owen	0.94	1.02
Quebec	1	1
Sept	1	1.03
Stephenville	1	1.02
Thunder	0.84	0.87
Manufacturer	0.92	0.91

5. Conclusions

For a multistage supply chain evolving in an uncertain environment, we construct a multiperiod inventory-production-distribution plan whose objective is to minimize the total cost of the supply chain, while achieving a prescribed cycle service level. It is a strategic plan that hedges against major risks and uncertainties and can be adjusted for taking advantage of favorable realizations of the random demand.

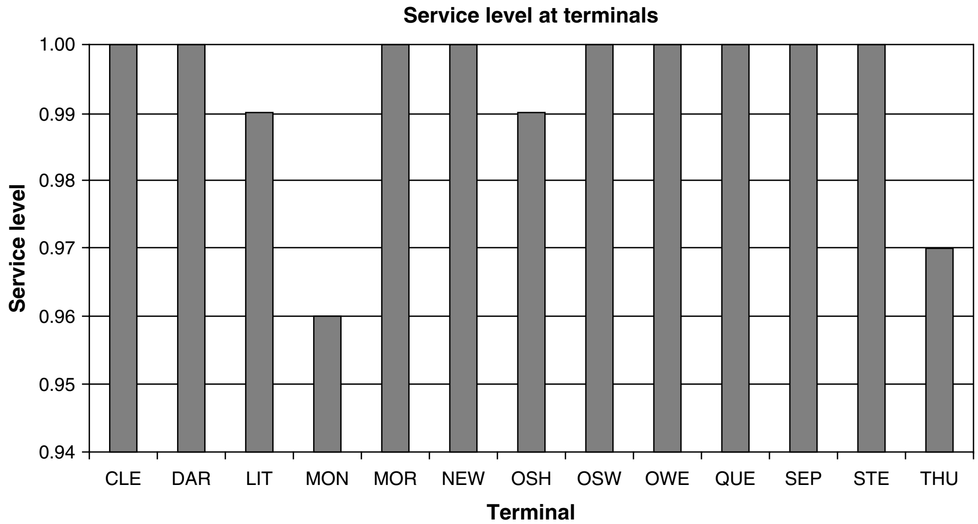
We first develop a large-scale static model formulation of the stochastic programming problem allowing the construction of such a replenishment plan. Formulating the probabilistic constraints in terms of the cumulative demand and supply up to a period t, rather than in terms of the demand and supply at a particular period t, facilitates the development of a tractable solving methodology.

Second, we propose an integrated modular solving methodology for the proposed model. The first module transforms the joint probabilistic constraints using the concept of p-efficiency. We construct terminal or customer p-efficient demand trajectories and supply chain p-efficient demand trajectories, which are combinations of the customer p-efficient demand trajectories.

Simultaneous consideration of all p-efficient demand trajectories and the brute force approach by solving many optimization problems associated with each p-efficient trajectory is not computationally tractable. The concept of p-efficiency is therefore complemented by a preprocessing algorithm, based on the concept of a preordered set. It is used for extracting a subset of promising p-efficient demand trajectories. Its contribution stems from the fact that it significantly reduces the number of p-efficient demand trajectories receiving further consideration, and that the requirements of those demand trajectories can be satisfied at much lower costs.

To solve the resulting disjunctive mixed-integer programming problem, we complement the branch-and-bound algorithm of the CPLEX solver with a congestion-relief column-generation algorithm, which reduces the risk and impact of congestion. The definition of the congestion with respect to the distribution resources results in a less costly distribution schedule anticipating the possible congestion due to the high demand volume in some periods.

Figure 4. Service-level estimates in the simulation study.



The proposed solution approach is validated on an example of a real industrial problem faced by one of the main North American chemical supply chains. The computational results show that the modular methodology proposed allows the finding of a near-optimal solution with a minimal optimality gap, and substantial savings when applied to industrial problems. Ahead-of-time deliveries are carried out, alleviating the impact of the bottleneck of resources of the supply chain, eliminating the need to resort to external logistics providers, and allowing substantial savings. Finally, a simulation study illustrates the potential of the method, when applied in a repetitive fashion, to adjust the plans made at earlier stages.

The potential of the proposed solving methodology is further reinforced by the fact that it does not rely on any independence assumptions on the demand, which can be time dependent or dependent between the nodes.

Appendix

Sets and Indices

We define the following sets and their corresponding indices:

$I = \{1, 2, \ldots, i, \ldots, \iota\}$: set of production facilities.

$J = \{1, 2, \ldots, j, \ldots, \gamma\}$: set of distribution terminals.

$K = I \cup J = \{1, 2, \ldots, k, \ldots, \iota + \gamma\}$: set of supply chain nodes, including facilities and terminals.

$T = \{1, 2, \ldots, t, \ldots, \tau\}$: set of time periods in the planning horizon.

$V = \{1, 2, \ldots, v, \ldots, \upsilon\}$: set of carriers (internal and third-party carriers).

Parameters

The parameters are denoted as follows:

$c[i, k, v]$ = unit shipping cost incurred by using carrier v for a shipment between nodes i and k.

$C[v, t]$ = maximum loading capacity of carrier v at time t.

$l[k, v]$ = estimated loading time of carrier v at node k.

$u[k, v]$ = estimated unloading time of carrier v at node k.

$Z^{\max}[j]$ = maximum storage capacity for end products at terminal j.

$S^{\max}[i]$ = maximum storage capacity for products at facility i.

$W^{\max}[i]$ = maximum storage capacity for raw material at facility i.

$Z^{\min}[j, t]$ = safety-stock requirement for end products at terminal j and time t.

$S^{\min}[i, t]$ = safety-stock requirement for products at facility i and time t.

$W^{\min}[i, t]$ = safety-stock requirement for raw material at facility i and time t.

$a[v, t]$ = availability of carrier v at time t.

$r[i, k, v]$ = traveling time from facility i to location k with carrier v.

$o[i, t]$ = demand for semifinished product at facility i and time t.

$g[i, t]$ = conversion rate for raw material into semifinished product at facility i and time t.

$f[i, t]$ = conversion rate for raw material into end product at facility i and time t.

$r[i, t]$ = conversion rate for semifinished product into end product at facility i and time t.

$h[k, t]$ = unit holding cost at node k and time t.

$p[i, t]$ = unit production cost at facility i and time t.

$b^{\max}[i, t]$ = maximum production capacity of raw material at facility i and time t.

$p^{\max}[i, t]$ = maximum production capacity of product at facility i and time t.

Decision Variables

The decision variables are the following:

$z[j, t]$ = product inventory level at terminal j and time t.

$x[i, k, v, t]$ = number of direct shipments from facility i to node k made at time t with carrier v.

$P[i, t]$ = amount of product produced at facility i and time t.

$s[i, t]$ = product inventory level at facility i at the end of period t.

$w[i, t]$ = raw material inventory level at facility i at the end of period t.

$b[i, t]$ = amount of raw material produced at facility i and time t.

$\delta[v, t]$ = binary variable associated with the availability of carrier v at time t.

$q[i, k, v, t]$ = amount of products delivered to node k at t with a carrier v leaving from facility i.

$m[k, t] = \sum_{i\in I}\sum_{v\in V} q[i, k, v, t]$ = total quantity supplied to location k at time t.

$\omega[k, t]$ = cumulative supply at node k by the end of period t.

Objective Function

The objective function takes the following form:

$$\min \underbrace{\sum_{t\in T}\sum_{i\in I}(p[i,t]\cdot P[i,t])}_{\text{production cost}} + \underbrace{\sum_{t\in T}\sum_{i\in I}\sum_{k\in K,\,k\neq i}\sum_{v\in V}((l[i,v]+2r[i,k,v]+u[v,k])c[i,k,v]\cdot x[i,k,v,t])}_{\text{distribution cost}} + \underbrace{\sum_{t\in T}\sum_{i\in I}\left(h[i,t]\left(\frac{p[i,t]}{2}+s[i,t]\right)\right)+\sum_{t\in T}\sum_{j\in J}(h[j,t]\cdot s[j,t]).}_{\text{inventory cost}}$$

Deterministic Constraints

Flow balance for raw materials at suppliers' facilities.

$$w[i,t]=w[i,t-1]+b[i,t]-\sum_{v\in V}\sum_{i'\in I,\,i'\neq i} q[i,i',v,t]-f[i,t]\cdot P[i,t]-g[i,t]\cdot o[i,t],\quad i\in I,\ t\in T. \tag{A1}$$

The raw material is consumed at the suppliers' and manufacturers' facilities. All the raw material needed by the manufacturers' facilities comes from the suppliers' facilities.

Flow balance for raw materials at manufacturers' facilities.

$$w[i,t]=w[i,t-1]+\sum_{v\in V}\sum_{i'\in I,\,i'\neq i} q[i',i,v,t]-f[i,t]\cdot P[i,t]-g[i,t]\cdot o[i,t],\quad i\in I,\ t\in T. \tag{A2}$$

Flow balance for end products at facilities.

$$s[i,t]=s[i,t-1]\frac{r[i,t-1]}{r[i,t]}+p[i,t]-\sum_{j\in J}\sum_{v\in V} q[i,j,v,t],\quad i\in I,\ t\in T. \tag{A3}$$

Constraints (A1), (A2), and (A3) enforce that the current raw material and product demand at facilities be met from the current production and the on-hand inventory.

Production capacity of raw material.

$$0\leqslant b[i,t]\leqslant b^{\max}[i,t],\quad i\in I,\ t\in T. \tag{A4}$$

Production capacity of products.

$$0\leqslant P[i,t]\leqslant p^{\max}[i],\quad i\in I,\ t\in T. \tag{A5}$$

Constraints (A4) and (A5) are the capacity restriction constraints, limiting the production of raw materials and products below a specified maximum level.

Storage capacity of raw materials at facilities.

$$0\leqslant w[i,t]\leqslant w^{\max}[i],\quad i\in I,\ t\in T. \tag{A6}$$

Storage capacity of products at facilities.

$$0\leqslant s[i,t]\leqslant s^{\max}[i],\quad i\in I,\ t\in T. \tag{A7}$$

Constraints (A6) and (A7) limit from above the inventories of raw materials and products. The nonnegativity restrictions ensure that no backlog occurs.

Operationalization of the distribution scheme.

$$x[i,j,v,t]=0\quad \text{if } j\in J\backslash\{i\} \text{ and } t\in T,\ i\in I,\ v\in V. \tag{A8}$$

The constraints above indicate the impossibility of delivering to a certain supply chain node j' at a given time period t', this resulting, for example, from bad weather conditions or the closing of facilities for some time.

Nonnegativity.

$$z[j,t],P[i,t],s[i,t],w[i,t],b[i,t],q[i,k,v,t],\omega[j,t]\geqslant 0,\quad i\in I, j\in J, t\in T, v\in V, k\in K. \tag{A9}$$

The constraints above enforce the non-negativity restrictions on the decision variables.

Endnote

1. In this paper, due to confidentiality requirements, the numbers representing costs and demand have been rescaled.

Acknowledgments

This research was supported by NSF awards DMS-0303545 and DMI-0354678.

References

Avital, I. 2005. Chance-constrained missile-procurement and deployment models for naval surface warfare. Dissertation, Naval Postgraduate School, Monterey, CA.

Balas, E. 1979. Disjunctive programming. *Ann. Discrete Math.* **5** 3–51.

Barnhart, C., E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, P. H. Vance. 1998. Branch-and-price: Column/generation for solving huge integer programs. *Oper. Res.* **46**(3) 316–329.

Beraldi, P., A. Ruszczyński. 2002. The probabilistic set covering problem. *Oper. Res.* **50** 956–967.

Bitran, G. R., H. H. Yanasse. 1984. Deterministic approximations to stochastic production problems. *Oper. Res.* **32**(5) 999–1018.

Bixby, R. E., M. Fenelon, Z. Gu, E. Rothberg, R. Wunderling. 1999. MIP: Theory and practice: Closing the gap. M. J. D. Powell, S. Scholtes, eds. *System Modeling and Optimization: Methods, Theory and Applications*. Kluwer, Dordrecht, The Netherlands, 19–50.

Charnes, A., W. W. Cooper, G. H. Symonds. 1958. Cost horizons and certainty equivalents: An approach to stochastic programming of heating oil. *Management Sci.* **4** 235–263.

Chen, F. Y., D. Krass. 2001. Inventory models with minimal service level constraints. *Eur. J. Oper. Res.* **134** 120–140.

Cooper, W. W., H. Deng, Z. Huang, S. X. Li. 2004. Chance constrained programming approaches to congestion in stochastic data envelopment analysis. *Eur. J. Oper. Res.* **155** 487–501.

Dentcheva, D., A. Prékopa, A. Ruszczyński. 2000. Concavity and efficient points of discrete distributions in probabilistic programming. *Math. Programming* **89** 55–77.

Dhaenens-Flipo, C., G. Finke. 2001. An integrated model for an industrial production-distribution problem. *IIE Trans.* **33** 705–715.

Färe, R., S. Grosskopf. 2000. Slacks and congestion: A comment. *Socio-Econom. Planning Rev.* **34** 27–33.

Fumero, F., C. Vercellis. 1999. Synchronized development of production, inventory, and distribution schedules. *Transportation Sci.* **33** 330–340.

Graves, S. C., S. P. Willems. 2000. Optimizing strategic safety stock placement in supply chains. *Manufacturing Service Oper. Management* **2**(1) 68–83.

Holmberg, K., H. Tuy. 1999. A production-transportation problem with stochastic demand and concave production costs. *Math. Programming* **85** 157–179.

Kleywegt, A. J., V. S. Nori, W. P. Savelsbergh. 2004. Dynamic programming approximations for a stochastic inventory routing problem. *Transportation Sci.* **38**(1) 42–70.

Kress, M. 2002. *Operational Logistics: The Art and Science of Sustaining Military Operations*. Kluwer Academic Publishers, Boston, MA.

Kress, M., M. Penn, M. Polukarov. 2005. Two-stage supply chain with recourse and probabilistic constraints. Submitted.

Paschalidis, I. C., Y. Liu, C. G. Cassandras, C. Panayiotou. 2004. Inventory control for supply chains with service level constraints: A synergy between large deviations and perturbation analysis. *Ann. Oper. Res.* **126**(1–4) 231–258.

Prékopa, A. 1970. On probabilistic constrained programming. *Proc. Princeton Sympos. Math. Programming*, Princeton University Press, Princeton, NJ, 113–138.

Prékopa, A. 1990. Dual method for a one-stage stochastic programming with random rhs obeying a discrete probability distribution. *Zeitschrift Oper. Res.* **34** 441–461.

Prékopa, A. 1995. *Stochastic Programming*. Kluwer, Boston, MA.

Prékopa, A. 2003. Probabilistic programming models. A. Ruszczyński, A. Shapiro, eds. *Stochastic Programming: Handbook in Operations Research and Management Science*, Vol. 10. Elsevier Science Ltd., Amsterdam, The Netherlands, 267–351.

Prékopa, A., B. Vizvari, T. Badics. 1998. Programming under probabilistic constraint with discrete random variable. F. Giannessi, S. Komlósi, T. Rapcsák, eds. *New Trends in Mathematical Programming*. Kluwer, Boston, MA, 235–255.

Ruszczyński, A., A. Shapiro. 2003. *Stochastic Programming: Handbook in Operations Research and Management Science*, Vol. 10. Elsevier Science Ltd., Amsterdam, The Netherlands.

Santoso, T., S. Ahmed, M. Goetschalckx, A. Shapiro. 2005. A stochastic programming approach for supply chain network design under uncertainty. *Eur. J. Oper. Res.* **167** 96–115.

Sarmiento, A. M., R. Nagi. 1999. A review of integrated analysis of production-distribution systems. *IIE Trans.* **31**(11) 1061–1074.

Sox, C. R., J. A. Muckstadt. 1996. Multi-item, multi-period production planning with uncertain demand. *IIE Trans.* **35**(11) 3023–3042.

Swaminathan, J. M., S. R. Tayur. 2003. Tactical planning models for supply chain management. S. C. Graves, T. de Kok, eds. *OR/MS Handbook on Supply Chain Management: Design, Coordination and Operation*. Elsevier Publishers, Amsterdam, The Netherlands, 423–456.

van der Heijden, M. 2000. Near cost-optimal inventory control policies for divergent networks under fill rate constraints. *Internat. J. Production Econom.* **63**(2) 161–179.

Yildirim, I., B. Tan, F. Karaesmen. 2005. A multiperiod stochastic production planning and sourcing problem with service level constraint. *OR Spectrum* **27** 471–489.