RC 20473 (06/04/96)
Computer Science/Mathematics

IBM Research Report

A Supply Network Model with Base-Stock Control and Service Requirements

Markus Ettl

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

Gerald E. Feigin

i2 Technologies, Inc.
Irving, TX 75039

Grace Y. Lin

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

David D. Yao

IEOR Department, Columbia University
New York, NY 10027

Research Division
Almaden · Austin · Beijing · Haifa · T.J. Watson · Tokyo · Zurich

A Supply Network Model with Base-Stock Control and Service Requirements

May 1996; Revisions: May 1997, February 1998, July 1998

Markus Ettl

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

Gerald E. Feigin[1]

i2 Technologies, Inc.
Irving, TX 75039

Grace Y. Lin

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

David D. Yao[2]

IEOR Department, Columbia University
New York, NY 10027

Abstract

We develop a supply network model that takes as input the bill of materials, the (nominal) lead times, the demand and cost data, and the required customer service levels. In return, the model generates the base-stock level at each store — the stocking location for a part or an end-product, so as to minimize the overall inventory capital throughout the network and to guarantee the customer service requirements. The key ingredient of the model is a detailed, albeit approximate, analysis of the *actual* lead times at each store and the associated demand over such lead times, along with a characterization of the operation at each store via an inventory-queue model. The gradients are derived in explicit forms; and a conjugate gradient routine is used to search for the optimal solution. Several numerical examples are presented to validate the model and to illustrate its various features.

[1] Research undertaken while employed at IBM T.J. Watson Research Center, Yorktown Heights, NY 10598.

[2] Research undertaken while an academic visitor at IBM T.J. Watson Research Center, Yorktown Heights, NY 10598; also supported by an NSF Grant DMI-95-23029.

1 Introduction

1.1 Overview

A supply chain can be viewed as a global network of suppliers, fabrication sites, assembly locations, distribution centers, and customer locations, through which components and products flow. The dynamics of a supply chain are driven by customer orders for finished products. Associated with each customer order is a request date, by which the customer would like to receive the complete order. The service level is the fraction of customer orders that are received by their request dates.

Asset managers of large manufacturing enterprises, such as those found in the computer, electronics, and auto industries, oversee extremely large supply chains involving multiple products each with a complex BOM (bill of materials). These asset managers have the responsibility of determining the appropriate inventory levels in the form of components and finished goods to hold at each level of the supply chain in order to guarantee specified end-customer service levels. Given the size and complexity of these supply chains, a common problem for these asset managers is not knowing how to quantify the tradeoff between service levels and the investment in inventory required to support those service levels. The problem is made difficult by the fact that the supply chains these asset managers oversee are dynamic: products have short lifetimes, new products are introduced frequently, customer demand is erratic and non-stationary, and service-level requirements may change over time. The dynamic nature of complex supply chains means that the tradeoff between service level and inventory investment changes over time, implying that asset managers must continually reevaluate this tradeoff so that they can make informed and timely decisions about inventory levels and service targets.

In this paper, we describe a model of complex supply chains that is appropriate for asset managers to use in quantitatively assessing the inventory-service level tradeoff. The model was developed with several considerations in mind. First, it is intended as a strategic model that will be used on an on-going basis. As such, it must strike a balance between capturing the complex dynamics of a large supply chain and yet not focusing too much attention on operational issues. Second, the model is to be used for both performance analysis and optimization. In the former case, a user first specifies inventory policies throughout the network, and then an analysis is performed resulting in estimates of end-customer service levels, fill rates at each stocking location in the network, and the average dollar value of inventory held in the network. In the case of optimization, the user specifies end-customer service levels and then an analysis is performed, which finds the (base-stock) inventory level at each stocking location so as to minimize the total inventory capital in the network and meet the specified end-customer service levels. Third, the model must deal with the dynamic aspects of the supply chain. Finally, the analysis of the supply network, whether in performance evaluation or optimization mode, must be performed quickly, so that frequent what-if analysis by asset managers is feasible.

We model all sites in a supply network as capable of building some set of products, whether finished goods, components, or sub-assemblies, which are shipped to a set of downstream customers, whether end customers or other sites in the network. Associated with each site in the network and each product it can produce is a single-level BOM, which contains the set of components required to produce a single item of the product. At each site, storage areas, which we refer to as "stores", exist to hold both components that appear on the single level BOMs and finished products. We assume a distributed inventory control mechanism whereby each store in the network operates according to a base-stock control policy. This is a continuous review, one-for-one replenishment policy, also known as an $(S-1, S)$ policy. The replenishment lead time of any store, given that there are no stockouts at upstream stores, is a random variable, which we assume to be independent of the number of outstanding orders (in this sense, the store is *uncapacitated*). Our control policy requires specifying a base-stock level at each store. We formulate our model in terms of these base-stock levels. For performance evaluation, we input these base-stock levels and then analyze the end-customer service levels as well as fill rates and inventory levels throughout the network. To perform optimization, we formulate a constrained non-linear optimization problem, which minimizes the total average dollar value of inventory in the network, subject to meeting the service-level requirements of end customers. The optimization is carried out using the conjugate gradient method, where we make use of gradient estimates obtained analytically.

The base-stock control policy assumes a one-for-one replenishment mechanism. Consequently, we do not address the problem of selecting order sizes at each store; nor, for that matter, do we address many other operational aspects of inventory management in supply chains. This is consistent with our objective of providing asset managers a *strategic* decision support tool, for which close attention to operational details is neither appropriate nor desirable.

To handle the dynamic nature of supply chains and in particular the non-stationarity of demand, we adopt a rolling-horizon point of view, which is quite standard in materials requirement planning and distribution resource planning tools (although often in a deterministic setting). That is, as time evolves and new information becomes available, the decision variables and the performance measures change as well. In most of the subsequent sections, we implicitly place ourselves at the beginning of an arbitrary time interval and derive our calculations assuming this to be the current time. In a dynamic environment, the calculations need to be updated on a regular basis with the latest available information. (See the demand offset scheme in §2.3 and the multi-period examples in §6.3.)

The remainder of the paper is organized as follows. We present a brief survey of the related literature in the rest of this section. In §2, we describe our model in detail. In subsequent sections, we present the performance analysis (§3), the optimization problem including the gradient estimates (§4), and implementation issues (§5). In §6, we present experimental results and application examples to illustrate the various features of our model. Finally, in §7, we conclude with pointing out several possible extensions and issues for further research.

1.2 Literature Review

The literature relating to the type of model we examine is extensive. Instead of making any attempt to be comprehensive, our intent here is to situate our work with respect to prior research that relates to our work in terms of motivation, model definition, and results.

Axsäter (1993) provides a good description of models and algorithms for the analysis of continuous review policies in multi-echelon networks. The classic model of a continuous review, multi-echelon inventory system is METRIC (Sherbrooke 1968), developed for the U.S. Air Force as a planning tool for controlling inventory of reparable items. The model consists of two echelons, the lower one consisting of n identical bases or stocking locations, the upper one consisting of a single depot or warehouse which supplies the stocking locations with repaired parts. The original model considered a single item, i.i.d. lead times, one-for-one replenishment, no capacity constraints, and a stationary compound Poisson demand process. The objective of the model was to identify stocking policies at the bases and the depot to minimize backorders at the base level subject to a constraint on inventory investment. Subsequent work has extended the model in various ways. MOD-METRIC (Muckstadt 1973) extended the model to handle multiple items and multi-indenture. VARI-METRIC (Sherbrooke 1986) and Graves (1985) proposed methods for improving the model's accuracy. Lee and Moinzadeh (1987a), (1987b) allowed for batch ordering and shipping. Deuermeyer and Schwartz (1981) and Svoronos and Zipkin (1988) developed simplified approaches based on decomposition. Svoronos and Zipkin (1991) considered a model with non-overtaking repair (or transit) times as opposed to the i.i.d. assumption used elsewhere. Based on their approximation model, Schwarz, Deuermeyer and Badinelli (1985) formulate and solve an optimization problem, not unlike ours, in which the objective is to maximize system fill-rate subject to a constraint on system safety stock.

The arborescent structure of the networks treated by METRIC and related models are appropriate for distribution networks. However, the supply chains we are modeling involve not only distribution but assembly, in which multiple components are required for the production of one part. Extensions to treat such networks are not straightforward. See Ernst and Pyke (1992) and Cohen and Lee (1988) for research in this direction.

Perhaps the work most closely related to ours is Lee and Billington (1993). Some similarities and differences in their model and ours are worth pointing out. First, both models pay close attention to capturing the interdependence of base-stock levels at different stores and how this interdependence affects overall system performance. Second, both models use normal approximations to characterize the demand over lead times and to compute related performance measures, although the two models differ significantly in characterizing the lead times; in particular, we model the difference between the *nominal* lead times (given data) and *actual* lead times (derived measures taking into account stockout possibilities at the supply stores) based on queueing analysis (see §3.3). Third, the model of Lee and Billington appears to be aimed principally at performance analysis, not optimization. As the authors state, "we have used a simple search heuristic to look for the 'best' combination of service level targets at different sites of the supply chain to support desired end-customer service targets. This is clearly an area for further research." In this respect, our work can be viewed as an extension of their model in that we have focused considerable attention on solving an explicit constrained optimization problem by making use of analytically derived gradient estimates. Fourth, whereas their model assumes stationary demand, we treat non-stationary demand via the rolling-horizon approach mentioned above. Consequently, the characterization of demand over lead times becomes more challenging. Finally, the inventory dynamics in our model are derived from a detailed analysis of a queueing model with batch arrivals.

In Glasserman and Tayur (1995), a simulation based approach (perturbation analysis) is developed to obtain gradient estimation (sensitivity) with respect to the base-stock levels in a multi-echelon system. In contrast, our approach here is to derive approximate gradient formulas that are in explicit, analytical forms, and are easy to compute. This is quite essential to the kind of large scale applications that motivates our model.

2 The Supply Network

2.1 Network Topology

As we mentioned in §1.1, the supply network consists of a collection $\mathcal{S}$ of stocking locations, or stores, each of which stocks one type of s.k.u. Each site in the network in general has two types of stores: *input* stores and *output* stores. (Hence, in Figure 1, for instance, a site is represented by the dashed box containing input and output stores). The input stores model the stocking of different types of components received from upstream suppliers, and the output stores model the stocking of finished-product inventory at the site. While this models naturally a site that is a manufacturing plant, it can also model a site that is a distribution center where no assembly takes place. For instance, let there be exactly one input store and one output store at the site, with zero transition time in between.

Each store $i \in \mathcal{S}$ has a set of supply stores, denoted by $\mathcal{S}_{>i}$, which consists of all those stores which *directly* supply store i. If i is an input store, we assume that $\mathcal{S}_{>i}$ is a singleton set, i.e., it contains exactly one store (which is necessarily an upstream output store). This means that each input store can obtain replenishments from only one supplier. (This assumption effectively rules out the operational issues involved in having multiple, alternative suppliers, such as which suppliers to use and how to split the order among the suppliers.) On the other hand, if i is an output store, then the set $\mathcal{S}_{>i}$ either is empty, in which case i is a *source* store, or contains one or more stores, which are necessarily input stores at the same site (as store i), corresponding to the components in the one-level BOM for the s.k.u. in store i. Moreover, this one-level BOM also specifies the usage counts, $u_{ji},\ j \in \mathcal{S}_{>i}$, where u_{ji} denotes the number of s.k.u.'s from store j needed to produce one s.k.u. at store i.

By way of symmetry, for each store $i \in \mathcal{S}$, denote by $\mathcal{S}_{i>}$ all those stores that receive *direct* supply from i. If i is an input store, $\mathcal{S}_{i>}$ consists of at least one output store at the same site. If i is an output store, either $\mathcal{S}_{i>}$ is empty, in which case i is an *end* store, or it contains one or more input stores at a downstream site. The set of end stores is denoted by $\mathcal{S}_0$.

2.2 Base-Stock Control and Nominal Lead Times

Each store i follows a base-stock control policy for managing inventory. The policy works as follows: When the *inventory position* (i.e., on-hand plus on-order minus backorder) at store i falls below some specified base-stock level, R_i, a replenishment order is placed.

If i is an input store, a replenishment order is placed with its unique supplier, an output store upstream. If the upstream supplier has on-hand inventory, the order is filled immediately and shipped to store i. The shipping time constitutes the *nominal* lead time of store i, denoted by L_i. All orders placed by store i have i.i.d. nominal lead times. If the upstream store has a stockout, the order joins a backorder queue at that store, i.e., the order has to wait to get filled.

If i is an output store, the replenishment order is essentially a production order whereby one s.k.u. of the output store i is assembled from the s.k.u.'s of the input stores at the same site, according to the one-level BOM at store i. If all of the input stores have stock on-hand, production starts immediately. (Upon completion, the s.k.u. is placed in the output store as part of its on-hand inventory.) The production time constitutes the nominal lead time of the output store (also denoted by L_i). If any one of the input stores has a stockout, the production order joins a backorder queue and is not executed until all components needed for assembly are available.

Note that the nominal lead time is different from, but related to, the *actual* lead time, $\tilde{L}_i$, which is the lead time taking into consideration the possibility of a stockout at the supply stores ($\mathcal{S}_{>i}$). To be more specific, the actual lead time associated with a store is the difference between the time an order is placed by the store and the time the complete order arrives to the store. Clearly the actual lead time (a random variable) is at least the nominal lead time but may be significantly more, depending on whether a stockout occurs at any of its supply stores. Whereas the nominal lead times are *primitive* data, the actual lead times are *derived* performance measures.

2.3 End Customers, Demands, and Service Levels

Customer classes are identified according to different product types and different service-level requirements. (Hence, if two orders are for the same product but with different service-level requirements, they will be classified as two different classes.) Denote by $\mathcal{M}$ the set of all customer classes. Associated with each customer class $m \in \mathcal{M}$ are the (end) store that supplies the demand, and the transit (delivery) time, T_m. In addition, let W_m denote the waiting time to receive an order for a typical class m customer; then the service-level requirement for class m customers is given by $\mathsf{P}[W_m \leq \beta_m] \geq \alpha_m$, where α_m and β_m are given parameters: β_m is a possibly random due date for class m orders, while α_m is the fraction of class m orders that are filled before the due date.

There is also a demand stream (forecast or real), $\{D(m,1), D(m,2), ...\}$, associated with each class m, where the second argument of $D(m,\cdot)$, $1, 2, ...$, denotes the time periods; e.g., day 1, day 2, etc. (Throughout, we use discretized time units to characterize demand.) Note that here we allow the demand data to be random, and non-stationary. Assume independence among the demand classes, and independence between the demands and the nominal lead times.

A base-stock control scheme requires translating the end-customer demand streams into the effective demand stream at each store i, denoted $\{D_i(1), D_i(2), D_i(3), ...\}$. To this end, first define the *offset* associated with customer class m as $o_m := \mathsf{E}[T_m]$. Next, for each end store $i \in \mathcal{S}_O$, define the effective demand stream as:

$$D_i(t) := \sum_{m:\mathtt{str}(m)=i} D(m, o_m + t) \quad t = 1, 2, 3, ...,$$

where $\mathtt{str}(m)$ denotes the (end) store that supplies the demand class m. Now we can proceed recursively, moving upstream in the BOM, for all other stores in the network. Specifically, define the offset of store i to be $o_i := \mathsf{E}[L_i]$, for all $i \in \mathcal{S}$. Then, for all stores $i \notin \mathcal{S}_0$, define the effective demand streams as:

$$D_i(t) := \sum_{j \in \mathcal{S}_{i>}} u_{ij} D_j(o_j + t) \quad t = 1, 2, 3, ...$$

These effective demand streams, $\{D_i(t);\ i \in \mathcal{S}, t = 1, 2, ...\}$, are then the starting point of our analysis below.

Note here we implicitly assume that the present time is period 0. In multiple-period implementation, when it comes to generating either the performance evaluation or the optimal decision for the next period, simply repeat the above procedure, applied to the updated demand data.

3 Performance Analysis

This is the main technical section of the paper. Roughly, the section consists of two parts. In the first part, we start from modeling each site as an inventory-queue, and analyzing the number of "jobs" in the queue (§3.1). This quantity is then related to performance measures such as on-hand inventory and backorders, and stockout probabilities and fill rates (§3.2). Involved in all these measures is the actual lead time at each site, which we derive in the second part of this section through characterizing the additional delay due to supplier stockout (§3.3). We then compound the demand streams with the actual lead times (§3.4), and relate these to customer service requirements (§3.5). Finally, in §3.6 we sum up the key assumptions and approximations used in our analysis.

3.1 The Inventory-Queue Model

The demand data that we assume as given take the following form: the number of orders per time unit (e.g., daily demand), D_i, at each store i, which is a random variable in general. For the time being, assume stationary demands at all stores. The non-stationary case will be discussed when the actual lead times are specified; see §3.4. This type of demand data leaves us with some degree of freedom in choosing an appropriate model for the demand *arrival process*, and this process is crucial in modeling the inventory dynamics at each store.

Specifically, we model each store i as an infinite-server queue operating under a base-stock control rule. (Refer to Buzacott and Shanthikumar (1993), Chapter 4, for general background materials for make-to-stock queues.) In particular, we use the $M^X/G/\infty$ model, where arrivals follow a Poisson process with rate λ_i, and each arrival brings a batch of X_i units, or orders. (Throughout, we assume each order is a single unit.) As it will become evident below, the batch Poisson arrival process is a good tradeoff between generality and tractability. In particular, it offers at least three parameters to model the demand data: the arrival rate and the first two moments of the batch size (whereas a simple Poisson arrival process has only one parameter).

We should emphasize that the batch Poisson arrival mechanism is a modeling device: it need not represent the way orders actually arrive at the stores — in most applications this kind of detailed information is unavailable, and in any event it would be too much of a burden for users to supply such information. This comment also applies to most of the operational details below: they are modeling assumptions more than anything else.

All arrived units are processed individually, by an infinite pool of servers so that all jobs are processed immediately on arrival, and the processing times are i.i.d. following a general distribution. The store produces up to a base-stock level of R_i units. Any arrival brings in general a batch of orders, which is filled from (and hence depletes) the on-hand inventory, if any. Otherwise, i.e., in a stockout situation, the arriving orders have to wait to be filled, i.e., for the desired units to be produced. Below, we shall refer to this as a "backorder" situation. Regardless, however, any order is processed immediately on arrival. In other words, any arriving order will trigger production (i.e., processing of orders), and the production will continue until the on-hand (surplus) inventory is built to the base-stock level of R_i.

Let $\tilde{L}_i$ be the *actual* lead time at store i. Note that $\tilde{L}_i$ is *not* the time to *fill* the order, which could be zero when there is positive inventory. The complete characterization of $\tilde{L}_i$ will be presented later. But first note that $\tilde{L}_i$ is equal to the service-time requirement of the jobs in the $M^X/G/\infty$ queue outlined above.

Observe that we have

$$\mathsf{E}(D_i) = \lambda_i \mathsf{E}(X_i), \quad \text{and} \quad \mathsf{Var}(D_i) = \lambda_i \mathsf{E}(X_i^2);$$

and that the variance expression above follows from the batch (compound) Poisson arrivals. Let N_i be the total number of jobs in the queue $M^X/G/\infty$ (in equilibrium). Following the results in Liu, Kashyap and Templeton (1990), the mean and the variance of N_i are as follows:

$$\mu_i := \mathsf{E}(N_i) = \lambda_i \mathsf{E}(X_i)\mathsf{E}(\tilde{L}_i) = \mathsf{E}(D_i)\mathsf{E}(\tilde{L}_i), \tag{1}$$

$$\begin{aligned} \sigma_i^2 := \mathsf{Var}(N_i) &= \mathsf{E}(N_i) + \lambda_i[\mathsf{E}(X_i^2) - \mathsf{E}(X_i)] \int_0^\infty \bar{F}^2_{\tilde{L}_i}(y)dy \\ &= \mu_i + [\mathsf{Var}(D_i) - \mathsf{E}(D_i)] \int_0^\infty \bar{F}^2_{\tilde{L}_i}(y)dy, \end{aligned} \tag{2}$$

where $F_{\tilde{L}_i}(y) = 1 - \bar{F}_{\tilde{L}_i}(y)$ denotes the distribution function of $\tilde{L}_i$. From (1) and (2) above, it is clear that the selection of a batch Poisson arrival process results in simple, explicit expressions of the key quantities, μ_i and σ_i^2, in terms of the given and derivable data, D_i and $\tilde{L}_i$.

Although the generating function of N_i is available in Liu *et. al.* (1993) from which one can in principle derive the distribution of N_i, here we choose to simply approximate the distribution as normal with mean μ_i and variance σ_i^2 as derived above. This way, we can write

$$N_i = \mu_i + \sigma_i Z, \tag{3}$$

where Z denotes the standard normal variate.

Note that whereas the mean of N_i, μ_i, is equal to the mean of demand over lead time, the variance of N_i, σ_i^2, is *not* equal to the variance of demand over lead time. In fact, σ_i^2 is significantly less than the variance of demand over lead time, as we shall explain later in §3.4.

3.2 Inventory and Service Measures

Let I_i and B_i be, respectively, the level of on-hand inventory and the number of backorders at store i. These relate to N_i and R_i through the following relations:

$$I_i = [R_i - N_i]^+, \qquad B_i = [N_i - R_i]^+, \tag{4}$$

where $[x]^+ := \max\{x, 0\}$. In view of (3), we can write

$$R_i = \mu_i + k_i \sigma_i, \tag{5}$$

where k_i is the *safety factor*. Then,

$$\mathsf{E}(B_i) = \mathsf{E}[N_i - R_i]^+ = \sigma_i \mathsf{E}[Z - k_i]^+ = \sigma_i \int_{k_i}^{\infty} (z - k_i)\phi(z)dz,$$

where $\phi(z) = \exp(-z^2/2)/\sqrt{(2\pi)}$ is the density function of Z. Denoting

$$G(k_i) := \int_{k_i}^{\infty} (z - k_i)\phi(z)dz, \tag{6}$$

we have

$$\mathsf{E}(B_i) = \sigma_i G(k_i).$$

Similarly, letting

$$H(k_i) := k_i + G(k_i) = \int_{-\infty}^{k_i} (k_i - z)\phi(z)dz, \tag{7}$$

we have

$$\mathsf{E}(I_i) = \sigma_i H(k_i).$$

Furthermore, we have

$$\mathsf{E}(I_i) - \mathsf{E}(B_i) = \sigma_i[H(k_i) - G(k_i)] = k_i \sigma_i,$$

which also follows directly from (4): $I_i - B_i = R_i - N_i$, and $\mathsf{E}(N_i) = \mu_i$.

For future reference, we summarize below the properties of the functions $G(x)$ and $H(x)$ in (6) and (7). These properties are all easily verified from first principle. ($\Phi(x)$ denotes the distribution function of Z, and $\bar{\Phi}(x) = 1 - \Phi(x)$.)

(a) $G(x) = \phi(x) - x\bar{\Phi}(x)$ (note that $\phi'(x) = -x\phi(x)$); and $H(x) = \phi(x) + x\Phi(x)$.
$G'(x) = -\bar{\Phi}(x)$, $H'(x) = \Phi(x)$, and $G''(x) = H''(x) = \phi(x)$.

(b) For all $x \in (-\infty, +\infty)$, $G(x) \geq 0$ and $H(x) \geq 0$; $G(x)$ is decreasing and $H(x)$ increasing, and both are convex. (Throughout, we use "increasing" and "decreasing" in the non-strict sense.)

(c) For $x >> 1$, $G(x) \approx 0$ and $G(-x) \approx x$, whereas $H(x) \approx x$ and $H(-x) \approx 0$.

The stockout probability at store i, denoted p_i, is simply

$$p_i := \mathsf{P}[I_i = 0] = \mathsf{P}[N_i \geq R_i] = \mathsf{P}[Z \geq k_i] = \bar{\Phi}(k_i). \tag{8}$$

That is, p_i is the fraction of time that the on-hand inventory at store i is zero.

The fill rate at store i, denoted f_i, is the fraction of customer orders that is filled by on-hand inventory. Let $\bar{f}_i := 1 - f_i$. Observe that when $N_i < R_i$, the difference, $R_i - N_i$, is the on-hand inventory; Arrivals and when $N_i > R_i$, there is a backorder queue of $N_i - R_i$ units. In equilibrium, we can equate the rate of orders entering the backorder queue and the rate of orders leaving the backorder queue, to arrive at the following relation:

$$\lambda_i \mathsf{E}(X_i)\bar{f}_i = [\mathsf{E}(\tilde{L}_i)]^{-1}\mathsf{E}[N_i|N_i > R_i]\mathsf{P}[N_i > R_i]. \tag{9}$$

(When there are $N_i = n$ orders in processing, the rate of output is $n/[\mathsf{E}(\tilde{L}_i)]$; conditioning on $N_i > R_i$ yields the right hand side above.) It is easy to verify that in the case of single-unit batches, i.e., $X_i \equiv 1$, the above yields $\bar{f}_i = p_i$, as expected from the $M/G/\infty$ queue.

Making use of the above relation, along with (1) and (6), we have

$$\begin{aligned}
\bar{f}_i &= \mu_i^{-1}\mathsf{E}[N_i|N_i > R_i]\mathsf{P}[N_i > R_i] \\
&= \mu_i^{-1}[\mathsf{E}(N_i - R_i)^+ + R_i\mathsf{P}(N_i > R_i)] \\
&= \mu_i^{-1}[\sigma_i G(k_i) + (\mu_i + k_i\sigma_i)\bar{\Phi}(k_i)] \\
&= \sigma_i\phi(k_i)/\mu_i + \bar{\Phi}(k_i),
\end{aligned} \tag{10}$$

where the last equality takes into account the relation $G(k_i) = \phi(k_i) - k_i\bar{\Phi}(k_i)$ (see property (a) at the end of §3.2).

3.3 Lead Times

To characterize $\tilde{L}_i$, the *actual* lead time at store i, reason as follows: When store i issues an order, the order goes to all of its supply stores; if these stores all have on-hand inventory, then the actual lead time is just store i's own nominal lead time, i.e., $\tilde{L}_i = L_i$. This happens with probability

$$\pi_i(\emptyset) := \prod_{j \in \mathcal{S}_{>i}} f_j, \tag{11}$$

where f_j is the fill rate, the fraction of orders supplied by on-hand inventory at store j. In other words, π represents the fraction of orders at store i that experiences the store's nominal lead time. On the other hand, if the order is not filled immediately at a subset of the supply stores, say, $A \subseteq \mathcal{S}_{>i}$, due to stockout, then there is an additional delay. Hence,

$$\tilde{L}_i = L_i + \max_{j \in A}\{\tau_j\}, \tag{12}$$

where τ_j denotes the additional delay at store $j \in A$ — a quantity that is quite intractable in general, and will be approximated later.

And this happens with probability

$$\pi_i(A) := \prod_{j\in A} \bar{f}_j \prod_{j\in\mathcal{S}_{>i}\setminus A} f_j. \tag{13}$$

In other words, $\pi_i(A)$ is the fraction of orders that experiences a lead time $\tilde{L}_i$ in (12) at store i.

Note that the above argument involves two approximations. First, the product-form assumes *independence* among the set of stores ($\mathcal{S}_{>i}$) that supplies i, whereas, insofar as these stores always receive the same replenishment request simultaneously from store i, they are not really independent. Second, without the i.i.d. setting (of a renewal process, for instance), it is hard to justify that each order will necessarily experience a lead time that follows the same distribution.

Even with these approximations, the above lead-time model still suffers from a combinatorial explosion: $\tilde{L}_i$ is a mixture of $2^{|\mathcal{S}_{>i}|}$ cases (of orders not filled on arrival). Hence, we further use the following approximation: Ignore those cases in which an order is not filled simultaneously at two or more stores, and renormalize the probabilities accordingly. This way, we have

$$\tilde{L}_i = \begin{cases} L_i & \text{w.p.} \quad \pi_{0i} \\ L_i + \tau_j & \text{w.p.} \quad \pi_{ji},\ j\in\mathcal{S}_{>i} \end{cases}, \tag{14}$$

where

$$\pi_{0i} := (1 + \sum_{h\in\mathcal{S}_{>i}} \bar{f}_h/f_h)^{-1}, \quad \pi_{ji} := (\bar{f}_j/f_j)\pi_{0i}. \tag{15}$$

Note that in the above expressions, we assume that $f_h > 0$ for all $h \in \mathcal{S}_{>i}$. In case $f_j = 0$ for some $j \in \mathcal{S}_{>i}$, we let $\tilde{L}_i = L_i + \tau_j$ with probability 1; in case there are two stores, $j, j' \in \mathcal{S}_{>i}$ with $f_j = 0$ and $f_{j'} = 0$, we associate equal probabilities $(1/2)$ with $\tilde{L}_i = L_i + \tau_j$ and $\tilde{L}_i = L_i + \tau_{j'}$; and so forth.

As mentioned above, the quantity τ_j is difficult to characterize. For instance, consider the first order that arrives to the backorder queue at store j. Upon its arrival at store j, there is zero on-hand inventory, and R_j orders are being processed. Hence, τ_j equals the minimum of the R_j (residual) processing times; and the output rate at store j is $R_j/\mathsf{E}(\tilde{L}_i)$ (treating residuals as whole processing times). However, this rate will change once another unit arrives.

Consider, therefore, the simpler model, $M/M/\infty$ (i.e., assume unit batch size). Without loss of generality, assume the arrival rate $\lambda = 1$. Let S denote the generic service time, and denote $\rho = \lambda\mathsf{E}(S) = \mathsf{E}(S)$. We know that the mean number of busy servers (which is equal to the mean number of jobs in the system) is $\mathsf{E}(N) = \rho$. Let Δ_ℓ be the time until the next service completion, starting with ℓ servers busy. Then, conditioning on the next event being either an arrival or a service completion, we have

$$\begin{aligned} \mathsf{E}(\Delta_\ell) &= \frac{1}{1+\ell/\rho}\cdot\frac{\ell/\rho}{1+\ell/\rho} + [\frac{1}{1+\ell/\rho} + \mathsf{E}(\Delta_{\ell+1})]\cdot\frac{1}{1+\ell/\rho} \\ &= \frac{\rho}{\rho+\ell} + \mathsf{E}(\Delta_{\ell+1})\cdot\frac{\rho}{\rho+\ell}. \end{aligned}$$

Hence, recursively, we have

$$\mathsf{E}(\Delta_\ell) = \frac{\rho}{\rho+\ell} + \frac{\rho^2}{(\rho+\ell)(\rho+\ell+1)} + \frac{\rho^3}{(\rho+\ell)(\rho+\ell+1)(\rho+\ell+2)} + \cdots.$$

Now, given ℓ $(> R)$ servers busy, the expected time until there are $\ell - R$ service completions is equal to the following:

$$\begin{aligned}\sum_{m=R+1}^{\ell} \mathsf{E}(\Delta_m) &\leq \sum_{m=R+1}^{\ell} [\frac{\rho}{\rho+m} + \frac{\rho^2}{(\rho+m)^2} + \frac{\rho^3}{(\rho+m)^3} + \cdots] \\ &= \sum_{m=R+1}^{\ell} \frac{\rho}{m} \leq \rho \cdot \frac{\ell - R}{R+1}.\end{aligned}$$

Therefore, we shall use the upper bound above as a basis to approximate τ_j. In our context, $\rho = \mathsf{E}(\tilde{L}_j)$, and ℓ should be replaced by $\mathsf{E}[N_j | N_j > R_j]$, and hence $\ell - R$ replaced by $\mathsf{E}[N_j - R_j | N > R_j] = \mathsf{E}(B_j)/p_j$. This way, we arrive at the following approximation:

$$\tau_j = \tilde{L}_j r_j, \quad \text{where} \quad r_j := \frac{\mathsf{E}(B_j)}{p_j(R_j + 1)}. \tag{16}$$

3.4 Non-Stationary Demands

We now return to the characterization of μ_i and σ_i, when demands are non-stationary. Denote

$$D_i^{\text{cum}}(\ell) := D_i(1) + \cdots + D_i(\ell).$$

(Note that the D_i's on the right hand sides above are demand data *after* the offsetting mechanism in §2.3 is carried out.)

From (1), it is clear that μ_i is simply the mean of the cumulative demand over the lead time: $\mu_i = \mathsf{E}[D_i^{\text{cum}}(\tilde{L}_i)]$. Conditioning on $\tilde{L}_i$ (refer to (14)), we have

$$\mu_i = \pi_{0i} \mathsf{E}[D_i^{\text{cum}}(L_i)] + \sum_{j \in \mathcal{S}_{>i}} \pi_{ji} \mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)] \tag{17}$$

As to σ_i^2, the situation is more involved. From (2), taking into account $\mathsf{E}(X_i^2) \geq \mathsf{E}(X_i)$ (since it is reasonable to assume $\mathsf{E}(X_i) \geq 1$), and recognizing

$$\int_0^\infty \bar{F}_i^2(y)dy \leq \int_0^\infty \bar{F}_i(y)dy = \mathsf{E}(\tilde{L}_i),$$

we have

$$\begin{aligned}\sigma_i^2 &\leq \mathsf{E}(N_i) + \lambda_i[\mathsf{E}(X_i^2) - \mathsf{E}(X_i)] \int_0^\infty \bar{F}_i(y)dy \\ &= \lambda_i \mathsf{E}(X_i^2) \int_0^\infty \bar{F}_i(y)dy = \mathsf{Var}(D_i)\mathsf{E}(\tilde{L}_i),\end{aligned} \tag{18}$$

since $\lambda_i \mathsf{E}(X_i^2) = \mathsf{Var}(D_i)$, and $\mathsf{E}(N_i) = \lambda_i \mathsf{E}(X_i)\mathsf{E}(\tilde{L}_i)$. Similarly, we have the following lower bound:

$$\sigma_i^2 \geq \mathsf{Var}(D_i) \int_0^\infty \bar{F}_i^2(y)dy.$$

Since the difference between $\int_0^\infty \bar{F}_i^2(y)dy$ and $\int_0^\infty \bar{F}_i(y)dy = \mathsf{E}(\tilde{L}_i)$ is likely quite small, we shall use the upper bound in (18) as a basis to approximate σ_i^2. This leads to the following expression, for the case of non-stationary demand:

$$\begin{aligned} \sigma_i^2 &= \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(\tilde{L}_i)|\tilde{L}_i]\,\} \\ &= \mathsf{E}\{\sum_{k=1}^{\tilde{L}_i}\mathsf{Var}[D_i(k)]\,\} \\ &= \pi_{0i}\mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i)|L_i]\,\} + \sum_{j\in\mathcal{S}_{>i}}\pi_{ji}\mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i+\tau_j)|L_i+\tau_j]\,\}. \end{aligned} \tag{19}$$

Note that the above should be compared against the following generic relationship:

$$\mathsf{Var}[D_i^{\text{cum}}(\tilde{L}_i)] = \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(\tilde{L}_i)|\tilde{L}_i]\,\} + \mathsf{Var}\{\,\mathsf{E}[D_i^{\text{cum}}(\tilde{L}_i)|\tilde{L}_i]\,\}.$$

In other words, σ_i^2, the variance of the number of jobs in the $M^X/G/\infty$ queue, is close to (or, more precisely, upper bounded by) the first term of $\mathsf{Var}[D_i^{\text{cum}}(\tilde{L}_i)]$, the variance of demand over lead time.

The two expectations in (19) can be derived as follows:

$$\begin{aligned} \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i)|L_i]\,\} &= \mathsf{E}\{\sum_{m=1}^{L}\mathsf{Var}[D_i(m)]\,\} \\ &= \sum_{\ell}\mathsf{Var}[D_i^{\text{cum}}(\ell)]\mathsf{P}[L=\ell], \end{aligned} \tag{20}$$

$$\begin{aligned} \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i+\tau_j)|L_i+\tau_j]\,\} &= \mathsf{E}\{\sum_{m=1}^{L_i+\tau_j}\mathsf{Var}[D_i(m)]\,\} \\ &= \sum_{\ell}\mathsf{Var}[D_i^{\text{cum}}(\ell)]\mathsf{P}[L_i+\tau_j=\ell]. \end{aligned} \tag{21}$$

3.5 Customer Service-Level Requirements

For each end store — a store that directly supplies external customer demand, the starting point is the fill rate, which is determined by the required customer service level; and the fill rate, in turn, determines the safety factor and the base-stock level. This is in contrast to the case of internal stores discussed above, where the base-stock level is first set, which then determines the stockout probability and the fill rate.

Consider a particular customer class, and suppose its demand is supplied by one of the end stores, $i \in \mathcal{S}_0$. Let W_i denote the waiting time to receive an order. Recall the required service level is

$$\mathsf{P}[W_i \le \beta_i] \ge \alpha_i, \tag{22}$$

where β_i and α_i are given data. (Here we have translated customer-class related data, indexed by m, into store specific data, indexed by i.) When the demand is supplied from on-hand inventory, the delay is simply the transportation time T_i — time to deliver the finished products to customers; otherwise, there is an additional delay of τ_i. (See the discussion in §3.3; in particular, τ_i follows the approximation in (16).) Hence,

$$\mathsf{P}[W_i \le \beta_i] = f_i\mathsf{P}[T_i \le \beta_i] + (1-f_i)\mathsf{P}[T_i+\tau_i \le \beta_i]. \tag{23}$$

In order for the above to be at least α_i, we need to set f_i at least at the following level:

$$f_i = \frac{\alpha_i - \mathsf{P}[T_i + \tau_i \le \beta_i]}{\mathsf{P}[T_i \le \beta_i] - \mathsf{P}[T_i + \tau_i \le \beta_i]}. \tag{24}$$

Two scenarios are exceptions to the above analysis:

(i) $\alpha_i > \mathsf{P}[T_i \le \beta_i]$. In this case, the required serviceability is too high — even a 100% fill rate will not be able to meet the desired service level. So the problem is infeasible.

(ii) β_i is large enough so that $\mathsf{P}[T_i \le \beta_i] = \mathsf{P}[T_i + \tau_i \le \beta_i] = 1$. Any fill rate will meet the service requirement. In this case, the constraint (i.e., required service level) becomes superfluous. As a regulation, we shall remove i from the set $\mathcal{S}_0$. (In other words, $\mathcal{S}_0$ only includes those end stores that have non-trivial service constraints.)

The quantity τ_i involved in the right hand side of (24) can be expressed as $\tau_i = \tilde{L}_i r_i$, following (16). Since r_i involves B_i and R_i, both of which are functions of k_i, and so is f_i (refer to (10)), we need to solve a fixed point problem defined by the equation in (16) to get f_i (or k_i). In the conjugate gradient search routine (to be elaborated later), however, this fixed point problem can be avoided by simply using the r_i value obtained from the previous iteration. Once f_i and k_i are derived, the base-stock level and the stockout probability then follow (5) and (8), respectively.

3.6 Summary of Assumptions and Approximations

The basic assumptions about our supply network model were already detailed in §2, such as the network configuration, and the independence among the demand streams and between the demand streams and the nominal lead times. Here we summarize the key assumptions and approximations made throughout our *analysis*.

- The operation of each store i is modeled as an $M^X/G/\infty$ queue. In particular, the inventory and backorder levels of each store i is related to the number of jobs N_i in the $M^X/G/\infty$ queue; refer to (1,2,4).
- The variance of N_i, in the case of non-stationary demands, is approximated based on an upper bound in the stationary case; refer to (18,19).
- The actual lead time $\tilde{L}_i$ at each store i is characterized by assuming (i) the stockout events at its supply stores are independent, and simultaneous stockout at two or more supply stores is ignored; refer to (11,13); (ii) the additional delay τ_j, due to a stockout at a supply store j, is approximated through an upper bound obtained from a birth-death queue analysis; refer to (16).

Furthermore, although not necessary, in most derived formulas we assumed an underlying normal distribution so as to facilitate the derivation of the first two moments.

Finally, another approximation will be made in the next section when we derive the gradients: we shall ignore the terms that are weighted by higher orders of the no-fill rates. (This is appropriate as the repeated multiplication of fill rates quickly leads to small weights.)

4 The Optimization Model

The objective of our optimization model is to minimize the total expected inventory *capital* (note: not inventory holding cost) throughout the network, while satisfying customer service-level requirements, as specified in (24).

There are two types of inventory at each store in the system: on-hand inventory (or, finished goods), and work-in-process (WIP; or, pipeline) inventory. (Note, the WIP includes the orders in transition (i.e., being transported) from one store to another.) From the analysis in §3.1, we know the expected on-hand inventory at store i is $\mathsf{E}[I_i] = \sigma_i H(k_i)$, and the expected WIP is $\mathsf{E}[N_i] = \mu_i$.

Let c_i denote the inventory capital per s.k.u. at store i. This applies to the finished-goods (i.e., on-hand) part of the inventory. For the part that is in the pipeline at store i, we apply another cost coefficient:

$$\hat{c}_i := \frac{1}{2}(c_i + \sum_{j \in \mathcal{S}_{>i}} c_j u_{ji}).$$

(Recall u_{ji} is the usage count — number of units from store j that is needed for each unit in store i.) That is, we take the average between the inventory value at store i and the value of all the raw materials that make up the s.k.u. at store i. (If i is a source store, the above formula can be adjusted to reflect the actual cost accounting of raw materials.)

Therefore, the objective function takes the following form:

$$C(\mathbf{k}) = \sum_{i \in \mathcal{S}} [\hat{c}_i \mu_i + c_i \sigma_i H(k_i)]. \tag{25}$$

(Note that the summation above is over all stores, including end stores.) We want to minimize $C(\mathbf{k})$, subject to meeting the fill-rate requirements set in (24) — for the end stores: $i \in \mathcal{S}_0$.

Note here we choose as decision variables the safety factors, k_i for store i. In view of (8), we could also choose p_i instead. Also note that here we allow zero or even negative base-stock levels. These correspond to policies that choose to keep back-orders instead of inventory. (This is indeed the case in a make-to-order system, for instance.) The objective function in (25), however, is always nonnegative: note in particular that $H(k_i)$ is nonnegative, cf. (7).

Next, let us have a closer look at the constrained minimization problem formulated above. Splitting the summation in (25) into two parts: end stores and non-end stores, we have

$$C(\mathbf{k}) = (\sum_{i \notin \mathcal{S}_0} + \sum_{i \in \mathcal{S}_0})[\hat{c}_i \mu_i + c_i \sigma_i H(k_i)]. \tag{26}$$

Two observations are in order. Note that the first summation, over the non-end stores, is independent of the decision variables in the second summation — the safety factors of the end stores, $(k_i)_{i \in \mathcal{S}_0}$. This is becuase the (actual) lead times at any store depend only on its supply store; hence the lead times at the non-end stores are independent of the safety factors of the end stores (since the latter can never supply the former). On the other hand, in each term of the second summation, $H(k_i)$ is the only quantity that depends on k_i, $i \in \mathcal{S}_0$, the mean and the standard deviation depend (through the lead time) on the non-end stores, specifically, on $(k_i)_{i \in \mathcal{S}}$.

Now, consider an end store, $i \in \mathcal{S}_0$. From (24), it is clear that the safety factor of the store, k_i, is increasing in its fill rate f_i, taking into account that $G(k_i)$ is decreasing in k_i, and μ_i and σ_i are independent of k_i. Since $H(k_i)$ is increasing in k_i, it is hence increasing in f_i. Therefore, in order to minimize the objective function in (26), the constraints for the end stores should all be binding, i.e., the f_i's should be set at the lowest possible values, as in (24). In other words, we can remove the constraints by removing $(k_i)_{i\in\mathcal{S}_0}$ from the set of decision variables, treating them instead as *dependent* — through the equations in (24) — on the decision variables, This way, we have an unconstrained problem of minimizing $C(\mathbf{k})$, with $\mathbf{k} = (k_i)_{i\notin\mathcal{S}_0}$ as decision variables.

Finally, another observation: although as mentioned earlier we allow negative base-stock levels, these will not likely be candidates for the optimal solution that minimizes the objective function in (25) (or (26)). This is because a negative $R_i < 0$ implies $k_i < -\mu_i/\sigma_i$. When μ_i/σ_i is reasonably large, $H(k_i) \approx 0$. Hence, by increasing k_i (i.e., making it less negative), we can reduce $\tilde{L}_{i'}$ for any store i' that is supplied by i, and hence reduce $\mu_{i'}$. This way, we will have reduced the μ_i part of (25) while keeping the σ_i part at zero. Based on this observation, below we shall ignore the possiblity of negative base-stock levels, or, $k_i < -\mu_i/\sigma_i$.

The gradients of the objective function play a very important role in the optimization routine. For any given store j, $j \notin \mathcal{S}_0$ (i.e., internal store, as explained above), we now derive the partial derivative $\frac{\partial}{\partial k_j}C(\mathbf{k})$.

First, consider the j-th term in $C(\mathbf{k})$ of (25). Note that the actual lead time at store j, $\tilde{L}_j$, is independent of k_j: it only depends on the fill rates, and hence the safety factors, of its supply stores; refer to (14). So both $\mu_j(\tilde{L}_j)$ and $\sigma_j(\tilde{L}_j)$ are independent of k_j. Hence, the partial derivative (w.r.t. k_j) of the j-th term is:

$$c_j\sigma_j H'(k_j) = c_j\sigma_j\Phi(k_j).$$

Second, consider $i \in \mathcal{S}_{j>}$ (or, equivalently, $j \in \mathcal{S}_{>i}$), i.e., a store that receives supplies directly from j. Suppose i is not an end store, i.e., $i \notin \mathcal{S}_0$. Then, $\tilde{L}_i$ depends on $\bar{f}_j$ [cf. (14)], and hence also on k_j; whereas k_i is independent of k_j. Hence, the partial derivative (w.r.t. k_j) of this i-th term is

$$\hat{c}_i\frac{\partial\mu_i}{\partial k_j} + c_iH(k_i)\frac{\partial\sigma_i}{\partial k_j}.$$

When i is an end store, $i \in \mathcal{S}_0$, k_i also depends on k_j, since it is determined by the fill rate, which, in turn, depends on $\tilde{L}_i$ and hence on k_j. In this case, the partial derivative is

$$\hat{c}_i\frac{\partial\mu_i}{\partial k_j} + c_i[H(k_i)\frac{\partial\sigma_i}{\partial k_j} + \sigma_i\Phi(k_i)\frac{\partial k_i}{\partial k_j}].$$

We ignore the contribution (to the derivative $\frac{\partial}{\partial k_j}C(\mathbf{k})$) from all other stores that are *indirectly* supplied by j. It is clear that these will be weighted by higher orders of the no-fill rates ($\bar{f}_j$'s). This is appropriate when the fill rates are reasonably high. (For instance, for two stores each having 80% fill rate, the product of their no-fill rates is only 4%.)

This way, we have

$$\frac{\partial}{\partial k_j}C(\mathbf{k}) \quad = \quad c_j\sigma_j\Phi(k_j) + \sum_{i\in\mathcal{S}_{j>}\setminus\mathcal{S}_0}[\hat{c}_i\frac{\partial\mu_i}{\partial k_j} + c_iH(k_i)\frac{\partial\sigma_i}{\partial k_j}]$$

$$
\begin{aligned}
& + \sum_{i \in \mathcal{S}_{j>} \cap \mathcal{S}_0} \{\hat{c}_i \frac{\partial \mu_i}{\partial k_j} + c_i [H(k_i) \frac{\partial \sigma_i}{\partial k_j} + \sigma_i \Phi(k_i) \frac{\partial k_i}{\partial k_j}]\} \\
= \quad & c_j \sigma_j \Phi(k_j) + \sum_{i \in \mathcal{S}_{j>}} [\hat{c}_i \frac{\partial \mu_i}{\partial k_j} + c_i H(k_i) \frac{\partial \sigma_i}{\partial k_j}] + \sum_{i \in \mathcal{S}_{j>} \cap \mathcal{S}_0} c_i \sigma_i \Phi(k_i) \frac{\partial k_i}{\partial k_j}.
\end{aligned}
$$

Since, from (10), we have

$$
\frac{\partial \bar{f}_j}{\partial k_j} = -\frac{\sigma_j}{\mu_j} k_j \phi(k_j) - \phi(k_j) = -\frac{R_j}{\mu_j} \phi(k_j),
$$

we can rewrite $\frac{\partial}{\partial k_j} C(\mathbf{k})$ as follows:

$$
\begin{aligned}
\frac{\partial}{\partial k_j} C(\mathbf{k}) \quad = \quad & c_j \sigma_j \Phi(k_j) - \frac{R_j}{\mu_j} \phi(k_j) \cdot \{ \sum_{i \in \mathcal{S}_{j>}} [\hat{c}_i \frac{\partial \mu_i}{\partial \bar{f}_j} + c_i H(k_i) \frac{\partial \sigma_i}{\partial \bar{f}_j}] \\
& + \sum_{i \in \mathcal{S}_{j>} \cap \mathcal{S}_0} c_i \sigma_i \Phi(k_i) \frac{\partial k_i}{\partial \bar{f}_j} \}. \qquad (27)
\end{aligned}
$$

What now remains is to derive $\frac{\partial \mu_i}{\partial \bar{f}_j}$ and $\frac{\partial \sigma_i}{\partial \bar{f}_j}$ for $i \in \mathcal{S}_{j>}$, and $\frac{\partial k_i}{\partial \bar{f}_j}$ for $i \in \mathcal{S}_{j>} \cap \mathcal{S}_0$. The detailed derivation of these derivatives is presented in the Appendix.

5 Implementation

We have implemented our model as a software system in the C++ programming language. The software tool has the capability of both performance evaluation and optimization. For performance evaluation, given the BOM, the lead times, cost and demand data, along with specified base-stock levels (or alternatively, stockout probabilities or safety factors) at all stores, the system generates performance measures, such as total inventory capital, inventory capital by stores, and the achieved customer service levels. Used in the optimization mode, with the given data, the system generates the base-stock levels at all stores in the network, which minimize the total inventory capital while meeting required customer service levels.

The system consists of the following modules: preprocessing, performance routine, optimization routine, and postprocessing. Below, we describe each module in detail.

Preprocessing

This part consists of the following steps:

- Input data files: BOM, products (location, nominal lead time, and costs), demands, periods, and serviceability.
- Construct the network: Each part in the product file is associated with an output store, and each entry in the BOM file is associated with an input store.
- Populate and offset demands: Recursively populates the demands from the end stores to their upstream supply stores until reaching the source stores, taking into account the demand offset using the scheme described in section §2.3.

- Generate initial safety factors: A set of safety factors for non-end stores are needed in order to initiate the optimization. Since the (local) optimum generated by the conjugate gradient method in general depends on the initial solution, we start with a limited search over say, a few hundred sets of safety factors generated at random. For every set the objective value is evaluated, and the one with the lowest objective value is selected as the starting solution for the optimization routine. (See §6.2 for further discussions.)

The Performance Routine

Performance evaluation is accomplished through first setting the actual lead time of each "leaf node" in the BOM tree — the source stores where parts are obtained from the external suppliers — to its nominal lead time. Then, move up the BOM tree, one level at a time to calculate the actual lead time of each store using the formula in (14). This part consists of two subroutines that handle, respectively, the convolution and the mixture operations. (Implementing the detailed calculation via (12) and (13) would require one more subroutine that handles the max operation.) This procedure is invoked repeatedly until the actual lead times of all stores are generated. Given the actual lead time and the demand data at each store, the mean and variance, μ_i and σ_i^2 for each store i, are then computed following the discussion in §3.4.

The performance routine also generates the gradients of the objective function, i.e. the rate of change of the total inventory capital with respect to the safety factor of each internal store. Following the formula in (27), in addition to the cost data, base-stock levels, (actual) lead times, the safety factors, and μ_i and σ_i^2 for each store i, we also need the derivatives of μ_i and σ_i^2 with respect to the safety factors of its immediate down stream stores. These derivatives follow the computation outlined in the Appendix.

The Optimization Routine

To minimize the objective function $C(\mathbf{k})$ [see (25)], we use the conjugate gradient method (Press *et al.* (1994), a standard technique for nonlinear optimization. Briefly, the method works as follows. Let the superscript n index the steps in the optimization. Let $\mathbf{g}$ and $\mathbf{h}$ denote two vectors, of the same dimension as $\mathbf{k}$. Let $\nabla C(\mathbf{k})$ denote the gradient (vector) of C at $\mathbf{k}$. Suppose the n-th step is done. In the next step $(n+1)$, the algorithm starts with a line search, along the direction of $\mathbf{h}^n$:

$$\min_{\lambda} C(\mathbf{k}^n + \lambda \mathbf{h}^n),$$

to yield $\mathbf{k}^{n+1}$; it then does the updating:

$$\mathbf{g}^{n+1} = -\nabla C(\mathbf{k}^{n+1}), \quad \mathbf{h}^{n+1} = \mathbf{g}^{n+1} + \frac{\mathbf{g}^{n+1} \cdot \mathbf{g}^{n+1}}{\mathbf{g}^n \cdot \mathbf{g}^n} \mathbf{h}^n.$$

The algorithm iterates in this fashion until convergence, i.e., when a prespecified tolerance, say the relative difference between the objective value of two consecutive steps, is achieved. (The two sequences of directions, $\mathbf{g}$ and $\mathbf{h}$, as specified above satisfy both orthogonality and conjugacy, hence the name "conjugate gradient".) At each step n, the algorithm requires the functional value, $C(\mathbf{k}^n)$, and the gradient, $\nabla C(\mathbf{k}^n)$. These are returned by the performance routine.

Postprocessing

Once either the performance evaluation or the optimization routine is completed, the system reports various performance measures. These include stockout probabilities, fill rates, base-stock levels, total inventory capital, and inventory capital breakdown at each store between the pipeline part and the on-hand part. It also reports the achieved customer service level.

6 Numerical Examples

The purpose of this section is two-fold. First, we want to illustrate the validity of our model through numerical examples that study various aspects of the model. In particular, we want to show numerical comparisons (a) between results from the performance evaluation routine and those obtained from Monte Carlo simulation (§6.1); and (b) between solutions from the optimization routine and solutions obtained from a random search (§6.2). Second, we want to illustrate the various features of our model: what kind of issues it can handle and what kind of insights it can generate. These will be illustrated in §6.3. For a more elaborate example of the application of the model to a personal computer manufacturing company, see Feigin (1998).

6.1 Performance Evaluation

Figure 1 and Table 1 illustrate our first example, a small supply network with five stores. The transit time from the suppliers to the assembly stage is a deterministic constant, whereas manufacturing lead times follow normal distributions, with mean and standard deviation listed in Table 1.

FIGURE 1 ABOUT HERE

Shipments of finished products to customers follow a normal distribution with mean 4 and standard deviation 0.5. There is a single type of finished product (at Store 1), and a single customer demand stream associated with it. (Note that in this system, as well as in system 2 below, the cost data are such that there is no value added in between an input store and its upstream output store. This is the actual inventory cost accounting mechanism in the asset management tool that has been the focus of our application. However, our model certainly applies to other applications in which a transportation cost, for instance, can be added to the inventory cost when an s.k.u. is moved from an upstream output store to a downstream input store.)

The first set of experiments investigates the issue of whether the solutions, in particular the base-stock levels, produced by our model will indeed meet the required end-customer service level, which is chosen to be $\beta = 5$ and $\alpha = 0.99$, meaning that 99% of the customers should have their orders filled within five days; or equivalently, the fraction of late orders should not exceed 1 percent. Table 2 presents the results for different volumes of the daily demand. In all cases the target service level falls within the confidence interval (95%). The simulation model is written in SIMAN (Pegden *et. al.* 1995). The confidence intervals are constructed from ten independent replications with 25,000 jobs in each replication.

Using the same scenario, Table 3 shows the comparison between simulation and our analytical model with respect to the total inventory capital held in the supply chain. Results in column $\mathtt{Total}_{appr}$ are obtained

Store i	Unit Cost c_i [in $]	Lead Time Distribution		
		Distribution Type	Mean	Std. Dev.
1	100	Normal	3.0	1.0
2	45	Constant	8.0	0.0
3	45	Constant	10.0	0.0
4	45	Normal	6.0	1.0
5	45	Normal	10.0	3.0

Table 1: Nominal lead time distributions and unit costs used in System 1.

Daily Demand	Fraction of Late Orders		Daily Demand	Fraction of Late Orders	
	Simul.	Approx.		Simul.	Approx.
Nor(10.0, 4.6)	0.0090 ± 0.0030	0.0100	Nor(10.0, 9.1)	0.0088 ± 0.0036	0.0100
Nor(20.0, 9.1)	0.0100 ± 0.0033	0.0100	Nor(20.0, 18.3)	0.0078 ± 0.0054	0.0100
Nor(30.0, 13.7)	0.0089 ± 0.0035	0.0100	Nor(30.0, 27.4)	0.0057 ± 0.0047	0.0100
Nor(40.0, 18.3)	0.0072 ± 0.0056	0.0100	Nor(40.0, 36.5)	0.0091 ± 0.0064	0.0100
Nor(50.0, 22.8)	0.0085 ± 0.0079	0.0100	Nor(50.0, 45.6)	0.0097 ± 0.0089	0.0100

Table 2: Simulation vs. approximation results for fraction of late orders.

by running the optimization routine of our model under the service constraint to generate the base-stock levels, which are then used in the simulation model to generate the results in column $\mathtt{Total}_{sim}$. The column under $\mathtt{Pipeline}$ lists the part of inventory capital that is due to work-in-process over *nominal* lead times. This part is independent of the decision variables (base-stock levels).

In Table 4 we focus on two demand distributions studied in Table 3, Nor(40, 18.3) and Nor(40, 36.5), and display the base-stock levels, the actual lead times, and the fill rates at each store. Note that except the end store, Store 1, the fill rates at the other stores are all relatively low. (Note that the fill rate at the end store relates to, but is different, by definition, from the required customer service level.) That is, in order to have a high customer service level, it is not necessary for the non-end stores to have high fill rates. This is consistent with the general recommendation in the multi-echelon inventory literature.

6.2 Optimization

In order to test whether our optimization routine does generate optimal or near-optimal solutions, we continue with the 5-store example in Figure 1. For each case, we randomly generate 500,000 points in the unit cube $[0,1]^4$ as stockout probabilities at the four non-end stores, and translate them into the base-stock levels via the safety factors. (Recall, the safety factor at the end store is not a decision variable — it is determined by the required customer service level; see §4.) We then evaluate the corresponding objective values, and pick the best to compare against the solution generated by the optimization routine, which starts with evaluating

	Inventory Capital [in $]					
Daily Demand	Total_{sim}	Total_{appr}	Pipeline	$\text{Total}_{sim} - \text{Pipeline}$	$\text{Total}_{appr} - \text{Pipeline}$	Rel.Err.
Nor(10.0, 4.6)	18,940	19,139	14,550	4,390	4,589	0.045
Nor(20.0, 9.1)	37,577	38,138	29,100	8,477	9,038	0.066
Nor(30.0, 13.7)	55,983	57,365	43,650	12,333	13,715	0.112
Nor(40.0, 18.3)	74,986	76,582	58,200	16,786	18,382	0.095
Nor(50.0, 22.8)	93,352	95,632	72,750	20,602	22,882	0.111
Nor(10.0, 9.1)	23,541	24,250	14,550	8,991	9,700	0.079
Nor(20.0, 18.3)	46,925	48,500	29,100	17,825	19,400	0.088
Nor(30.0, 27.4)	70,986	72,788	43,650	27,336	29,138	0.066
Nor(40.0, 36.5)	94,369	97,096	58,200	36,169	38,896	0.075
Nor(50.0, 45.6)	118,201	121,293	72,750	45,451	48,543	0.068

Table 3: Simulation vs. approximation results for total inventory capital.

Demand Nor(40.0, 18.3)		Actual Lead Time $\mathsf{E}[\tilde{L}_i]$		Fill Rate f_i	
Store i	Base-Stock R_i	Simul.	Approx.	Simul.	Approx.
1	213	3.28 ± 0.07	3.26	0.951 ± 0.023	0.989
2	384	8.52 ± 0.04	8.57	0.761 ± 0.030	0.720
3	490	10.60 ± 0.07	10.75	0.814 ± 0.034	0.817
4	236	6.00 ± 0.00	6.00	0.421 ± 0.029	0.390
5	396	10.00 ± 0.00	10.00	0.428 ± 0.046	0.412
Demand Nor(40.0, 36.5)		**Actual Lead Time $\mathsf{E}[\tilde{L}_i]$**		**Fill Rate f_i**	
Store i	Base-Stock R_i	Simul.	Approx.	Simul.	Approx.
1	321	3.69 ± 0.25	3.52	0.953 ± 0.036	0.989
2	423	8.74 ± 0.12	8.59	0.722 ± 0.104	0.685
3	590	11.10 ± 0.33	11.45	0.818 ± 0.106	0.794
4	279	6.00 ± 0.00	6.00	0.571 ± 0.107	0.531
5	398	10.00 ± 0.00	10.00	0.330 ± 0.137	0.376

Table 4: Simulation vs. approximation results for actual lead times and fill rates.

200 random solutions as part of the preprocessing (refer to §5). This step is necessary due to the non-smooth nature of the objective function. To illustrate the roughness of the response surface, in Figure 2 we take the case of Nor(40.0, 18.3) demand, and plot the objective value as a function of the stockout probabilities at stores 4 and 5, with the stockout probabilities at stores 2 and 3 fixed at 50%.

FIGURE 2 ABOUT HERE

Optimization is carried out in two ways: (I) apply conjugate gradient search to all the 200 initial solutions, and then pick the best solution; (II) pick the best among the 200 initial solutions, and then apply conjugate gradient search to it. Table 5 summarizes the comparisons.

		Inventory Capital [in $]			
Daily Demand	$R_i, i = 1, \ldots, 5$	Total_{appr}	Pipeline	Total_{appr} − Pipeline	Runtime
Nor(10.0, 4.6)					
Random Search	56, 93, 114, 60, 103	19,018	14,550	4,468	32.18 min
Optimization I	59, 90, 109, 57, 106	19,075	14,550	4,525	1.48 min
Optimization II	61, 85, 105, 61, 103	19,139	14,550	4,589	1.2 sec
Nor(20.0, 9.1)					
Random Search	114, 184, 222, 116, 204	38,049	29,100	8,942	33.45 min
Optimization I	113, 186, 223, 117, 205	38,042	29,100	8,949	2.10 min
Optimization II	114, 179, 224, 121, 210	38,138	29,100	9,038	1.4 sec
Nor(30.0, 13.7)					
Random Search	171, 273, 338, 174, 304	57,073	43,650	13,423	32.21 min
Optimization I	172, 278, 332, 170, 304	57,079	43,650	13,429	2.12 min
Optimization II	182, 254, 316, 189, 308	57,365	43,650	13,715	1.2 sec
Nor(40.0, 18.3)					
Random Search	226, 367, 452, 237, 404	76,084	58,200	17,884	32.29 min
Optimization I	227, 368, 442, 237, 413	76,090	58,200	17,890	2.24 min
Optimization II	213, 384, 490, 236, 396	76,582	58,200	18,382	1.4 sec
Nor(50.0, 22.8)					
Random Search	288, 454, 556, 284, 505	95,049	72,750	22,299	32.03 min
Optimization I	283, 455, 565, 296, 510	95,108	72,750	22,358	2.27 min
Optimization II	267, 480, 612, 291, 493	95,632	72,750	22,882	1.4 sec

Table 5: Comparison between random search and the proposed optimization.

In all cases, the solutions generated by Optimization I are virtually the same as the best solution from the random search. Optimization II yields solutions that are consistently within 3% (relative error) of the best solution from random search. The computation time of Optimization II is about 1.5 second, as compared against about 2 minutes of Optimization I, and over 30 minutes of the random search. All runs were carried out on an IBM Power PC 850; the computation times reported are user seconds measured by the system call *times*.

6.3 Multi-Period Implementation

While in all of the previous experiments the reported results were for the first period only, our model is capable of generating multi-period solutions, on a rolling-horizon basis, as explained before. To illustrate, consider the 11-store network example (System 2) shown in Figure 3 and Table 6. Again, there is a single type of finished products (at Store 1) and a single customer demand stream. The service-level parameters are set at $\alpha = 0.99$ and $\beta = 5$. Shipment times of finished products to the customer follow a normal distribution with mean 4 and standard deviation 0.5.

FIGURE 3 ABOUT HERE

		Lead-Time Distribution					Lead-Time Distribution		
Store	Unit Cost	Dist. Type	Mean	Std. Dev.	Store	Unit Cost	Dist. Type	Mean	Std. Dev.
1	\$200	Normal	3.0	0.5	7	\$2.50	Normal	4.0	0.33
2	\$100	Normal	4.0	0.33	8	\$1.80	Normal	4.0	0.33
3	\$100	Normal	10.0	0.66	9	\$36	Constant	30.0	0.0
4	\$80	Normal	4.0	0.33	10	\$2.50	Constant	20.0	0.0
5	\$80	Constant	3.0	0.0	11	\$1.80	Constant	45.0	0.0
6	\$36	Normal	4.0	0.33					

Table 6: Nominal lead time distibutions and unit costs used in System 2.

Data for three different scenarios of demand pattern are summarized in Table 7. We assume all demands follow normal distributions, but allow the mean and standard deviation to fluctuate over time. In the first scenario, demand is stationary (no fluctuation). In the second one, the demand distribution changes moderately from month to month (low fluctuation), whereas in the third scenario there are significant changes in the demand distributions (high fluctuation).

	Mean Demand per Planning Period (μ)								
Demand Scenario	jan	feb	mar	apr	may	jun	jul	aug	sep
Stationary	1000	1000	1000	1000	1000	1000	1000	1000	1000
Low Fluctuation	1000	1100	900	1000	1100	1000	1100	900	900
High Fluctuation	1500	600	900	1200	600	1200	900	600	1500

Table 7: Mean demand used in System 2; standard deviation $\sigma = \mu/10$.

Figure 4 illustrates the evolution over time of the objective values corresponding to the three cases. Clearly, in the high- and low-fluctuation cases, the curves track the fluctuation of the demand, whereas the curve stays flat in the stationary case as expected. Figures 5 and 6 show the base-stock levels at Store 1 and Store 9 over the planning horizon. Note that Store 1 is the end store; and hence, naturally, the base-stock level tracks closely the demand fluctuation. On the other hand, Store 9 is the other extreme, being a source store: its base-stock level exhibits a time-lag in tracking the demand fluctuation.

FIGURES 4–6 ABOUT HERE

Next, we apply the optimal base-stock levels for Stores 2 through 11 (leaving out the end store, Store 1) obtained from the stationary case to the low- and high-fluctuation cases, modifying the base-stock level at Store 1 to enforce the required customer service level. In other words, the base-stock levels at all non-end stores are maintained as constants, following the optimal solution from the stationary case, whereas the base-stock level at the end-store adjusts to demand fluctuation. This way, the overall inventory capital values (over the nine periods) are 21.47 and 21.42 for the low-fluctuation and the high-fluctuation cases, respectively. Note that these values are only slightly higher than the optimal values (21.30 and 21.27). Hence, it appears that even in the case of highly fluctuated demand pattern, our model is able to generate constant base-stock levels, for non-end stores, over the entire planning horizon. And these constant base-stock levels have the obvious advantage of facilitating practical implementation.

Our last experiment involving System 2 is to study the gains in using our optimization model as compared against a policy that sets base-stock levels at all internal stores so as to achieve 95% no-stockout probability. (The base-stock level at the end store is set to meet the service-level requirement.) Table 8 compares the inventory capital resulting from the above heuristic policy to the inventory capital resulting from our optimization model. Both low and high demand fluctuations are considered using the demand data in Table 7. The results show that our model leads to a reduction of roughly 20% in inventory capital.

	Inventory Capital (in weeks of supply)					
	Low Demand Fluctuation			High Demand Fluctuation		
Period	Heuristic	Optimization	Pct. Difference	Heuristic	Optimization	Pct. Difference
jan	28.5	23.8	19.9 %	40.4	34.5	17.1 %
feb	30.7	25.8	18.8 %	17.8	14.4	23.1 %
mar	25.7	21.4	20.0 %	26.1	21.5	21.8 %
apr	28.5	23.8	19.9 %	32.7	27.6	18.2 %
may	30.9	25.9	19.2 %	18.4	14.7	25.1 %
jun	28.5	23.8	19.9 %	33.2	27.8	19.3 %
jul	30.7	25.8	18.7 %	25.0	20.9	19.3 %
aug	25.5	21.3	19.6 %	19.1	15.0	27.3 %
sep	25.7	21.4	20.0 %	41.3	34.8	18.6 %

Table 8: Comparison between a heuristic inventory policy and the proposed optimization.

7 Concluding Remarks

We have developed here a supply network model that can be used to support decision making in the management of large, complex supply chains. The model features both performance evaluation and optimization with service-level constraints, and captures the essential dynamic aspects of the supply chain. Modelling the dynamics at each site in the network as an inventory-queue, we derive performance and service measures through a set of careful approximations. Gradient formulas are worked out in explicit forms, which lead to effective execution of the optimization algorithm. Preliminary numerical studies have found the model to perform well in both performance estimation and optimization; and pilot applications have led to several important insights (some are briefly cited in §6.3, more will appear in a sequel to this paper), as well as quantitative implications of alternative inventory-serviceability decisions.

There are several directions in which we can extend and enhance our model. Practitioners who used our model frequently asked us to incorporate order sizes (instead of one-for-one replenishment) into the system. As mentioned in the Introduction, order size is one of the operational issues beyond the scope and intent of our model. It appears, however, that our model can be easily adapted to handle at least one rather useful special case — constant order size, say, Q_i at store i. In this case, simply take the order size Q_i as the basic unit for counting the s.k.u. at store i. Then, the one-for-one replenishment policy still applies; the only change needed is to adjust the usage count: The quantity u_{ji} (recall, one s.k.u. at store i requires u_{ji} units of s.k.u. at store j) should be replaced by $\tilde{u}_{ji} := u_{ji}Q_i/Q_j$. More involved cases — e.g., when order sizes are random, or part of the decision — of course, require more research.

As mentioned in the Introduction, our model is essentially an uncapacitated model, in the sense that the lead times do not depend on the intensity of the demand, at least not directly. In order to capture the kind of queueing delay due to the contention upon limited production resources, we need a different kind of inventory-queue model — one with a *finite* number of servers (instead of the infinite-server model in §3.1). The response times derived from such queues will better model the (actual) lead times, in terms of capturing the interplay between capacity limit on the one hand and demand intensity on the other hand. Adapting some of the (approximate) queueing models in Buzacott and Shanthikumar (1993) appears to be a feasible first step in this direction.

In our current model, we assume all orders will be filled, with possible delay, in the entirety of the ordered quantity. In some applications, the issue of yield loss or partial orders can be quite significant. That is, only part of the ordered quantity will eventually arrive. Clearly, in this case, the safety-stock part of the target inventory should protect against not only lead-time variation, but also the variation in yield loss. In an assembly environment, this issue becomes more pressing: missing quantity in any single component could delay the whole assembly. This issue is also tied together with order size: in view of yield loss, it is obviously desirable to place a larger order. Prior studies that address this issue in the context of semiconductor manufacturing operations — e.g., Connors and Yao (1996), and Connors, Feigin and Yao (1996) — may provide some of the basic ingredients for incorporating yield losses into our supply network model.

8 Appendix

8.1 Derivation of $\frac{\partial \mu_i}{\partial \bar{f}_j}$ and $\frac{\partial \sigma_i}{\partial \bar{f}_j}$, for $i \in \mathcal{S}_{j>}$

From (17), we have

$$\frac{\partial \mu_i}{\partial \bar{f}_j} = \mathsf{E}[D_i^{\text{cum}}(L_i)]\frac{\partial \pi_{0i}}{\partial \bar{f}_j} + \sum_{h \in \mathcal{S}_{>i}} \mathsf{E}[D_i^{\text{cum}}(L_i + \tau_h)]\frac{\partial \pi_{hi}}{\partial \bar{f}_j}$$
$$+\pi_{ji}\frac{\partial}{\partial \bar{f}_j}\mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)]. \tag{28}$$

Similarly, from (19), we have

$$\frac{\partial \sigma_i}{\partial \bar{f}_j} = \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i)|L_i]\,\}\frac{\partial \pi_{0i}}{\partial \bar{f}_j} + \sum_{h \in \mathcal{S}_{>i}} \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_h)|L_i + \tau_h]\,\}\frac{\partial \pi_{hi}}{\partial \bar{f}_j}$$
$$+\pi_{ji}\frac{\partial}{\partial \bar{f}_j}\mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_j)|L_i + \tau_j]\,\}. \tag{29}$$

Now, it is easy to derive from (15) the following:

$$\frac{\partial \pi_{0i}}{\partial \bar{f}_j} = -\frac{\pi_{0i}^2}{f_j^2}, \qquad \frac{\partial \pi_{hi}}{\partial \bar{f}_j} = -\frac{\pi_{0i}\pi_{hi}}{f_j^2} \quad h \neq j,$$

and

$$\frac{\partial \pi_{ji}}{\partial \bar{f}_j} = \frac{\pi_{0i} - \pi_{0i}\pi_{ji}}{f_j^2}.$$

Hence, the first two terms on the right hand side of (28) and (29) are, respectively, as follows:

$$\mathsf{E}[D_i^{\text{cum}}(L_i)]\frac{\partial \pi_{0i}}{\partial \bar{f}_j} + \sum_{h \in \mathcal{S}_{>i}} \mathsf{E}[D_i^{\text{cum}}(L_i + \tau_h)]\frac{\partial \pi_{hi}}{\partial \bar{f}_j}$$
$$= \frac{\pi_{0i}}{f_j^2}\{\mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)] - \mathsf{E}[D_i^{\text{cum}}(\tilde{L}_i)]\}, \tag{30}$$

and

$$\mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i)|L_i]\,\}\frac{\partial \pi_{0i}}{\partial \bar{f}_j} + \sum_{h \in \mathcal{S}_{>i}} \mathsf{E}\{\,\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_h)|L_i + \tau_h]\,\}\frac{\partial \pi_{hi}}{\partial \bar{f}_j}$$
$$= \frac{\pi_{0i}}{f_j^2}\{\,\mathsf{E}[\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_j)|L_i + \tau_j]] - \mathsf{E}[\mathsf{Var}[D_i^{\text{cum}}(\tilde{L}_i)|\tilde{L}_i]]\,\}. \tag{31}$$

The conditional variance terms above follow the expressions in (20) and (21). Note that (30) and (31) above are for $f_j > 0$. When $f_j \to 0$, it can be verified, following (14) and (15) and the specifications there, that $\pi_{0i}/f_j^2 \to 1$.

Next, consider the third term on the right hand side of (28) and (29). We have (with sd denoting standard deviation)

$$\frac{\partial}{\partial \bar{f}_j}\mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)] = \frac{\partial}{\partial \bar{f}_j}\int_{-\infty}^{+\infty} \mathsf{E}[D_i^{\text{cum}}(L_i + x)]\phi(\frac{x - \mathsf{E}(\tau_j)}{\mathsf{sd}(\tau_j)})\frac{dx}{\mathsf{sd}(\tau_j)}. \tag{32}$$

Writing $z := (x - \mathsf{E}(\tau_j))/\mathsf{sd}(\tau_j)$, and recognizing $\phi'(z) = -z\phi(z)$, we have

$$
\begin{aligned}
\frac{\partial}{\partial \bar{f}_j}\frac{\phi(z)}{\mathsf{sd}(\tau_j)} &= -\frac{\phi(z)}{\mathsf{Var}(\tau_j)}\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j) - \frac{z\phi(z)}{\mathsf{sd}(\tau_j)}\frac{\partial z}{\partial \bar{f}_j} \\
&= -\frac{\phi(z)}{\mathsf{Var}(\tau_j)}\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j) + \frac{z\phi(z)}{\mathsf{Var}(\tau_j)}\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) + \frac{z^2\phi(z)}{\mathsf{Var}(\tau_j)}\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j) \\
&= \frac{\phi(z)}{\mathsf{Var}(\tau_j)}[z\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) + (z^2-1)\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j)].
\end{aligned}
$$

Therefore, substituting the above into (32), and writing

$$
Z(\tau_j) := \mathsf{E}(\tau_j) + Z \cdot \mathsf{sd}(\tau_j),
$$

with Z (on the right hand side) denoting the standard normal variate, we have

$$
\begin{aligned}
&\frac{\partial}{\partial \bar{f}_j}\mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)] \\
= \; &[\mathsf{sd}(\tau_j)]^{-1}\{\, \mathsf{E}[Z D_i^{\text{cum}}(L_i + Z(\tau_j))]\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) \\
&+(\mathsf{E}[Z^2 D_i^{\text{cum}}(L_i + Z(\tau_j))] - \mathsf{E}[D_i^{\text{cum}}(L_i + \tau_j)])\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j)\,\}.
\end{aligned} \tag{33}
$$

Similarly,

$$
\begin{aligned}
&\frac{\partial}{\partial \bar{f}_j}\mathsf{E}[\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_j)|L_i + \tau_j]] \\
= \; &[\mathsf{sd}(\tau_j)]^{-1}\{\, \mathsf{E}[Z \cdot \mathsf{Var}[D_i^{\text{cum}}(L_i + Z(\tau_j))|L_i + Z(\tau_j)]]\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) \\
&+(\, \mathsf{E}[Z^2 \cdot \mathsf{Var}[D_i^{\text{cum}}(L_i + Z(\tau_j))|L_i + Z(\tau_j)]] \\
&\quad -\mathsf{E}[\mathsf{Var}[D_i^{\text{cum}}(L_i + \tau_j))|L_i + \tau_j]]\,)\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j)\,\}.
\end{aligned} \tag{34}
$$

From (16), making use of $G(k_j) = \phi(k_j) - k_j\bar{\Phi}(k_j)$, and writing $g(k_i) := \phi(k_j)/\bar{\Phi}(k_j)$, we have

$$
r_j = \frac{\sigma_j G(k_j)}{\bar{\Phi}(k_j)(R_j + 1)} = \frac{\sigma_j[g(k_j) - k]}{\mu_j + \sigma_j k_j + 1}.
$$

Then,

$$
\mathsf{E}(\tau_j) = r_j\mathsf{E}(\tilde{L}_j), \qquad \mathsf{sd}(\tau_j) = r_j\mathsf{sd}(\tilde{L}_j),
$$

and hence

$$
\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) = -\frac{\mu_j\,\mathsf{E}(\tilde{L}_j)}{R_j\phi(k_j)}\frac{\partial r_j}{\partial k_j}, \qquad \frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j) = -\frac{\mu_j\mathsf{sd}(\tilde{L}_j)}{R_j\phi(k_j)}\frac{\partial r_j}{\partial k_j}. \tag{35}
$$

Since $g'(k_j) = g^2(k_j) - k_j g(k_j)$, we have

$$
\begin{aligned}
\frac{\partial r_j}{\partial k_j} &= \frac{\sigma_j[g^2(k_j) - k_j g(k_j) - 1]}{R_j + 1} - \frac{\sigma_j^2[g(k_j) - k_j]}{(R_j + 1)^2} \\
&= \frac{\sigma_j}{R_j + 1}[g^2(k_j) - k_j g(k_j) - r_j - 1] \\
&= r_j g(k_j) - \frac{\sigma_j(r_j + 1)}{R_j + 1}.
\end{aligned} \tag{36}
$$

8.2 Derivation of $\frac{\partial k_i}{\partial \bar{f}_j}$, for $i \in \mathcal{S}_{j>} \cap \mathcal{S}_0$

We have

$$\frac{\partial k_i}{\partial \bar{f}_j} = \frac{\partial k_i}{\partial \bar{f}_i}\frac{\partial \bar{f}_i}{\partial \bar{f}_j} = \frac{-\mu_i}{R_i\phi(k_i)}\frac{\partial \bar{f}_i}{\partial \bar{f}_j}.$$

Next, from (24), we have

$$\frac{\partial \bar{f}_i}{\partial \bar{f}_j} = \frac{\bar{f}_i}{\mathsf{P}[T_i \le \beta_i] - \mathsf{P}[T_i + \tau_i \le \beta_i]} \cdot \frac{\partial}{\partial \bar{f}_j}\mathsf{P}[T_i + \tau_i \le \beta_i].$$

Similar to the derivation of (33) and (34), writing $Y_i := [\beta_i - T_i - \mathsf{E}(\tau_i)]/\mathsf{sd}(\tau_i)$, and $\mathsf{P}[T_i + \tau_i \le \beta_i] = \mathsf{E}[\Phi(Y_i)]$, we have

$$\frac{\partial}{\partial \bar{f}_j}\mathsf{P}[T_i + \tau_i \le \beta_i] = -[\mathsf{sd}(\tau_i)]^{-1}\{\, \mathsf{E}[\phi(Y_i)]\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i) + \mathsf{E}[Y_i\phi(Y_i)]\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_i)\,\}. \tag{37}$$

Therefore, combining the above, we have

$$\frac{\partial k_i}{\partial \bar{f}_j} = \frac{\mu_i \bar{f}_i\{\, \mathsf{E}[\phi(Y_i)]\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i) + \mathsf{E}[Y_i\phi(Y_i)]\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_i)\,\}}{R_i\phi_i(k_i)\mathsf{sd}(\tau_i)\{\, \mathsf{P}[T_i \le \beta_i] - \mathsf{P}[T_i + \tau_i \le \beta_i]\,\}}.$$

To derive $\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i)$ and $\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_i)$, note $\tau_i = \tilde{L}_i r_i$, and following the discussion at the end of §3.5, we treat r_i as a constant (from the last iteration of the optimization routine) independent of $\bar{f}_j$. Hence, similar to the derivation of (30), we have

$$\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i) = r_i\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tilde{L}_i) = \frac{r_i\pi_{0i}}{f_j^2}[\mathsf{E}(L_i + \tau_j) - \mathsf{E}(\tilde{L}_i)] + r_i\pi_{ji}\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j).$$

Similarly,

$$\begin{aligned} & \frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i^2) = r_i^2\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tilde{L}_i^2) \\ = \; & \frac{r_i^2\pi_{0i}}{f_j^2}[\mathsf{E}((L_i + \tau_j)^2) - \mathsf{E}(\tilde{L}_i^2)] + r_i^2\pi_{ji}[2\mathsf{E}(L_i)\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j) + \frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j^2)]. \end{aligned}$$

Hence,

$$\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_i) = \frac{1}{2}[\mathsf{sd}(\tau_i)]^{-1/2}\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i^2) - 2\mathsf{E}(\tau_i)\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_i).$$

The derivatives $\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j)$ and $\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j^2)$ in the above expressions follow from (35), with

$$\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j^2) = \frac{\partial}{\partial \bar{f}_j}[\mathsf{sd}^2(\tau_j) + \mathsf{E}^2(\tau_j)] = 2\mathsf{sd}(\tau_j)\frac{\partial}{\partial \bar{f}_j}\mathsf{sd}(\tau_j) + 2\mathsf{E}(\tau_j)\frac{\partial}{\partial \bar{f}_j}\mathsf{E}(\tau_j).$$

References

AXSÄTER, S. 1993. Continuous review policies. In *Logistics of Production and Inventory*, S. Graves, A. Rinooy Kan, and P. Zipkin, Eds. North Holland, ch. 4.

BUZACOTT, J.A. AND SHANTHIKUMAR, J.G. 1993. *Stochastic Models of Manufacturing Systems,* Prentice-Hall, Englewood Cliffs, New Jersey.

COHEN, M., AND LEE, H. 1988. Strategic Analysis of Integrated Production-Distribution Systems: Models and Methods. *Operations Research*, **36**, 216 – 228.

CONNORS, D. AND YAO, D.D. 1996. Methods for Job Configuration in Semiconductor Manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, **9**, 401-411.

CONNORS, D., FEIGIN, G.E. AND YAO, D.D. 1996. A Queueing Network Model for Semiconductor Manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, **9**, 412-427.

DEUERMEYER, B. L., AND SCHWARZ, L. 1981. A Model for the Analysis of System Service Level in Warehouse/Retailer Distribution Systems: The Identical Retailer Case. In *Multi-Level Production/Inventory Control Systems: Theory and Practice*. North Holland, ch. 8.

ERNST, R., AND PYKE, D. (1992). Component Part Stocking Policies. *Naval Research Logistics*, **39**, 509 – 529.

FEIGIN, G. (1998). Inventory Planning in Large Assembly Supply Chains. In *Advances in the Quantitative Modeling for Supply Chain Management*. R. Ganeshan, M. Magazine, and S. Tayur, Eds. Kluver, ch. 24. To appear.

GLASSERMAN, P., AND TAYUR, S. (1995). Sensitivity Analysis of Base-Stock Levels in Multiechelon Production-Inventory Systems. *Management Science*, **41**, 263 – 281.

GRAVES, S. (1985). A Multi-Echelon Inventory Model for a Repairable Item with One-for-One Replenishment. *Management Science*, **31**, 1247 – 1256.

LEE, H., AND BILLINGTON, C. (1993). Material Management in Decentralized Supply Chains. *Operations Research*, **41**, 835 – 847.

LEE, H., AND MOINZADEH, K. (1987A). Operating Characteristics of a Two-Echelon Inventory System for Repairable and Consumable Items Under Batch Ordering and Shipment Policy. *Naval Research Logistics*, **34**, 365 – 380.

LEE, H., AND MOINZADEH, K. (1987B). Two-Parameter Approximations for Multi-Echelon Repairable Inventory Models with Batch Ordering Policy. *IIE Transactions*, **19**, 140 – 149.

LIU, L., KASHYAP, B.R.K., AND TEMPLETON, J.G.C. (1990). On the $GI^X/G/\infty$ System. *Journal of Applied Probability*, **27**, 671 – 683.

MUCKSTADT, J. (1973). A Model for a Multi-Item, Multi-Echelon, Multi-Indenture Inventory System. *Management Science*, **20**, 472 – 481.

PEGDEN, C.D., SHANNON, R.E., AND SADOWSKI, R.P. (1995). *Introduction to Simulation Using SIMAN,* McGraw-Hill, New York, 2nd Edition.

PRESS, W.H., TEUKOLSKY, S.A., VETTERING, W.T., AND FLANNERY, B.P. (1994). *Numerical Recipes in C,* Cambridge University Press, New York, 2nd edition.

SCHWARZ, L., DEUERMEYER, B., AND BADINELLI, R. (1985). Fill-Rate Optimization in a One-Warehouse N-Identical Retailer Distribution System. *Management Science*, **31**, 488 – 498.

SHERBROOKE, C. (1968). METRIC: A Multi-Echelon Technique for Recoverable Item Control. *Operations Research*, **16**, 122 – 141.

SHERBROOKE, C. (1986). VARI-METRIC: Improved Approximations for Multi-Indenture, Multi-Echelon Availability Models. *Operations Research*, **34**, 311 – 319.

SVORONOS, A., AND ZIPKIN, P. (1988). Estimating the Performance of Multi-Level Inventory Systems. *Operations Research*, **36**, 57 – 72.

SVORONOS, A., AND ZIPKIN, P (1991). Evaluation of One-for-One Replenishment Policies for Multiechelon Inventory Systems. *Management Science*, **37**, 68 – 83.

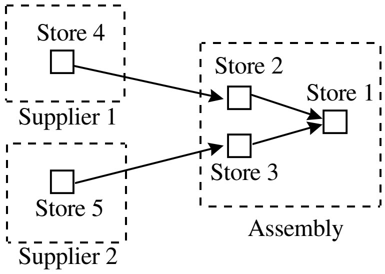

Figure 1: Supply chain for System 1.

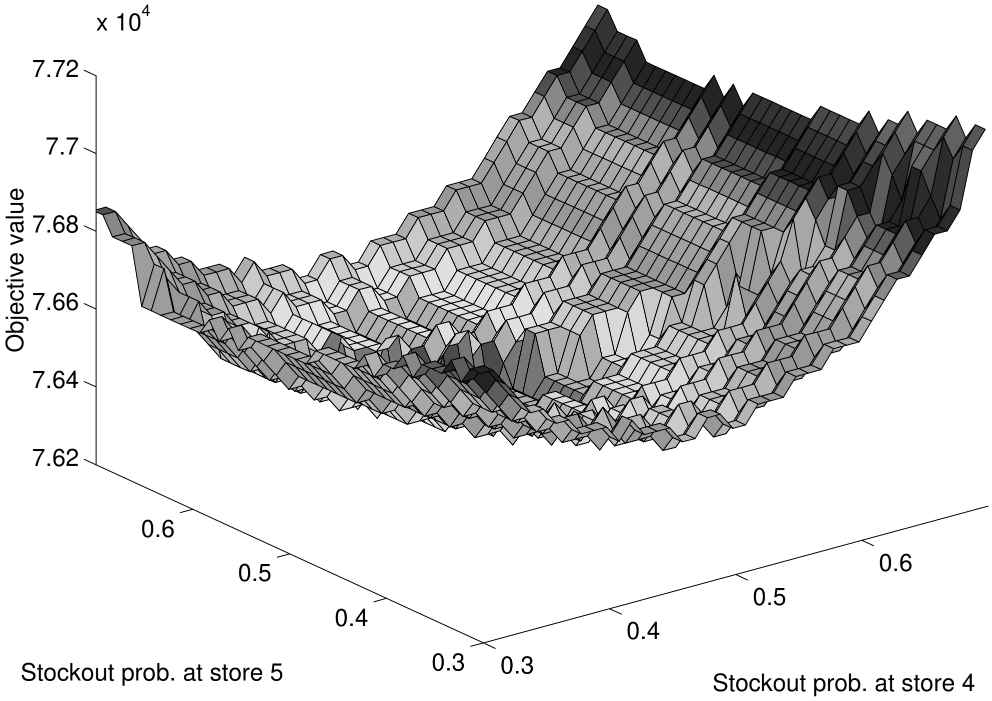

Figure 2: Response surface for System 1.

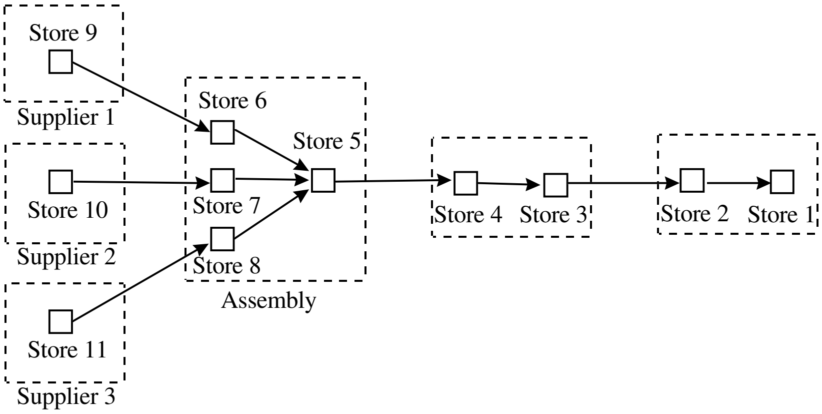

Figure 3: Supply chain for System 2.

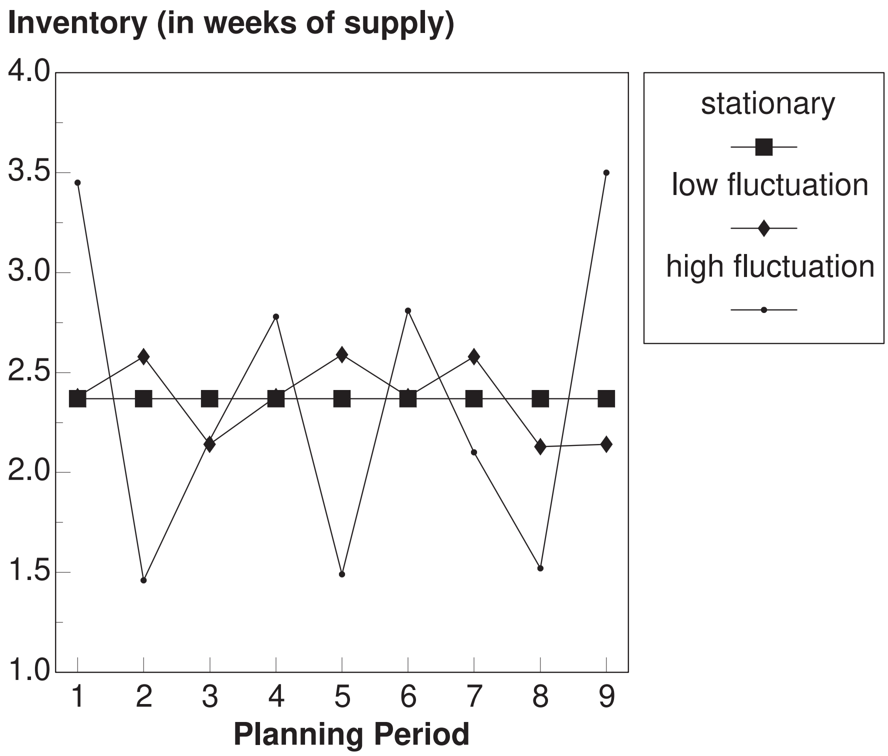

Figure 4: Total inventory capital for System 2.

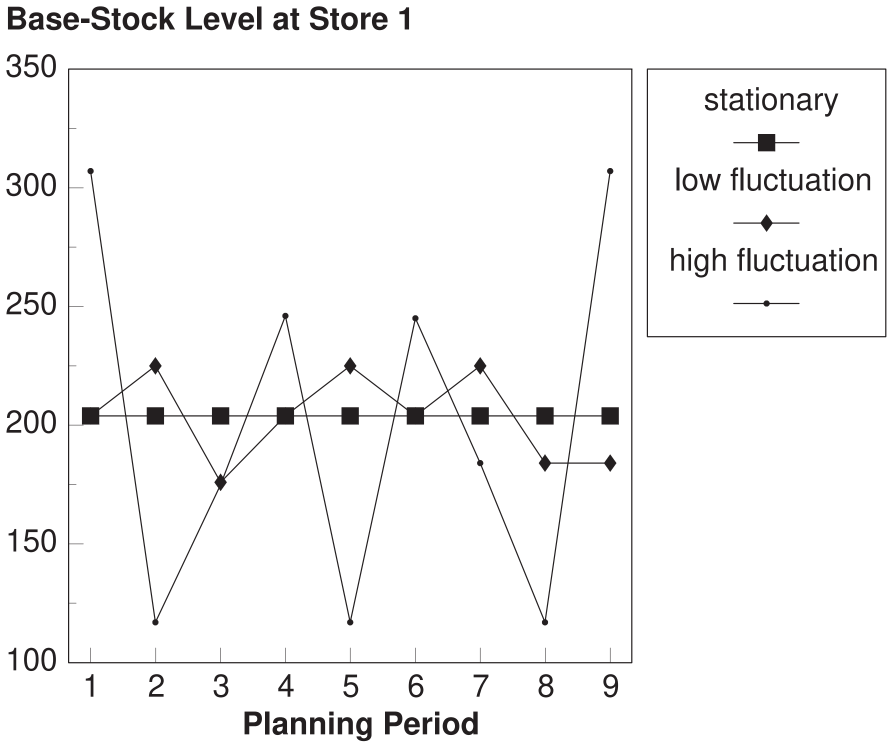

Figure 5: Base-stock level at Store 1 in System 2.

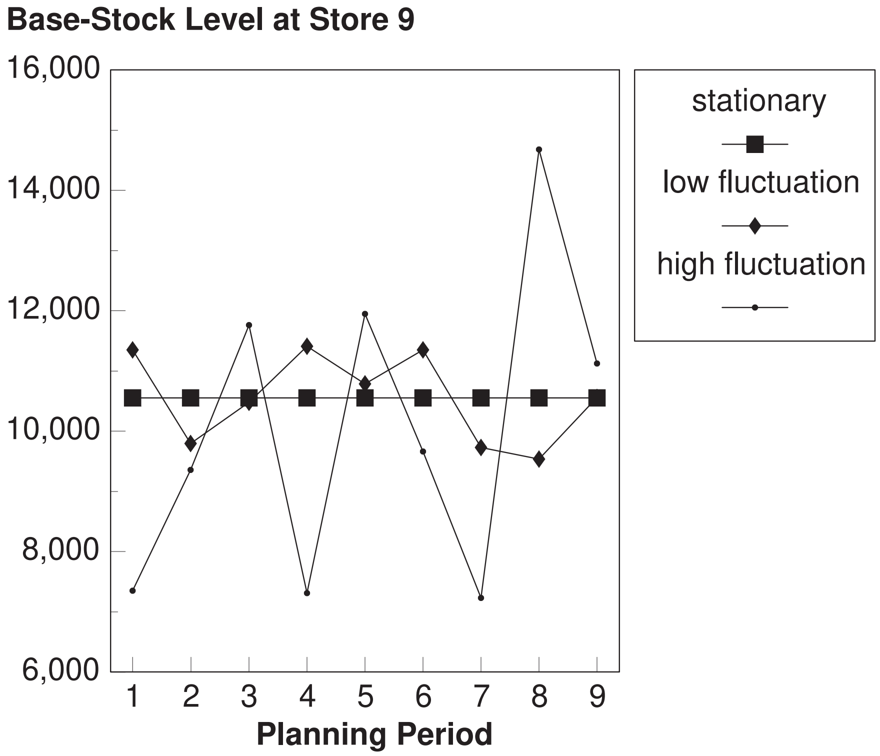

Figure 6: Base-stock level at Store 9 in System 2.