Optimizing Strategic Safety Stock Placement in Supply Chains

Stephen C. Graves • Sean P. Willems
Leaders for Manufacturing Program and A. P. Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139-4307, sgraves@mit.edu
College of Business Administration, University of Cincinnati, Cincinnati, Ohio 45221-0130

Manufacturing managers face increasing pressure to reduce inventories across the supply chain. However, in complex supply chains, it is not always obvious where to hold safety stock to minimize inventory costs and provide a high level of service to the final customer. In this paper we develop a framework for modeling strategic safety stock in a supply chain that is subject to demand or forecast uncertainty. Key assumptions are that we can model the supply chain as a network, that each stage in the supply chain operates with a periodic-review base-stock policy, that demand is bounded, and that there is a guaranteed service time between every stage and its customers. We develop an optimization algorithm for the placement of strategic safety stock for supply chains that can be modeled as spanning trees. Our assumptions allow us to capture the stochastic nature of the problem and formulate it as a deterministic optimization. As a partial validation of the model, we describe its successful application by product flow teams at Eastman Kodak. We discuss how these flow teams have used the model to reduce finished goods inventory, target cycle time reduction efforts, and determine component inventories. We conclude with a list of needs to enhance the utility of the model.
(*Base-Stock Policy*; *Dynamic Programming Application*; *Multi-echelon Inventory System*; *Multi-Stage Supply-Chain Application*; *Safety Stock Optimization*)

1. Introduction

Manufacturing firms are subject to pressure to do everything faster, cheaper, and better. Firms are expected to continue to improve customer service by increasing on-time deliveries and reducing delivery lead-times. At the same time, they must provide this service more cheaply and utilize fewer assets. Increasingly, firms need to do this for a global marketplace.

This pressure to improve forces companies to look at their operations from a supply-chain perspective and to seek improvements from better coordination and communication across the supply chain. A supply-chain perspective is essential to avoid some of the local suboptimization that occurs when each step in a process operates independently with its own metrics and rewards. Using a supply chain as a focusing mechanism challenges an organization to examine cross-functional solutions to address some of the barriers that inhibit improvements.

The primary intent of this research is to develop a tactical tool to help cross-functional teams in their efforts to model and improve a supply chain. We provide a framework for modeling a supply chain and develop an approach, within the framework, to optimize the inventory in a supply chain. More specifically, we provide an optimization algorithm for finding the optimal placement of safety stock in a supply chain, modeled as a spanning tree and subject to uncertain demand. Key assumptions for the optimization are that each stage of the supply chain operates with a

1523-4614/00/0201/0068$05.00
1526-5498 electronic ISSN

periodic-review, base-stock policy, that each stage quotes a guaranteed service time to its customers, and that demand is bounded.

We refer to this effort as the placement of "strategic" safety stock. As will be seen, the optimization model leads to the determination of where to place decoupling inventories that protect one part of the supply chain from another. In particular, a decoupling safety stock is an inventory large enough to permit the downstream portion of the supply chain to operate independently from the upstream, provided that the upstream portion replenishes the external demand. In this sense, the determination of where to place these decoupling points in a supply chain is a major design decision and is "strategic" in nature. Furthermore, this terminology is consistent with that used in industry.

In order to have an opportunity to test the research and validate its utility for industry, we have built a commercial-quality software application to implement the model described in this paper. The software can be downloaded from our website, http://web.mit.edu/lfmrg3/www/.

In the remainder of this section we briefly discuss related literature. In §2, we present our framework for modeling a supply chain by stating and discussing the key assumptions. We introduce the model for a single stage in §3; this serves as the building block for the multi-stage model described in §4. In §5 we develop the optimization algorithm for safety stock placement in a supply chain modeled as a spanning tree. We present an overview of our application experience with the model in §6, and conclude in §7 with thoughts on how to improve the tool.

Related Literature.
There is an extensive literature on inventory models for multi-stage or multi-echelon systems with uncertain demand; much of this literature is applicable to supply chains as now defined. We refer the reader to the survey articles by Axsäter (1993), Federgruen (1993), Inderfurth (1994), van Houtum et al. (1996), and Diks et al. (1996). Within this vast literature, we mention two sets of papers that are most related to our work.

First, we note the work by Simpson (1958) who determined optimal safety stocks for a supply chain modeled as a serial network. Our work is based on similar assumptions about the demand process and about the internal control policies for the supply chain. Our work is also closely related to that of Inderfurth (1991, 1993), Inderfurth and Minner (1998), and Minner (1997), who also build off Simpson's framework for optimizing safety stocks in a supply chain. We extend the work of Simpson and of Inderfurth and Minner by treating a more general network, namely spanning trees. We also provide a different, and we believe richer, interpretation of the framework and its applicability to practice. We provide new results in the appendix on the form of the optimal policies when we relax a constraint on the internal control policy for the supply chain.

Second, our work is closely related in intent to Lee and Billington (1993), Glasserman and Tayur (1995), and Ettl et al. (2000). Each of these papers examines the determination of the optimal base-stock levels in a supply chain, and tries to do so in a way that is applicable to practice. Glasserman and Tayur (1995) show how to use simulation and infinitesimal perturbation analysis to find the optimal base-stock levels for capacitated multi-stage systems. Both Lee and Billington (1993) and Ettl et al. (2000) develop performance evaluation models of a multi-stage inventory system, where the key challenge is how to approximate the replenishment lead-times within the supply chain. They then formulate and solve a nonlinear optimization problem that minimizes the supply chain's inventory costs subject to user-specified requirements on the customer service level. Our work is similar in that we also assume base-stock policies and focus on minimizing the inventory requirements in a supply chain. The resulting models and algorithms are much different, though, due to different assumptions about the demand process and different constraints on service levels within the supply chain.

2. Assumptions

Multi-Stage Network.
We model a supply chain as a network where nodes are stages in the supply chain and arcs denote that an upstream stage supplies a downstream stage. A stage represents a major processing function in the supply chain. A stage might represent the procurement of a raw material, or the production of a component, or the

manufacture of a subassembly, or the assembly and test of a finished good, or the transportation of a finished product from a central distribution center to a regional warehouse. Each stage is a potential location for holding a safety-stock inventory of the item processed at the stage.

We associate with each arc a scalar ϕ_{ij} to indicate how many units of the upstream component i are required per downstream unit j. If a stage is connected to several upstream stages, then its production activity is an assembly requiring inputs from each of the upstream stages. A stage that is connected to multiple downstream stages is either a distribution node or a production activity that produces a common component for multiple internal customers.

Production Lead-Times.
For each stage, we assume a known deterministic production lead-time; call it T_i. When a stage reorders, the production lead-time, is the time from when all of the inputs are available until production is completed and available to serve demand. The production lead-time includes the waiting and processing time at the stage, plus any transportation time to put the item into inventory. For instance, suppose stage k requires inputs from stage i and j; then for a production request made at time t, stage k completes the production at time $t + T_k$, provided that there are adequate supplies of i and j at time t.

We assume that the production lead-time is not impacted by the size of the order; hence, in effect, we assume that there are no capacity constraints that limit production at a stage.

Periodic-Review Base-Stock Replenishment Policy.
We assume that all stages operate with a periodic-review base-stock replenishment policy with a common review period. For each period, each stage observes demand either from an external customer or from its downstream stages, and places orders on its suppliers to replenish the observed demand. There is no time delay in ordering; hence, in each period the ordering policy passes the external customer demand back up the supply chain so that all stages see the customer demand.

Demand Process.
Without loss of generality, we assume that external demand occurs only at nodes that have no successors, which we term demand nodes or stages. For each demand node j, we assume that the end-item demand comes from a stationary process for which the average demand per period is μ_j.

An internal stage has only internal customers or successors; its demand at time t is the sum of the orders placed by its immediate successors. Since each stage orders according to a base-stock policy, the demand at internal stage i is:

$$d_i(t) = \sum_{(i,j)\in A} \phi_{ij}\, d_j(\mathrm{t})$$

where $d_j(t)$ denotes the realized demand at stage j in period t and A is the arc set for the network representation of the supply chain. For every arc $(i, j) \in A$, stage j orders an amount $\phi_{ij}\, d_j(t)$ from upstream stage i in time period t. The average demand rate for stage i is:

$$\mu_i = \sum_{(i,j)\in A} \phi_{ij}\, \mu_j.$$

We assume that demand at each stage j is *bounded* by the function $D_j(\tau)$, for $\tau = 1, 2, 3, \ldots M_j$, where M_j is the maximum replenishment time for the stage.[1] That is, for any period t and for $\tau = 1, 2, 3, \ldots M_j$, we have

$$D_j(\tau) \geq d_j(t - \tau + 1) + d_j(t - \tau + 2) + \cdots + d_j(t).$$

We define $D_j(0) = 0$ and assume that $D_j(\tau)$ is increasing and concave on $\tau = 1, 2, 3, \ldots M_j$.

Discussion of Assumption of Bounded Demand.
The assumption of bounded demand is contrary to most of the literature on stochastic-demand inventory models, and as such, is controversial. We need to discuss this assumption in the context of the intent of the research, namely to provide tactical guidance for where to position safety stock in a supply chain.

We presume that it is possible to establish a meaningful upper bound on demand over varying horizons for each end item. By meaningful, we mean in the context of setting safety stocks: the safety stock is set to cover all demand realizations that fall within the upper bounds. If demand exceeds the upper bounds, then the safety stock, by design, is not adequate. In such extraordinary cases, a manager resorts to other tactics to

[1]The maximum replenishment time for node j is defined as $M_j = T_j + \max\{M_i \mid i{:}(i,j) \in A\}$.

handle the excess demand. A manager might use expediting, subcontracting, premium freight transportation, and/or overtime to accommodate the windfall of demand. In specifying the demand bounds, a manager indicates explicitly a preference for how demand variation should be handled—what range is covered by safety stock and what range is handled by other actions or responses.

As an example, consider a typical assumption where demand for end item j is normally distributed each period and i.i.d., with mean μ and standard deviation σ. Then, for the purposes of positioning safety stock, a manager might specify the demand bounds at the demand node by:

$$D_j(\tau) = \tau\mu + k\sigma\sqrt{\tau} \qquad (1)$$

where k reflects the percentage of time that the safety stock covers the demand variation. The choice of k indicates how frequently the manager is willing to resort to other tactics to cover demand variability.

In some contexts there may be natural bounds on the end-item demand. For instance, suppose the end item is a component or subassembly for a manufacturing process whose production is limited by capacity constraints or by a frozen master schedule. An example is a supply chain that supplies components to an automobile assembly line or an OEM subassembly to a system integrator. In these cases, bounded demand for the component corresponds to the maximum consumption by the manufacturing process over various time horizons.

For each internal stage we assume that we can also establish meaningful demand bounds. If stage i has a single successor, say stage j, then $D_i(\tau) = \phi_{ij} D_j(\tau)$ for all relevant τ. For stages with more than one successor, we require some judgment for deciding how to combine the demand bounds for the downstream stages to obtain a relevant demand bound for the upstream stage for the purposes of positioning the safety stock. One possibility is just to sum the downstream demand bounds; however, this approach assumes that there is no risk pooling from combining the demand of multiple end items. An alternative approach is to assume that there will be some relative reduction in variability as we combine demand streams, i.e., some risk pooling. For instance, we might infer the demand bounds for internal stages by means of an expression like

$$D_i(\tau) = \tau\mu_i + p\sqrt{\sum_{(i,j)\in A} \{\phi_{ij}(D_j(\tau)' - \tau\mu_j)\}^p} \qquad (2)$$

where $p \geq 1$ is a given constant. Larger values of p correspond to more risk pooling. Setting $p = 1$ models the case of no risk pooling. If we were to model the end-item demand bounds by Equation (1), then setting $p = 2$ equates to combining standard deviations of independent demand streams.

We do not attempt to model what happens when actual demand exceeds the bounds. When this happens, we assume that the supply chain responds with an equally extraordinary measure, as noted above. We regard this as beyond the scope of the model, given the stated intention to provide tactical decision support. See Kimball (1988), Simpson (1958), and Graves (1988) for further discussion of this assumption.

Finally we note that there are *no assumptions made about the demand distribution.*

Guaranteed Service Times.
We assume that each demand node j promises a *guaranteed service time* S_j by which the stage j will satisfy customer demand. That is, the customer demand at time t, $d_j(t)$, must be filled by time $t + S_j$. Furthermore, we assume that stage j provides *100% service* for the specified service time: stage j delivers exactly $d_j(t)$ to the customer at time $t + S_j$.

Similarly, an internal stage i quotes and guarantees a service time S_{ij} for each downstream stage j, $(i, j) \in A$. Given a base-stock policy, stage j places an order equal to $\phi_{ij}\, d_j(t)$ on stage i at time t; then stage i delivers exactly this amount to stage j at time $t + S_{ij}$.

For the initial development, we assume that stage i quotes the same service time to all of its downstream customers; that is, $S_{ij} = S_i$ for each downstream stage j, $(i, j) \in A$. Graves and Willems (1998) describe how to extend the model to permit customer-specific service times. In brief, if there is more than one downstream customer, we can insert zero-cost, zero-production lead-time dummy nodes between a stage and its customers to enable the stage to quote different service times to each of its customers. The stage quotes the same service time to the dummy nodes and each dummy node is free to quote any valid service time to its customer stage.

The service times for both the end items and the internal stages are *decision variables* for the optimization model, as will be seen in §4. However, as a model input, we may impose bounds on the service times for each stage. In particular, we assume that for each end item we are given a maximum service time, presumably set by the marketplace.

Discussion of Assumption of Guaranteed Service Times.
We assume that there are no violations of the guaranteed service times; each stage provides perfect or 100% service to its customers. As such, we do not explicitly model a tradeoff between possible shortage costs and the costs for holding inventory. Rather, we pose the problem as being how to place safety stocks across the supply chain to provide 100% service for the assumed bounded demand with the least inventory holding cost.

In defense of this assumption, it is often very difficult in practice to assess shortage costs for an external customer. Similarly, when we have asked managers for their desired service level, more often than not the response is that there should be no stock-outs for external customers. We have found that managers seem more comfortable with the notion of 100% service for some range of demand; they accept the fact that if demand exceeds this range, they will have shortages unless they can somehow expand the response capability of their supply chain. The assumptions for the model presented herein are consistent with this perspective.

For an internal customer, guaranteed service times need not be optimal in terms of least inventory costs. Indeed we show in the Appendix how to relax this assumption for a serial network, and report the cost impact of this assumption for a set of 36 test problems: the safety stock holding cost is 26% higher on average, while the total inventory cost is 4% higher on average. However, guaranteed service times are very practical in contexts where there is a need to coordinate replenishments. For instance, any assembly or subassembly stage requires the concurrent availability of multiple components, not all of which might be explicitly included in the model. When we assume guaranteed service times, we make the challenge of coordinating the availability of these components much easier.

3. Single-Stage Model

In this section we present the single-stage model (see Kimball 1988 or Simpson 1958) that serves as the building block for modeling a multi-stage supply chain.

Inventory Model
We assume the inventory system starts at time 0 with initial inventory $I_j(0)$. Given our assumptions, we can express the finished inventory at stage j at the end of period t as

$$I_j(t) = B_j - d_j(t - SI_j - T_j, t - S_j) \tag{3}$$

where $B_j = I_j(0) \geq 0$ denotes the base stock, $d_j(a, b)$ denotes the demand at stage j over the time interval $(a, b]$, and SI_j is the inbound service time for stage j. Since we assume a discrete-time demand process, we understand $d_j(a, b)$ to be

$$d_j(a, b) = d_j(a + 1) + d_j(a + 2) + \cdots + d_j(b)$$

for $a < b$, where $d_j(t) = 0$ for $t \leq 0$. When $a \geq b$, we define $d_j(a, b) = 0$.

The inbound service time SI_j is the time for stage j to get supplies from its immediate suppliers and to commence production. In period t, stage j places an order equal to $\phi_{ij}\, d_j(t)$ on each upstream stage i for which $\phi_{ij} > 0$. Stage j cannot start production to replenish $d_j(t)$ until all inputs have been received; thus we have $SI_j \geq \max \{S_i \mid i{:}(i, j) \in A\}$.

We permit $SI_j > \max \{S_i \mid i{:}(i, j) \in A\}$ to allow for the possibility that the replenishment time for the inventory at stage j is less than its service time S_j; that is, the case when

$$\max \{S_i \mid i{:}(i, j) \in A\} + T_j < S_j.$$

In this case we would delay the orders to the suppliers by $S_j - \max \{S_i \mid i{:}(i, j) \in A\} - T_j$ periods, so that the supplies arrive exactly when needed. To account for this case, we set the inbound service time so that the effective replenishment time for the inventory at stage j, namely $SI_j + T_j$, equals the service time S_j, i.e., $SI_j + T_j = S_j$. Thus, we define the inbound service time as

$$SI_j = \max\{S_j - T_j, \max \{S_i \mid i{:}(i, j) \in A\}\}.$$

If the inbound service time is such that $SI_j > S_i$ for some $(i, j) \in A$, then by convention stage j delays orders from stage i by $SI_j - S_i$ periods to avoid unnecessary inventory.

Now, to explain Equation (3), we observe that in period t stage j completes the replenishment of the demand observed in period $t - SI_j - T_j$. By the end of period t, the cumulative replenishment to the inventory at stage j equals $d_j(0, t - SI_j - T_j)$. In period t, stage j fills the demand observed in time period $t - S_j$ from its inventory. By the end of period t the cumulative shipment from the inventory at stage j equals $d_j(0, t - S_j)$. The difference between the cumulative replenishment and the cumulative shipment is the inventory shortfall, $d_j(t - SI_j - T_j, t - S_j)$. The on-hand inventory at stage j is the initial inventory or base stock minus the inventory shortfall, as given by Equation (3).

Determination of Base Stock.
In order for stage j to provide 100% service to its customers, we require that $I_j(t) \geq 0$; from (3) we see that this requirement equates to

$$B_j \geq d_j(t - SI_j - T_j, t - S_j).$$

Since demand is bounded, we can satisfy the above requirement with the least inventory by setting the base stock as follows:

$$B_j = D_j(\tau) \quad \text{where } \tau = SI_j + T_j - S_j. \tag{4}$$

Any smaller value does not assure that $I_j(t) \geq 0$, and thus cannot guarantee 100% service.

In words, the base stock equals the maximum possible demand over the *net* replenishment time for the stage. The *net* replenishment time for stage j is the replenishment time ($SI_j + T_j$) minus its service time (S_j). At any time t, stage j has filled its customers' demand through time $t - S_j$, but has only been replenished for demand through time $t - SI_j - T_j$. The base stock must cover this time interval of exposure, namely the net replenishment time.

Safety Stock Model.
We use Equations (3) and (4) to find the expected inventory level $E[I_j]$:

$$E[I_j] = B_j - E[d_j(t - SI_j - T_j, t - S_j)]$$
$$= D_j(SI_j + T_j - S_j) - (SI_j + T_j - S_j)\mu_j. \tag{5}$$

The expected inventory represents the safety stock held at stage j, and depends on the net replenishment time and the demand bound. As an example, suppose the demand bound is given by Equation (1); then the safety stock is $E[I_j] = k\sigma\sqrt{SI_j + T_j - S_j}$.

Pipeline Inventory.
In addition to the safety stock, we may want to account for the in-process or pipeline stock at the stage. Following the development for Equation (3), we observe that the work-in-process inventory at time t is given by

$$W_j(t) = d_j(t - SI_j - T_j, t - SI_j).$$

That is, the work-in-process corresponds to T_j periods of demand. The expected work-in-process depends only on the lead-time at stage j and is not a function of the service times:

$$E[W_j] = T_j\mu_j.$$

Hence, in posing an optimization problem in the next section, we ignore the pipeline inventory and only model the safety stock. This is not to say that the work-in-process is not a significant part of the inventory in a supply chain. But for the purposes of this work, we assume that the lead-time of a stage, as well as the demand rate, are input parameters and thus the pipeline stock is predetermined. Nevertheless, in any application, we account for both the safety stock and the pipeline stock as both will contribute to the total supply chain inventory.

4. Multi-Stage Model

To model the multi-stage system, we use Equation (5) for every stage, but where the inbound service time is a function of the outbound service times for the upstream stages; to wit, the model for stage j is

$$E[I_j] = D_j(SI_j + T_j - S_j) - (SI_j + T_j - S_j)\mu_j, \tag{6}$$

$$SI_j + T_j - S_j \geq 0, \tag{7}$$

$$SI_j - S_i \geq 0 \quad \text{for all } (i, j) \in A. \tag{8}$$

Equation (6) expresses the expected safety stock as a function of the net replenishment time. Equation (7) assures that the net replenishment time is nonnegative. Equation (8) constrains the inbound service time to equal or exceed the service times for the upstream stages.

We assume that the production lead-times, the means and bounds of the demand processes, and the maximum service times for the demand nodes are known input parameters. This suggests the following optimization problem **P** for finding the optimal service times:

$$\mathbf{P} \ \min \sum_{j=1}^{N} h_j\{D_j(SI_j + T_j - S_j) - (SI_j + T_j - S_j)\mu_j\}$$

$$\text{s. t.} \quad S_j - SI_j \le T_j \quad \text{for } j = 1, 2, \ldots, N,$$

$$SI_j - S_i \ge 0 \quad \text{for all } (i, j) \in A,$$

$$S_j \le s_j \quad \text{for all demand nodes } j,$$

$$S_j, SI_j \ge 0 \text{ and integer} \quad \text{for } j = 1, 2, \ldots, N,$$

where h_j denotes the per-unit holding cost for inventory at stage j, and s_j is the maximum service time for demand node j. The objective of problem **P** is to minimize the holding cost for the safety stock in the supply chain. The constraints assure that the net replenishment times are nonnegative, the inbound service time equals the maximum supplier service time, and the end-item stages satisfy their service guarantee. The decision variables are the service times.

Problem **P** is a nonlinear optimization problem. The objective function is a concave function, provided that the demand bound $D_j(\,)$ is a concave function for each stage j. The feasible region is convex but not necessarily bounded; however, one can show that the optimal service times need not exceed the sum of the production lead-times, provided that $D_j(\,)$ is a nondecreasing function for each stage j. Thus, problem **P** is the minimization of a concave function over a closed, bounded convex set. An optimum for such problems is at an extreme point of the feasible region, e.g., Luenberger 1973.

Simpson (1958) considered a serial-line supply chain, where he assumed that the guaranteed service time for the external customer is zero. Simpson showed that there is an optimal extreme point solution for **P** for which $S_i = 0$ or $S_i = S_{i+1} + T_i$, where stage $i + 1$ supplies stage i. Thus, there is an "all or nothing" optimal solution; a stage either has no safety stock ($S_i = S_{i+1} + T_i$) or has sufficient safety stock ($S_i = 0$) to de-couple it from its downstream stage. Gallego and Zipkin (1999) provide supporting evidence that "all or nothing" policies can be near optimal in serial systems under more traditional assumptions where demand is not bounded.

Graves (1988) observed that the serial-line problem can be solved as a shortest path problem. In a series of papers. Inderfurth (1991, 1993), Inderfurth and Minner (1998), and Minner (1997) show how to solve problem **P** by dynamic programming when the supply chain is an assembly network or a distribution network. Graves and Willems (1996) developed similar results for assembly and distribution networks. In the next section we present a dynamic programming algorithm for the more general case of a spanning tree.

5. Algorithm for Spanning Tree

We describe in this section how to solve **P** by dynamic programming when the underlying network for the supply chain is a spanning tree, like in the Figure 1.

We solve **P** by decomposing the problem into N stages where N is the number of nodes in the spanning tree and there is one stage for each node. For a spanning tree, there is not a readily-apparent ordering of the nodes by which the algorithm would proceed. Indeed, we label the nodes in a spanning tree (and thus sequence the algorithm) so that there will be a single state variable for the dynamic programming recursion. However, the state variable for the dynamic program will be either the inbound service time at a stage or its outbound service time, where the determination depends on the topology of the network.

We first present the algorithm for labeling the nodes. Next we present the functional equations for the dynamic programming recursions, and then state the algorithm.

Labeling the Nodes.

The algorithm for labeling or re-numbering the nodes is as follows:

1. Start with all nodes in the unlabeled set, U.
2. Set $k := 1$.
3. Find a node $i \in U$ such that node i is adjacent to at most one other node in U. That is, the degree of node i is 0 or 1 in the subgraph with node set U and arc set A defined on U.
4. Remove node i from set U and insert into the labeled set L; label node i with index k.

5. Stop if U is empty: otherwise set $k := k + 1$ and repeat steps 3– 4.

For a spanning tree, there is always an unlabeled node in step 3 that is adjacent to at most one other unlabeled node. As a consequence, the algorithm will eventually label all of the nodes in N iterations. Indeed, each node labeled in the first $N - 1$ steps is adjacent to exactly one other node in set U. Thus, the nodes with labels 1, 2, . . . , $N - 1$ have one adjacent node with a higher label, denoted by $p(k)$ for $k = 1, 2, \ldots, N - 1$. Node N has no adjacent nodes with larger labels.

As an illustration, we renumber the nodes in Figure 1 to produce Figure 2. Note that the labeling is not unique as there may be multiple choices for node i in step 3.

For each node k we define N_k to be the subset of nodes {1, 2, . . . , k} that are connected to k on the subgraph with node set {1, 2, . . . , k}. We will use N_k to explain the dynamic programming recursion. We can determine N_k by the following equation:

$$N_k = \{k\} + \bigcup_{i<k,(i,k)\in A} N_i + \bigcup_{j<k,(k,j)\in A} N_j.$$

For instance, in Figure 2, N_k is {3} for $k = 3$, {1, 2, 3, 9} for $k = 9$, {1, 2, 3, 4, 5, 9, 11} for $k = 11$, and {6, 7, 8, 10, 12} for $k = 12$. We can compute N_k as part of the labeling algorithm.

Figure 1 **Spanning Tree**

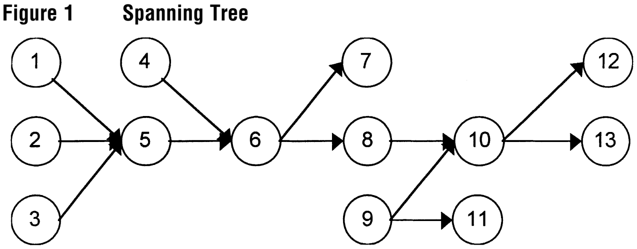
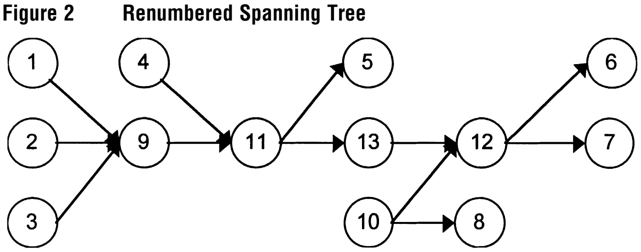


Figure 2 **Renumbered Spanning Tree**



Functional Equations.

The dynamic program evaluates a functional equation for each node in the spanning tree, where we have renumbered the nodes as described above. There are two forms for the functional equation. First, the function $f_k(S)$ is the minimum holding cost for safety stock in a subnetwork with node set N_k, where we assume that the outbound service time for stage k is S. Second, the function $g_k(SI)$ is the minimum holding cost for safety stock in a subnetwork with node set N_k, where we assume that the inbound service time for stage k is SI.

At node k (or stage k) for $1 \leq k \leq N - 1$, the algorithm determines either $f_k(S)$ or $g_k(SI)$, depending upon the location of the node with higher label that is adjacent to k. If $p(k)$ is downstream [upstream] of node k, then we evaluate $f_k(S)$ [$g_k(SI)$]. For node N, we can evaluate either functional equation.

To develop the functional equations, we first define the minimum inventory holding cost for the subnetwork with node set N_k as a function of both the inbound service time and the outbound service time at node k:

$$c_k(S, SI) = h_k\{D_k(SI + T_k - S) - (SI + T_k - S)\mu_k\} + \sum_{\substack{(i,k)\in A\\ i<k}} f_i(SI) + \sum_{\substack{(k,j)\in A\\ j<k}} g_j(S).$$

The first term is the holding cost for the safety stock at node k as a function of S and SI.

The second term corresponds to the nodes in N_k that are upstream of k. For each node i that supplies node k, we include the minimum inventory holding costs for the subnetwork with node set N_i, as a function of SI. The inbound service time to node k, SI, is an upper bound for the outbound service time for node i. We can show that $f_i(\ )$, the inventory holding costs for the subnetwork with node set N_i, is nonincreasing in the service time at node i. Hence, we equate the outbound service time at i to the inbound service time at k without loss of generality.

The third term corresponds to the nodes in N_k that are downstream of k. For each node j, $j \in N_k$ and $(k, j) \in A$, we include the minimum inventory holding costs for the subnetwork with node set N_j, as a function of S. The outbound service time for node k, S, is a lower bound for the inbound service time for node j. We can show that $g_j(\ )$, the inventory holding costs for the subnetwork with node set N_j, is nondecreasing in the inbound service time to node j; and thus we equate the inbound service time at j to the outbound service time at k without loss of generality.

We solve the following optimization by enumeration to find the functional equation $f_k(S)$:

$$f_k(S) = \min_{SI}\{c_k(S, SI)\}$$

$$\text{s. t. } \max(0, S - T_k) \leq SI \leq M_k - T_k, \text{ and } SI \text{ integer,}$$

where M_k is the maximum replenishment time for node k. The lower bound on SI comes from **P**, while the definition of M_k gives the upper bound.

The functional equation for $g_k(SI)$ is very similar in structure:

$$g_k(SI) = \min_{S}\{c_k(S, SI)\}$$

$$\text{s. t. } 0 \leq S \leq SI + T_k, \text{ and } S \text{ integer.}$$

If node k is a demand node, then we also constrain S by its maximum service time, i.e., $S \leq s_k$. The minimization can be done by enumeration.

Dynamic Program.
The dynamic programming algorithm is now as follows:

1. For $k := 1$ to $N - 1$
   2. If $p(k)$ is downstream of k, evaluate $f_k(S)$ for $S = 0, 1, \ldots, M_k$.
   3. If $p(k)$ is upstream of k, evaluate $g_k(SI)$ for $SI = 0, 1, \ldots, M_k - T_k$.
4. For $k := N$ evaluate $g_k(SI)$ for $SI = 0, 1, \ldots, M_k - T_k$.
5. Minimize $g_N(SI)$ for $SI = 0, 1, \ldots, M_N - T_N$ to obtain the optimal objective function value.

This procedure finds the optimal objective function value; we can find an optimal set of service times by the standard backtracking procedure for a dynamic program.

To summarize, at each stage of the dynamic program, we find the minimum inventory holding costs for the subnetwork with node set N_k, as a function of a state variable. The state variable depends on the direction of the arc that connects the subnetwork N_k to the rest of the network. When the connecting arc originates in N_k, then the state variable is the outbound service time (step 2); otherwise, the state variable is the inbound service time (step 3). We number the nodes so that we have previously determined the functions required to evaluate either $f_k(S)$ or $g_k(SI)$. At stage N (step 4), we determine the inventory costs for the entire network as a function of the inbound service time to node N. At step 5, we optimize over the inbound service time to find the optimal inventory cost.

The computational complexity of the algorithm is of order NM^2 where M is the maximum service time, which is bounded by the sum of the production leadtimes $\sum_{j=1}^{N} T_j$. We have implemented the algorithm for a PC in the C++ programming language. The run times for real problems with 25 to 30 nodes are effectively instantaneous on a Pentium PC with a 100 megahertz Intel processor.

6. Application

This section presents an application of the model at the Eastman Kodak Company. Starting in 1995, Kodak has applied the model to more than eleven finished products across two of its assembly sites within its equipment division. We first present the model's application to the internal supply chain for a high-end digital camera,[2] and then summarize Kodak's financial results, as of 1997 year-end.

Product Background.
The key subassemblies for the digital camera are a traditional 35 mm camera, an imager, and a circuit-board assembly. The 35mm camera is procured from an outside vendor. The imager (a charge-coupled device) and the circuit-board assembly are produced internally. The 35mm camera supplies the lens, shutter, and focus functions for the digital camera. The imager captures and digitizes the picture, and the circuit-board assembly processes and stores the image. To produce the digital camera, the back of the 35mm camera is removed and replaced with a housing containing the imager

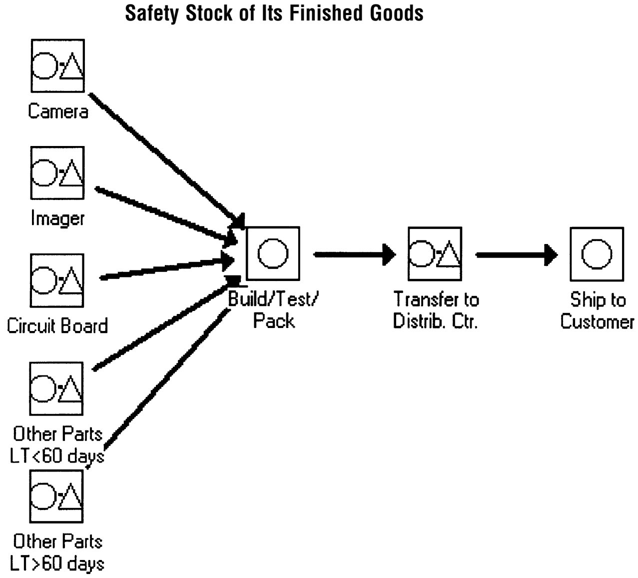
[2]The data presented in this section has been altered to protect proprietary information. However, the resulting qualitative relationships and insights drawn from this example are the same as they would be from using the actual data.

and circuit board. The camera is then tested to make sure that there are no defects in the imager. Once the camera passes the quality tests, the product is shipped to the distribution center. From the distribution center, the camera is shipped to the final customers, which for our purposes are high-end photography shops and computer superstores.

In Figure 3, we provide a high-level depiction of this supply chain. In addition to the three key subassemblies, we include the remaining parts in order to accurately represent the product's cost structure; since there are nearly 100 additional parts in a camera, modeling them in any level of detail would greatly expand the size of the model. Hence, we group these parts into two aggregate stages of the supply chain, where one stage represents all of the parts with long procurement lead-times (greater than 60 days) and the other stage represents the short lead-time parts (less than 60 days).

We also aggregate certain operations. As seen in Figure 3, we combine the build operation for a camera with the test operation and the packing operation. The imager stage and circuit board stage are also aggregates as each represents the flow through a separate department. The circuit-board stage entails circuit board assembly and test. The imager stage consists of a semiconductor operation to produce wafers, followed by packaging and testing of the semiconductors, followed by an assembly operation.

Figure 3 Implemented Safety Stock Policy for Digital Camera. All Stages Have a Circle That Denotes the Processing Activity at the Stage. A Triangle Denotes That the Stage Holds a Safety Stock of Its Finished Goods



Implementation Approach.
The product's supply chain crosses several functional boundaries within Kodak. Functional areas like circuit-board assembly and imager assembly are separate departments and act as suppliers to an assembly group that performs final assembly and test. Distribution is a separate organization and owns the product once it leaves the final assembly area. To improve coordination across the departments, the equipment division at Kodak has set up product flow teams with the general charge to optimize their supply chains.

The product flow team for the digital camera relied on the model to identify opportunities for better coordination and improved asset utilization. The team implemented the model in phases. The implementation strategy was to start simple and get experience with the model; once there was some evidence of the utility of the model, the team extended the application in increments to capture more and more of the supply chain.

Phase One.
The initial goal was to optimize the safety stock levels for the stages that were under the direct control of the final assembly area. The decision to start with the final assembly area was based on the product's high material cost and its relatively simple supply chain structure, as described above. The (disguised) costs and production lead-times are:

Table 1 Phase One Digital Camera Information

Stage Name	Production Lead-Time	Cost Added
Camera	60	750
Imager	60	950
Circuit Board	40	650
Other Parts LT $<$ 60 days	60	150
Other Parts LT $>$ 60 days	150	200
Build/Test/Pack	6	250
Transfer to DC	2	50
Ship to Customer	3	0

The demand bound was estimated by Equation (1) where $\mu = 11$, $\sigma = 7$ and $k = 1.645$. From looking at historical demand and future demand estimates, Kodak felt that this function realistically captured the range of demand for which they wanted to use safety stock.

This demand characterization excluded large one-time orders from the government and some large corporations. These orders are typically for 200 – 300 units with delivery scheduled less than a month from when the order is placed. However, since there is advance warning about these orders and they are independent of the other demand for the product, we developed a separate anticipatory stock policy to deal with large, infrequent orders.

Marketing determined that the maximum service time to the final customer is five days.

Finally, the assembly group imposed the constraint that a safety stock of imagers must be held on-site at final assembly. Therefore, we set the service time for the imager stage to be zero; the effect of this constraint increased the total safety stock cost by 8.7%.

In the optimal solution, the subassembly stages, the aggregate parts stages, and the build/test/package stages hold safety stocks and quote zero service times. The ship-to-distribution and ship-to-customer stages each quote their maximum feasible service times, two and five days, respectively. The annual holding cost for the safety stock is $78,000. Thus, the optimal solution holds an inventory of components, subassemblies, and completed cameras at the manufacturing site, but holds no inventory in the distribution center. In effect, the distribution center would act only as an order processing and transshipment center. This is feasible since it is possible to get the product from the assembly area through the distribution center and to the final customer within the maximum service time of five days.

The product flow team decided to explore some near-optimal solutions because they felt that there were some additional organizational constraints not captured in the model; in particular, distribution would want to hold safety stock on-site. To ameliorate the situation, the team suggested that both manufacturing and distribution hold safety stock and quote zero service times. However, the model showed that the cost for the safety stock would increase to $89,000. The team also investigated a policy in which the distribution center would hold safety stock but the manufacturing site would not. The safety stock cost for this policy was $81,000, which was deemed to be acceptable as it was quite close to the unconstrained optimum and satisfied distribution's desire to hold inventory. This policy, as shown in Figure 3, was implemented at the end of phase one of the application.

Phase Two.
After the initial phase, the product flow team expanded the model to incorporate the internal supply chain for the imager. The resulting supply chain is shown in Figure 4.

Prior to this study, safety stocks of (in-process) imagers had been held at each stage of the supply chain. By application of the model, the team decided to remove safety stocks from two stages in the supply chain for the imagers, as shown in the figure. This required some increase in the downstream safety stocks of finished imagers, but overall the supply chain's safety stock of imagers (measured in terms of finished imagers) was more than halved.

Now that the model has been successfully piloted with an internal supplier, the product flow team is in the process of extending the model to incorporate other key internal and external suppliers.

Financial Results.
Table 2 contains the financial summary for two assembly sites that use the model. Site A has applied the

Figure 4 **Digital Camera Supply Chain**

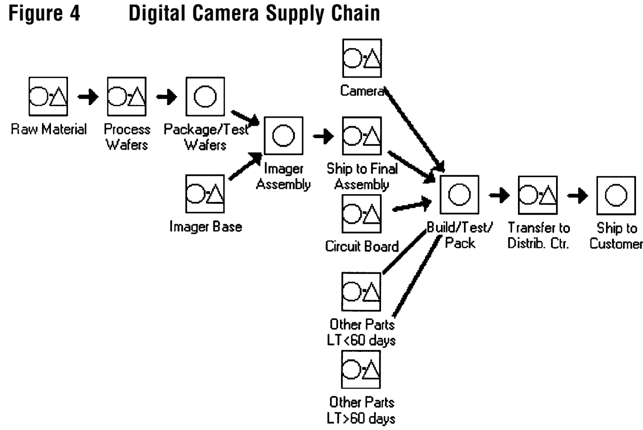


Table 2 Financial Summary for Assembly Sites *A* and *B*

Assembly Site *A*	*Y/E* 95	*Y/E* 96	*Y/E* 97
Worldwide FGI	\$6.7*m*	\$3.3*m*	\$3.6*m*
Raw Material & WIP	\$5.7*m*	\$5.6*m*	\$2.9*m*
Delivery Performance	80%	94%	97%
Manufacturing Operation	MTS	RTO	RTO
Assembly Site *B*			
Worldwide FGI	\$4.0*m*	\$4.0*m*	\$3.2*m*
Raw Material & WIP	\$4.5*m*	\$1.6*m*	\$2.5*m*
Delivery Performance	Unavailable	78%	94%
Manufacturing Operation	MTS	RTO	RTO

model to each of its eight products and Site B has applied the model to each of its three product families. The sales volume has remained relatively constant over the three years.

At the start of 1996, the sites moved from a make-to-schedule (MTS) to a replenish-to-order (RTO) system. The modeling effort began at the end of 1995 and was used to help guide the transition to replenish-to-order. The increase in worldwide finished goods inventory for 1997 is due to a marketing promotion that was underway in Europe. By our estimate, this promotion increased the finished goods inventory by as much as \$0.5 million. In the first year of the project, the emphasis was on reducing the areas directly under the control of final assembly. Over the following year, the effort was on reducing the raw material costs and WIP in the manufacturing supply chain. The total value of the inventory for these products has been reduced by over one third over the two years.

Kodak's product flow teams have also used the model for a variety of purposes other than setting safety stocks. Some products have tens of components with long procurement lead-times. The model has helped to prioritize the suppliers with whom to work to reduce these lead-times. The teams have used the model to determine the cost effectiveness of lead-time reduction efforts in manufacturing. One can compare the investment required to reduce a lead-time versus the cost savings from the reductions in pipeline and safety stock cost. Finally, manufacturing and marketing personnel have used the model to help quantify the cost of quoting a specific maximum service time to the final customer. With the model, the supply-chain team can accurately estimate the costs of a one-day, one-week, or two-week guaranteed service time to the customer, and weigh the costs of the policy against the marketing benefits of the policy.

Another benefit of the model is that it provides a common, objective framework with which a cross-functional supply-chain team can work. In particular, we note that it provides a standard terminology and set of assumptions for these teams to use as they work together to improve or optimize a supply chain. As such the model has been a very effective communication vehicle or platform.

7. Conclusion

In this paper we introduce and develop a model for positioning safety stock in a supply chain. We model the supply chain as a network, where the nodes of the network are the stages of a supply chain. We assume that each stage uses a base-stock policy to control its inventory. We also assume that each stage quotes a service time to its customers, both internal and external, and that each stage provides 100% service for these quoted service times. Finally we assume that external customer demand is bounded.

We show how to evaluate the inventory requirements at each stage as a function of the service times. For supply chains that can be modeled as spanning trees, we develop an optimization algorithm for finding the service times that minimize the holding cost for the safety stock in the supply chain.

As a form of validation, we describe an application of the model at Kodak to an internal supply chain for a digital camera. This application helped Kodak to reposition its inventories in this supply chain to reduce its inventory and increase its service performance. In particular, Kodak realized the benefit from creating a few strategic locations to hold safety stocks rather than spreading the safety stock across the entire supply chain. We have also applied the model to a number of other related products at Kodak and at two other companies (Black 1998, Coughlin 1998, Felch 1997, Wala 1999).

As with any research, we end with a number of unresolved issues and new questions. We discuss these in the relative order of importance, based on our experience in applying the research to date.

Stochastic Lead-Times. We assume that associated with each stage is a known, deterministic lead-time. In practice, this is often not true; for instance, component procurement times are often highly uncertain. It will be of value to capture this in the model. We know how to extend the model in an approximate way for stages that procure raw materials or components from an outside vendor. In effect, for such a stage we just need to approximate its inventory requirements as a function of the outbound service time quoted by the stage and the stochastic procurement time. But it is less clear how to extend the model, either exactly or approximately, to permit stochastic lead-times at stages whose function is not procurement.

Non-Stationary Demand. We assume that the end-item demand processes are stationary. Yet virtually all of the products with which we have worked have short lifetimes over which demand is never really stationary. In practice, one runs the model under various (stationary) scenarios to see how sensitive the safety stock is to the demand characterization (Coughlin 1998). Fortunately, we have found empirically that where the model locates safety stock in the supply chain is fairly insensitive to the demand. The size of the safety stock, though, does depend directly on the demand characterization. We currently are conducting research to understand these observations better, to extend the model to treat non-stationary demand.

Different Review Periods. We assume that each stage operates with a base-stock policy with a common review period. In many supply chains different stages will operate with different reorder frequencies. That is, whereas one stage may place replenishment orders on a daily basis, another stage may do this weekly. In other cases, a stage may operate with a continuous-review policy so that the time between orders varies. We can extend the model to evaluate nested periodic-review base-stock polices in which whenever one stage reorders, all stages downstream also reorder. That is, the review period for an upstream stage is an integer multiple of the review period of its immediate customers. However, we have not yet built the software to implement this extension, as it is a major programming task and it may only be a partial fix to the issue.

Capacity Constraints. In the model we ignore capacity constraints. For certain stages in a supply chain, the consideration of a capacity limit may be necessary in order to get a credible model for determining safety stock requirements. At this time, we do not have a good idea of how to add this to the model.

General Networks. In this paper, we have developed and implemented an optimization algorithm for supply chains that can be modeled as spanning trees. We describe in Graves and Willems (1998) how to extend this algorithm to general networks. However, we have not done a systematic study of this extension beyond some exploratory work. More research is needed to test and refine these ideas as well as uncover better approaches.[3]

Appendix

In this appendix we examine the assumption that each internal stage quotes a guaranteed service time to its customers. To get some insight, we consider a serial system for which we can determine the optimal policy when we relax the assumption of guaranteed service times for internal customers. We then compare the inventory holding costs for the optimal policies with and without this assumption for a small set of test problems.

Consider a serial supply chain with N stages where stage 1 is the demand node and stage i supplies stage $i - 1$ for $i = 2, \ldots, N$. The same assumptions hold as in the original model, except that we do not require guaranteed service times to internal customers. There are no restrictions on the service level that stage i provides to its customer, stage $i - 1$ for $i = 2, \ldots, N$; rather, these internal service levels depend on the base stocks, which are chosen to minimize the inventory holding costs for the entire supply chain. We do assume that stage 1 provides a 100% service level to the external customer, and, without loss of generality, we assume that the service time quoted to the external customer is zero.

For ease of presentation, we assume that $\phi_{i,i-1} = 1$ for $i = 2, \ldots, N$. We let $d(t)$ denote the end-item demand in period t; $d(a, b]$; denote the end-item demand over the time interval $(a, b]$; and $D(\tau)$ denote the maximum possible end-item demand over a time interval of τ periods.

[3]This research has been supported in part by the Eastman Kodak Company; by the MIT Leaders for Manufacturing Program, a partnership between MIT and major U.S. manufacturing firms; and by the MIT Integrated Supply Chain Management consortium. The authors acknowledge and thank Dr. John Ruark who contributed significantly to this research effort; John played a lead role in developing the software application for implementing the results of this research. We also wish to thank the editors and referees for their very helpful and constructive feedback on earlier versions of the paper.

For each stage i, we define $Q_i(t)$ to be the shortfall or backlog at time t, namely the amount that has been ordered by the stage's customer but not yet delivered. We assume at $t = 0$, $I_i(t) = B_i \geq 0$ and $Q_i(t) = 0$ for all stages.

We can show for $i = 1, 2, \ldots, N$ that the on-hand inventory and backlog at time t are:

$$I_i(t) = [B_i - d(t - T_i, t) - Q_{i+1}(t - T_i)]^+,$$
$$Q_i(t) = [d(t - T_i, t) + Q_{i+1}(t - T_i) - B_i]^+, \qquad \text{(A1)}$$

where $[x]^+ = \max(0, x)$, and $Q_{N+1}(t) = 0$ by definition. Equation (A1) requires that each stage has a deterministic lead-time and that each stage follows a base-stock policy in which, for each period, each stage observes end-item demand and places a replenishment order for this amount. The essence of the argument is to observe that the net inventory on hand at a stage equals the stage's base stock minus the inventory on order. For stage i, the inventory on order at time t is the backlog as of time $t - T_i$, plus all of the demand over the interval $(t - T_i, t]$.

From Equation (A1) we can show by induction that for $i = 1,2, \ldots, N$,

$$Q_i(t) = \max[0,\ d(t - T_i, t) - B_i,\ d(t - T_i - T_{i+1}, t) - B_i - B_{i+1}, \ldots, d(t - T_i - T_{i+1} - \cdots - T_N, t) - B_i - B_{i+1} - \cdots - B_N]. \qquad \text{(A2)}$$

In order for the supply chain to provide 100% service to the external customer, we must never have a backlog at stage 1; thus, we must select base stocks so that $Q_1(t) = 0$ for all t. From Equation (A2) we see that $Q_1(t) = 0$ is assured if the base stocks satisfy the following constraints:

$$B_1 + B_2 + \cdots + B_i \geq D(T_1 + T_2 + \cdots + T_i) \quad \text{for } i = 1, 2, \ldots, N. \qquad \text{(A3)}$$

Thus, if the base stocks satisfy Equation (A3), there will never be a shortfall at stage 1 and end-item demand will be satisfied with 100% service. As we assume that the demand bounds can be realized, then the constraint set (A3) provides not just sufficient but also necessary conditions for assuring 100% service for end-item demand.

In order to select the base stocks to minimize the inventory holding costs for the supply chain, we must develop an expression for the inventory holding costs; we note from Equation (A1) that the net inventory on hand at stage i is given by:

$$I_i(t) - Q_i(t) = B_i - d(t - T_i, t) - Q_{i+1}(t - T_i). \qquad \text{(A4)}$$

From Equation (A4) we can write the inventory holding costs for the supply chain as:

$$\sum_{i=1}^{N} h_i E[I_i(t)] = \sum_{i=1}^{N} h_i \{B_i - \mu T_i + E[Q_i(t)] - E[Q_{i+1}(t - T_i)]\} \qquad \text{(A5)}$$

where μ is the expected demand rate, and $E[\ ]$ denotes expectation.

We now pose an optimization problem to select the base stocks; namely, we minimize Equation (A5) subject to Equation (A3) and nonnegativity constraints on the base stocks. After dropping constant terms in Equation (A5) and noting that $Q_1(t) = 0$ for any feasible solution, we write the optimization as

$$\min \sum_{i=1}^{N} h_i B_i - \sum_{i=2}^{N} e_{i-1} E[Q_i]$$

P* s.t.

$$B_1 + B_2 + \cdots + B_i \geq D(T_1 + T_2 + \cdots + T_i) \quad \text{for } i = 1, 2, \ldots, N,$$
$$B_i \geq 0 \quad \text{for } i = 1, 2, \ldots, N,$$

where $e_i = h_i - h_{i+1}$ is the echelon holding cost. We note from Equation (A2) that $E[Q_i]$ is a nonlinear function of $B_i, \ldots, B_N$ for $i = 1, 2, \ldots, N$.

Our main result is that there is an optimal solution to **P*** in which all the constraints in Equation (A3) are binding. More formally we state the following:

RESULT. If the echelon holding costs are nonnegative and if $D(\ )$ is a nondecreasing function, then an optimal solution to **P*** is given by

$$B_1 = D(T_1),$$
$$B_i = D(T_1 + \cdots + T_i) - D(T_1 + \cdots + T_{i-1}) \quad \text{for } i = 2, \ldots, N. \qquad \text{(A6)}$$

PROOF. We note that the solution given by Equation (A6) is nonnegative and satisfies the constraints in Equation (A3) as equalities; thus it is a feasible solution to **P***. To prove that this is also an optimal solution, we will argue that there must be an optimal solution in which the constraints in Equation (A3) are binding.

Suppose we have a solution $B_1^*, \ldots, B_N^*$ such that Equation (A3) holds as a strict inequality for one or more constraints. Suppose the kth constraint is the first constraint that is not binding and that $k < N$; we will treat the case when $k = N$ later. Thus, we assume

$$B_1^* + B_2^* + \cdots + B_i^* = D(T_1 + T_2 + \cdots + T_i) \quad \text{for } i = 1, 2, \ldots, k - 1 \text{ and}$$
$$B_1^* + B_2^* + \cdots + B_k^* > D(T_1 + T_2 + \cdots + T_k).$$

We now define a new solution $B_1^{**}, \ldots, B_N^{**}$ in which the first k constraints are satisfied as equalities, and show that its objective value is no worse than that for $B_1^*, \ldots, B_N^*$:

$$B_i^{**} = B_i^* \quad \text{for } i = 1, \ldots, N \text{ and } i \neq k, k + 1,$$
$$B_k^{**} = B_k^* - \Delta$$
$$B_{k+1}^{**} = B_{k+1}^* + \Delta,$$

where

$$\Delta = B_k^* - D\left(\sum_{i=1}^{k} T_i\right) + D\left(\sum_{i=1}^{k-1} T_i\right).$$

We first observe that $\Delta > 0$ due to the supposition that the solution $B_1^*, \ldots, B_N^*$ satisfies the kth constraint in Equation (A3) as a strict inequality. Thus, we have $B_{k+1}^{**} \geq 0$. We also see that $B_k^{**} \geq 0$ since $D(\ )$ is nondecreasing. Hence the new solution $B_1^{**}, \ldots, B_N^{**}$ is nonnegative. By construction, the new solution satisfies the kth constraint as an equality, and there are no changes in the remaining constraints. Thus, the new solution $B_1^{**}, \ldots, B_N^{**}$ is a feasible solution.

To express the objective function for the new solution, we decompose it into two parts. The first part of the objective function is

$$\sum_{i=1}^{N} h_i B_i^{**} = (-h_k + h_{k+1})\Delta + \sum_{i=1}^{N} h_i B_i^* = -e_k\Delta + \sum_{i=1}^{N} h_i B_i^*. \quad \text{(A7)}$$

For the second part of the objective function, let $E[Q_i]^*$ and $E[Q_i]^{**}$ denote the expected backlog at stage i for the first and second solution. Then we find from Equation (A2) that

$$E[Q_i]^{**} = E[Q_i]^* \quad \text{for } i > k + 1,$$

$$E[Q_i]^* \leq E[Q_i]^{**} \leq E[Q_i]^* + \Delta \quad \text{for } i < k + 1, \text{ and}$$

$$E[Q_{k+1}]^* \geq E[Q_{k+1}]^{**} \geq E[Q_{k+1}]^* - \Delta.$$

Thus, for nonnegative echelon holding costs, we can bound the second part of the objective function as follows:

$$-\sum_{i=2}^{N} e_{i-1} E[Q_i]^{**} \leq -\sum_{i=2}^{N} e_{i-1} E[Q_i]^* + e_k\Delta. \quad \text{(A8)}$$

By combining Equations (A7) and (A8), we see that the objective function for the second solution is no greater than the objective for the first. Thus, we have found a new solution in which the first k constraints in Equation (A3) are binding and whose objective value is no worse than that for the first solution. This argument can be continued in this fashion to construct a solution in which the first $N - 1$ constraints in Equation (A3) are binding and whose objective value is no worse than that for the solution $B_1^*, \ldots, B_N^*$. The argument for the case when $k = N$ is similar in structure but easier; we just have to reduce the base stock for stage N until the Nth constraint is binding, which can be done with no penalty to the objective function.

Hence, there is a feasible solution that satisfies all the constraints in Equation (A3) as equalities and that has an objective value no higher than that for the solution $B_1^*, \ldots, B_N^*$. Furthermore, this new solution must be given by Equation (A6), as it is easy to see that it is the unique binding solution to Equation (A3). Finally we conclude that Equation (A6) must be an optimal solution, as its objective value equals or is less than that for any interior solution $B_1^*, \ldots, B_N^*$. This completes the proof.

We note that the optimal base-stock policy does not depend at all on the holding costs. All we need to know is that the holding costs do not decrease as we move down the supply chain, closer to the customer. We also note that this result generalizes to assembly systems by means of the transformation given by Rosling (1989); namely, we can transform an assembly system into an equivalent serial system, and the result applies.

We use this result to compare the performance of the base stock policies with and without the assumption of guaranteed service times for internal customers. The test problems were all for a 3-stage serial system; the problems differed according to their demand process, their production lead-times, and their holding costs.

For the demand process, we start with a Poisson demand distribution with mean λ and with a specified percentile α to truncate the demand. For each time window of length τ, we set the demand bound $D(\tau)$ as the smallest integer such that the cumulative probability for the Poisson random variable with mean $\lambda\tau$ exceeds α. We then normalize the demand distribution over the truncated range. We consider four possible demand processes: $\lambda = 10$, $\alpha = 0.90$; $\lambda = 10$, $\alpha = 0.98$; $\lambda = 50$, $\alpha = 0.90$; $\lambda = 50$, $\alpha = 0.98$.

We permit three settings for the production lead-times and three settings for the holding costs, as follows:

$$(T_1, T_2, T_3) = (4, 4, 4); \quad (1, 3, 8); \quad (8, 3, 1).$$

$$(h_1, h_2, h_3) = (1, 0.5, 0.2); \quad (1, 0.66, 0.33); \quad (1, 0.8, 0.5).$$

By evaluating all combinations we have a total of 36 test problems. For each test problem we determine the optimal policy for the model with guaranteed internal service times and the optimal policy (given by the result above) for the model without this requirement. For each instance, we evaluate the base stocks, the safety-stock holding cost and the total inventory holding cost. The safety stock holding cost is given by the objective function of **P** for the model with guaranteed internal service times and by Equation (A5) for the model without this requirement. The total inventory holding cost is the sum of the safety-stock holding cost plus the pipeline-stock holding cost. The expected pipeline stock at stage i is μT_i; we assume that the holding cost for the pipeline stock at stage i is $(h_i + h_{i+1})/2$.

For the 36 test problems we find that the safety-stock holding cost for the model with guaranteed internal service times is on average 26% higher than that for the model without this requirement; the range is between 7% and 43%. The size of the gap is insensitive to the choice of demand process. However, the gap becomes larger as the production lead-time at stage 1 increases and as the echelon holding cost at stage 1 increases.

The impact on the total inventory holding cost is less dramatic. The difference in holding costs is 4% on average, with a range from less than 1% to 14%. The gap increases as the holding cost of the pipeline stock decreases, namely as the production lead-time at stage 1 decreases and as the demand rate decreases.

From the limited computational study we see that there can be a significant increase in safety stock due to the assumption of guaranteed internal service times. Relative to the total inventory, this increase does not look as large. Nevertheless, there is a cost in terms of higher inventories from the requirement of guaranteed internal service times. This cost needs to be considered in light of the practical benefits, as discussed in the body of the paper, from imposing this requirement. Based on our observations from industrial projects, this requirement, and the resulting increase in safety stock, has not been an issue as the assumption of guaranteed internal service times is already ingrained in practice.

References

Axsäter S. 1993. Continuous review policies for multi-level inventory systems with stochastic demand. S. C. Graves. A. H. Rinnooy

Kan, P. H. Zipkin, eds. *Handbooks in Oper. Res. and Management Sci. Vol 4., Logistics of Production and Inventory*. North-Holland Publishing Company, Amsterdam, The Netherlands. Chapter 4.

Black, B. E. June 1998. Utilizing the principles and implications of the base stock model to improve supply chain performance. S.M. Thesis, Leaders for Manufacturing Program, MIT, Cambridge, MA.

Coughlin, R. L. June 1998. Optimization and measurement of a world-wide supply chain. S.M. Thesis, Leaders for Manufacturing Program, MIT, Cambridge, MA.

Diks, E. B., A. G. de Kok, A. G. Lagodimos. 1996. Multi-echelon systems: A service measure perspective: *European J. Oper. Res.*, **95** 241–263.

Ettl, M., G. E. Feigin, G. Y. Lin, D. D. Yao. 2000. A supply network model with base-stock control and service requirements. *Oper. Res.* **48**, March-April.

Federgruen, A. 1993. Centralized planning models for multi-echelon inventory systems under uncertainty. S. C. Graves, A. H. Rinnooy Kan, P. H. Zipkin, eds. *Handbooks in Operations Research and Management Science, Vol 4., Logistics of Production and Inventory*, North-Holland Publishing Company, Amsterdam, The Netherlands, Chapter 3.

Felch, J. A. June 1997. Supply chain modeling for inventory analysis. S. M. Thesis, Leaders for Manufacturing Program, MIT, Cambridge, MA.

Gallego, G., P. Zipkin. 1999. Stock positioning and performance estimation in serial production-transportation systems. *Manufacturing & Service Oper. Management*. **1** 77–88.

Glasserman, P., S. Tayur. 1995. Sensitivity analysis for base-stock levels in multiechelon production-inventory systems *Management Sci.* **41** 263–281.

Graves, S. C. 1988. Safety stocks in manufacturing systems. *J. of Manufacturing and Oper. Management* **1** 67–101.

——, D. B. Kletter, W. B. Hetzel. 1998. A dynamic model for requirements planning with application to supply chain optimization. *Oper. Res.* **46** S35–S49.

——, S. P. Willems. 1996. Strategic safety stock placement in supply chains. *Proceedings of the 1996 MSOM Conference*, Hanover, NH.

——, ——. August 1998. Optimizing strategic safety stock placement in supply chains. Working Paper available from http://web.mit.edu/sgraves/www/papers/.

Inderfurth, K. 1991. Safety stock optimization in multi-stage inventory systems. *Interna. J. of Production Econom.* **24** 103–113.

—— 1993. Valuation of leadtime reduction in multi-stage production systems. G. Fandel, T. Gulledge, A. Jones, eds. *Oper. Res. in Production Planning and Inventory Control*, Springer, Berlin, Germany, 1993, 413–427.

—— 1994. Safety stocks in multistage, divergent inventory systems: A survey. *International J. of Production Economics* **35** 321–329.

——, S. Minner. 1998. Safety stocks in multi-stage inventory systems under different service measures. *European J. Oper. Res.* **106** 57–73.

Kimball, G. E. 1988. General principles of inventory control. *J. of Manufacturing and Oper. Management*. **1** 119–130.

Lee, H. L., C. Billington. 1993. Material management in decentralized supply chains. *Oper. Res.*, **41** 835–847.

Luenberger, D. G. 1973. *Introduction to Linear and Nonlinear Programming*. Addison Wesley, Reading, MA, 1973.

Minner, S. 1997. Dynamic programming algorithms for multi-stage safety stock optimization. *OR Spektrum* **19** 261–271.

Rosling, K. 1989. Optimal inventory policies for assembly systems under random demands. *Oper. Res.*, **37** 565–579.

Simpson, K. F. 1958. In-process inventories. *Oper. Res.*, **6** 863–873.

Van Houtum, G. J., K. Inderfurth, W. H. J. Zijm. 1996. Materials coordination in stochastic multi-echelon systems. *European J. of Oper. Res.*, **95** 1–23.

Wala, T. June 1999. Build-to-order/direct ship model for professional digital cameras. S. M. Thesis, Leaders for Manufacturing Program, MIT, Cambridge MA.

The consulting Senior Editor for this manuscript was Paul Zipkin. This manuscript was received on June 1, 1998, and was with the authors 127 days for 3 revisions. The average review cycle time was 53 days.