SCHOOL OF OPERATIONS RESEARCH
AND INDUSTRIAL ENGINEERING
COLLEGE OF ENGINEERING
CORNELL UNIVERSITY
ITHACA, NEW YORK  14850

TECHNICAL REPORT NO. 674

September 1983
revised August 1985

94%-EFFECTIVE LOT-SIZING IN
MULTI-STAGE ASSEMBLY SYSTEMS

by

Robin Roundy

This research was supported by NSF Grant ECS-8404641.
TECHNICAL REPORT NO. 674

September 1983
Revised August 1985

94%-EFFECTIVE LOT-SIZING IN
MULTI-STAGE ASSEMBLY SYSTEMS

By

Robin Roundy
94\% - EFFECTIVE LOT-SIZING IN MULTI-STAGE ASSEMBLY SYSTEMS.

by Robin Roundy

September 1983
revised August 1985.

Introduction.
In the last few years, a number of heuristics for constant-demand, multi-stage lot-sizing models have been shown to have very high worst-case effectiveness. Unfortunately the model that I consider to be the most appropriate for instructional purposes has never been published in a self-contained form. The purpose of this paper is to present this model with the associated algorithm proof as an aid to instructors and students.

A brief review of the other constant-demand lot-sizing models with high worst-case effectiveness and their history is given at the end of this paper. For a more thorough discussion of the relevant literature, the reader is referred to [Ro1984].

Model and Summary of Results.
We consider a one-product, multi-stage assembly system in continuous time. Both in-process and finished goods are referred as components. The unique component that has positive external demand is called the final product. The external demand for this component is constant and must be met without shortages or backlogging. Each of the other components in the system is assumed in the production of a unique direct successor component.

When an order is placed for a component it is instantly delivered, and the required amounts of the components consumed in
producing the given component are simultaneously withdrawn from their respective inventories. The assumption of zero lead times is not essential and is made principally for convenience [Ro1984, SS1975].

There are nonnegative setup costs and linear holding costs associated with each of the components. The goal is to schedule orders over the infinite horizon so as to minimize the total setup-plus-holding cost.

A power-of-two policy is a policy in which each component is ordered at time zero and at equal intervals of time thereafter, and these intervals are all powers of two times a common base planning period. We show how to find a good policy within this class in $O(N \log N)$ time where $N$ is the number of components. We then prove that the cost of this policy is within 6% of the cost of any feasible solution to the original problem. It is also possible to find a policy whose cost is within 2% of optimal in $O(N \log N)$ time [Ro1984].

Nested Power-of-Two Policies and their Costs.

Let $s_n$ be the direct successor of component $n$, $p_n$ be the set of direct predecessors of component $n$, $S_n$ be the set composed of $n$ and all of its predecessors, direct or indirect, and let $P_n$ be similarly defined. We assume without loss of generality that if $m \in p_n$, one unit of $m$ is consumed for each unit of $n$ ordered.

The echelon holding cost $h_n$ for component $n$ is the amount by which the holding cost of component $n$ exceeds the sum of the holding costs of its direct predecessors. The concept of an
Echelon holding cost is closely related to the value added at an operation. It is assumed that the echelon holding costs of all of the components are positive. The echelon inventory $E_n^t$ of component $n$ at time $t$ is the sum of the inventories of all components $m, m \in S_n$. It is easily verified that the total rate at which holding costs are being incurred at time $t$ is $\sum_n h_n E_n^t$.

Let the components of the assembly system be numbered 1 through $N$. We denote a power-of-two policy by $J = (T_n', 1 \leq n \leq N)$ where $T_n'$ is the interval between orders for component $n$. A nested policy is a policy in which an order is placed for the successor of component $n$ simultaneously with every order for $n$. The power-of-two policy $J$ is nested if and only if

$$T_n \geq T_{S_n}$$

for all $n$. \hspace{1cm} (1)

It is easy to show that an optimal power-of-two policy is nested and has the following property:

Property 1. Orders are placed for component $n$ only when the inventory of $n$ is zero.

In the sequel we will restrict attention to nested power-of-two policies that satisfy property 1. We also assume without loss of generality that the demand rate is 2. Therefore the echelon
inventory of component $n$ follows the traditional sawtooth pattern, i.e., $E_n^t = 2 - f(t/T_n)$ where $f(x)$ is twice the fractional part of $x$. This implies that the average echelon inventory of component $n$ is $T_n$, and that the average cost of $J$ can be written as

$$c(J) = \sum_n \left( K_n/T_n + h_n T_n \right). \tag{2}$$

**Computing Approximate Reorder Intervals and Clusters.**

The problem of finding an optimal power-of-two policy can therefore be written as:

$$\text{minimize: } \quad c(J)$$

such that:

$$T_n \geq T_{s_n} \quad \text{for all } n, \text{ and} \tag{1}$$

$$T_n = 2^{p_n \beta}, \quad p_n \text{ integer, } \beta > 0. \tag{3}$$

The first step in our heuristic is relaxing (3) and solving

$$(P) \quad \text{minimize: } \quad c(J)$$

such that:

$$T_n \geq T_{s_n} \quad \text{for all } n, \text{ and} \tag{1}$$

This is done for two reasons. First, it will give us a set of approximate reorder intervals that we will use in our heuristic. Second, it will give us a lower bound on the average cost of an
arbitrary schedule for this problem. The lower bound will be used in evaluating our heuristic.

The solution to (P) will divide the system into subsets called clusters, sets of components whose costs induce them to place orders simultaneously. The Kuhn-Tucker conditions for (P) are:

\[-K_n/T_n^2 + h_n - \lambda_n + \sum_{m \in \mathcal{P}_n} \lambda_m = 0 \tag{4}\]

\[\lambda_n \geq 0 \text{ and complementary slack with } T_n - T_{s_n} \geq 0. \tag{5}\]

Given a solution to (4) and (5), we define the clusters as follows. If \( \lambda_n > 0 \) then components \( n \) and \( s_n \) are in the same cluster. Otherwise they are in different clusters. Clearly a cluster is a connected subsystem of the assembly system.

The root of a connected subsystem of the assembly system is the unique component in the subsystem that is closest to the final product. Let \( C^i \) be the set of components in the cluster whose root is component \( i \). Let \( K^i = \sum_{n \in C^i} K_n \) be the total setup cost for cluster \( i \), and \( H^i = \sum_{n \in C^i} h_n \) be the total holding cost for cluster \( i \). For \( n \in C^i \) we define \( k_n = \sum_{n \in \mathcal{P}_n \cap C^i} K_n \) and \( h_n = \sum_{n \in \mathcal{P}_n \cap C^i} h_n \). If \( i \neq n \in C^i \) and we were to split cluster \( i \) in two by letting \( n \) be the root of a new cluster, the total setup cost of the new cluster rooted at \( n \) would be \( k_n \) and the total holding cost would be \( h_n \).
Lemma. The following is equivalent to (4), (5): There is a partition of the components into clusters \( C^i \) that satisfies the following conditions.

\( C^i \) is a connected subsystem of the assembly system with root \( i \).

\[ T_n = T^i \text{ for all } n \in C^i \text{ and for all clusters } i. \]  
\[ (6) \]

\[ T^i = (K^i/H^i)^{\frac{5}{2}} \text{ for all clusters } i. \]  
\[ (7) \]

\[ K^i/H^i \geq K^j/H^j \text{ whenever } s_i \in C^j. \]  
\[ (8) \]

\[ \kappa_n/\nu_n < K^i/H^i \text{ for all } n \in C^i. \]  
\[ (9) \]

Proof. We first show that (4), (5) imply (6) - (9). Given a solution to (4) and (5), let clusters \( C^i \) be defined as above. Equation (6) is a consequence of the complementary slackness property (5). Summing (4) over all \( n \in C^i \) gives us (7), which is the standard Wilson square-root formula. Equation (8) is a direct consequence of (5). On summing (4) over all \( m \in P_n \cap C^i \), and on substituting \( (K^i/H^i)^{\frac{5}{2}} \) for \( T_m \), we get

\[ -\kappa_n/(K^i/H^i) + \nu_n - \lambda_n = 0. \]  
\[ (10) \]

Since \( \lambda_n > 0 \), (10) implies (9).

We now show that (6) - (9) imply (4), (5). Let \( \lambda_n \) be given by (10) for all \( n \), and note that \( \lambda_n = 0 \) for all \( n \) that are...
roots of clusters. Clearly (6) - (8) imply (5). Let $F(n)$ be the left-hand side of (10). If $n \in \mathbf{C^i}$ and we subtract $F(m)$ from $F(n)$ for all $m \in p_n \cap \mathbf{C^i}$, we get (4). \[ \square \]

Equation (9) states that if we try to split cluster $i$ in two by letting $n$ be the root of a new cluster, the constraint $T_n \geq T_{s_n}$ will be violated. To see this, note that the total setup cost of the piece of cluster $i$ that does not contain $n$ is $K^i - \kappa_n$ and the total holding cost is $H^i - \nu_n$. But $\kappa_n/\nu_n < K^i/H^i$ implies that $T_n^2 = \kappa_n/\nu_n < (K^i - \kappa_n)/(H^i - \nu_n) = T_{s_n}^2$.

We are now ready to give an algorithm for solving (P). This algorithm is known as the minimum-violators algorithm and was developed to solve a problem in statistics. It is given below.

The Minimum Violators Algorithm

**Step 1:** Create a separate cluster for each component in the system. Let the set $S$ initially consist of all clusters other than the one containing the final product. The set $S$ is the set of clusters $\mathbf{C^i}$ for which we do not know that $K^i/H^i \geq K^j/H^j$ when $s_i \in \mathbf{C^j}$.

**Step 2:** If $S$ is empty, the current clusters are optimal. Go to Step 4. Otherwise find a cluster $i \in S$ for which $K^i/H^i$ is minimal. In the event that there is a tie, choose a cluster $\mathbf{C^i}$ for which $|S_i|$ is minimal. Let $s_i \in \mathbf{C^j}$. 
Step 3. Remove i from S. If $K^i/H^i \geq K^j/H^j$, or if i is
the final product, go to Step 2. If $K^i/H^i < K^j/H^j$ then
collapse cluster i into cluster j and go to Step 2.

Step 4. Calculate $T_n$ for all n using (6) and (7).

Lemma 2. At each iteration of the algorithm,

$$K^k/H^k \geq K^\emptyset/H^\emptyset \quad \text{for all} \quad k \in S, \emptyset \not\in S. \quad (11)$$

Furthermore the value of $\min_{m \in S} K^m/H^m$ is nondecreasing as the
algorithm progresses.

Proof. The proof is by induction on the iterations of the
algorithm. Clearly (11) holds on the first iteration. Assume
that Step 3 is about to be executed and that (11) holds. If
$K^i/H^i \geq K^j/H^j$ then (11) will hold on the next iteration, and
$\min_{m \in S} K^m/H^m$ will not decrease. If $K^i/H^i < K^j/H^j$ then $j \in S$
and $\min_{m \in S} K^m/H^m = K^i/H^i < (K^i + K^j)/(H^i + H^j)$. Therefore (11)
will hold on the next iteration and $\min_{m \in S} K^m/H^m$ will not
decrease. \[ \square \]

Theorem 1. The clusters produced by the Minimum Violators
Algorithm satisfy (6) - (9) and consequently solve (P).

Proof. Clearly (6) and (7) hold. If cluster i is removed
from S in Step 3, This fact implies that (8) holds for i and
the selection rule in Step 2 implies that j is not in S.
Lemma 2 and Step 3 imply that once a cluster has been removed from S it is not subsequently changed. Therefore (8) holds. We will prove (9) by showing that if \( k \in S \) and \( k \neq n \in C^k \) then

\[
\frac{k_n}{H_n} \leq \min_{m \in S} \frac{k_m}{H_m} \quad \text{if } k \in S \text{ and } k \neq n \in C^k.
\]  

(12)

This will imply that when the cluster containing n is removed from S, (9) holds for n.

The proof is by induction on the iterations of the algorithm. Suppose that \( n = i \) and that cluster \( i \) is about to be combined with cluster \( j \). Since \( K^i/H^i < K^j/H^j \), \( K^i/H^i < (K^i + K^j)/(H^i + H^j) \). Thus at the next iteration (12) will hold for n.

Now assume that at the current iteration (12) holds for n. There is no change in cluster k unless cluster i and cluster j are combined and either \( k = i \) or \( k = j \). By (12)

\[
\frac{k_n}{H_n} \leq \frac{k^i}{H^i} < \frac{K^j}{H^j}.
\]  

(13)

There are two cases to consider.

Case 1. Suppose that \( i \notin P_n \). By (13), \( K^i/H^i < (K^i + K^j)/(H^i + H^j) \), so at the next iteration (12) will hold for n.

Case 2. Suppose that \( i \in P_n \). By (13), \( (K_n + K^i)/(H_n + H^i) \leq K^i/H^i < (K^i + K^j)/(H^i + H^j) \). Thus (13) will hold for n at the next iteration of the algorithm. \( \square \)
We denote the values of $T_n$ that solve (P) by $T_n^*$. By substituting (6) and (7) into (1), we see that the solution to (P) is

$$B = \Sigma_i 2(K^iH^i)^{0.5}.$$  \hspace{1cm} (14)

The Lower Bound Theorem states that $B$ is a lower bound on the average cost of any feasible solution to the original lot-sizing problem.

The minimum violators algorithm can be implemented in $O(N \log N)$ time by using an appropriate data structure for \( (K^i/H^i : C^i \in S) \). At each iteration of the algorithm we remove the smallest member of this set, and in many of the iterations we alter the value of one of the elements of the set. Using a heap (or any of several other data structures), these two operations can both be performed in $O(\log N)$ time [AHU1974]. Since the number of iterations is at most $N - 1$, the overall running time of the algorithm is $O(N \log N)$.

Finding a Good Power-of-Two Policy.

The approximate reorder intervals $T_n^*$ computed by the Minimum Violators Algorithm are rounded off to create a power-of-two policy $\bar{T} = (\bar{T}_n, 1 \leq n \leq N)$ as follows. Let $\beta$ be the base planning period, and let

$$\bar{T}_n = 2^{p_n} \beta$$  \hspace{1cm} (15)

where $p_n$ is the integer that satisfies
\[ \sqrt{\frac{5}{2}} \leq \frac{T^*_n}{T_n} < \sqrt{2} \]. \hspace{1cm} (16)

Since (1) holds for \( T^*_n \), it must hold for \( \tilde{T}_n \). Therefore \( \mathcal{J} \) is nested, and its cost is given by (2).

**Theorem 2.** \( c(\mathcal{J}) \leq \frac{1}{2}(\sqrt{5} + \sqrt{2})B < 1.061B \).

**Proof.** By (6) and (1), we can write \( c(\mathcal{J}) \) as

\[ \sum_{i} \left[ \frac{K^i}{T^i} + H^i T^i \right] \]

where \( T^i = \tilde{T}_n \) for all \( n \in C^i \). By (7) and (16), \( \frac{K^i}{T^i} + H^i T^i = (K^i H^i)^{5} \left( \frac{T^i}{\tilde{T}_n} + \frac{\tilde{T}_n}{T^i} \right) \leq (K^i H^i)^{5} (\sqrt{5} + \sqrt{2}) \).

By (14), the cost of \( \mathcal{J} \) is at most \( \frac{1}{2}(\sqrt{5} + \sqrt{2})B \).

\[ \Box \]

If one is willing to use the base planning period \( \beta \) as a variable, it is possible to find a power-of-two policy whose average cost is within 2% of \( B \) in \( O(N \log N) \) time [Ro1983, Ro1984].

**Proof of the Lower Bound Theorem.**

All that remains is to prove that \( B \) is a lower bound on the average cost of an arbitrary policy. The proof follows.

**The Lower Bound Theorem.** The solution to (P), \( B = \sum_{i} 2(K^i H^i)^{5} \), is a lower bound on the average cost of an arbitrary policy over any (finite or infinite) horizon.
Proof. It suffices to prove the result for finite horizons only. Consider an arbitrary feasible policy over a horizon of length \( T \). We will show that the cost incurred by this policy is at least \( BT \).

We denote the echelon inventory for component \( n \) at time \( t \) by \( E_n^t \). Clearly \( E_n^t \geq E_{S_n}^t \). For all \( k \), \( 0 \leq k \leq N \), we define

\[
A(\lambda, t) = \sum_n [h_n - x_n + \sum_{m \in P_n} x_m] E_n^t.
\]

where \( \lambda = (\lambda_n: 1 \leq n \leq N) \). Note that \( \frac{\partial A}{\partial \lambda_n} = E_{S_n}^t - E_n^t \leq 0 \). Also, (4) implies that \( \lambda = (\lambda_n: 1 \leq n \leq N) \geq 0 \).

Therefore \( A(0) \geq A(\lambda) \), i.e.,

\[
\sum_n h_n E_n^t \geq \sum_n H_n E_n^t \text{ where } H_n \equiv h_n - \lambda_n + \sum_{m \in P_n} \lambda_m.
\]

Now, if component \( n \) is in cluster \( i \) then by (4) \( K_n K_i^{H_n} = H_n \). Since \( K_i = \sum_{n \in C_i} K_n \) we have \( \sum_{n \in C_i} (K_n H_n)^5 = (K_i H_i)^5 \). Thus if \( J_n \) is the total number of orders placed for component \( n \), the total cost incurred by the policy is

\[
C = \sum_n \left[ J_n K_n + \int_0^T h_n E_n^t dt \right] \\
\geq \sum_n \left[ J_n K_n + \int_0^T H_n E_n^t dt \right].
\]
Note that $E_n^t \geq 0$, that $E_n^t$ increases at all times $t$ when an order for $n$ is placed, and that $\frac{d}{dt} E_n^t = -2$ at all other times $t$. Therefore $J_n K_n + \int_0^T H_n E_n^t \, dt$ is the cost of a feasible policy for a single-facility lot-sizing problem on a finite horizon of length $T$ with setup cost $K_n$, holding cost rate $H_n$, and demand rate 2. Consequently

$$C \geq \sum_n 2T(K_n H_n)^{-5} = BT.$$  

Summary of Related Models and Guide to the Literature.

The following table summarizes the related models for which the same type of results are available. The models differ in the structure of the bill-of-materials network, in the type of policies allowed, and in the algorithms used to solve (P). They are arranged in the suggested reading order, determined mostly by the complexity of the model and the algorithm used to solve (P).

For all of these models the cost of the heuristic is within 6% of optimal for a fixed base planning period and within 2% if the base planning period is variable. However in the models that are restricted to nested policies, the costs of the heuristics are within the specified percentage of the cost of an optimal nested policy. In the other models the costs of the heuristics are within the specified percentage of the infimum of the costs of all feasible policies. The cost of an optimal nested policy can be arbitrarily large relative to that of an optimal policy. However
it is widely conjectured that nested policies are near-optimal in a majority of the systems one would encounter in practice.

Table 1. Related Models.

<table>
<thead>
<tr>
<th>Structure of the Bill of Materials Network</th>
<th>Restricted to Nested Policies</th>
<th>Running Time</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joint Replenishment</td>
<td>no</td>
<td>$O(N \log N)$</td>
<td>[JMM1985]</td>
</tr>
<tr>
<td>One-Warehouse, Multi-Retailer, Single Item</td>
<td>no</td>
<td>$O(N \log N)$</td>
<td>[Ro1983]</td>
</tr>
<tr>
<td>General Circuitless</td>
<td>yes</td>
<td>$O(N^4)$</td>
<td>[MM1983]</td>
</tr>
<tr>
<td>General Circuitless</td>
<td>no</td>
<td>$O(N^4)$</td>
<td>[Ro, Qu1985]</td>
</tr>
<tr>
<td>Series Digraph, including Single-Item Assembly and Distribution Systems</td>
<td>yes (optimal in assembly systems)</td>
<td>$O(N \log N)$</td>
<td>see [Ro1984]</td>
</tr>
<tr>
<td>One-Warehouse, Multi-Retailer, Many Items</td>
<td>yes</td>
<td>$O(N \log N)$</td>
<td>[MR1985]</td>
</tr>
<tr>
<td>Oriented Trees</td>
<td>yes</td>
<td>$O(N \log N)$</td>
<td>[JR1985]</td>
</tr>
</tbody>
</table>

Historically, the first models for which heuristics were developed are the joint replenishment system, the one-warehouse, multi-retailer, single-item system, and nested policies in general systems. The first models for which bounds on worst-case effectiveness were established were for non-nested policies in the one-warehouse, multi-retailer, single-item system and non-nested policies in the general circuitless system.
References


