Models and Algorithms for Distribution Problems with Uncertain Demands

Raymond K.-M. Cheung
Department of Industrial and Manufacturing Systems Engineering
Iowa State University
Ames, IA 50011

Warren B. Powell
Department of Civil Engineering and Operations Research
Princeton University
Princeton, NJ 08544

October, 1994 (revised)

Abstract

We consider the problem of distributing goods from one or more plants through a set of warehouses in anticipation of forecasted customer demands. Two results are provided in this paper. First, we present a methodology for approximating stochastic distribution problems that are computationally tractable for problems of realistic size. Comparisons are made to standard deterministic formulations and shown to give superior results. Then, we compare logistics networks with varying degrees of redundancy represented by the number of warehouses which serve each customer. Overlapping service regions for warehouses provides additional flexibility to handle real-time demands. We quantify the expected savings that might result from such strategies.

1 Introduction

Distribution problems involve the allocation of goods or resources to storage areas in anticipation of forecasted market demands. A classic example is the movement of inventory from plant to warehouse in anticipation of future customer demands. Decisions of how much to store at each warehouse must be made prior to knowing what the actual demands are. For example, a major retailer keeps inventories of refrigerators in local warehouses for quick delivery. A customer choosing a refrigerator on Saturday can then schedule delivery the following Monday. Such a fast response requires that inventories be stored in advance of the purchase decision.

Stochastic distribution problems arise in other settings as well. Railroads, motor carriers and shipping companies need to manage large fleets of containers (or trailers or rail cars) to maintain inventories which can meet shipper demands. As containers become empty, they need to be repositioned empty from one storage location (often referred to as trailer pools or container yards) to the next in anticipation of shipper needs. While the terminology is different, the structure of the problem is the same as that faced by manufacturers or retailers.

These problems fall within the framework of multilocation inventory and distribution planning. Static, deterministic models can be found in many logistics books (e.g. Robeson and House [15],Ballou [4], Graves *et al.* [6]). These models focus on determining warehouse size and location, customer allocation to warehouses and transportation planning. Dynamic models consider the operational planning of inventories to meet forecasted demands over a specified planning horizon. Deterministic versions of these models have been in use for some time (see, for example, Klingman and Mote[11]), and have been studied in depth within the research community (Aronson and Chen [3], Aronson [1]), although it is not clear how widely these models have been adopted in practice. One limitation that is often cited is that the models are not able to handle uncertainties in forecasted demands. This calls into question the value of solving models with long planning horizons, such as those posed in Aronson and Chen [2],[3].

Considerably less progress has been made on formulating and solving stochastic, multilocation inventory models. Karmarkar [8],[9], provides bounds and approximations of the expected recourse function for stochastic, (convex) multilocation inventory problems. More recently Shapiro [17] notes that stochastic versions of these problems are computationally intractable. Standard methods based on stochastic dynamic programming suffer from the "curse of dimensionality" and therefore have seen limited practical application. An alternative approach is to formulate a stochastic linear program (Wets [20]) which can then be solved using specialized methods for large scale linear programs (Rockafellar and Wets [16], Lustig *et al.* [12]). It is not clear, however, how many scenarios are required to provide an accurate solution, and problems with even a modest number of scenarios can create extremely large linear programs. While this approach is quite general, it does little to take advantage of the structure of the problem.

In this paper, we formulate the multilocation inventory problem with uncertain demands as dynamic networks. Furthermore, we model the uncertain demands in the problem as random arc capacities. This formulation allows us to take advantage of the recent results by Powell and Cheung ([14], [13]) which present algorithms for approximating the expected recourse function for networks with random arc capacities. This network model is used for dynamic fleet management problems which have been solved as sequences of two-stage stochastic networks by Cheung and Powell[5].

There is a close similarity between classical distribution problems in logistics and fleet management problems. In fleet management, empty vehicles must be allocated in anticipation of future shipper demands. However, vehicles are reusable, and vehicles are "consumed" by allowing the shipper to move the vehicle loaded from one city to the next, at which point the vehicle becomes empty again and available for reuse. By contrast, standard distribution problems move product which is consumed and leaves the network permanently. Both problems can be modeled as networks with random arc capacities, but the characteristics of the networks are different.

This paper makes the following contributions:

1. The multilocation inventory problem with uncertain demand forecasts is formulated as a stochastic programming problem using the framework of dynamic networks with random arc capacities. Two stage and multistage formulations of these problems are proposed. These formulations provide a fresh perspective to the classical multilocation inventory problem and may leading to new classes of solution techniques.
2. We investigate how the methods developed by Powell and Cheung ([14], [13]) for fleet management problems can be adapted in the context of multilocation inventory planning. In particular, we show that the two-stage stochastic distribution problem can be solved

easily using standard network algorithms.

3. Using a two-stage model, we evaluate the effectiveness of using overlapping service areas for distribution problems. These experiments allow us to test the hypothesis that a spatially distributed set of inventories can work as effectively as a single inventory as long as the fraction of customers served by more than one warehouse exceeds a certain percentage.

Section 2 formulates the distribution problem as a two-stage stochastic program. In this section, we show how the methods developed in [14] and [13] can be applied to find an approximate solution. Next, section 3 considers a dynamic version of the distribution problem and shows how a backward recursion can be used to solve the problem. Finally, section 4 uses the two-stage formulation first to test the quality of the approximation, and second to evaluate the effectiveness of flexible distribution strategies.

2 Two-stage distribution planning

One of the most fundamental models for discrete time stochastic multilocation inventory problem is the newsboy model. In our context, the newsboy model considers first the flow of goods from plants to customers and second the costs associated with the overage and the underage in meeting the customer demands. The objective of this model is to minimize the transportation cost and the expected underage costs and overage costs. This model can be formulated as a stochastic program with simple recourse and thus can be solved by using classical nonlinear programming techniques.

However, in some situations where consolidation facilities are involved (warehouses in our case), the newsboy model may not apply. Consider a two-stage decision process: (1) we must ship goods from plants to warehouses before customer demands are realized; (2) only after we know the exact customer demands, we then ship the goods from warehouses to customers. In other words, after the customer demands are realized, when a warehouse do not have enough inventory to meet the demand for a customer, the unsatisfied demand may be met by the shipment from another warehouse. The newsboy model does not capture these recourse actions.

We consider the multilocation inventory problem as a two-stage stochastic network. The first stage involves the shipments from plants to warehouses whereas the second stage involves the shipments from warehouses to customers. The challenge in these problems is that customer demands are not known with certainty. Therefore, decisions in stage one must be made before customer demands are revealed. When customer demands become available, the second stage decisions are then made. The objective is to find the best distribution plan, on "average," to ship products from plants to warehouses while hedging against uncertain customer demands. Models as such can be generalized in a multi-echelon setting(Zangwill[22]). The resulting model can be applied to other problems such as production planning (Karmarkar[7, 8, 9] and Karmarkar and Patel[10]).

2.1 Problem formulation

Let ξ be a random vector defined over a probability space $(\Omega, \mathcal{F}, P)$ where Ω is the set of elementary outcomes ω, $\mathcal{F}$ is the event space and P is the probability measure. We have the following notation:

Deterministic parameters:

$\mathcal{P}$ = set of indexes representing plants, with $i \in \mathcal{P}$;

$\mathcal{W}$ = set of indexes representing warehouses, with $j \in \mathcal{W}$;

$\mathcal{C}$ = set of indexes representing customers, with $k \in \mathcal{C}$;

c_{ij} = cost of shipping a unit of product from plant i to warehouse j;

q_{jk} = cost of shipping a unit of product from warehouse j to customer k;

r_k = penalty cost per unit of unsatisfied demand for customer k;

R_i = amount of goods produced at plant i;

u_{ij} = capacity of shipment from plant i to warehouse j;

u_{jk} = capacity of shipment from warehouse j to customer k;

Decision variables:

x_{ij} = amount of goods shipped from plant i to warehouse j;

y_{jk} = amount shipped from warehouse j to customer k;

s_j = amount of products available at warehouse j;

$= \sum_{i \in \mathcal{P}} x_{ij}$

z_k = amount of goods received by customer k;

$$= \sum_{j \in \mathcal{W}} y_{jk}$$

Random parameters:

ξ_k = random demand of customer k;

$\xi_k(\omega)$ = a realization of the demand of customer k;

Note that s_j and z_k are simply definitional variables which give the total flow into a warehouse or customer.

We assume the customer demands ξ_k to be independent, discrete, and finite random variables. Furthermore, we assume that no backlogging is allowed to meet the unsatisfied demands. This assumption is reasonable for the distribution planning of very short-cycle products, such as fresh vegetables. In this situation, the lost revenue for the unsatisfied demands are represented by the penalty cost, z_k. We feel that this model will also serve an approximation of multistage distribution problems with backlogging, where our underage cost can be used to approximate the value of refusing a demand now (with the possibility of being satisfied later). Today, transportation companies often use a one period transportation problem to solve the equipment repositioning problem, which is just a deterministic version of our model. In this sense, we are extending, in a computationally tractable fashion, the state of the art in this problem area.

In the following, we take the convention that a variable with no subscript represents a vector. Furthermore, we let $x(\omega)$, $y(\omega)$ and $z(\omega)$ be the vectors of decision variables for a particular realization of customer demand $\xi(\omega)$. Assuming zero inventory at warehouses, the two-stage formulation of a static distribution network can be written as follows. The stage 1 problem is given by:

$$\min_x \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{W}} c_{ij} x_{ij} + \bar{Q}(s) \tag{1}$$

subject to:

$$\begin{aligned} \sum_{j \in \mathcal{W}} x_{ij} &= R_i \qquad \forall i \in \mathcal{P} \\ \sum_{i \in \mathcal{P}} x_{ij} &= s_j \qquad \forall j \in \mathcal{W} \\ x_{ij} &\leq u_{ij} \qquad \forall i \in \mathcal{P},\ j \in \mathcal{W} \end{aligned} \tag{2}$$

where $\bar{Q}(s) = E_\omega Q(s, \xi(\omega))$ is the expected cost of the following stage 2 problem. The function $\bar{Q}(s)$ is also known as the expected recourse function. For a given realization $\xi(\omega)$ and a fixed vector s, the total cost of the stage 2 problem is obtained by solving a minimization problem:

$$Q(s,\xi(\omega)) \;=\; \min_{y(\omega)} \sum_{j\in\mathcal{W}}\sum_{k\in\mathcal{C}} q_{jk}y_{jk}(\omega) + \sum_{k\in\mathcal{C}} r_k(\xi_k(\omega) - z_k(\omega)) \tag{3}$$

subject to:

$$\begin{aligned}
\sum_{k\in\mathcal{C}} y_{jk}(\omega) & & = \;& s_j & \forall j \in \mathcal{W} \\
\sum_{j\in\mathcal{W}} y_{jk}(\omega) & \; -z_k(\omega) & = \;& 0 & \forall k \in \mathcal{C} \\
y_{jk}(\omega) & & \leq \;& u_{jk} & \forall j \in \mathcal{W},\; k \in \mathcal{C} \\
& z_k(\omega) & \leq \;& \xi_k(\omega) & \forall k \in \mathcal{C}
\end{aligned} \tag{4}$$

In general, suppliers will not ship more than what customers ask. We represent this situation by $z_k(\omega) \leq \xi_k(\omega)$ (the last set of constraints in (4)). When the demand for customer k is not met, then a penalty cost of r_k will be incurred. As a result, in the objective function of the stage 2 problem, we have the term $\sum_{k\in\mathcal{C}} r_k(\xi_k(\omega) - z_k(\omega))$ representing the total penalty cost for unsatisfied demands.

Let us consider the objective function (3). Notice that $\xi_k(\omega)$ is not a decision variable and thus can be taken away from the objective function from the optimization point of view. Therefore, we can replace (3) with

$$Q(s,\xi(\omega)) \;=\; \min_{y(\omega)} \sum_{j\in\mathcal{W}}\sum_{k\in\mathcal{C}} q_{jk}y_{jk}(\omega) - \sum_{k\in\mathcal{C}} r_k z_k(\omega) \tag{5}$$

The revised problem is a two-stage network where all parameters in stage 1 are deterministic and some arc capacities in stage 2 are random variables (representing the customer demands). Such a two-stage network is depicted in Figure 1. Note that in Figure 1, we also have a dummy customer. There is an arc joining each warehouse to the dummy customer where the arc flow is simply the excess supply to the warehouse and the arc cost is the inventory cost at this warehouse. However, for simplicity, we do not explicitly distinguish this dummy customer with other customers in our mathematical formulation.

2.2 Basic solution approach

Our ability to solve the two-stage stochastic problem depends on our ability to approximate the expected recourse function $\bar{Q}(s)$ in (1). Due to the embedded minimization in expectation,

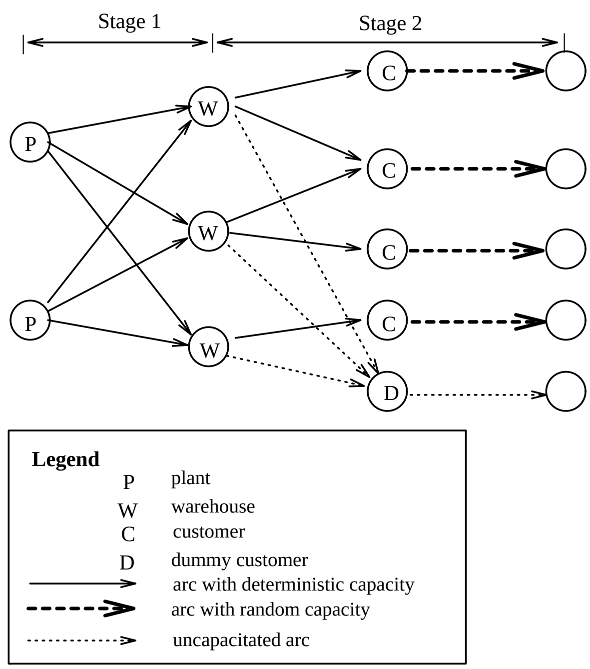


Figure 1: Two-stage stochastic network formulation of static distribution planning.

obtaining the exact expected recourse function is generally intractable. However, the special structure of our problem enables the development of specialized techniques. Powell and Cheung [14] introduce a procedure to compute $\bar{Q}(s)$ exactly when the stage 2 problem consists of trees with random arc capacities. Moreover, Powell and Cheung [13] develop a method to approximate $\bar{Q}(s)$ when the stage 2 problem is a network with random arc capacities. The application of these techniques has appeared in [5].

The main idea of these methods is to replace the complicated expected recourse function $\bar{Q}(s)$ by a tractable function $\hat{Q}(s)$ which is convex, piecewise linear and separable in s, that is

$$\hat{Q}(s) = \sum_{j \in \mathcal{W}} \hat{Q}_j(s_j)$$

The convexity and the piecewise linearity are the properties of the expected recourse function (see Wets[21]), while we impose an approximation of separability to simplify our calculations.

Each component $\hat{Q}_j(s_j)$ of $\hat{Q}(s)$ has an intuitive interpretation: it measures the expected marginal value for each unit of goods available in warehouse j. Since $\bar{Q}(s)$ is convex and piecewise linear, we would like the components $\hat{Q}_j(s_j)$ to be convex and piecewise linear as well,

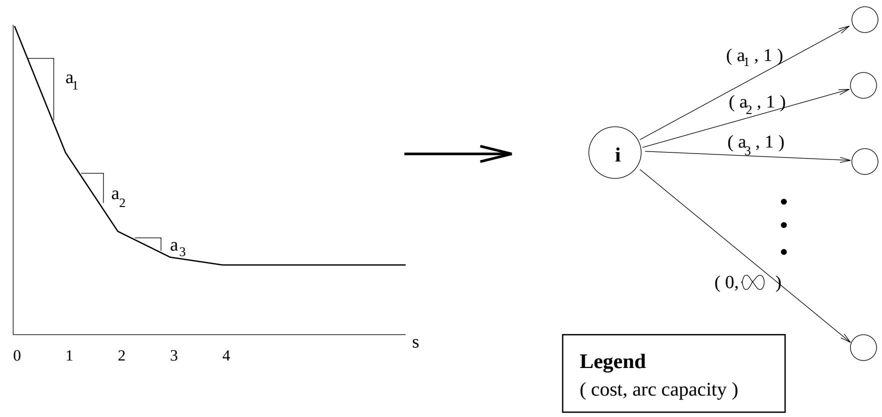


Figure 2: Representation of a convex, piecewise linear function by a set of links.

reflecting the diminishing return of each incremental unit of available goods. As illustrated in Figure 2, each piecewise linear, convex function $\hat{Q}_j(s_j)$ can be represented by a set of deterministic links, which we refer to as the *recourse links*. Except for the last link which is uncapacitated to ensure the feasibility, all links have arc capacities of one unit. The cost of the first link is the slope of $\hat{Q}_j(1)$ and the cost of the second link is the slope of $\hat{Q}_j(2)$ and so on. These links capture the impact of the stage 1 decisions in stage 2. We then augment the stage 1 problem by adding these links to the terminal nodes (representing warehouses). The resulting augmented stage 1 problem is a pure network which can be solved by standard network flow algorithms efficiently. Figure 3 shows the augmented stage 1 problem of the network in Figure 1.

Notice that this approach captures the future interaction of activities as simple functions, allowing the stage 1 decisions to be made in real time. Furthermore, the network structure of the augmented stage 1 problem naturally leads to an integer solution as a minimum cost flow problem which is desirable in many applications.

The core of this approach is to obtain the separable function $\hat{Q}(s)$. In the following, we focus on two methods for obtaining $\hat{Q}(s)$. Section 2.3 describes a solution method for the special class of problems with a single warehouse per customer. This method takes advantage of the problem structure and allows us to find $\bar{Q}(s)$ exactly. Then, section 2.4 describes a method

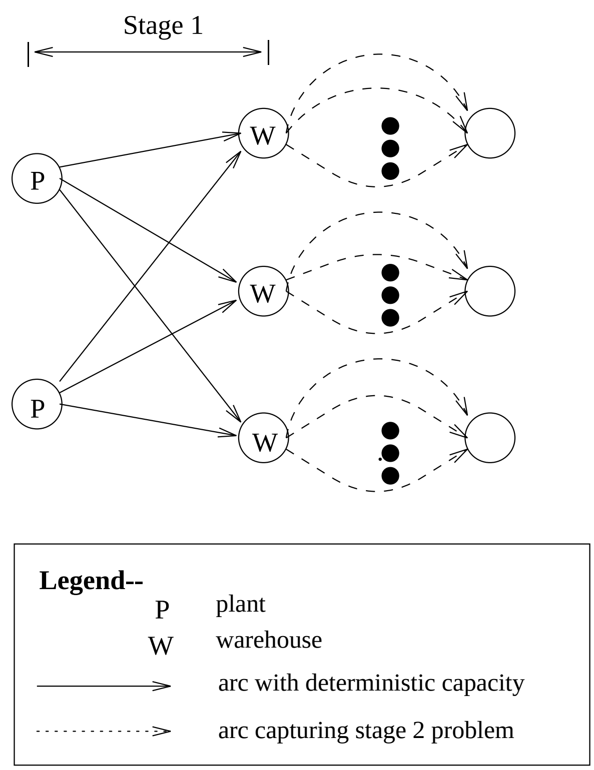


Figure 3: Two-stage stochastic network formulation of static distribution planning.

for problems with several warehouses per customer. This method produces an approximation solution which can be iteratively improved.

2.3 Tree problems

This section considers a special class of two-stage distribution planning problems where each customer receives goods from only one warehouse. In such a problem, the stage 2 problem consists of a set of trees where each tree is rooted at a node representing warehouse. This root node is the only entry point for flow entering the tree. Links in the trees have deterministic cost coefficients but may have random arc capacities. The spatial separability of this problem makes the expected recourse function separable, that is,

$$\bar{Q}(s) = \sum_{j \in \mathcal{W}} \bar{Q}_j(s_j) \tag{6}$$

where $\bar{Q}_j(s_j)$ is the expected total cost of the tree rooted at $j \in \mathcal{W}$ where the amount of goods available at warehouse j is s_j.

The separability of the recourse function together with the special structure of trees allow us to compute each component $\bar{Q}_j(s_j)$ exactly. Powell and Cheung[14] develop an algorithm to obtain the expected recourse function for trees with random arc capacities. Our recourse problem is simply the special case of two-level trees where arc capacities in the first level are deterministic and arc capacities in the second level are random. Thus, the method of [14] can be applied. Below, we show how to obtain the expected recourse function for a tree rooted at node j parametrically as a function of the supply to the root node.

One mathematical way to express a tree is the path-flow formulation where a path k is defined as a sequence of arcs joining the root node j to a customer k. Since we are now focusing on a particular tree, for simplicity, we suppress the index j unless otherwise specified. Define

$\mathcal{P}$ = the set of all paths,

N_p = $|\mathcal{P}|$,

c_k = cost of the path for customer k,

$\phi(l,k)$ = probability that the l^{th} unit of goods available in the warehouse j is shipped to customer k,

$\mu(l)$ = expected marginal cost for the l^{th} unit of goods available at warehouse j.

Following the definition of $\phi(l,k)$ and $\mu(l)$, we know that the expected recourse function $\bar{Q}_j(s_j)$ can be obtained by first calculating $\mu(l)$ for any l:

$$\mu(l) = \sum_{k=1}^{N_p} c_k \cdot \phi(l,k) \tag{7}$$

and then summing up all $\mu(l)$ for $l = 1, 2, ..., s_j$:

$$\bar{Q}_j(s_j) = \sum_{l=1}^{s_j} \mu(l) = \sum_{l=1}^{s_j} \sum_{k=1}^{N_p} c_k \cdot \phi(l,k) \tag{8}$$

Equations (7) and (8) imply that knowing the probabilities $\phi(l,k)$ is sufficient to obtain $\bar{Q}_j(s_j)$. Therefore, in the remainder of this section, we concentrate on the calculation of $\phi(l,k)$.

Let, for a given realization $\xi(\omega)$:

$\psi_k(\omega)$ = capacity of path k under realization $\xi(\omega)$,

$Z^k(\omega)$ = the total capacity of the first k paths under realization $\xi(\omega)$.

By definition, we have

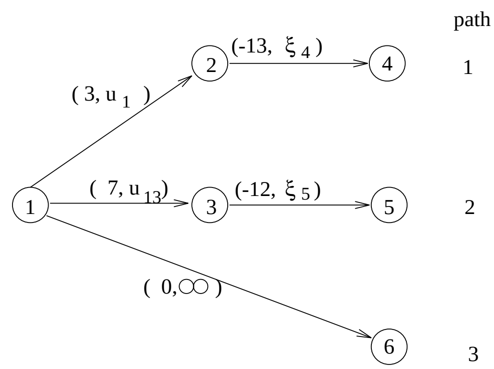
$$Z^k(\omega) = \sum_{m=1}^{k} \psi_m(\omega). \tag{9}$$

Without loss of generality, assume that all paths are ranked according to their cost, from the least to the most, that is

$$c_1 \leq c_2 \leq \cdots \leq c_k \leq \cdots \leq c_{N_p}$$

Powell and Cheung[14] shows that an optimal strategy for assigning flow to paths is to put as much flow as possible on the lowest cost paths. Equivalently, if the paths are ranked in the order of their cost, then the l^{th} unit of goods will be shipped over path k if and only if the total capacity of the first $k-1$ paths is less than l and the total capacity of the first k paths is at least l. Therefore, the probabilities $\phi(l,k)$ can be characterized as:

$$\begin{aligned} \phi(l,k) &= P\{Z^k \geq l \cap Z^{k-1} < l\} & (10) \\ &= P\{Z^k \geq l\} - P\{Z^{k-1} \geq l\}. & (11) \end{aligned}$$



Path costs:
$c_1 = -10$, $c_2 = -5$ and $c_3 = 0$

Deterministic arc capacities:
$u_{1,2} = 2$, $u_{1,3} = 1$ and $u_{1,6} = \infty$

Random arc capacities:
The probability distributions are:

$$P\{\xi_4 = k\} = \begin{cases} 0.1 & \text{if } k = 0 \\ 0.2 & \text{if } k = 1 \\ 0.3 & \text{if } k = 2 \\ 0.4 & \text{if } k = 3 \end{cases} \qquad P\{\xi_5 = k\} = \begin{cases} 0.3 & \text{if } k = 0 \\ 0.5 & \text{if } k = 1 \\ 0.2 & \text{if } k = 2 \end{cases}$$

Figure 4: An example of a tree recourse problem.

Notice that the path capacity ψ_k is simply a truncation of ξ_k:

$$\psi_k = \min\{u_{jk}, \xi_k\} \tag{12}$$

Since we assume that ξ_k are independent, ψ_k are also independent. Consequently, the random variable Z^k is a sum of independent random variables. Hence, we can obtain the distribution of Z^k by using convolution of independent random variables.

To illustrate the idea, consider the example in figure 4. First, we compute the distributions of the path capacities Z^k, $k = 1, 2, 3$. Since $\psi_1 = \min\{u_{1,2}, \xi_4\}$, ψ_1 is equivalent to truncating ξ_4 at 2. Similarly, ψ_2 is equivalent to truncating ξ_5 at 1. Thus, the probability distributions of ψ_k, $k = 1, 2$, are:

$$P\{\psi_1 = l\} = \begin{cases} 0.1 & \text{if } l = 0 \\ 0.2 & \text{if } l = 1 \\ 0.7 & \text{if } l = 2 \end{cases} \qquad P\{\psi_2 = l\} = \begin{cases} 0.3 & \text{if } l = 0 \\ 0.7 & \text{if } l = 1 \end{cases}$$

Whereas, the last path is uncapacitated, that is, $P\{\psi_3 = \infty\} = 1$. By using convolution (see equation (9)), the probability distributions of Z^k, $k = 1, 2$, are:

$$P\{Z^1 = l\} = \begin{cases} 0.1 & \text{if } l = 0 \\ 0.2 & \text{if } l = 1 \\ 0.7 & \text{if } l = 2 \end{cases} \qquad P\{Z^2 = l\} = \begin{cases} 0.03 & \text{if } l = 0 \\ 0.13 & \text{if } l = 1 \\ 0.35 & \text{if } l = 2 \\ 0.49 & \text{if } l = 3 \end{cases}$$

and $P\{Z^3 = \infty\} = 1$. With equation (11), the values of $\phi(l, k)$ can be obtained as:

$\phi(l,k)$	k=1	2	3
$l = 1$	0.90	0.07	0.03
2	0.70	0.14	0.16
3	0.00	0.49	0.51
4	0.00	0.00	1.00

Thus, by using (7) and (8) , the expected marginal values $\mu(l)$ and the expected recourse function are:

k	$\mu(k)$	$\bar{Q}_j(k)$
1	9.35	9.35
2	7.70	17.05
3	2.45	19.50
≥ 4	0.00	19.50

The above procedure is quite efficient. The numerical experiments in [14] show that the exact expected recourse functions for trees with more than one thousand random arc capacities can be found in a few seconds.

2.4 Network recourse problems

In the previous section, we assume that each customer receives goods from only one warehouse. This assumption produces the tree-structured distribution planning problem which can be solved exactly. Let us now consider a more general case where each customer can receive goods from several warehouses. In this case, the stage 2 problem is no longer separable. This problem is known as a special class of stochastic programs with network recourse (see Wallace [19]). Except for very small problems, obtaining exact expected recourse functions $\bar{Q}(s)$ are numerically intractable.

Extending the technique for solving tree problems to solving network recourse problems, Powell and Cheung[13] introduce a decomposition approach called the *network recourse decomposition method* (NRD). Instead of obtaining the exact expected recourse function, this method seeks to obtain a convex, separable approximation of the expected recourse function. The method involves decomposing the stage 2 network into a set of trees whose expected recourse function can be obtained using the method described in the previous section.

Recall that our basic approach is to approximate the expected recourse function by a separable function of s (the supply to warehouses). Thus, we decompose the stage 2 problem by warehouses so that each component of the separable function can be obtained individually. This leads to the notion of a multicommodity formulation of problem (3) – (4). Let us define the goods available at warehouse j as the *commodity* of type $t(j)$. Let

$y_{jk}^{t(j)}$ = amount of commodity $t(j)$ being shipped from warehouse j to customer k
$z_k^{t(j)}$ = amount of commodity $t(j)$ received by customer k

Furthermore, for an outcome ω, let $\mathcal{T}_{t(j)}(\omega)$ be the set of $\{y_{jk}^{t(j)}(\omega), z_k^{t(j)}(\omega)\}$ satisfying the conditions:

$$\left\{ \begin{array}{llcll} \sum_{k \in \mathcal{C}} y_{jk}^{t(j)}(\omega) & & = & s_j & \forall j \in \mathcal{W} \\ \sum_{j \in \mathcal{W}} y_{jk(\omega)}^{t(j)} & -z_k(\omega) & = & 0 & \forall k \in \mathcal{C} \\ y_{jk}^{t(j)}(\omega) & & \leq & u_{jk} & \forall j \in \mathcal{W},\ k \in \mathcal{C} \\ & z_k^{t(j)}(\omega) & \leq & \xi_k(\omega) & \forall k \in \mathcal{C} \end{array} \right\} \tag{13}$$

In other words, (13) represents the constraint set for each commodity $t(j)$ with the bundle constraints:

$$z_k(\omega) \leq \xi_k(\omega) \quad \forall k \in \mathcal{C}$$

replaced by the looser constraints

$$z_k^{t(j)}(\omega) \leq \xi_k(\omega) \quad \forall k \in \mathcal{C}$$

With such definitions and notation, the recourse problem (3) – (4) can be rewritten as a multi-commodity problem:

$$Q(s, \xi(\omega)) \;=\; \min_{y(\omega)} \sum_{j \in \mathcal{W}} \sum_{k \in \mathcal{C}} q_{jk} y_{jk}^{t(j)}(\omega) - \sum_{k \in \mathcal{C}} r_k \sum_{j \in \mathcal{W}} z_k^{t(j)}(\omega) \tag{14}$$

subject to:

$$\{y^{t(j)}(\omega), z^{t(j)}(\omega)\} \;\in\; \mathcal{T}_{t(j)}(\omega) \;\; \forall j \in \mathcal{W} \tag{15}$$

$$\sum_j z_k^{t(j)}(\omega) \;\leq\; \xi_k(\omega) \quad \forall k \in \mathcal{C} \tag{16}$$

Clearly, problem (14) – (16) is separable up to the bundle constraints (16). Moreover, each set of constraints (15) defines a tree rooted at a node j which represents a warehouse. Therefore, without constraints (16), the expected value of $Q(s, \xi(\omega))$ can be obtained parametrically (with respect to s) using the technique described in the previous section.

A natural strategy to decouple the bundle constraints (16) is using relaxation. Assume that for a fixed vector s and a realization $\xi(\omega)$, we relax the bundle constraints (16) through a vector of penalties λ. Consequently, problem (14) – (16) can be relaxed to:

$$\begin{aligned} &\mathcal{L}(s, \lambda, \omega) \\ &= \min_{\{y^{t(j)}, z^{t(j)}\} \in \mathcal{T}_{t(j)}} \sum_{j \in \mathcal{W}} \sum_{k \in \mathcal{C}} \left(q_{jk} y_{jk}^{t(j)}(\omega) - r_k z_k^{t(j)}(\omega) \right) + \sum_{k \in \mathcal{C}} \lambda_k^T \left(\sum_{j \in \mathcal{W}} z_k^{t(j)}(\omega) - \xi_k(\omega) \right) \end{aligned} \tag{17}$$

Rearranging the terms, we have

$$\mathcal{L}(s, \lambda, \omega) = \sum_{j \in \mathcal{W}} \min_{\{y^{t(j)}, z^{t(j)}\} \in \mathcal{T}_{t(j)}} \sum_{k \in \mathcal{C}} \left(q_{jk} y_{jk}^{t(j)}(\omega) + (-r_k + \lambda_k) z_k^{t(j)}(\omega) \right) - \lambda^T \xi(\omega) \tag{18}$$

When we take the expectations of both sides of (18), we have

$$E_\omega \mathcal{L}(s, \lambda, \omega) = E_\omega \left\{ \sum_{j \in \mathcal{W}} \min_{\{y^{t(j)}, z^{t(j)}\} \in \mathcal{T}_{t(j)}} \sum_{k \in \mathcal{C}} \left(q_{jk} y_{jk}^{t(j)}(\omega) + (-r_k + \lambda_k) z_k^{t(j)}(\omega) \right) - \lambda^T \xi(\omega) \right\} \tag{19}$$

Interchanging expectation and summation, we get

$$
\begin{aligned}
& E_\omega \mathcal{L}(s, \lambda, \omega) \\
&= \sum_{j \in \mathcal{W}} E_\omega \left\{ \min_{\{y^{t(j)}, z^{t(j)}\} \in \mathcal{T}_{t(j)}} \sum_{k \in \mathcal{C}} \left(q_{jk} y_{jk}^{t(j)}(\omega) + (-r_k + \lambda_k) z_k^{t(j)}(\omega) \right) \right\} - \lambda^T E_\omega \xi(\omega) \qquad (20) \\
&= \sum_{j \in \mathcal{W}} \hat{Q}_j(s_j, \lambda) - \lambda^T E_\omega \xi(\omega) \qquad (21)
\end{aligned}
$$

where

$$
\hat{Q}_j(s_j, \lambda) = E_\omega \left\{ \min_{\{y^{t(j)}, z^{t(j)}\} \in \mathcal{T}_{t(j)}} \sum_{k \in \mathcal{C}} \left(q_{jk} y_{jk}^{t(j)}(\omega) + (-r_k + \lambda_k) z_k^{t(j)}(\omega) \right) \right\} \qquad (22)
$$

Notice that each component $\hat{Q}_j(s_j, \lambda)$ is simply the expected recourse function of the tree associated by commodity $t(j)$ where the link costs are modified by λ. λ puts a penalty on a customer when too many warehouses are trying to satisfy the same demand. Since $\mathcal{L}(s, \lambda, \omega)$ is a relaxed problem of $Q(s, \omega)$, we know that

$$
\mathcal{L}(s, \lambda, \omega) \leq Q(s, \omega)
$$

for any value of s, λ and ω. Thus, this inequality is still true in expectation, meaning that

$$
\hat{Q}(s, \lambda) = E_\omega \mathcal{L}(s, \lambda, \omega) \leq E_\omega Q(s, \omega) = \bar{Q}(s) \qquad (23)
$$

Therefore, the relaxation procedure produces a lower bound, $\hat{Q}(s)$, of $\bar{Q}(s)$. This result suggests that a tighter lower bound can be obtained by using a λ^* in (22) (instead of an arbitrary λ) where

$$
\lambda^* = \arg\max_{\lambda} \hat{Q}(s, \lambda) \qquad (24)
$$

The parameter λ^* can be obtained by using the standard subgradient method (see, for example, Shor [18]). Notice that $\hat{Q}(s, \lambda)$ consists of a separable term $\sum_j \hat{Q}_j(s_j, \lambda)$ and a constant term $\lambda^T E_\omega \xi(\omega)$, where the constant can be dropped without altering the decisions being made.

Although this method can provide a lower bound of the expected recourse function, the main objective of this method is to capture the shape of the expected recourse function by a set of convex, piecewise linear functions $\hat{Q}_j(s_j, \lambda)$. When we represent these functions by sets of recourse links and add these links to the terminal nodes in stage 1, the augmented stage 1 problem is a pure network problem. For the purpose of capturing the shape, numerical experiments in [13] suggests that problem (24) not need be solved exactly; a few iterations of the subgradient method can produce a reasonably good approximation.

3 Multi-stage distribution planning

In contrast to static distribution problems where decisions are made once after realizing the random coefficients, dynamic distribution problems require decisions to be made over time. Consider a distribution planning problem with an N-stage planning horizon where a *stage* is defined as a period of time during which the customer demands are realized at once and where decisions must be made prior to realization of future demands. In this problem, decisions in stage 1 are made before the demands in stage 2 are revealed. Decisions in stage 2 are made before the uncertain demands in stage 3 are revealed. Thus, we can formulate these problems as multistage stochastic programming problems with recourse. In the following, we provide the formulation and describe a solution methodology.

3.1 Formulation

In a multi-stage distribution planning problem, an elementary outcome $\omega \in \Omega$ consists of a set of outcomes $\omega_2, \ldots, \omega_t, \ldots, \omega_N$ where ω_t represents an outcome in stage t. We denote by $\xi(t)$ the vector of customer demands in stage t. We add the time dimension to all previously defined coefficients and variables to reflect the time dependency. For example, $c_{ij}(t)$ is now interpreted as the cost of shipping a unit of product from plant i to warehouse j in stage t and $y_{jk}(t, \omega_t)$ denotes the amount of products being shipped from warehouse j to customer k in stage t when the realization of customer demands is $\xi(t, \omega_t)$. To ensure feasibility, we assume all unshipped products from a plant or a warehouse can be held at their current locations as inventory for the next stage. Let

$x_{ii}(t)$ = inventory in plant i in stage t;

$c_{ii}(t)$ = holding cost per unit of product in plant i in stage t;

$y_{jj}(t)$ = inventory in warehouse j in stage t;

$q_{jj}(t)$ = holding cost per unit of product in warehouse j in stage t;

Figure 5 illustrates a dynamic distribution problem in the time-space framework.

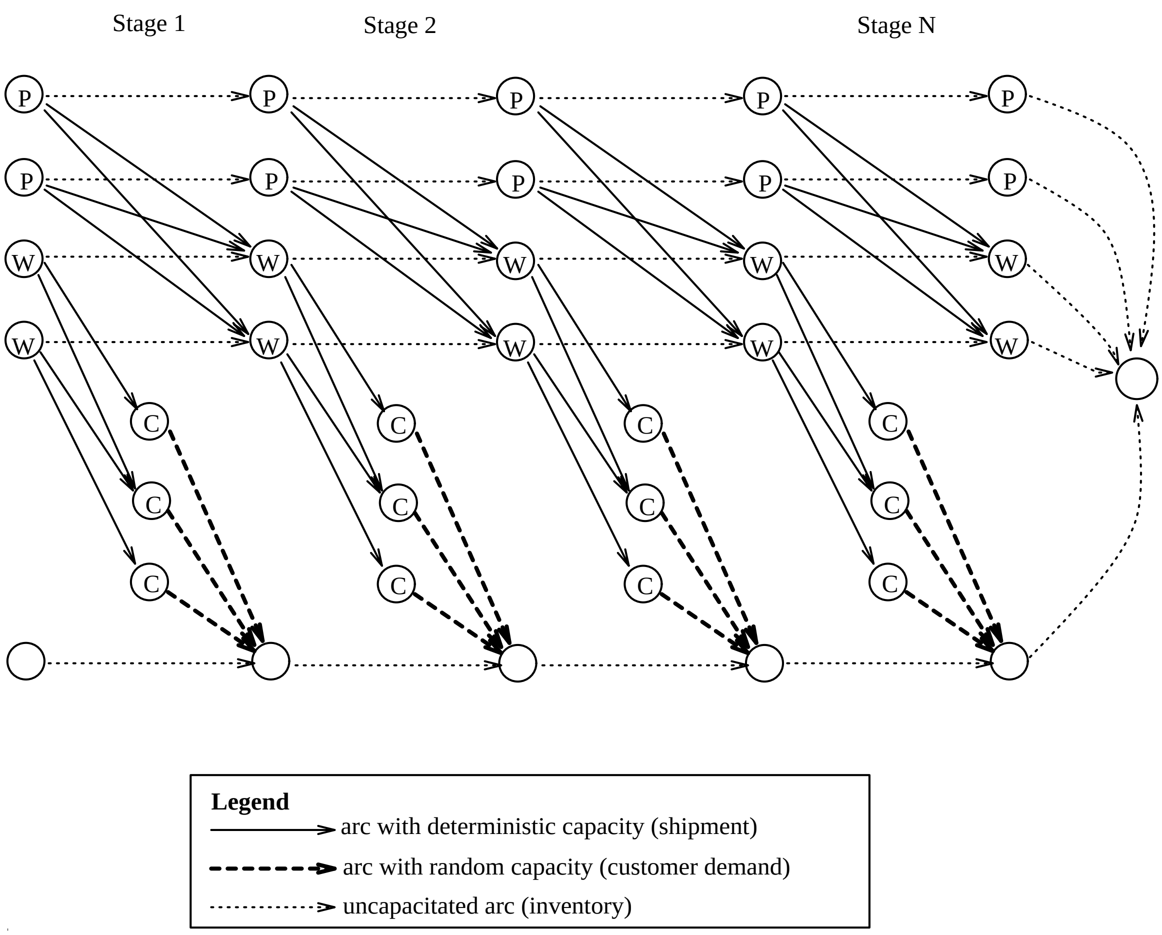


Figure 5: Multi-stage stochastic network formulation of dynamic distribution planning.

For simplicity, let $\hat{s}(t)$ be a state vector representing the inventory level of the plants and warehouses, that is,

$$\hat{s}(t) = [x_{11}(t), \ldots, x_{ii}(t), \ldots, x_{|\mathcal{C}||\mathcal{C}|}(t), s_1(t), \ldots, s_j(t), \ldots, s_{|\mathcal{W}|}(t)]$$

Assuming there is no inventory in warehouses at the beginning, the stage 1 problem is given by:

$$\min_{x} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{W}} c_{ij}(1) x_{ij}(1) + \bar{Q}_2(\hat{s}(1)) \tag{25}$$

subject to:

$$\begin{aligned} \sum_{j \in \mathcal{W}} x_{ij}(1) \quad + x_{ii}(1) &= R_i(1) && \forall i \in \mathcal{P} \\ \sum_{i \in \mathcal{P}} x_{ij}(1) &= s_j(1) && \forall j \in \mathcal{W} \\ x_{ij}(1) &\leq u_{ij}(1) && \forall i \in \mathcal{P},\ j \in \mathcal{W} \end{aligned} \tag{26}$$

where $R_i(1)$ is the planned production level at plant i.

The expected recourse function $\bar{Q}_2(\hat{s}(1)) = E_{\omega_2} Q_2(\hat{s}(1), \omega_2)$ is recursively defined as follows. For a given realization of demands $\xi(t, \omega_t)$, the stage t problem is given by:

$$\begin{aligned} &Q_t(\hat{s}(t-1), \omega_t) = \\ &\min_{x,y,z} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{W}} c_{ij}(t) x_{ij}(t) + \sum_{j \in \mathcal{W}} \sum_{k \in \mathcal{C}} q_{jk}(t) y_{jk}(t, \omega_t) - \sum_{k \in \mathcal{C}} r_k(t) z_k(t, \omega_t) + \bar{Q}_t(\hat{s}(t), \omega_{t+1}) \end{aligned} \tag{27}$$

subject to:

$$\begin{aligned} \sum_{j \in \mathcal{W}} x_{ij}(t) + x_{ii}(t) &= R_i(t) + x_{ii}(t-1) && \forall i \in \mathcal{P} \\ \sum_{k \in \mathcal{C}} y_{jk}(t, \omega_t) &= s_j(t-1) && \forall j \in \mathcal{W} \\ \sum_{i \in \mathcal{P}} x_{ij}(t) + y_{jj}(t) - s_j(t+1) &= 0 && \forall i \in \mathcal{P} \\ \sum_{j \in \mathcal{W}} y_{jk}(t, \omega_t) - z_k(t, \omega_t) &= 0 && \forall k \in \mathcal{C} \\ x_{ij}(t) &\leq u_{ij}(t) && \forall i \in \mathcal{P},\ j \in \mathcal{W} \\ y_{jk}(t, \omega_t) &\leq u_{jk}(t) && \forall j \in \mathcal{W},\ k \in \mathcal{C} \\ z_k(t, \omega_t) &\leq \xi_k(t, \omega_t) && \forall k \in \mathcal{C} \end{aligned} \tag{28}$$

where $R_i(t)$ is the planned production level at plant i in stage t.

The first four sets of constraints represent flow conservation while the remaining three are capacity constraints. As depicted in figure 5, we can see that the recourse problem in each stage is a transshipment network, where the arcs representing customer demands have random arc capacities (the last set of constraints in (28)).

Notice that the function $\bar{Q}_2(s(1), \omega_2)$ consists of a recursive sequence of minimization problems recursively embedded in expectation. For real world applications, the number of possible realizations can be in the range of 10^{1000} (practically infinite). Thus, obtaining expected recourse functions for multi-stage problems is generally believed to be much more difficult, especially with a large number of stages. Below, we describe a method which avoids the curse of dimensionality arising from the number of stages.

3.2 Backward recursion

When we solve a two-stage distribution planning problem, either the tree recourse logic or the network recourse decomposition method produces a separable function parametrically as a function of $s(1)$. The merit of these techniques is that we can obtain the approximation of the expected recourse function without knowing the actual value of $s(1)$ in advance. This feature motivates the use of the network recourse decomposition recursively when we are solving multi-stage problems.

Cheung and Powell[5] introduce a backward recursion procedure called the *successive convex approximation method* for solving multi- stage stochastic networks in the area of dynamic fleet management. The idea is to apply NRD successively, starting from the last stage back to the second stage. We can adopt this technique in solving dynamic distribution problems.

The idea of this backward recursion is illustrated in figure 6. The stages $t-1$, t and $t+1$ of a multi-stage network is given in figure 6a. Suppose we have obtained the approximation of the expected recourse function for stage $t+1$ (see figure 6b), we can represent this function by sets of deterministic links. We add these links to the terminal nodes in stage t, forming an augmented stage t problem (see figure 6c). Notice that the augmented stage t problem is a network with random arc capacities. Thus, we can apply NRD to this network and obtain an approximation of the expected recourse function for stage t (see figure 6d). Next, the resulting

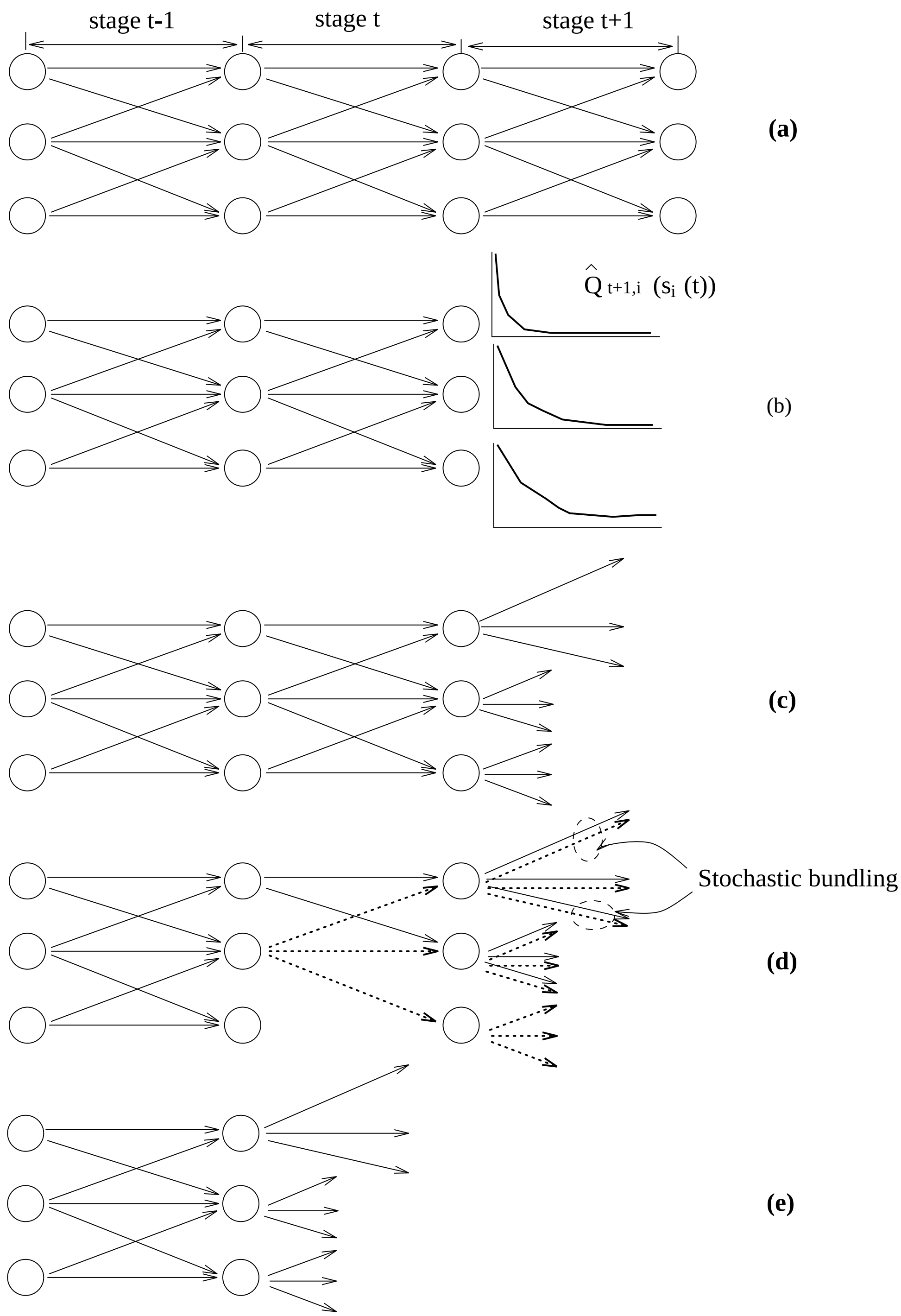


Figure 6: A step in SCAM. (a) Recourse problems in stage $t-1$, t and $t+1$. (b) Approximations for stage $t+1$ problem. (c) Augmented stage t problem. (d) Decomposition by origins, producing trees. (e) Augmented stage $t-1$ problem.

function is represented by sets of deterministic links which are then added to the terminal nodes in stage $t-1$. Again, the augmented stage $t-1$ problem is a network with random arc capacities (see figure 6e). We repeat this step starting from stage N back to stage 2. As a result, we have an augmented stage 1 problem which is a pure network which can be solved easily.

4 Numerical investigation

We have now presented methods for solving distribution problems under uncertainty. In this section, we consider the class of two-stage distribution planning problems. We perform a series of numerical experiments to address two questions, one methodological, and the other substantive. First, we undertake comparisons between three competing solution approaches to estimate the quality of the solution provided by each approach. And second, we use our solution method to evaluate the effectiveness of using multiple warehouses to serve a single customer.

Section 4.1 outlines the experimental design used to address these two questions. Next, section 4.2 describes our random problem generator. Finally, section 4.3 summarizes the results of the experiments and discusses areas for further research.

4.1 Experimental design

Our experimental methodology is as follows. We first use a random problem generator to generate locations of the plant(s), warehouses and customers. Although these randomly generated problems are designed only to provide an indication of the performance of the models, care was taken in the design of the generator to produce problems with appropriate variability (by contrast, more naive problem generators run the risk of generating datasets that are unrealistically uniform).

Next, for each problem, we tested three methods for approximating the expected recourse function. These are:

1. Deterministic model - In this case, we use the expected customer demands to form a deterministic distribution model encompassing plant to warehouse and warehouse to

customer. This model can be solved as a pure network.

2. Tree recourse algorithm - Customers served by more than one warehouse are duplicated into multiple copies of the same customer, one associated with each warehouse that serves the customer. Each duplicated customer has the same original demand. With this modification, flows from a warehouse to customers forms a tree. We then use the tree recourse algorithm described in section 2.3 to calculate the expected recourse function.

3. Network recourse decomposition - Finally, we use the network decomposition strategy given in section 2.4 to provide a more accurate estimate of the expected recourse function.

Once we have an approximation of the expected recourse function, the first stage problem is solved to determine the flows from plants to warehouses.

Once the flows into each warehouse are found, we then use Monte Carlo simulation to generate 1000 observations of customer demands. For each observation, we solve the problem of shipping from warehouse to customer optimally as a network problem. The total cost of a particular solution, then, is obtained by adding the shipping costs from plant to the warehouses to the average shipping cost from warehouses to customers. Since the average cost of shipping from warehouse to customer is derived using Monte Carlo methods, our total cost for a solution is a random variable.

To address the more substantive question of the effectiveness of using multiple warehouses, we generated a series of networks which varied in terms of the number of warehouses which served each customer. Two strategies were used to generate these networks. In the first, we limited every customer to at most two warehouses, and varied the percentage of customers that are served by two warehouses, given by r_p. For example, $r_p = 50$ means that 50 % of the customers are served by two warehouses while the other 50 % are served by a single warehouse. However, there is no restriction on the size of the shipment from a warehouse to a customer. In the second strategy, we allowed a customer to be served by every warehouse within a given radius (subject to the constraint that the closest warehouse was always included) and then varied the radius to obtain different values for the average number of warehouses serving each customer, denoted by r_s. In both datasets, $r_p = 0$ and $r_s = 1$ produces networks with one warehouse per customer, which results in a tree from plant to warehouses to customers. Using the tree recourse method, this problem can be solved optimally. For problems where customers can be served by two or more warehouses, we have to resort to approximations.

4.2 Problem generator

The random problem generator creates a set of points representing the locations of plants, warehouses and customers in a 1000 by 2000 mile area. Let N_p, N_w and N_c be the numbers of plants, warehouses and customers respectively. To obtain the locations of the plants, the area is first partitioned into N_p grids (equally large rectangles). Then, a point is uniformly chosen within each grid to represent a plant's location. The locations for warehouses and customers are similarly obtained. We set the transportation cost to \$1 per mile per truck. Thus, the cost c_{ij} for the shipment from point i to point j is simply the Euclidean distance between point i and point j.

We assume that goods can be shipped from any plant to any warehouse. On the other hand, each customer may only be served by a specific set of warehouses. First, each customer received goods from its closest warehouse. Second, each warehouse serves all customers within a radius of D miles.

To generate customer demands, we first divide the area into N_r regions uniformly and assign profit potential for these regions. The profit potential α_n is drawn uniformly between 0.2 and 1.8, representing the customer's ability to generate demand in this region. Next, we generate a set of Poisson random variables v_k with mean m_k which are given by

$$m_k = \sum_{n=1}^{N_r} \left(\delta_n^k \alpha_n \right) \cdot \mu$$

where

$$\delta_n^k = \begin{cases} 1 & \text{if customer } k \text{ is in region } n \\ 0 & \text{otherwise} \end{cases}$$

$$\mu = \text{is an exponential random variable with mean } \hat{m}$$

Then, the demand ξ_k of customer k is obtained by truncating v_k at v_k' where $P\{v_k > v_k'\} < 0.0001$.

The penalty of each unsatisfied demand at customer k was denoted by r which is chosen after a number of calibrating runs. Finally, the production of plants, R_i are directly proportional

Name	N_p	N_w	N_c	r	d	$\hat{m}$
P1	2	25	100	2000	0.7	0.22
P2	1	12	120	2000	0.7	0.5
P3	4	25	400	1200	0.6	0.3
P4	4	12	150	1500	0.6	0.5
P5	1	8	150	2000	0.7	0.45

Table 1: Characteristics of test problems

to the customer demands. Specifically,

$$R_i = \frac{\sum_{k=1}^{N_c} m_k}{N_p} \cdot d \quad \forall i$$

where $d > 0$.

The parameters used for our test problems are given in table 1.

4.3 Experimental results

A series of simulations was undertaken using the five basic networks described above. Table 2 describes the results with the restriction that every customer be served by at most two warehouses. Column r_p gives the percent of customers served by two warehouses. For example, $r_p = 0$ means that all customers are served by one warehouse while $r_p = 100$ means that all customers are served by two warehouses. The next three columns gives the expected total logistics costs obtained using each of the three approximations. These columns include the plant to warehouse distribution cost, plus the expected costs from warehouse to customer obtained using Monte Carlo simulation. The last column gives the average standard error in these estimates as a result of the Monte Carlo component. Table 3 gives the results of the experiments where each customer is served by all the warehouses within a given radius (including the closest warehouse, if this does not fall within the radius). The average number of warehouses per customer is given in the column marked r_s. In this table, the standard error

Figure 7: Performance of a stochastic model using up to two warehouses per customer

from each of the three models is shown alongside the expected logistics costs.

These tables are used to address both the quality of the solution algorithms, as well as the issue of the effectiveness of using multiple warehouses. First, we note for the case $r_p = 0$ in table 2 and $r_s = 1$ in table 3, we have an instance of a tree out of each warehouse, which can be solved exactly using the tree recourse algorithm. In this case, the tree recourse algorithm and the network recourse decomposition method give identical results.

To assess the quality of different solution algorithms, we present the results in tables 2 and 3 as the relative improvement over the deterministc models. Figure 7 shows the relative improvement produced using a stochastic model (and the NRD algorithm) for the case with at most two warehouses per customer, while figure 8 gives the same information for multiple warehouses. From the two figures, we can see that the tree recourse algorithm outperformed

Problem	$r_p(\%)$	DETM	TREE	NRD	mean stand. error
P1	0	-19036	-20580	-20580	93
	20	-20469	-22643	-22798	94
	40	-22137	-23155	-23155	95
	60	-22860	-23740	-23740	95
	80	-23351	-23962	-23842	90
	100	-24034	-25030	-25251	91
P2	0	-39058	-41931	-41931	153
	10	-39834	-42557	-42557	151
	20	-40881	-43035	-43035	152
	30	-41295	-43593	-43593	158
	40	-42093	-43136	-43797	168
	50	-43473	-42838	-43092	159
	60	-42708	-44814	-45082	158
	70	-43508	-44868	-45304	160
	80	-45019	-44893	-46239	172
	90	-45350	-44893	-46394	175
	100	-45593	-45571	-46587	169
P3	0	-20499	-21041	-21041	88
	20	-21251	-21339	-21746	102
	40	-22441	-22892	-22892	92
	60	-23269	-22798	-23236	98
	80	-23887	-23816	-23816	89
	100	-24445	-24588	-24588	78
P4	0	-33382	-34629	-34629	97
	20	-36797	-37614	-37576	98
	40	-38090	-38868	-38678	96
	60	-39280	-40270	-40068	98
	80	-40786	-42120	-41782	93
	100	-41218	-43278	-43635	92
P5	0	-51880	-54985	-54985	141
	20	-53325	-55747	-55747	147
	40	-55339	-55691	-56495	142
	70	-56140	-56831	-57211	127
	100	-57734	-57814	-58138	97

Table 2: Expected distribution costs with at most two warehouses per customer, given the percent of customers served by two warehouses

Problem	r_s	DETM	stand. error	TREE	stand. error	NRD	stand. error
P1	1.0	-19007	140	-20586	124	-20586	124
	1.2	-19953	134	-21174	144	-21420	135
	1.5	-21978	138	-23053	128	-23053	128
	2.0	-23051	140	-23641	135	-24035	132
	2.5	-23421	144	-24555	129	-24517	126
	3.0	-23601	144	-24705	129	-24675	126
	3.5	-23661	145	-24816	128	-24627	127
	4.0	-23661	145	-24816	128	-24627	127
	4.5	-23661	145	-24816	128	-24627	127
	5.0	-23661	145	-24816	128	-24627	127
P2	1.0	-39102	174	-41975	141	-41975	141
	1.2	-41091	172	-42997	152	-42653	125
	1.5	-43245	177	-43056	179	-41224	136
	2.0	-43339	186	-43022	189	-43942	176
	2.5	-44160	184	-43265	194	-44305	105
	3.0	-45158	182	-45010	191	-46516	151
	3.5	-45158	182	-45010	191	-46407	145
	4.0	-45158	182	-45010	191	-46407	145
	4.5	-45158	182	-45010	191	-46407	145
P3	1.0	-20564	93	-21060	86	-21060	86
	1.2	-22136	98	-22039	108	-22111	103
	1.5	-22874	93	-22325	109	-22849	101
	2.0	-23675	87	-23277	101	-23854	91
	2.5	-24062	82	-23444	100	-23988	90
	3.0	-24140	80	-24052	90	-24021	90
	3.5	-24493	77	-24652	78	-24652	78
	4.0	-24500	77	-24659	78	-24659	78
	4.5	-24500	77	-24659	78	-24659	78
	5.0	-24505	77	-24664	78	-24664	78
P4	1.0	-33364	143	-34592	135	-34592	135
	1.2	-35389	153	-36013	150	-36385	142
	1.5	-37945	151	-38426	155	-35327	148
	2.0	-39740	139	-40593	146	-41139	131
	2.5	-40812	134	-41816	140	-41598	128
	3.0	-41252	137	-42666	137	-42482	133
	3.5	-41403	137	-43059	131	-43241	124
	4.0	-41859	138	-43202	131	-43473	123
	4.5	-41624	141	-43345	131	-43562	122
	5.0	-41896	139	-43368	130	-43634	122
P5	1.0	-49785	192	-52573	122	-52573	122
	1.2	-50819	209	-53043	160	-53573	146
	1.5	-52451	207	-53008	190	-54252	159
	2.0	-53947	210	-53418	229	-55892	159
	2.5	-57573	107	-57818	108	-58081	95
	3.0	-57872	102	-57945	105	-57725	105
	3.5	-57872	102	-57946	105	-57726	105
	4.0	-57872	102	-57946	105	-57726	105
	4.5	-57887	101	-57956	104	-58337	85
	5.0	-57887	101	-57956	104	-58337	85

Table 3: Expected distribution costs given the average number of warehouses serving each customer

Figure 8: Performance of a stochastic model using multiple warehouses per customer

Figure 9: Relative improvement using up to two warehouses per customer

the deterministic model in 64 out of 83 runs, while the NRD algorithm outperformed the deterministic model in 71 out of 83 runs. In terms of total cost, the stochastic model outperformed the deterministic model by 5.5 percent on average. It suggests that the explicit consideration of the stochastic aspects of the customer demands can produce a substantial savings. From these results, we speculate that we have a very high quality solution to the stochastic distribution problem, especially when we use the NRD algorithm to solve the stochastic model.

On the other hand, when we compare the two methods of solving the stochastic model, we found that the NRD algorithm outperformed the pure tree recourse algorithm by approximately 0.5 percent. Given the relative simplicity of the tree recourse algorithm, it is not clear that the additional complexity of the NRD algorithm is warranted.

We now turn to the more substantive question of the economics of using multiple warehouses for at least some percentage of the customers. Using the data from tables 2 and 3, we calculated the relative improvement in total logistics costs (using the NRD algorithm) as a function of the number of warehouses serving each customer. Figure 9 shows the results for the case with

Figure 10: Relative improvement using multiple warehouses per customer

at most two warehouses per customer, and figure 10 shows the results for multiple warehouses (can be more than two) per customer. These figures both suggest an almost linear improvement in total costs as the number of warehouses per customer increases from a base of one up to two. In figure 10, however, when the number of warehouses per customer is beyond two, the rate of improvement drops off sharply. For example, problem p5 increases up to 2.5, and then levels out, while, problem p4 continues to show improvement up to 3.5 warehouses per customer. In any case, using more than two warehouses per customer does not warrant a significant savings. In practice, on the other hand, using more warehouses can induce substantial administative costs.

The results here are somewhat surprising, and raise some questions that deserve further study. Some observations that can be derived from this data are:

1. Standard distribution networks use one warehouse per customer. Most companies prefer a one-to-one relationship between customers and warehouses for practical reasons, which of course are not captured by this study. However, we suspect that if a survey were con-

ducted of distribution systems to determine the average number of warehouses that serve each customer, the answer would be a number significantly less than two. This limited study suggests that large improvements can be attained in a real-time environment by connecting *most* customers with two warehouses.

2. The improvements that are attained as the system approaches two warehouses per customer are large, and surprising. A line of investigation is that whether most of the savings of using multiple warehouses would be attained when only a portion of customers are served by multiple warehouses.

3. Further research is required to understand the factors that affect the performance of the design of different logistics networks under uncertainty. It is quite likely that performance is sensitive to choosing specific customers for assignment to multiple warehouses. Also, we have not investigated the sensitivity to other parameters, such as holding cost and the opportunity cost of lost sales. Finally, we need to better understand the stochastic properties of actual customer demands, and determine the sensitivity of the solution to the characteristics of the demand process.

Acknowledgments

We would like to thank Professor Mark Daskin, the referees and the associate editor for their valuable comments and suggestions. This research was supported in part by grant DDM-9102134 from the National Science Foundation, and by grant AFOSR-F49620-93-1-0098 from the Air Force Office of Scientific Research.

References

[1] J. Aronson. A survey of dynamic network flows. In P. Hammer, editor, *Annals of Operations Research*, pages 1–66. J.C. Baltzer AG, 1989.

[2] J. Aronson and B. Chen. Decision horizon results for an infinite horizon, production planning network model. Technical report 85-or-3, Department of Operations Research and Engineering Management, Southern Methodist University, 1985.

[3] J. Aronson and B. Chen. A forward network simplex algorithm for solving multiperiod network flow problems. *Naval Research Logistics Quarterly*, 33(3):445–467, 1986.

[4] R. Ballou. *Business Logistics Management.* Prentice Hall, New York, 1992.

[5] R. K.-M. Cheung and W. B. Powell. An algorithm for multistage dynamic networks with random arc capacitites. Technical Report SOR-92-11, Dept. of Civil Engineering and Operations Research, Princeton University, 1992.

[6] S. Graves, A. Rinnooy Kan, and P. Zipkin. *Logistics of Production and Inventory.* North-Holland, New York, 1993.

[7] U. S. Karmarkar. Convex/stochastic programming and multilocation inventory problems. *Naval Res. Logist. Quart.*, 26:1–19, 1979.

[8] U. S. Karmarkar. The multiperiod multilocation inventory problems. *Operations Research*, 29:215–228, 1981.

[9] U. S. Karmarkar. The multilocation multiperiod inventory problem: bounds and approximations. *Management Science*, 33(1):86–94, 1987.

[10] U. S. Karmarkar and N. Patel. The one-period, N-location distribution problem. *Naval Res. Logist. Quart.*, 24:559–575, 1977.

[11] D. Klingman and J. Mote. A multi-period production, distribution and inventory planning model. *Advances in Management Studies*, 1(2):56–76, 1982.

[12] I. J. Lustig, J. M. Mulvey, and T. J. Carpenter. Formulating two-stage stochastic programs for interior point methods. *Operations Research*, 39:757–770, 1991.

[13] W. B. Powell and R. K.-M. Cheung. Network recourse decomposition method for dynamic networks with random arc capacities. *Networks*, pages 369–384, 1994.

[14] W. B. Powell and R. K.-M. Cheung. Stochastic programs over trees with random arc capacities. *Networks*, 24:161–175, 1994.

[15] J. Robeson and R. House. *The Distribution Handbook.* Free Press, New York, 1981.

[16] T. Rockafellar and R. Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, 16(1):119–147, 1991.

[17] J. Shapiro. Mathematical programming models and methods for production planning and scheduling. In S. Graves, A. Rinnooy Kan, and P. Zipkin, editors, *Logistics of Production and Inventory*, pages 371–444. North-Holland, 1993.

[18] N. Shor. *The Methods of Nondifferentiable Optimization and their Applications.* Naukova Dumka, Kiev, 1979.

[19] S. W. Wallace. Solving stochastic programs with network recourse. *Networks*, 16:295–317, 1986.

[20] R. Wets. Stochastic programs with fixed resources: the equivalent deterministic problem. *SIAM Review*, 16:309–339, 1974.

[21] R. Wets. The aggregation principle in scenario analysis and stochastic optimization. In S. Wallace, editor, *Algorithms and Model Formulations in Mathematical Programming.* Springer-Verlag, Berlin, 1989.

[22] W. Zangwill. A backlogging model and a multi-echelon model of a dynamic economic lot size production system – a network approach. *Management Science*, 15(9):506–527, 1969.