
Manufacturing & Service Operations Management

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

# Price-Directed Replenishment of Subsets: Methodology and Its Application to Inventory Routing

Daniel Adelman,

# Price-Directed Replenishment of Subsets: Methodology and Its Application to Inventory Routing

Daniel Adelman
*The University of Chicago, Graduate School of Business, 1101 East 58th Street, Chicago, Illinois 60637*
*dan.adelman@gsb.uchicago.edu*

The idea of price-directed control is to use an operating policy that exploits optimal dual prices from a mathematical programming relaxation of the underlying control problem. We apply it to the problem of replenishing inventory to subsets of products/locations, such as in the distribution of industrial gases, so as to minimize long-run time average replenishment costs. Given a marginal value for each product/location, whenever there is a stockout the dispatcher compares the total value of each feasible replenishment with its cost, and chooses one that maximizes the surplus. We derive this operating policy using a linear functional approximation to the optimal value function of a semi-Markov decision process on continuous spaces. This approximation also leads to a math program whose optimal dual prices yield values and whose optimal objective value gives a lower bound on system performance. We use duality theory to show that optimal prices satisfy several structural properties and can be interpreted as estimates of lowest achievable marginal costs. On real-world instances, the price-directed policy achieves superior, near optimal performance as compared with other approaches.
(*Inventory Routing; Approximate Dynamic Programming; Price-Directed Operations; Semi-Markov Decision Processes*)

## 1. Introduction

We first give a generic description of the operational system under study and motivate the managerial issues using a numerical example. Then, we present an overview of our price-directed approach and give a literature review, before proceeding with our analysis.

### 1.1. System Description

A dispatcher continuously monitors and controls inventories for a set of items $\mathbf{I}$. An item may represent a product, a location, or a product-location pair. The inventory of each item $i \in \mathbf{I}$ is infinitely divisible, is consumed at a constant deterministic rate of $\lambda_i > 0$, and cannot exceed a maximum allowable inventory level of $0 < \overline{S}_i < \infty$. As inventories deplete, the dispatcher may in real time replenish a subset $I \subseteq \mathbf{I}$ of items, which incurs a cost of $0 < C_I < \infty$ and is completed instantaneously. Without loss of generality, we may assume $C_{I_1} \leq C_{I_2}$ $\forall I_1 \subseteq I_2$, because otherwise the dispatcher can replenish $I_1$ by executing $I_2$ without replenishing items $I_2 \backslash I_1$. Although we can accommodate different item sizes, we assume for simplicity that all demands and inventories are measured in the same units, e.g., liters, and that no more than $0 < \overline{D} < \infty$ total units can be replenished across all items in a single dispatch. The dispatcher's problem is to minimize the long-run time average replenishment cost, subject to allowing no stockouts.

Our numerical examples come from inventory routing problems faced by industrial gas distributors (Fisher et al. 1982, Bell et al. 1983, Dror and Ball 1987) such as Air Products and Chemicals, and Praxair. A

Manufacturing & Service Operations Management © 2003 INFORMS
Vol. 5, No. 4, Fall 2003, pp. 348–371
1523-4614/03/0504/0348
1526-5498 electronic ISSN

vendor distributes a single product to geographically dispersed customers $\mathbf{I}$ by dispatching delivery vehicles from a central depot. Each vehicle that is dispatched visits some subset of customers $I \subseteq \mathbf{I}$ before returning to the depot, incurring a cost $C_I$ that is based on an optimal traveling salesman tour on $I$. At most, $\overline{D}$ units of product can be carried on a vehicle. Each customer $i \in \mathbf{I}$ has a storage facility that can hold at most $\overline{S_i}$ units, and consumes inventory at a constant deterministic rate $\lambda_i$.

In these and other industrial contexts it is common for inventory holding costs to be either insignificant relative to transportation costs or absent altogether, for instance, because the vendor is paid upon delivery. Therefore, like the references above we do not consider the version of the problem with holding costs. However, in settings where holding costs are present, industrial managers can control them outside of our model by lowering the upper bounds $\overline{S_i}$ and instructing the dispatcher and delivery drivers to abide by them, rather than the physical tank capacities.

In general, items $\mathbf{I}$ may be products replenished from back to forward storage in a warehouse-picking operation, or products delivered to grocery store shelves, provided that infinite divisibility is a reasonable approximation. The replenishment costs $C_I$ may include any fixed costs for replenishment, with vehicle routing costs being just a special case.

### 1.2. Motivating Example

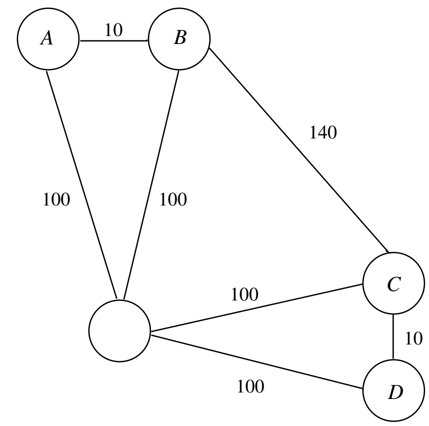
This example was first introduced by Bell et al. (1983) in the context of inventory routing.

Example 1. The replenishment costs are optimal traveling salesman costs on the network shown in Figure 1, starting and ending at the "depot" node. For example, $C_{AB} = 210$ miles. Set $\overline{D} = 5{,}000$ gallons, and the rest of the data are as follows:

| Item $i$ | $\overline{S_i}$ (gallons) | $\lambda_i$ (gallons per day) |
|---|---|---|
| $A$ | 5,000 | 1,000 |
| $B$ | 3,000 | 3,000 |
| $C$ | 2,000 | 2,000 |
| $D$ | 4,000 | 1,500 |

A simple schedule jointly replenishes customers $A$ and $B$ together, and customers $C$ and $D$ together, on a daily basis. This schedule is natural because $A$ and $B$ ($C$ and $D$, respectively) are near each other. Each customer $i$ receives a quantity equal to its daily consumption $\lambda_i$. The long-run time average cost of this schedule is 420 miles per day.

**Figure 1** **An Example with Four Items**




An improved schedule consists of a cycle that repeats every two days. On the first day, one trip is taken that replenishes 3,000 gallons to $B$ and 2,000 gallons to $C$, at a cost of 340 miles. On the second day, two trips are taken. The first trip replenishes 2,000 gallons to $A$ and 3,000 gallons to $B$. The second trip replenishes 2,000 gallons to $C$ and 3,000 gallons to $D$. Each trip costs 210 miles. The long-run time average cost of this improved schedule is 380 miles per day, which is nearly 10% lower than the first schedule.

Although Bell et al. (1983) described this improved schedule 20 years ago, to our knowledge no one yet knows how to either derive it or prove that it is optimal. In §7 we demonstrate that our methodology does both, for this example.

In the next section, we propose an operating policy that makes short-run decisions, from any starting inventory state. Consider the following scenario, which demonstrates the issues a real-world dispatcher faces daily. Suppose customer $A$ currently has 4,999 gallons in inventory, but customer $B$ is stocked out and must be replenished. Although it is only an extra 10 miles to also replenish customer $A$, it is doubtful that delivering one gallon is enough to justify it. What is the minimum delivery quantity that is economically viable? On the other hand, suppose customer $C$ has one gallon in inventory, so that 1,999 gallons can be delivered after $B$ is filled. Is it worth incurring an additional 140 miles to replenish 1,999 gallons to $C$? Our policy provides a simple resolution to such questions.

A fundamental question asked by managers in this setting, and for which to date no answer has been given, is how to estimate the marginal cost (per unit delivered) for replenishing a customer? With estimates of marginal costs in hand, a manager could compare them against marginal revenues to determine which customers are the least and most profitable. This information could be used, for example, to renegotiate existing service contracts, or to evaluate new ones the firm is considering, based on sales price and tank size.

Estimating marginal costs is difficult because in general the costs $C_I$ do not separate by customer, but are borne jointly by all customers $i \in I$. Customers have different consumption rates $\lambda_i$ and tank sizes $\overline{S_i}$, and therefore they are likely to be replenished using other dispatch subsets in addition to $I$. All of this must be taken into account. Intuitively, customers with small storage limits $\overline{S_i}$ who are geographically isolated and far from the depot should have the highest marginal costs. Our methodology yields estimates of (lowest achievable) marginal costs, and we show that they satisfy several intuitive properties.

### 1.3. Price-Directed Dispatching

In general, searching for an optimal cyclic schedule of replenishments, such as in Example 1, is an extremely difficult task. We do not know how many dispatches $N$ the schedule should contain and for any fixed $N$ there are $(2^{|\mathbf{I}|}-1)^N$ sequences of replenishment subsets to consider. It is even an open question whether for optimality it suffices to consider only cyclic schedules.

Although we formulate the control problem as a semi-Markov decision process, it is defined on continuous spaces with dimension equal to the number of items. Even if the continuous spaces are discretized, it cannot be solved in practice due to Bellman's "curse of dimensionality" (Bellman 1957). For example, in an industrial problem with 30 items each having an inventory state of "high" and "low", there would be $2^{30}$ (over one billion) possible states. It is unlikely that a simple structured policy exists that is provably optimal and computationally tractable for general problem data.

Instead, we study a control policy that is based on a simple economic mechanism for dispatching, which later we formally derive from the control formulation. Suppose the dispatcher receives from management a value, or transfer price, $V_i$ for replenishing one unit of item $i \in \mathbf{I}$. Whenever a stockout occurs, the dispatcher must choose a subset $I \subseteq \mathbf{I}$ of items to replenish, along with quantities $\vec{d} = \langle d_1, d_2, \ldots \rangle$ that satisfy the constraints on maximum inventory levels $\overline{S_i}$ $\forall i$ and the constraint on total replenishment quantity $\overline{D}$. Although he is responsible for paying the associated cost $C_I$, he also receives the total value $\sum_{i\in I} V_i d_i$ so that his net income equals the replenishment's net value

$$\sum_{i\in I} V_i d_i - C_I. \tag{1}$$

If this net value is positive or zero, then the dispatcher keeps the surplus and we call the replenishment *favorable*. On other hand, if quantity (1) is negative, then the dispatcher pays the shortfall and we call the replenishment *unfavorable*. The dispatcher chooses the replenishment $(I, \vec{d})$ that maximizes his net income (1).

This mechanism motivates the dispatcher to replenish an item $i$ whose current inventory level is low, because then $d_i$ can be set large relative to its upper bound $\overline{S_i}$. Furthermore, although we analyze a deterministic system, this policy can be implemented without modification in a more realistic setting with demand stochasticity, whereas a cyclic schedule cannot.

When faced with the option of expanding a replenishment of subset $I$ to include an item $i$ that is not currently stocked out, a real-world dispatcher will consider the incremental cost $C_{I\cup\{i\}} - C_I$ and determine if a quantity $d_i$ can be replenished that is large enough to justify it. We demonstrated this kind of thinking in our discussion of Example 1 above. Our economic mechanism provides a simple implementation of this logic. If the incremental net value, $d_i V_i - (C_{I\cup\{i\}} - C_I)$, is greater than or equal to 0, then the addition of $i$ to the replenishment is incrementally favorable. Isolating $d_i$ yields

$$d_i \geq \frac{C_{I\cup\{i\}} - C_I}{V_i} \quad \forall i \in \mathbf{I}, I \subseteq \mathbf{I}\backslash\{i\}, \tag{2}$$

where $i$ is some item with positive current inventory. Hence, the *minimum replenishment quantities* on the right-hand side of (2) are implicit in our economic mechanism.

In this paper, we focus on management's problem, which is to set the $V_i$'s so that the dispatcher is motivated to (ideally) minimize the long-run time average replenishment costs. If (1) is regularly positive, then the dispatcher's performance exceeds management's long-range expectations. Management should decrease the $V_i$'s to make them consistent with actual performance. On the other hand, if (1) is regularly negative, then the $V_i$'s impose unrealistic expectations on the dispatcher and management should increase them. Ideally, management should set the $V_i$'s equal to the lowest achievable marginal costs.

The primal-dual pair of math programs (NLP) and (D) of §4.1 approximates management's problem, and leads us to the following two-step procedure:

(1) Solve the primal-dual pair (NLP) and (D). Set the $V_i$'s to be used by the dispatcher equal to optimal dual prices $V_i^*$, and announce them once and for all.

(2) From now on whenever there are stockouts, the dispatcher chooses dispatches $(I, \vec{d})$ that maximize the net value (1).

In our implementation, we take as input a table of costs $C_I$ for all relevant $I \subseteq \mathbf{I}$. It is also possible to implement our approach by generating subsets $I$ as needed, for example, by constructing traveling salesman tours. We do not explore this here.

Although we seek to minimize long-run time average costs, just as in Markov decision processes (Puterman 1994) our policy can be implemented starting from any inventory state and will produce transient costs before reaching steady state. In this paper, we only measure steady-state cost rates in evaluating the performance of our policy, but we do not expect our policy to perform too poorly in the transient because it favors high value/low cost replenishments. The economic justification for discounting is stronger. However, if items are replenished daily or weekly rather than annually, then optimizing average cost simplifies the mathematics yet in steady state can produce nearly the same decisions for practical discount rates (e.g., see §3.7 of Zipkin 2000).

In this paper, we address four central questions:

(1) Where does this operating policy come from? (§6)

(2) How should management compute the $V$'s? (§4)

(3) What properties should the $V$'s satisfy? (§5)

(4) How well does this operating policy perform relative to (a bound on) optimal performance and other approaches? (§8)

We begin in §2 with an analysis of the single-item version of our problem, after which we outline the remainder of the paper.

### 1.4. Literature Review

Our idea of price-directed control is a new way to approach problems in this context. At present, there does not exist an algorithm in the literature that solves our problem to optimality. However, our math program to compute the $V_i$'s gives a lower bound on the optimal cost, and, as we discuss later, we believe that our approach may eventually lead to such an algorithm.

**1.4.1. Inventory Routing.** The paper by Bell et al. (1983), which won the TIMS (now INFORMS) Franz Edelman Award for Achievement in Operations Research and the Management Sciences, discusses the implementation of a large-scale integer program at Air Products and Chemicals to dispatch vehicles and inventory on a daily basis. The model's objective function maximizes the net value of replenishments, such as in (1), but over a discrete-time planning horizon. Their work demonstrates that the net-value concept is useful to industrial managers. In our Appendix, we show that although the objective function used by Dror and Ball (1987) is written in a different form, it can be transformed into one that maximizes net value. This result unifies these important papers.

Bell et al. (1983) do not address the question of how to compute the $V_i$'s. Dror and Ball (1987) suggest a heuristic approach to computing them, but say "It is possible that, through further investigation into our model, an accurate definition [of $V_i$] can be obtained" (Appendix II, p. 904). Our work fills this gap by addressing the methodological questions 1–4 stated at the end of §1.3. We in fact show that the net-value policy, the values $V_i$ themselves, and a lower bound on optimal cost, all *simultaneously* arise from the same underlying source. This puts the idea of maximizing net value on a solid, methodological footing. Furthermore, our computational results show that with intelligent values $V_i$, near optimal policy performance can be obtained by maximizing net value over just one or a few dispatches, rather than many hundreds as in Bell et al. (1983) and Dror and Ball (1987).

A thorough survey of the inventory routing literature has recently appeared in Campbell et al. (1998). Additional recent papers include Kleywegt et al. (2002), Reiman et al. (1999), and Berman and Larson (2001).

In an inventory control setting, Rosenblatt and Kaspi (1985) and Queyranne (1987) study a dynamic-programming algorithm that finds the best fixed partitioning policy, whereby, whenever an item is replenished all items in the same group are replenished. Bramel and Simchi-Levi (1995) study a general approach for finding fixed partitions, and apply it to a different inventory routing problem with holding costs instead of upper bounds $\overline{S}_i$. Chan et al. (1998) study the asymptotic behaviour of fixed partition policies in such settings, as do Anily and Federgruen (1990) for a more general policy based on partitioning ideas.

**1.4.2. Approximate Dynamic Programming.** We explicitly derive our math program by making a functional approximation to the value function of a dynamic program. Schweitzer and Seidmann (1985) do this for a generic discrete dynamic program. Recently, de Farias and Van Roy (2003) have considered this approach and implemented it in the setting of a queueing control problem. Adelman (2004) considers a problem in multi-item stochastic inventory/routing. Other than these papers and the present one, which also appears to be the first to consider a semi-Markov decision process, to date there is very little experience implementing the idea.

Functional approximations in dynamic programming were first considered by Bellman and Dreyfus (1959) and Bellman et al. (1973), but much of the literature that followed focused on discretizations of continuous state spaces and approximating functions of low dimension with as high accuracy as possible (Rust 1996). Recently, some authors have been developing statistical methods for adapting parameters in functional approximations (Bertsekas and Tsitsiklis 1996, Sutton and Barto 1998, Singh and Yee 1994) but without using dual price information.

Powell and Topaloglu (2003) and Powell and Carvalho (1998) develop methods in (adaptive) approximate dynamic programming for operational problems. They statistically adapt functional parameters using dual prices from subproblems solved through time. In contrast, we obtain these parameters as optimal prices from a math program solved a priori.

**1.4.3. Prices and Math Programming.** The idea of using prices to manage internal operations is old, having roots in the accounting and economics literature, e.g., see Hirshleifer (1956), Dopuch and Drake (1964), and Demski (1994), and in the early development of linear programming, see Dantzig (1963). However, in the operations research literature, prices have come to be used almost exclusively as an algorithmic device, e.g., for pricing out new columns during the solution of a linear program and in Lagrangian optimization algorithms.

There have been a few studies among disparate research communities that use prices from math programs in operating policies, but this has been done only as a heuristic without our dynamic-programming machinery. For instance, prices have been used in production scheduling (Roundy et al. 1991), the restless bandits problem (Bertsimas and Niño-Mora 2000), revenue management (Talluri and van Ryzin 1998, Simpson 1989, Williamson 1992), congestion-based vehicle dispatching (Gans and van Ryzin 1999), and multiclass queueing (Chen and Meyn 1999). Prices also have been used in remnant inventory management (Adelman et al. 1999, Adelman and Nemhauser 1999, Adelman 1997).

## 2. Single-Item Analysis

The single-item problem and our associated models all can be solved by inspection: the optimal policy replenishes $d^* = \min\{\overline{S}, \overline{D}\}$ whenever a stockout occurs. (We drop the subscripts $i$ and $I$ for convenience.) However, their simplicity will allow us to illustrate the major steps of the more complex multi-item analysis that begins in §3.

### 2.1. Formulation

We first formulate the problem using dynamic system equations. Let $n = 1, 2, \ldots \equiv \mathbb{Z}^+$ index the dispatches through time, and let $T_n$ denote the elapsed time between dispatches $n$ and $n+1$. Hence, assuming dispatch 1 occurs at time 0, then dispatch $N$ occurs at time $\sum_{n=1}^{N-1} T_n$. Let $s_n$ denote the inventory level (not necessarily 0) immediately before dispatch $n$, and let $d_n$ denote the replenishment quantity chosen. We assume the initial inventory level $s_1$ is fixed and given. We also assume that replenishments occur instantaneously, so that after dispatch $n$ the inventory immediately jumps to $s_n + d_n$.

Formally, the control problem can be expressed as

$$
\begin{aligned}
(\textit{control}) \quad \text{minimize} \quad & \limsup_{N\to\infty} \frac{NC}{\sum_{n=1}^{N} T_n} && (3)\\
& s_{n+1} = s_n + d_n - \lambda T_n \quad && \forall n \in \mathbb{Z}^+, \quad (4)\\
& d_n \le \overline{S} - s_n && \forall n \in \mathbb{Z}^+, \quad (5)\\
& d_n \le \overline{D} && \forall n \in \mathbb{Z}^+,\\
& d, s, T \ge 0.
\end{aligned}
$$

The objective function (3) minimizes the long-run time average replenishment cost. Constraints (4) maintain inventory flow balance, saying that the inventory immediately before dispatch $n+1$ equals the inventory after dispatch $n$ minus consumption since then. Constraints (5) require that inventory after replenishment does not exceed the storage limit $\overline{S}$.

Because we know that an optimal policy replenishes only when there is a stockout, we can now formulate the problem as a (trivial) semi-Markov decision process in which the state space consists of a single inventory state 0. The action space at each decision epoch, i.e., stockout time, is the set of replenishment quantities $d$ on the closed real interval $[0, \min\{\overline{S}, \overline{D}\}]$. The next stockout occurs after $d/\lambda$ time units. The optimality equation is

$$(dp) \quad h(0) = \inf_{0 \le d \le \min\{\overline{S}, \overline{D}\}} \{C - g(d/\lambda) + h(0)\},$$

where the decision variables are the optimal loss $g$ and the bias $h(0)$ of state 0. The optimality equation says that the bias of state 0 equals the total cost $C$ that will be incurred until the next stockout, minus the cost that would be accumulated over this time duration if it did so at constant rate $g$, plus the bias of the next state 0. For any arbitrary finite constant $\theta$, the solution is

$$g^* = \frac{C\lambda}{\min\{\overline{S}, \overline{D}\}}, \qquad h^*(0) = \theta, \tag{6}$$

so that $g^*$ equals the optimal objective value of (3).

### 2.2. Derivation of Math Programs

We now use the formulations above to derive a primal-dual pair of math programs. First, we work with (*control*) to derive the primal problem. Rewrite (4) as

$$s_{N+1} = s_1 + \sum_{n=1}^{N} d_n - \lambda \sum_{n=1}^{N} T_n,$$

divide both sides by $\sum_{n=1}^{N} T_n$, and take the limit as $N \to \infty$. This yields

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N} d_n}{\sum_{n=1}^{N} T_n} = \lambda,$$

because $0 \le s_{N+1} \le \overline{S} < \infty$. Now, multiply the top and bottom inside the limit by $N$, to obtain

$$
\begin{aligned}
\lambda &= \lim_{N\to\infty} \left[ \left( \frac{\sum_{n=1}^{N} d_n}{N} \right) \left( \frac{N}{\sum_{n=1}^{N} T_n} \right) \right]\\
&= \left[ \lim_{N\to\infty} \frac{\sum_{n=1}^{N} d_n}{N} \right] \left[ \lim_{N\to\infty} \frac{N}{\sum_{n=1}^{N} T_n} \right] = dZ,
\end{aligned}
$$

where

$$d = \lim_{N\to\infty} \frac{\sum_{n=1}^{N} d_n}{N} \quad \text{and} \quad Z = \lim_{N\to\infty} \frac{N}{\sum_{n=1}^{N} T_n}$$

are the long-run average replenishment quantity and replenishment frequency, respectively, assuming the limits exist. Hence, in the long-run averages, replenishment must equal consumption. Continuing in this manner, we can then derive the nonlinear program

$$(nlp) \quad \text{minimize } CZ \quad dZ=\lambda, \quad d\le \min\{\overline{S},\overline{D}\}, \quad d,Z\ge 0, \tag{7}$$

whose optimal solution is $d^*=\min\{\overline{S},\overline{D}\}$, $Z^*=\lambda/\min\{\overline{S},\overline{D}\}$.

Rather than formulate the dual by working with ($nlp$) directly, we instead show that it can be derived by first reformulating ($dp$) as the linear program

$$\text{Sup } g \quad g(d/\lambda)\le C \quad \forall 0\le d\le \min\{\overline{S},\overline{D}\},$$

with a single decision variable $g$ and an infinite number of constraints. In the the multi-item case, the bias function does not cancel out as it did here, and so the corresponding program has an infinite number of both variables and constraints. We now wish to make a change of variables from $g$, the long-run time average cost, to $V$, the marginal cost per unit replenished. We achieve this by substituting $g=\lambda V$ into the above program to obtain

$$(d) \quad \text{maximize } \lambda V \quad dV\le C \quad \forall 0\le d\le \min\{\overline{S},\overline{D}\}. \tag{8}$$

If we interpret $V$ as a transfer price received by the dispatcher for replenishing one unit, then this program maximizes the rate at which transfer revenue accumulates, subject to the constraint that the total transfer payment cannot exceed cost on any replenishment. In the multi-item case we will substitute $g=\sum_{i\in\mathbf{I}}\lambda_i V_i$.

As expected from (6), the optimal solution to ($d$) is indeed the optimal marginal cost $V^*=C/\min\{\overline{S},\overline{D}\}$, obtained by making (8) tight at the right-most extreme point of the constraint index set $\{d\colon 0\le d\le \min\{\overline{S},\overline{D}\}\}$. Hence, we could rewrite ($d$) as a finite linear program with a single constraint. Furthermore, $V^*$ is the optimal dual price on the primal constraint (7). The optimal objective values of the two programs ($nlp$) and ($d$) equal each other (there is no duality gap) and a form of complementary slackness is satisfied: $Z^*(d^*V^*-C)$.

### 2.3. Derivation of the Price-Directed Operating Policy

Heuristically, we could construct an operating policy that minimizes the reduced cost of the current action, as measured by the optimal dual solution of ($nlp$). However, we now see that this is equivalent to using the policy dictated by the right-hand side of the optimality equation ($dp$), except substituting $g=\lambda V^*$ where $V^*$ is the optimal dual price from the math program ($d$) above. In this manner, we obtain what we call the *price-directed operating policy* (in state 0):

$$(r) \quad \max_{0\le d\le \min\{\overline{D},\overline{S}\}}\{V^*d-C\},$$

which maximizes the net value of the replenishment.

In the single-item case, any choice of $V^*>0$ recovers the optimal policy, but using the optimal $V^*$ from ($d$) offers several advantages. First, $V^*$ is managerially meaningful as an estimate of marginal cost, in this case exact. Second, $\lambda V^*$ gives a (tight) lower bound on the optimal cost rate to which the performance of any policy, including the price-directed policy, can be compared. Third, our $V^*$ gives a certificate of policy optimality: any policy which chooses replenishments $d_n$ so that $V^*d_n-C=0$ for all $n$ is optimal. In the single-item case such an optimal policy always exists and is unique, although in general neither are guaranteed in the multi-item case. Finally, in the multi-item case the policy depends on the choice of $V_i^*$ for each item $i$, not just their sign, and so the extension of ($d$) provides a rigorous methodology for computing them.

### 2.4. The Linear Structure of the Bias Function

For strictly positive inventory states $s>0$, we can restrict the action space to the singleton $d=0$ so that the optimality equation becomes

$$h(s)=-g(s/\lambda)+h(0) \quad \forall 0\le s\le \overline{S},$$

where $s/\lambda$ is the time until stockout. Substituting $g=\lambda V$, and setting $h(0)=\theta$ for some constant $\theta$, we see that

$$h(s)=\theta-Vs \quad \forall 0\le s\le \overline{S}.$$

Hence, the bias is a linear function of the inventory level.

Similarly, in the multi-item case the bias function is linear along any trajectory having strictly positive inventories leading to a stockout, giving some rationale for making the linear functional approximation $h(s) \approx \theta - \sum_{i\in\mathbf{I}} V_i s_i$. This approximation yields models that are tractable in large scale, give good lower bounds, and produce a policy whose empirical performance is quite good. In this sense, the justification for our approach is similar in spirit to that of Lagrangian duality for solving integer programs (Fisher 1981). In addition, our methodology opens up the opportunity for even stronger bounds and policies, perhaps with increased computational expense, by simply replacing the linear approximation with a stronger functional form such as piecewise linear. If the terms in the form chosen constitute a basis for the space of functions that exactly solve the optimality equations, then our approach recovers an optimal policy. We leave this for future work.

### 2.5. Outline of the Multi-Item Analysis

Our analysis of the multi-item case mimics the single-item case, except now (*control*) becomes (CONTROL), (*dp*) becomes (DP), etc. We begin in §3 by formulating two dynamic models of the underlying control problem. The model (CONTROL) formally specifies the problem using dynamic system equations to track inventory trajectories. Lemmas 1 and 2 permit us to then reformulate (CONTROL) as a semi-Markov decision process (DP) in which decision epochs correspond with stockout times.

By taking limits in (CONTROL), in §4 we derive the nonlinear program (NLP). By employing linear functional approximations to the loss and bias functions in (DP), we derive the linear semi-infinite program (D). Theorems 1 and 2 show that (NLP) and (D), respectively, give a lower bound on the optimal cost rate. In Theorem 3 we establish that (NLP) and (D) are dual to each other. We show that the nonlinear program (NLP) is equivalent to a finite linear program (P) whose decision variables correspond with extreme points of the constraint index set of the dual program (D). This is analogous to our remarks above regarding extreme points in the constraint index set of (*d*). In practice, we solve (NLP) by instead solving (P) using column generation, as described in §4.4. Theorem 4 provides a powerful condition that can reduce problem sizes to a substantially smaller group of subsets $I \subseteq \mathbf{I}$ than the entire power set. This makes solving (NLP) practical for industrial size instances.

In §5, we use strong duality to prove a few economic properties satisfied by optimal prices. In §6, we derive our operating policy directly from the semi-Markov decision process using the loss and bias function approximations from before, and we formally state the price-directed policy as the optimization problem (R). In Theorem 5, we give a certificate of policy optimality, which says that if net value is zero on all dispatches, then the price-directed policy is optimal for (CONTROL). In §7, we use our methodology to solve Example 1 given above. Finally, in §8, we present results from implementing the price-directed dispatching policy on real-world instances, and compare against other policies in the literature.

## 3. The Multi-Item Formulation

### 3.1. System Dynamic Equations

Let $T_n$ $\forall n \in \mathbb{Z}^+ = \{1, 2, \ldots\}$ represent the elapsed time between dispatch $n$ and dispatch $n+1$. Let $s_{i,n}$ be the inventory of item $i \in \mathbf{I}$ immediately before the $n$th dispatch, which does not necessarily replenish $i$. The initial inventories are then fixed constants $s_{i,1}$ $\forall i \in \mathbf{I}$ that satisfy $0 \le s_{i,1} \le \overline{S}_i$. Also, let $Z_{I,n} = 1$ if items $I \subseteq \mathbf{I}$ but no others are replenished together during the $n$th dispatch, and zero otherwise. Let $d_{i,n}$ be the quantity replenished of item $i$ during the $n$th dispatch. For notational convenience, let $\vec{s}$ and $\vec{d}$ denote vectors in $\mathbb{R}^{|\mathbf{I}|}$ of inventory levels and replenishment quantities, respectively, which we subscript by $n$ to associate them with dispatch $n$.

For each $s_{i,1}$ $\forall i \in \mathbf{I}$ the problem we wish to solve is

$$\text{(CONTROL)} \quad \inf \limsup_{N\to\infty} \frac{\sum_{n=1}^{N} \sum_{I\subseteq\mathbf{I}} C_I Z_{I,n}}{\sum_{n=1}^{N} T_n} \tag{9}$$

$$s_{i,n+1} = s_{i,n} + d_{i,n} - \lambda_i T_n \quad \forall i \in \mathbf{I},\ n \in \mathbb{Z}^+, \tag{10}$$

$$d_{i,n} \le (\overline{S}_i - s_{i,n}) \sum_{\{I\subseteq\mathbf{I}: i\in I\}} Z_{I,n} \quad \forall i \in \mathbf{I},\ n \in \mathbb{Z}^+, \tag{11}$$

$$\sum_{i\in\mathbf{I}} d_{i,n} \le \overline{D} \quad \forall n \in \mathbb{Z}^+, \tag{12}$$

$$\sum_{I\subseteq\mathbf{I}} Z_{I,n} = 1 \quad \forall n \in \mathbb{Z}^+, \tag{13}$$

$$Z_{I,n} \in \{0,1\} \quad \forall I \subseteq \mathbf{I},\ n \in \mathbb{Z}^+, \tag{14}$$

$$d, s, T \ge 0. \tag{15}$$

Constraints (10) maintain inventory balance for each item $i$, saying that the inventory level before the $(n+1)$st dispatch equals the inventory level after the $n$th dispatch minus consumption since then. Constraints (13) ensure that exactly one dispatch is made at a time. Constraints (11) restrict replenishments to those that don't violate item storage capacities, and they also force $d_{i,n}=0$ if $i$ is not contained in the $n$th dispatch. Constraints (12) restrict the total quantity replenished across items, and can be extended to permit different item sizes. The objective (9) minimizes the long-run time average replenishment costs.

A feasible *solution* to (CONTROL) is an infinite sequence, or trajectory, of decision variables that satisfies the constraints. In contrast, a *policy* is a procedure for each $n$ that feasibly chooses $Z_{I,n+1}$ $\forall I \subseteq \mathbf{I}$ and $\vec{d}_{n+1}$ given the complete history up to dispatch $n+1$, i.e., the fixed values of $\vec{s}_{n+1}$ and all variables with subscripts $n$ or lower. However, the distinction for us is unimportant and we will speak loosely of solutions and policies interchangeably.

We do not address the existence of optimal solutions. Hassin and Megiddo (1991) prove existence for a specialized single product inventory model, but otherwise Federgruen and Zheng (1992) declare this as an open question for a broad class of inventory problems.

A direct replenishment policy, known as a direct shipment policy in the inventory routing literature, always induces a feasible solution to (CONTROL). Whenever an item $i$ stocks out we replenish the quantity $\min\{\overline{S}_i, \overline{D}\}$, at a cost of $C_{\{i\}}$, i.e., only one item is replenished on each dispatch. The objective value (9) then equals

$$\sum_{i\in\mathbf{I}} \frac{C_{\{i\}}}{\min\{\overline{S}_i, \overline{D}\}} \lambda_i < \infty. \tag{16}$$

When $\overline{D} < \overline{S}_i$, it is not necessary to wait until item $i$ stocks out, but doing so doesn't change the long-run time average cost (16).

In general, multiple items may be replenished in a single dispatch, but as in direct replenishment the objective value (9) can be no worse by waiting until a stockout occurs.

Property 1 (Just-in-Time Dispatching). $Z_{I,n}=1$ *for some* $I \subseteq \mathbf{I}$, $n \in \mathbb{Z}^+$ *implies* $\exists i \in I$ *such that* $s_{i,n}=0$.

An implication of Property 1 is that if $Z_{I,n+1}=1$, then $T_n = \min_{i\in\mathbf{I}}\{(s_{i,n}+d_{i,n})/\lambda_i\}$.

Lemma 1. *Any feasible solution to* (CONTROL) *that violates Property* 1 *can be transformed into one that satisfies it without increasing the objective value* (9).

Proof. Given a feasible solution to (CONTROL), suppose dispatch $\bar{n}$ violates Property 1 for some $\bar{I}$. Consider the corresponding right-continuous inventory process with left-hand limits:

$$Q_i(t) = s_{i,\bar{n}} - \lambda t + \sum_{n=\bar{n}}^{\bar{n}+N[0,t]-1} d_{i,n} \quad \forall i \in \mathbf{I},\ t \ge 0,$$

where $N[0,t] = 1 + \min\{m\colon \sum_{n'=\bar{n}}^{\bar{n}+m} T_{n'} > t\}$ counts the number of dispatches in the closed interval $[0,t]$, and time 0 corresponds to the time of dispatch $\bar{n}$. Let $\tau = \min\{t \ge 0\colon Q_i(t) = d_{i,\bar{n}}$ for some $i \in \bar{I}\}$, which is strictly positive because $s_{i,\bar{n}} > 0$ implies $Q_i(0) > d_{i,\bar{n}}$ $\forall i \in \bar{I}$.

Now, construct a new inventory process

$$Q_i'(t) = \begin{cases} Q_i(t) - d_{i,\bar{n}}, & 0 \le t \le \tau \\ Q_i(t), & t > \tau \end{cases} \quad \forall i \in \mathbf{I},$$

which postpones dispatch $\bar{n}$ until time $\tau$ but executes all intervening dispatches as planned. If $\tau = \infty$, then dispatch $\bar{n}$ is simply canceled. By definition of $\tau$ and because the original process $Q_i(t)$ is feasible, we have $0 \le Q_i'(t) \le \overline{S}_i$ $\forall i \in \mathbf{I}, t \ge 0$, and so $Q_i'(t)$ is feasible. □

### 3.2. Semi-Markov Decision Process

Because of Lemma 1 we can reformulate (CONTROL) as a semi-Markov decision process in which decision epochs correspond with stockout times. Define the state space to be the nonnegative $|\mathbf{I}|$-dimensional inventory space

$$\mathbf{S} \equiv \{\vec{s} \in \mathbb{R}_+^{|\mathbf{I}|}\colon \exists j \in \mathbf{I} \text{ such that } s_j = 0,\ s_i \le \overline{S}_i\ \forall i \in \mathbf{I}\setminus\{j\}\}.$$

For any $\vec{s}\in\mathbf{S}$ denoting current inventories, the action space consists of all pairs $(I,\vec{d})$ corresponding to feasible replenishments that follow Property 1, i.e.,

$$\mathbf{D}_{\vec{s}}\equiv\left\{\begin{array}{l}(I,\vec{d})\colon I\subseteq\mathbf{I},\ |I|>0,\ \vec{d}\in\mathbb{R}_{+}^{|\mathbf{I}|},\ s_i+d_i\leq\overline{S}_i\\ \quad\forall i\in I,\ \sum_{i\in I}d_i\leq\overline{D},\\ d_i=0\ \forall i\in\mathbf{I}\backslash I,\ \exists i\in I\text{ such that }s_i=0\end{array}\right\}\ \forall\vec{s}\in\mathbf{S}.$$

Using dispatch $(I,\vec{d})\in\mathbf{D}_{\vec{s}}$ when in state $\vec{s}\in\mathbf{S}$ incurs a cost $C_I$. The time until the next decision epoch is

$$\tau(\vec{s},(I,\vec{d}))=\min_{i\in\mathbf{I}}\left(\frac{s_i+d_i}{\lambda_i}\right)\quad\forall(\vec{s},(I,\vec{d}))\in\mathbf{S}\times\mathbf{D}_{\vec{s}},$$

which is equal to 0 if not all stocked-out items are replenished. The next inventory state after executing $(I,\vec{d})$ is

$$s_i\leftarrow s_i+d_i-\lambda_i\tau(\vec{s},(I,\vec{d}))\quad\forall i\in\mathbf{I}.$$

The objective is to find a policy that minimizes the "loss," which we define as the objective value (9). Because finite costs that accumulate over a finite time interval disappear under the limit in (9), a consequence of the next lemma is that the optimal loss does not depend on the initial inventories $s_{i,1}$ $\forall i\in\mathbf{I}$.

Lemma 2. *For every pair of inventory states $\vec{s}',\vec{s}''\in\mathbf{S}$, there exists a feasible sequence of replenishments that moves the system from $\vec{s}'$ to $\vec{s}''$ without violating Property* 1. *This sequence has a finite number of steps, finite time duration, and finite cost.*

Proof. We demonstrate a finite sequence of replenishments that lead to state $\vec{0}$, and then from $\vec{0}$ builds up inventories to reach $\vec{s}''$.

Whenever an item stocks out, execute $I\equiv\mathbf{I}$ with replenishment quantities

$$d_i=(\lambda_i\epsilon-s_i)^+\quad\forall i\in\mathbf{I},$$

where $s_i$ is the current inventory, $(\cdot)^+$ takes the positive part of the enclosed quantity, and $\epsilon>0$ is chosen small enough so that the replenishment is feasible. In particular, let $\mathbf{I}_0''=\{i\in\mathbf{I}\colon s_i''=0\}$ and $\widetilde{\mathbf{I}}_0''=\mathbf{I}\backslash\mathbf{I}_0''$. It suffices to choose an $\alpha$ such that $0<\alpha<\overline{D}$ and an $\epsilon>0$ such that

$$\epsilon\leq\min\left\{\min_{i\in\widetilde{\mathbf{I}}_0''}\frac{s_i''}{\lambda_i},\ \min_{i\in\mathbf{I}_0''}\frac{\overline{S}_i}{\lambda_i},\ \frac{\overline{D}-\alpha}{\sum_{i\in\mathbf{I}}\lambda_i}\right\},$$

the right-hand side of which is strictly positive because of the restrictions we impose on the input data.

Each item $i$ stocks out at most once before reaching inventory level $\lambda_i\epsilon$, and so $\vec{0}$ is reached within a time duration of length $\max_{i\in\mathbf{I}}s_i'/\lambda_i<\infty$. Within this time frame, there can only be a finite number of replenishments spaced $\epsilon$ time units apart, and so state $\vec{0}$ will be reached in a finite number of replenishments.

Now, let all items $i\in\mathbf{I}_0''$ stock out every $\epsilon$ time units, at which times we execute $I\equiv\mathbf{I}$ with replenishment quantities

$$d_i=\begin{cases}\lambda_i\epsilon & \forall i\in\mathbf{I}_0''\\ \min\{\lambda_i\epsilon+\alpha(s_i''/\sum_{i\in\mathbf{I}}s_i''),\ s_i''-s_i\} & \forall i\in\widetilde{\mathbf{I}}_0''\end{cases}\quad\forall i\in\mathbf{I}.$$

Again, by our choice of $\epsilon$ and $\alpha$ these replenishments are feasible. Because $\alpha>0$, the net inventory level of each item $i\in\widetilde{\mathbf{I}}_0''$ increases after each replenishment by $\alpha(s_i''/\sum_{i\in\mathbf{I}}s_i'')>0$ until $s_i''$ is reached. Therefore, the inventory state with $s_i=0$ $\forall i\in\mathbf{I}_0''$ and $s_i=s_i''-\lambda_i\epsilon$ $\forall i\in\widetilde{\mathbf{I}}_0''$ is reached in a finite number of steps. As a final step, execute $I\equiv\mathbf{I}$ with replenishment quantities

$$d_i=\begin{cases}0 & \forall i\in\mathbf{I}_0''\\ \lambda_i\epsilon & \forall i\in\widetilde{\mathbf{I}}_0''\end{cases}\quad\forall i\in\mathbf{I},$$

which leads into state $\vec{s}''$. □

Following Bhattacharya and Majumdar (1989) and Luque-Vásquez and Hernández-Lerma (1999), the optimality equations for this semi-Markov decision process are

$$\text{(DP)}\quad h(\vec{s})=\inf_{(I,\vec{d})\in\mathbf{D}_{\vec{s}}}\big\{C_I-g\tau(\vec{s},(I,\vec{d}))+h(\vec{s}+\vec{d}-\vec{\lambda}\tau(\vec{s},(I,\vec{d})))\big\}\quad\forall\vec{s}\in\mathbf{S},\quad(17)$$

where $\vec{\lambda}$ is the vector of consumption rates $\lambda_i$ $\forall i\in\mathbf{I}$. The constant $g$ is the optimal loss, and the function $h(\vec{s})$ is known as the optimal bias function and reflects transient costs starting from state $\vec{s}$.

Because we do not directly solve (DP), we do not address whether there exists a solution. However, one solution approach is to discretize the sets $\mathbf{S}$ and $\mathbf{D}_{\vec{s}}$, and thereby make them finite. It has been known for some time (Fox 1966, Jewell 1963) that (under mild conditions) there exists a solution to the resulting optimality equations that corresponds with a stationary deterministic policy. Because the discretized state and action spaces are finite, this corresponds with a deterministic cyclic schedule, which is a repeating sequence $\{\vec{s}_n, (I_n, \vec{d}_n)\}$ of $n=1,\ldots,N$ dispatches for some $N$.

For more on dynamic programming with average cost criterion, see Schäl (1992), Denardo and Fox (1968), Arapostathis et al. (1993), and Puterman (1994). The books by Dynkin and Yushkevich (1979), Hernández-Lerma (1989), and Hernández-Lerma and Lasserre (1996) treat discrete-time Markov decision processes on infinite state and action spaces.

## 4. Math-Programming Relaxation

### 4.1. Primal and Dual Models

Let $Z_I$ be a decision variable that represents the long-run time average rate that the subset $I$ of items is replenished together. For each $I$, let $d_{i,I}\ \forall i\in I$ be a decision variable that represents the average quantity of item $i$ replenished on dispatches to subset $I$. Consider the nonlinear program

$$\text{(NLP)}\quad \text{minimize} \sum_{I\subseteq \mathbf{I}} C_I Z_I \tag{18}$$

$$\sum_{I\subseteq \mathbf{I}} d_{i,I} Z_I = \lambda_i \quad \forall i\in \mathbf{I}, \tag{19}$$

$$\sum_{i\in I} d_{i,I} \le \overline{D} \quad \forall I\subseteq \mathbf{I}, \tag{20}$$

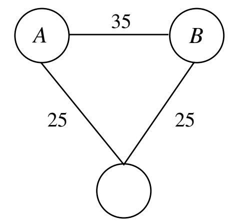
$$d_{i,I} \le \overline{S}_i \quad \forall I\subseteq \mathbf{I},\ i\in I, \tag{21}$$

$$Z, d \ge 0. \tag{22}$$

The objective (18) minimizes the long-run time average replenishment cost. Constraints (19) say that for each item $i$, the rate at which quantities are replenished must equal the rate at which they are consumed. Constraints (20) say that a dispatch can deliver (on average) at most $\overline{D}$ units across all items, and constraints (21) ensure that no more quantity is replenished (on average) of item $i$ than can be stored. The program (NLP) is a relaxation because it does not explicitly model inventory state dynamics.

Example 2 (Solutions to (NLP) may not be Implementable). Derive the replenishment costs from path

**Figure 2** **Arc Costs for Example 2**




costs in Figure 2, and set $\overline{D}=5$. The rest of the data are as follows:

| Item $i$ | $\overline{S}_i$ | $\lambda_i$ |
|---|---|---|
| $A$ | 2 | 1 |
| $B$ | 3 | 1 |

An optimal solution to (NLP) yields $Z^*_A=1/6$ with $d^*_{A,A}=2$, and $Z^*_{AB}=1/3$ with $d^*_{A,AB}=2$ and $d^*_{B,AB}=3$. Denote as time 0 the moment replenishment $A2B3$ occurs. At time 2 the replenishment $A2$ must be taken. At time 3 item $B$ stocks out but executing $A2B3$ violates $\overline{S}_A=2$.

After deriving (NLP) from limiting system dynamic equations, we use a functional approximation to the optimal loss and bias function in (DP) to derive the linear semi-infinite program

$$\text{(D)}\quad \text{maximize} \sum_{i\in \mathbf{I}} \lambda_i V_i \tag{23}$$

$$\sum_{i\in I} d_i V_i \le C_I \quad \forall (I,\vec{d})\in \mathbf{D}_{\bar{0}}, \tag{24}$$

having decision variables $V_i$. We show that it is dual to (NLP), in that there is no duality gap between (23) and (18) and a version of complementary slackness holds.

Whereas in (NLP) the $d_{i,I}$'s are decision variables, in (D) the $d_i$ are data derived from the set $\mathbf{D}_{\bar{0}}$. The decision variables $V_i$ at optimality are the marginal costs, or prices, associated with satisfying constraints (19) of (NLP). This means $\lambda_i V_i$ at optimality is the total allocated cost rate for replenishing item $i$ in an optimal solution to (NLP). Each $V_i$ can be interpreted as the payment management transfers to the dispatcher for replenishing one unit of item $i$. Hence, the objective (23) maximizes the total transfer rate, subject to constraints (24) that the payments can be no larger than the cost of any replenishment.

Another interpretation is that $V_i$ is the transfer price item $i$ gives to the dispatcher for his services. The dispatcher combines the $V_i$'s across items to pay for joint replenishment costs.

## 4.2. Derivations

**4.2.1. Limiting System Dynamics.** The following derivation of (NLP) from (CONTROL) leads to Theorem 1, which says that (18) at optimality is a lower bound on a restricted class of operating policies. We can extend this result to *all* operating policies by combining Theorems 2 and 3. We begin with a simple lemma.

Lemma 3. *Solutions to* (CONTROL) *that do not satisfy*

$$\lim_{N\to\infty}\sum_{n=1}^{N} T_n = \infty \tag{25}$$

*are suboptimal.*

Proof. If $\lim_{N\to\infty}\sum_{n=1}^{N} T_n < \infty$, then from (13) and $C_I > 0$ we have

$$\sum_{n=1}^{\infty}\sum_{I\subseteq \mathbf{I}} C_I Z_{I,n} = \infty.$$

The objective value (9) then equals $\infty$. This is dominated by the direct replenishment policy, which is always feasible and has objective value (16). $\square$

The proof depends on $C_{\{\varnothing\}} > 0$, but alternatively we could force $Z_{\{\varnothing\},n} = 0$ $\forall n$ to eliminate empty dispatches.

When they exist, define the limits

$$Z_I \equiv \lim_{N\to\infty} \frac{\sum_{n=1}^{N} Z_{I,n}}{\sum_{n=1}^{N} T_n} \quad \forall I\subseteq \mathbf{I}, \tag{26}$$

$$d_{i,I} \equiv \begin{cases} \lim_{N\to\infty} \dfrac{\sum_{n=1}^{N} d_{i,n} Z_{I,n}}{\sum_{n=1}^{N} Z_{I,n}} & \text{if } Z_I > 0 \\ 0 & \text{if } Z_I = 0 \end{cases} \quad \forall I\subseteq \mathbf{I},\ i\in I. \tag{27}$$

The simplest case in which these limits exist is when the underlying controlled process is regenerative (Sigman and Wolff 1993). Because there is no stochasticity in (CONTROL), regenerative policies are deterministic cyclic schedules.

Theorem 1. *Any feasible solution to* (CONTROL) *for which the limits* (26) *and* (27) *exist produces a feasible solution to* (NLP). *Therefore, the optimal objective value* (18) *of* (NLP) *is a lower bound on* (9) *when* (CONTROL) *is restricted to such solutions.*

Proof. For each $i\in\mathbf{I}$, $N\in\mathbb{Z}^+$, rewrite (10) as

$$s_{i,N+1} = s_{i,1} + \sum_{n=1}^{N} d_{i,n} - \lambda_i \sum_{n=1}^{N} T_n$$
$$\Longrightarrow s_{i,N+1} = s_{i,1} + \sum_{n=1}^{N}\sum_{\{I\subseteq\mathbf{I}:i\in I\}} d_{i,n}Z_{I,n} - \lambda_i\sum_{n=1}^{N} T_n.$$

Now divide both sides by $\sum_{n=1}^{N} T_n$ and take the limit as $N\to\infty$. Because we may assume (25) holds, and also $s_{i,N+1}, s_{i,1} \le \overline{S}_i < \infty$, we now have

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N}\sum_{\{I\subseteq\mathbf{I}:i\in I\}} d_{i,n}Z_{I,n}}{\sum_{n=1}^{N} T_n} = \lambda_i. \tag{28}$$

Multiply the top and bottom inside the limit by $\sum_{n=1}^{N} Z_{I,n}$, and pull out the summation $\sum_{\{I\subseteq\mathbf{I}:i\in I\}}$ in front of the limit to get

$$\sum_{\{I\subseteq\mathbf{I}:i\in I\}} \lim_{N\to\infty} \frac{\sum_{n=1}^{N} d_{i,n}Z_{I,n}}{\sum_{n=1}^{N} Z_{I,n}} \frac{\sum_{n=1}^{N} Z_{I,n}}{\sum_{n=1}^{N} T_n} = \lambda_i.$$

Now apply (26) and (27) to yield (19). Constraints (20), (21), and (22) follow trivially from (11), (12), and (14)–(15), respectively. The objective (18) follows directly by substituting (26) into (9). $\square$

**4.2.2. Dynamic Programming Functional Approximation.** To derive (D) from (DP), substitute

$$g = \sum_{i\in\mathbf{I}} \lambda_i V_i, \tag{29}$$

$$h(\vec{s}) \approx \theta - \sum_{i\in\mathbf{I}} s_i V_i \quad \forall \vec{s}\in\mathbf{S}, \tag{30}$$

for some constants $\theta$ and $V_i$ $\forall i\in\mathbf{I}$. Given a solution $(g, h(\cdot))$ to (DP), one can always find $V_i$ that allocate $g$ across items. Therefore, (30) is really the only approximation. Consider the infinite-dimensional linear program

$$\text{Sup } g \tag{31}$$

$$h(\vec{s}) - h(\vec{s} + \vec{d} - \vec{\lambda}\tau(\vec{s},(I,\vec{d}))) \le C_I - g\tau(\vec{s},(I,\vec{d})) \quad \forall \vec{s}\in\mathbf{S},\ (I,\vec{d})\in\mathbf{D}_{\vec{s}}. \tag{32}$$

Fox (1966) shows that in the finite case, this program is equivalent to solving (DP). In the infinite case, Hernández-Lerma and Lasserre (1996) give measurability conditions that are required for well posedness.

Make the substitutions (29) and (30), which restricts $(g, h(\cdot))$ onto the finite-dimensional space in which $(\theta, V)$ resides. The objective (31) becomes (23). The left-hand side of (32) becomes

$$\begin{aligned} &h(\vec{s}) - h(\vec{s}+\vec{d}-\vec{\lambda}\tau(\vec{s},(I,\vec{d}))) \\ &\quad = \left(\theta - \sum_{i\in\mathbf{I}} s_i V_i\right) - \left(\theta - \sum_{i\in\mathbf{I}}\left(s_i + d_i - \lambda_i \tau(\vec{s},(I,\vec{d}))\right)V_i\right) \\ &\quad = \sum_{i\in\mathbf{I}} d_i V_i - \sum_{i\in\mathbf{I}} \lambda_i V_i \tau(\vec{s},(I,\vec{d})) = \sum_{i\in\mathbf{I}} d_i V_i - g\tau(\vec{s},(I,\vec{d})), \end{aligned}$$

which when combined with the right-hand side of (32) yields

$$\sum_{i\in I} d_i V_i \le C_I \quad \forall \vec{s}\in\mathbf{S},\ (I,\vec{d})\in\mathbf{D}_{\vec{s}}.$$

This yields (24) because $\mathbf{D}_{\vec{0}} \supseteq \mathbf{D}_{\vec{s}}\ \forall \vec{s}\in\mathbf{S}$, and so constraints with $\vec{s}\neq\vec{0}$ are redundant.

Theorem 2. *The objective value* (23) *corresponding with any bounded feasible solution to* (*D*), *and in particular the optimal objective value, gives a lower bound on the cost rate* (9) *produced by any feasible solution to* (*CONTROL*).

Proof. In any feasible solution $(g, h(\cdot))$ to (32) for which

$$\sup_{\vec{s}\in\mathbf{S}} |h(\vec{s})| < \infty, \tag{33}$$

$g$ is a lower bound on (9). To see this, let $I_n$ denote the $n$th subset selected, so that for any $n = 1, 2, \ldots$

$$\begin{aligned} h(\vec{s}_{n+1}) &= h(\vec{s}_n + \vec{d}_n - \vec{\lambda}T_n) \\ &= C_{I_n} + h(\vec{s}_n + \vec{d}_n - \vec{\lambda}T_n) - C_{I_n} \\ &\ge h(\vec{s}_n) + gT_n - C_{I_n}, \end{aligned}$$

applying (32). Summing over $n = 1, \ldots, N$ yields

$$h(\vec{s}_{N+1}) - h(\vec{s}_1) + \sum_{n=1}^{N} C_{I_n} \ge g\sum_{n=1}^{N} T_n.$$

Because of Lemma 3 this becomes

$$\frac{h(\vec{s}_{N+1})}{\sum_{n=1}^N T_n} - \frac{h(\vec{s}_1)}{\sum_{n=1}^N T_n} + \frac{\sum_{n=1}^N C_{I_n}}{\sum_{n=1}^N T_n} \ge g,$$

and then (33) implies

$$\limsup_{N\to\infty} \frac{\sum_{n=1}^N C_{I_n}}{\sum_{n=1}^N T_n} \ge g.$$

An optimal solution to (D) yields bounded $V_i^*$ from Property 2 of §5 below, and $\theta$ can be chosen to be any finite constant. Therefore, $0 < \overline{S}_i < \infty\ \forall i \in \mathbf{I}$ implies $h(\vec{s})$ under (30) is bounded in the sense of (33). □

In fact, because the objective function (23) maximizes, (D) finds the largest lower bound obtainable under the approximation (29)–(30).

### 4.3. Duality

We demonstrate a dual relationship between (NLP) and (D) by first converting (D) into a finite linear program, and then showing that its dual is equivalent to (NLP).

Fix a set of $V_i$'s. Suppose there is a violated constraints (24), meaning that

$$\max_{(I,\vec{d})\in\mathbf{D}_0} \sum_{i\in I} V_i d_i - C_I \tag{34}$$

is positive. Then, there exists an optimal solution $(I^*, \vec{d}^*)$ to (34) for which $\vec{d}^*$ is an extreme point of the polytope $\mathbf{D}_{I^*}$, where we define

$$\begin{aligned} \mathbf{D}_I &\equiv \{\vec{d}\colon (I,\vec{d})\in\mathbf{D}_{\vec{0}}\} \\ &= \left\{\vec{d}\colon \sum_{i\in I} d_i \le \overline{D},\ \ 0\le d_i\le \overline{S}_i\ \ \forall i\in I,\ \ d_i = 0\ \ \forall i\in\mathbf{I}\backslash I\right\} \\ &\qquad\qquad \forall I\subseteq\mathbf{I}. \end{aligned}$$

Therefore, the infinite set of constraints (24) can be reduced to a finite set over

$$\overline{\mathbf{D}}_{\vec{0}} \equiv \{(I,\vec{d})\in\mathbf{D}_{\vec{0}}\colon \vec{d} \text{ is an extreme point of } \mathbf{D}_I\}.$$

The dual of the resulting linear program is

$$\text{(P)}\quad \text{minimize} \sum_{(I,\vec{d})\in\overline{\mathbf{D}}_{\vec{0}}} C_I Z_{I,\vec{d}} \tag{35}$$

$$\sum_{\{(I,\vec{d})\in\overline{\mathbf{D}}_{\vec{0}}:\, i\in I\}} d_i Z_{I,\vec{d}} = \lambda_i \quad \forall i\in\mathbf{I}, \tag{36}$$

$$Z_{I,\vec{d}} \ge 0 \quad \forall (I,\vec{d})\in\overline{\mathbf{D}}_{\vec{0}}. \tag{37}$$

Lemma 4. *Any feasible solution to* (*P*) *can be converted into a feasible solution to* (*NLP*) *having the same objective value, and vice versa.*

Proof. For each $I \subseteq \mathbf{I}$, denote the set of extreme points of $\mathbf{D}_I$ by $\overline{\mathbf{D}}_I$. Given a feasible solution to (36)–(37), construct a feasible solution to (NLP) by setting

$$Z_I = \sum_{\vec{d} \in \overline{\mathbf{D}}_I} Z_{I,\vec{d}} \quad \forall I \subseteq \mathbf{I} \tag{38}$$

and

$$d_{i,I} = \begin{cases} \sum_{\vec{d} \in \overline{\mathbf{D}}_I} d_i \left( \frac{Z_{I,\vec{d}}}{Z_I} \right) & \text{if } Z_I > 0 \\ 0 & \text{if } Z_I = 0 \end{cases} \quad \forall I \subseteq \mathbf{I},\ i \in I. \tag{39}$$

This choice is feasible to (20)–(22) because $\mathbf{D}_I$ $\forall I$ are convex sets. It also satisfies (19) because feasibility with respect to (36) is retained. The reverse transformation follows from Minkowski's Theorem, which uses the polyhedral convexity of $\mathbf{D}_I$ $\forall I$. □

This is enough to establish a dual relationship between (D) and (NLP).

Theorem 3. *There exists a pair of optimal solutions $V^*$, $(Z^*, d^*)$ to (D) and (NLP), respectively, for which*

$$\text{(no duality gap)} \quad \sum_{i \in \mathbf{I}} \lambda_i V_i^* = \sum_{I \subseteq \mathbf{I}} C_I Z_I^* \tag{40}$$

*and*

$$\text{(complementary slackness)} \quad Z_I^* \left( \sum_{i \in I} d_{i,I}^* V_i^* - C_I \right) = 0 \quad \forall I \subseteq \mathbf{I}. \tag{41}$$

Proof. The direct replenishment policy produces a feasible solution to (P) with objective value (16), and (35) is bounded below by 0. Hence, there exists an optimal solution $Z^*$ to (P), and by Lemma 4 it produces an optimal solution to (NLP). We can let $V^*$ be the corresponding optimal solution to the dual of (P), which also optimizes (D). From (38) and strong duality applied to (P) we obtain

$$\sum_{i \in \mathbf{I}} \lambda_i V_i^* = \sum_{(I,\vec{d}) \in \overline{\mathbf{D}}_{\vec{0}}} C_I Z_{I,\vec{d}}^* = \sum_{I \subseteq \mathbf{I}} C_I Z_I^*,$$

which is (40).

As in Lemma 4, for each $I \subseteq \mathbf{I}$, denote the set of extreme points of $\mathbf{D}_I$ by $\overline{\mathbf{D}}_I$. By complementary slackness applied to (P) we have

$$Z_{I,\vec{d}}^* \left( \sum_{i \in I} d_i V_i^* - C_I \right) = 0 \quad \forall (I, \vec{d}) \in \overline{\mathbf{D}}_{\vec{0}}.$$

When $Z_I^* > 0$ as defined by (38), we can manipulate this to achieve

$$0 = \left( \sum_{\vec{d} \in \overline{\mathbf{D}}_I} Z_{I,\vec{d}}^* \left( \sum_{i \in I} d_i V_i^* - C_I \right) \right) \frac{Z_I^*}{Z_I^*}$$

$$= Z_I^* \left( \sum_{\vec{d} \in \overline{\mathbf{D}}_I} \sum_{i \in I} d_i \left( \frac{Z_{I,\vec{d}}^*}{Z_I^*} \right) V_i^* - C_I \right) = Z_I^* \left( \sum_{i \in I} d_{i,I}^* V_i^* - C_I \right)$$

by applying (39). □

### 4.4. Solution

We solve (NLP) by instead solving (P) using column generation and then applying the mapping (38)–(39). This approach follows §4.3 of Lasdon (1970), which is based on Chapter 22 of Dantzig (1963).

The subproblems (34) are continuous knapsack problems when $I$ is fixed. These are trivially solvable using a greedy algorithm as follows. Enumerate the $i \in I$ in order of nonincreasing $V_i$ and set $d_i = \overline{S}_i$ for each until the total replenishment quantity violates $\overline{D}$ upon considering some $j \in I$. Give item $j$ the quantity $d_j = \overline{D} - \sum_{i \in I \setminus \{j\}} d_i$, and set all subsequent $d_i$'s to zero. For a fixed $I$, this algorithm yields an extreme point $\vec{d}$ of $\mathbf{D}_I$ whose components satisfy

$$d_i = \overline{S}_i \text{ or } 0 \qquad \text{for all } i \in I \tag{42}$$

except for at most one $i \in I$, call it $j$, in which case

$$d_j = \overline{D} - \sum_{i \in I \setminus \{j\}} d_i > 0. \tag{43}$$

If $\sum_{i \in I} V_i d_i - C_I$ is positive, then this new $Z_{I,\vec{d}}$ is entered into the formulation, and if it is not positive for any $(I, \vec{d})$, then we have an optimal solution to $(P)$, and consequently to (NLP).

In practice we can often dramatically reduce the number of subsets $I$ that need to be considered. If in (P) $Z_{I,\vec{d}} > 0$ but $d_i = 0$ for some $i \in I$, then there exists an alternative feasible solution that increases $Z_{I \setminus \{i\}, \vec{d}}$ by $Z_{I,\vec{d}}$ and sets $Z_{I,\vec{d}} = 0$. This alternative solution does not increase the objective (35) because $C_{I \setminus \{i\}} \le C_I$ by monotonicity. Therefore, we only need to consider $Z$'s in (P) that correspond to such *strictly positive* extreme points.

Theorem 4. *In an optimal solution to (NLP), we can set $Z_I^* = 0$ for all $I \subseteq \mathbf{I}$ such that*

$$\sum_{i\in I} \overline{S}_i - \max_{j\in I} \overline{S}_j \geq \overline{D}. \tag{44}$$

Proof. For any $I$ that satisfies (44), $\mathbf{D}_I$ does not have any strictly positive extreme points because

$$\overline{D} \leq \sum_{i\in I} \overline{S}_i - \max_{j\in I} \overline{S}_j \leq \sum_{i\in I\setminus\{k\}} \overline{S}_i$$

for all $k \in I$. □

In words, we need not consider subsets in which more than one item receives a partial replenishment.

## 5. Economic Properties

The optimal value $V_i^*$ of an item $i$ is bounded above by the marginal cost of the direct replenishment policy. It is bounded below by the lowest marginal cost of adding the item to a replenishment, which is nonnegative because $C_I$ is monotone nondecreasing in $I$.

Property 2 (Boundedness).

$$\frac{\min_{I\subseteq\mathbf{I}\setminus\{i\}} C_{I\cup\{i\}} - C_I}{\min\{\overline{D}, \overline{S}_i\}} \leq V_i^* \leq \frac{C_{\{i\}}}{\min\{\overline{D}, \overline{S}_i\}} \quad \forall i \in \mathbf{I}. \tag{45}$$

Proof. Fix $i$. The second inequality follows directly from (24). Because $\lambda_i > 0$, in any optimal solution to (*NLP*) $\exists J \subseteq \mathbf{I}\setminus\{i\}$ (possibly $J = \varnothing$) such that $d^*_{i,J\cup\{i\}} > 0$ and $Z^*_{J\cup\{i\}} > 0$. From complementary slackness,

$$\begin{aligned} C_{J\cup\{i\}} &= \sum_{j\in J\cup\{i\}} d^*_{j,J\cup\{i\}} V_j^* \\ &= d^*_{i,J\cup\{i\}} V_i^* + \sum_{j\in J} d^*_{j,J\cup\{i\}} V_j^* \\ &\leq d^*_{i,J\cup\{i\}} V_i^* + C_J \\ &\leq \min\{\overline{D}, \overline{S}_i\} V_i^* + C_J \end{aligned}$$

from primal feasibility (20), (21), and dual feasibility (24). Hence,

$$\frac{\min_{I\subseteq\mathbf{I}\setminus\{i\}} C_{I\cup\{i\}} - C_I}{\min\{\overline{D}, \overline{S}_i\}} \leq \frac{C_{J\cup\{i\}} - C_J}{\min\{\overline{D}, \overline{S}_i\}} \leq V_i^*. \quad \square$$

Although our price-directed policy uses a single set of optimal prices $V_i^*$, this set assigns zero reduced cost to all alternative optimal solutions of (NLP).

Property 3 (Completeness). *Let $V^*$ be an optimal solution to (D) for which there exists a corresponding solution to (NLP) as in Theorem 3. Any alternative optimal solution to (NLP) satisfies complementary slackness (41) with respect to this $V^*$.*

Proof. Suppose not. Then $\exists I$ such that $Z_I^* > 0$ and $C_I - d^*_{i,I} V_i^* > 0$. This and dual feasibility (24) imply

$$\begin{aligned} 0 &< \sum_{I\subseteq\mathbf{I}} Z_I^* \left( C_I - \sum_{i\in\mathbf{I}} d^*_{i,I} V_i^* \right) \\ &= \sum_{I\subseteq\mathbf{I}} C_I Z_I^* - \sum_{i\in\mathbf{I}} \sum_{\{I\subseteq\mathbf{I}:\ i\in I\}} Z_I^* d^*_{i,I} V_i^* \\ &= \sum_{I\subseteq\mathbf{I}} C_I Z_I^* - \sum_{i\in\mathbf{I}} \lambda_i V_i^*, \end{aligned}$$

where the last equality follows from primal feasibility (19). This contradicts the optimality of the alternative solution with respect to (NLP) because of (40). □

To illustrate the completeness property, consider the following simple example.

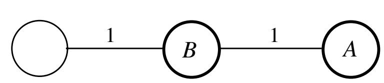
Example 3. Derive the replenishment costs from path costs in Figure 3, so that $C_A = 4$, $C_B = 2$, and $C_{AB} = 4$. Set $\overline{D} = 12$, and the rest of the data are as follows:

| Item $i$ | $\overline{S}_i$ | $\lambda_i$ |
|---|---|---|
| $A$ | 4 | 2 |
| $B$ | 8 | 2 |

What are the values $V_A^*$ and $V_B^*$? First, observe that one optimal policy fills $A$ and $B$ whenever $A$ stocks out, and has a cost rate of 2. The corresponding optimal solution to (NLP) is $Z^*_{AB} = 1/2$ and $d^*_{A,AB} = 4$, $d^*_{B,AB} = 4$. Although complementary slackness (41) implies that $4V_A^* + 4V_B^* = C_{AB}$, this is not enough to assign unique values to $V_A^*$ and $V_B^*$. However, the alternative optimal solution $Z^*_{AB} = 1/4$ with $d^*_{A,AB} = 4$, $d^*_{B,AB} = 8$ and $Z_A^* = 1/4$ with $d^*_{A,A} = 4$, corresponds with an alternative optimal operating policy. For this solution complementary slackness (41) implies the two equations

$$\begin{aligned} 4V_A^* &= 4, \\ 4V_A^* + 8V_B^* &= 4, \end{aligned}$$

**Figure 3** **An Example with Two Items**



which assigns the unique values $V_B^*=0$ and $V_A^*=1$. This same dual solution satisfies complementary slackness with respect to the first optimal solution to (NLP).

Item $B$ is an example of the most profitable type of item that the firm can add to its portfolio, because its marginal cost is 0. However, our price-directed policy requires a strictly positive value to provide an incentive to replenish. This can be achieved by assigning a small positive constant to $V_B^*$, or from the left-hand side of (45) it suffices to have $C_{I\cup\{i\}}>C_I\ \forall I\subseteq \mathbf{I}\backslash\{i\}$.

Optimal prices $V_i^*$ are not necessarily unique.

Example 4 (Optimal $V_i^*$ not Unique). Consider two identical items $A$ and $B$ with $\lambda_A=\lambda_B=1$, $\overline{S}_A=\overline{S}_B=2$, and arc costs as depicted in Figure 4. Set $\overline{D}=4$.

An optimal solution to (NLP) for this example is $Z_{AB}^*=0.5$ with $d_{A,AB}^*=d_{B,AB}^*=2$, which corresponds with a dual optimal solution $V_A^*=2$, $V_B^*=25$. Because items $A$ and $B$ are identical, by symmetry an alternative dual optimal solution is $V_A^*=25$, $V_B^*=2$. In fact, the entire continuum between these points, i.e., solutions to $V_A^*+V_B^*=27$ and $2\le V_A^*\le 25$, is an infinite set of alternative dual optima.

The firm may compare the values $V_i^*$ to ascertain which items are more profitable than others. Intuitively, items $i$ that are more expensive to replenish than others, and have smaller storage capacities, will require larger values $V_i^*$ to pay for replenishments. This is made precise by the next property. Interestingly, the demand rates $\lambda_i$ do not come into play.

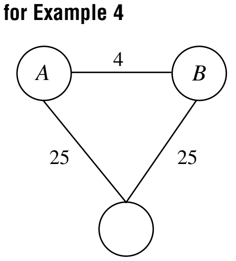
Property 4 (Comparability). *If $i_1,i_2\in\mathbf{I}$ satisfy*

$$\overline{S}_{i_1}\le\overline{S}_{i_2},$$

$$C_{I\cup\{i_1\}}\ge C_{I\cup\{i_2\}}\quad\forall I\subseteq\mathbf{I}\backslash\{i_1,i_2\},$$

*then $V_{i_1}^*\ge V_{i_2}^*$.*

**Figure 4** **Arc Costs for Example 4**




Proof. Because $\lambda_{i_1}>0$, in any optimal solution to (NLP) $\exists I$ with $i_1\notin I$ such that $d_{i_1,I\cup\{i_1\}}^*>0$ and $Z_{I\cup\{i_1\}}^*>0$. Therefore, complementary slackness (41) implies

$$\sum_{i\in I\cup\{i_1\}} d_{i,I\cup\{i_1\}}^* V_i^* = C_{I\cup\{i_1\}}\ge C_{I\cup\{i_2\}}$$
$$\ge \sum_{i\in I} d_{i,I\cup\{i_1\}}^* V_i^* + d_{i_1,I\cup\{i_1\}}^* V_{i_2}^*,$$

where the last inequality follows from dual feasibility (24). The $\vec{d}^*$ on the right-hand side is feasible for subset $I\cup\{i_2\}$, because $d_{i_1,I\cup\{i_1\}}^*\le\overline{S}_{i_1}\le\overline{S}_{i_2}$. Eliminating common terms on both sides yields

$$d_{i_1,I\cup\{i_1\}}^* V_{i_1}^*\ge d_{i_1,I\cup\{i_1\}}^* V_{i_2}^*,$$

and the result follows because $d_{i_1,I\cup\{i_1\}}^*>0$ can be divided on both sides. □

This property does not imply that the $V_i^*$ are independent of $\lambda_i$.

The next property shows how optimal solutions change when parameters are scaled. Consistent with Property 4, optimal prices $V^*$ decrease as storage capacity expands and increase as replenishment costs enlarge.

Property 5 (Homogeneity). *Let $(d^*,Z^*)$ and $V^*$ be an optimal primal-dual pair of solutions to (NLP) and (D), respectively. For any constants $\alpha,\beta,\gamma>0$, transform the problem data so that $\lambda_i\to\alpha\lambda_i\ \forall i\in\mathbf{I}$, $\overline{S}_i\to\beta\overline{S}_i\ \forall i\in\mathbf{I}$, $\overline{D}\to\beta\overline{D}$, and $C_I\to\gamma C_I\ \forall I\subseteq\mathbf{I}$. Then, $(\beta d^*,(\alpha/\beta)Z^*)$ and $(\gamma/\beta)V^*$ is an optimal primal-dual pair for the transformed problem.*

Proof. Primal feasibility is preserved because (19) can be rewritten as

$$\sum_{I\subseteq\mathbf{I}}\beta d_{i,I}^*(\alpha/\beta)Z_I^*=\alpha\lambda_i\quad\forall i\in\mathbf{I},$$

and both sides of (20) and (21) can be multiplied by $\beta$. Dual feasibility is preserved because (24) can be rewritten as

$$\sum_{i\in I}\beta d_i(\gamma/\beta)V_i^*\le\gamma C_I\quad\forall(I,\vec{d})\in\mathbf{D}_{\vec{0}}.$$

Finally, complementary slackness (41) is preserved because

$$(\alpha/\beta)Z_I^*\left(\sum_{i\in I}\beta d_{i,I}^*(\gamma/\beta)V_i^*-\gamma C_I\right)$$
$$=(\alpha\gamma/\beta)Z_I^*\left(\sum_{i\in I}d_{i,I}^*V_i^*-C_I\right)=0\quad\forall I\subseteq\mathbf{I}.\quad\square$$

## 6. The Price-Directed Operating Policy

To avoid unnecessary complication, we assume throughout this section that the replenishment cost function is *strictly* monotonic, meaning that $C_I < C_J$ $\forall I \subset J$. Property 2 implies then that $V_i^* > 0$ $\forall i \in \mathbf{I}$.

### 6.1. Derivation and Optimality

For any given loss $g$ and bias function $h(\cdot)$, when the system is in state $\vec{s} \in \mathbf{S}$ an operating policy is given by the right-hand side of (17). Bringing $h(\vec{s})$ over to the right-hand side, this policy chooses an action $(I, \vec{d})$ that solves

$$\inf_{(I,\vec{d}) \in \mathbf{D}_{\vec{s}}} \left\{ C_I - g\tau(\vec{s},(I,\vec{d})) + h(\vec{s}+\vec{d}-\vec{\lambda}\tau(\vec{s},(I,\vec{d}))) - h(\vec{s}) \right\} \quad \forall \vec{s} \in \mathbf{S}.$$

Under the approximation (29)–(30), and using optimal prices $V_i^*$ from (D), this equals

$$\begin{aligned}&\inf_{(I,\vec{d}) \in \mathbf{D}_{\vec{s}}} \{ C_I - \vec{\lambda} \cdot \vec{V}^* \tau(\vec{s},(I,\vec{d})) \\ &\qquad - (\vec{s}+\vec{d}-\vec{\lambda}\tau(\vec{s},(I,\vec{d}))) \cdot \vec{V}^* + \vec{s} \cdot \vec{V}^* \} \\ &= \inf_{(I,\vec{d}) \in \mathbf{D}_{\vec{s}}} \{ C_I - \vec{V}^* \cdot \vec{d} \},\end{aligned}$$

where $\cdot$ represents the vector dot product. This last expression becomes

$$\text{(R)} \qquad \max_{(I,\vec{d}) \in \mathbf{D}_{\vec{s}}} \left\{ \sum_{i \in I} V_i^* d_i - C_I \right\} \quad \forall \vec{s} \in \mathbf{S}$$

because $\mathbf{D}_{\vec{s}}$ is compact. The dispatcher waits until the moment some item stocks out, and then he solves (R) to identify a single dispatch to execute. For a fixed $I$, (R) is just a continuous knapsack problem solvable with a greedy algorithm, as in §4.4.

When (R) is too myopic, we can correct for inaccuracy in using the linear functional approximation, and thereby improve the performance of the policy, by maximizing the net value over the next $N$ dispatches through time. This *look-ahead* problem can be formulated as an integer program using constraints similar to those in (CONTROL). A heuristic for this problem enumerates the $\mathbf{M}$ highest net-value dispatches to do now, and then for each solves (R) repeatedly through simulated time to determine $N-1$ subsequent dispatches. From among these $\mathbf{M}$ prospective replenishment sequences, the heuristic implements the first dispatch of the one that maximizes the total net value over $N$ replenishments, rolls the horizon forward, and then repeats.

The price-directed policy seeks replenishments $(I, \vec{d})$ that have minimum reduced cost $C_I - \sum_{i \in I} V_i^* d_i$ with respect to (NLP). This leads to a simple sufficient condition for a sequence of replenishments to be optimal for (CONTROL).

Theorem 5. *If for some $M < \infty$ a feasible solution to (CONTROL) satisfies* (25) *and*

$$\sum_{i \in I_n} V_i^* d_{i,n} - C_{I_n} = 0 \quad \forall n \geq M, \tag{46}$$

*then it is optimal.*

Proof. From Property 2, $0 < \bar{D} < \infty$, and dual feasibility (24), it follows that

$$\left| \sum_{n=1}^{M-1} \left( C_{I_n} - \sum_{i \in \mathbf{I}} V_i^* d_{i,n} \right) \right| < \infty.$$

Hence,

$$\begin{aligned}&\limsup_{N \to \infty} \frac{\sum_{n=1}^{N} (C_{I_n} - \sum_{i \in \mathbf{I}} V_i^* d_{i,n})}{\sum_{n=1}^{N} T_n} \\ &= \limsup_{N \to \infty} \left( \frac{\sum_{n=1}^{M-1} (C_{I_n} - \sum_{i \in \mathbf{I}} V_i^* d_{i,n})}{\sum_{n=1}^{N} T_n} \right. \\ &\qquad \left. + \frac{\sum_{n=M}^{N} (C_{I_n} - \sum_{i \in \mathbf{I}} V_i^* d_{i,n})}{\sum_{n=1}^{N} T_n} \right) \\ &= \limsup_{N \to \infty} \frac{\sum_{n=1}^{M-1} (C_{I_n} - \sum_{i \in \mathbf{I}} V_i^* d_{i,n})}{\sum_{n=1}^{N} T_n} = 0.\end{aligned}$$

Because all the limits exist, the lim sup can be changed to lim. Now, (28) implies that

$$\lim_{N \to \infty} \frac{\sum_{n=1}^{N} C_{I_n}}{\sum_{n=1}^{N} T_n} = \sum_{i \in \mathbf{I}} V_i^* \lambda_i.$$

The result follows from Theorem 2 because $\sum_{i \in \mathbf{I}} V_i^* \lambda_i$ is a lower bound for all feasible solutions. $\square$

The proof shows that the problem of minimizing the long-run time average reduced cost is equivalent to minimizing our original objective function (9).

Because (NLP) is a relaxation, the condition in Theorem 5 is sufficient but not necessary.

This leads to a simple sufficient condition for direct replenishment to be optimal, assuming $C_I$ is monotonic.

Proposition 1. *If* $\overline{D} \le \overline{S}_i \ \forall i \in \mathbf{I}$, *then direct replenishment is an optimal policy with cost* (16).

Proof. From Theorem 4, in an optimal solution to (NLP), $Z_I^* = 0$ for all $I \subseteq \mathbf{I}$ having $|I| > 1$. The only remaining feasible solution to (NLP) corresponds with the direct replenishment policy, which is feasible for (CONTROL). $\square$

When this condition is satisfied, the optimal solution to $(D)$ is

$$V_i^* = \frac{C_{\{i\}}}{\overline{D}} \quad \forall i \in \mathbf{I}, \tag{47}$$

and direct replenishment will produce a sequence of replenishments that satisfies (46).

The conditions of Theorem 5 can hold beyond the special case in which direct replenishment is optimal, for example, they hold whenever an optimal solution to (NLP) uses multiple subsets $I$ but each item $i$ is replenished by only one subset (i.e., the solution forms a fixed partition). They also can hold in more complex settings, as the next section illustrates.

## 7. Solution to Example 1

It is a simple matter to prove that the schedule in §1.2, which is repeated from Bell et al. (1983), is in fact an optimal policy. An optimal solution to (D) in this case is $V_A^* = 0.005$, $V_B^* = 0.06\bar{6}$, $V_C^* = 0.07$, $V_D^* = 0.02\bar{3}$. The schedule itself maps directly into an optimal solution to (NLP), i.e., $Z_{BC}^* = Z_{AB}^* = Z_{CD}^* = 1/2$ and the $d^*$'s chosen accordingly. From complementary slackness (41), each of the three trips in the schedule has zero reduced cost, i.e.,

$$3{,}000V_B^* + 2{,}000V_C^* = 340,$$
$$2{,}000V_A^* + 3{,}000V_B^* = 210,$$
$$2{,}000V_C^* + 3{,}000V_D^* = 210,$$

and so the conditions of Theorem 5 are satisfied.

With the fourth equation

$$3{,}000V_B^* = 200,$$

the solution $V_i^*$ is uniquely specified. This equation arises because the optimal primal solution is degenerate, with $Z_B^* = 0$ and $d_{B,B}^* = 3{,}000$ (actually, the corresponding basic optimal solution to (P) is degenerate). Nevertheless, these four equations correspond with the only dispatches having zero reduced cost, and so from Property 3, this $V_i^*$ is the unique optimal dual solution.

Consider again the short-term decisions discussed in §1.2. Suppose $B$ is stocked out, but $A$ and $C$ have positive inventory, i.e., $s_B = 0$, $s_A > 0$, and $s_C > 0$. We must replenish customer $B$, but what are the minimum delivery quantities to $A$ and $C$ that make including them in the dispatch to $B$ profitable? With the $V_i^*$'s now in hand, (2) tells us that we must deliver at least 2,000 gallons to customer $A$ or $C$ for dispatches $AB$ or $BC$, respectively, to be incrementally profitable over dispatching to $B$ alone. This implies that we must have $s_A \le 3{,}000$ and $s_C = 0$, respectively. So, delivering one gallon to $A$ is too little to justify the extra 10 miles, and we must be able to deliver a full storage tank to customer $C$ to justify the extra 140 miles.

We can also order the customers from highest to lowest marginal cost: $C$, $B$, $D$, $A$. This ordering is not immediately obvious, but Property 4 proves useful. With respect to dispatching costs $C_I$, customers $B$ and $C$ are symmetrical, as are customers $A$ and $D$. However, $C$ has a smaller storage tank than $B$, and so Property 4 tells us $V_C^* \ge V_B^*$. Similarly, D has a smaller storage tank than $A$, and so $V_D^* \ge V_A^*$. Because of the higher dispatching cost $C_{BC} = 340$, we would expect the marginal costs for $B$ and $C$ to dominate $A$ and $D$.

## 8. Computational Study

On a diverse set of real-world instances of inventory routing problems, we compare our price-directed policy against several others and the lower bound given by (NLP).

### 8.1. Comparisons

Our first comparison is with direct shipment, which is a standard baseline in the inventory routing literature due to its simplicitly, although it ignores the benefits of consolidating deliveries for multiple customers onto a single vehicle. Direct shipment is a special case of a more powerful class of policies based on the notion of a fixed partition that captures consolidation benefits. Let $g_I$ be the cost rate for jointly replenishing all items in subset $I$, i.e., whenever an item $i \in I$ is replenished all items in $I$ are replenished. It is easy to see that

$$g_I = C_I \Big/ \min\left\{\min_{i\in I} \frac{\overline{S}_i}{\lambda_i}, \frac{\overline{D}}{\sum_{i\in I}\lambda_i}\right\}.$$

Therefore, to construct an optimal fixed partition policy we

(FP) find an integer $M$ and a partition $\{I_1, I_2, \ldots, I_M\}$ of $\mathbf{I}$ that minimizes $\sum_{m=1}^{M} g_{I_m}$.

To our knowledge, the fixed partition policy does not appear in the literature for our setting with explicit storage limits instead of holding costs.

Unfortunately, finding the optimal partition requires consideration of the entire power set of $\mathbf{I}$, and therefore quickly becomes intractable as the number of items grows. The best heuristic algorithm we have tested is due to Bramel and Simchi-Levi (1995), given in their §4.2.[1] Their basic approach can be applied in any setting where a fixed partition is sought and a subset function $g_I$ can be defined. For example, they apply their algorithm to a different inventory routing problem with holding costs instead of storage limits, by using a different choice of $g_I$'s.

Compared with our price-directed policy, fixed partition policies are still relatively inflexible because customers can only receive deliveries in one way, i.e., always with their fixed subset and in fixed quantities. So, we also compare against an adaptation of Dror and Ball (1987), which shares our subset flexibility and was developed for the same problem as ours (at least in the case of deterministic demand). However, in light of our result in the Appendix, both Dror and Ball (1987) and Bell et al. (1983) are (different) discrete-time extensions of (R) over a planning horizon. Our look-ahead version of the price-directed policy is a heuristic for maximizing the total net value over a continuous-time planning horizon. Consequently, the essential difference between our price-directed policy and Dror and Ball (1987) is how the $V_i$'s are computed, whereas Bell et al. (1983) do not indicate how to do so.

We ran two experiments to evaluate different approaches for setting the $V_i$'s in (R). The first uses direct-shipment prices (47), which generalizes to

$$V_i = \frac{C_{\{i\}}}{\min\{\overline{D}, \overline{S}_i\}} \quad \forall i \in \mathbf{I}$$

and is trivial to compute. The second computes the $V_i$'s by adapting the method proposed in Appendix II of Dror and Ball (1987), as explained in our Appendix. We complement this with a third experiment that adds a constraint used in Dror and Ball (1987): replenishments must always fill up customer tanks (or deliver a full truckload), i.e., partial replenishments are not permitted.

We ran a separate experiment to test the impact of a longer planning horizon. We compare the performance of the price-directed policy when it is implemented myopically using (R), i.e., optimizing over one dispatch at a time, versus using the look-ahead procedure discussed in §6.1. We look ahead up to four dispatches from each of the top $\mathbf{M} = 50$ dispatches in terms of net value.

### 8.2. Real-World Instances

For each instance shown in Table 1, the number in the name of the instance corresponds with the number of items (customers). The memphis instances come from Praxair, Inc., and are available online at www.tli.gatech.edu/research/casestudy/irp2/irp.cfm. The datasets estimate the demand rates $\lambda_i$ from (stochastic) consumption patterns observed in practice. Figure 5 shows a typical distribution of storage tank sizes at customer locations, relative to truck capacity, based on data from cleveland48, which is also derived from Praxair data posted online. Notice that usually one to four customers can be packed into a truck, which is in fact observed in actual routes used in practice. Consequently, in real-world instances the condition in Theorem 4 for collapsing the power set is extremely powerful, as depicted in Table 2.

[1] We obtain somewhat better performance when their argmax is replaced with an argmin, and so we report the latter.

**Table 1 Performance Guarantees for Various Policies on Real-World Instances**

| | Price-Directed | | Fixed Partition | | Alternative Pricing | | | |
|---|---|---|---|---|---|---|---|---|
| Instance | Look Ahead | Myopic ($R$) | ($FP$) Heuristic | Direct Ship | DB ('87) Fill | DB ('87) ($R$) | Direct Ship | Absolute LB |
| memphis25s | 1.003 | 1.011 | 1.035 | 1.077 | 1.041 | 1.129 | 1.033 | 35.103 |
| memphis25c | 1.117 | 1.270 | 1.152 | 1.463 | 1.122 | 1.120 | 1.201 | 17.653 |
| memphis35 | 1.006 | 1.012 | 1.032 | 1.109 | 1.046 | 1.035 | 1.063 | 37.117 |
| memphis49 | 1.010 | 1.016 | 1.042 | 1.148 | 1.055 | 1.045 | 1.080 | 41.931 |
| memphis50c | 1.033 | 1.033 | 1.063 | 1.205 | 1.064 | 1.081 | 1.097 | 101.065 |
| memphis50s | 1.005 | 1.008 | 1.025 | 1.087 | 1.032 | 1.026 | 1.047 | 71.429 |
| memphis62 | 1.028 | 1.028 | 1.063 | 1.208 | 1.067 | 1.054 | 1.098 | 102.404 |
| cleveland48 | 1.011 | 1.012 | 1.051 | 1.326 | 1.060 | 1.046 | 1.153 | 83.111 |
| fisher71 | 1.022 | 1.044 | 1.171 | 1.407 | 1.133 | 1.150 | 1.193 | 34.578 |
| newengland125 | 1.016 | 1.016 | 1.097 | 1.601 | 1.104 | 1.135 | 1.267 | 14.171 |
| fisher134 | 1.007 | 1.007 | 1.112 | 1.372 | 1.080 | 1.080 | 1.139 | 145.895 |

Figure 6 depicts the maximum days until stockout after a customer is fully replenished, which for each customer $i$ equals $\min\{\overline{S}_i, \overline{D}\}/\lambda_i$. (Customer numbers do not refer to the same customer shown in Figure 5.) What is important to note here is the sheer heterogeneity of the customer population, with many customers requiring daily replenishments, yet others requiring only monthly or even quarterly replenishments.

The instance newengland125 is based on real-world geographic data from a distribution operation in a large New England city (obtained courtesy of Professor Linus Schrage). The instances fisher71 and fisher134 use the real-world geographic data reported in Problems 11 and 12 of Fisher (1994). All three of these instances use tank sizes and consumption rates that were randomly generated based on the empirical distributions shown in Figures 5 and 6.

**Figure 5 Storage Tank Sizes for Instance cleveland48—Truck Capacity Equals 660**

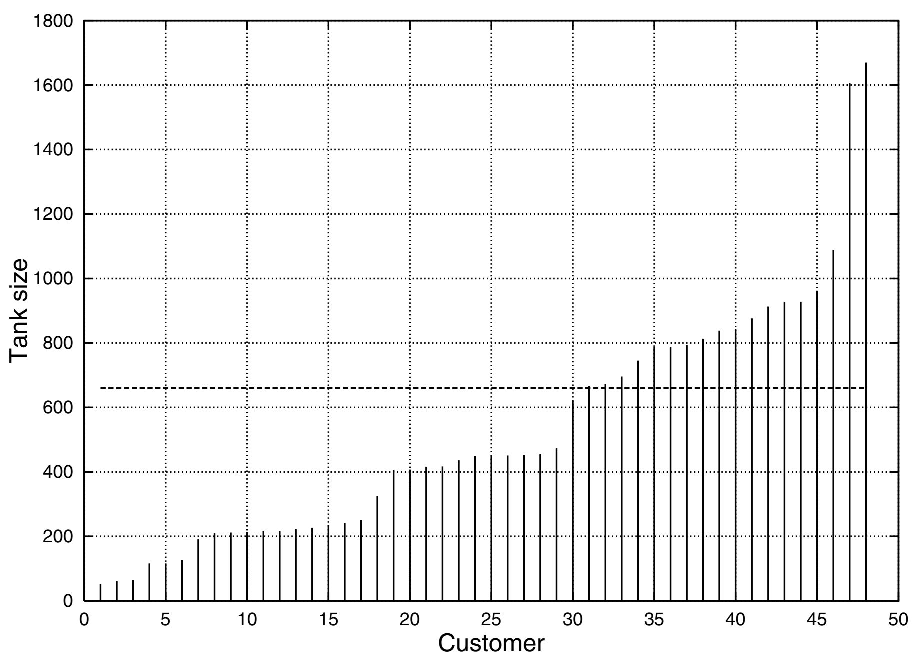


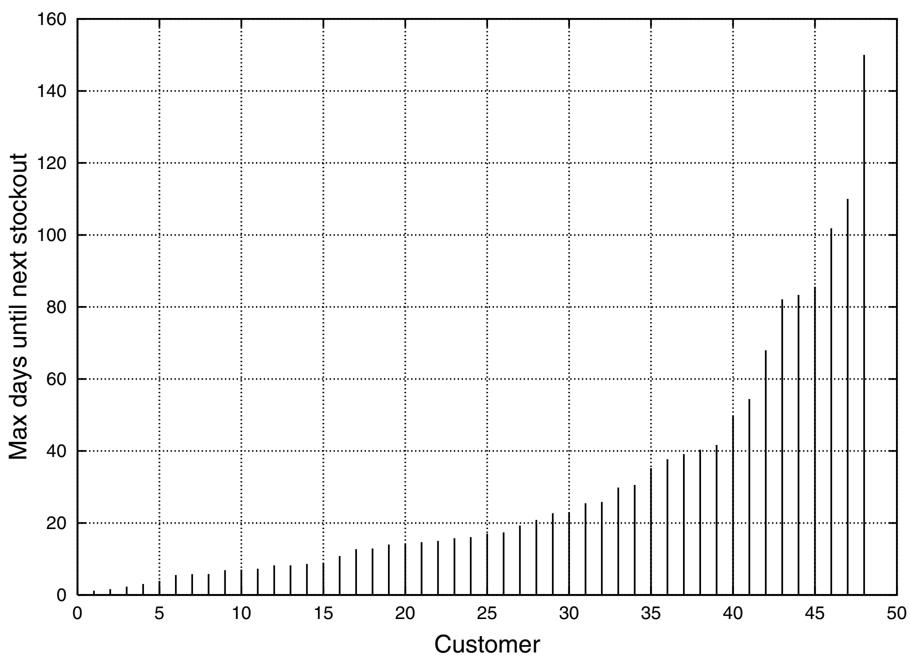
**Table 2 Reduction in the Number of Eligible Subsets Due to Theorem 4, and Corresponding Computational Effort to Solve (NLP)**

| Instance | # Eligible Subsets | # Subsets Total | # Columns | TSP CPU (sec) | (NLP) CPU (sec) |
|---|---|---|---|---|---|
| memphis25s | 1,110 | 33.5 million ($2^{25}$) | 88 | 0.11 | 0.01 |
| memphis25c | 78,391 | 33.5 million | 206 | 36.7 | 0.50 |
| memphis35 | 14,215 | 34.4 billion | 143 | 3.1 | 0.06 |
| memphis49 | 132,548 | $5.6 \times 10^{14}$ | 232 | 67.13 | 0.44 |
| memphis50c | 277,175 | $1.1 \times 10^{15}$ | 240 | 131.19 | 1.58 |
| memphis50s | 136,405 | $1.1 \times 10^{15}$ | 233 | 68.54 | 0.44 |
| memphis62 | 2,092,842 | $4.6 \times 10^{18}$ | 351 | 2,084.32 | 13.53 |
| cleveland48 | 80,330 | $2.8 \times 10^{14}$ | 352 | 26.93 | 0.41 |
| fisher71 | 227,432 | $2.4 \times 10^{21}$ | 412 | 40.84 | 0.76 |
| newengland125 | 9,876,783 | $4.3 \times 10^{37}$ | 1,086 | 4,851.94 | 51.71 |
| fisher134 | 15,232,191 | $2.2 \times 10^{40}$ | 1,040 | 5,913.27 | 78.25 |

### 8.3. Experimental Design

For each instance, we first solved (NLP) using column generation as described in §4.4, and thereby obtained optimal prices $V_i^*$. We used simulation to evaluate all policies other than the fixed partition policies, which can be evaluated using the $g_I$ from above. Starting from a random initial inventory state vector $\vec{s}_1$, we simulated each policy through time until 10,000 dispatches were performed. We excluded subsets that satisfied the condition of Theorem 4. Excluding the initial 100 dispatches as "initialization bias," we computed the time-average cost by dividing the total cost accumulated by the total simulated time that transpired. To obtain a performance guarantee relative to an optimal policy, we then computed the *relative gap* by dividing this time-average cost by the lower bound (18). We use the same lower bound in all of our comparisons with other policies. The upper limit of the interval of convergence in the last 500 dispatches was no more than 0.003 above the relative gaps reported, and was typically less than 0.001.

**Figure 6 Maximum Days Until the Next Stockout for Instance cleveland48, Equal to** $\min\{\overline{S}_i, \overline{D}\}/\lambda_i$



### 8.4. Results

Our results in Table 1 show that the price-directed policy is guaranteed to be near optimal on these instances and had superior performance compared with all other policies.

There is substantial improvement over an optimal direct shipment policy. We find that the myopic policies based on Dror and Ball (1987) perform about as well as the fixed partition policy. The policy reported under column "Fill" disallows partial replenishments. The policy using direct shipment prices can be viewed as a policy improvement step over the direct shipment policy.

In our experiments, we generally have observed that the performance gap between the best policies and (NLP) increases as $\overline{D}$ becomes relatively large compared with the $\overline{S}_i$'s. The memphis25c instance is such an example. Intuitively, as $\overline{D}$ increases, the optimal number of customers per route increases, requiring tighter coordination between their inventories and making the linear functional approximation to the bias less accurate. It is these kind of instances, in fact, that benefited the most from look ahead, which otherwise had little or no effect. All of this suggests that it is the bound offered by (NLP) that degrades, rather than the policies. This issue could be studied in future research that produces tighter bounds using stronger functional approximations.

Lastly, we make a few remarks regarding computation in solving (NLP). In general, we have found that the number of columns needed to prove optimality grows slowly with the number of eligible subsets, both of which are shown in Table 2. Furthermore, we have found that the vast majority of the computational effort in solving (NLP) is in calculating optimal traveling salesman costs $C_I$ on eligible subsets, for which we used CONCORDE by D. Applegate, R. Bixby, V. Chvátal, and W. Cook. Once these were tabulated, solving (NLP) took a relatively insignificant amount of time. These results were obtained on a Pentium IV Xeon 1.7GHz processor.

## Acknowledgments

The author thanks The Logistics Institute at Georgia Institute of Technology for sharing their knowledge and data concerning Praxair's gas distribution operations. He especially thanks Anton Kleywegt. He also thanks Linus Schrage, Gary Eppen, Don Eisenstein, and Martin Puterman for their many useful insights and suggestions. He also thanks the senior editor and two anonymous referees for many constructive comments. The author, however, assumes responsibility for any errors. The author thanks the developers of CONCORDE for making their software for solving traveling salesman problems publicly available to the research community. The author is also grateful for financial support from The University of Chicago, Graduate School of Business.

## Appendix

We show that the objective function used by Dror and Ball (1987) maximizes net value. We adopt their notation. It is assumed that customer storage tanks are larger than vehicle capacity, i.e., $T_i \geq q\ \forall i$. Let $V_i = b_i/T_i$, where $b_i$ is the (unknown) cost to replenish customer $i$. Then, we can rewrite $c_{it} = V_i(I_i' - \mu_i)$ and $g_{it} = V_i(T_i - I_i' + \mu_i t) = V_i d_{it}$, where $d_{it}$ is the delivery quantity if customer $i$ is replenished in period $t$. Now, subtract the constant $\sum_{i\in\overline{M}} V_i T_i$ to their objective function to obtain

$$\min \sum_{t=1}^{m}\sum_{w=1}^{W}\left(\mathrm{TSP}(N_{wt}) + \sum_{i\in\overline{M}} c_{it} y_{iwt} - \sum_{i\in M-\overline{M}} g_{it} y_{iwt}\right) - \sum_{i\in\overline{M}} V_i T_i$$
$$= \sum_{t=1}^{m}\sum_{w=1}^{W}\left(\mathrm{TSP}(N_{wt}) + \sum_{i\in\overline{M}} (c_{it} - V_i T_i) y_{iwt} - \sum_{i\in M-\overline{M}} g_{it} y_{iwt}\right),$$

where we use their constraint that customers $i \in \overline{M}$ are replenished exactly once during the planning horizon $m$. Because $c_{it} - V_i T_i = -V_i d_{it}$ and substituting the expression above for $g_{it}$, we then have

$$\max \sum_{t=1}^{m}\sum_{w=1}^{W} V_i d_{it} y_{iwt} - \mathrm{TSP}(N_{wt}),$$

which is the desired result.

For a representative TSP tour containing customer $i$, we use the method given in Appendix II of Dror and Ball (1987) to allocate the cost to obtain $b_i$. However, the authors do not suggest how to choose this representative tour in the absence of historical data, only that customers should be "nearby." Therefore, we use the location-based clustering model (CCLP) described in Bramel and Simchi-Levi (1995) for capacitated vehicle routing to partition the customers into TSP subsets, and assume that the demand for each customer equals $\min\{\overline{D}, \overline{S}_i\}$ (returning to our notation). Then, given $b_i$ we set $V_i = b_i/\min\{\overline{D}, \overline{S}_i\}\ \forall i$.

## References

Adelman, D. 1997. Remnant inventory systems. Ph.D. dissertation, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

——. 2004. A price-directed approach to stochastic inventory/routing. *Oper. Res.* Forthcoming.

——, G. L. Nemhauser. 1999. Price-directed control of remnant inventory systems. *Oper. Res.* **47** 889–898.

——, ——, M. Padron, R. Stubbs, R. Pandit. 1999. Allocating fibers in cable manufacturing. *Manufacturing Service Oper. Management* **1** 21–35.

Anily, S., A. Federgruen. 1990. One warehouse multiple retailer systems with vehicle routing costs. *Management Sci.* **36** 92–114.

Arapostathis, A., V. S. Borkar, E. Fernández-Gaucherand, M. K. Ghosh, S. I. Marcus. 1993. Discrete time controlled Markov processes with average cost criterion: A survey. *SIAM J. Control Optim.* **31** 282–344.

Bell, W. J., L. M. Dalberto, M. L. Fisher, et al. 1983. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces* **13**(6) 4–23.

Bellman, R. E. 1957. *Dynamic Programming*. Princeton University Press, Princeton, NJ.

——, S. E. Dreyfus. 1959. Functional approximations and dynamic programming. *Math. Tables Other Aids Comput.* **13** 247–251.

——, R. Kalaba, B. Kotkin. 1973. Polynomial approximation—A new computational technique in dynamic programming allocation processes. *Math. Comput.* **17** 155–161.

Berman, O., R. C. Larson. 2001. Deliveries in an inventory/routing problem using stochastic dynamic programming. *Transportation Sci.* **35** 192–213.

Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific, MA.

Bertsimas, D., J. Niño-Mora. 2000. Restless bandits, linear programming relaxations, and a primal-dual index heuristic. *Oper. Res.* **48** 80–90.

Bhattacharya, R. N., M. Majumdar. 1989. Controlled semi-Markov models under long-run average rewards. *J. Statist. Planning Inference* **22** 223–242.

Bramel, J., D. Simchi-Levi. 1995. A location based heuristic for general routing problems. *Oper. Res.* **43** 649–660.

Campbell, A., M. Savelsbergh, L. Clarke, A. Kleywegt. 1998. The inventory routing problem. T. G. Crainic, G. Laporte, eds. *Fleet Management and Logistics*. Kluwer Academic Publishers, Boston, MA, 95–112.

Chan, L. M., A. Federgruen, D. Simchi-Levi. 1998. Probabilistic analyses and practical algorithms for inventory routing models. *Oper. Res.* **46** 96–106.

Chen, R.-R., S. Meyn. 1999. Value iteration and optimization of multiclass queueing networks. *Queueing Systems* **32** 65–97.

Dantzig, G. B. 1963. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ.

de Farias, D., B. Van Roy. 2003. The linear programming approach to approximate dynamic programming. *Oper. Res.* **51**(6) 850–865.

Demski, J. 1994. *Managerial Uses of Accounting Information*. Kluwer Academic Publishers, Boston, MA.

Denardo, E. V., B. L. Fox. 1968. Multichain Markov renewal programs. *SIAM J. Appl. Math.* **16** 468–487.

Dopuch, N., D. Drake. 1964. Accounting implications of a mathematical programming approach to the transfer price problem. *J. Accounting Res.* **2**(Spring) 10–24.

Dror, M., M. Ball. 1987. Inventory routing: Reduction from an annual to a short-period problem. *Naval Res. Logist.* **34** 891–905.

Dynkin, E. B., A. A. Yushkevich. 1979. *Controlled Markov Processes*. Springer, Berlin-Heidelberg, New York.

Federgruen, A., Y. S. Zheng. 1992. The joint replenishment problem with general joint cost structures. *Oper. Res.* **40** 384–403.

Fisher, M. 1981. The Lagrangian relaxation method for solving integer programming problems. *Management Sci.* **27** 1–18.

——. 1994. Optimal solution of vehicle routing problems using minimum *k*-trees. *Oper. Res.* **42** 626–642.

——, A. Greenfield, R. Jaikumar, P. Kedia. 1982. Real-time scheduling of a bulk delivery fleet: Practical application of Lagrangean relaxation. Technical report, The Wharton School, University of Pennsylvania, Philadelphia, PA.

Fox, B. 1966. Markov renewal programming by linear fractional programming. *SIAM J. Appl. Math.* **14** 1418–1432.

Gans, N., G. van Ryzin. 1999. Dynamic vehicle dispatching: Optimal heavy traffic performance and practical insights. *Oper. Res.* **47** 675–692.

Hassin, R., N. Megiddo. 1991. Exact computation of optimal inventory policies over an unbounded horizon. *Math. Oper. Res.* **16** 534–546.

Hernández-Lerma, O. 1989. *Adaptive Markov Control Processes*. Springer-Verlag, New York.

——, J. B. Lasserre. 1996. *Discrete-Time Markov Control Processes: Basic Optimality Criteria*. Springer-Verlag.

Hirshleifer, J. 1956. On the economics of transfer pricing. *J. Bus.* **29**(July) 172–184.

Jewell, W. S. 1963. Markov-renewal programming, I and II. *Oper. Res.* **11** 938–971.

Kleywegt, A. J., V. S. Nori, M. W. Savelsbergh. 2002. The stochastic inventory routing problem with direct deliveries. *Transportation Sci.* **36** 94–118.

Lasdon, L. S. 1970. *Optimization Theory for Large Systems*. Macmillan.

Luque-Vásquez, F., O. Hernández-Lerma. 1999. Semi-Markov control models with average costs. *Appl. Math. (Warsaw)* **26** 315–331.

Powell, W. B., T. Carvalho. 1998. Dynamic control of logistics queueing networks for large scale fleet management. *Transportation Sci.* **32** 90–109.

——, H. Topaloglu. 2003. Stochastic programming in transportation and logistics. A. Ruszczynski, A. Shapiro, eds. *Handbook in Operations Research and Management Science: Stochastic Programming*, Vol. 10. Elsevier, Amsterdam, The Netherlands.

Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, New York.

Queyranne, M. 1987. Comment on "A dynamic programming algorithm for joint replenishment under general order cost functions." *Management Sci.* **33** 131–133.

Reiman, M. I., R. Rubio, L. M. Wein. 1999. Heavy traffic analysis of the dynamic stochastic inventory-routing problem. *Transportation Sci.* **33** 361–380.

Rosenblatt, M. J., M. Kaspi. 1985. A dynamic programming algorithm for joint replenishment under general order cost functions. *Management Sci.* **31** 369–373.

Roundy, R. O., W. L. Maxwell, Y. T. Herer, S. R. Tayur, A. W. Getzler. 1991. A price-directed approach to real-time scheduling of production operations. *IIE Trans.* **23** 149–160.

Rust, J. 1996. Numerical dynamic programming in economics. H. Amman, D. Kendrick, J. Rust, eds. *Handbook of Computational Economics*, Chapter 14. Elsevier, Amsterdam, The Netherlands, 614–722.

Schäl, M. 1992. On the second optimality equation for semi-Markov decision models. *Math. Oper. Res.* **17** 470–486.

Schweitzer, P. J., A. Seidmann. 1985. Generalized polynomial approximations in Markovian decision processes. *J. Math. Anal. Appl.* **110** 568–582.

Sigman, K., R. W. Wolff. 1993. A review of regenerative processes. *SIAM Rev.* **35** 269–288.

Simpson, R. W. 1989. Using network flow techniques to find shadow prices for market and seat inventory control. *MIT Flight Transportation Laboratory Memorandum M89-1*, Cambridge, MA.

Singh, S. P., R. C. Yee. 1994. An upper bound on the loss from approximate optimal value functions. *Machine Learning* **16** 227–233.

Sutton, R. S., A. G. Barto. 1998. *Reinforcement Learning*. MIT Press, Cambridge, MA.

Talluri, K., G. van Ryzin. 1998. An analysis of bid-price controls for network revenue management. *Management Sci.* **44** 1577–1593.

Williamson, E. L. 1992. Airline network seat control. Ph.D. dissertation, MIT, Cambridge, MA.

Zipkin, P. 2000. *Foundations of Inventory Management*. McGraw-Hill.

*Received: May 28, 2002; days with author: 63; revisions: 5; average review cycle time: 75 days; Senior Editor: Marshall Fisher.*