SCHOOL OF OPERATIONS RESEARCH
AND INDUSTRIAL ENGINEERING
COLLEGE OF ENGINEERING
CORNELL UNIVERSITY
ITHACA, NEW YORK 14853

TECHNICAL REPORT NO. 830

September 1988

MULTI–ITEM PRODUCTION PLANNING WITH JOINT REPLENISHMENT AND CAPACITY CONSTRAINTS: THE TRUCKING PROBLEM

by

Dev Joneja[1]

[1]This research was supported in part by NSF Grant DMC–8451984, and by AT&T Information Systems and DuPont Corporation.

Multi-Item Production Planning with Joint Replenishment and Capacity Constraints: The Trucking Problem

by

Dev Joneja

School of Operations Research

and Industrial Engineering

Cornell University

Ithaca, N.Y. 14853

September 1988

Abstract

Consider a production system producing several items on any of several identical machines. Each machine has a finite production capacity. The system operates in discrete time to satisfy non-stationary demand of each items in each time period. There is a fixed cost of setting up a machine for production in any time period, and a fixed cost of setting up the production of any item on any machine. In addition, there is a linear inventory holding cost for each item. The objective is to decide when each item is to be produced, and in what quanities, in order to minimize the total setup and inventory holding costs over the planning horizon. The problem can effectively model several distribution systems where trucks are used for transportation, and certain production systems in which production can be subcontracted at a cost. The well known joint replenishment problem is a special case of this model.

We consider both the single item and multi-item versions of this problem. For the single item problem an efficient dynamic programming formulation is developed to obtain the optimal solution. We also develop heuristics for this case which have very tight theoretical bounds on their performance if demand is constant over time. The multi-item case is NP-complete, and we develop fast (linear computation time) forward heuritics for this problem. The behavior of our heuristics for a randomly generated set of problems is discussed.

Introduction

Consider the lot sizing problem for several items over a finite time horizon. The system operates in discrete time, and demands for each item in each time period over the time horizon are assumed to be known. The demand may not be constant over time, but may vary from one time period to the next. This external demand for each item must be satisfied in each time period, and backorders are not permitted.

Each item has a fixed ordering cost which is incurred whenever it is ordered, and a linear inventory holding cost. In addition, there is a shipping cost. All the items which are ordered in a single time period are shipped together in one shipment. The shipment is made in trucks of capacity C each, with an associated cost K_0 for each truck. If the total volume of all the items that are ordered in any time period is x, then a shipping cost of $K_0\lceil x/C \rceil$ will be incurred, where $\lceil y \rceil$ denotes the smallest integer larger than or equal to y. Hence this problem will be referred to as the *Trucking Problem.* We are focussing on decisions of when to order each item, and in what lot sizes, in order to minimize the total ordering, inventory holding, and shipping costs over the entire time horizon.

In a manufacturing environment, this problem arises if the items can be manufactured on any of several identical machines. Each item has a setup cost which is incurred in any time period when a machine is set up to produce it, and a linear inventory holding cost. In a time period, C now denotes the manufacturing capacity of a machine. If a machine is used to produce any item in a time period, a setup cost of K_0 is incurred for that machine. Then if x denotes the total production of all the items in a time period, the number of machines which are required is $\lceil x/C \rceil$, at a total setup cost of $K_0\lceil x/C \rceil$.

Alternately, instead of identical machines, consider the manufacturing of the items as being subcontracted to any of several identical subcontractors. Each of the subcontractors has a production capacity of C in any time period, and the fixed cost of subcontracting is K_0. As before, there is also an individual ordering cost and linear inventory holding cost for each item. The problem again is an instance of the trucking problem.

If the truck capacity C is infinite, then only one truck will be required whenever any shipment is made. Thus a joint cost of K_0 will be incurred whenever any one or more items are ordered together in the same time period. This special case is referred to in the literature as the *Joint Replenishment Problem.* Both the nonstationary demand, discrete time, finite horizon version and the constant demand, continuous time, infinite horizon version of the joint replenishment problem have been extensively studied in the literature. For the non-stationary demand case, effective approximation algorithms are provided by Joneja (1988). An extensive review of this problem is provided by Aksoy and Erenguk (1987).

The joint replenishment problem has been shown to be NP-complete (Joneja, 1987). Since it is a special case of the trucking problem, the latter is also NP-complete. In this paper, we will develop approximation algorithms for the trucking problem which are both fast and effective. The algorithms are forward algorithms, which use the solution of the problem over the first t time periods to construct the solution for time period $t+1$. In so doing, only the last order in the solution for periods 1 through t depends on demands in time periods after t. Thus changes in demand forecasts far into the future have little effect on current ordering policies. The algorithms thus have the advantage of controlling nervousness of the generated policy.

This paper is organized as follows. In the next section we will consider the trucking proglem when there is only one item. We have discussed the problem so far in the more general context of several items. The single item trucking problem is interesting in its own right. In addition, this will guide the development of the algorithm for the multi-item problem. The optimal solution for the constant demand, continuous time, infinite horizon version will be obtained. This will guide the development of a heuristic for the non-stationary demand, finite horizon case. In addition, we will provide a dynamic programming formulation for the latter case to obtain the optimal policy. In section 2 we will consider the multi-item version in discrete time with non-stationary demands, and develop a heuristic for this problem. A lower bound on the cost of the optimal solution will also be obtained. In section 3 we will report our computational experience with both of the heuristics. The last section presents our conclusions.

1 The Single Item Trucking Problem

1.1 The Constant Demand Version

In this section, we discuss the trucking problem for a single item. This is a relevant problem, and will later guide the development of the algorithms for the multi-item version. We will first consider the case where the demand for the item is constant over time. The problem is considered in continuous time over an infinite time horizon, with the objective of finding a policy with the minimum average cost per unit time. The exact solution to this problem can be obtained, and will be useful in developing appriximation algorithms for the case of non-stationary demands. We will use the following notation:

d = demand rate of the item per unit time,

H = holding cost rate of the item per unit quantity per unit time,

K = ordering cost of the item,

K_0 = truck cost,

Without loss of generality, we assume that one unit of the item fills one truck, so that the truck capacity is 1. This can easily be accomplished by scaling the demand rate and the holding cost rate of the item. If T denotes the reorder interval of the item, then the average cost per unit time is:

$$A(T) = \frac{K + K_0\lceil dT \rceil}{T} + \frac{HdT}{2}.$$

To find the optimal reorder interval, we first note that the average cost function is bounded below by the lower bounding function:

$$L(T) = \frac{K}{T} + \frac{HdT}{2} + dK_0.$$

In addition, the two functions are equal at every value of T where dT is an integer. Let $T_L = \sqrt{2K/Hd}$, and $T_A = \sqrt{2(K + K_0\lceil dT_L \rceil)/Hd}$ (see Figure 1). Also, let $T_+ = \lceil dT_L \rceil / d$, and $T_- = \lfloor dT_L \rfloor / d$. Note that $L(T)$ achieves its minimum at T_L, and that dT_L is the economic order quantity (EOQ) for the item. T_- and T_+ are respectively obtained by rounding the reorder quantity dT_L down or up to the closest full truck. The minimum of $A(T)$ is given by the following theorem:

Theorem 1 *If $T_- < T_A \le T_+$ then $A(T)$ achieves its minimum at T_- or T_A. Otherwise $A(T)$ achieves its minimum at either T_- or T_+.*

Proof: At all $T < T_-$, we have $A(T) \ge L(T) > L(T_-) = A(T_-)$, where the second inequality arises from the convexity of the function L and the fact that $T_- \le T_L$. Similarly, for all $T > T_+$ we have $A(T) \ge L(T) > L(T_+) = A(T_+)$. Thus the minimum of $A(T)$ lies in the interval $[T_- \, , \, T_+]$. Next, note that $A(T)$ is continuously differentiable at all points where dT is non-integer. If $A(T)$ is minimized at a point where dT is non-integer, it must satisfy $\partial A(T)/\partial T = 0$, which gives $T = \sqrt{2(K + K_0\lceil dT \rceil)/Hd}$. If this happens for a T in the interval $[T_- \, , \, T_+]$, then $T = \sqrt{2(K + K_0\lceil dT_L \rceil)/Hd} = T_A$. Further, in this case $\partial A(T)/\partial T > 0 \; \forall \, T_A <$

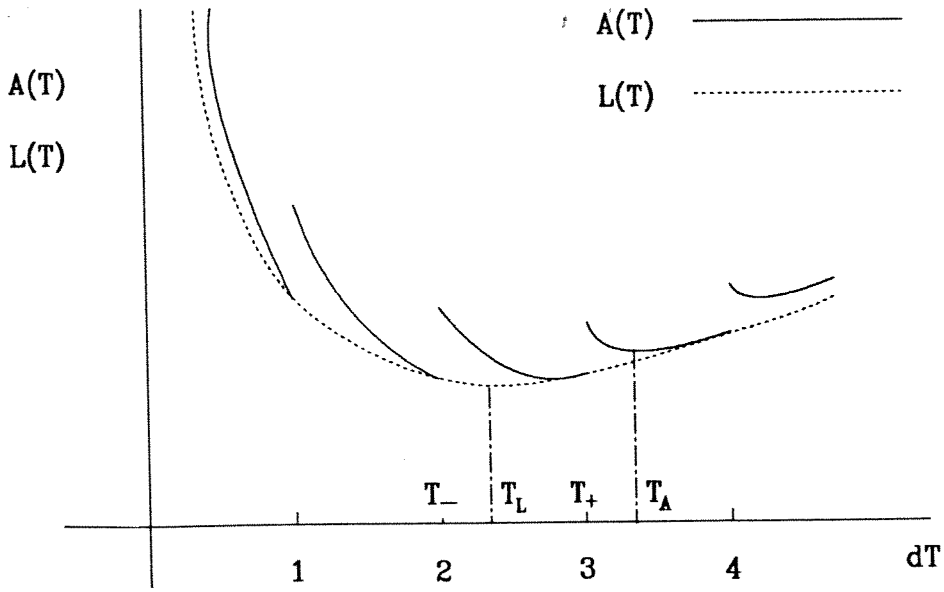

Figure 1: The average cost funtion $A(T)$ and lower bounding function $L(T)$

$T < T_+$, so that $A(T_A) \leq A(T_+)$. Hence in this case $A(T)$ achieves its minimum at either T_-, *or* T_A. Otherwise $A(T)$ is not minimized by any non-integer dT for any T in the interval $[T_- \ , \ T_+]$. In this case, the minimum is at either T_- or at T_+. □

Thus the optimal policy can be computed by finding T_L, T_A, T_- and T_+, and computing $A(T)$ at each of these points. Notice that the optimal policy may send partially filled trucks if the minimum is at T_A. On the other hand, if the minimum is at T_- or T_+, the optimal policy sends only full trucks. This observation allows an interesting approximate solution to the problem. If we restrict ourselves to policies in which only full trucks are sent, clearly the best solution is at T_- or at T_+. This is called a *full truck policy*. If the global minimum of $A(T)$ occurs at T_A, this would then be rounded down to T_- or up to T_+. To evaluate the performance of this policy, define the *relative cost* of a policy as the ratio of the cost of the policy to the cost of the optimal policy.

Let the reorder interval of the optimal full truck policy be T_F, where $T_F = T_-$ *or* T_+. Clearly, the relative cost of this full truck policy is 1 if T_A does not lie in the interval $[T_- \ , \ T_+]$. If T_A lies in this interval, then the relative cost can be larger than 1. In this case, consider any full truck policy with a reorder interval of T, where dT is an integer. The relative cost of this policy is $A(T)/A(T_A) \leq L(T)/L(T_A) \leq L(T)/L(T_L)$. Simple algebra gives $L(T_L) = 2K/T_L + dK_0$, and $L(T) = K(1/T + T/T_L^2) + dK_0$. Thus:

$$\text{relative cost} \leq L(T)/L(T_L) \leq \frac{K(1/T + T/T_L^2)}{2K/T_L} \leq \frac{1}{2}\left(\frac{T_L}{T} + \frac{T}{T_L}\right).$$

Theorem 2 *If $dT_L \geq 1/\sqrt{2}$, then the relative cost of the optimal full truck policy is no worse than 1.06.*

Proof: For all T_L such that $dT_L \geq 1/\sqrt{2}$, there is at least one value of T such that dT is an integer and $T_L/\sqrt{2} \leq T \leq T_L\sqrt{2}$. For any such value of T, consider the full truck policy whose reorder interval is T. Its relative cost is no larger than $1/2(T/T_L + T_L/T) \leq 1/2(\sqrt{2} + 1/\sqrt{2}) = 1.06$. Thus the optimal full truck policy has a relative cost no larger than 1.06. □

This result is very similar to that obtained for a special form of routing problem with one warehouse supplying several retailers, obtained by Gallego and Simchi-Levy (1988). Notice that dT_L is just the econimic order quantity for the item. This theorem then says that if the EOQ fills at least 0.7 of a truck, then a full truck policy will always be very close to optimal. On the other hand, the EOQ may fill less than 0.7 of a truck, in which case the full truck policy can be arbitrarily bad. This motivates the *Modified Full Truck Policy*, which has a reorder interval of T_A if $dT_A < 1$, and T_F otherwise. In other words, if the optimal solution would ship less than one truck, then this optimal solution is used and partial trucks are sent. If the optimal solution would ship one or more truck, then partial trucks are not sent, and only full trucks are used.

For the modified full truck policy, if $dT_A < 1$ then from Theorem 1 $A(T)$ achieves its minimum at T_A (note $T_- = 0$, so $A(T_-) = \infty$). Since the policy ships dT_A in this case, its relative cost is 1. If $dT_A \geq 1$ and $dT_L < 1/\sqrt{2}$, then by Theorem 1 $A(T)$ achieves its minimum at T_+, which is what the modified full truck policy uses. In this case again its relative cost is thus 1. Finally, if $dT_A \geq 1$ and $dT_L \geq 1/\sqrt{2}$, then the policy sends full trucks and the result of Theorem 2 holds. Thus the following result is clear:

Corollary 3 *The relative cost of the modified full truck policy is never larger than 1.06.*

In the next section we will use the idea behind this policy to develop a heuristic for the non-stationary demand version of the trucking problem.

1.2 The Non-stationary Demand Version

1.2.1 A Dynamic Programming Formulation

In this section we consider the single item trucking problem with non-stationary demands in discrete time. The demand in time period t is d_t. The objective is to find a policy with minimum total ordering, inventory holding, and trucking cost over a finite time horizon of length T. Let x_t denote the quantity ordered in time period t, and I_t denote the inventory of the item at the end of time period t. The first lemma below is very similar to that obtained for capacitated dynamic lot sizing problems by Baker et. al. (1978). Again, with no loss of generality we are assuming that the truck capacity is 1.

Lemma 4 *In any optimal policy, if $x_t I_{t-1} > 0$ in a time period t then x_t must be an integer.*

Proof: If in time period t the policy has $x_t I_{t-1} > 0$ and x_t non-integer, then there is a $\delta > 0$ satisfying $x_t + \delta \leq \lceil x_t \rceil$ and $\delta \leq I_{t-1}$. If s is the last time period prior to t at which an order is placed, then decreasing the order quantity at s by δ and increasing that at t correspondingly will decrease the cost of the policy by at least $H(t-s)\delta > 0$. $\square$

Lemma 5 *In any optimal policy, let $x_t I_{t-1} > 0$ in a time period t. Let $v \geq t$ be the first time period after period $t-1$ with $I_v = 0$, and let $D_{tv} = \sum_{s=t}^{v} d_s$. Then $I_{t-1} = D_{tv} - \lfloor D_{tv} \rfloor$.*

Proof: In an optimal policy let $x_t I_{t-1} > 0$ and $I_{t-1} \geq 1$ in any time period t. Let s be the last time period prior to t in which the item was ordered. Decreasing the order quantity at s by $\delta = \min(1,\ x_s)$ and correspondingly increasing that at t will decrease the cost of the policy by at least $H(t-s)\delta$. Thus $I_{t-1} < 1$. Also, by lemma 4 the order quantity in every time period from t through v is integer. Thus I_{t-1} equals the fractional part of the total demand in time periods t through v, which is given by $D_{tv} - \lfloor D_{tv} \rfloor$. $\square$

To obtain a dynamic programming formulation of this problem, note that if we know $I_t = 0$ in an optimal policy, then any ordering decisions in a time period after t can be made independently of the ordering decisions up to time period t. Thus requiring that $I_t = 0$ decomposes the problem into independent subproblems over the time intervals $[1\ ,\ t]$ and $[t+1\ ,\ T]$ which are of the same form as the original trucking problem. Define C_{uv} as the minimum cost in time periods $(u+1),\ \ldots,\ v$ assuming $I_u = I_v = 0$, and $I_t > 0 \ \ \forall\ u < t < v$. Also, let $f(v)$ denote the minimum cost in time periods $1, \ldots, v$ assuming $I_v = 0$. Then $f(v)$ can be obtained by the following dynamic programming recursion:

$$\begin{aligned} f(0) &= 0 \\ f(v) &= \min_{0 \leq u < v} (\ f(u) + C_{uv}\ ) \qquad \forall\ v = 1, \ldots,\ T. \end{aligned}$$

The optimal cost over the time horizon is given by $f(T)$.

We are now left with the problem of finding C_{uv}. This optimization problem is denoted by $P(u,v)$. For this, let $u < s < t \leq v$, where s and t are two successive time periods in which an order is placed. Note that lemma 5 specifies I_{s-1} and I_{t-1} (with $I_u = 0$). Then the quantity ordered at time s is given by $x_s = I_{t-1} + D_{s,t-1} - I_{s-1}$. Thus the cost incurred in time periods s through $t-1$ can be calculated. Let $\bar{C}_{st}$ denote this cost, which can be obtained using $\bar{C}_{s+1,t}$ in constant computational time. Similarly, $\bar{C}_s$, which is the cost incurred in periods s through v assuming that the last order in $P(u,v)$ is at s can be obtained for all values of s, $u < s \leq v$. Also, let $\bar{f}(s)$ denote the minimum cost in time periods s through v in problem $P(u,v)$. Then $\bar{f}(s)$ can be computed using the following dynamic programming recursion:

$$\begin{aligned} \bar{f}(v) &= \bar{C}_v \\ \bar{f}(s) &= \min(\min_{s<t\leq v}(\bar{C}_{st} + \bar{f}(t)) \; , \; \bar{C}_s) \qquad \forall \; s = v-1, \ldots, u. \end{aligned}$$

Then $\bar{f}(u)$ is equal to C_{uv}.

To estimate the computation time of this algorithm, first note that the solution to $P(u,v)$ can be obtained using the solution to $P(u+1,v)$ as follows. In $P(u+1,v)$, we have $I_{u+1} = 0$, while in $P(u,v)$ the value of I_{u+1} is specified by lemma 5. Hence $\bar{C}_{u+1,s}$ $\forall \; s, \; u+1 < s \leq v$ will have to be recomputed. In addition, $\bar{C}_{u,s}$ $\forall \; s, \; u < s \leq v$ have to be calculated. This can be done with $O(v-u)$ computational effort. Note that the values of $\bar{f}(t)$ in $P(u,v)$ are the same as in $P(u+1,v)$ $\forall \; t > u+1$. Thus in $P(u,v)$ only $\bar{f}(u+1)$ and $\bar{f}(u)$ have to be computed, which can be done in $O(v-u)$ time. Thus $P(u,v)$ can be solved using the solution of $P(u+1,v)$ in $O(v-u)$ computational effort. Thus for any v, all the C_{uv}, $u = 0, \ldots, v$ can be obtained in $O(\sum_u (v-u)) = O(v^2)$ time. Hence all the C_{uv} can be computed in $O(T^3)$ computataional time. Finally, the dynamic programming recursion to find $f(T)$ can be performed in $O(T^2)$ time. Thus the overall computation time of this algorithm is $O(T^3)$.

1.2.2 An Approximation Algorithm

In this section we develop an approximation algorithm for the non-stationary demand version of the single item trucking problem. The optimal algorithm discussed in the previous section has a computation time of $O(T^3)$. The heuristic we develop is considerably faster, with a computation time which is linear in T. In addition, the ideas will later be extended to the multi- item trucking problem, which is NP-complete.

The heuristic is very similar to the modified full truck policy developed for the constant demand version of the trucking problem. This policy first finds the economic order quantity of the item. If the policy sends full trucks, then the reorder quantity is obtained by rounding the EOQ up or down to an integer number of trucks. In the following discussion, we will follow the convention that if an inventory of I_t is carried from time period t to $t+1$, the the holding cost of HI_t is incurred in time period t.

To extend this idea to the non-stationary demand version, we first have to develop an analog of the EOQ. Simple algebra shows that the constant demand EOQ can be obtained by equating the holding cost of the item in one reorder cycle to its ordering cost (i.e. $HdT_L^2/2 = K$). Similarly, at time T_A the holding cost equals the total ordering and trucking cost (i.e. $HdT_A^2/2 = K + K_0\lceil dT_A \rceil$). This observation about the EOQ is used in the well known part-period balancing heuristic for the single item dynamic lot sizing problem (DeMatteis and Mendoza, 1968). It proceeds by placing an order in time period 1, and in each successive time period it decides whether or not an order will be placed. When considering time period t, the ordering policy for each period prior to t has already been decided. Then in period t it calculates the total amount of holding cost incurred since the last order was placed. If this total holding cost exceeds the ordering cost, an order is placed in the current time period. It then proceeds to the next time period.

This idea is extended to the trucking problem. The heuristic for the single item trucking problem is stated in Figure 2. It starts in time period 1 by placing an order (step 1). At any time period t, let the last order placed by the heuristic be in time period s. This order will satisfy a part $f_s d_s$ of the demand in time period s. The value of f_s has already been fixed. It will further satisfy the demand in time periods $s+1,\ s+2,\ldots$. First we assume that it satisfies the total demand through time period t, and find the total holding cost incurred in time periods s through $t-1$. Algebraically, this is given by:

$$\text{holding cost } = H \sum_{\tau=s+1}^{t} (\tau - s) d_\tau. \tag{1}$$

If this holding cost exceeds the order cost of the item, it becomes a *candidate for an order* (step 3 of the algorithm.)

When the item becomes a candidate for an order, the tentative order quantity of the last order is $x_{st} = f_s d_s + \sum_{\tau=s+1}^{t} d_\tau$. There are two cases: $x_{st} < 1$ and $x_{st} \geq 1$. If at time t the item is a candidate for order and $x_{st} < 1$, then we determine whether the holding cost exceeds the sum of the ordering cost and the trucking cost. Since $\lceil x_{st} \rceil = 1$, the trucking cost is just K_0. Hence an order is placed in time period t if the holding cost exceeds $K + K_0$. The size of the last order in time period s is set to $x_{s,t-1}$, which covers the demand in time periods s through $t-1$. The order in period t will cover the entire demand in period t, and we set $f_t = 1$. This order will be referred to as a *regular order* of the item (step 3(a) of the heuristic). This is similar to the order of size dT_A in the modified full order policy for the constant demand version, where a single partially filled truck is sent.

On the other hand, if this order quantity of the last order is at least 1 truck ($x_{st} \geq 1$), then we restrict ourselves to sending only full trucks. This is parallel to the modified full truck policy for the constant demand case. It is accomplished as

1. Set last order period, $s = 1$, set $f_1 = 1$; set time index $t = 1$.

2. Set t = t + 1; If $t > T$ stop, otherwise go to step 3.

3. Find holding cost since s given by equation 1; find $x_{st} = f_s d_d + \sum_{\tau=s+1}^{t} d_\tau$. If holding cost $> K$:

   (a) If holding cost $> K + K_0$, place regular order at t, set $s = t$ set $f_t = 1$.

   (b) Otherwise, if tentative order quantity, $x_{st} > 1$, place a full truck order in time period p obtained from equations 2. Set $s = p$, and set $f_s = f_p$.

4. Go to step 2

Figure 2: Heuristic for the single item trucking problem.

follows. Since $x_{st} \geq 1$, and the order size in time period s will be $\lfloor x_{st} \rfloor$. Clearly there exists a time period p and fraction $0 \leq f_p < 1$ such that

$$s < p \leq t, \quad and \tag{2}$$

$$f_s d_s + \sum_{\tau=s+1}^{p-1} d_\tau + f_p d_p = \lfloor x_{st} \rfloor.$$

The next order after s will be placed in time period p, and f_p will be fixed. This will be referred to as a *full truck order* of the item.

The heuristic then proceeds to the next time period. To estimate the computational effort for the heuristic, note that the time period p in equations 2 can be obtained with $O(t - s)$ effort. All other computations in time period t can be done with constant computational effort. Clearly the heuristic can be implemented so that it will run in $O(T)$ time.

2 The Multi-Item Trucking Problem

In this section we discuss the multi-item trucking problem with non-stationary demands. The system operates in discrete time, and the objective is to minimize the total holding, ordering, and trucking cost over a finite time horizon of length T. As discussed previously, the multi-item trucking problem with infinite truck capacity gives the joint replenishment problem, which is NP-complete. Thus the multi-item trucking problem is also NP-complete. In this section we will develop a fast approximation algorithm for this problem.

The system is made of N items, and the subscript n will be used for item n. K_n, H_n, and d_{nt} respectively denote the ordering cost, holding cost rate, and demand in period t for item n. Again, with no loss of generality we assume that one unit of any item fills one truck, and that the truck capacity is 1.

As in the single item heuristic, the multi-item algorithm starts by placing an order for each item in time period 1, and proceeds successively from one time period to the next. The two basic elements of the single item heuristic, the regular order and the full truck order, will be used in the multi-item heuristic. To this is added a third element, the *pull order*. Let the last order of an item n be in time period s_n. Then the holding cost incurred in time periods s_n through t is

$$HC_{nt} = H \sum_{\tau=s_n+1}^{t} (\tau - s_n)d_\tau. \tag{3}$$

Recall that an item becomes a *candidate* for an order when its total holding cost since its last order exceed its ordering cost. At time period t, let item n become a candidate for an order. Let the last order of any item prior to t be in time period s. If $s > s_n$, then an order of item n is placed in period s. This is a pull order of item n in time period s. The pull order of item n in period s_n will cover demands in time periods s_n through $s-1$. If f_{nt} denotes the fraction of the demand in time period t which is covered by the order in time period t for item n, then $f_{ns} = 1$. The underlying idea of a pull order is to increase the coordination of the order periods of the various items in order to permit future sharing of trucks by the items. This concept originated in the cost covering heuristic for the joint replenishment problem (Joneja, 1988).

The algorithm is stated formally in Figure 3. It starts by placing an order for each item in time period 1 (step 1 in Figure 3). In each time period, the heuristic first places a pull order of each item if possible (step 3). It then determines the candidate items and places a regular order of all the candidate items in time period t if possible (step 4). This is done if the total holding cost of the candidate items exceeds their total setup cost and the truck cost. As above, HC_{nt} denotes the holding cost if item n in time periods s_n through t. A regular order is placed for the candidate items in time period t if:

$$\sum_{n=1}^{N}(HC_{nt} - K_n)^+ \geq K_0.$$

Here $(x)^+$ denotes $\max(0,\ x)$. If a regular order is placed in time period t, then the last order of each candidate item n before t covers all demands through $t-1$, and $f_{nt} = 1$.

If no regular order is placed in period t, the algorithm checks whether a full truck order of the candidate items can be placed (step 5). Notice that due to the pull order, the last order of every candidate item is in time period s. In addition, the fraction f_{ns} has been fixed for each item. The size of the tentative order at t is then x_{st}, where:

$$x_{st} = \sum_{n:n \text{ a candidate item}} \left(f_{ns}d_{ns} + \sum_{\tau=s+1}^{t} d_{n\tau} \right). \qquad (4)$$

A full truck order is placed if $x_{st} \geq 1$. As in the single item problem it is placed in

1. For each item n, set last order period, $s_n = 1$, set $f_{n1} = 1$; set time index $t = 1$. Set last order period of any item, $s = 1$.

2. Set t = t + 1; If $t > T$ stop, otherwise go to step 3.

3. For each item n, find the holding cost HC_{nt} incurred since time s_n from equation 3. If $HC_{nt} > K_n$ and $s_n > s$: place a pull order of item n in period s, set $s_n = s$, and update HC_{nt}.

4. If $\sum_n (HC_{nt} - K_n)^+ \geq K_0$: place a regular order of all candidate items at t, set $s = t$, and for every candidate item n set $s_n = t$ and $f_{nt} = 1$.

5. Otherwise, if tentative order quantity $x_{st} > 1$ from equation 4, place a full truck order of each candidate item in time period p as determined from equation 5, and for each candidate item n set $s_n = p$ and $f_{s_n} = f_{np}$.

6. Go to step 2

Figure 3: Heuristic for the multi-item trucking problem.

a time period p where:

$$s < p \leq t, \quad and \tag{5}$$

$$\sum_{n:n \text{ is candidate}} f_{np} d_{cp} + x_{s,p-1} = \lfloor x_{st} \rfloor$$

Every candidate item is ordered in time period p, and f_{np} for all candidate items n is equal, and is set so that the equality in the last line is maintained.

The heuristic then proceeds to the next time period. Clearly the algorithm can be implemented to run in $O(NT)$ computational time.

3 Computational Results

3.1 The Single-Item Trucking Problem

The heuristic for the single item trucking problem with non-statinary demands was coded in the C programming language and tested on a randomly generated set of problem instances. The optimal solution for this problem was also obtained using the dynamic programming formulation. This allowed a comparison of the heuristic solution to the optimal solution.

The data for the tests was generated as follows. In the following discussion, a $U(\mu, \sigma)$ distribution denotes a uniform probability distribution with mean μ and standard deviation σ.

- The individual order costs were generated from a $U(15, 5)$ distribution. The holding cost of the item was taken as 1.
- Demands in each time period were generated from a $U(\mu, \sigma)$ distribution.
- A planning horizon of 120 time periods was used.

A total of 750 test problems were used to evaluate the performance of the heuristic. The average difference of the heuristic solution from the optimal cost was 3.82%. About 81% of the problems were less than 5% from the optimal cost. Only 4 problems had a difference of over 10%, the maximum obtained being 12.3% from the optimal cost. The heuristic thus appears to perform very well in general. The computation time for each problem was under 1 second on an AT&T 6300 personal computer.

Next we investigate the sensitivity of the heuristic to the various cost and demand parameters. We first consider the effect of demand variability and average demand on the performance of the heuristic. The results are presented in Table 1.

In the tables we report $\bar{Y}$, which is the average percentage differnce of the heuristic solution from the cost of the optimal solution ($\bar{Y} = 100\times$(relative cost - 1)). The numbers within brackets in the tables are the standard deviation of $\bar{Y}$. The average demand could take one of three possible values of 1, 2, or 3. Corresponding to each value of mean demand μ, three values of standard deviation σ were used: 0.1μ, 0.3μ, and 0.5μ. Each of these nine combinations was tested over a wide spectrum of values of truck cost K_0 and capacity C. The relative cost of the heuristic is fairly insensitive to the average demand and variability of demands, considering the standard deviation of the relative costs obtained.

Next we consider the sensitivity of the heuristic to variations in truck cost K_0 and truck capacity C. These tests used demands generated from a $U(1, 0.5)$ distribution. Six values of K_0 were used: 15, 30, 45, 60, 75 and 90. Note that the expected value of the individual order cost is 15. Eight values of truck capacity were used: 1, 3, 5, 7.5, 10, 12.5, 15 and 20. This varies the truck capacity from the very low, where 5 to 7 trucks were used each time an order was placed, to very high, where each order filled only a fraction of one truck. The values of relative cost obtained in each case are reported in Table 2.

If the truck capacity is high, so that only one partially filled truck is used in any time period in which an order is placed, then the problem is essentially equivalent to the single item dynamic lot sizing problem with an ordering cost of $K + K_0$. It is also easy to see that our heuristic in this case acts in the same manner as the part-period balancing heuristic (DeMatteis and Mendoza, 1968) for the single item dynamic lot sizing problem. Since the part-period balancing heuristic generally performs very well, a similar performance would be expected of our heuristic in this case. In Table 2, the relative cost figures for $C = 12.5, 15$ and 20 show that this is indeed the case.

At the other extreme, if the truck capacity is low, then a large number of trucks will be used, and only a few of them will be partially filled. Thus the truck utilization should be high. This is exactly what our heuristic does. In addition, in the policy generated by our heuristic, in most cases when an item becomes a candidate for an order it is immediately ordered. Thus the performance of the heuristic in this case is also very good, as is clear from Table 2 for $C = 1 \text{and} 3$. In addition, in this case the heuristic sends very few partially filled trucks. As the cost of a truck, K_0, increase, the truck cost dominates other costs. As a result, the relative cost of the heuristic improves.

The performance of the heuristic is the worst when the costs and truck capacity are such that close to 1 truck are used when the heuristic places an order. If the quantity shipped in a time period is slightly over 1 truck, then the heuristic would use 2 trucks, leading to poor truck utilization. For a fixed value of truck capacity, if the ordering costs are low then the order quantities will be small. In this case the heuristic performs well. As the truck cost increases, the order quantity increases until it is close to 1 truck. As discussed above, this results in an increase in the relative cost of the heuristic. Further increases in the truck cost lead to a small improvement in the performance of the heuristic. Table 2 shows that even in this worst case the average difference of the heuristic from the optimal cost is typically no larger than 6%.

3.2 The Multi-Item Trucking Problem

The heuristic for the multi-item trucking problem was coded in the C programming language, and tested over a broad spectrum of randomly generated problems. For the multi-item problem, no efficient optimal solution procedure is available. The solution generated by the heuristic was thus compared to a lower bound on the cost

Table 1: Single item trucking problem: sensitivity to demand parameters.

σ	Mean of demand process, μ		
	1	2	3
0.1μ	2.7 (1.9)	3.9 (2.1)	3.1 (2.2)
0.3μ	3.4 (2.1)	3.6 (2.0)	3.9 (1.8)
0.5μ	4.0 (1.7)	4.8 (2.1)	6.2 (2.1)

Demand $\sim U(\mu, \sigma)$, $T = 120$

Table shows $\bar{Y} = 100\times$(relative cost - 1), averaged over 30 replications. Numbers in brackets are standard deviation over the 30 replications. Divide these by $\sqrt{30}$ to obtain standard error of $\bar{Y}$.

Table 2: Single item trucking problem: sensitivity to truck cost and capacity.

Truck cost K_0	Truck capacity, C: 1	3	5	7.5	10	12.5	15	20
15	3.6 (0.6)	7.1 (1.6)	6.4 (1.7)	6.2 (1.7)	4.7 (2.0)	4.3 (1.4)	3.7 (2.2)	3.9 (1.6)
30	2.0 (0.4)	4.1 (0.8)	3.9 (1.0)	6.0 (2.0)	3.8 (1.2)	3.8 (1.2)	4.0 (1.4)	3.5 (1.8)
45	1.4 (0.3)	2.3 (0.8)	3.5 (0.5)	4.5 (2.0)	6.4 (1.7)	2.9 (1.5)	3.0 (1.0)	3.7 (1.0)
60	1.1 (0.1)	2.1 (0.6)	1.9 (0.8)	4.8 (1.9)	6.1 (2.6)	5.6 (1.8)	3.9 (2.4)	3.6 (1.0)
75	1.0 (0.1)	1.5 (0.4)	2.2 (0.6)	3.1 (2.0)	4.3 (2.6)	5.5 (2.3)	4.3 (2.1)	3.6 (1.4)
90	0.7 (0.1)	1.5 (0.5)	2.1 (0.7)	3.1 (1.2)	4.0 (1.5)	6.4 (3.4)	5.0 (2.1)	3.4 (1.6)

Demand $\sim U(1, 0.5)$, $T = 120$

Table shows $\bar{Y} = 100\times$(relative cost - 1), averaged over 10 replications. Numbers in brackets are standard deviation over the 10 replications. Divide this be $\sqrt{10}$ to obtain the standard error of $\bar{Y}$.

of the optimal policy. The relative cost figures reported in this section thus are with respect to the lower bound. The relative cost with respect to the optimal cost is expected to be lower.

To evaluate the performance of the heuristic for the multi-item problem, we require a lower bound on the cost of the optimal policy. In order to obtain this bound, we will formulate the problem as a mixed integer programming problem. For this, we use the following notation:

$$x_{nt} = \text{order size of item } n \text{ in time period } t,$$

$$I_{nt} = \text{inventory of item } n \text{ at the end of time period } t,$$

$$z_{it} = \begin{cases} 1 & \text{if truck } i \text{ is used in time period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$y_{nt} = \begin{cases} 1 & \text{if item } n \text{ orders in period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$M = \lceil \textstyle\sum_{n,t} d_{nt} \rceil$$

The formulation, referred to as the problem (TP), is given in Figure 4. The first constraint set is the inventory balance constraint. The constraint set (iii) ensures that an adequate number of trucks is used in each time period. Constraint set (iv) says that if any item places an order in time period t, then the first truck must be used in time period t. This constraint is redundant in the present formulation. However, it will be useful in the relaxations we will next consider.

To obtain a lower bound on the cost of the optimal policy, a relaxation of the problem (TP) is used. To this end, the integrality constraint on the variables z_{it} is relaxed for all $i \geq 2$. We can then substitute the variable $w_t = \sum_{i=2}^{M} z_{it}$, $w_t \geq 0$ in the constraint sets and in the objective function. In this relaxed problem, the constraint set (iv) is no longer redundant. Finally, we obtain a lower bound to the solution of the relaxed problem by using Lagrangian relaxation, dualizing the

$$
\begin{array}{llllllll}
\text{Min.} & \multicolumn{7}{l}{\sum_{n=1}^{N}\sum_{t=1}^{T}(H_n I_{nt} + K_n y_{nt}) \; + \; K_0 \sum_{i=1}^{M}\sum_{t=1}^{T} z_{it}} \\
\text{s.t.:} & I_{n,t-1} & + & x_{nt} & - \; I_{nt} & = \; d_{nt} & \forall \; n, \; t & (i) \\
 & M y_{nt} & - & x_{nt} & & \geq \; 0 & \forall \; n, \; t & (ii) \\
 & \sum_{i=1}^{M} z_{it} & - & \sum_{n=1}^{N} x_{nt} & & \geq \; 0 & \forall \; t & (iii) \\
 & z_{1t} & - & y_{nt} & & \geq \; 0 & \forall \; n, \; t & (iv) \\
 & x_{nt}, \; I_{nt} & \geq & 0, & y_{nt}, \; z_{it} & 0 \; or \; 1. & &
\end{array}
$$

Figure 4: Problem (TP): Integer programming formulation of the trucking problem.

constraint sets (iii) and (iv). The Lagrangian relaxation decomposes the problem into N single item dynamic lot-sizing problems. These can be solved efficiently using the Wagner-Whitin algorithm. We used the standard subgradient optimization procedure (see, for example, Fisher 1981) to solve the Lagrangian dual problem.

The heuristic was tested on a wide variety of test problems. Three specific cases, with 2 items, 4 items, and 8 items were considered. Actually, if there is only one item, then it is easy to see that the multi-item heuristic is identical to the single-item heuristic, the performance of which has been discussed in the previous section. In each case, the demand for item n in each time period was selected from a $U(\mu_n, \sigma_n)$ distribution. The mean demand μ_n for item n was between 1 and 3, and the standard deviation σ_n was between 0 and $0.5\mu_n$. A variety of cases, from identical items to very different item demands, and from low to very high variability of demands, were tested in each case. The item ordering costs were drawn from $U(15, 5)$ distributions, and the item holding costs were taken as 1. A planning horizon of 120 time periods was used in each case.

The heuristic was run on a total of 1080 test problems. The average percentage difference from the lower bound over all the tests was 9.0%. About 68% of the problems had a differnece from the lower bound of below 10%. The largest difference obtained was 31.1%. It must be noted that these numbers are the percentage difference of the cost of the heuristic policy from the lower bound on the cost of the optimal policy. The true optimal cost lies somewhere in between the heuristic cost and the lower bound. The only information we have about the true optimal solution is for the single-item problem. To get an idea of where the true optimum lies between the cost of the heuristic and the lower bound, we calculated the lower bound as describle above for the 750 test problems which were used to test the single item heuristic. For these problems the average percentage difference of the heuristic cost from the lower bound was 6.4%. As discussed in the last section, the average difference of the single-item heuristic cost from the optimal solution was 3.82%. Thus the gap between the optimal cost and the lower bound in the single item problem accounts for aproximately 40% of the difference of the heuristic from the lower bound. It is expected that a similar relationship holds for the multi-item problem also.

The details of the results for the 2 item, 4 item and 8 item test problems are shown in Tables 3, 4, and 5 respectively. In each case, the sensitivity of the solution to variations in truck cost and truck capacity was investigated. The truck cost was varied from about one-sixth of the total expected ordering cost of the items to 6 times the total expected ordering cost of the items. The truck capacity was varied so that the order size in any time period would be from one-half to about 8 trucks.

We first point out that if the truck capacity is large, so that only one partially filled truck is used in any time period in which an order is placed, then the problem is identical to the joint replenishment problem. In this case, the multi-item heuristic actually generates an ordering policy identical to variant 2 of the cost covering heuristic for the joint replenishment problem discussed in Joneja (1988).

The heuristics for the joint rpelenishment problem in Joneja (1988) had an average percentage difference from the lower bound of about 4%. When the truck capacity is large and truck cost is small, we observe a similar performance from our heuristic for the multi-item trucking problem.

The results further show that for a fixed truck capacity, as the truck cost increases the relative cost first increases and then has a slight decrease. The value of truck cost for which the relative cost is the highest increases as truck capacity increases. Both of these effects were observed in the single-item heuristic, as discussed in the last section. The relative cost appears to increase slightly as the number of items is increased. The greatest effect is when going from a single item to 2 items. As discussed above, this could be because the single item heuristic is evaluated with respect to the optimal solution, while the multi-item heuristic is evaluated with respect to the lower bound on the cost of the optimal policy.

4 Conclusions

In this paper we introduce the single and multi-item trucking problem. This problem can effectively model, for example, certain distribution systems with transportation costs, and manufacturing systems with parallel machines or where production can be subcontracted. The joint replenishment problem is a special case of the trucking problem. For the single item problem we show that the continuous time, infinite horizon, constant demand version of the problem can be solved optimally. For this version we also introduce the concepts of the Full Truck Policy and the Modified Full Truck Policy, which are approximation solutions to the problem. We prove that the cost of these policies will be very close to the optimal cost even in the worst case. For the single item problem in discrete time over a finite time horizon with non-stationary demands, we develop an $O(T^3)$ dynamic programming formulation

Table 3: The multi-item heuristic for 2 items: sensitivity to truck cost and capacity.

Truck Cost, K_0	Truck capacity, C: 2	5	15	22.5	30	50
5	7.9% (1.0)	11.6 (4.1)	6.0 (3.6)	3.0 (1.7)	2.7 (1.6)	2.6 (2.1)
15	8.2 (0.7)	13.3 (2.8)	14.1 (2.2)	3.1 (1.8)	2.4 (0.8)	2.2 (1.3)
30	6.7 (1.0)	13.1 (2.4)	13.5 (2.8)	8.6 (5.4)	2.4 (0.8)	1.1 (1.0)
60	5.0 (1.3)	11.1 (2.9)	9.0 (1.3)	9.2 (2.4)	4.6 (1.9)	1.4 (0.4)
120	4.0 (1.4)	9.0 (1.6)	8.4 (3.4)	7.7 (1.5)	7.3 (1.1)	3.5 (0.7)
180	4.0 (1.5)	8.5 (2.9)	5.3 (0.9)	6.0 (1.4)	6.7 (2.2)	4.8 (0.5)

Table shows $\bar{Y} = 100\times$(relative cost - 1), averaged over 10 replications. Numbers in brackets are std. deviation over the 10 replications. Divide these by $\sqrt{10}$ to obtain the standard error of $\bar{Y}$

Table 4: The multi-item heuristic for 4 items: sensitivity to truck cost and capacity.

Truck Cost, K_0	Truck capacity, C: 4	10	30	45	60	100
10	10.9% (2.0)	13.8 (4.0)	5.9 (2.5)	2.1 (1.0)	1.9 (0.8)	1.2 (0.8)
30	8.9 (1.4)	16.8 (3.0)	16.9 (4.3)	2.7 (1.2)	1.7 (0.3)	1.4 (0.6)
60	7.4 (1.5)	16.1 (4.1)	18.5 (4.3)	5.3 (3.1)	2.9 (0.3)	1.4 (0.5)
120	6.8 (0.9)	14.2 (1.8)	16.2 (8.2)	9.7 (1.4)	4.7 (0.7)	2.6 (0.3)
240	5.5 (1.1)	13.1 (3.1)	19.8 (5.7)	9.3 (2.7)	8.5 (1.0)	5.6 (0.3)
360	5.1 (0.9)	11.0 (2.7)	16.9 (8.4)	7.4 (1.8)	7.5 (1.2)	6.5 (0.3)

Table shows $\bar{Y} = 100\times$(relative cost - 1), averaged over 10 replications. Numbers in brackets are std. deviation over the 10 replications. Divide these by $\sqrt{10}$ to obtain the standard error of $\bar{Y}$

Table 5: The multi-item heuristic for 8 items: sensitivity to truck cost and capacity.

Truck Cost, K_0	Truck capacity, C: 8	20	60	90	120	200
20	11.5 (2.1)	14.3 (3.4)	6.0 (2.4)	2.5 (1.3)	2.1 (0.6)	1.6 (0.8)
60	8.8 (0.8)	16.0 (1.6)	17.5 (2.8)	3.7 (2.3)	2.5 (0.3)	1.3 (0.3)
120	7.4 (1.1)	15.2 (1.6)	24.3 (4.0)	5.8 (2.6)	4.2 (0.2)	2.7 (0.1)
240	6.0 (0.4)	12.5 (1.6)	23.1 (2.6)	10.9 (2.8)	7.0 (0.8)	5.9 (0.3)
480	5.6 (1.0)	12.4 (2.1)	24.4 (2.4)	10.5 (4.6)	9.3 (0.8)	9.3 (0.5)
720	5.2 (0.5)	12.2 (2.6)	22.3 (2.5)	10.3 (7.0)	8.8 (1.0)	10.2 (0.7)

Table shows $\bar{Y} = 100\times$(relative cost - 1), averaged over 10 replications. Numbers in brackets are std. deviation over the 10 replications. Divide these by $\sqrt{10}$ to obtain the standard error of $\bar{Y}$

to obtain the optimal policy. We also develop a heuristic for this problem which performed very well on a wide spectrum of test problems. Finally, for the multi-item trucking problem in discrete time over a finite time horizon with non-stationary demands we develop and test a heuristic algorithm. The heuristic is compared to a lower bound on the cost of the optimal policy, and the test results over a wide variety of problems are very encouraging.

The multi-item trucking problem is NP-complete. For this problem no efficient optimal algorithm is currently known. Three research directions appear to hold some promise for solving this problem to optimality. One is to develop structural results about the optimal solution, which could be used to restrict the state space in a dynamic programming formulation. A similar exercise should also allow effective integer programming formulations, which can be solved in several ways. Finally, there has recently been successful development of strong cutting planes for several multi-item lot sizing problems (see, for example, Barany, Van Roy and Wolsey, 1984, and Leung, Magnanti and Vachani, 1988). This approach should prove effective for solving medium size problems to optimality.

Acknowledgments

The author would like to thank Professor Robin Roundy for his advice at various stages in the research for this paper. This research was supported in part by NSF grant DMC-8451984, by AT&T Information Systems and DuPont Corporation, and by the Faculty Research Grant of the Graduate School of Business at Columbia University.

References

[1] Aksoy, Y. and S.S.Erenguk. 1988. Multi-Item Inventory Models with Coordinated Replenishments: A Survey. *International Journal of Operations and Production Management,* **8**, pp. 63-73.

[2] Baker, K., P.S.Dixon, M.J.Magazine and E.A.Silver. 1978. An Algorithm for the Dynamic Lot-Size Problem Time-Varying Production Capacity Constraints. *Management Science,* **24**, 16, pp. 1710-1720.

[3] Barany, I., T.J.Van Roy and L.A.Volsey. 1984. Strong Formulations for Multi-Item Capacitated Lot Sizong. *Management Science,* **30**, 10, pp. 1255-1261.

[4] DeMatteis, J.J. and A.G.Mendoza. 1968. An Economic Lot-Sizing Technique. *IBM Systems journal,* **7**, pp. 30-46.

[5] Fisher, M.L. 1981. The Lagrangian Relaxation Method for Solving Integer Programming Problems. *Management Science,* **27**, pp. 1-19.

[6] Gallego, G. and D. Simchi-Levi. 1988. On the Effectiveness of Direct Shipping Strategy for the One Warehouse Multi-Retailer R-Systems. *Technical Report, School of IE&OR, Columbia University, New York.*

[7] Joneja. D. 1988. The Joint Replenishment Problem: New Heuristics and Worst Case Performance Bounds. *Tech. Report 765, School of OR&IE, Cornell University.*

[8] Leung, J., T.L.Magnanti and R.Vachani. 1987. Facets and Algorithms for Capacitated Lot Sizing. *Tech Report OR 171-87, MIT, Cambridge, Mass.*

[9] Wagner, H.M. and T.M.Whitin. 1958. Dynamic Version of the Economic Lot Size Model. *Management Science,* **5**, 1, pp. 89-96.