A Reinforcement Learning Approach for Inventory Replenishment in Vendor-Managed Inventory Systems With Consignment Inventory

Zheng Sui, Terra Technology
Abhijit Gosavi, Missouri University of Science and Technology
Li Lin, University at Buffalo, SUNY

Abstract: In a Vendor-Managed Inventory (VMI) system, the supplier makes decisions of inventory management for the retailer; the retailer is *not* responsible for placing orders. There is a dearth of optimization models for replenishment strategies for VMI systems, and the industry relies on well-understood, but simple models, e.g., the newsvendor rule. In this article, we propose a methodology based on reinforcement learning, which is rooted in the Bellman equation, to determine a replenishment policy in a VMI system with consignment inventory. We also propose rules based on the newsvendor rule. Our numerical results show that our approach can outperform the newsvendor.

Keywords: Vendor-Managed Inventory, Supply Chains, Simulation

EMJ Focus Areas: Quantitative Methods & Models

In the past two decades, interest in supply chain management (SCM) has grown rapidly because of the increasing competition in today's global markets. The introduction of products with shortening life cycles and the heightened expectations of customers have forced businesses to invest in and focus attention on their supply chain. A driving force behind effective SCM is the effective collaboration of the numerous activities involved. A lack of coordination in the associated activities in general results in the well-known bullwhip effect, and more specifically in low service levels, high inventory, and high transportation costs. To overcome these problems, many forms of supply chain coordination, such as vendor-managed inventory (VMI) and continuous replenishment and collaborative forecasting, planning, and replenishment (CPFR), have been implemented in recent years in supply chain software. Known as direct replenishment or supplier managed inventory in the early days, VMI was popularized in the late 1980s and is now widely used in various industries. According to a 2003 technology survey (Automotive Warehouse Distributors Association, 2003), over 60% of manufacturers in the U.S. have implemented VMI. A survey of electronic components (Survey, 2003) and a survey of the grocery industry (Kurt Salmon Associates, 2002) also support the contention for a broad use of VMI. VMI has been widely used in the following industries:

- *Grocery*: Campbell Soup Company (Clark and McKenny, 1994), and Barilla SpA (Hammond, 1994),
- *Electronics industry*: Intel (Kanellos, 1998) and Hewlett-Packard (Waller et al., 1999),
- *Food manufacturing*: Kraft Inc. and Mott's USA (Emigh, 1999),
- *Petrochemical*: Shell Chemical (Hibbard, 1998)
- *Men's undergarments*: Fruit of the Loom (Cetinkaya and Lee, 2000).

The use of VMI at Wal-Mart on a large scale has attracted a great deal of attention in the industrial world.

In a VMI program, the supplier assumes control of the inventory management for one or more retailers (Fry, 2002). The supplier monitors the inventory level at the retailers and makes the decisions related to the quantities of replenishment and the timing of shipments to the retailers. It has been claimed that this can play a significant role in reducing the bull-whip effect. Compared to traditional retailer-managed inventory (RMI) programs in which retailers are responsible for placing orders and are entirely responsible for their own inventory shortages or excesses, VMI can bring benefits to both retailers and manufacturer, as discussed next. Some of the advantages for the retailer are as follows:

- *Increase in the service level:* As the supplier has access to databases at the retailer and to the current demand at the retailer, the supplier can better coordinate its own activities of supplying materials to its various customers. As a result, the out-of-stock situation is rarely encountered at the retailer, improving its service levels.
- *Reduction of inventory levels:* Since the supplier has access to inventory databases at the retailer, the supplier is able to coordinate replenishment decisions in a way that keeps inventory levels at the retailer from being excessive. This leads to reduced holding costs and increased inventory turns at the retailer.
- *Reduction of ordering and planning costs:* Ordering and planning costs are eliminated (or reduced) for the retailer since the responsibility of ordering/planning is shifted to the supplier.

From the supplier's perspective, the advantages are:

- *Ease in coordination of supply process:* Since the supplier has access to the retailer's inventory database, as well as its recent sales transactions, the supplier can more easily coordinate its own activities of supplying goods to various retailers. In particular, it is not overloaded by orders at the beginning of the week or month as is the case in RMI. Often in RMI the supplier simply cannot meet all the demand since orders

Refereed management tool manuscript. Accepted by Associate Editor Murray.

from different retailers tend to come at the same time. As a result, the supplier can increase its own profits.

- *Reduced transporters*: Ideally, since the supplier can better coordinate its activities, it can do with fewer trucks or, at the very least, it can reduce the number of trips in which the truck load is not full; a trip with "less-than-truckload" shipments are relatively expensive.

In a *consignment-inventory* VMI system, the supplier retains ownership of the inventory at the retailer. Until the item is sold at the retailer, payment is not made to the supplier, so the inventory-holding costs are absorbed by the supplier. In this article, we present a model for VMI systems with consignment inventory. Our optimization model will be built from the perspective of the supplier, but VMI systems have significant benefits for both suppliers and retailers.

The sudden shift in responsibility of managing retailers' inventory and dealing with the associated risks is "like jumping into a cold pool early in the morning" (Betts, 1994). Betts also quotes a vice president at a Cleveland firm as saying, "…if the scheme does not change the production process or squeeze out excess costs and inventory, then VMI has really just shifted costs to the vendor." In addition, VMI sometimes involves high transportation costs, since the supplier has to ship more frequently to achieve inventory-turn targets at the retailers (Copacino, 1993). These issues make it quite challenging for the supplier to operate a VMI program. Some of the important questions that arise are: How much should one replenish, i.e., what should the replenishment quantities be? If trucks are used as transporters, how many truckloads should be dispatched in a given day?

This article develops models that address these questions. Poor decision-making on these problems can prevent the supplier from enjoying the benefits of VMI. Some MRP programs, such as SAP and I2, have addressed this issue in designing their software; however, the algorithms used are not always transparent to the user, so it is unclear how appropriate these solutions are.

The literature on solving supply-demand problems is extensive. Versions of the problem we consider have been studied since Veinott (1965) and Evans (1967), where the focus was on developing optimal strategies for inventory allocation in multiple retailer scenarios. More recently, Higginson and Bookbinder (1995) used a Markov decision process in their model for shipment consolidation. Van Roy et al. (1997) also developed a model based on Markov decision processes and use neuro-dynamic programming for solution purposes. Their model assumed identical retailers, no transportation costs, and fixed (non-random) transportation time, which allowed them to use a Markov decision process. DeCroix and Arreola-Risa (1998) have characterized an optimal policy for production and inventory control under a finite resource and have also developed a heuristic for solving the problem studied. Cetinkaya and Lee (2000) present a model based on renewal (reward) theory in which the timing of the shipment is determined along with its quantity. Axsater (2001) provided a simple procedure and an exact optimization procedure for the model. Chaouch (2001) developed a VMI model with demand consisting of two components: one deterministic and one random. Their result showed that for a fixed delivery rate, the order-up-to level can be determined much like the optimal stock level in the newsvendor model. Fry et al. (2001) built an interesting model for a type of VMI agreement, called the *(z, Z)* contract, based on their analysis of a number of VMI systems in practice, and developed a solution that satisfied a "newsvendor–type" relationship. Cheung and Lee (2002) built a model of coordinated shipment and stock rebalancing in the VMI system and examined the benefits of two initiatives to the supplier and the retailers. Bernstein and Federgruen (2003) considered a "centralized" system, which is similar to a VMI system, in which the supplier determines sales quantities and the complete chain-wide replenishment strategy. They considered a problem in which the pricing of the product is an additional decision variable for the retailer. Subramaniam and Gosavi (2007) developed a simulation-based optimization approach based on simultaneous perturbation to optimize the inventory dispatching policies; however, the policies they obtained are *static* and do not depend on the dynamically changing state of the inventory levels at the retailers.

Contributions of this Article

This research is focused on determining the replenishment quantities and the number of trucks dispatched by the vendor to each retailer, assuming that truck routes are fixed and known. We make many assumptions that real-world systems share. In particular, we consider multiple non-identical retailers with non-zero transportation times between retailers and the vendor and assume the transportation times to be random variables. Transportation costs are also assumed to be non-negligible. We believe transportation costs should be an important part of any VMI model, since transport frequency can be quite high in a VMI setting. Our solution methodology is based on reinforcement learning (RL), which is an approximate dynamic programming method rooted in the Bellman equation for the semi-Markov decision process. To show the usefulness of our methodology, we conduct a series of numerical tests. In any reinforcement learning based application, unless there is a way to determine the optimal solution, it is customary to compare the solution's performance to that of some other well-accepted approach, i.e., benchmark. This is because RL requires a great deal of tuning and experimentation, and its behavior can be highly unpredictable; therefore, without such a comparison, one is usually unsure of the quality of the solution. We use the newsvendor model as the benchmark. Our numerical results show that our method outperforms the newsvendor model.

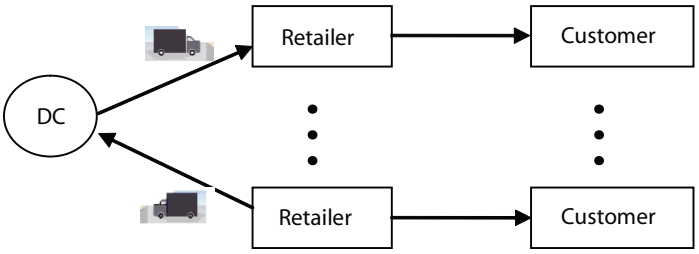
The choice of the newsvendor as a benchmark is natural here for at least three reasons. First, the newsvendor rule elegantly captures the tradeoff between under-stocking (penalty) and over-stocking (holding) costs in a single-period periodic review setting, to which class our problem belongs. Second, the newsvendor model can be easily used by any engineering manager within spreadsheet-based software. The usefulness of new methods is usually best shown via comparisons to methods that can be easily implemented in practice, because favorable comparisons are likely to increase their practical appeal. Also, although the newsvendor rule was proposed a long time ago, its variants continue to be used in supply chain software today. The newsvendor, like the Economic Order Quantity (EOQ), is a well-understood, transparent model that is also quite versatile and can be adapted easily to a large number of periodic-review settings in supply chains. Finally, the recent work of Suo et al. (2004) and Wong et al. (2008) in VMI systems is reflective of the fact that the newsvendor continues to be a baseline VMI model for academic research, while the solution in Fry et al. (2001) exploits a newsvendor-type relation for setting the upper and lower limits of their contract.

Most of the existing analytical models in the literature are developed under a specific set of system assumptions that deviate from ours. The model in Bernstein and Federgruen (2003) considers a pricing-cum-replenishment problem in a game-theoretic setting in which they extensively use the EOQ formula. The model in Fry et al. (2001) does not consider transportation costs, and is developed for a single-product single-retailer combination. Further, theirs is not a consignment VMI system, unlike ours, and their holding costs at the supplier and the retailer are modeled separately. Hence, under the assumptions that we make, comparison of these models to our simulation-based methodology is not feasible.

Problem Description

In this article, we focus on a two-echelon system with one Distribution Center (DC) and the retailers that the DC serves (see Exhibit 1). We will assume that the route followed by the truck (or fleet of trucks) is already known, and we make no attempt to optimize it. The supplier pays the holding cost of the inventory at the retailers (in a consignment inventory VMI system). Whenever a stock-out occurs at the retailer, the supplier is penalized. The manufacturer has to decide upon the timing and quantity of the order from the manufacturer to the DC, which we assume is done via a *(Q, R)* policy. The review of the inventory at the retailers is performed when the truck fleet comes back from the retailers to the DC, and the new decision is to be made regarding the number of trucks to be sent in the next period. We refer to the period between two successive inventory reviews as a cycle.

Exhibit 1. The 2-Echeleon System Considered in this Article



In one cycle, the following events occur: Customer demand arrives at the retailers. We model this via a compound Poisson process. The retailer's demand forecast update refers to whether the demand has been predicted to be high or low at the retailer. The demand forecast for retailer *i* is denoted by *I(i)*, and the *n*-tuple $\vec{I}$ collectively denotes the demand forecast at all retailers. The retailers' current inventory levels are denoted by the *n*-tuple $\vec{x}$, with *x(i)* denoting the inventory at the *i*th retailer. Similarly, the DC's inventory level is denoted by another *n*-tuple $\vec{y}$. After the products are loaded onto the trucks, the truck fleet departs from the DC. The majority of the literature assumes that the distribution of the demand at the retailers is either normal (Gavirneri et al., 1999; Jackson, 1988) or Poisson with a demand of size always equaling 1 (Cetinkaya and Lee, 2000; Deuermeyer and Schwarz, 1981; Graves, 1996). In this work, we assume the customer demand to be a compound Poisson process, where the demand size is assumed to have the discrete uniform distribution. Our use of this distribution is motivated by the fact that the compound Poisson process is more general than a Poisson process with demand size equaling 1. Also, the normal distribution is *not* easily usable in a low demand scenarios because it can become negative (Nahmias and Smith, 1993). In addition, we assume that the demand is signaled as either "high" or "low," where a high demand is characterized by a higher arrival rate and a low demand by a lower arrival rate. With this we have attempted to capture a modern trend in modeling demand in supply chains called forecast updating (Sethi et al., 2001). Forecast updating is a broader concept and requires updating of the entire forecast and perhaps even the demand distributions on the basis of signals received about the state of the demand. The customer's demand, *D*, at any retailer for a given product in one cycle, can be modeled as: $D = \sum_{i=1}^{N} d_i$ where

- d_i denotes the demand from the *i*th customer, which has a discrete uniform distribution.
- *N* is a Poisson distributed random variable with parameter λ; essentially *N* stands for the number customers that arrive in one cycle, and the value of λ depends on *i,* the retailer.

Some additional notation that we need is:

- c_T: the transportation cost is the cost of operating one truck for unit time.
- h_r: the holding cost of one item for unit time at a given retailer
- p_r: the cost of stock-out penalty at a given retailer.
- *Rev*: the revenue generated for the supplier when the sale of a product takes place at the retailer.
- *Cap*: the capacity of one truck.
- $t_{DC\text{-}RET}$: the time of travel between DC and a retailer.
- $t_{RET\text{-}RET}$: the time of travel between retailers.
- t_{RET}: the service time at the retailer.
- t_{DC}: the service time at the DC.
- t_0: the lead time for the DC's order from the manufacturer.

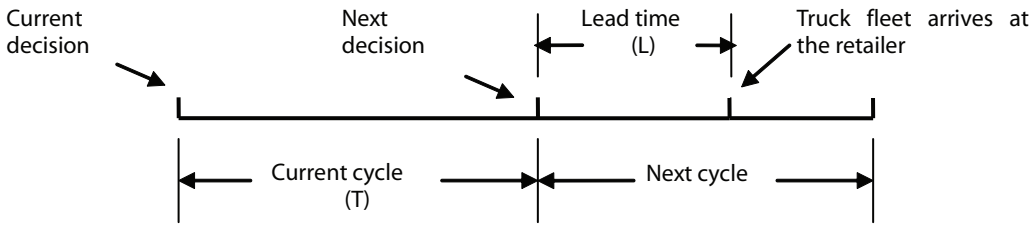
One of the other decisions to be made at the DC, which our RL approach does not seek to optimize, is its own inventory management policy. We will assume that it uses a *(Q, R)* policy (Askin and Goldberg, 2003). When such a policy is pursued, the inventory level is observed continuously. As soon as it becomes less than *R*, an order of size *Q* is delivered from the manufacturer to the DC. There are a number of algorithms to find the optimal values of *Q* and *R* (Askin and Goldberg, 2002). We let t_o denote the lead time for the DC's order from the manufacturer.

At the start of each cycle, the inventory level of the retailers, $\vec{x}$ the inventory level of the DC, $\vec{y}$, and the demand forecast of the retailers, $\vec{I}$, are observed. Then the decision to be made is related to (i) the number of trucks to be sent and (ii) the allocation of the amounts of each product to the total amount within each truck. Sub-optimal decisions can either lead to excessive holding costs or stock-out costs at the retailers. Indeed, it can also lead to excessive holding costs for one product and stock-out costs for some other product. We will discuss our RL approach in the next section. We now describe a robust approach based on a well-known paradigm.

Newsvendor Solution

One approach to solving this problem is to use the newsboy or newsvendor rule. The newsvendor rule is designed for perishable items that have a holding cost and a stock-out cost. In order to adapt the model to our problem setting, we need some additional work that we now present.

Exhibit 2. Lead Time and Cycle Time



The value of the retailer's order up-to level S^* is determined in the newsvendor model by solving:

$$F(S^*) = \frac{p}{p+h}$$

where p is the penalty cost, h is the holding cost, and $F(.)$ is the cumulative distribution function of the customer demand in the cycle. Let T denote our cycle time and L denote the lead time for the retailer in question (see Exhibit 2). L is equal to the truck fleet transportation time from the DC to the retailer in question. Hence, the inventory replenishment decision made at the start of the current cycle will affect the inventory system from the time the current cycle starts until the truck fleet arrives at the retailer in the next cycle. In our setting, the customer demand is assumed to be a compound Poisson process with a demand forecast update and is defined by:

$$D = \sum_{i=1}^{N} d_i$$

where d_i, the amount of demand for ith customer, has a discrete uniform distribution, $(U(a, b))$. In the current cycle, the value of the demand forecast update (I) is known and will be denoted by g. So in unit time, the mean $\mu_{current}$ and variance $\sigma^2_{current}$ of the customer demand can be given as (Ross, 2002):

$$\begin{aligned}\mu_{current} &= g \cdot E(N) \cdot E(d) \\ &= \frac{g\lambda(a+b)}{2}, \\ \sigma^2_{current} &= g^2\left[E(N)Var(d) + Var(N)E^2(d)\right] \\ &= g^2\left[\lambda \cdot \frac{(b-a)^2}{12} + \lambda \cdot (\frac{a+b}{2})^2\right] \\ &= \frac{g^2\lambda(a^2+b^2+ab)}{3}.\end{aligned}$$

Now if we assume T to be a discrete random variable, the mean μ_{cycle} and the variance σ^2_{cycle} of the demand during T time units can be computed (Nahmias, 2001) as:

$$\begin{aligned}\mu_{cycle} &= \mu_{current} \cdot \mu_T, \\ \sigma^2_{cycle} &= \mu_T\sigma^2_{current} + \mu^2_{current}\sigma^2_T,\end{aligned}$$

where μ_T and σ^2_T are the mean and the variance of T. During L time units in the next cycle, the value of the demand forecast update I is unknown; therefore we use the expected value of I, $E(I)$, to estimate it. In unit time of the next cycle, the mean μ_{next} and variance σ^2_{next} of the customer demand can be computed as follows:

$$\begin{aligned}\mu_{next} &= E(I) \cdot E(N) \cdot E(d) \\ &= \frac{E(I) \cdot \lambda(a+b)}{2}, \\ \sigma^2_{current} &= E^2(I) \cdot \left[E(N)Var(d) + Var(N)E^2(d)\right] \\ &= E^2(I) \cdot \left[\lambda \cdot \frac{(b-a)^2}{12} + \lambda \cdot (\frac{a+b}{2})^2\right] \\ &= \frac{E^2(I) \cdot \lambda(a^2+b^2+ab)}{3}.\end{aligned}$$

Again, assuming that L is a discrete random variable, the mean μ_{lead} and the variance σ^2_{lead} of the demand experienced during L time units in the next cycle can be computed as:

$$\begin{aligned}\mu_{lead} &= \mu_{next} \cdot \mu_L, \\ \sigma^2_{lead} &= \mu_L\sigma^2_{next} + \mu^2_{next}\sigma^2_L,\end{aligned}$$

where μ_L and σ^2_L are the mean and the variance of L respectively. Then, via the central limit theorem, the customer demand in the replenishment cycle (T) and lead time (L) can be approximated (Ross, 2002) as:

$$\begin{aligned}\mu &= \mu_{cycle} + \mu_{lead}, \\ \sigma^2 &= \sigma^2_{cycle} + \sigma^2_{lead}.\end{aligned}$$

If the demand is normally distributed with mean μ and variance σ^2, the optimal order up-to level (S^*) is:

$$S^* = \phi^{-1}(\frac{p}{p+h}) \cdot \sigma + \mu,$$

where $\phi^{-1}(.)$ is the inverse of the cumulative distribution function of the standard normal distribution. Let $S^*(i,k)$ denote the order up-to level of retailer i and product k based on the newsvendor heuristic. From these values, one can determine the number of trucks to be sent as follows:

$$a = \left[\!\left[\frac{\sum_{i=1}^{n}\sum_{k=1}^{m}(S^*(i,k) - x_{ik})}{cap}\right]\!\right]$$

where $[\![\cdot]\!]$ denotes the nearest integer of the quantity inside the brackets and x_{ik} denotes the current inventory for retailer i and product k.

Cost Structure

We summarize our cost economics model used in our simulation model. We have four elements for the costs and revenues: the holding cost (h_r) at the retailers for each product (note that this cost is absorbed by the supplier in our model), the revenues (*Rev*) transmitted to the supplier when a sale occurs, the stock-out penalty (p_r) which is transmitted to the supplier, and the transportation cost (c_T) per truck per unit time for the supplier.

Solution Methodology

The problem that we have presented above can also be solved via a more detailed model that looks at the Markov chains underlying the VMI system. In this section, we first present the underlying stochastic process that we have used for developing the model and then discuss the method used for solution.

Semi-Markov Decision Process

The stochastic process that we use in our model is called the semi-Markov decision process (SMDP) (Bertsekas, 1995). The SMDP is characterized by a set of states, actions, rewards, transition times, and transition probabilities. In each decision-making state, the agent can choose from a set of actions. The random trajectory followed by the system depends on the action chosen in each state and upon the random variables that govern the system's behavior. Underlying the semi-Markov process is an embedded Markov chain. Let *p(i,a,j)* denote the *transition probability* of transitioning from state *i* to state *j* under the influence of action *a* in the Markov chain associated to action *a.* Similarly, let *r(i,a,j)* denote the immediate reward (revenue) earned in the same transition. Then, the expected immediate reward earned in state *i* by selecting action *a* is: $\bar{r}(i,a) = \sum_j p(i,a,j)r(i,a,j)$. In an SMDP model, the time of transition from one state to the next is also a random variable with an arbitrary distribution. So if *t(i,a,j)* denotes the mean time in transiting from state *i* to state *j* under *a*, the mean time in any transition out of *i* under *a* is given by $\bar{t}(i,a) = \sum_j p(i,a,j)t(i,a,j)$.

Typically, when we study the system over an infinite time horizon, the objective function is to maximize a cumulative function of the immediate rewards (or costs) earned in each state transition. Two performance metrics frequently used in stochastic control theory are: the *average reward,* which is the expected reward per unit time calculated over an infinitely long trajectory, and the *discounted reward,* which is the expected total discounted reward calculated over an infinitely long trajectory. Solving the SMDP with respect to the discounted reward metric requires solution of the following Bellman equation:

$$J(i) = \max_{a \in A(i)} \left(\sum_j p(i,a,j) \left\{ r(i,a,j) + \int_0^\infty \exp(-\gamma t) J(j) F_{ia}(t)\, dt \right\} \right),$$

where $F_{ia}(t)$ denotes the cumulative distribution function of the time spent in transition out of *i* under *a*, γ denotes the discount rate (with $\exp(-\gamma t)$ being the discount factor during a time interval of length *t*), *A(i)* denotes the set of actions allowed in state *i*, and *J(i)* denotes the value function for state *i*. The solution to the SMDP can be expressed as a policy π, which prescribes $\pi(i)$ as the action to be taken in state *i*.

We now consider the average reward performance metric. If the starting state for the system is *s*, the average reward (performance criterion) of a given policy π can be defined as:

$$\liminf_{T\to\infty} \frac{E\{\sum_{m=1}^{T} (r(i_m, \pi(i_m), i_{m+1}) | i_1 = s)\}}{E\{\sum_{m=1}^{T} (t(i_m, \pi(i_m), i_{m+1}) | i_1 = s)\}}$$

where *E* denotes the expectation operator and i_m denotes the state in the *m*th jump of the embedded Markov chain associated with policy π. It can be shown that if the Markov chain associated with every policy is "regular," then the above criterion does not depend on the starting state. Our goal in average-reward SMDPs is to maximize the above function, which is also called the long-run average reward of the SMDP. As $\gamma \to 0$, which implies that $\exp(-\gamma t) \to 1$, from the Bellman equation for discounted reward, it can be shown that optimization with respect to discounting becomes equivalent to optimizing with respect to the average-reward criterion. Use of such a small value of γ, so that the discount factor tends to 1, is called a vanishing-discount approach. In this article, we will use a discounting algorithm, but via the vanishing discount approach will solve an average-reward problem.

SMDPs can be solved by via classical dynamic programming method (Bertsekas, 1995), provided the number of states is small and the system has a tractable transition probability model; however, for a problem with a large number of states, which is true of the problem considered in this article, it generally becomes difficult to store all the transition probabilities, *p(.,.,.)*, and the elements of the value function, *J(.)*. This is attributed to the curse of dimensionality (Bellman, 1957). If the transition probabilities are too difficult to find in an exact manner because of the complex stochastics of the underlying random process, the analysis is said to have the curse of modeling (Sutton and Barto, 1998); this may be true even of problems with a few states. One approach to avoid these twin curses of dynamic programming is to use the methodology called reinforcement learning (RL), which we next describe.

Reinforcement Learning (RL)

RL has attracted a considerable amount of attention recently (see Sutton and Barto, 1998; Bertsekas and Tsitsiklis, 1996; Gosavi, 2003) for textbooks on this topic). Watkins (1989) discovered the Q-Learning algorithm, which is one of the most popular algorithms in RL, to solve the MDP in a simulator. It was extended to the SMDP in Bradtke and Duff (1995) where they used a continuous reward rate. We use the algorithm proposed in Gosavi (2003) for the lumpsum reward that is applicable to the problem here. The algorithm's convergence has been established in Gosavi (2007). The algorithm is described next.

Steps in the Q-Learning Algorithm

Step I. Let S denote the set of states, and *A(i)* denote the set of actions allowed in state *i*. Initialize the Q-factors, *Q(i,u)* = 0 for all $i \in S$ and all $u \in A(i)$. Set *m=0* and γ to a very small positive number, e.g., 0.001. Compute the step-size using a rule such as $\alpha = A/(B+m)$, where *A* and *B* are constants, e.g., *A= 99* and *B=100*. Set MAX_STEPS to a large positive integer, e.g., 10,000. Start system simulation.

Step II. While m < MAX_STEPS do:

Let the system start in state *i*.

1. With probability of $1/|A(i)|$ (note that $|X|$ denotes the number of elements in set *X*), choose an action $a \in A(i)$ that maximizes $Q(i,a)$. (In other words, compare all the Q-factors for state *i*, and choose the action for which the Q-factor is the maximum).

2. Simulate the chosen action a. Let the system state at the next decision epoch be j.
3. Update $Q(i,a)$ using the following rule:

$$Q(i,a) \leftarrow (1-\alpha)Q(i,a) + \alpha[r(i,a,j) + \exp(-\gamma t(i,a,j)\max_{b \varepsilon A(j)} Q(j,b)]$$

4. Set the current state i to the new state j. Increment m by 1, re-compute α, and then go to Step II(*1*).

When the algorithm has run for MAX_STEPS, we can identify the policy π returned from the Q-factors as follows. For all $i \in S$, $\pi(i) = \text{argmax}_{a \epsilon A(i)} Q(i,a)$.

It is to be noted that the algorithm given above does not need the transition probabilities of the underlying Markov chains, but can be used within a simulator. Thus the algorithm avoids the curse of modeling. Dynamic programming algorithms require these transition probabilities, which are notoriously hard to compute in a real-world problem such as the one we study here; however, the problem of having to store a large number of Q-factors remains. This issue is resolved with the use of function approximation that we next describe.

Function Approximation

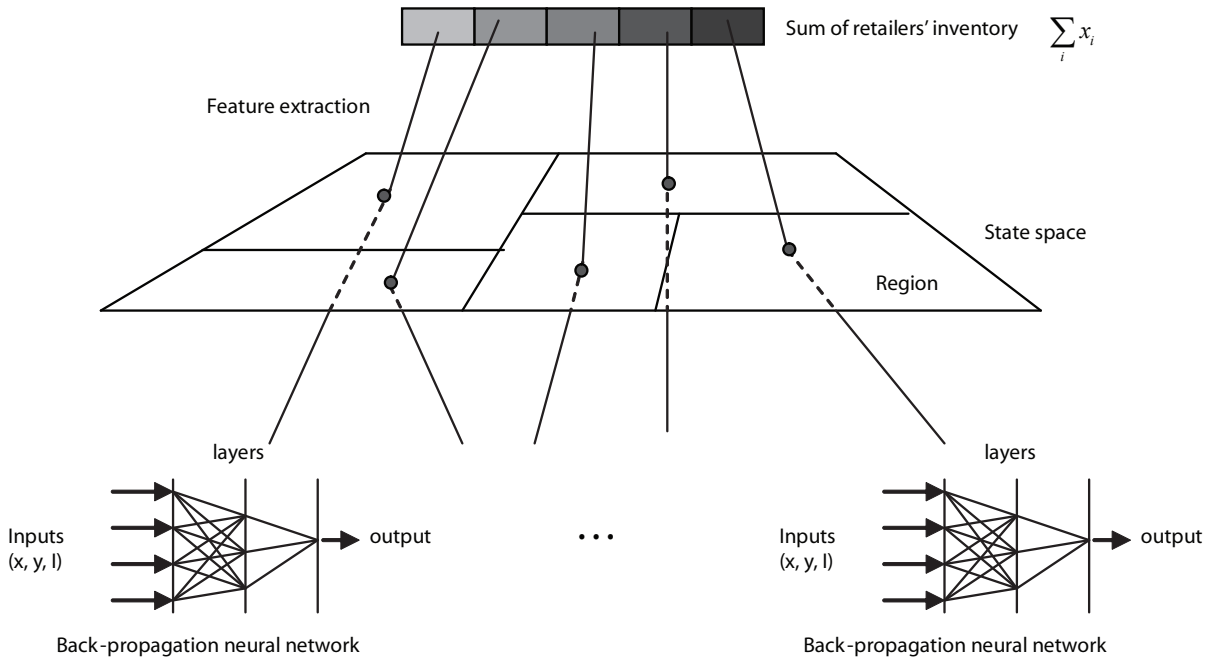
The idea underlying function approximation is to model the Q-factors for a given action as a function of the state, and store only the relatively smaller number of scalars that define a function instead of storing all the Q-factors explicitly. This is said to avoid the curse of dimensionality. Of course, the function that we store should be an appropriate one. In other words, the function should return a reasonably accurate estimate of the Q-factor's value when the relevant state and action are fed into it. For instance for a 2-action problem, consider the following linear representation (where the state is a scalar): $Q(i,a) = \Omega + \kappa i$ for $a = 1$ and $Q(i,a) = \psi + \omega i$ for $a = 2$, where i denotes the state. Then instead of storing the Q-factors for all state-action pairs, we can do with storing only 4 scalars: Ω, ω, κ, and ψ. This strategy works well if the Q-factors are linear or nearly linear functions. In most complex problems, however, the Q-factors tend to be non-linear functions, whose closed forms are unknown. In such a scenario, the use of back-propagated neural networks has been widely cited in the literature on RL (Tesaru, 1995; Crites and Barto, 1998). Hence, in this article, we use a back-propagated neural network (Werbos, 1974) to approximate the Q-factors. See Exhibit 3.

Although the RL algorithm can help identify the number of trucks to be dispatched at the start of the cycle, it cannot solve the problem of allocating the inventory to different products and retailers. To this end, we present a technique to allocate the inventory that must "help" the RL algorithm.

Inventory Allocation

An allocation method for the products to be sent in a given truck has been discussed in the literature of inventory management since the late 1950s. Simpson (1959) discusses an optimum allocation policy to minimize the weighted number of lost sales, in which the necessary condition is to equalize a weighted probability. Simpson's allocation method has two drawbacks that make it unsuitable for the problem in this research. First, only lost sales are considered in the allocation method; the holding cost at each retailer is also not taken into account. The other drawback is that it is not easily implemented when the retailers are not identical. This drawback is also seen in other works (Jackson, 1988; Jonsson and Silver, 1987). Bertrand and Bookbinder (1998) consider non-identical retailers, and use a search procedure to determine the allocation method. Using these ideas in the literature, we propose a rule for inventory allocation, which can work for non-identical retailers, and is computationally easy. The aim here is to make the ratio of the retailer's inventory after allocation, which equals the sum of the current retailer's inventory and the allocated inventory, to the retailer's order up-to inventory level (computed via the newsvendor model), the same for each retailer and each product. That is, $\frac{x_{ik} + Inv(i,k)}{S^*(i,k)}$ should be roughly equal for all *(i, k)* pairs, where x_{ik} is the current inventory of retailer i and product k, and *Inv(i,k)*

Exhibit 3. The Neural-Network Architecture for the Function Approximator Used



is the inventory allocated to retailer *i* and product *k*. We must also ensure that $\sum_i \sum_k Inv(i,k) = a.cap$. The computational scheme for this can be described as follows: For every *(i,k)* combination, compute the following:

$$Y(i,k) = \left[\!\left[\frac{(\sum_i \sum_k x_{ik} + a.cap)S^*(i,k)}{\sum_i \sum_k S^*(i,k)} \right]\!\right] - x_{ik}.$$

(In the above, $[\![.]\!]$ denotes the nearest integer for the quantity inside the brackets.) If *Y(i,k)< 0, Inv(i,k) =0,* and otherwise *Inv(i,k)= Y(i,k).* Note that this rule uses the newsvendor relationship.

Numerical Experiments

In this section, we describe the results of our numerical experiments with the technique developed in the previous section. We will consider a scenario with two products and ten retailers. The parameters of each product at different retailers are listed in Exhibit 4. The values of the parameters of each product at the DC, the transportation time, and truck-related information are listed in Exhibit 5. The parameters in Exhibits 4 and 5 present our baseline case. The parameters for the other cases are defined in Exhibit 6. Exhibit 7 presents the results obtained with all the eight cases that we study.

Exhibit 4. Parameters (Demand Arrival, Holding Costs, Stock-Out Penalties and Revenues) for the Retailers and the Products

Product	*Retailer*	*λ*	*Demand Uniform(a,b)*	h_r	p_r	*Rev*
1	1	*0.25*	(1, 2)	0.06	4	5
	2	0.5	(0.5, 1.5)	0.05	4	5
	3	0.3	(1, 2)	0.03	4	5
	4	0.25	(1, 2)	0.04	4	5
	5	0.1	(2, 4)	0.03	4	5
	6	0.2	(1, 3)	0.05	4	5
	7	0.3	(1, 1.5)	0.03	4	5
	8	0.5	(0.5, 1.5)	0.06	4	5
	9	0.15	(2, 3)	0.04	4	5
	10	0.2	(1, 3)	0.05	4	5
2	1	0.5	(0.5, 1.5)	0.04	3	5
	2	0.1	(2, 3)	0.06	3	5
	3	0.15	(1, 3)	0.04	3	5
	4	0.3	(0.5, 2)	0.05	3	5
	5	0.35	(0.5, 1.5)	0.03	3	5
	6	0.25	(1, 2)	0.06	3	5
	7	0.4	(0.5, 1.5)	0.04	3	5
	8	0.2	(1, 3)	0.03	3	5
	9	0.15	(1.5, 2.5)	0.05	3	5
	10	0.25	(1, 2)	0.03	3	5

Exhibit 5. Additional Parameters for Our Numerical Experiments

Parameters	*Values*	
	Product 1	*Product 2*
h_{DC}	0.005	0.005
p_{DC}	1	1
(Q, R)	(500, 150)	(500, 150)
K	50	50
$t_{DC\text{-}ret}$	Uniform (2, 4)	
$t_{ret\text{-}DC}$	Uniform (2, 4)	
$t_{ret\text{-}ret}$	Uniform (0.5, 1)	
t_{DC}	Uniform (0.2, 0.3)	
t_{ret}	Uniform (0.01, 0.015)	
t_0	Uniform (30, 50)	Uniform (30, 50)
c_T	10	
Cap	100	

Exhibit 6. Designed Experiment for Parameter Values

Factors	*Level (-1)*	*Level (+1)*
p_r	Original values	Increase the value by 50%
h_r	Original values	Increase the value by 100%
λ	Original values	Increase the value by 50%

We now describe how our value function is approximated. The inventory at each retailer and DC is encoded using "buckets" (Sutton and Barto, 1998). The state is composed of: the retailers' inventory $x = \{x_{11}, ..., x_{mn}\}$, where x_{kj} denotes the retailer's inventory of the *k*th product at the *i*th retailer; the DC's inventory $y = \{y_1, ..., y_m\}$, where y_k denotes the DC's inventory of the *k*th product; and the retailers' demand forecast update $I = \{I_{11}, ..., I_{mn}\}$, where I_{ik} denotes the forecast update of the *k*th product at the *i*th retailer; *m* is the number of the product items and *n* is the number of the retailers. The actual state space is too large to store all the Q-factors explicitly. We used a neural network for which typically the state space must first be encoded. We encode the inventory to generate signals (levels) for the neural network as follows. Let the inventory for a given retailer have a maximum value of *b* and a minimum value of *a*, with *c, d*, and *e*, chosen in a manner such that: $a < c < d < e < b$ and $c - a = d - c = e - d = b - e$. Inventory values from *a* to *c* will be assumed to have a signal of 1, those from *c* to

Exhibit 7. Results of Using the RL Algorithm and the Newsvendor Heuristic

Case	p_r	h_r	λ	*Newsvendor heuristic*	*RL-based algorithm*	*% Improvement*
1	-1	-1	-1	15.41	16.31	5.84
2	-1	-1	+1	25.68	26.75	4.17
3	-1	+1	-1	7.55	8.15	7.95
4	-1	+1	+1	15.01	15.86	5.66
5	+1	-1	-1	14.86	15.7	5.65
6	+1	-1	+1	24.99	25.93	3.76
7	+1	+1	-1	6.39	6.80	6.42
8	+1	+1	+1	13.60	14.21	4.49

d will generate a signal of 2, those from *d* to *e* a signal of 3, and those from *e* to *b* a signal of 4. For the case of two products and ten retailers, we used 5 signal levels for the inventory of each product at the retailers, 3 signal levels for inventory of each product at the DC, and 2 signal levels of each demand forecast update (one for high and one for low) for each product. We assume three actions: 0 trucks, 1 truck and 2 trucks. The number of these encoded Q-factors is $5^{2\times10}\times3^2\times2^{2\times10}\times3=2.7\times10^{21}$. (Note that before the encoding, the state space is even larger!) With the look-up table method in which the values of the encoded Q-factors are stored explicitly, if it takes one byte to store a value of Q-factor, we will need a memory of 2.7×10^{12} GB to store all the Q-factors, which is clearly unrealistic for current computer technologies. The state space is separated into 50 regions. In each region, we place a neural network to approximate the values of the Q-factors in that region (See Exhibit 3). The (Q,R) policy for the DC is computed using the technique in Nahmias (2001). We run each case for 1,000,000 units of simulation time in the learning stage, and 10 replications with each replication lasting 100,000 units of simulation time in the frozen stage. For Case 1 of our experimental setup (see Exhibit 7), the average profit for the entire VMI system using the RL-based algorithm proposed in the work is 16.31. Via the newsvendor heuristic, the average profit is 15.41. The RL-based algorithm outperforms the newsvendor heuristic by 5.84%. Note that the RL algorithm uses the newsvendor in deriving its allocation strategy. The highest improvement over the newsvendor is about 8%, and the lowest improvement is about 4%. The problems that we solved did not require more than 10 minutes on a PC. These are encouraging results because they show that our RL algorithm outperforms a newsvendor. The newsvendor heuristic, which is essentially a by-product of our analysis, appears to be a solid method in its own right; what is attractive about the newsvendor is that it can be implemented easily on any spreadsheet software.

Conclusions

Optimizing the replenishment quantities for a VMI system is a problem that has been studied in the literature mostly via analytical models. Usually, in the process of constructing analytical models, one has to make simplifying assumptions about the system. In this article, we presented a simulation-based approach to determine the replenishment quantities. The attractive feature of the simulation-based approach is that one can make realistic assumptions about the system. We used a reinforcement-learning model based on semi-Markov decision processes for solution purposes. Our model requires solution of the newsvendor problem. As a by-product, our analysis also resulted in a robust newsvendor heuristic. Our approach outperformed the newsvendor solution by at least 4% in all our experiments. Our computer programs generated solutions within minutes for the problems we solved, which we view as a positive outcome. As such, our approach can easily be used by an engineering manager. The newsvendor heuristic can be implemented within any standard spreadsheet software.

There are some clear implications of our work for the practicing engineering manager. When VMI systems are used, they need to be carefully optimized. Newsvendor-based rules and simulation-based approaches, such as the one proposed here, can produce a significant impact on system profits. The computer programs needed for these approaches can produce solutions within minutes. The newsvendor-based rules can be programmed within spreadsheet software.

We should point out that our simulation-based approach could potentially be used in any supply chain where the supplier has to determine the shipment quantities to be sent to multiple retailers. That is a problem with a broader scope; however, some of the assumptions that we have made, e.g., holding and stock-out costs absorbed by suppliers, would have to be altered to construct a more general model. It is not difficult to make such changes in the simulation model, and hence this can potentially form an

avenue for additional research. A number of other directions for future research can be envisioned.

First, the policy considered in this article belongs to the periodic review class in which the inventory is reviewed at the start of the cycle, when the truck (or trucks) comes back. There is a body of literature that looks at *(Z,z)* type of policies (including in particular Fry at al, 2001) for VMI systems. An exciting challenge would be to use reinforcement learning for deriving such policies. Second, optimization of the routes for the truck fleet along with optimization of the replenishment quantities is another direction in which the simulation-optimization procedure could be developed. It turns out that there is an interesting tabu-search procedure (Gendreau et al, 1994) that could be integrated within the simulation-optimization framework. Third, optimizing the system for both the retailer and the supplier would require a game-theoretic formulation. This would perhaps require first showing the existence of Nash equilibria, but identifying a solution that is optimal for both the retailers and the suppliers should provide insights that our single-agent optimization model cannot provide. Finally, including the manufacturer in the optimization model will form a very interesting topic for further research. The full impact of the bull-whip effect can only be studied if all three levels are taken into account. It will be also interesting to consider the impact of risk (Bahill and Smith, 2009) within the analysis.

References

Askin, Ronald., and Jeffrey Goldberg, *Design and Analysis of Lean Production Systems*, Wiley (2002).

Automotive Warehouse Distributors Association, *2003 AWDA Technology Survey*, www.awda.org (2003).

Axsater, Sven, "A Note on Stock Replenishment and Shipment Scheduling for Vendor-Managed Inventory Systems," *Management Science, 47*:9 (2001), pp. 1306-1310.

Bahill, Terry, and Eric D. Smith, "An Industry Standard Risk Analysis Technique," *Engineering Management Journal*, *21*:4 (2009), pp. 16-29.

Bellman, Richard E., *Dynamic Programming*, Princeton University Press (1957).

Bernstein, Fernando, and Awi Federgruen, "Pricing and Replenishment Strategies in a Distribution System with Competing Retailers," *Operations Research, 51*:3 (2003), pp. 409-426.

Bertrand, Louise P., and James H. Bookbinder, "Stock Redistribution in Two-Echelon Logistics Systems," *The Journal of the Operational Research Society, 49*:9 (1998), pp. 966-975.

Bertsekas, Dimitri P., *Dynamic Programming* (2nd ed.), Athena Scientific (1995).

Bertsekas, Dimitri P., and John Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific (1996).

Betts, Mitch, "Manage My Inventory Or Else," *Computerworld, 28*:5 (1994), pp. 93-95.

Cetinkaya, Sila, and Chung Y. Lee, "Stock Replenishment and Shipment Scheduling for Vendor-Managed Inventory Systems," *Management Science, 46*:2 (2000), pp. 217-232.

Chaouch, Benjamin A., "Stock Levels and Delivery Rates in Vendor-Managed Inventory Programs," *Production and Inventory Management, 10*:1 (2001), pp. 31-44.

Cheung, Ki L., and Hau Lee, "The Inventory Benefit of Shipment Coordination and Stock Rebalancing in a Supply Chain," *Management Science, 48*:2 (2002), pp. 300-306.

Clark, Theodore, and James L. McKenny, "Campbell Soup Company: A Leader in Continuous Replenishment Innovations," in *Harvard Business School Case*, Harvard Business School, Harvard University (1994).

Crites, Robert, and Andrew Barto, "Elevator Group Control Using Multiple Reinforcement Learning Agents," *Machine Learning*, *33*:2-3 (1998), pp. 235-262.

DeCroix, Gregory A., and Antonio Arreola-Risa, "Optimal Production and Inventory Policy for Multiple Products under Resource Constraints," *Management Science*, *44*:7 (1998), pp. 950-961.

Deuermeyer, Brian L., and Leroy Schwarz, "The 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 (1981).

Emigh, Jacqueline, "Vendor-Managed Inventory," *Computerworld, 33* (1999), pp. 52-55.

Evans, Richard V., "Inventory Control of a Multiproduct System with a Limited Production Resource," *Naval Research Logistics Quarterly*, *14*:2 (1967), pp. 173-184.

Fry, Michael J., *Collaborative and Cooperative Agreements in the Supply Chain*, unpublished Ph. D dissertation, University of Michigan (2002).

Fry, Michael J., Roman Kapuscinski, and Tava L. Olsen, "Coordinating Production and Delivery Under a (z, Z)-Type Vendor-Managed Inventory Contract," *Manufacturing & Service Operations Management, 3*:2 (2001), pp. 151-173.

Gavirneni, Srinagesh, Roman Kapuscinski, and Sridhar Tayur, "Value of Information in Capacitated Supply Chains," *Management Science, 45*:1 (1999), pp. 16-24.

Gendreau, Michel, Alain Hertz, and Gilbert Laporte, "A Tabu Search Heuristic for the Vehicle Routing Problem," *Management Science, 40*:10 (1994), pp. 1276-1290.

Gosavi, Abhijit, *Simulation-Based Optimization: Parametric Optimization Techniques and Reinforcement Learning*, Springer (2003).

Gosavi, Abhijit, "Adaptive Critics for Airline Revenue Management," in *Proceedings of the 18th Annual Conference of the Production and Operations Management Society* (2007).

Graves, Stephen C., "A Multi-Echelon Inventory Model With Fixed Reorder Intervals," *Technical Report*, Sloan School of Management, MIT, Cambridge (1996).

Hammond, Janice, "Barilla SpA (A) and (B)," In *Harvard Business School Case*. Cambridge, MA: Harvard Business School, Harvard University (1994).

Hibbard, Janet, "Supply-Side Economics," *Information Week, 707* (1998), pp. 85-87.

Higginson, James, and James Bookbinder, "Markovian Decision Processes in Shipment Consolidation," *Transportation Science*, *29*:3 (1995), pp. 242-255.

Jackson, Peter L., "Stock Allocation in a Two Echelon Distribution System or 'What to do Until Your Ship Comes In'," *Management Science, 34* (1988), pp. 880-895.

Jonsson, Horace, and Silver, Edward A., "Analysis of a Two-Echelon Inventory Control System with Complete Redistribution," *Management Science, 33* (1987), pp. 215-227.

Kanellos, Michael, "Intel to Manage PC Inventory," http://ww.cnet.com (1998).

Kurt Salmon Associates., *Survey of Supply Chain Effectiveness*, Food Distribution International (2002).

Nahmias, Steven, *Production and Operations Analysis* (4th ed.), McGraw-Hill (2001).

Nahmias, Steven, and Stephen A. Simth, "Mathematical Models of Retailer Inventory Systems: A Review," in R. Sarin (Ed.), *Perspectives on Operations Management* (pp. 249-278), Kluwer Academic Publishers (1993).

Ross, Sheldon M., *Introduction to Probability Models* (8th Ed.) Academic Press (2002).

Sethi, Suresh P., Han Yan, and Qin Zhang, "Peeling Layers of an Onion: Inventory Model with Multiple Delivery Modes and Forecast Updates," *Journal of Optimization Theory and Applications, 108*:2 (2001), pp. 253-281.

Simpson, Kenneth F., "A Theory of Allocation of Stocks to Warehouses," *Operations Research, 7*:6 (1959), pp. 797-805.

Subramaniam, Ganesh, and Abhijit Gosavi, "Simulation-Based Optimization for Material Dispatching in a Retailer Network," *International Journal of Simulation and Process Modeling*, *3*:4 (2007), pp. 238-245.

Suo, Hansheng, Jingchun Wang, and Yihui Jin., "Coordinating a Loss-Averse Newsvendor with Vendor Managed Inventory," in *Proceedings of the IEEE Conference on Systems, Man, and Cybernetics* (2004), pp. 6026-6030.

Survey, Distributor Customer Evaluation Survey on electronic components *Electronic Buyers News* (2003).

Sutton, Richard S., and Andrew Barto, *Reinforcement Learning: An Introduction*, MIT Press (1998).

Tesauro, Gerald, "Temporal Difference Learning and Td-gammon," *Communications of the Association for Computing Machinery, 38*:3 (1995), pp. 58-68.

Van Roy, Benjamin, Dimitri P. Bertsekas, Yuchus Lee, and John N. Tsitsiklis, "A Neuro-Dynamic Programming Approach to Retailer Inventory Management," *Proceedings of the 36th IEEE Conference on Decision and Control*, pp. 4052-4057 (1997).

Veinott, Arthur F., "Optimal Policies for a Multi-Product, Dynamic Non-Stationary Inventory Model with Several Demand Classes," *Operations Research*, *13*:5 (1965), pp. 761-778.

Waller, Matt, Eric Johnson, and Tom Davis, "Vendor-Managed Inventory in the Retailer Supply Chain," *Journal of Business Logistics, 20* (1999), pp. 183-203.

Watkins, Chris J., *Learning from Delayed Rewards*, unpublished Ph.D. thesis, Kings College, Cambridge, England (1989).

Werbos, Paul J., *Beyond Regression: New Tools for Prediction and Analysis of Behavioral Sciences*, Ph.D. thesis, Harvard University, Cambridge, MA (1974).

Wong, Wai-Keung, Jian Qi, and Sunney Leung, "Coordinating Supply Chains with Sales Rebate Contracts and Vendor-Managed Inventory," to appear in *International Journal of Production Economics* (2009).

Acknowledgements

The authors thank the two anonymous reviewers for numerous comments that improved the quality of the article and the special issue editor Dr. Susan Murray for her suggestions.

About the Authors

Zheng Sui holds a BS in plastic forming engineering, an MS in mechanical engineering from Shanghai Jiao Tong University, and a PhD in industrial engineering from University at Buffalo, SUNY. He currently serves as a product manager in Terra Technology, where he uses his knowledge of supply chain and optimization to design and develop supply chain software which has been used in some of the world's largest consumer-packaged goods companies.

Abhijit Gosavi has a BE and an M.Tech in mechanical engineering, and a PhD in industrial engineering from the University of South Florida. He is currently an assistant professor in the department of engineering management and systems engineering in Missouri University of Science and Technology. His research interests are in simulation, reinforcement learning, and manufacturing.

Li Lin is professor of industrial and systems engineering, University at Buffalo. His research interests include computer simulation, manufacturing and healthcare system analysis and design. He has published 60 refereed papers in major research journals. His current research focuses on the delivery of healthcare using system engineering methods. As a volunteer he serves on the Board of Directors of Catholic Health System in Buffalo, NY.

Contact: Dr. Abhijit Gosavi, Missouri University of Science and Technology, 210 Engineering Management, Rolla, MO 65409; phone: 573-341-4624; gosavia@mst.edu