Strategic Safety Stock Placement In Supply Chains

Stephen C. Graves • Sean P. Willems

Massachusetts Institute of Technology, A. P. Sloan School of Management, Cambridge, Massachusetts 02139

Abstract

We consider a multi-stage production/distribution supply chain subject to stochastic demand. We formulate an optimization problem to determine where to place decoupling inventories, so-called strategic inventories, across the supply chain so as to minimize inventory holding costs subject to a service constraint for satisfying customer demand. We assume demand is normally distributed. For each stage, we know its lead time, which we assume is deterministic, and we know the cost added at the stage. There are no capacity constraints in the supply chain, and each stage quotes and guarantees a service time by which it will supply its immediate successors. These service times are decision variables for the optimization model. Associated with each stage is a service level, denoting the percentage of time that the guaranteed service time is to be met from inventory. Finally each stage operates with a base-stock control policy; that is, each period each stage orders a quantity equal to its demand.

1 Introduction

An emerging principle for the management of supply chains is that a supply-chain perspective provides the opportunity for significant savings in inventories from better coordination and communication across the supply chain. One component of this savings is due to a coordinated strategy for setting safety stocks to protect against uncertainty and variability; that is, a supply-chain perspective can avert some of the local suboptimization that occurs, for instance, when each stage of a manufacturing or distribution process independently determines its own safety stocks. In this paper we describe ongoing research to develop tools and general principles for determining how to set safety stocks in a supply chain.

In particular, we address how to determine the optimal placement of safety stock inventories in a supply chain subject to uncertain demand. In Section 2 we formulate a model, originally given in Simpson (1958), for the simplest supply chain, a serial line, and present a solution procedure to find the optimal safety stocks. In Section 3 we extend the Simpson model to more general multi-stage supply chains, including assembly and distribution networks. We conclude the paper in Section 4 with a status report on this research project. We are currently testing the model in two industrial settings, and briefly comment on these ongoing case studies in this section. We also describe the focus of ongoing work to improve the solution algorithms and to extend the model to more realistic assumptions.

Related work on determining the inventory requirements for a supply chain include Lee and Billington (1993) (and the references therein) and Graves et al. (1996). This paper differs from the earlier work in terms of the underlying model assumptions, which result in our focus on where to place strategic inventories that completely decouple the upstream part of the supply chain from the downstream. In contrast Lee and Billington determine how much inventory is needed at each stage in the supply chain, so as to minimize total inventory. And Graves et al. determine the safety stock for a supply chain that is subject to dynamic requirements planning.

2 Serial Line

We first review the Simpson model of a serial production system and then present a solution procedure. In section 2.1 we detail the model's assumptions, formulation, and the notation we will be using throughout the paper. In section 2.2 we describe how the problem can be solved by dynamic programming.

2.1 Model Assumptions

We consider an N-stage serial system, where stage i is the immediate upstage stage or supplier for stage i+1, for i = 1, 2, ... N-1. Hence, stage 1 is the raw material stage and has no supplier; and stage N is the finished goods inventory node, from which customer demand is served. Each stage represents a major processing function in the supply chain; a typical stage might represent the manufacturing of a subassembly or the shipment of the finished product from a regional warehouse to the customer's distribution center. The only requirement for stage selection is that the process flow map that results from the selection captures the actual supply chain's characteristics; i.e., the model can be tactical in scope but it should not leave any gaps in how the product gets from one stage to another.

We assume demand each period is an independent normally-distributed random variable with mean μ and standard deviation σ. This is the only source of uncertainty in the model.

For each stage i, we know its production lead-time T_i, which we assume is deterministic, and we know the holding cost, h_i. There are no capacity constraints in the supply chain, and each stage i quotes and guarantees a service time S_i by which it will supply its immediate successor. We assume that the finished goods stage provides immediate service from inventory to the final customer; i.e. $S_N = 0$. But for the other stages, these service times are decision variables for the optimization model. Thus, if $S_i = 3$, say, then when an order is placed on stage i at time t, stage i will fulfill that order at time t + 3. Finally each stage operates with a (echelon) base-stock control policy; that is, each period each stage observes the current customer demand at stage N, and orders a quantity equal to replenish the current period's demand.

To determine the base stock level for a stage, we assume that associated with each stage is a service level, denoting the percentage of time that the guaranteed service time is to be met from inventory. Furthermore we do not attempt to model what happens when the guaranteed service time is violated, i. e., when demand exceeds some maximal level. In effect we assume that the base stocks will be sufficient as long as demand is within the range implied by the desired service level. For example, if the service level is 95%, then we assume that the base stocks will be set to cover a maximum demand equal to the 95th percentile of the demand distribution for any t-period time window; in effect, for setting the base stocks, we assume the maximum demand over t periods is $t\mu + k\sigma\sqrt{t}$ for k = 1.64. We ignore what happens when demand exceeds this level; that is, when demand might be regarded as being extraordinary, we assume that the operation would respond with an equally extraordinary measure, beyond the scope of the model and the assumed base-stock policy. See Simpson (1958) and Graves (1988) for further discussion of this assumption.

As a consequence of this assumption we can express the safety stock required by stage i as the following:

$$c_i(S_{i-1}, S_i) = k h_i \sigma \sqrt{S_{i-1} + T_i - S_i}$$

where k is the safety factor implied by the specified service level (e. g., k = 1.64 for 95% service level). In the above expression, $S_{i-1} + T_i$ is the replenishment lead time for stage i, since it takes S_{i-1} time units for stage i to be supplied by its upstream stage, and another T_i time units for stage i to complete its processing. The safety stock at stage i must cover the variability in demand over the net replenishment time, namely the difference between the replenishment time for stage i ($S_{i-1} + T_i$) and the service time promised by stage i (S_i). Since without capacity constraints there is no reason for stage i to promise a service time longer than its own replenishment time, we assume here that $S_i \le S_{i-1} + T_i$.

With these observations and assumptions, we can now formulate the following optimization problem for finding the optimal service times, or equivalently safety stocks, for each stage:

$$\min \sum_{i=1}^{N} h_i I_i$$

s.t.

$$I_i = k\sigma\sqrt{S_{i-1} + T_i - S_i} \qquad i = 1, \ldots, N$$

$$0 \le S_i \le S_{i-1} + T_i \qquad i = 1, \ldots, N$$

where I_i denotes the expected safety stock at stage i and S_0 is assumed to be 0. This problem formulation was first given by Simpson, who also showed that there is an optimal extreme point solution such that $S_i^* = 0$ or $S_i^* = S_{i-1}^* + T_i$ for all i = 1, 2, ... N-1 (S_N, the service time for the customer, equals 0 by assumption). Thus, there is an "all or nothing" optimal solution; either a stage has no safety stock ($S_i^* = S_{i-1}^* + T_i$) or the stage has sufficient safety stock ($S_i^* = 0$) to decouple it from its downstream stage.

2.2 Solution Procedure

The serial line case can be solved to optimality using dynamic programming or equivalently solving a shortest path problem. The dynamic program is a forward recursion starting at stage 1 and proceeding to stage N. For each stage, the algorithm finds the service time from the upstream stage that minimizes the cost of the current stage quoting a given service time; this procedure is repeated for each possible service time that stage i can quote. Define $f_i(S_i)$ as the optimal value of the network from stage 1 up to and including stage i given stage i quotes a service time of S_i.

$$f_i(S_i) = \min_{0 \le S_{i-1} \le \sum_{j=1}^{i-1} T_j} \left(f_{i-1}(S_{i-1}) + c_i(S_{i-1}, S_i) \right) \qquad 0 \le S_i \le \sum_{j=1}^{i} T_j \tag{1}$$

$$c_i(S_{i-1}, S_i) = kh_i\sigma\sqrt{S_{i-1} + T_i - S_i} \qquad i = 1, \ldots, N \tag{2}$$

Equation 1 is the optimal cost-to-go function for a given service time S_i. Equation 2 is just the safety stock cost at stage i given stage i quotes a service time of S_i and is quoted a service time of S_{i-1}.

We can improve the computational efficiency of the dynamic program by exploiting the observation that $S_i^* = 0$ or $S_i^* = S_{i-1}^* + T_i$. In particular, for $S_N = 0$, we can solve the problem as a shortest path from node 0 to node N on an N+1 node network with arcs (i, j) for all i < j, and i, j = 0, 1, 2, ... N. The cost of arc (i, j) is the inventory holding cost for having a decoupling inventory at stage i, assuming that the next upstream decoupling inventory is at stage j.

3 Extensions To Serial Case

There are several extensions to the serial case. In section 3.1, we formulate the optimization problem for finding the service times for an assembly network. We then discuss how to solve this problem. In Section 3.2 and 3.3, we address how to approach distribution networks, and more general networks, respectively.

3.1 Assembly Networks

Suppose that we can represent the supply chain as an acyclic network, given by a graph G where N(G) is the node set and A(G) is the arc set. There is a one-to-one mapping between the stages of the supply chain and the nodes in N(G). There is an arc between stage i and stage j, i. e., (i, j) ∈ A(G), if and only if stage i is a direct supplier to stage j. This section considers the assembly network case; that is, each stage has at most one successor (see Figure 1).

To formulate the optimization problem, for all (i, j) ∈ A(G) we define a service time S_i as the service time that stage i quotes to stage j. Since in an assembly network stage i feeds only one downstream stage, this notation is sufficient. As in the serial line case, the service times S_i will be the decision variables for the optimization problem.

For each stage i, we define L_i to be replenishment lead time for stage i; since there may be several upstream suppliers to stage i, we note that the replenishment lead time equals the production lead time at stage i, plus the longest service time of its suppliers: $L_i = \max \{ S_j + T_i \}$ where the maximization is over all arcs (j, i) ∈ A(G).

The formulation is now given by:

$$\min \sum_{i=1}^{N} h_i I_i$$

s.t.

$$
\begin{aligned}
& I_i = k\sigma\sqrt{L_i - S_i} && i = 1,\ldots,N \\
& L_i \geq S_j + T_i && \forall\ (j,i) \in A(G) \\
& 0 \leq S_i \leq L_i && i = 1,\ldots,N
\end{aligned}
$$

The first step in solving assembly networks is to identify the echelons in the network. Informally, this can be done as follows. First condense all the serial lines into super nodes. Then perform a breadth first search from Stage N to all the remaining stages. By definition of our aggregating procedure, each remaining stage must be in the echelon that is one higher than the echelon of the predecessor stage.

Figure 1 shows the echelon structure of an assembly network. In this example, stage 13 is the Nth stage and all of the serial lines are to be condensed except for the lines composed of only one stage; i.e., stages 4 and 6.

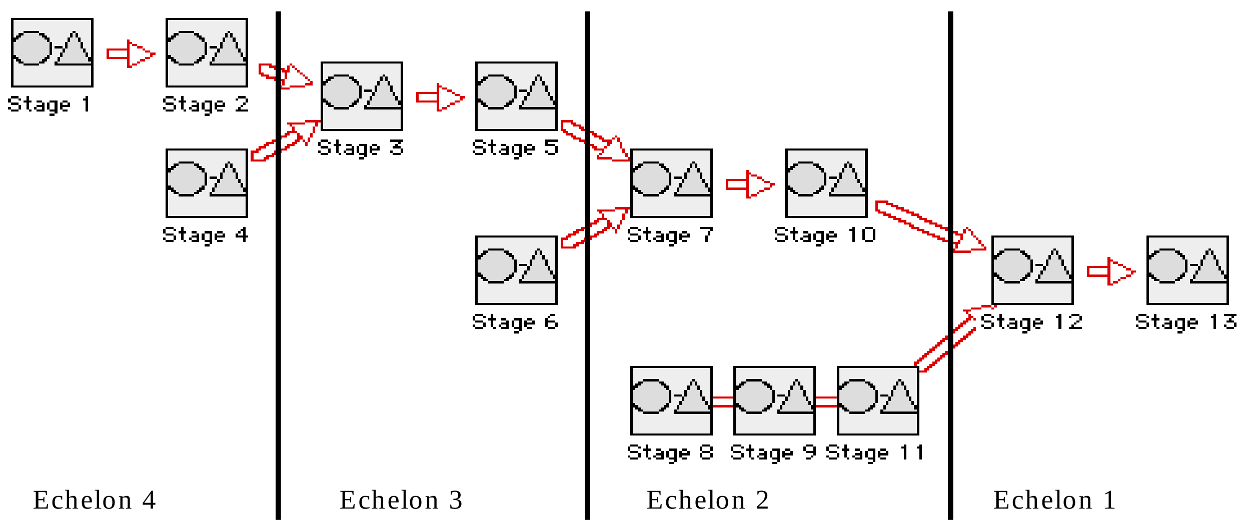

Figure 1: Assembly network broken into echelons

The algorithm to determine the service times for an assembly network is as follows. First, determine the echelon structure of the network. Second, start at the highest echelon number and work toward echelon 1, Within each echelon, loop over the serial lines in the echelon and solve the following cost-to-go function for each stage:

$$f_j(S_j) = \min_{0 \le s \le D_j} \left(\sum_{\{i:(i,j) \in A(G)\}} f_i(s) + c_j(s, S_j) \right) \qquad (3)$$

where D_j is the longest possible time path to stage j. The cost-to-go function $f_j(S_j)$ is the minimum holding cost for the inventory at stage j and all upstream stages to stage j, given that stage j quotes a service time equal to S_j. Within the above expression, $c_j(s, S_j)$ denotes the inventory holding cost at stage j, given that the longest supplier service time is s and given that stage j quotes a service time of S_j to its downstream stage. For a given service time for stage j, the algorithm loops over all possible incoming service times and finds the minimum cost solution for stage j and its suppliers. The cost-to-go function needs to be evaluated for all possible choices for S_j. Note that this solution procedure exploits the observations that the longest supplier service time determines the replenishment lead time for a stage, and that all suppliers to a stage will quote the same service time in the minimum-cost solution.

3.2 Distribution Networks

In a distribution network, each stage has at most one upstream stage that supplies it. But each stage may now feed several downstream stages or serve customer demand, and may quote a distinct service time to each of its downstream customers. Also, unlike a serial line or an assembly network, there will be more than one stage that directly serves customer demand. Hence, each stage will see the convolution of several demand streams and will need to have safety stock to protect against the variation in aggregate demand. The expression for the safety stock is slightly more complex (see Willems, 1996) than given in Section 3.1.

For the special case where we assume that each stage specifies a single service time applicable to each of its downstream customers, we can adapt the algorithm given above for assembly networks. In particular, the dynamic programming recursion is a “mirror image” of that for the assembly network. Whereas for an assembly network the recursion starts with the most upstream stages and works downstream, the algorithm for a distribution system starts with the most downstream stages and works upstream.

We have not examined how to solve the case when a single stage quotes different service times to its downstream customers.

3.3 General Networks

We are developing recursive approaches for supply chains represented by general acyclic networks. The general idea involves breaking the network into assembly and distribution subgraphs. Each subgraph is then solved using the appropriate technique described in sections 3.1 and 3.2. If stage j is a stage where two subgraphs are connected, then once both subgraphs have been solved, we only need to loop over the possible service times at stage j to find out which one is optimal. When solving these more complicated networks, a key difficulty in the construction of an algorithm is in how to traverse the network. In particular we need to assure that once a stage is reached in the algorithm, that all of the relevant associated stages, either upstream or downstream, have been evaluated.

4 Current Research

We are working with two industrial settings to test and validate the model. At the first company, we are applying the model to determine strategic inventory levels within the company's large internal supply chain. The supply chain is represented as an assembly network with about 15 stages. At the second company we are using the model to look at reducing redundant inventories between the company and a major customer. Here we have represented the supply chain as a serial system with 6 stages. These cases will provide some insight into how useful the current model is, as well as what are the most restrictive assumptions.

The model has been programmed in C for the Macintosh. The graphical interface allows the user to quickly construct various supply chain configurations. The program allows the user to either use the program as an optimizer that determines optimal service times or as a calculator where the user can enter service times and see what impact the inputs have on the total inventory costs for the network. The user-friendly nature of the software has been a critical success factor in the case studies to date.

We are currently looking at several extensions. As mentioned in Section 3, the first extension involves solving networks that have component commonality between adjacent echelons. The solution procedure will involve breaking the network into assembly and distribution subgraphs. After these subgraphs are solved, we are working on intelligent ways to recombine these subgraphs with the remaining nodes that do not belong in either an assembly or distribution subgraph.

Another area of research is to develop computationally efficient ways to allow a stage to quote different service times to its immediate successor nodes. We will also be examining how to extend the work to incorporate stochastic production lead times and capacity constraints.

References

Graves, S. C., "Safety Stocks in Manufacturing Systems," *Journal of Manufacturing and Operations Management*, 1 (1988), 67-101.

Graves, S. C., D. B. Kletter and W. B. Hetzel, "A Dynamic Model for Requirements Planning with Application to Supply Chain Optimization," working paper, March 1994, revised March 1995, January 1996.

Lee, H. L. and C. Billington, "Material Management in Decentralized Supply Chains," *Operations Res.*, 41 (1993), 835-847.

Simpson, K. F., "In-process Inventories," *Operations Res.*, 6 (1958), 863-873.

Willems, S. P., "Strategic Safety Stock Placement In Assembly/Distribution Systems," Master's Thesis, MIT Operations Research Center, May 1996