

Optimizing Multi-Stage Production with Constant Lot Size and Varying Numbers of Batches

ANDREW Z SZENDROVITS

McMaster University, Canada

ZVI DREZNER

University of Michigan-Dearborn, USA

(Received June 1979; in revised form April 1980)

The model presented in this paper assumes that a uniform lot size is produced through a series of manufacturing stages, with a single set-up and without interruption at each stage. Transportation of partial lots, called batches, is allowed between stages after the whole batch is completed. The batch sizes must be equal at any particular stage, but the optimal number of equal-sized batches may differ across stages. Of course, the set-up costs, the inventory-holding costs and the transportation costs influence both the optimal batch-sizes at the various stages and the uniform lot size. An optimization method for this deterministic model is developed and is illustrated by an example.

INTRODUCTION

PRODUCING PARTS for a complex product often involves a series of operations (stages) being performed on lots of a single item. Specifically, an overwhelming number of such parts are produced in the automobile industry and generally in the machine industry. It is also notable that the number of manufacturing stages is usually rather large (5-15 stages). Although a single part may not be (and usually is not) sold directly, the assembly of the product incorporating this part could be continuous over time. Since part-manufacturing is seldom synchronized with the rhythm of the assembly line, a single part may be manufactured intermittently. When this happens, the net effect is that of a continuous demand fed by a part produced in lots. This is the reason for the applicability of lot-size models.

The single-stage EPQ (Economic Production Quantity) model ignores the work-in-process inventory; consequently it is frequently misused in multi-stage manufacturing. Since it overstates

the optimal lot-size, dramatic savings could be generated by replacing the single-stage EPQ model with a multi-stage model when appropriate. Obviously, scheduling precedence must be given to products whose production must follow the process organization implied by the multi-stage model. When facilities are shared by several intermittently-manufactured product lots, the scheduling of EPQs for the entire spectrum of products is rarely feasible. However, experience has often shown that a relatively small portion of all products constitute a very large part of the work-in-process inventory. Scheduling priorities in manufacturing a small number of products, as the multi-stage model requires, do not present an obstacle in practice. The reason is that these products do not tie-up a prohibitive proportion of the manufacturing capacity and leave ample room to manipulate the manufacturing schedule for the remaining product line. This practical consideration justifies the limited application of EPQ models even in very tightly scheduled MRP (Material Requirement Planning) systems. The

key issue is to select the appropriate products for production according to an EPQ model.

Any multi-stage production/inventory model is intended for some particular organization of the manufacturing process. In our model a 'lot' denotes the quantity produced with one set-up at a stage and a 'batch' denotes the portion of a lot transported to the next stage.

The main characteristics of our model can be summarized as follows: (a) a uniform lot size is produced through all stages in a serial manufacturing system; (b) the lot is produced without interruption with a single set-up at each stage; (c) before the lot is finished, equal-sized batches can be transported from one stage to the next upon completion of a batch; (d) a different (integer) number of equal-sized batches can be produced at the various stages. The availability of batches for shipment to the next stage prior to the completion of the entire lot reduces the average inventory considerably. Since only one set-up is required, the total cost of the system is much smaller than the cost with full-lot shipments. Of course, the magnitude of the set-up costs and the transportation costs of batches will influence the optimal lot-size and the optimal number of batch shipments at various stages.

The model assumes deterministic and constant demand and production rates, fixed set-up costs and linear inventory-holding costs over an infinite time horizon. The transportation cost of batches for a stage is related to the load capacity of transport equipment used at that particular stage. The cost of holding one unit is related to the physical inventory on which stage i was completed. Unrestricted capacity is assumed in the manufacturing process. Set-up and transportation times are considered to be insignificant and hence are ignored. No backlogging (deliberate shortage) is permitted in the system. The lot size and the batch size do not have to be integers and, hence, the assumption is that units of the product are 'infinitely divisible'. This is a frequently-made assumption in such models and is not restrictive if optimal lot sizes and batch sizes are found to be sufficiently large. There are n manufacturing stages, $i = 1, \dots, n$; $i = n$ represents the final stage that converts the process inventory to finished product.

The following notation is used:

D = demand (consumption) rate of the final product (at stage n);

P_i = production rate at stage i (note that $P_i > D$);

F_i = fixed (set-up) cost per lot at stage i ;

T_i = transportation cost of a batch from stage i to the next stage;

c_i = unit inventory holding cost, per unit time, at stage i ;

y_i = the number of equal-sized batches at stage i (note that y_i is an integer);

Q = uniform lot size for the system;

$x_i = Q/y_i$, the size of one batch at stage i .

Note that all parameters and variables are greater than zero.

The model presented in this paper significantly improves the uniform lot-size models of Szendrovits in [1, 2] which require the same number of equal-sized batches at all stages. Relaxing this assumption by allowing different numbers of batches at various stages considerably reduces the work-in-process inventory and the total cost of the production/inventory system.

THE MODEL AND THE OBJECTIVE FUNCTION

Figure 1 is intended to facilitate understanding of the method by which the magnitude of stage i inventory is determined when $P_i < P_{i+1}$. The first (slanted) solid line represents the cumulative production at stage i . The corners of the dotted-line triangles below this indicate the availability of five completed batches for stage $i + 1$. The second (slanted) solid line represents the cumulative production at stage $i + 1$ and the dotted-lines crossing this show the depletion of stage i inventory. Thus, the trapezoid enclosed by solid lines illustrates the time-weighted inventory at stage i .

The inventory at stage i builds up over time period Q/P_i , during which y_i number of x_i -sized batches are transported to stage $i + 1$. To manufacture lot size Q with a single set up, when $P_i < P_{i+1}$, the time delay between the start of stages i and $i + 1$ must be $Q/P_i + x_i/P_{i+1} - Q/P_{i+1}$. Stage i inventory is gradually converted to stage $i + 1$ inventory as y_{i+1} number of x_{i+1} -sized sub-batches are produced during time period Q/P_{i+1} .

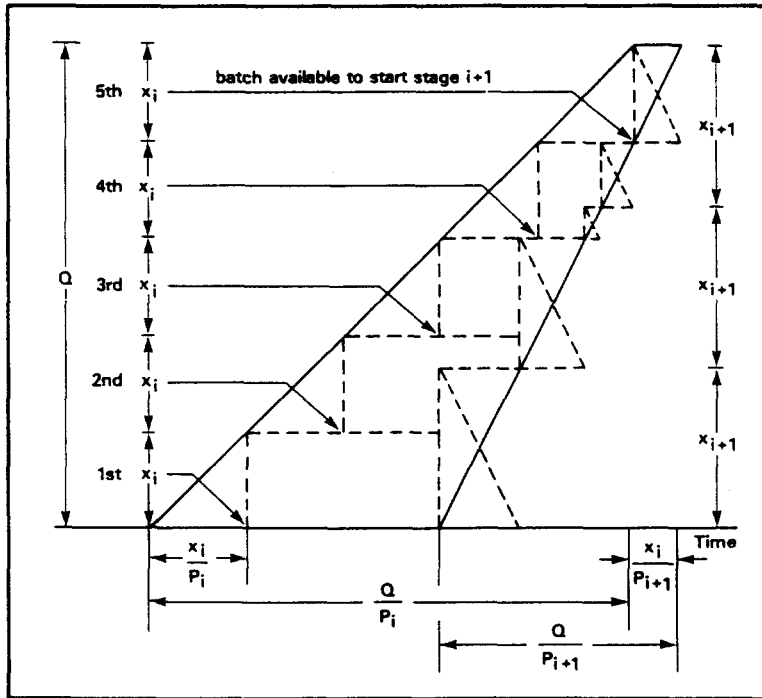


FIG. 1. Time-weighted inventory at stage i when $y_i = 5$, $y_{i+1} = 3$ and $P_i < P_{i+1}$.

The time-weighted inventory at stage i is represented by the area of the trapezoid in Fig. 1; this is

$$\frac{Q}{2} \left[\left(\frac{Q}{P_i} + \frac{x_i}{P_{i+1}} - \frac{Q}{P_{i+1}} \right) + \frac{x_i}{P_{i+1}} \right]$$

$$= Q \frac{x_i}{P_{i+1}} + \frac{Q^2}{2} \left(\frac{1}{P_i} - \frac{1}{P_{i+1}} \right), \text{ if } P_i < P_{i+1},$$

and

$$\frac{Q}{2} \left[\frac{x_i}{P_i} + \left(\frac{Q}{P_{i+1}} + \frac{x_i}{P_i} - \frac{Q}{P_i} \right) \right]$$

$$= Q \frac{x_i}{P_i} + \frac{Q^2}{2} \left(\frac{1}{P_{i+1}} - \frac{1}{P_i} \right), \text{ if } P_i \geq P_{i+1}.$$

Since $x_i = Q/y_i$, the general expression for the time-weighted inventory at stage i is

$$Q^2 \left(\frac{1}{y_i \max \{P_i, P_{i+1}\}} + \frac{1}{2} \left| \frac{1}{P_i} - \frac{1}{P_{i+1}} \right| \right).$$

By multiplying this expression by c_i and D/Q , the number of inventory cycles per unit time, we obtain the average inventory cost for stage i . Then, by summing over $i = 1, 2, \dots, n$ and setting $P_{n+1} = D$, we find that the average inventory cost per unit time for the whole system is

$$D \sum_{i=1}^n c_i \left(\frac{Q}{y_i \max \{P_i, P_{i+1}\}} + \frac{Q}{2} \left| \frac{1}{P_i} - \frac{1}{P_{i+1}} \right| \right).$$

The sum of the fixed (set-up) and transportation costs per unit time of the system is

$$D \sum_{i=1}^n \left(\frac{F_i}{Q} + \frac{y_i T_i}{Q} \right).$$

The total cost of the system per unit time is

$$C(Q, Y) = D \sum_{i=1}^n \left(\frac{F_i}{Q} + \frac{Q c_i}{2} \left| \frac{1}{P_i} - \frac{1}{P_{i+1}} \right| \right)$$

$$+ D \sum_{i=1}^n \left(\frac{T_i}{Q y_i} + \frac{Q}{y_i} \frac{c_i}{\max \{P_i, P_{i+1}\}} \right) \quad (1)$$

where

$$Y = \{y_1, y_2, \dots, y_n\}, y_i = \text{positive integer}$$

and

$$P_{n+1} = D.$$

The objective is to minimize the sum of fixed costs and inventory-holding costs of this production/inventory system.

Deriving the objective function from (1) we obtain:

$$\text{Minimize } C(Q, Y) = A Q + B/Q + \sum_{i=1}^n [(b_i/Q) y_i + a_i Q/y_i] \quad (2)$$

subject to

$$y_i = \text{positive integer for } i = 1, \dots, n;$$

where

$$Y = \{y_1, y_2, \dots, y_n\};$$

$$A = D \sum_{i=1}^n \frac{c_i}{2} \left| \frac{1}{P_i} - \frac{1}{P_{i+1}} \right|;$$

$$B = D \sum_{i=1}^n F_i;$$

$$a_i = \frac{Dc_i}{\max\{P_i, P_{i+1}\}};$$

$$b_i = DT_i;$$

$$P_{n+1} = D;$$

$$A, B, a_i, b_i > 0.$$

THE OPTIMIZATION METHOD

The optimization of the objective function (2) is carried out in two phases.

In the first phase we find a heuristic solution. The relevant equations and the algorithm for this solution are described in Appendix A.

The heuristic solution is attained in a finite number of iterations. In the numerous examples that we solved, convergence was very fast; the algorithm produced the solution in three iterations or fewer. Computational experience with many problems also has shown that the heuristic solution was optimal in the overwhelming majority of cases. Nevertheless, we regard this solution only as the upper bound on the optimal solution.

In the second phase we obtain the optimal solution to the problem. The relevant equations and the algorithm for obtaining this solution are given in Appendix B.

In finding the optimal solution, we use the results of the heuristic procedure as starting values in developing a lower bound and an upper bound within which the optimal cost lies. An efficient scanning process for the optimal cost ensures that the maximum number of iterations in the scanning algorithm is only a minuscule fraction of the number of iterations required to enumerate completely within the bounds.

A computational example for the heuristic and exact algorithms is given in Appendix C.

CONCLUSION

The model presented in this paper relaxes the constraint that batch sizes must be the same

at each stage. This added flexibility naturally results in a lower cost. Unless no batch shipment is warranted at any stage, the cost resulting from this model is invariably lower than the cost generated by a model requiring the same batch sizes at each stage. The greater the variation among the transportation costs for a batch at various stages, the more significant the cost difference between these two inventory policies.

As an area for investigation we suggest the development of a model that combines variable lot-sizes and batch transshipments at various stages. This would permit even more flexible inventory policies and would further reduce costs. However, it is unlikely that an optimization method could be found to solve this considerably more complex problem; instead, a heuristic procedure is foreseen.

ACKNOWLEDGMENT

This research was supported by a grant from the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

1. SZENDROVITS AZ (1975) Manufacturing cycle time determination for a multi-stage economic production quantity model. *Mgmt Sci.* 22(3), 298–308.
2. SZENDROVITS AZ (1976) On the optimality of sub-batch sizes for a multi-stage EPQ model—a rejoinder. *Mgmt Sci.* 23(3), 334–338.

APPENDIX A

A heuristic solution

Given $Y = \{y_1, y_2, \dots, y_n\}$ the cost equation in (2) can be written as follows:

$$C(Q, Y) = \left(A + \sum_{i=1}^n a_i y_i \right) Q + \left(B + \sum_{i=1}^n b_i y_i \right) / Q \quad (\text{A.1})$$

in which the part that expresses the influence of y_i s on the cost, for $i = 1, 2, \dots, n$, is

$$E(y_i) = (b_i/Q)y_i + a_i Q/y_i. \quad (\text{A.2})$$

If we momentarily disregard the integrality constraint on y_i s, a so-called continuous solution can be found for Q and for the vector $Y = \{y_1, y_2, \dots, y_n\}$ by applying Lemma 1.

Lemma 1

For $u, v > 0$ the minimum of $f(Z) = uZ + v/Z$ is

$$Z^* = (v/u)^{1/2}.$$

Also,

$$f(Z^*) = 2(uv)^{1/2}.$$

Proof:

The function of $f(Z)$ is convex for $Z \geq 0$. By solving the extremal equation $df(Z)/dZ = 0$ we obtain Z^* . Substituting Z^* , we obtain $f(Z^*)$.

Using Lemma 1, the continuous solutions (denoted by superscript c) to (A.1) and (A.2) can be expressed as follows:

$$Q^c = (B/A)^{1/2}, \tag{A.3}$$

$$y_i^c = Q^c(a_i/b_i)^{1/2}, \tag{A.4}$$

$$C^c = C(Q^c, Y^c) = A Q^c + B/Q^c + 2 \sum_{i=1}^n (a_i b_i)^{1/2}. \tag{A.5}$$

It is worthwhile to note that the 'continuous optimal cost' in (A.5) is a lower bound on the optimal costs obtainable for the objective function (2) at integer y_i values.

Since the objective function (2) must be optimized subject to integer y_i s we must now determine the vector $Y = \{y_1, y_2, \dots, y_n\}$ according to Lemma 2.

Lemma 2.

For $u, v > 0$ the positive integer \hat{Z} that minimizes $f(Z) = uZ + v/Z$ is

$$\hat{Z} = (e)^{\uparrow} = \lceil (v/u + 0.25)^{1/2} \rceil,$$

where $(e)^{\uparrow}$ is the closest integer to e .

Proof:

The optimal \hat{Z} must satisfy

$$f(\hat{Z} - 1) \geq f(\hat{Z}) \leq f(\hat{Z} + 1).$$

The right hand side of the inequality yields

$$u\hat{Z} + v/\hat{Z} \leq u(\hat{Z} + 1) + v/(\hat{Z} + 1), \\ \therefore \hat{Z}(\hat{Z} + 1) \geq v/u, \therefore (\hat{Z} + 0.5)^2 \geq (v/u) + 0.25.$$

The left hand side of the inequality yields

$$u\hat{Z} + v/\hat{Z} \leq u(\hat{Z} - 1) + v/(\hat{Z} - 1), \\ \therefore \hat{Z}(\hat{Z} - 1) \leq v/u, \therefore (\hat{Z} - 0.5)^2 \leq (v/u) + 0.25.$$

Solving these two quadratic inequalities we obtain:

$$\hat{Z} - 0.5 \leq (v/u + 0.25)^{1/2} \leq \hat{Z} + 0.5$$

This proves that \hat{Z} is the closest integer to $(v/u + 0.25)^{1/2}$. If this value were to fall exactly between two integers, then rounding in either direction gives the same cost.

Now, we can apply Lemma 2 to equation (A.2) to obtain a general expression for calculating integer y_i values:

$$y_i(Q) = \lceil [(Q_2 a_i/b_i) + 0.25]^{1/2} \rceil. \tag{A.6}$$

Note that $y_i(Q) = Q$ whenever $a_i/b_i \geq 1$ and hence $x_i = 1$. However, this only happens when $T_i(\max\{P_i, P_{i+1}\}) \leq c_i$.

By using equation (A.6) we can establish the vector, $Y = Y(Q)$, which gives the lowest cost for a given Q . In fact, there is usually a range of Q values for which the same Y vector yields the lowest cost. On the other hand, there is only one \hat{Q} value which generates the lowest possible cost for a given Y vector. This \hat{Q} value can be determined by applying Lemma 1 to equation (A.1). Hence,

$$\hat{Q}(Y) = \left[\left(B + \sum_{i=1}^n b_i y_i \right) / \left(A + \sum_{i=1}^n \frac{a_i}{y_i} \right) \right]^{1/2}. \tag{A.7}$$

The corresponding optimal cost is

$$\hat{C}(Y) = 2 \left[\left(B + \sum_{i=1}^n b_i y_i \right) \left(A + \sum_{i=1}^n \frac{a_i}{y_i} \right) \right]^{1/2}. \tag{A.8}$$

One must realize, however, that another Y vector may exist which improves the cost at \hat{Q} . Hence, we calculate a new Y vector at \hat{Q} . If this is the same as the previous one, then \hat{Q} has stabilized and we have found a possible optimal heuristic solution which we designate by Q^m, Y^m and C^m . Otherwise, we must continue the iterations until \hat{Q} stabilizes. This procedure is called Algorithm 1.

Algorithm 1:

- Step 1. Calculate Q^c in (A.3). Set iteration number $k = 1$. $Q_k = Q^c$.
- Step 2. Find vector $Y_k = Y(Q_k)$ according to (A.6).
- Step 3. Calculate $\hat{Q}_k = \hat{Q}(Y_k)$ according to (A.7).
- Step 4. If $\hat{Q}_k \neq Q_k$ set $Q_{k+1} = \hat{Q}_k$, $k = k + 1$ and go to Step 2.
- Step 5. Set $Q^m = \hat{Q}_k$, $Y^m = Y_k$. Calculate $C^m = \hat{C}(Y_k)$ from (A.8). Stop.

The equations used to calculate Y_k and \hat{Q}_k in our algorithm ensure that the value of the objective function decreases at each iteration. That is, the Y vector would only change at a certain Q if it improves the cost. This process continues until \hat{Q} stabilizes. Thus, the heuristic solution is obtained in a finite number of iterations.

APPENDIX B

The optimum solution

The heuristic solution in Appendix A gives us an upper bound on the cost of the optimal integer solution. Also, we obtain an absolute lower bound by calculating the continuous optimal cost for Q^c in (A.5). Similarly, the best 'continuous' cost, which is a lower bound at a given Q , can be determined as

$$C^0 = A Q + B/Q + 2 \sum_{i=1}^n (a_i b_i)^{1/2}. \tag{B.1}$$

Using (B.1) we can establish a lower bound (Q_L) and an upper bound (Q_U) within which the optimal lot size (Q^*) lies, i.e. $Q_L \leq Q^* \leq Q_U$. This is achieved by finding the Q values that yield a continuous cost equal to the lowest cost obtained so far. We denote this cost C^{min} . It follows from (B.1) that when $C^0 = C^{min}$ we obtain

$$Q^2 - 2\alpha Q + \beta = 0 \tag{B.2}$$

where

$$\alpha = \left[C^{min} - 2 \sum_{i=1}^n (a_i b_i)^{1/2} \right] / 2A;$$

$$\beta = B/A.$$

The solution of the quadratic equation in (B.2) gives

$$Q_L = \alpha - (\alpha^2 - \beta)^{1/2}, \tag{B.3}$$

$$Q_U = \alpha + (\alpha^2 - \beta)^{1/2}. \tag{B.4}$$

To calculate Q_L and Q_U we use the cost of the heuristic solution, $C^{min} = C^m$. The next task is to search for Q^* within the bounds Q_L and Q_U . We choose Q_L as the starting point in searching for Q^* . It is helpful to recall that a Y vector is optimal for a range of Q values. If one member of this vector, some y_i , has increased by one (while all others have remained the same), a new range of Q values will be associated with the new Y vector.

We denote the starting point of this new range by \bar{Q}_i . We now establish a principle that enables us to design a scanning process which eliminates a large number of Q values from the enumeration.

Let us examine the effect of the change in value of some y_i on the objective function. Since the part of the objective function that y_i influences is given in (A.2) we can use this to find the condition under which $E(y_i + 1) \leq E(y_i)$. Using (A.2) we can state that

$$\frac{b_i(y_i + 1)}{Q} + \frac{a_i Q}{(y_i + 1)} \leq \frac{b_i y_i}{Q} + \frac{a_i Q}{y_i}$$

and solving this inequality we can conclude that this is true if and only if

$$Q \geq [(b_i/a_i)y_i(y_i + 1)]^{1/2} \tag{B.5}$$

The starting points of the ranges associated with changing y_i to $y_i + 1$ for each i separately are

$$\bar{Q}_i = [(b_i/a_i)y_i(y_i + 1)]^{1/2} \tag{B.6}$$

We can start the scanning process by setting $C^{\min} = C^m$ and finding the optimal Y vector for Q_L . Then, we define the best \bar{Q} for this Y vector and designate the corresponding cost \bar{C} . Selecting the lower of the costs \bar{C} and C^m as the new C^{\min} and denoting the pertinent values Q and Y with Q^{\min} and Y^{\min} , we calculate the \bar{Q}_i values according to (B.6). If the smallest of the \bar{Q}_i values is larger than Q_U , i.e. if $\min\{\bar{Q}_i\} \geq Q_U$, this means that the range of Q values associated with the new Y vector exceeds Q_U (at which the upper bound on costs occurred). Consequently, we have found the optimal values $C^* = C^{\min}$, $Q^* = Q^{\min}$ and $Y^* = Y^{\min}$. If $\min\{\bar{Q}_i\} < Q_U$, we update the Y vector by changing only one y_i to $(y_i + 1)$, namely the one which has the same subscript as $\min\{\bar{Q}_i\}$. We then iterate until $\min\{\bar{Q}_i\} \geq Q_U$. Algorithm 2 specifies the scanning process.

Algorithm 2:

- Step 1. Set iteration number $k = 1$, $Q^{\min} = Q^m$, $Y^{\min} = Y^m$ and $C^{\min} = C^m$.
- Step 2. Calculate Q_L from (B.3) and Q_U from (B.4).
- Step 3. Find $Y_k = Y(Q_L)$ from (A.6).
- Step 4. Calculate $\bar{Q}_k = \bar{Q}(Y_k)$ from (A.7) and $\bar{C}_k = \bar{C}(Y_k)$ from (A.8).
- Step 5. If $\bar{C}_k \geq C^{\min}$, go to Step 7.
- Step 6. Update $C^{\min} = \bar{C}_k$, $Q^{\min} = \bar{Q}_k$, $Y^{\min} = Y_k$.
- Step 7. Find $\{\bar{Q}_1, \bar{Q}_2, \dots, \bar{Q}_n\}$ with Y_k from (B.6).
- Step 8. Select $\bar{Q}_j = \min\{\bar{Q}_1, \bar{Q}_2, \dots, \bar{Q}_n\}$ and note subscript j .
- Step 9. If $\bar{Q}_j < Q_U$, set $k = k + 1$. Define $Y_k = \{y_1, y_2, \dots, y_n\}$.

where

$$y'_m = \begin{cases} y_m & | m \neq j \\ (y_j + 1) & | m = j \end{cases} \text{ for } m = 1, 2, \dots, n.$$

Go to Step 4.

- Step 10. Set $C^* = C^{\min}$, $Q^* = Q^{\min}$, $Y^* = Y^{\min}$. Stop.

The efficiency of this algorithm is obvious if we compare the number of iterations in our scanning process with that necessary for complete enumeration. For this, let the feasible values for y_i be $\{y_i^{\min}, \dots, y_i^{\max}\}$ and let K_i be the number of such feasible values. For n stages, the number of iterations required for a complete enumeration is

$$I_e = K_1 \cdot K_2 \cdot \dots \cdot K_n \tag{B.7}$$

In our scanning process only one of the y_i s is increased in every iteration and since every y_i could increase $K_i - 1$

times, the maximum number of iterations is

$$I_s = \left(\sum_{i=1}^n K_i \right) - n + 1 \tag{B.8}$$

For instance, when $n = 5$ and $K_i = 10$ for $i = 1, 2, \dots, n$ we find from (B.7) that complete enumeration would involve $I_e = 10^5 = 100,000$ iterations. In contrast we find, from (B.8), that our scanning process would need a maximum of $I_s = (5)(10) - 5 + 1 = 46$ iterations. It is noteworthy, however, that in several examples solved with the algorithm most of the results were obtained in three or fewer iterations.

APPENDIX C

A computational example

The following problem parameters are used to illustrate the computations: $A = 0.1$, $B = 43,300$, $a_1 = 0.1$, $a_2 = 0.1$, $a_3 = 0.01$, $b_1 = 1,000$, $b_2 = 9,000$, $b_3 = 5,000$. The computations using Algorithm 1 described in Appendix A and Algorithm 2 in Appendix B are shown below.

ALGORITHM 1

Iteration 1

- Step 1: Calculate Q^c in (A.3).
 $Q^c = (43,300/0.1)^{1/2} = 658.0274$.
 Set $k = 1$ and $Q_1 = Q^c$; $Q_1 = 658.0274$.
- Step 2: Find vector $Y_1 = Y(Q_1)$ according to (A.6);
 $y_1(Q_1) = [433,000(0.1)/1,000 + 0.25]^{1/2} = (6.600)_1 = 7$.
 $y_2(Q_1) = [433,000(0.1)/9,000 + 0.25]^{1/2} = (2.250)_1 = 2$.
 $y_3(Q_1) = [433,000(0.01)/5,000 + 0.25]^{1/2} = (1.056)_1 = 1$.
 Thus, $Y_1 = \{7, 2, 1\}$.
- Step 3: Calculate $\bar{Q}_1 = \bar{Q}(Y_1)$ according to (A.7);
 $\sum_{i=1}^3 b_i y_i = 1,000(7) + 9,000(2) + 5,000(1) = 30,000$,
 $\sum_{i=1}^3 a_i/y_i = 0.1/7 + 0.1/2 + 0.01/1 = 0.0743$
 and
 $\bar{Q}(Y_1) = [(43,300 + 30,000)/(0.1 + 0.0743)]^{1/2} = 648.5166$.
 Thus, $\bar{Q}_1 = 648.5166$.
- Step 4: Test $\bar{Q}_1 \neq Q_1$;
 $\bar{Q}_1 = 648.5166$ and $Q_1 = 658.0274$,
 hence $\bar{Q}_1 \neq Q_1$ and $Q_2 = \bar{Q}_1$. Thus, $Q_2 = 648.5166$.
 Set $k = k + 1$; $k = 2$. Go to Step 2.

Iteration 2

- Step 2: Find vector $Y_2 = Y(Q_2)$ according to (A.6);
 $y_1(Q_2) = [420,573.78(0.1)/1,000 + 0.25]^{1/2} = (6.504)_1 = 7$,
 $y_2(Q_2) = [420,573.78(0.1)/9,000 + 0.25]^{1/2} = (2.219)_1 = 2$,
 $y_3(Q_2) = [420,573.78(0.01)/5,000 + 0.25]^{1/2} = (1.045)_1 = 1$.
 Thus, $Y_2 = \{7, 2, 1\}$.
- Step 3: Calculate $\bar{Q}_2 = \bar{Q}(Y_2)$ according to (A.7);
 $\sum_{i=1}^3 b_i y_i = 1,000(7) + 9,000(2) + 5,000(1) = 30,000$,
 $\sum_{i=1}^3 a_i/y_i = 0.1/7 + 0.1/2 + 0.01/1 = 0.0743$

and

$$\hat{Q}(Y_2) = [(43,300 + 30,000)/(0.1 + 0.0743)]^{1/2} = 648.5166.$$

Thus, $\hat{Q}_2 = 648.5166$.

Step 4: Test $\hat{Q}_2 \neq Q_2$;

$$\hat{Q}_2 = 648.5166, Q_2 = 648.5166, \text{ hence } \hat{Q}_2 = Q_2.$$

Thus, continue.

Step 5: Set $Q^m = \hat{Q}_2$, $Y^m = Y_2$;

$$Q^m = 648.5166 \text{ and } Y^m = \{7, 2, 1\}.$$

Calculate $C^m = \hat{C}(Y_2)$ from (A.8);

with Y_2 in Step 3 we have $\sum_{i=1}^3 b_i y_i = 30,000$

$$\sum_{i=1}^3 (a_i/y_i) = 0.0743, \text{ hence } \hat{C}(Y_2) = [(43,300 + 30,000)(0.1 + 0.0743)]^{1/2} = 226.0544.$$

Thus, $C^m = 226.0544$. Stop.

Note that the heuristic solution: C^m , Q^m , Y^m may be optimal but only Algorithm 2 would verify whether it is or not.

ALGORITHM 2

Iteration 1

Step 1: Set iteration number $k = 1$, $Q^{\min} = Q^m$, $Y^{\min} = Y^m$ and $C^{\min} = C^m$; $Q^{\min} = 648.5166$, $Y^{\min} = \{7, 2, 1\}$ and $C^{\min} = 226.0544$.

Step 2: Calculate Q_L from (B.3);

$$\sum_{i=1}^3 (a_i b_i)^{1/2} = [0.1(1,000)]^{1/2} + [0.1(9,000)]^{1/2} + 0.01(5,000)^{1/2} = 47.0711,$$

$$\alpha = [226.0544 - 2(47.0711)]/[2(0.1)] = 659.5611,$$

$$\beta = 43,300/0.1 = 433,000 \text{ and}$$

$$Q_L = 659.5611 - (659.5611^2 - 433,000)^{1/2} = 614.6080.$$

Calculate Q_U from (B.4);

$$Q_U = 659.5611 + (659.5611^2 - 433,000)^{1/2} = 704.5141.$$

Thus, $Q_L = 614.6080$ and $Q_U = 704.5141$.

Step 3: Find $Y_1 = Y(Q_L)$ from (A.6);

$$y_1(Q_L) = [377,743(0.1)/1,000 + 0.25]^{1/2} = (6.166)_1 = 6,$$

$$y_2(Q_L) = [377,743(0.1)/9,000 + 0.25]^{1/2} = (2.109)_1 = 2,$$

$$y_3(Q_L) = [377,743(0.1)/5,000 + 0.25]^{1/2} = (1.003)_1 = 1.$$

Thus, $Y_1 = \{6, 2, 1\}$.

Step 4: Calculate $\hat{Q}_1 = \hat{Q}(Y_1)$ from (A.7);

$$\sum_{i=1}^3 b_i y_i = 1,000(6) + 9,000(2) + 5,000(1) = 29,000,$$

$$\sum_{i=1}^3 a_i/y_i = 0.1/6 + 0.1/2 + 0.01/1 = 0.0767 \text{ and}$$

$$\hat{Q}(Y_1) = [(43,300 + 29,000)/(0.1 + 0.0767)]^{1/2} = 639.7228.$$

Thus, $\hat{Q}_1 = 639.7228$. Calculate $\hat{C}_1 = \hat{C}(Y_1)$ from (A.8);

$$\hat{C}(Y_1) = 2[(43,300 + 29,000)(0.1 + 0.0767)]^{1/2} = 226.0354.$$

Thus, $\hat{C}_1 = 226.0354$.

Step 5: Test $\hat{C}_1 \geq C^{\min}$;

$$\hat{C}_1 = 226.0354, C^{\min} = 226.0544, \text{ hence } \hat{C}_1 < C^{\min}.$$

Thus, continue.

Step 6: Update $C^{\min} = \hat{C}_1$, $Q^{\min} = \hat{Q}_1$ and $Y^{\min} = Y_1$;

$$C^{\min} = 226.0354, Q^{\min} = 639.7228 \text{ and } Y^{\min} = \{6, 2, 1\}.$$

Step 7: Find $\{\bar{Q}_1, \bar{Q}_2, \bar{Q}_3\}$ with Y_1 from (B.6);

$$\bar{Q}_1 = [(1,000/0.1)(6)(6 + 1)]^{1/2} = 648.0741,$$

$$\bar{Q}_2 = [(9,000/0.1)(2)(2 + 1)]^{1/2} = 734.8469,$$

$$\bar{Q}_3 = [(5,000/0.01)(1)(1 + 1)]^{1/2} = 1,000.0000.$$

Thus, $\bar{Q}_1 = 648.0741$, $\bar{Q}_2 = 734.8469$, $\bar{Q}_3 = 1,000.0000$.

Step 8: Select $\bar{Q}_j = \min\{\bar{Q}_1, \bar{Q}_2, \bar{Q}_3\}$ and note subscript j ;

$$\bar{Q}_j = \bar{Q}_1 = 648.0741 \text{ and subscript } j = 1.$$

Step 9: Test $\bar{Q}_j < Q_U$ with $j = 1$;

$$\bar{Q}_1 = 648.0741, Q_U = 704.5141, \text{ hence}$$

$$\bar{Q}_j < Q_U \text{ and } k = k + 1, k = 2.$$

Define $Y_2 = \{y'_1, y'_2, y'_3\}$; since $j = 1$,

$$y'_1 = y_1 + 1 = 6 + 1 = 7; y'_2 = y_2$$

$$\text{and } y'_3 = y_3 (y_2 \text{ and } y_3 \text{ are unchanged from Step 3}).$$

Thus, $Y_2 = \{7, 2, 1\}$. Go to Step 4.

Iteration 2

Step 4: Calculate $\hat{Q}_2 = \hat{Q}(Y_2)$ from (A.7);

$$\sum_{i=1}^3 b_i y_i = 1,000(7) + 9,000(2) + 5,000(1) = 30,000,$$

$$\sum_{i=1}^3 a_i/y_i = 0.1/7 + 0.1/2 + 0.01/1 = 0.0743 \text{ and}$$

$$\hat{Q}(Y_2) = [(43,300 + 30,000)/(0.1 + 0.0743)]^{1/2} = 648.5166.$$

Thus, $\hat{Q}_2 = 648.5166$. Calculate $\hat{C}_2 = \hat{C}(Y_2)$ from (A.8);

$$\hat{C}(Y_2) = 2[(43,300 + 30,000)(0.1 + 0.0743)]^{1/2} = 226.0544.$$

Thus, $\hat{C}_2 = 226.0544$.

Step 5: Test $\hat{C}_2 \geq C^{\min}$;

$$\hat{C}_2 = 226.0544, C^{\min} = 226.0354,$$

hence $\hat{C}_2 > C^{\min}$. Thus, go to Step 7.

Step 7: Find $\{\bar{Q}_1, \bar{Q}_2, \bar{Q}_3\}$ with Y_2 from (B.6);

$$\bar{Q}_1 = [(1,000/0.1)(7)(7 + 1)]^{1/2} = 748.3315.$$

(\bar{Q}_2 and \bar{Q}_3 calculated in Step 7 of Iteration 1 remain unchanged as only $\bar{Q}_1 = \bar{Q}_1$ was to change.)

Thus, $\bar{Q}_1 = 748.3315$, $\bar{Q}_2 = 734.8469$, $\bar{Q}_3 = 1,000.0000$.

Step 8: Select $\bar{Q}_j = \min\{\bar{Q}_1, \bar{Q}_2, \bar{Q}_3\}$ and note subscript j ;

$$\bar{Q}_j = \bar{Q}_2 = 734.8469 \text{ and subscript } j = 2.$$

Step 9: Test $\bar{Q}_j < Q_U$ with $j = 2$;

$$\bar{Q}_2 = 734.8469; Q_U = 704.5141, \text{ hence}$$

$$\bar{Q}_2 > Q_U. \text{ Thus, continue.}$$

Step 10: Set $C^* = C^{\min}$, $Q^* = Q^{\min}$, $Y^* = Y^{\min}$;

$$C^* = 226.0354, Q^* = 639.7228, Y^* = \{6, 2, 1\}. \text{ Stop.}$$

Algorithm 2 shows that the heuristic solution obtained in Algorithm 1 was not optimal for this example. The difference between the optimal cost (C^*) and the cost of the heuristic solution (C^m) is rather small due to a very flat cost (envelope) curve.

ADDRESS FOR CORRESPONDENCE: Professor Andrew Z Szendrovits, Production and Management Science Area, Faculty of Business, McMaster University, Hamilton, Ontario, Canada.