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

TECHNICAL REPORT NO. 633

August 1984
Revised November 1985

PLANNING COMPONENT DELIVERY
INTERVALS IN CONSTRAINED
ASSEMBLY SYSTEMS

by

John A. Muckstadt

*This research was supported in part by the NSF Grant MEA-81-21429.

PLANNING COMPONENT DELIVERY
INTERVALS IN CONSTRAINED
ASSEMBLY SYSTEMS

John A. Muckstadt
School of Operations Research
and Industrial Engineering
Cornell University
Ithaca, New York 14853

ABSTRACT

Lot sizes for externally purchased components used in assemblies are often determined so that total inventory carrying costs and fixed ordering costs are approximately minimized. By focusing primarily on this criterion, a number of constraints relating to receiving dock capacity and storage area capacities are usually ignored. The goals of this paper are to develop a model and an algorithm that can be used to establish the time between external procurements and between movement of stock from stage to stage for each component in a multi-stage assembly system recognizing the presence of these capacity constraints.

I. INTRODUCTION

Lot sizes for externally purchased components used in final assemblies are often computed using simple rules. For example, when production rates for final assemblies remain relatively constant over time, some variation of the Wilson lot size formula is frequently used to determine component procurement lot sizes. If the demand rates for final products change over time, then some other technique might be used, such as the part period balancing method, the least unit cost approach, or the Silver-Meal heuristic [8]. All of these methods establish component lot sizes so that inventory carrying costs are compared to fixed ordering costs in such a way that total component procurement costs are approximately minimized. By focusing primarily on this criterion, a number of issues of practical significance are often ignored. The purposes of this paper are to present a model and to describe an algorithm for planning component lot sizes for assembly plants recognizing some of these factors such as limitations on available labor, equipment and space.

Frequently the amount ordered from a vendor for a component may not be delivered at one time, but is scheduled to be delivered in several lots. This concept is central to the so-called "Just-in-Time" inventory control philosophy. For example, an order may be for 52,000 units with weekly deliveries of 1000 units. The component order quantities are normally based on demand projections for final assemblies as stated in a master production schedule; however, as in the example, the delivery quantities may be smaller than the order quantities. If the production rate for the final product is approximately constant, then the order quantity is often an integer multiple of the delivery quantity. Delivery quantities, and the time between the receipt of such quantities, called the delivery intervals, are set based on other considerations such as receiving dock capacity, warehouse space, storage space on the shop floor, storage and delivery container capacities, material handling capacities between a receiving dock and a warehouse and between a warehouse and the shop floor, and any fixed ordering costs and inventory carrying costs. The capacities we have mentioned can be related to both equipment and labor. In addition, the selection of an optimal delivery interval should take into account the lot splitting possibilities that can take place after a delivery is made (e.g. the quantity of a component sent from a warehouse to the shop floor can be less than that component's delivery quantity). We call the optimal delivery quantity the economic delivery quantity and the optimal delivery interval the economic delivery interval.

Our specific goal in this paper is to develop a model and an algorithm that can be used to establish consistent and realistic economic delivery intervals for each component while recognizing the presence of some of the capacity constraints mentioned earlier.

There are a number of reasons for formulating the model in terms of delivery intervals rather than lot sizes. First, in many real environments that we have observed, planners think in terms of the frequency of receipt of components and the frequency of material movements within a plant rather than in terms of lot sizes. Second, when the production plan for each final product varies, it is easy to determine the delivery quantities knowing the value of the delivery intervals. There are also some mathematical simplifications present when formulating the problem in this way. Additional discussion concerning the reasons for choosing delivery intervals to be the decision variables rather than the delivery quantities can be found in Maxwell and Muckstadt [6].

We also stated that the economic delivery intervals must be chosen so that a

consistent flow of material will result. By this we mean that the delivery interval between the receipt of a shipment from an external supplier and between successive stages in the assembly plant should be planned so that schedules can be effectively implemented. For example, a solution is not practical if a delivery interval for a component from an external supplier is $\sqrt{5}$ times as long as the delivery interval of that component from the main warehouse to the shop floor. Therefore, we have limited attention to a particular class of policies.

To insure that the economic delivery quantities are both consistent and realistic, the class of policies we consider are nested and stationary. By stationary we mean that the time between deliveries from one stage to the next is always the same. By nested we mean that a delivery will not arrive at a stage unless a delivery is also made from that stage to a successor stage. Nested policies are used in practice to prevent the build up of in-process inventories. While stationary and nested policies need not be optimal in more general situations [7,11], they are appropriate for the situation we model since we assume a constant and continuous consumption rate for each component and assume a serial flow for a component through the assembly plant [5]. These and other assumptions will be subsequently discussed more completely. For certain extensions of the present model, the nestedness assumption may be inappropriate. Non-nested policies can be introduced using ideas presented by Maxwell and Muckstadt [6] and Roundy [9].

We also restrict attention to solutions for which the delivery interval for a component to a given stage must be $1, 2, 4, 8, \dots, 2^k$ times as long as the delivery interval for that component to a successor stage, where k is a non-negative integer. Restricting attention to such policies greatly simplifies the detailed scheduling problem for component deliveries to and throughout a plant [6]. Furthermore, limiting the solutions to this class of policies has little affect on the objective function value. For unconstrained distribution type system structures including serial systems Roundy [9] and Maxwell and Muckstadt [6] have proven that by restricting solutions to this class, the objective function can never be more than 6% higher than the value obtained by following any other policy. We show that a similar result holds for the constrained problem considered here.

Finally, we assume a base planning period exists for each component. This period can be an hour, shift, day, week, etc. Each component delivery interval must be a power of two multiple of this base planning period. The notion of a base planning period is frequently used in practice. For example, a certain minimal amount of inventory is often moved to a line station on a delivery. This amount

could correspond to a specific number of hours of work for a laborer. The length of a base planning period may also reflect the capabilities of an information system. We note that the base planning period can be set arbitrarily small if no constraints exist on the minimum length of the delivery intervals. Thus a variety of applications can be represented based on this concept.

The remainder of this paper is organized in the following way. In the next section we present the assumptions and nomenclature used in the paper. The model is stated in Section III and an algorithm is developed and analyzed in Section IV.

II. MODELING THE ECONOMIC DELIVERY INTERVAL PROBLEM

In this section we develop a model of the economic delivery interval problem described in Section I. The model is a planning model, which is used only to find delivery intervals for each type of component. Detailed scheduling of the exact arrival and shipping times from stage-to-stage cannot be accomplished using the model. Consequently only aggregate activity is considered in each stage of the system.

The model is based on the following assumptions.

1. The assembly rate for all final products is constant and continuous and equals λ_p units per year.
2. The material flow for component p can be represented by a graph G_p that corresponds to a serial system. The nodes in the graph represent storage areas and the arcs indicate the movement of component p from one storage area to the next. $N(G_p)$ and $A(G_p)$ represent the node and arc sets corresponding to G_p , respectively. The first and final nodes have special attributes. The first node in G_p is a receiving area through which all components must flow. The receiving area is designated node 1. Hence $1 \in N(G_p)$ for all p . The final node in G_p (the one without a successor node) is called a line station.
3. Each component is ultimately consumed (placed into an assembly) at only one line station. This assumption is made only to reduce notational complexity. Minor modifications to both the model and algorithm must be made if this and the preceding assumption are relaxed.
4. There is a fixed cost associated with the placement and receipt of a delivery from an external supplier for component p , call it K_p .
5. The cost of holding one unit of component p for one year is h_p .

6. Capacities exist for each storage area and the receiving area.

a) The total number of deliveries to the receiving area per year cannot exceed e .

b) The time available for operating each storage area per year is fixed.

We assume the time to process a lot of component p entering and leaving storage area i has both a fixed and variable component. The total annual average material flow through storage area i is proportional to $\sum_{p \in N(G_p)} \lambda_p$. Thus the total variable processing time is unaffected by the lot sizing/delivery interval decisions. Deducting this projected known time from the total available time results in an estimate of the total annual hours available for the fixed portion of the processing activity associated with placing units into and taking them out of storage area i . Let c_i be this amount and let a_{ip} and b_{ip} represent the fixed times required to process a lot of any size of component p into and out of storage area i , respectively, where a_{ip} and b_{ip} are also measured in hours.

c) Storage capacity is limited at each storage area, including line stations. Let s_i represent the storage capacity at area i at any point in time.

d) There is no storage capacity in the receiving area. Hence each arriving lot for component p moves immediately to storage area i , where $(1, i) \in A(G_p)$.

7. We assume a storage area at line station l is dedicated to only one component; furthermore, we assume that $a_{lp} = 0$ for $l \in N(G_p) \cap I_L$, where I_L is the set of line stations; that is, there is no fixed processing time at a line station.

8. Delivery intervals are restricted to be a powers of two multiple of some base planning period of length B .

9. Since we assume a serial nature for graph G_p , only nested policies are considered.

Additionally, let T_{ip} represent the delivery interval for component p (measured in years) into storage area i , $i \in N(G_p)$; $g_p = h_p \lambda_p / 2$, the average yearly holding cost for component p if $T_{1p} = 1$; and M_{ip} represent the powers of two multiple of the base planning period B per delivery interval for component p into storage area i , $i \in N(G_p)$. Note that since there is no storage capacity in

the receiving area (assumption 6d), $T_{lp} = T_{ip}$ for $(l,i) \in A(G_p)$. Let i_p represent the storage area i for which $(l,i_p) \in A(G_p)$.

Also, let d_{ip} represent the amount of storage capacity of area i , $i \in N(G_p)$, consumed per unit of component p . The value of d_{ip} depends on the storage system being used (e.g. dedicated storage for a component, random storage, etc.) in combination with the sequencing rules used to process the flow of components. Thus we allow the space factors to differ by storage area. For example, the factor would be 1 for the line station at which component p is used; however, it would have another value for a large automatic storage/retrieval system (ASRS). We assume the storage capacity consumed at storage area i for component p is measured proportional to the maximum inventory, that is, $\lambda_p(T_{ip} - T_{jp})$, where $(i,j) \in A(G_p)$. Thus if the entering and departing lots of component p at storage area i are of equal size, then no storage capacity is consumed. The storage capacity used at storage area i corresponding to a delivery interval plan is measured by

$$\sum_{p \ni i \in N(G_p)} d_{ip} \lambda_p (T_{ip} - T_{jp}).$$

and $(i,j) \in A(G_p)$

For notational simplicity, let $f_{ip} = d_{ip} \lambda_p$.

We close this section by describing an example system. The example illustrates the notion of a storage area and the structure of the graphs G_p . Suppose there are four components each of which is stored in storage area 1. Upon leaving this area, components 1 and 2 are sent to storage area 2 and components 3 and 4 are sent to storage area 3. From the storage areas, each component is sent to a line station. Figure 1 illustrates this system's structure. There are 7 storage areas.

Each component graph G_p is displayed in Figure 1. Then, G_1 has $N(G_1) = \{1,2,4\}$ and $A(G_1) = \{(1,2), (2,4)\}$, G_2 has $N(G_2) = \{1,2,5\}$ and $A(G_2) = \{(1,2), (2,5)\}$, G_3 has $N(G_3) = \{1,3,6\}$ with $A(G_3) = \{(1,3), (3,6)\}$, and G_4 has $N(G_4) = \{1,3,7\}$ and $A(G_4) = \{(1,3), (3,7)\}$.

III. THE MODEL OF THE ECONOMIC DELIVERY INTERVAL PROBLEM

In this section we state a model of the economic delivery interval problem (EDIP) that is based on the assumptions and notation discussed in Section II. We also present some observations concerning the model and its solution.

The EDIP can be stated as follows.

$$\min \sum_p \{K_p/T_{1p} + g_p T_{1p}\} = \min \sum_p \{K_p/T_{ip} + g_p T_{ip}\} \quad (1)$$

$$\text{subject to } \sum_p 1/T_{1p} \leq e, \quad (2)$$

(Receiving Area Constraint, Assumption 6d)

$$\sum_{p \ni i \in N(G_p)} (a_{ip}/T_{ip} + b_{ip}/T_{jp}) \leq c_i, \quad i \notin I_L, \quad (3)$$

and $(i,j) \in A(G_p)$

(Fixed Time Constraint for Storage Area i, Assumption 6b)

$$\sum_{p \ni i \in N(G_p)} f_{ip}(T_{ip} - T_{jp}) \leq s_i, \quad i \notin I_L, \quad (4)$$

and $(i,j) \in A(G_p)$

(Storage Capacity Constraint for Area i, Assumption 6c).

$$f_{ip} T_{ip} \leq s_i, \quad i \in N(G_p) \cap I_L, \quad (5)$$

(Line Station Storage Capacity, Assumption 7)

$$T_{ip} = M_{ip} \cdot B \quad i \in N(G_p) \quad (\text{Assumption 8}) \quad (6)$$

$$M_{ip} \geq M_{jp} \quad (i,j) \in A(G_p) \quad (\text{Assumption 9}) \quad (7)$$

$$M_{ip} \in \{1,2,4,8,\dots\}. \quad (\text{Assumption 8}) \quad (8)$$

EXAMPLE SYSTEM STRUCTURE

Figure 1

Since $T_{lp} = T_{i_p p}$, we can remove T_{lp} from the problems's formulation.

We note that the EDIP is a large-scale, non-linear integer programming problem. Consequently, it is not necessarily solved in an easy manner. Also observe that if $K_p = 0$ for all p , e is large and $a_{ip} = b_{ip} = 0$ for all i , then the EDIP is a linear integer program whose solution is $T_{ip} = B$. Thus if the fixed costs and times can be eliminated, or made relatively small, and the receiving area can accommodate $\sum_p 1/B$ shipments per year, the solution to the EDIP is immediate. In this case, virtually all storage area space could be eliminated. Furthermore, the solution in this situation indicates that the "Just-in-Time" principle is optimal. Thus Just-in-Time policies are optimal whenever fixed costs and times are inconsequential and the receiving area capacity is adequate. Note that if the fixed processing times and/or fixed procurement costs are not relatively small, then the optimal value for T_{ip} can be greater than B , that is, the Just-in-Time solution can be suboptimal.

IV. FINDING A SOLUTION TO THE EDIP

Rather than solving the EDIP directly, we first show how a solution to a relaxed version of this problem can be obtained. Then we discuss how the solution to this relaxed problem can be used to obtain values for the M_{ip} variables.

A. Relaxation of the EDIP

Suppose we relax the assumptions that M_{ip} must be a power of two and that a base planning period exists. The resulting relaxed problem, which we designate REDIP, can be stated as

$$\min \sum_p \{K_p/T_{i_p p} + g_p T_{i_p p}\}$$

subject to constraints (1) - (5) of the EDIP and $T_{ip} \geq T_{jp} \geq 0$, $(i,j) \in A(G_p)$.

This problem is also not easy to solve. We propose obtaining its solution using an infinitely convergent procedure, each iteration of which consists of two steps. The first step requires solving a Lagrangian relaxation of the REDIP; the second step obtains new estimates for the multiplier values.

The Lagrangian relaxation used in the first step is obtained by dualizing with

respect to certain capacity constraints. Let γ be the multiplier corresponding to the receiving area constraint, α_i be the multiplier for the storage area i ($i \notin I_L$) fixed time constraint, and β_i be the multiplier corresponding to the area i storage space capacity constraint ($i \notin I_L$). Then the relaxation, which we call the LREDIP, is

$$\begin{aligned} \min \quad & \sum_p \left\{ (K_p + \gamma)/T_{ip} + g_p T_{ip} \right\} \\ & + \sum_p \left\{ \sum_{\substack{i \geq 2 \\ i \in N(G_p), i \notin I_L \\ (k,i) \in A(G_p)}} [(\alpha_k b_{kp} + \alpha_i a_{ip})/T_{ip} + (\beta_i f_{ip} - \beta_k f_{kp}) \cdot T_{ip}] \right\} \end{aligned}$$

subject to

$$T_{ip} \geq T_{jp} \geq 0, (i,j) \in A(G_p),$$

$$T_{ip} \leq s_i/f_{ip}, i \in N(G_p) \cap I_L,$$

where, by definition, $\alpha_1 = b_{1p} = \beta_1 = 0$. This LREDIP is chosen, as we will subsequently show, because it is easily solved given the values of γ , α_i and β_i .

To simplify notation, let

$$\bar{K}_{ip} = K_p + \gamma + \alpha_i a_{ip}; \quad \bar{g}_{ip} = g_p + \beta_i f_{ip}, \quad i = i_p, i \geq 2,$$

$$\bar{K}_{ip} = (\alpha_k b_{kp} + \alpha_i a_{ip}); \quad \bar{g}_{ip} = \beta_i f_{ip} + \beta_k f_{kp}, \quad i \neq i_p, i \geq 2, (k,i) \in A(G_p),$$

$$\bar{s}_i = s_i/f_{ip}, \quad i \in N(G_p) \cap I_L,$$

where $\alpha_i = \beta_i = 0, i \in I_L$. Then, for fixed values of γ , α_i , and β_i , the LREDIP can be stated as

$$\begin{aligned} \min \quad & \sum_{i \geq 2, p} \{ \bar{K}_{ip}/T_{ip} + \bar{g}_{ip} T_{ip} \} \\ & T_{ip} \geq T_{jp} \geq 0, (i,j) \in A(G_p), \\ & T_{ip} \leq \bar{s}_i, \quad i \in N(G_p) \cap I_L. \end{aligned}$$

Observe that this problem is separable by component since the constraints reflect the precedence and line station capacity conditions by component. Furthermore, by assumption, the precedence relationships for component p can be represented by a graph of a serial system. We next show how this mathematical structure can be exploited to find an optimal solution to the LREDIP. To reduce notation, we temporarily drop the subscript p in the discussion in the next subsection.

B. Finding a Solution to the LREDIP

In this section we show how a solution to the LREDIP can be obtained. Since the LREDIP is separable by component, its solution can be found by solving p independent problems. We show in two steps how an optimal solution can be determined for each component.

In the first step, we consider the problem

$$\begin{aligned} \min \quad & \sum_{i=2}^n \{ \bar{K}_i / T_i + \bar{g}_i T_i \} \\ & T_i \geq T_j \geq 0, (i,j) \in A(G), \end{aligned} \quad \text{Problem 1}$$

where G is the graph of the serial system corresponding to component p . We begin by showing the relationship between a solution to Problem 1, and the ordered partitions of the graph G . The notion of an ordered partition is developed in Jackson, Maxwell and Muckstadt [2] and is discussed in the following subsection.

In the second step, we establish the relationship between the solution to Problem 1 and an optimal solution to the following problem, which we call Problem 2.

$$\begin{aligned} \min \quad & \sum_i \{ \bar{K}_i / T_i + \bar{g}_i T_i \} \\ & T_i \geq T_j, (i,j) \in A(G), \\ & T_\ell \leq \bar{s}_\ell. \end{aligned} \quad \text{Problem 2}$$

We show how the solution to Problem 1 can be used to find a solution to Problem 2 and then present an algorithm for finding the solution to this problem and hence to the LREDIP.

B.1 Properties of an Optimal Solution to Problem 1

Define a subgraph G' of the graph G to consist of a subset $N(G')$ of the node set $N(G)$ together with the associated arc set $A(G')$, where $(i,j) \in A(G')$ if and only if $i \in N(G')$, $j \in N(G')$, and $(i,j) \in A(G)$. An ordered collection of subgraphs $(G_1, G_2, \dots, G_N)$ of G is said to be ordered by precedence if for any $1 \leq \ell < k \leq N$ there does not exist a node $j \in N(G_\ell)$ and a node $i \in N(G_k)$ such that $(j,i) \in A(G)$. That is, no node in $N(G_\ell)$ precedes any node in $N(G_k)$ if $k > \ell$. We say that a collection of subgraphs $(G_1, G_2, \dots, G_N)$ forms an ordered partition of the graph G if

- (a) the node subsets $N(G_1), N(G_2), \dots, N(G_N)$ form a partition of the node set $N(G)$, and
- (b) the subgraphs are ordered by precedence.

A directed cut of a subgraph G' is simply an ordered (binary) partition (G'^+, G'^-) of the subgraph G' .

Suppose we have an arbitrary feasible solution to Problem 1 in which there are N distinct delivery intervals in the solution. Each delivery interval corresponds to a subset of the storage areas. Each storage area in that subset shares a common delivery interval. These subsets partition $N(G)$. Define $G_1, \dots, G_N$ to be the subgraphs of G corresponding to the subsets. Note that $G_1, \dots, G_N$ must form an ordered partition of the graph G . Furthermore, the solution to Problem 1 can be optimal only if the delivery interval for G_k , call it $T(G_k)$, satisfies $T(G_k) = (K(G_k)/g(G_k))^{1/2}$, $k = 1, \dots, N$, where $K(G_k) = \sum_{i \in N(G_k)} \bar{K}_i$ and $g(G_k) = \sum_{i \in N(G_k)} \bar{g}_i$.

Suppose $(G_1, \dots, G_N)$ is an ordered partition of G with $T(G_k) = (K(G_k)/g(G_k))^{1/2}$ and $T_i = T(G_k)$ for all $i \in N(G_k)$, $k = 1, \dots, N$. Then the solution $\{T_i\}$ is a feasible solution to Problem 1 if $T(G_1) \leq T(G_2) \leq \dots \leq T(G_N)$; that is, if

$$\frac{K(G_1)}{g(G_1)} \leq \frac{K(G_2)}{g(G_2)} \leq \dots \leq \frac{K(G_N)}{g(G_N)}.$$

The relationship between an optimal solution to Problem 1 and ordered partitions of the graph G is specified by the following theorem that is proved in Jackson, Maxwell and Muckstadt [2].

Theorem 1: Consider an arbitrary collection of N delivery intervals, $T(G_1), \dots, T(G_N)$. The following three conditions are necessary and sufficient for these delivery intervals to correspond to an optimal solution to Problem 1:

- (1) There exists an ordered partition $(G_1, \dots, G_N)$ of the graph G such that $T(G_k) = (K(G_k)/g(G_k))^{1/2}$ for each $k = 1, \dots, N$;
- (2) The sequence of ratios $\{K(G_k)/g(G_k); k = 1, \dots, N\}$ is nondecreasing; and
- (3) For each $k = 1, \dots, N$, there does not exist a directed cut (G_k^-, G_k^+) of G_k such that

$$\frac{K(G_k^-)}{g(G_k^-)} < \frac{K(G_k^+)}{g(G_k^+)}.$$

Properties (2) and (3) are called the convexity property and concavity property of an optimal solution, respectively.

B.2 Finding an Optimal Solution to Problem 1

The optimal solution to Problem 1 is particularly easy to find due to the serial structure of the graph G . We will not present an algorithm here. Procedures are described in Maxwell and Muckstadt [6], Jackson and Roundy [3], or Schwarz and Schrage [10] that can be used to solve this problem. We note that due to the serial nature of G , the solution time is proportional to the number of nodes in G .

B.3 The Relationships Between Problem 1 and Problem 2

Suppose we have an optimal solution to Problem 1. This implies that we have found an ordered partition $G_1, \dots, G_N$. Furthermore, since line station ℓ is the final node in the serial graph G , assume $\ell \in N(G_1)$ and $T_\ell = T(G_1) = \bar{K}(G_1)/\bar{g}(G_1)$. If $T_\ell \leq \bar{s}_\ell$, then this solution to Problem 1 is also an optimal solution to Problem 2. However, if $T_\ell > \bar{s}_\ell$, then this solution is not feasible to Problem 2. As we establish in the following theorem, even in this case the solution to Problem 1 is the basis for a solution to Problem 2.

Theorem 2: Suppose $G_1, \dots, G_N$ is an optimal partition to Problem 1 and $T_\ell > \bar{s}_\ell$. Then there exists an ordered partition of G_1 , say $G_1^1, \dots, G_1^Q$ having the following properties:

- (a) $T(G_1^1) > \bar{s}_\ell$.

- (b) For any directed cut of G_1^1 , say (G_1^{1+}, G_1^{1-}) , such that $\ell \in N(G_1^{1-})$,

$$T(G_1^{1+}) \leq \bar{s}_\ell.$$

- (c) $\bar{s}_\ell < T(G_1^2) < T(G_1^3) < T(G_2) < \dots < T(G_1^Q) < \dots < T(G_N)$

and

- (d) For each $k = 2, \dots, Q$, for any directed cut (G_1^{k+}, G_1^{k-}) ,

$$T(G_1^{k+}) < T(G_1^{k-}).$$

Furthermore, an optimal solution to Problem 2 is given by

$$T_i = \bar{s}_\ell, \quad i \in N(G_1^1),$$

$$T_i = T(G_1^k), \quad i \in N(G_1^k), \quad k = 2, \dots, Q,$$

and

$$T_i = T(G_j), \quad i \in N(G_j), \quad j = 2, \dots, N.$$

The proof of this theorem is lengthy and for that reason is omitted.

Observe that the above theorem does not rely on the serial nature of G .

However, the following specialized algorithm for finding G_1^1 does take advantage of the serial structure of G_1 .

Suppose there are n elements in $N(G_1)$. For ease of notation assume the nodes in this serial system are numbered 1 through n , where n corresponds to the appropriate line station. The algorithm builds a subgraph G_1^1 having the properties stated in Theorem 2. Note that a subgraph in a serial system can be described by two nodes; i.e. G' is a subgraph iff $N(G') = \{j, j+1, \dots, k\}$ for $j \leq k$. The algorithm searches for a j^* such that $N(G_1^1) = \{j^*, \dots, \ell\}$.

PARTITIONING ALGORITHM FOR
FINDING G_1^1 (Algorithm 1)

Step 1. Set $N(G_1^1) = \{\ell\}$

Step 2. Set $j \leftarrow \ell-1$
 $k \leftarrow \ell-1$

Step 3. While $(j \geq 1)$

{If $\sum_{i=j}^k (\bar{K}_i / \bar{g}_i) \leq s_\ell$, then $N(G_1^1) \leftarrow N(G_1^1) \cup \{j, j+1, \dots, k\}$

$k \leftarrow j-1$

$j \leftarrow j-1$

}

else $j \leftarrow j-1$

}

End

B.4 An Algorithm for Finding an Optimal Solution to Problem 2

By combining the results developed in the preceding portions of this section, it is clear that the following algorithm finds an optimal solution to Problem 2 and hence can be used to find an optimal solution to the LREDIP.

ALGORITHM FOR FINDING THE
OPTIMAL SOLUTION FOR PROBLEM 2
(Algorithm 2)

Step 1. Find an optimal solution to Problem 1, which establishes an optimal partition $G_1, \dots, G_N$.

Step 2. Using Algorithm 1, find the graph G_1^1 .

Step 3. Find the optimal partition of $G_1 - G_1^1$ using the same algorithm as used in Step 1.

Step 4. Determine the optimal reorder intervals as follows:

$$T_i = \bar{s}_\ell, \quad i \in N(G_1^1),$$

$$T_i = \left(\sum_{j \in N(G_1^k)} \bar{K}_j \right) / \left(\sum_{i \in N(G_1^k)} \bar{g}_j \right), \quad i \in N(G_1^k), \quad k = 2, \dots, Q,$$

$$T_i = \left(\sum_{j \in N(G_k)} \bar{K}_j \right) / \left(\sum_{j \in N(G_k)} \bar{g}_j \right), \quad i \in N(G_k).$$

V. A SOLUTION METHOD FOR THE EDIP

In this section we present a two step method for obtaining a solution to the EDIP. First, we describe a procedure that finds a solution to the REDIP. This method depends heavily on the results developed in the preceding section. Second, we apply an optimal powers of two rounding rule for unconstrained problems to generate a solution to the EDIP. We then briefly discuss the quality of the rounded solution.

A. Finding a Solution to the REDIP

The procedure we propose for solving the REDIP has two steps. The first step assumes the existence of values for the Lagrangian multipliers γ , α_i , and β_i . Given these values, delivery intervals are established for each component using Algorithm 2. If the solution satisfies the constraints of the EDIP (i.e., if the left and right hand side values for each active constraint do not differ by more than $\epsilon > 0$), then the procedure terminates. If not, step 2 is employed in which new estimates of the multiplier values are calculated. Their values are found using a large scale programming technique called column generation. References 1 and 4 contain a complete discussion of this method. We briefly summarize the idea behind the procedure.

Each time the second step is entered, the column generation technique requires solving a linear program. The rows of the linear program correspond to the constraints on space and fixed time availability for the receiving area and all storage areas except the line stations. Each column represents the space and time requirements corresponding to a solution for Problem 2 for all components generated in Step 1. The variables in the linear program are weights that correspond to each column, or proposed solution generated in Step 1. One additional constraint exists in the formulation of the linear program that forces the sum of the weights to equal 1. The solution found to this linear program at each iteration yields a feasible solution to the REDIP (we assume the procedure is started with a known feasible solution) and a solution to its dual, that is, it yields an estimate of γ , α_i , and β_i .

The line station space constraints are not included in the Step 2 linear program. Recall that the linear program assigns optimal weights to each proposed

solution so that feasibility to the REDIP is assured. Since all solutions obtained from Step 1 satisfy the line station space constraints, any convex combination of them also satisfies these constraints. Hence these constraints are not needed in the formulation of the linear program.

We now state the algorithm for obtaining a solution to the REDIP. The algorithm finds an ϵ optimal solution. This procedure guarantees convergence of the multipliers to their optimal values. However, convergence is infinite. A complete discussion of the convergence properties of this type of algorithm can be found in References 1 and 4.

ALGORITHM FOR FINDING AN ϵ -OPTIMAL
SOLUTION TO THE REDIP
(Algorithm 3)

- Step 0. Generate initial values for γ , α_i , β_i , and a feasible solution for the REDIP.
- Step 1. (a) Given the values of γ , α_i and β_i , find delivery intervals for each storage area for each component using Algorithm 2.
- (b) For each constraint, compute the capacity utilization ratio corresponding to the solution generated in the first portion of this step. The utilization ratio measures capacity used corresponding to the proposed solution divided by the available capacity. Let U_i measure the fixed time constraint utilization ratios for area i , $i \notin I_L$ and V_i measure the utilization ratio for the area i storage capacity constraint, $i \notin I_L$. For $\epsilon > 0$, if $U_i - 1 < \epsilon$ for all i for which $\alpha_i = 0$ (and $\gamma = 0$) and $V_i - 1 < \epsilon$ for all i for which $\beta_i = 0$ and $|U_i - 1| < \epsilon$ for all i for which $\alpha_i > 0$ ($\gamma > 0$) and $|V_i - 1| < \epsilon$ for i for which $\beta_i > 0$, then stop, the solution is ϵ -optimal; otherwise go to step 2.
- Step 2. Find new values for γ , α_i and β_i . Generate a new column and weighting variable for the linear program. The components of this column reflect the respective total capacities required corresponding to the solution generated for all components in step 1. The final component is a 1, which appears in the weighting constraint. The new column has the following form.

$$\begin{bmatrix} \text{(total receiving area capacity consumed)} \\ \text{(storage area 1 fixed time capacity used)} \\ \text{(storage area 1 storage space used)} \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix}$$

Solve the linear program with the addition of this new column. Obtain new estimates for γ , α_i and β_i and return to Step 1.

The above algorithm works well for practical problems in which a component passes through only a few storage areas. The time required to solve the step 1 problem is proportional to $P \cdot J$, where J is the maximum number of storage areas a component actually enters and P is the number of components. Consequently, Step 1 can be accomplished very quickly. The total number of storage area constraints present in the Step 2 linear program should rarely exceed 25. Thus the basis and its inverse can be computed easily. Furthermore, the solution to the Step 2 linear program can often be obtained in one or a few pivots so that Step 2 is also accomplished rapidly. The number of iterations will, of course, vary depending on the problem and the selection of initial values for the multipliers and the quality of an available initial feasible solution to the REDIP.

We have assumed that an initial feasible solution can be obtained relatively easily. In real situations we have examined this has been possible. However, if one cannot be found quickly, the proposed algorithm can still be used by adding artificial variables to the formulation of the linear program used in Step 2. More iterations will most likely be required when an initial feasible solution is not available.

B. Finding an Approximate Solution to the EDIP

The solution obtained to the REDIP need not satisfy the powers of two restrictions stipulated in the formulation of the EDIP. The solution to the REDIP yields optimal values for the multipliers γ^* , α_i^* and β_i^* and values for the delivery intervals, T_{ip} . Assume $\bar{K}_{ip}$ and $\bar{g}_{ip}$ are computed using γ^* , α_i^* and β_i^* . Then $\{T_{ip}\}$ solves

$$\min \sum_{i,p} (\bar{K}_{ip}/T_{ip} + \bar{g}_{ip}T_{ip})$$

$$T_{ip} \geq T_{jp} \geq 0, (i,j) \in A(G_p),$$

(Problem 3)

$$T_{lp} \leq \bar{s}_l, l \in N(G_p) \cap I_L.$$

Assume, without loss of generality, that $\bar{s}_l$ is set so that it is a power of two multiple of B.

The power of two solutions T_{ip}^* and $T^*(G_{kp})$ can be obtained as follows. For component p, if $T_{lp} = \bar{s}_l$, then set $T_{ip}^* = \bar{s}_l$ for all i for which $T_{ip} = T_{lp}$, $i \in N(G_p)$, in the solution of Problem 3. Corresponding to the solution to Problem 3 there is an ordered partition of the graph for component p, G_p , which we designate $G_{1p}, \dots, G_{N_p p}$. Let $i \in N(G_{kp})$ for which the corresponding delivery interval $T(G_{kp})$ is greater than $\bar{s}_l$. Then, as shown by Maxwell and Muckstadt [6], the optimal delivery interval for component p and storage area i is the smallest power of 2, M_{ip} , for which

$$M_{ip} \geq \frac{1}{B} \sqrt{\frac{\frac{1}{2} \left(\sum_{i \in N(G_{kp})} \bar{K}_{ip} \right)}{\sum_{i \in N(G_{kp})} \bar{g}_{ip}}}.$$

Then $T_{ip}^* = T^*(G_{kp}) = M_{ip} \cdot B$ for $i \in N(G_{kp})$. Assume $M_{ip} = 2^{n_{ip}}$, $n_{ip} \in \{0, 1, \dots\}$. It is easy to show that in the worst case the cost of the powers of two solution cannot exceed the value found from the solution to the REDIP by more than roughly 6%.

Let us now turn our attention to the feasibility of the powers of two solution. It is easy to prove that $\sqrt{2} \cdot T(G_{kp}) \geq T^*(G_{kp}) \geq T(G_{kp})/\sqrt{2}$. A worst case analysis shows that a powers of two solution cannot exceed capacity by more than 41%, that is, the solution could use up to $\sqrt{2}$ times the available capacity. The worst case occurs for the fixed time constraints when all $T^*(G_{kp}) = \sqrt{2} T(G_{kp})$ and for the storage area constraints only if each $T^*(G_{kp})$ assumes its lowest possible value, a highly unlikely event. If the total number of component partitions is large, we would expect that the powers of two solution should consume about the same amount of capacity as the solution to the relaxed problem, the REDIP.

Assume $T^*(G_{kp})$ is uniformly distributed over the interval $[T(G_{kp})/\sqrt{2}, T(G_{kp}) \cdot \sqrt{2}]$. Then for the fixed time constraints that were active in the solution to

the REDIP,

$$E \left[\sum_{\substack{p \supset i \in N(G_p) \\ \text{and } (i,j) \in A(G_p)}} \{a_{ip}/T_{ip}^* + b_{ip}/T_{jp}^*\} \right] = \sum_{\substack{p \supset i \in N(G_p) \\ \text{and } (i,j) \in A(G_p)}} \left\{ a_{ip} E[1/T_{kp}^*] + b_{ip} E[1/T_{jp}^*] \right\}$$

$$= c_i \sqrt{2} \log 2.$$

If the $T^*(G_{kp})$ are independent, then the strong law of large numbers implies that the limit of the left hand side of the active fixed time constraints approaches $.98c_i$ as the number of components and partitions grows large. Hence for a large number of components we would expect the solution to be feasible.

The storage area capacity use that results from the powers of two rounding can be analyzed similarly. Take the expectation of the left hand side capacity use for an active storage area i constraint, $i \notin I_L$.

$$E \left[\sum_{\substack{p \supset i \in N(G_p) \\ \text{and } (i,j) \in A(G_p)}} f_{ip}(T_{ip}^* - T_{jp}^*) \right] = \sum_{\substack{p \supset i \in N(G_p) \\ \text{and } (i,j) \in A(G_p)}} \left\{ f_{ip} E[T_{ip}^*] - E[T_{jp}^*] \right\}$$

$$= 3/4 \cdot \sqrt{2} \cdot s_i \approx 1.06s_i.$$

Again we can apply the strong law of large numbers. The analysis shows that for a large number of components we would expect to use about 106% of the available capacity and therefore would expect the power of two solution to exceed the storage area constraints. However, by appropriately adjusting the value of s_i , that is, by setting s_i equal to 94% of the actual capacity, a feasible solution would be expected.

In summary we have seen that the power of two solution cannot increase the objective function by more than 6% above the value obtained when solving the REDIP. Furthermore, for a large number of components we expect the power of two solution to yield a feasible solution to the problem assuming the s_i values are properly adjusted.

ACKNOWLEDGEMENTS

I would like to thank Professors Peter Jackson and William Maxwell for their assistance during the development of this paper.

REFERENCES

- [1] Dantzig, G.B., Linear Programming and Extensions, Princeton University Press, Princeton, NJ, 1963.
- [2] Jackson, P.L., W.L. Maxwell and J.A. Muckstadt, "Determining Optimal Reorder Intervals in Capacitated Production-Distribution Systems," Technical Report 624, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, February 1984.
- [3] Jackson, P.L. and R.O. Roundy, "Constructive Algorithms for Planning Production in Multi-Stage Systems," Technical Report 632, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, August 1984.
- [4] Lasdon, L.S., Optimization Theory for Large Systems, MacMillan, New York, NY, 1970.
- [5] Love, S.F., "A Facilities in Series Inventory Model with Nested Schedules," Management Science, Vol. 18, No. 5, January 1972.
- [6] Maxwell, W.L. and J.A. Muckstadt, "Establishing Consistent and Realistic Reorder Intervals in Production-Distribution Systems," Technical Report 561, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, January 1983.
- [7] Muckstadt, J.A. and H.M. Singer, "Comments on Single Cycle Continuous Review Policies for Arborescent Production/Inventory Systems," Management Science, Vol. 24, No. 16, December 1978.
- [8] Peterson, R. and E.A. Silver, Decision Systems for Inventory Management and Production Planning, John Wiley & Sons, New York, NY, 1979.
- [9] Roundy, R.O., "98%-effective Integer-ratio Lot sizing for One-warehouse Multi-retailer Systems," Technical Report No. 35, Department of Operations Research, Stanford University, Stanford, CA, September 1983.
- [10] Schwarz, L.B. and Schrage, L., "Optimal and System Myopic Policies for Multi-Echelon Production/Inventory Assembly Systems," Management Science, Vol. 21, No. 11, July 1975.
- [11] Williams, J.F., "On the Optimality of Integer Lot Size Ratios in Economic Lot Size Determination in Multi-Stage Assembly Systems," Management Science, Vol. 28, No. 11, November 1982.