Scheduling Trains and Containers with Due Dates and Dynamic Arrivals

Candace A. Yano • Alexandra M. Newman
Department of Industrial Engineering and Operations Research, University of California, Berkeley, California 94720-1777
Division of Economics and Business, Colorado School of Mines, Golden, Colorado 80401

We consider the problem of scheduling trains and containers (or trucks and pallets) between a depot and a destination. Goods arrive at the depot dynamically over time and have distinct due dates at the destination. There is a fixed-charge transportation cost for each vehicle, and each vehicle has the same capacity. The cost of holding goods may differ between the depot and the destination. The goal is to minimize the sum of transportation and holding costs.

For the case in which all goods have the same holding costs, we consider two variations: one in which the holding cost at the destination is less than that at the origin, and one in which the relationship is reversed. For the first variation, we derive properties of the optimal solution which provide the basis for an $O(T^2)$ solution procedure. For the second variation, we introduce a new definition of a regeneration state, derive strong characterizations of the shipment schedule within a regeneration interval, and develop an $O(T^4)$ procedure.

We also analyze two multi-item scenarios. In the first, for each item, the holding cost at the origin is less than that at the destination; in the second, the relationship is reversed for all items. We generalize several of the structural results for the single-item problem to the corresponding multi-item case. We also show that the optimal vehicle schedule can be obtained by solving a related single-item problem in which the item demands are aggregated in a particular way. The optimal assignment of customer orders to vehicles can then be found by solving a linear program.

Introduction

Our work is motivated by the problem of scheduling trains or trucks outbound from a single depot and assigning goods to these vehicles, with the objective of achieving on-time delivery at minimum cost. We consider a finite time horizon during which customer orders dynamically become available at the origin, and have different due dates at their destinations.

We assume that the demand, distinguished by origin, arrival date at the origin, destination, and due date at the destination, is known in advance, or can be forecasted accurately enough over the time horizon for planning purposes. We consider *direct* shipments between a depot and each of the various destinations. Assuming there is no dependence among the destinations, we can decompose the problem by destination.

Although our initial motivation was derived from train scheduling applications, a similar situation also arises when finished goods must be transported by truck from a factory to distant markets or break-bulk warehouses. Customer orders with distinct due dates and destined for a particular market are produced at the factory and become available for shipment over time. The manufacturer faces the decision of when to dispatch trucks and which customer orders to assign to each truck. Of course, this problem would

0041-1655/01/3502/0181$05.00
1526-5447 electronic ISSN

be simplified if goods destined for a particular market were manufactured close together in time. However, there are often competing and/or overriding considerations in establishing the production schedule, thus necessitating the solution of the transportation scheduling problem.

As is typical in rail, trucking, and sea transport operations, there is a (nearly) fixed charge associated with the direct movement of the vehicle or vessel from the origin to the destination which includes the cost of labor for the movement of the vehicle and any other costs that are not volume dependent, including the portion of fuel and maintenance costs that do not depend on the volume of goods in the vehicle. Furthermore, the capacity of the transport vehicle is known. Because we are addressing a short-horizon problem, we assume that the fixed cost of transportation per vehicle is constant over time and independent of the number of vehicles sent.

In the vast majority of situations, total volume-dependent costs will differ only slightly, if at all, as the shipping schedule changes. For example, the incremental cost for fuel associated with carrying a given weight or volume of goods via truck from origin to destination, above and beyond that required to operate the truck(s) empty, would be roughly the same irrespective of the allocation of goods among trucks. It would be more efficient, of course, to ship all of the goods on as few trucks as possible, provided their capacity is not exceeded, and this aspect is captured by the fixed-charge transportation cost. We assume that total volume-dependent costs are insensitive to the details of the shipping schedule. Similarly, in many settings the cost of holding the goods at the origin differs little from the cost at the destination, but they may differ widely in other settings. For this reason, we also analyze situations in which the holding costs differ among items and between locations.

Because penalties for tardy delivery may be substantial, our primary goal is to minimize the sum of fixed-charge transportation costs while delivering the goods on time. We will, however, consider the more general problem in which the holding costs may differ between the origin and destination to reflect the relative cost of storage space, as well as the extent to which the customer is willing to accept early shipments. In some instances, the customer may wish to receive the shipment as early as possible, while in other instances a just-in-time shipping schedule may be preferable. The former situation can be modeled by imposing a higher holding cost at the origin than at the destination, while the latter scenario is represented by the reverse relationship. We note that any opportunity costs of capital associated with holding the goods in transit are usually borne by the shipper or consignee, and not by the transporter. The transporter does, however, bear the opportunity cost of having equipment, such as trailers and containers, unavailable for use.

We analyze the case in which goods have homogeneous holding costs, where the holding cost is more expensive at the destination than at the origin. The opposite holding cost relationship can be addressed by the same approach using an appropriate transformation of the problem. We also model versions of the problem with item-dependent holding costs. We present results for the case in which, for each item, the holding cost at the destination is higher than that at the origin. If the reverse relationship holds for all items, a transformation similar to that mentioned above can be used to solve the problem. We do not consider the most general case in which it is less expensive to hold some items at the origin than at the destination, and the reverse holds for the other items. The latter situation may arise, for example, if the freight has different temperature requirements; storage costs will depend upon the ambient weather.

Our problem is similar to other capacitated fixed-charge network flow problems but contains three important complicating features: (i) not all goods are available at the beginning of the horizon, i.e., goods arrive dynamically, (ii) it may be necessary and/or optimal to send more than one vehicle in a given period, i.e., to incur "multiple setups," and (iii) goods are not homogeneous and have distinct due dates at the destination, and thus must be treated as distinct customer orders. We assume, however, that the goods are homogeneous in their use of transport capacity.

Although there is a large body of literature on lot-sizing problems with a fixed-charge structure, little work has been done on problems in which there is a

Table 1 Comparison of Our Model to Related Literature

Paper	Nature of Demand	No. of Items	Dynamic Arrivals?	Multiple Setups?
Florian & Klein (1971)	deterministic, dynamic	one	no	no
Lippman (1969)	deterministic, dynamic	one	no	yes
Lippman (1971)	constant	one	no	yes
Iwaniec (1979)	stochastic, dynamic	one	no	yes
Lee (1989)	deterministic, dynamic	one	no	yes
Ben-Kheder (1990)	deterministic, dynamic	multiple	no	yes
Our paper	deterministic, dynamic	one or more	yes	yes

capacity associated with each fixed charge incurred. Florian and Klein (1971) consider a single-item capacitated lot-sizing problem where all materials are available for processing (or, alternately, for shipment) at the beginning of the horizon, and there is a fixed charge for each production run with fixed capacity. At most, one production run is allowed in each period. They show that when capacity is constant over time, the optimal solution has a production level equal either to zero or to the capacity in all periods except at most one between two consecutive regeneration points. (For Florian and Klein's problem, a regeneration point is defined as a point in time with no on-hand inventory.) This result provides the basis for an efficient solution procedure based on an underlying shortest path problem. Lippman (1969) addresses the single-product problem with static arrivals and multiple setups. He derives several properties of the optimal solution and develops an $O(T^3)$ algorithm for the problem. We relate his characterization of the optimal solution to our structural results as our discussion proceeds. In a later paper, Lippman (1971) treats a continuous-time, constant-demand version of the problem. Iwaniec (1979) studies a base stock policy, rounded up to the next full vehicle, in the context of a discrete-time version of the problem with stochastic demand. Lee (1989) addresses the multiple setup problem where all materials are available for shipment at the beginning of the horizon and there is a separate setup cost per order. He presents an $O(T^4)$ procedure for the problem.

All of the aforementioned articles consider only a single product. The only research of which we are aware that treats the multi-item case (where the items have different holding costs) in Ben-Kheder (1990). He studies the case with dynamic demands under the assumption that the holding cost at the destination is higher than at the origin. He presents a solution procedure with an underlying shortest path network, where the path costs are computed using a branch-and-bound algorithm. Table 1 contrasts our models with others in the literature. Note that none of the models in the literature allows for dynamic arrivals of goods.

The remainder of the paper is organized as follows. In §1, we present a dynamic programming formulation of the problem. In §2, we characterize the optimal solution and present an optimal $O(T^4)$ algorithm for the case in which all goods have the same holding costs, and the holding cost is larger at the destination than at the origin. We also explain how the reverse holding cost relationship can be handled by an appropriate transformation of the network. In §3, we extend these results to allow for items with different holding cost rates. Section 4 concludes the paper with a discussion of the relationship between our results and those for related models with static arrivals.

1. Problem Formulation

Because of the dynamic arrivals and the nonhomogeneity of the goods, we need to distinguish the goods by arrival date at the depot and due date at the destination. For simplicity, we assume that shipment quantities can be treated as if they were continuous. Practically speaking, this means that shipments occur in increments of standard pallet loads for truck travel, or standard container sizes for rail shipments. Assuming that all shipments are in multiples of the standard load size, and the capacity of the transport vehicle is expressed as an integer multiple of the standard load, we can treat the shipment quantities as if they were continuous without loss of generality. We also assume that the transportation times are deterministic and constant over time, and therefore, without loss of generality, can be treated as if they were instantaneous by an appropriate reindexing of the time periods.

We formulate the problem as a dynamic program. We define the state of the system by a pair of vectors. One vector represents the containers available to be shipped, and contains as many entries as there are remaining due dates in the horizon. The other vector represents the inventory at the destination with an entry corresponding to each remaining due date in the horizon. For ease of exposition, in the remainder of the paper, we use the rail terminology of *trains* and *containers* to represent the vehicle and the unit shipment load, respectively. Let us define the following notation:

$t =$ time period index, $t = 1, \ldots, T$

$S =$ fixed charge per shipment

$C =$ capacity of each train (expressed as number of containers)

$h_o =$ holding cost for one container for one period at the origin

$h_d =$ holding cost for one container for one period at the destination

$D_t(d) =$ quantity of containers arriving *at the origin* in period t that are due at the destination in period d, $d \geq t$

$\mathbf{D_t} = (D_t(t), D_t(t+1), D_t(t+2), \ldots, D_t(T))$

$\widehat{\mathbf{D}}_\mathbf{t} = (\sum_{u \leq t} D_u(t), 0, \ldots, 0)$, i.e., a vector in which the first element is the total quantity of containers due in period t and the remaining $T-t$ elements are zero

$A_t(d) =$ quantity of containers *available to be shipped from the origin* in period t that are due in period d, $d \geq t$

$\mathbf{A_t} = (A_t(t), A_t(t+1), A_t(t+2), \ldots, A_t(T))$

$I_t(d) =$ inventory of containers due in period d held *at the destination* at the end of period t, $d \geq t$

$\mathbf{I_t} = (I_t(t+1), I_t(t+2), \ldots, I_t(T))$

$x_t(d) =$ number of containers shipped in period t that are due in period d, $d \geq t$

$\mathbf{x_t} = (x_t(t), x_t(t+1), x_t(t+2), \ldots, x_t(T))$; decision (row) vector

$f_t^*(\mathbf{A_t}, \mathbf{I_{t-1}}) =$ the minimum cost for periods $t, \ldots, T$ given container availability vector $\mathbf{A_t}$ and container inventory vector $\mathbf{I_{t-1}}$ if the optimal number of containers, $\mathbf{x_t^*}$, is shipped at time t.

The dynamic programming recursion equations are:

$$\begin{aligned} f_t^*(\mathbf{A_t}, \mathbf{I_{t-1}}) = \min_{\mathbf{x_t}} \Bigg\{ & S \left\lceil \sum_{d \geq t} \frac{x_t(d)}{C} \right\rceil + h_o \sum_{d>t} [A_t(d) - x_t(d)] \\ & + h_d \sum_{d>t} \big[I_{t-1}(d) + x_t(d) \big] \\ & + f_{t+1}^* \big(\mathbf{A_t} + \mathbf{D_{t+1}} - \mathbf{x_t}, \mathbf{I_{t-1}} + \mathbf{x_t} - \widehat{\mathbf{D}}_\mathbf{t} \big) \Bigg\} \end{aligned}$$

where

$$f_{T+1}^*(\cdot) = 0, \tag{1}$$

$$x_t(t) = A_t(t), \ \forall\, t \tag{2}$$

$$x_t(d) \leq A_t(d), \ \forall\, d \geq t \tag{3}$$

$$x_t(d) \geq 0, \ \forall\, d, t \tag{4}$$

Equation (1) is the boundary condition. Constraints (2) and (3) ensure that demand is satisfied on time and that only available containers are shipped, respectively. Constraints (4) ensure nonnegative shipment quantities.

We present a complete analysis for the case in which the holding cost is more expensive at the destination than at the origin. This cost structure provides motivation for sending containers as late as possible while accounting for economies of scale in transportation. In §2, we derive properties of the optimal solution and present an $O(T^4)$ algorithm. Following our analysis, we explain how the case with the opposite holding cost relationship can be treated by a reversal of the multicommodity network in time and space.

2. Analysis and Algorithm for $h_d \geq h_o$

The main economic tradeoff in this problem is between the economies of scale associated with sending full trains and the additional inventory holding cost incurred if containers are shipped early.

To develop an efficient solution procedure for this case, we employ the concept of a *regeneration interval,* as has been used for similar problems. However, our definition of a regeneration state differs significantly from the traditional one. The traditional regeneration state in a lot-sizing setting is defined as a state with no on-hand inventory. In the context of our problem, this would correspond to having nothing remaining to be shipped. This state would rarely occur in our problem due to the economic incentive to ship as late as possible. Using the traditional definition would often lead to regeneration points only at the beginning and end of the horizon, and the solution procedure would degenerate to the enumerative dynamic programming procedure described earlier.

Definition 1. We say that a regeneration occurs at the beginning of period t (equivalently, at the end of period $t-1$) if no containers due in periods $t, t+1, \ldots, T$ have been shipped in periods $1, 2, \ldots, t-1$, i.e., not containers due in period t or later have been shipped before period t.

Such a state clearly defines a point in time such that the problem can be separated into two distinct problems: (i) shipping containers due in periods $1, 2, \ldots, t-1$, and (ii) shipping containers due in periods t, $t+1, \ldots, T$. Thus, this definition provides the same type of decomposition as in earlier models. Our definition of a regeneration interval affords the advantage of permitting multiple regenerations during the horizon, which, in turn, reduces the computational effort required. Note that in addition to any regeneration points that may occur between periods 2 and $T-1$, regenerations occur at the beginning and end of the horizon. Hence, the optimal solution for the entire horizon can be obtained by finding the optimal solution for each potential regeneration interval, and then solving a shortest path problem over the entire horizon to determine the optimal set of regeneration intervals.

2.1. Characteristics of the Optimal Solution

Using our definition of a regeneration interval, we derive several properties of the optimal solution. We note that if $h_d = h_o$, the optimal solution generally is not unique, so alternate schedules considered in our proofs may not be strictly dominant but may instead represent alternate optimal solutions.

Proposition 1. *There exists an optimal solution in which the number of containers sent ahead of schedule in any individual time period is strictly less than C. In other words, items are shipped early only for the purpose of filling up a train either completely or partially.*

Proof. Omitted. A proof appears in Yano and Newman (1998).

Note that Proposition 1 does not necessarily imply that the *cumulative* number of containers shipped early must be less than C. It may be optimal for the cumulative number of containers shipped early to be greater than C, especially when transportation costs dominate holding costs, making it desirable to send trains full or nearly full.

Proposition 2. *There exists an optimal schedule in which, whenever containers are shipped early, they are shipped in increasing order of their due dates among the containers available to be shipped.*

Proof. We briefly sketch the proof here. The proof relies on two observations with respect to feasible solutions: (i) the cost in the *current* period depends only on the total number of containers shipped (assuming that all containers due in the current period are included among the shipped containers), and (ii) the "cost to go" for the *remaining* periods is the same or smaller for an earliest due date (EDD) container shipping schedule than for any non-EDD shipping schedule. Part (i) is self-evident. Part (ii) can be proved by showing that an arbitrary change toward an (EDD) shipping schedule in the *current* period reduces the quantities in the vector of cumulative shipments required to ensure on-time delivery for the *remaining* periods, which, in turn, reduces the cost to go. See Yano and Newman (1998) for a complete proof.

Proposition 3. *There exists an optimal schedule in which, if any containers due in period t are shipped in period $t' < t$, trains sent in periods $t'+1, t'+2, \ldots, t$ are full.*

Proof. Suppose that we have an optimal solution in which we ship containers that are due in period $t' < t$, and that all trains (if any) sent in periods $t'+1$, $t'+2, \ldots, t$ are not full. Then, because these containers are not due until period t, a feasible shipment plan exists in which some or all of these early containers are sent in some period(s) in $t'+1, t'+2, \ldots, t$. Furthermore, because $h_o \geq h_d$, a lower- or equal-cost shipment plan exists in which some or all of these containers are sent in these subsequent periods. Hence, the schedule cannot be optimal, which contradicts our original hypothesis. □

THEOREM 1. *The optimal schedule within a regeneration interval has full trains in every period, except possibly the first.*

PROOF. Let s be the first period in the regeneration interval and τ be the last period in the regeneration interval. Let t be a period such that $s < t \leq \tau$, i.e., an arbitrary period either in the middle of, or at the end of, the regeneration interval. Because the system does not regenerate at the beginning of t by assumption, we must have:

$$\sum_{d\geq t} I_{t-1}(d) > 0$$

by our definition of a regeneration interval. Suppose that if any trains are sent in period t, they are not full. Then it would be possible to delay some or all of the earlier-sent containers with due date t or later, and thereby reduce the holding costs without increasing the transportation cost. Thus, if any trains are sent in period t, they must be sent full. Because t is an arbitrary period between $s+1$ and τ, the result follows for all periods except the first. □

Theorem 1 provides a stronger characterization of shipment quantities during a regeneration interval than that given by Lippman (1969), and it does so for the more general case in which dynamic arrivals are permitted. It also generalizes the result of Florian and Klein (1971) to the case of multiple setups.

OBSERVATION 1. From our definition of a regeneration interval, the first period of the regeneration interval satisfies:

$$\sum_{d\geq s} I_{s-1}(d) = 0,$$

i.e., there are no containers due in period s or later that have been shipped prior to period s. Thus, we may have a less-than-full train.

The following corollary generalizes a result of Lippman (1969) to the case of dynamic arrivals.

COROLLARY 1. *There exists an optimal solution in which for all t,*

$$\left[\sum_{d\geq t} I_{t-1}(d)\right] * \left[\sum_{d\geq t} x_t(d) \bmod C\right] = 0.$$

PROOF. The result follows from Theorem 1 and Observation 1. □

2.2. Test for Feasibility of a Period as the End of a Regeneration Interval

The results in the previous subsection can be used to determine whether a period is eligible to be the last period in a regeneration interval, and could substantially reduce computation times in practical applications. We first describe the logic underlying the procedure, and then describe the procedure.

Recall that in the last period of any regeneration interval, τ, the trains must be full. Because it is optimal to delay containers as much as possible, the optimal shipment quantity in period τ is:

$$x^*_\tau(\tau) = \left\lfloor \frac{\sum_{u\leq\tau} D_u(\tau)}{C} \right\rfloor * C.$$

The balance of period τ's demand is given as follows:

$$\sum_{u\leq\tau} D_u(\tau) - \left\lfloor \frac{\sum_{u\leq\tau} D_u(\tau)}{C} \right\rfloor * C = \sum_{u\leq\tau} D_u(\tau) \bmod C.$$

This balance must be shipped early, that is, in period $\tau-1$ or earlier. If this quantity is not available at the beginning of period $\tau-1$ because all or nearly all of the demand for period τ arrives in τ, then we cannot have all trains full in period τ *and* have some of its demand shipped early.

If

$$\sum_{u=1}^{\tau-1} D_u(\tau) < \sum_{u=1}^{\tau} D_u(\tau) - \left\lfloor \frac{\sum_{u=1}^{\tau} D_u(\tau)}{C} \right\rfloor * C,$$

τ cannot be the last period in any regeneration interval, because we cannot send all trains full in period τ.

In addition to the infeasibilities noted above, some regeneration intervals may not be feasible because container arrivals may not allow full-train shipments according to the pattern described in Theorem 1. Although we can test for this in advance, it also can be done as efficiently within the context of the procedure for finding the optimal schedule within a regeneration interval. If no feasible schedule exists, we assign an infinite cost to the corresponding arc in the shortest path network.

2.3. Algorithm for Determining the Optimal Schedule: $h_d \geq h_o$

For any potentially feasible regeneration interval, the optimal schedule can be constructed as follows:

Step 1. Let $[s, \tau]$ denote a regeneration interval, where s is the first period in the interval, and τ is the last period in the interval. Compute the quantity (if any), L, to be sent in the less-than-full train in the first period of the regeneration interval:

$$L = \sum_{d=s}^{\tau} \sum_{u \leq d} D_u(d) \bmod C.$$

Step 2. In period s, send $\lceil \frac{\sum_{u \leq s} D_u(s)}{C} \rceil$ trains if $(\sum_{u \leq s} D_u(s) \bmod C) \leq L$. Otherwise, send $\lceil \frac{\sum_{u \leq s} D_u(s)}{C} \rceil + 1$ train(s). Fill all but one train; the last train will be filled with $L < C$ containers. Containers should be assigned to the trains in increasing order of due date. If there are insufficient containers available, terminate the algorithm. The regeneration interval is infeasible. Otherwise, go to Step 3.

Step 3. For $t = s+1$ to τ:

(a) Update $\mathbf{A_t}$.

(b) Send $\lceil \frac{A_t(t)}{C} \rceil$ trains, filling them with containers in increasing order of due date. If there are insufficient containers available to ship the trains full, terminate the algorithm. The regeneration interval is infeasible. Otherwise, continue Step 3 (incrementing t).

This procedure has a complexity of $O(T^2)$ and must be performed for each possible regeneration interval, of which there are $O(T^2)$. Hence, the computation of arc costs for the shortest path problem is $O(T^4)$. The shortest path problem itself is $O(T^2)$. Thus, the overall procedure has complexity $O(T^4)$. Note that in computing the cost of each arc in the shortest path network, one needs to include the origin holding cost for containers that are due during the regeneration interval but arrive at the origin before the beginning of the regeneration interval.

If few containers arrive early, many regeneration intervals will be infeasible. On the other hand, for the static case in which all containers are available to be shipped at the beginning of the horizon, all regeneration intervals are feasible because the availability of goods does not limit the shipments in Steps 2 and 3(b) above.

2.4. Modification for $h_d < h_o$

We can solve the case in which holding costs are higher at the origin than at the destination by reversing the network in time and space. That is, we treat the problem as if goods arrive at the destination at their respective due dates and are due at their origins on their respective arrival dates. Trains and containers flow from destination to origin and backward in time. Costs on all arcs remain the same. The solution procedure described in the previous section can be applied to this problem.

When solving the problem "forward" rather than "backward," the regeneration state is defined as a state in which there are no further containers available to be shipped. All trains are full, except possibly in the *last* period of the regeneration interval.

3. Multiple Items with Different Holding Costs

A great deal of literature treats multi-item lot-sizing problems with either fixed-charge joint replenishment costs or a single capacity constraint. However, little research has been done on multi-item problems with multiple setup costs. The only work of which we are aware is that of Ben-Kheder (1990), who considers the case of $h_{d_i} \geq h_{o_i}$ for all items i, given static arrivals. He employs the traditional definition of a regeneration state and characterizes properties of the shipment schedule within a regeneration interval, such as the timing of full and partially full trains within the regeneration interval. From this, he develops an $O(T^3)$ solution procedure.

We generalize our results in §2 to the case of multiple items, each with different holding costs. The dynamic programming formulation is essentially the same as that given earlier, except that items are also distinguished by holding cost. Thus, the state and decision vectors in the single-item case become a state matrix and decision matrix, respectively, in the multi-item case. We first treat the case in which $h_{d_i} \geq h_{o_i}$ for all items i, and then explain how to adapt our results to the case in which $h_{d_i} < h_{o_i}$ for all items i. In most practical settings, the goods being shipped are owned by either the shipper or consignee, so the value of the goods does not play an important role in the determination of holding costs borne by the transporter. However, the holding costs incurred by the transporter may be location related. For example, the cost of electricity may be more expensive at one location than another, leading to higher costs for holding refrigerated containers. For such realistic situations, we show that within a regeneration interval, the following characteristics developed for the single-item case do not change when multiple items are considered: (i) the structure of the train schedule with respect to full versus partial loads; and (ii) the optimal solution regarding how many trains to send in each period. We also show that given (i) and (ii), the problem of allocating containers to trains can be solved using linear programming. In the interest of brevity, we state some results without detailed proofs; in all of these instances, the logic follows in a straightforward manner from that of the single-item case.

The case in which $h_{d_i} \geq h_{o_i}$ for some items and $h_{d_i} < h_{o_i}$ for other items proves to be very difficult because the construction of an optimal train schedule is much more complex. We elaborate on these implications in the concluding section.

3.1. Multiple Items with $h_{d_i} \geq h_{o_i} \forall i$

Although we do not use these results directly, we note that Propositions 1 through 3 and Theorem 1 all generalize to the case of multiple items. If the holding cost at the destination is greater than that at the origin for each item, the proofs can be constructed in a similar way.

We now extend the definition of a regeneration state:

Definition 2. We say that a regeneration occurs at the beginning of period t (equivalently, at the end of period $t-1$) if no containers of any type due in period $t, t+1, \ldots, T$ have been shipped in periods $1, 2, \ldots, t-1$, i.e., no containers of any type due in period t or later have been shipped before period t.

We first construct a solution consisting of a train schedule derived from an adaptation of the single-item solution procedure combined with an optimal allocation of containers for this train schedule obtained by solving a linear program. We first show how to construct this solution, then demonstrate that it is optimal for the multi-item problem.

Consider a variant of the multi-item problem in which, for each (arrival date, due date) pair, we aggregate demands across holding costs. For this aggregate item, choose an arbitrary positive holding cost at the destination and assume, without loss of generality, that the inventory holding cost at the origin is zero. We now use the algorithm described in §2.3 to find the optimal solution for this revised problem within the regeneration interval. From the solution to the revised problem, we can construct a feasible schedule for the original problem with the same train schedule and the same aggregate shipment quantities. We show how to construct an optimal detailed, multi-item schedule from this aggregate schedule.

Consider an assignment of (available-to-ship) containers in which containers are assigned to trains starting in the first period of the regeneration interval in increasing order of their due dates. Such a schedule would be comparable to the single-item schedule in that differences in holding costs among the items are ignored. Clearly, we can improve the solution by maintaining the same train schedule and modifying the shipment of containers to minimize inventory holding costs subject to satisfying on-time delivery. This can be done by solving a linear program for each regeneration interval.

Define:

$s =$ first period in the regeneration interval
$\tau =$ last period in the regeneration interval
$z_t =$ number of trains scheduled in period t

$$L = \sum_i \sum_{d=s}^{\tau} \sum_{u \leq d} D_{iu}(d) \bmod C$$

$$N = \left\lfloor \frac{\sum_i \sum_{u \leq s} D_{iu}(s)}{C} \right\rfloor \text{ if } \sum_i \sum_{u \leq s} D_{iu}(s) \bmod C \leq L$$
$$= \left\lceil \frac{\sum_i \sum_{u \leq s} D_{iu}(s)}{C} \right\rceil \text{ otherwise.}$$

Note that N represents the number of full trainloads and L is the fractional trainload sent in the first period of the regeneration interval. The remaining notation parallels that of the single-item case, with i denoting the item (or container) type.

Omitting the sunk inventory holding costs from the objective function, the problem for fixed z_t, $t = s, \ldots, \tau$, can be formulated as:

$$\min \sum_i (h_{d_i} - h_{o_i}) \sum_{t=s}^{\tau} \sum_{d>t} x_{it}(d)$$

subject to:

$$\sum_i \sum_{d \geq s} x_{is}(d) = N * C + L, \tag{5}$$

$$\sum_i \sum_{u=s}^{t} \sum_{d \geq u} x_{iu}(d) = N * C + L + C * \sum_{u=s+1}^{t} z_u, \quad t \in [s+1, \tau] \tag{6}$$

$$\sum_{u \leq t} x_{iu}(d) \leq \sum_{u \leq t} D_{iu}(d), \ \forall i, d \in [s, \tau], t < d \tag{7}$$

$$\sum_{u \leq d} x_{iu}(d) = \sum_{u \leq d} D_{iu}(d), \ \forall i, d \in [s, \tau] \tag{8}$$

$$x_{it}(d) \geq 0, \ \forall i, t, d. \tag{9}$$

The objective is to minimize total inventory holding costs. The objective function represents the holding cost incurred by the containers shipped early, summed across all items and periods. Constraints (5) and (6) ensure that no more containers are shipped than the capacity of the scheduled trains will allow. Constraints (7) ensure that for each due date, cumulative shipments in each period do not exceed cumulative arrivals. Constraints (8) ensure that all containers are shipped on time. Constraints (9) ensure nonnegativity of shipment quantities.

The linear program simply reorganizes the containers within the confines of a fixed train schedule to minimize the combined holding costs incurred at the origin and at the destination. We note that the problem has the structure of a minimum cost network-flow problem, and thus, there exists an optimal integral solution. The total quantity shipped in each period is the same as in the related single-item problem, but the mix of containers now differs. For convenience, let us refer to the optimum solution from the linear program, along with the associated train schedule, as the *reference schedule*.

In the reference schedule all trains are full in each regeneration interval, except possibly one train in the first period. Thus, within each regeneration interval, the transportation cost cannot be reduced. Because the train schedule for the related single-item problem is feasible for the multi-item problem, it is also the minimum *transportation* cost schedule for the multi-item problem. We show that no other train schedule, along with its optimal allocation of containers to trains (from the linear program), with the same or greater number of trains in the regeneration interval, yields lower overall (transportation and inventory) costs. In doing so, we also show that the optimal train schedule within a regeneration interval has the same properties for the multi-item case as for the single-item case.

Theorem 2. *For each regeneration interval, the reference schedule produces an optimal solution to the multi-item problem in which $h_{d_i} \geq h_{o_i}$ for all i.*

Proof. The proof contains three parts. We show that (a) it is not feasible to reduce the cumulative number of trains in any period in the reference schedule while maintaining the same total number of trains in the regeneration interval; (b) increasing the cumulative number of trains in any period while maintaining the same total number of trains in the regeneration interval increases costs; and (c) increasing the total number of trains in the regeneration interval, which requires increasing the cumulative number of trains in at least one period, increases costs. Together, these results demonstrate that the number and timing of trains in the reference schedule is optimal for the multi-item problem.

Part (a). Let us consider reducing the cumulative number of trains in some period, retaining the same total number of trains in the regeneration interval. In the reference schedule, the containers are sent as late as possible while maintaining the full-train property in all but the first period of each regeneration interval, and while satisfying on-time delivery. Thus, it is not possible to reduce the cumulative number of trains in any time period without causing some containers to be tardy.

Part (b). We now consider the case in which the cumulative number of trains sent on or before some arbitrary period increases but the total number of trains during the regeneration interval remains the same. Suppose that in some period t there are sufficient containers available to increase the cumulative number of train movements, e.g., to ship one train one period earlier. Then, the revised linear program will have the same structure as that above, except that the right-hand side of (5) or (6) for the given period t is increased accordingly. The holding costs are higher at the destination than at the origin; thus all coefficients on the $x_{it}(\mathrm{d})$ terms in the objective function are positive. Hence, we can replace the equalities in Constraints (5) and (6) with greater than or equal to relations. Increasing the right-hand side of any of these revised constraints (i.e., increasing the cumulative number of trains) would increase the objective function value. Thus, it is not possible to modify the train schedule to reduce inventory costs by increasing the cumulative number of trains in one period while retaining the same total number of trains.

Part (c). The proof for the case in which the cumulative number of trains shipped on or before some arbitrary period increases while the total number of trains in the regeneration interval increases parallels that in Part (b). □

The procedure described above produces an optimal solution for a given regeneration interval. The test for determining the feasibility of a period as the end of a regeneration interval for the single-item case (described in §2.2) also can be applied to the multi-item case using the aggregate item data. After the container shipment schedule is determined using the linear program for each potential regeneration interval, a shortest path problem must be solved to find the best set of regeneration intervals.

The linear program allows the tradeoff between urgency of the containers and their holding costs to be made optimally in assigning containers to trains. It may seem intuitive to follow a naive approach with respect to a container shipment order, i.e., ship containers in such a way that the holding costs are minimized. However, because of the competing factor of due-date requirements, such an approach may result in more train shipments. The fixed costs associated with these shipments may more than offset any inventory holding costs saved. In general, neither urgency nor costs can be considered alone.

3.2. Modification for $h_{d_i} < h_{o_i}\ \forall i$

The multi-item problem with $h_{d_i} < h_{o_i}$ for all i can be solved by (i) aggregating the demands across items for each (arrival date, due date) combination, (ii) reversing the network in time and space to find the optimal train schedule for the aggregated single-item problem, then (iii) using a linear program analogous to the one described for the case of $h_{d_i} \geq h_{o_i}$ for all i to assign containers to trains.

4. Conclusions and Discussion

We now relate the results in this paper to the special case in which arrivals are static, i.e., all containers are available at the beginning of the horizon.

1. Single-item case with $h_d \geq h_o$: The special case of static arrivals reduces to the problem considered by Lippman (1969), who provides an $O(T^3)$ algorithm for the cost structure that we use here (in addition to more complex algorithms for more general cost structures). We extend certain properties of the optimal solution to the case of dynamic arrivals, present tests that eliminate some periods as the last period in a regeneration interval, and devise an optimal algorithm with $O(T^4)$ complexity. The static-arrival version of our problem is also a special case of Lee (1989), who includes an order setup cost in addition to the multiple setup cost for transportation. His algorithm has $O(T^4)$ complexity.

2. Single-item case with $h_d < h_o$: In the case of static arrivals, the problem is trivial; everything is sent in the first period. When arrivals are dynamic, the problem is more complicated and can be solved with a variant of the algorithm presented for the case in which $h_d \geq h_o$.

3. Multiple items with $h_{d_i} \geq h_{o_i}\ \forall i$: The special case of static arrivals reduces to a problem discussed in Ben-Kheder (1990). He shows that within a regeneration interval, the structure of the optimal policy has full trains in all periods except possibly the first. He presents an $O(T^3)$ solution procedure. We extend this structural result to the case of dynamic arrivals, and

show that the optimal train schedule can be obtained by solving a related single-item problem in place of the more complex multi-item problem. Due to the dynamic arrival pattern, however, the allocation of containers to the trains requires the solution of a linear program.

4. Multiple items with $h_{d_i} < h_{o_i} \forall i$: We are not aware of any prior work on this problem. Most work on lot sizing is motivated by manufacturing applications where inventory holding costs typically increase as the product becomes more complete. However, location-dependent holding costs arise frequently in transportation applications. In the static case, the problem is trivial as all containers are shipped in the first period. For the case of dynamic arrivals, the problem can be solved with a variant of the algorithm presented for the case in which $h_{d_i} \geq h_{o_i}$ for all i.

The ability to find an optimal train schedule using an aggregate item can provide considerable reductions in computation time when there are many types of items and/or many trains to be scheduled. We are not aware of any prior observations of this phenomenon. Of course, the result applies only when $h_{d_i} < h_{o_i}$, or $h_{d_i} > h_{o_i}$ for all i. When the relationships are mixed, both the train schedule and the container shipment schedule may be quite complex. Some items should be shipped as soon as possible while others should be shipped just in time, making it difficult to characterize when full and partial trainloads should be sent. This remains a topic for future research.

Acknowledgments
This research was supported by National Science Foundation Grant GER/HRD 93-96288 to the University of California, Berkeley. The authors gratefully acknowledge the helpful comments of the referees and associate editor on earlier versions of the paper.

References

Ben-Kheder, N. 1990. Economic lot sizing for multi-supplier, multi-item procurement systems. Unpublished Ph.D. dissertation, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI.

Florian, M., M. Klein. 1971. Deterministic production planning with concave costs and capacity constraints. *Management Sci.* **18**(1) 12–20.

Iwaniec, K. 1979. An inventory model with full load ordering. *Management Sci.* **25**(4) 374–384.

Lee, C. Y. 1989. A solution to the multiple set-up problem with dynamic demand. *IIE Trans.* **21**(3) 266–270.

Lippman, S. A. 1969. Optimal inventory policy with multiple setup costs. *Management Sci.* **16**(1) 118–138.

———. 1971. Economic order quantities and multiple setups. *Management Sci.* **18**(1) 39–47.

Yano, C. A., A. M. Newman. 1998. Scheduling trains and containers with due dates and dynamic arrivals. Working paper, Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA.

(Received: September 1998; revisions received: September 1999, April 2000; accepted: June 2000).