The Dynamic Transshipment Problem

Yale T. Herer[1*] **and Michal Tzur**[2]

[1] *Faculty of Industrial Engineering and Management, Technion-Israel Institute of Technology, Haifa 32000, Israel*

[2] *Department of Industrial Engineering, Tel Aviv University, Tel Aviv 69978, Israel*

Received October 1998; revised July 2000; accepted 2 February 2001

Abstract: We investigate the strategy of transshipments in a dynamic deterministic demand environment over a finite planning horizon. This is the first time that transshipments are examined in a dynamic or deterministic setting. We consider a system of two locations which replenish their stock from a single supplier, and where transshipments between the locations are possible. Our model includes fixed (possibly joint) and variable replenishment costs, fixed and variable transshipment costs, as well as holding costs for each location and transshipment costs between locations. The problem is to determine how much to replenish and how much to transship each period; thus this work can be viewed as a synthesis of transshipment problems in a static stochastic setting and multilocation dynamic deterministic lot sizing problems. We provide interesting structural properties of optimal policies which enhance our understanding of the important issues which motivate transshipments and allow us to develop an efficient polynomial time algorithm for obtaining the optimal strategy. By exploring the reasons for using transshipments, we enable practitioners to envision the sources of savings from using this strategy and therefore motivate them to incorporate it into their replenishment strategies. © 2001 John Wiley & Sons, Inc. Naval Research Logistics 48: 386–408, 2001

Keywords: transshipments; supply chains; dynamic inventory model

1. INTRODUCTION

Supply chain management has become an increasingly important consideration for many firms due to its impact on cost, service level, and production quality. Among other issues, it entails defining replenishment and associated inventory policies which are cost effective. One such policy, commonly practiced in multilocation inventory systems, involves movement of stock between locations at the same echelon level. These stock movements are termed *lateral stock transshipments*, or simply, *transshipments*.

Research efforts have generally viewed transshipments as an emergency recourse when unexpected circumstances have caused a surplus at one location and a shortage at another. One reason for considering only this reason for transshipments is the general lack of consideration of fixed replenishment costs. When these costs are present, we may want to replenish at one location

*This research was performed, in part, while the author was at the Department of Industrial Engineering, Tel Aviv University.

Correspondence to: Y.T. Herer or M. Tzur

and transship items to another location, in order to save on the fixed costs. Another reason for transshipments which was not discussed previously in the literature is to save on the holding costs, exploiting cases where different locations have different holding costs. (The latter reason may indeed be more important in settings where fixed replenishment costs exist, as in such cases more inventory is held.)

While both these reasons for transshipments would surely manifest themselves in the stochastic setting, we study their impact in the deterministic setting where the ‘‘traditional’’ reason for transshipments (i.e., to satisfy an unexpected shortage with a surplus) does not affect and complicate the issue. Moreover, we consider dynamic deterministic demand while only static stochastic demand has been considered in the literature. As in the majority of the literature on deterministic inventory models, the problem is important when planning is performed on the basis of a predetermined schedule. The latter can represent, for example, early orders, or a forecast of future demands. The dynamics of the demand over time may be a result of its randomness, and possibly also a result of the product’s seasonal or life-cycle trends.

The dynamic transshipment problem is characterized by several locations, each of which is characterized by fixed replenishment (or a set of locations may incur a joint replenishment cost), per unit replenishment, and per unit holding costs. In addition there is a fixed and per unit cost for transshipping between locations. A transshipment is defined as the transfer of stock between two locations at the same level of the inventory/distribution system. The problem is to determine replenishment quantities and how much to transship each period so as to satisfy deterministic dynamic demand at each location at minimal cost. The planning horizon is finite and no backorders are allowed.

Consider, for example, a department store which sells vacuum cleaners. The store may have several branches at various locations, with some customers requiring this item at one location and others requiring it at another location. When replenishing their stocks, the branches can either place separate orders with the supplier, or alternatively only one order may be placed by one of the branches, taking into consideration all branches’ needs for stock. In the first instance, several fixed replenishment costs are incurred (one for each branch), or, more generally, a joint replenishment cost is incurred. In the second case, only one fixed cost is incurred, and the vacuums will be delivered to the replenishing location and possibly transshipped (incurring transshipment costs) to (some of) the other locations; these transshipments can take place immediately or any time before the vacuum cleaners will be demanded. In other words, the use of transshipments allows the firm to determine the desired location(s) within the supply chain in which a global quantity of inventory is to be held, as opposed to the case where each location independently holds its own inventory. Such supply chain coordination is also in line with recent (technological) communication enhancements such as EDI, among the various participants of the supply chain (see, e.g., Stalk, Evans, and Shulman [23]).

The dynamic transshipment problem can be viewed as an extension of the classical Wagner–Whitin problem (Wagner and Whitin [27]). In this problem, a single product at a single location is considered and that location can replenish stock only from a single source, its supplier. When two or more locations are considered, the optimal strategy for each of them is determined independently. On the other hand, in the transshipment problem each location has two or more potential sources of supply [the supplier and the other location(s)]; therefore, the optimal strategy for all locations must be determined jointly. Since our research both explores additional reasons for transshipments (beyond the one studied in the literature for the static stochastic transshipment problem) and generalizes the Wagner–Whitin setting, it can be viewed as a synthesis of these two lines of research.

This work focuses on the dynamic transshipment problem with two locations, providing detailed analysis of properties, interesting structure of optimal policies, as well as an efficient polynomial time algorithm [$O(T^4)$, where T is the number of periods] to solve the problem. For a realistic problem, a typical planning horizon that is used for the determination of replenishment quantities, is not very large (10–30), thus allowing practitioners to implement the suggested algorithm. The interesting structure enhances understanding of the issues that are relevant in our problem setting and is the basis for our algorithm. This is the first time that transshipments are examined in a dynamic or deterministic setting as well as the first time with fixed transshipment costs. Following this work, Herer and Tzur [12] analyzed the multilocation dynamic transshipment problem in which the holding costs are identical at all locations. There, several transshipment mechanisms are discussed, the complexity is established to be NP-Hard, and an exponential time procedure which provides an optimal solution is developed. Since the time required to obtain an optimal solution in the multilocation environment is too large for realistic problems, they present and analyze the performance of a heuristic algorithm.

The rest of the paper is organized as follows. In the next section we review the most relevant literature. In Section 3 we introduce our notation and give preliminary results. In Section 4 we define the framework and properties used to develop the algorithm for the dynamic transshipment problem. In Section 5 we present our analysis and algorithm. For the sake of clarity we develop up to this point in the paper the model without joint replenishment costs and without fixed transshipment costs. In Section 6 we treat these two aspects of the model as extensions which can be incorporated into the framework and algorithm of Sections 4 and 5. Finally Section 7 contains some discussion and conclusions.

2. LITERATURE REVIEW

The issue of transshipments in a multilocation discrete time environment has been studied by many authors in a stochastic static setting without fixed transshipment costs. In this literature a differentiation is properly made between models that allow transshipments before demand is realized (see Allen [2, 3], Gross [9], Karmarkar and Patel [17], and Karmarkar [14–16]) and models that allow transshipments after demand is realized but before it needs to be satisfied (see Krishnan and Rao [18], Tagaras [24], Robinson [22], and Herer and Rashit [10, 11]); Tagaras and Cohen [25] add to the model nonzero lead times. An interesting variation on these two extremes is allowing transshipments while demand is being realized (Archibald, Sassen, and Thomas [4]). In the deterministic setting this differentiation is clearly not meaningful.

In his investigation of the two location problem, Tagaras [24] determined a transshipment policy which he called *complete pooling*. Under this policy the amount transshipped from location i to location j is the minimum between the excess at location i and the shortage at location j. Under certain, fairly general, cost structures one can show that complete pooling is optimal. Krishnan and Rao [18], Robinson [22], Pasternack and Drezner [21], Tagaras and Cohen [25], and Herer and Rashit [11] all assume this or related cost structures. Herer and Rashit [10] investigate alternative cost structures, i.e., what happens when complete pooling is not optimal.

The environment of inventory models with deterministic dynamic demand is known mostly due to the classical single product problem, introduced by Wagner and Whitin [27]. Since then, many variations of the classical model have been introduced: allowing backorders, considering capacity constraints, including startup costs and limiting the maximum inventory levels, to name just a few.

The literature on inventory models with dynamic demand for the class of multiproduct and/or multilocation models is relatively restricted. A polynomial time solution procedure which provides an optimal solution is known only for a *serial* system, where the production of a single item consists of several stages (see Zangwill [28] and Love [19]). Other multi-product models with dynamic demand include the *joint replenishment problem* (see, e.g., Joneja [13] and Federgruen and Tzur [7]) and the *capacitated multiproduct* model (see Maes and Van Wassenhove [20] for a review). Federgruen and Tzur [8] consider a multilocation model known as the *one-warehouse multiretailer* problem, which represents a two-level distribution system. These problems are known to be NP-complete, and only heuristic solutions or exponential algorithms (typically branch and bound) are available for them. None of the existing models with deterministic dynamic demand consider transshipment between locations.

3. NOTATION AND PRELIMINARIES

For the sake of clarity we first introduce notation and develop our model without fixed transshipment costs and without joint order costs. These two aspects of the model will be treated as extensions in Section 6.

The dynamic transshipment problem is defined by the following parameters:

- L number of locations $(i = 1, \dots, L)$ (in this paper we address the case where $L = 2$);
- T number of periods $(t = 1, \dots, T)$;
- d_{it} demand at location i in period t (for ease of exposition we assume $d_{it} > 0$ for all i and t);
- h_i holding cost incurred at location i for every unit held there for one period (we assume without loss of generality that $h_1 \le h_2$);
- K_i setup cost incurred whenever location i is replenished;
- c_i replenishment cost per unit at location i;
- $\hat{c}_{ij}$ direct transshipment cost per unit transshipped from location i to location j;
- c_{ij} effective transshipment cost, or simply the transshipment cost, per unit transshipped from location i to location j, $c_{ij} = \hat{c}_{ij} + c_i - c_j$.

Note that c_{ij} is considered the effective transshipment cost because when a unit is transshipped from location i to location j we pay, in addition to the direct transshipment cost, a cost of c_i instead of c_j to satisfy a unit of demand at location j. (Therefore, the constant $\sum_{i=1}^{2} c_i \sum_{t=1}^{T} d_{it}$ has to be added to the total cost that we obtain in order to get the true cost of a given solution.) Further note that c_{ij} can be less than zero; even though we would expect such a situation to be rare, it can be handled by our model without modification. In fact, we would expect that in most situations $c_i = c_j$ is satisfied, that is, $c_{ij} = \hat{c}_{ij} > 0$. In this case, the difference between h_1 and h_2 results solely from physical and geographical characteristics of the locations. For example, the size of the warehouse and its material handling efficiency, and whether the location is in an expensive business area or in a distant suburb. We observe that $c_{ij} + c_{ji} = \hat{c}_{ij} + \hat{c}_{ji} > 0$; this implies that it is not optimal to transship items back-and-forth.

The dynamic transshipment problem is to find a replenishment strategy for all locations over the finite horizon, such that demand at every location in every period is satisfied on time (no backorders are allowed) and the sum of fixed and variable replenishment costs, holding costs, and fixed and variable transshipment costs is minimized. Since in our analysis we use the effective transshipment costs, we do not further consider the variable replenishment costs.

The transshipment problem may be represented as a network flow problem on the following network, see Figure 1.

DEFINITION 1: The *replenishment network* has $2T + 1$ nodes: a source node, denoted as node 0, and a node for each location i for every period t, denoted as node (i, t) $(i = 1, 2$ and

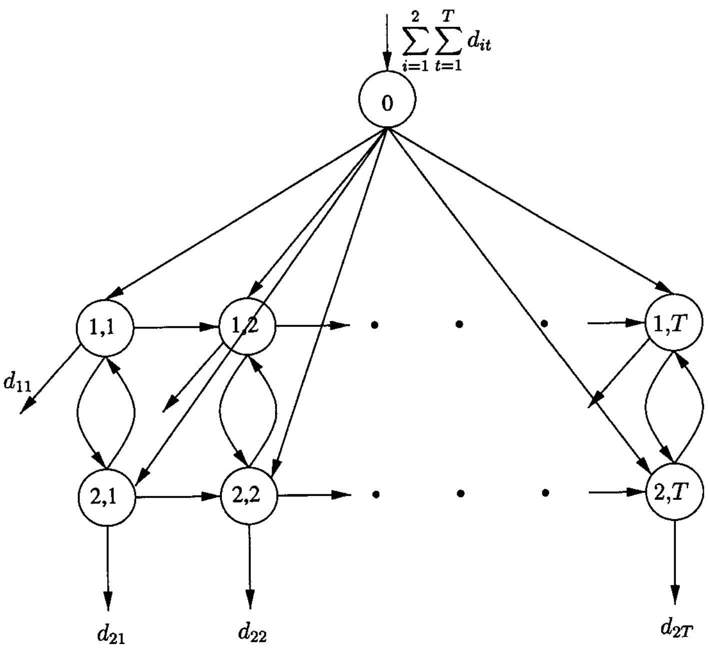

Figure 1. The replenishment network.

$1 \le t \le T$). The set of arcs consist of replenishment arcs, inventory arcs, and transshipment arcs: Replenishment arcs exist between the source node and every other node; inventory arcs exist between node (i, t) to $(i, t+1)$ for $i = 1, 2$ and all $1 \le t < T$; two transshipment arcs (one in each direction) exist between nodes $(1, t)$ and $(2, t)$ for every $1 \le t \le T$. The cost of the flow on these arcs are the replenishment costs (fixed and variable), holding costs (variable) and transshipment costs (fixed and variable), respectively. There is a demand of d_{it} units at node (i, t), and a supply at the source node of the sum of all demands.

A feasible flow (i.e., a flow that satisfies the demand requirements of the nodes) in this network corresponds with a replenishment plan for the transshipment problem, with the same cost value. Therefore, a feasible flow with minimum cost on this network minimizes the cost of the transshipment problem as well.

We note that the cost functions of all arcs are concave (fixed plus linear for the replenishment and transshipment costs, and linear for the holding costs). The theory on concave cost network flow problems provides us with an important theorem from which we derive a corollary that we will apply to the replenishment network.

THEOREM 1 (see, e.g., Denardo [5]): In any minimum cost uncapacitated network flow problem with concave costs there exists an optimal flow which does not contain any loop (an undirected cycle of arcs) with a positive flow on all arcs of the loop.

COROLLARY 1: In the replenishment network associated with the dynamic transshipment problem there exists an optimal flow in which no more than one arc with a positive flow enters each node (i, t). In particular, the demand at node (i, t) is supplied from a single source of replenishment.

If there were more than one arc with positive flow entering node (i, t), then the combination of the two paths with positive flow from the source node to node (i, t) would contain a loop. Corollary 1 is a generalization of the zero-inventory property, which is well known in classical deterministic inventory models that do not consider transshipments. In those models, only two possible sources exist for each location in every period: replenishment and inventory. The generalization in our problem is with respect to the third source, transshipments. An intuitive explanation of this corollary is the following: For a concave cost structure, the marginal cost of each unit is decreasing; therefore, it is not desirable to split an entering flow between two different sources.

4. SOLUTION FRAMEWORK

Our solution framework can be summarized as follows. We first define a structure which we call a *basic block*, a generalization of a similar structure that can be found in the solution to the Wagner–Whitin problem, although more intricate. We then show, as in the solution to the Wagner–Whitin problem, that there exists an optimal solution to the transshipment problem which can be described as a series of basic blocks. Next we describe how to formulate a shortest path problem, associating the arcs of a properly defined network with these basic blocks. Finally, we present an efficient way to calculate the cost of a basic block.

In the following definition s_i is used to denote the Starting period of the block at location i; similarly $e_i - 1$ is used to denote its Ending period. In addition, in the following definition and throughout the rest of the paper, whenever $s_i = e_i$ the series $(i, s_i), \ldots, (i, e_i - 1)$ is considered to be empty.

DEFINITION 2: A *block*, denoted by $\boldsymbol{(s_1, s_2) \to (e_1, e_2)}$, where $1 \leq s_1 \leq e_1 \leq T+1, 1 \leq s_2 \leq e_2 \leq T+1$, and either $s_1 < e_1$ or $s_2 < e_2$ (or both), is a set of nodes of the form $(1, s_1), \ldots, (1, e_1 - 1)$ and $(2, s_2), \ldots, (2, e_2 - 1)$ whose demand is satisfied from replenishment within these nodes and the replenishment within these nodes is not used to satisfy demand at nodes outside the block. We distinguish between two types of blocks:

- If $s_j = e_j$ for some $j = 1, 2$, then the block is called a *single-location block* because it has nodes from only one location.
- If $s_j \neq e_j$ for $j = 1, 2$, then the block is called a *two-location block* because it has nodes from both locations.

Note that the notation $(1, s_1)$ denotes a node in the replenishment network while $\boldsymbol{(s_1, s_2) \to (e_1, e_2)}$ denotes a block. To avoid any confusion, the latter is typeset bold.

DEFINITION 3: Two blocks are said to be *disjoint* if they have no common nodes.

We use Figure 2 to illustrate the definition of a block and of disjoint blocks. In the figure a solution is represented by a flow on the replenishment network. The arcs with positive flow have been bolded, and the flow satisfies Corollary 1. Thus, the path that each unit follows, from the source to a particular destination node, is unambiguous. The paths can be found by following the bolded arcs backward from every node up to the source node. According to Definition 2, the solution represented in Figure 2 contains five two location blocks **[(1, 1) → (4, 4), (1, 1) → (7, 5), (1, 1) → (7, 7), (4, 4) → (7, 5), (4, 4) → (7, 7)]** and two single location blocks (**(*t*, 2) → (*t*, 3)** and **(*t*, 5) → (*t*, 7)**, where $1 \leq t \leq T+1$, note that the identity of these blocks is independent of t). Note that the pair of blocks **(1, 1) → (4, 4)** and **(1, 1) → (7, 5)** are *not* disjoint whereas the pair **(1, 1) → (4, 4)** and **(4, 4) → (7, 5)** are.

The following observation is immediate from the definition of a block.

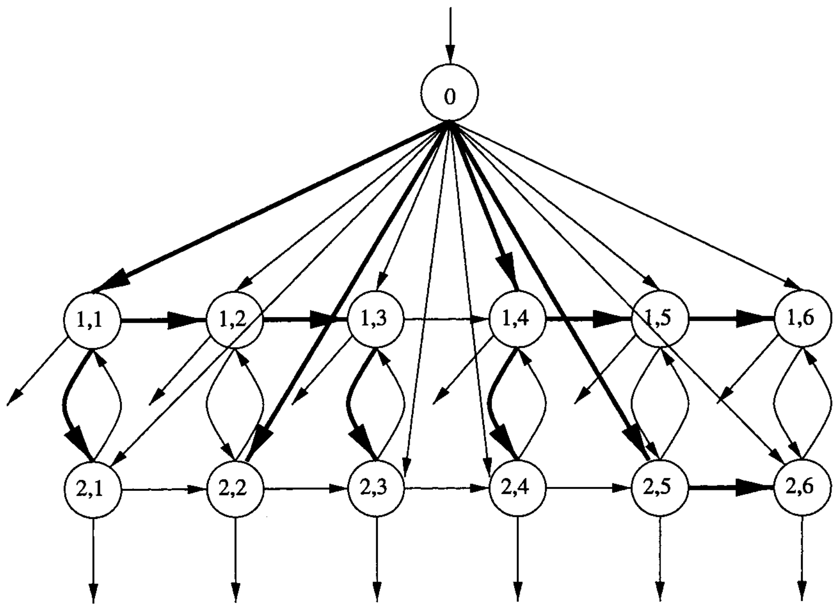

Figure 2. An example replenishment network showing the positive flows.

OBSERVATION 1: In any two location block $(s_1, s_2) \rightarrow (e_1, e_2)$ the starting inventory at location 1 in period s_1 and at location 2 in period s_2 [i.e., at nodes $(1, s_1)$ and $(2, s_2)$] is zero. In addition, the ending inventory at location 1 in period $e_1 - 1$ and at location 2 in period $e_2 - 1$ [i.e., at nodes $(1, e_1 - 1)$ and $(2, e_2 - 1)$] is zero.

While one may expect in an optimal solution that a block may be associated with a single replenishment, we find that there exists an exception to this rule. We refer to this exception as a hole of a block. In the following definition of a hole of a block u_i denotes the start of a hole of a block at location i; similarly $v_i - 1$ denotes its end.

DEFINITION 4: $(u_i) \rightarrow (v_i)$, for some $i = 1, 2$, is a *hole of a block* $(s_1, s_2) \rightarrow (e_1, e_2)$, if $s_i < u_i \leq v_i - 1 < e_i - 1$ and the replenishment at node (i, u_i) satisfies the demand of nodes $(i, u_i), \ldots, (i, v_i - 1)$.

The definition of a hole of a block does not include the situation where $u_i = s_i$ (without loss of generality let $i = 2$) because then the "hole" would be a block by itself and the block $(s_1, s_2) \rightarrow (e_1, e_2)$ would contain two disjoint blocks, i.e., $(s_1, s_2) \rightarrow (s_1, v_2)$ and $(s_1, v_2) \rightarrow (e_1, e_2)$ and the definition of a hole of a block would not be required here. For similar reasons the definition of a hole does not include the situation where $v_i = e_i$. Returning to Figure 2, we see one example of Definition 4, namely, we see that $(2_2) \rightarrow (3_2)$ is a hole of the block $(1, 1) \rightarrow (4, 4)$.

As will be seen in Theorem 2 below, only blocks, which we call basic, with a very specific replenishment structure are of interest.

DEFINITION 5: A *basic block* is a block which has the following properties:

1. If a block is a single-location block with $s_j = e_j$, then a replenishment occurs in period s_i at location i $(i \neq j)$ which is used to satisfy the demand at location i in periods $s_i, \ldots, e_i - 1$.
2. If a block is a two-location block, then a replenishment occurs either in period s_1 at location 1 or in period s_2 at location 2 which is used to satisfy the demand for location 1 in periods $s_1, \ldots, e_1 - 1$ and for location 2 in periods $s_2, \ldots, e_2 - 1$ except possibly for demand in periods whose replenishment is associated with a hole of the block.

Again returning to Figure 2 we see that out of the five two location blocks, only two are basic blocks **[(1, 1) → (4, 4)** and **(4, 4) → (7, 5)]**. All the single-location blocks are also basic blocks **[(*t*, 2) → (*t*, 3)** and **(*t*, 5) → (*t*, 7)**, where $1 \leq t \leq T + 1$; note that the first one is also a hole of the block **(1, 1) → (4, 4)]**.

In the sequel, we denote the location which replenishes as location p ($p = 1$ or $p = 2$).

THEOREM 2: There exists an optimal solution to the dynamic transshipment problem which can be described as a collection of disjoint basic blocks.

PROOF: Assume that the theorem is false and consider a problem for which there does not exist an optimal solution which can be described as a collection of disjoint basic blocks.

Of all optimal solutions satisfying Corollary 1, choose the one that has the maximal number of disjoint blocks. Such a solution always exists since in every solution the block **(1, 1) → (*T* + 1, *T* + 1)** is a collection of one or more disjoint blocks. Choose a block that is not basic and call this block $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$. Such a block exists due to the assumption made at the beginning of the proof. We consider two cases depending on whether the block $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$ is a single or two location block.

Case 1. $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$, ***is a single location block.*** Let location p be the location where $s_p < e_p$. We first note that we must replenish at location p in period s_p; this is because we do not allow backorders. Then, due to Corollary 1, these units cannot be transshipped from location p to the other location and back in some future period. Since the block is not basic, we know that there is another period (call it $\hat{s}_p, s_p < \hat{s}_p \leq e_p - 1$) in which replenishment occurs, and from Corollary 1 we know that the starting inventory at location p in period $\hat{s}_p$ is zero. Hence, the block can be split into two disjoint blocks (thus contradicting our maximality assumption).

Case 2. $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$ ***is a two location block.*** Since the block $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$ is not basic, there must be at least two replenishments within the block which are not holes. We shall examine the latest of these replenishments and refer to it simply as the latest replenishment. (If there is replenishment both at location 1 and location 2 in this period, then we arbitrarily choose one of them.) We now assume, for the ease of exposition, that the latest replenishment is at location 1. Let $\hat{s}_1$ be the period in which the latest replenishment occurs. The case in which the latest replenishment is at location 2 is analogous. We again consider two cases depending on whether the latest replenishment is used to satisfy some of the demand at location 2.

Case 2a. The latest replenishment does not *satisfy any demand at location* 2. According to Corollary 1 the units in the latest replenishment can not pass through location 2 on their way to satisfying demand at location 1. Thus the units of this replenishment are held at location 1 until they are depleted. If these units are used to satisfy demand at node $(1, e_1 - 1)$ [and therefore at all nodes $(1, \hat{s}_1 + 1), \ldots, (1, e_1 - 2)$ as well], then the block $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$ can be split into two disjoint blocks [namely, $(\boldsymbol{s_1}, \boldsymbol{s_2}) \rightarrow (\boldsymbol{\hat{s}_1}, \boldsymbol{e_2})$ and $(\boldsymbol{\hat{s}_1}, \boldsymbol{e_2}) \rightarrow (\boldsymbol{e_1}, \boldsymbol{e_2})$], a contradiction to the maximality assumption. If these units are not used to satisfy demand at node $(1, e_1 - 1)$, then this replenishment forms a hole, a contradiction to the definition of the latest replenishment.

Case 2b. The latest replenishment does satisfy some of the demand at location 2. Let the first period in which some of the demand at location 2 is satisfied by the latest replenishment be called period $\hat{s}_2$. Since demand at nodes $(2, \hat{s}_1), (2, \hat{s}_1 + 1), \ldots, (2, \hat{s}_2 - 1)$ is not satisfied from this replenishment, the demand at node $(2, \hat{s}_2)$ must be received from location 1 in period $\hat{s}_2$; therefore, the demand at node $(1, \hat{s}_2)$ [and nodes $(1, \hat{s}_1), (1, \hat{s}_1 + 1), \ldots, (1, \hat{s}_2 - 1)$ as well] is also satisfied from the replenishment at $(1, \hat{s}_1)$. From Corollary 1 applied to nodes $(1, \hat{s}_2)$ and $(2, \hat{s}_2)$ (a cut set in the network) we know that the demand at nodes $(1, \hat{s}_2), (1, \hat{s}_2 + 1), \ldots, (1, e_1 - 1)$ and nodes $(2, \hat{s}_2), (2, \hat{s}_2 + 1), \ldots, (2, e_2 - 1)$ are all met from the replenishment at node $(1, \hat{s}_1)$ (possibly except for holes). Putting this all together, we have that this latest replenishment satisfies all the demand (possibly except for holes) at nodes $(1, \hat{s}_1), (1, \hat{s}_1 + 1), \ldots, (1, e_1 - 1)$ and nodes $(2, \hat{s}_2), (2, \hat{s}_2 + 1), \ldots, (2, e_2 - 1)$ and nowhere else. Therefore, the block $\boldsymbol{(s_1, s_2) \rightarrow (e_1, e_2)}$ can be split into two disjoint blocks [namely, $\boldsymbol{(s_1, s_2) \rightarrow (\hat{s}_1, \hat{s}_2)}$ and $\boldsymbol{(\hat{s}_1, \hat{s}_2) \rightarrow (e_1, e_2)}$], a contradiction to the maximality assumption. □

Now that the form of an optimal solution has been identified, let us consider the cost of a solution that takes this form. Since the costs of disjoint blocks are independent, the total cost is simply the sum of the cost of the basic blocks that make up the solution. The cost of a basic block is the minimum cost of satisfying the demand of all the nodes in the block with replenishment being as outlined in the definition of a basic block. A detailed algorithm for calculating the cost of a block is presented in the next section.

We now turn our attention to finding an optimal series of basic blocks, assuming that the cost of each basic block is already known. For this we define the following network which is illustrated in Figure 3.

DEFINITION 6: The *block network* is the network whose nodes are all ordered pairs of the form $\boldsymbol{(s_1, s_2)}$, $1 \leq s_1, s_2 \leq T + 1$ and an arc exists between two nodes $\boldsymbol{(s_1, s_2)}$ and $\boldsymbol{(e_1, e_2)}$ whenever $s_1 \leq e_1$) and $s_2 \leq e_2$ (except, of course, if $s_1 = e_1$ and $s_2 = e_2$).

DEFINITION 7: The *cost of an arc* in the block network from node $\boldsymbol{(s_1, s_2)}$ to node $\boldsymbol{(e_1, e_2)}$ is the cost of the basic block $\boldsymbol{(s_1, s_2) \rightarrow (e_1, e_2)}$**,** and is denoted by $M(s_1, s_2; e_1, e_2)$. We also denote this arc in the same notation as a block, i.e., $\boldsymbol{(s_1, s_2) \rightarrow (e_1, e_2)}$.

We note that each path in the block network from node $\mathbf{(1, 1)}$ to node $\boldsymbol{(T+1, T+1)}$ corresponds to a series of mutually disjoint basic blocks which, when considered together, cover all the nodes in the replenishment network. Thus each such path corresponds to a feasible solution of the dynamic transshipment problem. Furthermore, since the costs associated with the arcs in the block network correspond to the costs of these basic blocks, the sum of the costs of the arcs on a path from node $\mathbf{(1, 1)}$ to node $\boldsymbol{(T+1, T+1)}$ corresponds to the costs of a feasible solution. Moreover, any series of mutually disjoint basic blocks which, when considered together, cover all the nodes in the replenishment network, is associated with a path in the block network from node $\mathbf{(1, 1)}$ to node $\boldsymbol{(T+1, T+1)}$. In particular this is true for the optimal solution characterized in Theorem 2. Thus, an optimal solution can be identified by finding the shortest path in the block network from node $\mathbf{(1, 1)}$ to node $\boldsymbol{(T+1, T+1)}$.

The block network can be viewed as a generalization of the network which one defines in order to solve the Wagner–Whitin problem (Wagner and Whitin [27]). The network used to solve this problem has one node for every time period and an arc going from every node s to every other node e, such that $s < e$. An arc from node s to node e in this network indicates that

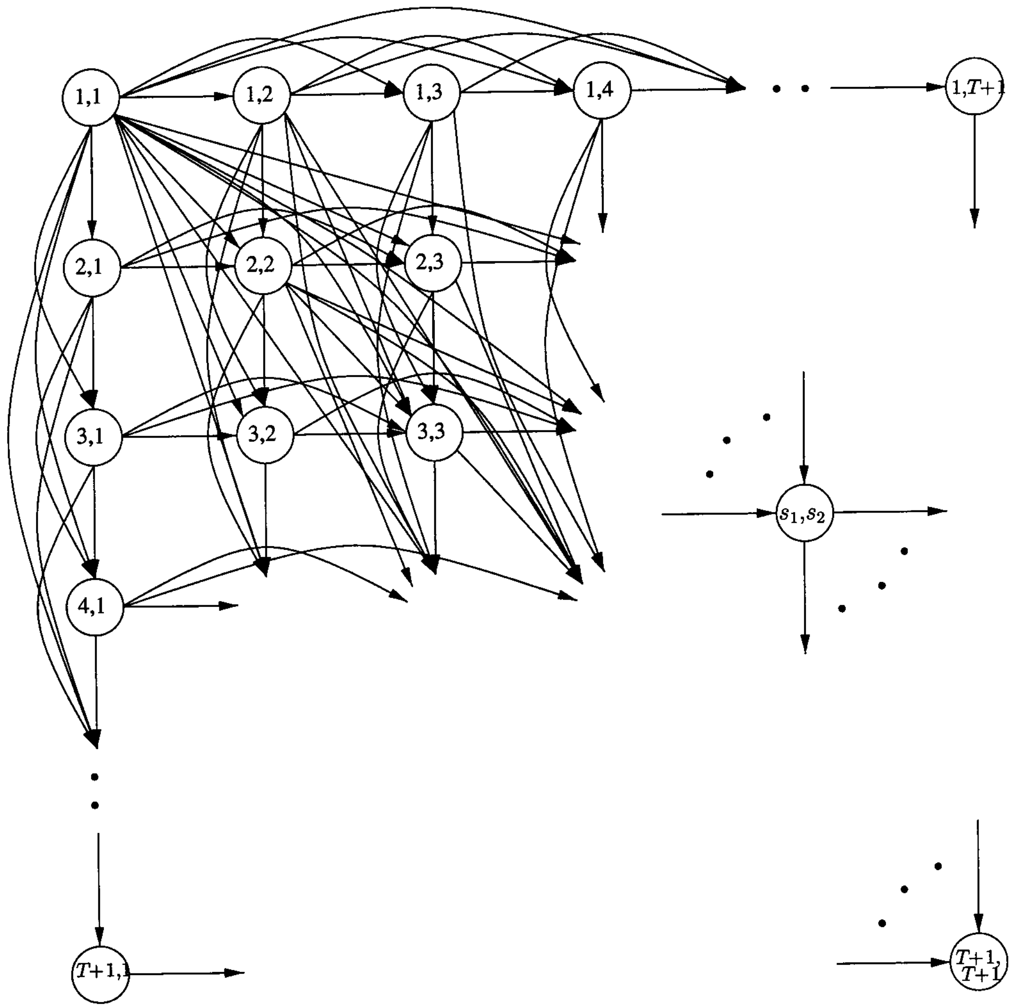

Figure 3. The block network.

- replenishment occurs in periods s and e and nowhere in between,
- all the demands in periods s through $e-1$ are met through the single replenishment in period s, and
- the starting inventory is zero in periods s and e and nowhere in between.

The block network used to solve our dynamic transshipment problem has one node for every pair of time periods and an arc from every node (s_1, s_2) to every other node (e_1, e_2) whenever $s_1 \le e_1$ and $s_2 \le e_2$ (and at least one of these inequalities is strict). An arc from node (s_1, s_2) to node (e_1, e_2) in this network indicates that:

- Replenishment may occur only at nodes $(1, s_1)$ or $(2, s_2)$, and $(1, e_1)$, and/or $(2, e_2)$ and nowhere in between (except possibly for holes).
- All demands of the block are met from replenishment either at node $(1, s_1)$ or at node $(2, s_2)$ (except possibly for holes).

- The starting inventory is zero at nodes $(1, s_1), (2, s_2), (1, e_1)$, and $(2, e_2)$ and nowhere in between at location 1 [i.e., there does not exist a node $(1, b_1), s_1 < b_1 < e_1$, such that its starting inventory is zero]. If the block is a single location block at location 2, then there does not exist a node $(2, b_2), s_2 < b_2 < e_2$, such that its starting inventory is zero.

For both problems a shortest path in the associated network corresponds with an optimal solution. To complete the analogy, we use the following definition:

DEFINITION 8: $\boldsymbol{u} \rightarrow \boldsymbol{v}$, $1 \leq u < v \leq T$ is a *Wagner–Whitin basic block* if a replenishment in period u is used to satisfy all the demand in periods $u, \ldots, v-1$ and no other demand.

Now we see the full extent and limitation of the analogy between the basic blocks of the Wagner–Whitin problem and the basic blocks of the dynamic transshipment problem. In each case there exist an optimal solution which is a series of basic blocks. However, in the Wagner–Whitin problem these blocks are always "whole,"[1] whereas in the dynamic transshipment problem they may contain "holes." These holes create exceptional and nontrivial situations in each of the above-mentioned points.

In order to illustrate the relationship between the replenishment network, basic blocks, and the block network for the dynamic transshipment problem, consider a four period problem $(T = 4)$ in which an optimal solution is as described in the following example.

EXAMPLE 1:

- Replenish location 1 in period 1 for location 1 in periods 1 and 2 and for location 2 in period 1.
- Replenish location 2 in period 2 for location 2 in periods 2 and 3.
- Replenish location 1 in period 3 for location 1 in periods 3 and 4 and for location 2 in period 4.

This solution is represented in the replenishment network of Figure 4 in which the arcs with positive flow have again been bolded. This solution corresponds to the following three basic blocks. **(1, 1) → (3, 2), (3, 2) → (3, 4)**, and **(3, 4) → (5, 5)** and is represented in the block network of Figure 5. The arcs corresponding to basic blocks in the optimal solution have been bolded and, for the purpose of clarity, the other arcs have been eliminated.

5. THE SOLUTION TO THE DYNAMIC TRANSSHIPMENT PROBLEM

In the previous section we showed that there exists an optimal solution which is a series of disjoint basic blocks. We also showed that given the cost of all the basic blocks we can efficiently find an optimal solution. In this section we

1. show how one can determine the cost of a basic block,
2. show how these calculations can be performed efficiently.

[1]That is, without holes.

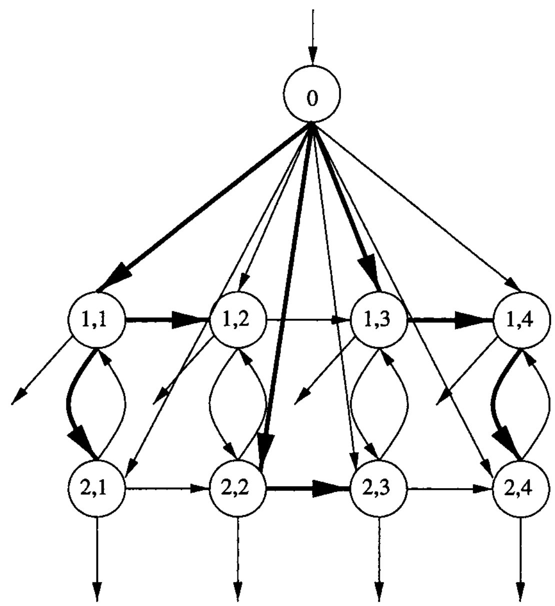

Figure 4. The replenishment network for Example 1.

5. 1. Determining the Cost of a Basic Block

We first consider basic blocks without holes. In this case, we have a replenishment at node (p, s_p) (recall that p denotes the replenishing location and similarly we now use r to denote the receiving location) that covers demand at nodes $(p, s_p), (p, s_p + 1), \ldots, (p, e_p - 1)$ if it is a single-location block and demand at these nodes and nodes $(r, s_r), (r, s_r + 1), \cdots (r, e_r - 1)$ if it is a two-location block. Once a replenishment is performed, all the costs of sending the material from node (p, s_p) to the various nodes in the block are independent of each other. We thus let $SP(p, s_p; i, k)$ be the shortest path (the minimum cost of transferring one unit of the item) in the replenishment network from the node where replenishment occurs [i.e., (p, s_p)] to an arbitrary node in the block [e.g., (i, k)].

We now present some properties of optimal flows between two nodes of a block in the replenishment network, leading to an expression for the cost of a basic block. These properties follow from the holding cost differential assumption, $h_1 \leq h_2$, which was made without loss of generality.

PROPERTY 1: If a unit is replenished at location 1 for use at location 1, it is held in inventory only at location 1. $SP(1, s_1; 1, k) = \sum_{t=s_1}^{k-1} h_1 = h_1(k - s_1)$.

PROPERTY 2: If a unit is replenished at location 1 for use at location 2, it is held in inventory at location 1 until it is needed at location 2 at which time it is transshipped. $SP(1, s_1; 2, k) = \sum_{t=s_1}^{k-1} h_1 + c_{12}$.

PROPERTY 3: If a unit is replenished at location 2 for use at location 1, then it is immediately transshipped to location 1 and held there until it is needed. $SP(2, s_2; 1, k) = c_{21} + \sum_{t=s_2}^{k-1} h_1$.

PROPERTY 4: If a unit is replenished at location 2 for use at location 2, then it is either held in inventory at location 2 until it is needed, or it is immediately transshipped to location 1 and

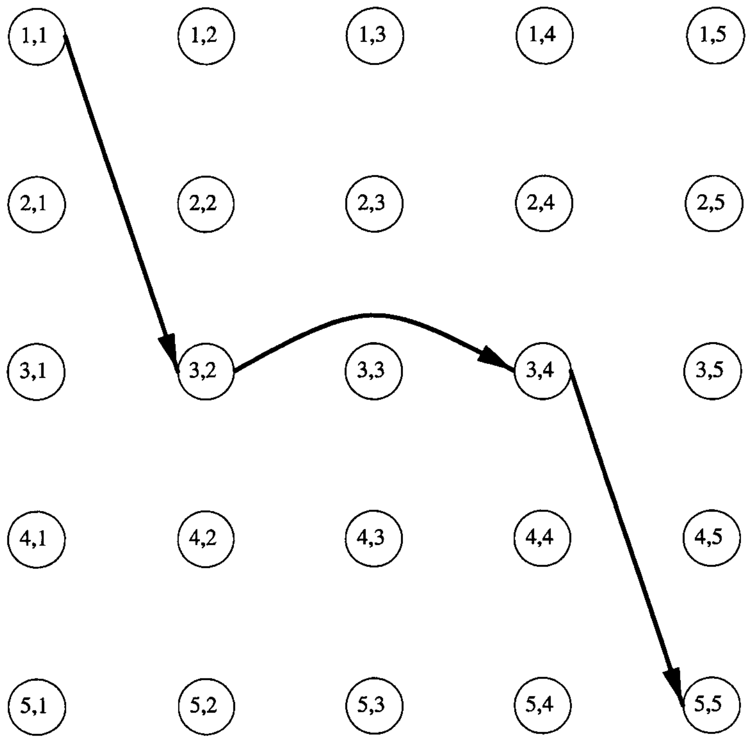

Figure 5. The block network for Example 1.

held there until it is needed at location 2 (at which time it is transshipped back). $SP(2, s_2; 2, k) = \min(\sum_{t=s_2}^{k-1} h_2, c_{21} + \sum_{t=s_2}^{k-1} h_1 + c_{12})$.

We can now write the cost of a basic block (both single and two location) that does not have any holes [denoted by $M^{NH}(s_1, s_2; e_1, e_2)$]] as follows:

$$M_p^{NH}(s_1, s_2; e_1, e_2) = K_p + \sum_{t=s_1}^{e_1-1} SP(p, s_p; 1, t) d_{1t} + \sum_{t=s_2}^{e_2-1} SP(p, s_p; 2, t) d_{2t}. \tag{1}$$

We now consider holes by first demonstrating that a hole [denoted $(\boldsymbol{u_i}) \to (\boldsymbol{v_i})$] can only occur at location 2 when location 1 is replenishing.

PROPERTY 5: A hole can not occur at location 1 when location 1 is replenishing because of Corollary 1 and Property 1 above. If such a hole did exist, node $(1, u_1)$ would receive inventory both from replenishment and from node $(1, s_1)$ [on its way to node $(1, e_1 - 1)$].

PROPERTY 6: A hole can not occur at location 1 when location 2 is replenishing because of Corollary 1 and Property 3 above. If such a hole did exist, node $(1, u_1)$ would receive inventory both from replenishment and from node $(2, s_2)$ [on its way to node $(1, e_1 - 1)$].

PROPERTY 7: A hole can not occur at location 2 when location 2 is replenishing because in this case it would be cheaper to supply node $(2, e_2 - 1)$ through the replenishment at node $(2, u_2)$ rather than node $(2, s_2)$.

We now consider the possibility of having holes at location 2 when location 1 is replenishing. To motivate the existence of holes, we consider the following example problem in which the optimal solution has a hole.

EXAMPLE 2:

Location 1: $K_1 = 100, h_1 = 1, c_{12} = 2, d_{11} = 100, d_{12} = 2, d_{13} = 2$;

Location 2: $K_2 = 100, h_2 = 4.5, c_{21} = 2, d_{21} = 2, d_{22} = 100, d_{23} = 1$.

After careful examination[2] one can see that the optimal policy, which is illustrated in the first three periods of the replenishment network found in Figure 2, is to replenish 107 units at node (1, 1) [for nodes (1, 1), (1, 2), (1, 3), (2, 1), and (2, 3)] and to replenish 100 units at node (2, 2) [for node (2, 2)]. Here, $\mathbf{(2_2) \to (3_2)}$ is a hole in the block $\mathbf{(1, 1) \to (4, 4)}$.

Since we have already identified the optimal flow in the replenishment network except for holes, the only question remaining is how to determine where the hole(s) are (if at all). As we showed in Theorem 2, even though a basic block may contain holes, demands during periods s_2 and $e_2 - 1$ are satisfied from the replenishment at location 1. Thus, it remains to determine if holes exist anywhere in the periods $s_2 + 1, s_2 + 2, \ldots, e_2 - 2$. First note, that the hole $\boldsymbol{(u_2) \to (v_2)}$ has cost

$$H(u_2; v_2) = K_2 + \sum_{t=u_2+1}^{v_2-1} (t - u_2) h_2 d_{2t},$$

since, by the definition of a hole, replenishment in period u_2 satisfies the demand of all periods in the hole, i.e., periods $u_2, u_2 + 1, \ldots, v_2 - 1$. Note that, even though in general it may be optimal to replenish items at location 2, transship them to location 1, hold them there, and transship them back to location 2 in some future period, this will never occur with holes because this would mean that node $(1, u_2)$ would have two sources of supply.

Recall that a Wagner–Whitin basic block $\boldsymbol{u \to v}$ consists of periods $u, u + 1, \ldots, v - 1$, where the replenishment in period u satisfies the demand of all the periods in the block. This is clearly the same structure as the blocks which are holes in our problem. Thus, the existence of holes implies that Wagner–Whitin basic blocks may occur at location 2 as part of the optimal arrangement of a basic block.

We define a *modified* Wagner–Whitin problem, whose purpose is to find the minimum cost of satisfying the demand of periods $s_2 + 1, s_2 + 2, \ldots, e_2 - 2$, since those are the periods in which holes may exist. Recall that a cost of a hole (a Wagner–Whitin basic block) is denoted by $H(u_2; v_2)$. Solving the standard Wagner–Whitin problem for periods $s_2 + 1, s_2 + 2, \ldots, e_2 - 2$

[2]For the two demand points of 100 (d_{11} and d_{22}) it is preferable to make a special replenishment whose cost is 100 each, since transferring or holding each of these 100 units would cost more than 100. For no other demand points is it worth making a special replenishment; thus these are the only two replenishments. For location 1, once the replenishment in period 1 is performed, the most economical way of satisfying the two units demand of period 2 and the two units demand of period 3 is by holding them from period 1, at a cost of 2 and 4, respectively. (Satisfying these demand points from transshipment is more costly.) For location 2, once the replenishments were determined as above, the demand of the first period can only be satisfied from transshipment. The one unit demand of period 3 can be satisfied either from the replenishment at location 2 in period 2, in which case the cost of holding the unit for one period is 4.5, or from transshipment from location 1. In the latter case, the unit is ordered in period 1, held at location 1 until period 3, and then transshipped to location 2. The cost of holding and transshipping is then 4, and therefore preferred over the former possibility.

[i.e., using $H(u_2; v_2)$ as the cost of a Wagner–Whitin basic block] would mean satisfying these nodes through a series of holes at minimal cost. However, in our optimal solution some of these nodes may be supplied through the original replenishment at node $(1, s_1)$. To have the standard Wagner–Whitin algorithm consider this, we must modify the cost of some Wagner–Whitin basic blocks. Letting $M^{s_1}(u_2; v_2)$ be the cost of a modified Wagner–Whitin basic block, we observe that if it contains two (or more) nodes, then it does not involve replenishment at node $(1, s_1)$ because it is just as economical to supply the two (or more) nodes separately from node $(1, s_1)$; thus, if $u_2 < v_2 - 1$, then $M^{s_1}(u_2; v_2) = H(u_2; v_2)$. As a result, the only necessary modification is with respect to Wagner–Whitin basic blocks that contain only one node. In this case the cost of a special replenishment at node $(2, u_2)$ must be compared with the cost of receiving the supply from node $(1, s_1)$. Therefore, the cost of this modified block is the least costly alternative, i.e., $M^{s_1}(u_2; u_2 + 1) = \min(K_2, SP(1, s_1; 2, u_2)d_{2u_2})$. We conclude that, in order to determine where the holes are, we need to solve this modified Wagner–Whitin problem at location 2 for periods $s_2 + 1, s_2 + 2, \ldots, e_2 - 2$.

If we denote the cost of the solution to the modified Wagner–Whitin problem described above by $WW^{s_1}(s_2 + 1, e_2 - 2)$ [with $WW^{s_1}(s_2 + 1, s_2) = 0$], then we have the cost of a two-location block when replenishment is at location 1 as being:

$$M^H(s_1, s_2; e_1, e_2) = K_1 + \sum_{t=s_1}^{e_1-1} SP(1, s_1; 1, t)d_{1t} + SP(1, s_1; 2, s_2)d_{2,s_2} + WW^{s_1}(s_2 + 1, e_2 - 2) + SP(1, s_1; 2, e_2 - 1)d_{2,e_2-1},$$

where the superscript H denotes the possibility of holes being included in the block. At the end of this section we present a simple algorithm to calculate the cost of a basic block $\boldsymbol{(s_1, s_2) \to (e_1, e_2)}$ based on the above analysis. In addition, we determine whether a given basic block is admissible in the block network [i.e., whether the arc from $\boldsymbol{(s_1, s_2)}$ to $\boldsymbol{(e_1, e_2)}$ exists]. This determination is based on the following properties:

PROPERTY 8: If replenishment is at location 1 for a two-location block, then we know that $s_1 \le s_2 \le e_2 - 1 \le e_1 - 1$. The first inequality is due to the fact that we do not allow backorders. The second holds by definition. Finally, the third inequality holds because, as explained in Property 2 above, units replenished at location 1 for location 2 are held at location 1 until they are needed; if $e_1 - 1$ were less than $e_2 - 1$, then node $(1, e_2 - 1)$ would have units from two different sources.

PROPERTY 9: If replenishment is at location 2 for a two-location block, then we know that $s_2 = s_1$. The equality holds due to the fact that any replenishment intended for location 1 is immediately transshipped, and if s_2 were less than s_1, then node $(1, s_2)$ would have units from two different sources.

We now present the algorithm for determining whether a two-location basic block is an admissible block in an optimal solution, and if so, we give its cost, $M(s_1, s_2; e_1, e_2)$. Single location basic blocks are all admissible and we can use Eq. (1) to calculate their costs, $M(s_1, s_2; e_1, e_2) = M_p^{NH}(s_1, s_2; e_1, e_2)$.

Algorithm

STEP 1: If ($s_1 < s_2$ and $e_1 < e_2$) or ($s_2 < s_1$) then this block is not admissible; go to Step 4.
STEP 2: If $s_1 < s_2$ then $p = 1$,
if $e_2 = s_2 + 1$ then $M(s_1, s_2; e_1, e_2) = M_1^{NH}(s_1, s_2; e_1, e_2)$; go to Step 4
else ($e_2 > s_2 + 1$) $M(s_1, s_2; e_1, e_2) = M^H(s_1, s_2; e_1, e_2)$; go to Step 4.

STEP 3: If $s_1 = s_2$ then
if $e_1 < e_2$ then $p = 2$ and $M(s_1, s_2; e_1, e_2) = M_2^{NH}(s_1, s_2; e_1, e_2)$; go to Step 4
else $(e_1 \geq e_2)$ # Is it cheaper to replenish at location 1 or 2?
$M_2 = M_2^{NH}(s_1, s_2; e_1, e_2)$
If $e_2 = s_2 + 1$ then $M_1 = M_1^{NH}(s_1, s_2; e_1, e_2)$.
else $(e_2 > s_2 + 1)$ $M_1 = M^H(s_1, s_2; e_1, e_2)$.
If $M_1 \leq M_2$ then $p = 1$ and $M(s_1, s_2; e_1, e_2) = M_1$; go to Step 4
else $(M_1 > M_2)$ $p = 2$ and $M(s_1, s_2; e_1, e_2) = M_2$; go to Step 4

STEP 4: End.

5. 2. Efficiently Calculating the Cost of All Basic Blocks

The block network has $O(T^2)$ nodes and $O(T^4)$ arcs, and therefore a shortest path from node $\mathbf{(1, 1)}$ to node $\mathbf{(T+1, T+1)}$ can be found in $O(T^4)$ time, given that the costs of all arcs are known. In this section we show how all arc costs can be calculated in $O(T^4)$ time, therefore leading to a total complexity of $O(T^4)$ time for the entire algorithm.

In calculating the cost of an arc in the block network, the shortest path between the replenishing location and the receiving location in the replenishment network is naturally used frequently; therefore, we start by calculating the cost of these shortest paths. Since the replenishment network has $O(T)$ nodes, the cost of all shortest paths can be calculated in $O(T^2)$ time. Alternatively, the formulas in Properties 1–4 may be used (and implemented efficiently) to calculate these costs, with the same complexity.

The next step is to calculate the cost of all *single-location* basic blocks, i.e., arcs in the block network with $s_j = e_j$ for some $j = 1, 2$. We use the following recursive calculations for $s_2 = e_2$, where we use the notation $M_p(s_1, s_2; e_1, e_2)$ to denote the cost of the basic block $\mathbf{(s_1, s_2) \to (e_1, e_2)}$ when replenishment is at location p:

$$M_1(s_1, s_2; s_1 + 1, s_2) = K_1,$$

$$M_1(s_1, s_2; e_1, s_2) = M_1(s_1, s_2; e_1 - 1, s_2) + SP(1, s_1; 1, e_1 - 1)d_{1,e_1-1}.$$

The above calculation can be done in constant time for given s_1 and e_1, therefore in $O(T^2)$ time for all pairs of s_1 and e_1. A similar recursive calculation applies for $s_1 = e_1$, which can be done again in $O(T^2)$ time.

The arc costs of the *two-location* basic blocks are calculated next, using the costs of single location blocks as a starting point. In particular, for $p = 2$ we start with the basic block $\mathbf{(s_2, s_2) \to (s_2, e_2)}$ and use the following relationship:

$$M_2(s_2, s_2; e_1, e_2) = M_2(s_2, s_2; e_1 - 1, e_2) + SP(2, s_2; 1, e_1 - 1)d_{1,e_1-1}.$$

The limitation of $s_1 = s_2$ is based on Property 9, and therefore does not exclude any admissible blocks. Due to this limitation there are a total of $O(T^3)$ possible combinations of $s_1(= s_2), e_1$ and e_2, for every combination the above calculation takes constant time and therefore these arc costs can be calculated in $O(T^3)$ time.

For the two location blocks when $p = 1$ there is one special case, associated with blocks that necessarily don't contain holes, whose calculation is different from the general case, namely, when $e_2 = s_2 + 1$:

$$M_1(s_1, s_2; e_1, s_2 + 1) = M_1(s_1, s_2; e_1, s_2) + SP(1, s_1; 2, s_2)d_{2,s_2}.$$

A single-location block is again used as the starting point, and each calculation is performed in constant time; for all possible combinations of s_1, s_2, e_1, and e_2 $(= s_2 + 1)$, the complexity is $O(T^3)$.

The general case of two-location blocks when $p = 1$ involves the preexecution of the dynamic program of Wagner and Whitin on the modified Wagner–Whitin costs.[3] Recall that the modified Wagner–Whitin problem is associated with nodes $s_2 + 1, s_2 + 2, \ldots, e_2 - 2$, of location 2, where holes may exist. It is well known (see, e.g., Federgruen and Tzur [6]) that for any starting period u_2 $(u_2 \geq s_2 + 1)$, the cost up to the end of the horizon [i.e., $WW^{s_1}(u_2, T)$] can be calculated in $O(T^2)$ time, and the costs $WW^{s_1}(u_2, v_2)$ for all $u_2 \leq v_2 < T$ are obtained as well during these calculations. There are $O(T)$ possible starting points (values of u_2) and $O(T)$ possible values for s_1, and therefore the complexity to calculate all $WW^{s_1}(u_2, v_2)$ for $1 \leq s_1 < u_2 \leq v_2 < T$ is $O(T^4)$. Given the costs of the modified Wagner–Whitin problem, the cost of a two-location block with $p = 1$ and $e_2 - 1 > s_2$ is calculated as follows:

$$M_1(s_1, s_2; e_1, e_2) = M_1(s_1, s_2; e_1, s_2) + SP(1, s_1; 2, s_2)d_{2,s_2} + WW^{s_1}(s_2 + 1, e_2 - 2) + SP(1, s_1; 2, e_2 - 1)d_{2,e_2-1}.$$

This case is computationally the most expensive, with total complexity of $O(T^4)$ time since all $O(T^4)$ combinations of s_1, s_2, e_1, e_2 have to be considered, each of which takes constant time to calculate. As mentioned earlier, the calculation of all arc costs in $O(T^4)$ time implies that the complexity of the entire dynamic transshipment problem is $O(T^4)$ time as well.

6. EXTENSIONS TO THE MODEL

The dynamic transshipment problem as defined in the Introduction includes fixed transshipment and joint replenishment costs. To simplify the exposition and facilitate the reader's understanding, we have left these two aspects of the model to this section. We describe separately the impact of each of these aspects on the analysis; when both aspects prevail together, the modifications required are straightforward.

6.1. Fixed Transshipment Costs

In this section we investigate the impact of introducing fixed transshipment costs into the model. The variable transshipment costs as well as all other elements of the model remain unchanged. When referring to the analysis of the previous sections, we sometimes call the model discussed there the *basic model*.

With the existence of fixed transshipment costs the development up to and including Section 4 remains unchanged; most importantly, the definition of a basic block, Theorem 2, and its proof are unaffected. In contrast, not all the properties stated in Section 5 continue to hold. More specifically, Properties 2, 4, and 8 are no longer satisfied since, due to the fixed transshipment costs, it now may be more economical to consolidate the items to be transshipped from location 1 to location 2 into one large transshipment rather than to transship small quantities whenever they are needed. In other words, when transshipping units from a single source for use in two different periods,

[3]Because of the modification in the cost structure, it doesn't appear that the enhancement of the recently developed $O(n \log n)$ or $O(n)$ algorithms (Federgruen and Tzur [6], Wagelmans, Van Hoesel, and Kolen [26], and Aggarwal and Park [1]) can be used here, in any case this does not affect the complexity of the overall algorithm.

the two paths can no longer be determined independently of each other based on the shortest path of each source/destination combination. In addition, Property 7 is no longer satisfied since supplying node $(2, e_2 - 1)$ through location 1 from the replenishment at node $(2, u_2)$ now incurs an extra fixed transshipment cost, and therefore may not be cheaper than supplying it through node $(2, s_2)$.

On the other hand, a useful property which *is* preserved from the basic model is that a transshipment from location 2 to location 1 may occur in only one situation, that is, when location 2 replenishes in a two-location block. In this case, units that satisfy the demand in the first few periods of the block remain at location 2 until they are depleted; the rest of the units are transshipped immediately to location 1 and return to location 2 in future period(s) in consolidated transshipments.

In this section we again use a modified Wagner–Whitin problem in calculating the cost of the block $\boldsymbol{(s_1, s_2) \rightarrow (e_1, e_2)}$ which may contain holes. This problem is very similar to the modified Wagner–Whitin problem introduced in Section 5.1. The difference is that now, in addition to a *single* node which may receive supply from the initial replenishment, we have to consider a *group* of nodes which may receive supply from the initial replenishment. Because of Corollary 1 this supply must be enough to supply a whole number of periods. Thus, we consider Wagner–Whitin basic blocks within the block $\boldsymbol{(s_1, s_2) \rightarrow (e_1, e_2)}$ which consist of nodes $(2, u_2)$ through $(2, v_2 - 1)$ and are denoted (u_2, v_2), $s_2 \leq u_2 \leq v_2 \leq e_2$. Most of these Wagner–Whitin basic blocks may receive supply from either the initial replenishment of the basic block or through a special replenishment associated with a hole of the block. For the same reason we did not consider holes which included location 2 in period s_2 or in period $e_2 - 1$, we again do not consider holes which include these periods. Thus, when $u_2 = s_2$ or $v_2 - 1 = e_2 - 1$ the Wagner–Whitin basic block must receive its supply from the initial replenishment. In all other cases we must decide whether the Wagner–Whitin basic block receives its supply from initial replenishment or from the outside supplier, whichever is less costly. Moreover, since, with the presence of fixed transshipment costs, transshipment from location 1 to location 2 may or may not be consolidated, nodes $(2, s_2)$ and $(2, e_2 - 1)$ must be included in the modified Wagner–Whitin problem.

We now show how the analysis and cost calculations differ from the basic model with the existence of fixed transshipment costs. To do that, we need to introduce some more definitions and notations.

A_{ij}	fixed cost incurred whenever a transshipment is made from location i to location j;
$I_p^{s_p}(u_2; v_2)$	the cost of satisfying the demand of all the nodes from u_2 to $v_2 - 1$ at location 2 using stock from location p in period s_p, excluding any fixed transshipment cost whenever period u_2 coincides with period s_p;
$\hat{M}_p^{s_p}(u_2; v_2)$	the cost of the modified Wagner–Whitin basic block (u_2, v_2) when location p replenishes in period s_p;
$\widehat{WW}_p^{s_p}(s_2, e_2 - 1)$	the solution to the modified Wagner–Whitin problem at location 2 for periods $s_2, s_2 + 1, \ldots, e_2 - 1$ when location p replenishes in period s_p.

The definition of $I_p^{s_p}(u_2; v_2)$ is similar in spirit to the definition of $H(u_2; v_2)$, except for the source of the items. $I_p^{s_p}(u_2; v_2)$ can be represented mathematically as

$$I_p^{s_p}(u_2; v_2) = A_{12} 1_{(u_2 \neq s_p)} + SP(p, s_p; 2, u_2) \sum_{t=u_2}^{v_2-1} d_{2t} + \sum_{t=u_2+1}^{v_2-1} (t - u_2) h_2 d_{2t},$$

where $1_{(\text{relation})}$ is equal to 1 if the relation is true and 0 otherwise.

The reason for the indicator function (or equivalently, the exception in the definition) is that the fixed transshipment cost from location 1 to location 2 should not be charged in two situations. The first is when location 2 is replenishing; in this case, if $u_2 = s_p$, the units are not transshipped from location 1 to location 2. In contrast, if $u_2 \neq s_p$, the units will be transshipped to location 1, held there, and transshipped back to location 2 at a future time. The second is when location 1 is replenishing; in this case the units *are* actually transshipped from location 1 to location 2, but in this case we know that there will invariably be a transshipment at this time. In this case, we account for the fixed transshipment cost, A_{12}, when developing the cost of the basic block of the dynamic transshipment problem [see Eq. (2)].

The other parts of the equation are clear. The second term represents the cost of delivering the material from the initial replenishment to node $(2, u_2)$. The third term represents the cost of holding the material at location 2 through period $v_2 - 1$.

Now the cost of the modified Wagner–Whitin basic block can be written as

$$\hat{M}_p^{s_p}(u_2; v_2) = \begin{cases} I_p^{s_p}(u_2; v_2) & \text{if } u_2 = s_2 \text{ or } v_2 = e_2, \\ \min(I_p^{s_p}(u_2; v_2), H(u_2; v_2)) & \text{otherwise.} \end{cases}$$

With these Wagner–Whitin basic block costs, the value of $\widehat{WW}_1^{s_1}(s_2, e_2 - 1)$ is determined by applying the standard Wagner–Whitin algorithm.

The expression for the cost of the entire dynamic transshipment problem block depends on which of the locations is replenishing. If $s_1 < s_2$, then we know that location 1 is replenishing. In this case, the cost of the block is

$$M_1(s_1, s_2; e_1, e_2) = K_1 + \sum_{t=s_1}^{e_1-1} SP(1, s_1; 1, t)d_{1t} + A_{12} + \widehat{WW}_1^{s_1}(s_2, e_2 - 1). \tag{2}$$

The first term represents the fixed replenishment cost for the block, the second term the cost of satisfying all the demand at location 1, the third term the fixed transshipment cost in period s_2 which was not accounted for in the modified Wagner–Whitin basic blocks, and the fourth term the cost of satisfying all the demand at location 2.

If $s_1 = s_2$ (as explained in Property 9, it is impossible for $s_2 < s_1$), then we must compare the costs of location 1 and location 2 being the replenishing location. The cost for location 1 being the replenishing location is given by Eq. (2). If location 2 is the replenishing location, then the cost is as follows:

$$M_2(s_1, s_2; e_1, e_2) = K_2 + A_{21} + \sum_{t=s_1}^{e_1-1} SP(2, s_2; 1, t)d_{1t} + \widehat{WW}_2^{s_2}(s_2, e_2 - 1).$$

The first term represents the fixed replenishment cost for the block, the second term the fixed cost of transshipping from location 2 to location 1 in period s_2, the third term the cost of satisfying all the demand at location 1, and the fourth term the cost of satisfying all the demand at location 2. Finally,

$$M(s_1, s_2; e_1, e_2) = \min(M_1(s_1, s_2; e_1, e_2), M_2(s_1, s_2; e_1, e_2)).$$

The algorithm to calculate the cost of a block remains mostly unchanged. One required modification is to remove the first clause of the if statement of Step 1 of the algorithm, since it refers to Property 8 which no longer holds; in the rest of the algorithm the expressions for the various cases have to be replaced by their new formulas.

The complexity of finding the optimal solution to the dynamic transshipment problem with fixed transshipment costs is still $O(T^4)$. The complexity of calculating all $\widehat{WW}_p^{s_p}(s_2, e_2 - 1)$ is $O(T^4)$; the explanation parallels the explanation for $WW^{s_1}(s_2 + 1, e_2 - 2)$ in Section 5.2. The precalculation of the quantities $I_p^{s_p}(u_2; v_2), H(u_2; v_2)$, and $\hat{M}_p^{s_p}(u_2; v_2)$ requires $O(T^3)$ time. Since the first clause of the if statement of Step 1 of the algorithm must be removed, there will be more basic blocks, but the analysis of the complexity in Section 5.2 did not use this fact.

6. 2. Joint Replenishment Costs

Up to now we have considered only two possible replenishment modes, replenish at location 1 and replenish at location 2. In this section we consider a third possibility, replenishing jointly at locations 1 and 2. Even though our model has considered the possibility of replenishing both at location 1 and location 2 at the same time (i.e., in two separate basic blocks), this does not address the fact that there are generally savings involved in replenishment coordination; this means that fixed joint replenishment costs are generally less than the sum of the two individual fixed replenishment costs. In this section we describe how to incorporate into our model a joint replenishment cost of K_{12}. While we expect the conditions $\max(K_1, K_2) \leq K_{12} \leq K_1 + K_2$ to be satisfied, this is not technically required for the analysis that follows.

The definition of a block is unaffected by the presence of joint replenishment costs, but we must add to the definition of a two-location basic block the following possibility: When $s_1 = s_2$, a replenishment occurs both at location 1 and at location 2 which is used to satisfy the demand for location 1 in periods $s_1, \ldots, e_1 - 1$ and for location 2 in periods $s_2, \ldots, e_2 - 1$ except possibly for holes.

The proof of Theorem 2 must be modified to consider a third case for the "latest replenishment," when the latest replenishment is a joint replenishment:

> If the latest replenishment is a joint replenishment, let period t be the period at which the joint replenishment occurs. Applying Corollary 1 at nodes $(1, t)$ and $(2, t)$ (a cut set in the network) we see that the block $\boldsymbol{(s_1, s_2) \to (e_1, e_2)}$ can be split into two disjoint blocks [namely, $\boldsymbol{(s_1, s_2) \to (t, t)}$ and $\boldsymbol{(t, t) \to (e_1, e_2)}$], a contradiction to the maximality assumption.

The rest of Theorem 2 and its proof remain unchanged.

The next issue is the cost of a basic block in which a joint replenishment is made. In order to demonstrate how to calculate this cost we present the following properties of a basic block in which a joint replenishment occurs.

PROPERTY 10: An item purchased at location 2 during a joint replenishment will not be used to satisfy demand at location 1. The reason is that it is cheaper (because of the holding cost differential and because no transshipment is necessary) to satisfy location 1's demand through location 1.

PROPERTY 11: There are no holes at location 1 for the same reason given in Property 5.

PROPERTY 12: If $e_1 < e_2$, then there are no transshipments within the basic block. This is known from Property 10 and the fact that all of the demand at a location 2 is satisfied through the initial replenishment at location 2, as a result of Properties 2 and 4 and Corollary 1 applied at node $(1, e_2 - 1)$. That is, the demand at node $(2, e_2 - 1)$ is met through the replenishment at node $(2, s_2)$ and is held in inventory only at location 2. Note that this also implies that there are no holes.

PROPERTY 13: If $e_1 \geq e_2$, some of the demand at location 2 may be satisfied from location 1, but only after the units of location 2's initial replenishment have been consumed. From this point on location 2 may or may not contain holes.

If there are no transshipments within the basic block (as is the case when $e_1 < e_2$), then the cost of the basic block is the sum of the cost of two separate single-location blocks, except that a different fixed replenishment cost is used, namely,

$$M_{12}^{NT}(s_1, s_2; e_1, e_2) = M_1(s_1, s_2; e_1, s_2) + M_2(s_1, s_2; s_1, e_2) - K_1 - K_2 + K_{12}.$$

If, however, $e_1 \geq e_2$, we can again have no transshipments, but we must also consider the case that we do have transshipments. In this latter case we must take the minimum over all possible periods, k, at which the initial replenishment at location 2 can run out.

$$\begin{aligned} &M_{12}(s_1, s_2; e_1, e_2) \\ &= \min\{M_{12}^{NT}(s_1, s_2; e_1, e_2), \min_{s_2 \leq k < e_2 - 1} [M_2(s_1, s_2; s_1, k+1) + M_1(s_1, s_2; e_1, s_2) \\ &\quad + WW^{s_1}(k+1, e_2 - 2) + SP(1, s_1; 2, e_2 - 1)d_{2, e_2 - 1} - K_1 - K_2 + K_{12}]\}. \end{aligned}$$

The possibility of a joint replenishment considered here will not affect the far majority of the arcs in the block network. In fact the only affected arcs are the ones originating on the diagonal, i.e., nodes of the form $(\boldsymbol{s_1}, \boldsymbol{s_1})$. Presently these arc costs are determined by considering replenishment at location 2 (if $e_1 < e_2$) or by taking the cheaper of the two alternatives of replenishing at location 1 or replenishing at location 2. In addition to both of these, we need to consider the possibility of a joint replenishment.

The complexity of the algorithm is unaffected. There are $O(T^3)$ joint replenishment possibilities, combinations of $s_1 = s_2, e_1$, and e_2, each of which requires $O(T)$ time to calculate its cost. The $O(T)$ time factor comes from the need to compare all possible runout times of the initial replenishment at location 2. Thus calculating the cost of all the joint replenishment possibilities requires $O(T^4)$ time, the same complexity as the algorithm as a whole.

7. DISCUSSION AND CONCLUSIONS

Transshipments are often used as a mechanism to improve supply chain management. This work has made a first step towards studying transshipments in a dynamic as well as deterministic environment, where the use of transshipments as emergency source of supply is not applicable. In fact, we explored two (previously not mentioned) reasons for using transshipments, namely, the presence of fixed replenishment costs and a holding cost differential between locations. Finally, we have considered fixed transshipment costs. In this way, this work extends the extensive literature available on dynamic deterministic inventory models, to the growing practice of transshipments. The simple and intuitive structure and properties associated with the solution of the dynamic transshipment problem identified in this work can be understood and implemented by practitioners. For example, practitioners can use the fact that each location in each period has a single source of supply and/or the fact that replenishment quantities are the sum of several demand points to aid in planning.

Two structures of fixed replenishment costs were considered: In the first a fixed cost is incurred in a given period for each replenishing location; in the second a joint replenishment cost is incurred when both locations replenish. In fact, the latter cost structure is the same as in the well known Joint Replenishment Problem (JRP) (see, e.g., Federgruen and Tzur [7]), but we allow transshipments between the locations. Thus, by appropriately defining K_1, K_2, and K_{12} and by setting $c_{12} = c_{21} = \infty$, our algorithm can be used to solve the two location JRP.

We developed a polynomial $[O(T^4)]$ time algorithm to solve the dynamic transshipment problem. This algorithm is based on our extensive analysis of the problem which includes the definition of blocks, holes, and basic blocks and the demonstration that an optimal solution consists of a series of basic blocks. The identification of the basic blocks which make up the optimal solution is made by applying a shortest path algorithm on a properly defined network. Our analysis was performed for two locations; this helped to simplify the analysis, which in turn enabled us to obtain deep insights into the problem. We expect that further work on dynamic deterministic transshipment problems, with an arbitrary number of locations, would most likely be based on the two-location results presented here.

A topic closely related to transshipments is substitutions. Even though transshipments and substitutions are related, they are two distinct recourses. A transshipment concerns the same item located at two different locations, while a substitution concerns two different items located at the same location. One may think that a substitution in which item A replaces item B can be modeled as a transshipment from location A to location B; in fact, with certain inventory models the two problems can be represented by one unified model, defining the transshipment cost in one direction as infinite. However, the difference between the problems becomes clear when one considers the question of whether a substitution can be made in a time period before the demand occurs, as can be done with transshipments. Whereas the motivation to transship before the demand occurs would be to save holding costs, this is meaningless in the substitution problem where the substituting item never really changes to become the other (cheaper) item. Thus solving the substitution problem would require a different analysis, but we feel that we have laid the foundation for it in this paper.

REFERENCES

[1] A. Aggarwal and J.K. Park, Improved algorithms for economic lot size problems, Oper Res 41 (1993), 549–571.

[2] S.G. Allen, Redistribution of total stock over several user locations, Nav Res Logistics Quart 5 (1958), 337–345.

[3] S.G. Allen, A redistribution model with set up charge, Manage Sci 8 (1962), 99–108.

[4] T.W. Archibald, S.A.E. Sassen, and L.C. Thomas, An optimal policy for a two depot inventory problem with stock transfer, Manage Sci 43 (1997), 173–183.

[5] E.V. Denardo, Dynamic programming models and applications, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[6] A. Federgruen and M. Tzur, A simple forward algorithm to solve general dynamic lot sizing models with n periods in $O(n \log n)$ or $O(n)$ time, Manage Sci 37 (1991), 909–925.

[7] A. Federgruen and M. Tzur, The joint replenishment problem with time-varying parameters: Efficient, asymptotic and ϵ-optimal solutions, Oper Res 42 (1994), 1067–1086.

[8] A. Federgruen and M. Tzur, Time-partitioning heuristics: Application to one warehouse, multi-item, multi-retailer lot-sizing problems, Nav Res Logistics 46 (1999), 463–486.

[9] D. Gross, "Centralized inventory control in multilocation supply systems," Multistage inventory models and techniques, H. Scarf, D. Gilford, and M. Shelly (Editors), Stanford University Press, Stanford, CA, 1963, pp. 47–84.

[10] Y.T. Herer and A. Rashit, Policies in a general two-location infinite horizon inventory system with lateral stock transshipments, Technical report, Technion—Israel Institute of Technology, Israel, 1997.

[11] Y.T. Herer and A. Rashit, Lateral stock transshipments in a two-location inventory system with fixed ordering costs, Nav Res Logistics 46 (1999), 525–547.

[12] Y.T. Herer and M. Tzur, Optimal and Heuristic Algorithms for the multi-location dynamic transshipment problem with fixed transshipment costs, Technical report, Department of Industrial Engineering, Tel Aviv University, Israel, 1999.

[13] D. Joneja, The joint replenishment problem: New heuristics and worst case performance bounds, Oper Res 38 (1990), 711–723.

[14] U.S. Karmarkar, Convex stochastic programming and multilocation inventory problem, Nav Res Logistics Quart 26 (1979), 1–19.

[15] U.S. Karmarkar, The multiperiod multilocation inventory problem, Oper Res 29 (1981), 215–228.

[16] U.S. Karmarkar, The multilocation multiperiod inventory problem: Bounds and approximations, Manage Sci 33 (1987), 86–94.

[17] U.S. Karmarkar and N. Patel, The one-period, n-location distribution problem, Nav Res Logistics Quart 24 (1977), 559–575.

[18] K.S. Krishnan and V.R.K. Rao, Inventory control in n warehouses, J Ind Eng 16 (1965), 212–215.

[19] S.F. Love, A facilities in series inventory model with nested schedules, Manage Sci 18 (1972), 327–338.

[20] J. Maes and L.N. Van Wassenhove, Multi-item single-level capacitated dynamic lot-sizing heuristics: A general review, J Oper Res Soc 39 (1988), 991–1004.

[21] B.A. Pasternack and Z. Drezner, Optimal inventory policies for substitutable commodities with stochastic demand, Nav Res Logistics Quart 38 (1991), 221–240.

[22] L.W. Robinson, Optimal and approximate policies in multiperiod, multilocation inventory models with transshipments, Oper Res 38 (1990), 278–295.

[23] G. Stalk, P. Evans, and L.E. Shulman, Competing on capabilities: The new rules of corporate strategy, Harvard Bus Rev 70 (1992), 57–69.

[24] G. Tagaras, Effects of pooling on the optimization and service levels of two-location inventory systems, IIE Trans 21 (1989), 250–257.

[25] G. Tagaras and M.A. Cohen, Pooling in two-location inventory systems with non-negligible replenishment lead times, Manage Sci 38 (1992), 1067–1083.

[26] A.P.M. Wagelmans, C.P.M. Van Hoesel, and A. Kolen, Economic lot-sizing: An $o(n \log n)$ algorithm that runs in linear time in the wagner-whitin case, Oper Res 40 (1992), S145–S156.

[27] H.M. Wagner and T.M. Whitin, Dynamic version of the economic lot size model, Manage Sci 5 (1958), 89–96.

[28] W.I. Zangwill, A backlogging model and a multi-echelon model of a dynamic economic lot size production system—a network approach, Manage Sci 15 (1969), 506–527.