COMPUTATION OF CONSTRAINED OPTIMUM QUANTITIES AND REORDER POINTS FOR TIME-WEIGHTED BACKORDER PENALTIES

John P. Matthews

University of Wisconsin—Madison
Madison, Wisconsin

ABSTRACT

The purpose of this paper and the accompanying tables is to facilitate the calculation of constrained optimum order quantities and reorder points for an inventory control system where the criterion of optimality is the minimization of expected inventory holding, ordering, and time-weighted backorder costs. The tables provided in the paper allow the identification of the optimal solution when order quantities and/or reorder points are restricted to a set of values which do not include the unconstrained optimal solution.

1. INTRODUCTION

There are many situations in which the analyst is forced to choose values of order points and order quantities from a finite set of alternatives. If stock can be ordered in multiples of say, m units, it would be under the most rare circumstances that an unconstrained optimal solution (UOS) would be equal to one of the admissible lot sizes. Similarly, it may not be possible to measure the inventory level at any other than, say, n different levels. For example, it may not be economical to measure accurately the exact level of content in pressurized gas containers. Thus, for practical purposes, the reorder point will be an integer number of containers. The tables in this paper allow the identification of the optimal solution when order point and order quantities are constrained.

The continuous review inventory system envisioned here is the same as the one examined by Holt, Modigliani, Muth, and Simon (HMMS) (Ref [4], p. 226). The assumptions concerning the inventory system follow.

- The lead time is shorter than the time between orders, the so-called "lot time."
- The order point or trigger level, T, is nonnegative.
- The lead time is constant and known.
- There is no serial correlation of sales rates between periods.
- An order not satisfied immediately from inventory is backlogged.
- The backorder penalty is a function of the time duration and amount of backorders.

The above assumptions allow the development of a total cost function which HMMS identify as Model Two (Ref. [3], Eq. 12–23). The cost function is equivalent to

$$K(Q,T)=C_F\frac{\overline{S}}{Q}+C_I\left(\frac{Q}{2}+T-\overline{S}_L\right)+\frac{\overline{S}_L}{Q}(C_I+C_D)\int_T^\infty\frac{(S_L-T)^2}{2S_L}f(S_L)\,dS_L, \tag{1}$$

where C_F is the ordering cost,
C_I is the holding cost/unit/year,
C_D is the backorder cost/unit/year,
$\overline{S}$ is the expected annual sales,
$\overline{S}_L$ is the expected sales over the lead time,
$f(S_L)$ is the probability density of S_L,
Q is the order quantity, and
T is the trigger level or reorder point.
Unfortunately, the values of Q and T which minimize $K(Q,T)$ are not easy to identify since, as *HMMS* express it, "the integral above is difficult to evaluate for many density functions of interest, . . ." in reference to the integral in (1).

When demand over lead time is normally distributed we may, with the aid of the accompanying table, evaluate the integral in (1) and employ a procedure presented in the paper to search for the optimal values (constrained or otherwise) of Q and T. Furthermore, the tables allow the $K(Q,T)$ cost surface to be easily generated.

2. COMMENTS ON RELATED PAPERS

The mathematical approach employed by Galliher, Morse, and Simond (GMS) [3] to obtain a total cost function is different from the one employed by HMMS. Their method of arriving at optimal values of Q and T is based upon an approximation to the cost function they derive. Deemer and Hoekstra [2] have developed tables which identify the optimal values of Q and T for the *GMS* model, but the tables do not facilitate cost evaluation or aid in the search for constrained optimal Q, T strategies. Koenigsberg [6] uses a model similar to that of GMS, but adopts a method which he states to be equivalent to minimizing holding and ordering costs subject to a fixed protection against shortage. Backorders are not time-weighted. Thatcher [8] uses a model in which stockholding costs are proportioned to the maximum amount of stock, which seems a doubtful approximation for many applications. Buckland [1] uses a nomogram to simplify the joint calculation of Q and I_t. Unfortunately, the construction of the nomogram is left to the reader. Backorders were not time-weighted in Buckland's treatment. Lampkin and Flowerdew [7] present an iterative procedure for the optimization of a related cost function, but require the generation of a table of values to be used in the optimization procedure.

Herron [4] generated a series of graphs suitable for identifying the UOS for the GMS model. The graphs in the Herron paper may be employed to assess the sensitivity of the minimum cost solution to changes in demand uncertainty over lead time. However, they cannot be used in the identification of a constrained optimal solution (COS), nor any sensitivity in cost to changes in the Q,T strategy from the UOS.

The paper proceeds in several sections. Section 3 displays the derivatives of $K(Q, T)$ and Section 4 describes the cost surface $K(Q, T)$. Section 5 describes properties of the isocost rings and the UOS. Section 6 discusses types of constraints that may be incurred and procedures to identify the COS. Section 7 illustrates the use of the tables and presents several examples. The tables and optimization procedure presented here allow the constrained optimal solution and unconstrained optimal solution to be identified in several minutes of computations manually, or in seconds by computer. The tables allow the calculation of Q and T to within one-tenth of one standard deviation of demand over lead time.

3. PARTIAL DERIVATIVES OF K(Q, T)

Denote as D_{QQ} and D_{TT} the second partial derivatives of $K(Q, T)$ with respect to Q and T, respectively. From (1), then, we have

$$D_{QQ}=\frac{2C_F\bar{S}}{Q^3}+\frac{2S_L}{Q^3}(C_I+C_D)\int_T^\infty \frac{(S_L-T)^2}{2S_L}f(S_L)dS_L, \tag{2}$$

$$D_{TT}=\frac{2\bar{S}_L}{Q}(C_I+C_D)\int_T^\infty \frac{1}{2S_L}f(S_L)dS_L. \tag{3}$$

Note that D_{QQ} and D_{TT} are greater than zero for all values of $\bar{S}_L$, C_F, C_I, C_D, and Q greater than zero.

4. THE TOTAL COST SURFACE

The isocost (IC) rings of the cost function $K(Q, T)$ over the Q, T quadrant form nested rings whose size diminish as $K(Q, T)$ decreases. The rings are symmetric about their major axis for reasons which will be discussed later in this paper. The major axis of the isocost ring is negative in slope, and the major axis of an IC ring is closer to the Q, T origin the lower the total cost value associated with the ring. Figure 1 shows an example where K_j represents total cost associated with ring j.

Since D_{QQ} and D_{TT} are both strictly positive, any point within an IC ring must yield a lower total cost value than any point *on* the IC ring thus ensuring that the IC rings are nested. As the reader may expect, the cost surface becomes relatively flat near the optimal solution. Since the IC rings are nested but not concentric, the rate of increase in $K(Q, T)$ with respect to movement

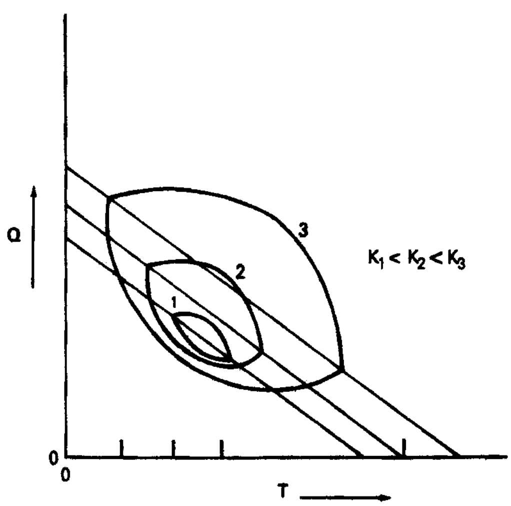

FIGURE 1. IC ring.

from the UOS is very sensitive to the direction of movement. Thus, when it is not possible to implement the UOS, great care must be exercised in choosing among alternative constrained solutions.

5. PROPERTIES OF THE UNCONSTRAINED OPTIMAL SOLUTION

It will be shown that the UOS, under conditions discussed later in the paper, will lie at the point of tangency of one line of a family of parallel lines and a curve convex to the Q, T origin.

Denote as $Q^*(T)$ that value of Q which, for a given value of T, minimizes $K(Q, T)$. An expression for $Q^*(T)$ may be found by setting the first derivative of (1) with respect to Q equal to zero and solving for Q. Doing so yields

$$Q^*(T)=[C_1+C_2EBP(T)]^{1/2}, \tag{4}$$

where $C_1=2C_F\bar{S}/C_I$, $C_2=2\bar{S}_L(C_I+C_D)/C_I$, and $EBP(T)$ represents the integral in (1). When T is large, $EBP(T)$ is small and $Q^*(T)$ approaches the familiar Wilson lot-size formula. From (1), it follows easily that $EBP(T)$ and therefore $Q^*(T)$ are monotonically decreasing functions of T.

Denote the UOS to (1) as Q^{**}, T^{**} and the optimal cost as K^*. Since $Q^*(T)$ is a single-valued function of T, it follows that Q^{**} must lie on the curve defined in (4), i.e., $Q^*(T^{**})=Q^{**}$. It will be assumed in the discussion to follow that $Q^*(T)$ is convex, although the assumption is not supported by a proof. $Q^*(T)$ has always been found to be convex for all values assigned to C_1 and C_2 by the author. To ensure that $Q^*(T)$ is convex for a particular problem, the curve may be traced out for selected values of T and $EBP(T)$ with the aid of the tables. Under the assumption of convexity of $Q^*(T)$, the optimal solution will be shown to be unique.

Consider the isocost rings over the Q, T plane. Setting $K(Q, T)$ equal to some constant, say $\bar{K}$, and solving (1) for Q, we obtain

$$Q(\bar{K}, T)=\frac{\bar{K}}{C_I}+\bar{S}_L-T\pm\left[\left(T-\bar{S}_L-\frac{\bar{K}}{C_I}\right)^2-C_1-C_2EBP(T)\right]^{1/2} \tag{5}$$

Thus, for a given value of $\bar{K}$ and T, (5) yields the value(s) of Q for a given value of T on the IC ring associated with total cost $\bar{K}$. Substituting from (4) yields

$$Q(\bar{K}, T)=L(\bar{K}, T)\pm[(-L(K, T))^2-Q^*(T)^2]^{1/2}, \tag{6}$$

where

$$L(\bar{K}, T)=\frac{\bar{K}}{C_I}+S_L-T. \tag{7}$$

$L(\bar{K}, T)$ is a family of lines whose intercept is

$$\frac{\bar{K}}{C_I}+\bar{S}_L$$

and slope is -1. As $\bar{K}$ decreases, the intercept grows smaller and the lines move toward the origin.*

*Note that the diameter of the isocost ring in the K (Q, T), T plane for a given value of T is related to the amount by which $(-L(K, T))^2$ exceeds $Q^*(T)^2$. As one goes to lower values of $K(Q, T)$ holding T constant, $-L(K, T)$ approaches $Q^*(T)$, thus ensuring that the diameter is diminishing. Since D_{QQ} is positive for all Q and T, it follows that the IC rings are nested, since all points within (outside) the ring yield cost values less (more) than all points on the ring.

In the upper frame of Figure 2, three members of the family of lines $Q=L(K, T)$ are drawn, with $L(K^*, T)$ denoting the line associated with the minimum value of $K(Q, T)$. For values of T such that $L(\bar{K}, T)$ exceeds $Q^*(T)$, it follows from (6), as shown in Frame a of Figure 2, that the isocost ring has two values of $Q(\bar{K}, T)$. Thus it follows from (6) that for values of $\bar{K}$ and T such that

(8) $L(\bar{K}, T)>Q^*(T)$, $Q(\bar{K}, T)$ has two real solutions,

(9) $L(\bar{K}, T)=Q^*(T)$, $Q(\bar{K}, T)$ has one real solution,

(10) $L(\bar{K}, T)<Q^*(T)$, $Q(\bar{K}, T)$ has no real solution.

If, for a given T, say T_k, $Q(\bar{K}, T_k)$ has two real solutions, say Q_1 and Q_2, then by definition of the IC ring,

(11) $K(Q_1, T_k)=K(Q_2, T_k)=\bar{K}$.

But since $D_{QQ}>0$, it follows that there must be a λ, $0<\lambda<1$, such that

(12) $K(Q_\lambda, T_k)<\bar{K}$

where $Q_\lambda=\lambda Q_1+(1-\lambda)Q_2$. Thus, $\bar{K}$ cannot be optimum if for some T it is found that $L(\bar{K}, T)>Q^*(T)$. The value of $\bar{K}$ which corresponds to the line tangent to $Q^*(T)$ yields one value of $Q(\bar{K},T)$ for one value of T and imaginary values of $Q(\bar{K}, T)$ for all other values of T. Since all lines above the tangent line lie above $Q^*(T)$ for some value of T and therefore yield two values of $Q(\bar{K}, T)$, those lines must be associated with a value of $\bar{K}$ greater than K^*. All lines below the line of tangency do not intersect $Q^*(T)$ and therefore do not yield real valued solutions. For $Q^*(T)$ convex, one and only one line in the family of $L(\bar{K}, T)$ lines will be tangent to $Q^*(T)$. Therefore, the point of tangency between that line and the $Q^*(T)$ must be the optimal solution.

With the aid of the tables, the convexity of $Q^*(T)$ for a given problem may be investigated and the UOS easily identified. An iterative procedure which will identify the unconstrained optimal solution in seven iterations is provided in the appendix.

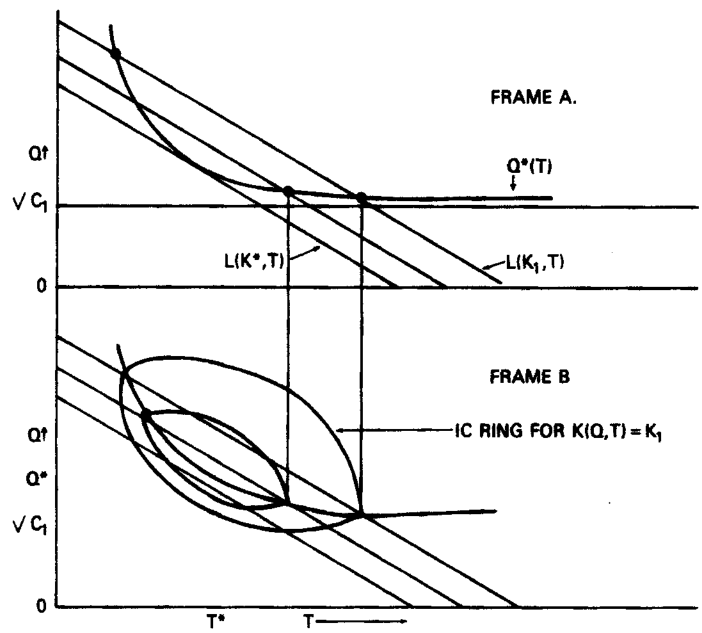

FIGURE 2

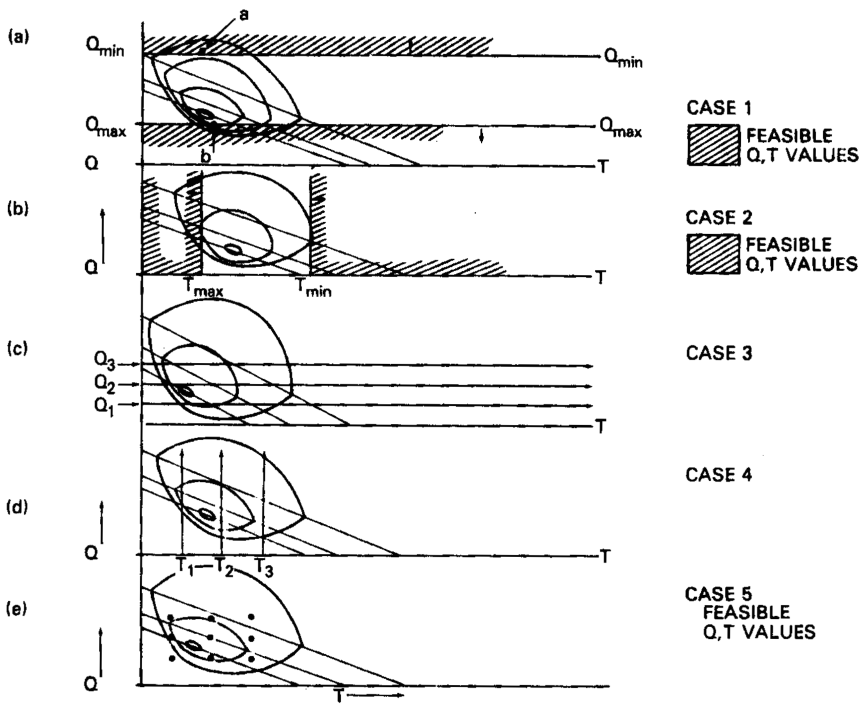

FIGURE 3.

6. FORMS OF CONSTRAINTS ON Q AND T

CASE 1: $Q>Q_{min}$ or $Q<Q_{max}$ where $Q_{min}>Q^{**}>Q_{max}$ and T is unconstrained.

Figure 3 Frame (a) displays an example of the situation. Since D_{TT} is strictly positive, the value of T which minimizes $K(Q, T)$ for a given value of Q is unique. Since both D_{QQ} and D_{TT} are strictly positive and the *IC* rings are nested, it follows that if $Q_{min}>Q^{**}$, there is no Q greater than Q_{min} which will yield a cost less than $K(Q_{min}, T^*(Q_{min}))$. Similar reasoning holds for $Q^{**}>Q_{max}$ and $K(Q_{max}, T^*(Q_{max}))$. The *COS* is found by calculating $K(Q, T)$ with the aid of the tables and evaluating successive values of T in the direction of decreasing values of $K(Q_{min}, T)$. Clearly, as one moves toward point $a(b)$ along line $Q=Q_{min}$ (Q_{max}), the cost function will decrease until the *COS* is passed. Thus, incrementing T by 0.1 unit from $S_L-3.0$ units and evaluating $K(Q, T)$ until an increase in $K(Q, T)$ is found will identify the *COS*.

CASE 2: $T>T_{min}$ or $T<T_{max}$ where $T_{min}\geq T^{**}\geq T_{max}$ and Q is unconstrained.

Refer to Frame (b) of Figure 3. The *COS* is found directly by substituting the value of T min (or T max) in (4). By similar arguments to those employed in Case 1, the value of Q which minimized $K(Q, T)$ for a given T is unique.

CASE 3: $Q=\Phi=\{Q_1, Q_2, Q_3, \ldots, Q_k\}$, T is unconstrained. Assume $Q_j<Q_{j+1}$, $j=1, \ldots, k-1$.

Refer to Frame (c) of Figure 3. It is apparent from Figure 3 that as successive values of $K(Q_j,\ T^*(Q_j))$ are evaluated, the first nondecreasing value of $K(Q_j,\ T^*(Q_j))$ indicates that the optimal constrained solution has been passed. Starting with $j=1$, calculate $K(Q_j,\ T^*(Q_j))$ as in Case 1. Continue to increment j until for some w,

$$K(Q_{w-1},T^*(Q_{w-1}))\geq K(Q_w,T^*(Q_w))\leq K(Q_{w+1},T^*(Q_{w+1})).$$

Since the isocost rings are nested and D_{QQ} and D_{TT} are strictly positive, it follows that $Q_w,\ T^*(Q_w)$ is the *COS*.

CASE 4: $T\epsilon\tau=\{T_1,\ T_2,\ T_3,\ .\ .\ .,\ T_l\}$, Q is unconstrained.

Refer to Frame (d) of Figure 3. Assume $T_j<T_{j+1}, j=1,\ .\ .\ .,\ l-1$. It is apparent from Figure 3 that as successive values of $K(Q^*(T_j),\ T_j)$ are evaluated, the first nondecreasing value of $K(Q^*(T_j),\ T_j)$ indicates that the optimal solution has been passed. Starting with $j=1$ calculate $K(Q^*(T_j),\ T_j)$ as in Case 2. Continue to increment j and calculate $K(Q^*(T_j),\ T_j)$ until for some V,

$$K(Q^*(T_{V-1}),T_{V-1})\geq K(Q^*(T_V),T_V)\leq K(Q^*)T_{V+1}),T_{V+1}).$$

If we employ arguments similar to Case 3, it follows that $Q^*(T_V),\ T_V$ is the *COS*.

CASE 5: $Q\epsilon\phi,\ T\epsilon\tau$.

Refer to Frame (c) of Figure 3. If the number of feasible $Q,\ T$ strategies is small, each of the $k\times l$ points may be evaluated. If the number is large, the cost surface may be generated to visually locate the *COS*. Figure 4 indicates the flow of the computations which will generate the $K(Q,\ T)$ surface from $Q_{\min}$ to $Q_{\max}$ and for T from $\overline{S}_L-3$ to $\overline{S}_L+3$.

7. USE OF THE TABLES

In order to be applicable to a specific problem, the units of measure must be standardized; therefore, all measurements are in terms of standard deviations. Thus, an annual sales rate of 1000 units, a lead time of 0.08 year, an order point of 90, and a standard deviation of 10 units would yield parameters as follows:

$$\overline{S}=100,\ \overline{S}_L=8,\text{ and }T=9$$

The C_F, C_I, and C_D would be in dimensions of \$/order, \$/σ·year, and \$/σ·year, respectively. Thus, for a T of 9 and $\overline{S}_L$ of 8, the corresponding time-weighted value is $EBP(T)=0.003700$. The interpretation is that for every order cycle we expect on the average to accumulate

$$0.\left(003700*\frac{10\text{ units}}{\sigma}*\frac{29\text{ days}}{\text{lead time}}\right)$$

or 1.07 unit days of backorders.

Continuing, let us assume that the cost parameters of the problem are as follows: $C_I=\$100/\sigma$ year, $C_F=\$200$, $C_D=\$40{,}000/\sigma$ year. We obtain $C_1=400$ and $C_2=6416$. If the procedure presented in the appendix were employed, the UOS would be found to be

$$Q^{**}=20.465\text{ and }T^{**}=9.1. \tag{13}$$

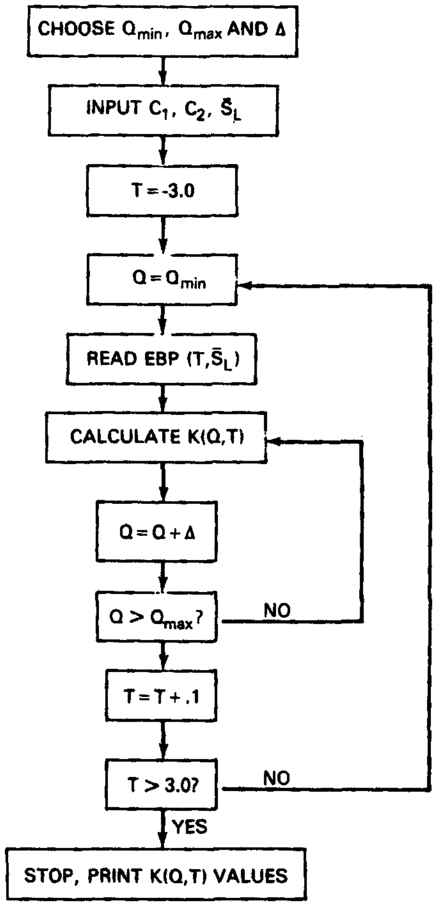

FIGURE 4. Flow chart to map the K (Q, T) surface.*

*The evaluation of 441 points on the $K(Q, T)$ surface has been found to require less than 1 second of CPU time on a Univac 1110.

Other, more complex constraining relationships would best be examined by observing the location of feasible Q, T values on the $K(Q, T)$ surface.

Table 1 displays the results of applying the techniques discussed in Section 6 in solving the problem above and constraining the solution by several example methods. The values in the Q and T columns are cost-minimizing values unless otherwise constrained. For example, for Case 1, a trigger level of 8.8 will minimize $K(Q,T)$, given Q must not be less that 40 and T is unconstrained.

Note that if the values of Q and T were rounded down from the *UOS* to (for Case 5) the next smallest feasible values of $Q=18$ and $T=8$, the resulting expected cost of that strategy would be 12% higher than the *COS* of $Q=22$ and $T=10$. Rounding Q up to 22 and T down to 8 is slightly better, costing 8% more than the *COS*. It is clear that simply rounding the *UOS* values up or down is not necessarily going to yield a very satisfactory solution to the constrained problem.

TABLE 1.

Case	Constraint	Values Q	Values T	COS?	Cost
1.	$Q \geq 40$, T unconstrained	40.0	8.8	yes	2626.30
2.	$T \geq 10.5$, Q unconstrained	20.0	10.5	yes	2250.00
3.	$Q \epsilon \phi = \{18, 22, 26\}$ T unconstrained	18.0	9.2	no	2172.33
		22.0	9.1	yes	2162.90
		26.0	9.0	no	2214.88
4.	$T \epsilon \tau = \{6, 8, 10\}$ Q unconstrained	47.3	6.0	no	4530.00
		23.8	8.0	no	2380.00
		20.0	10.0	yes	2200.00
5.	$Q \epsilon \phi$, $T \epsilon \tau$	18.0	6.0	no	6914.91
		18.0	8.0	no	2477.67
		18.0	10.0	no	2215.81
		22.0	6.0	no	5984.92
		22.0	8.0	no	2390.83
		22.0	10.0	yes	2212.94
		26.0	6.0	no	5402.63
		26.0	8.0	no	2392.24
		26.0	10.0	no	2272.49

BIBLIOGRAPHY

[1] Buckland, J. C. L., "A Nomogram for Stock Control," Operational Research Quarterly *20* (4) 445 (1969).

[2] Deemer, R. L., and D. Hoekstra, "Improvement of M.I.T. Non-Reparables Model," United States Army Logistics Management Center Publication, Fort Lee, Virginia (1968).

[3] Galliher, H. P., P. M. Morse, and M. Simond, "Dynamics of Two Classes of Continuous-Review Inventory Systems", Operations Research *7* (3) (1959).

[4] Herron, D. P., "Use of Dimensionless Ratios to Determine Minimum-Cost Inventory Quantities," Naval Research Logistics Quarterly 167–175 (July 1966).

[5] Holt, Charles C., Franco Modigliani, John F. Muth, and Herbert A. Simon, *Planning Production, Inventories, and Work Force.* (Prentice-Hall, Inc.: Englewood Cliffs, N.J., 1960).

[6] Koenigsberg, E., "On a Multiple Re-order Point Inventory Policy," Operational Research Quarterly *12*, 27 (1961).

[7] Lampkin, W., and A. D. J. Flowerdew, "Computation of Optimum Re-order Levels and Quantities for a Re-order Level Stock Control System," Operational Research Quarterly *14*, (3) 263 (1963).

[8] Thatcher, A. R., "Some Results on Inventory Problems," Journal of the Royal Statistical Society B *24*, (1) (1962).

APPENDIX

Search Procedure for T^{**}

It follows from (10) that as $\overline{K}$ increases, $L(\overline{K}, T)$ moves outward from the origin. For $Q^*(T)$ convex, the points of intersection of a line $L(\overline{K}, T)$ and the curve $Q^*(T)$ move apart and away from T^{**} as $\overline{K}$ increases. Thus, if an increase in T from, say, T_m to T_l along the curve $Q^*(T)$ results in an increase such that $K(Q^*(T_l),T_l)>K(Q^*(T_m),T_m)$, then all points beyond T_l may be eliminated from consideration in the search for T^{**}. Conversely, if a decrease in T from, say, T_q to T_p results in an increase such that $K(Q^*(T_p),T_p)>K(Q^*(T_q),T_q)$, then all points beyond T_q may be eliminated from consideration. The search procedure makes use of these observations.

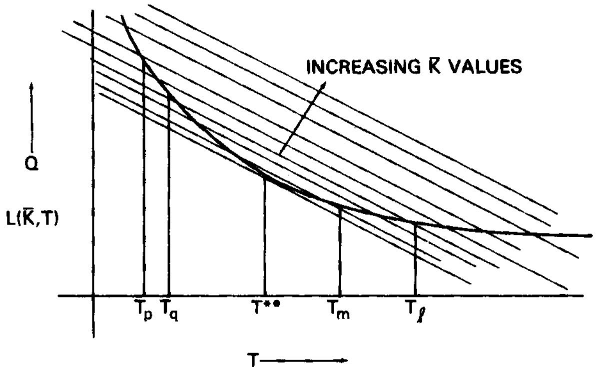

FIGURE A1.

For each value of $\overline{S}_L$, there are 61 tabled values of $T-\overline{S}_L$. The search procedure will be used to identify the tabled value of T and $Q^*(T)$ which minimizes (1). Let n, $1\leq n\leq 61$ denote a row of the table. Let $T_j=\overline{S}_L-3.1+(0.1)\ N_j$ where N_j denotes a value of n. Let $K_j=K(Q^*(T_j),\ T_j)$ where $Q^*(T_j)$ is determined in (4).

The search procedure at each iteration eliminates from further consideration sets of values of n. At the start of each iteration, the uneliminated values of n are divided into three mutually exclusive sets. Given N_1 and N_2, sets S_1, S_2, and S_3 are formed.

If $N_1<N_2$, S_1 contains values of $n\leq N_1$,
S_2 contains values of $n\geq N_2$,
S_3 contains values of $n>N_1$ and $<N_2$.

If $N_1>N_2$, S_1 contains values of $n\geq N_1$,
S_2 contains values of $n\leq N_2$,
S_3 contains values of $n>N_2$ and $<N_1$.

The procedures may now be presented.

STEP 1. Set $N_1=24$, $N_2=38$.

STEP 2. Form sets S_1, S_2, and S_3 as indicated above.

STEP 3. Calculate K_1 and K_2.

STEP 4. If $K_1>K_2$, eliminate set S_1 from further consideration and set N_1 equal to N_2. If $K_1<K_2$, eliminate set S_2 from further consideration.

Let the larger of the two remaining sets be denoted as S_0.

ITERATION 1. If $\forall\ n\epsilon S_0$, $n>N_1$, then $N_2=N_1+10$, otherwise $N_2=N_1-10$. Perform Steps 2 through 4 and proceed to Iteration 2.

ITERATION 2. If $\forall\ n\epsilon S_0$, $n>N_1$, then $N_2=N_1+4$, otherwise $N_2=N_1-4$. Perform Steps 2 through 4 and proceed to Iteration 3.

ITERATION 3. If $\forall\ n\epsilon S_0$, $n>N_1$, then $N_2=N_1+6$, otherwise $N_2=N_1-6$. Perform Steps 2 through 4 and proceed to Iteration 4.

ITERATION 4. If $\forall\ n\epsilon S_0$, $n>N_1$, then $N_2=N_1+2$, otherwise $N_2=N_1-2$. Perform Steps 2 through 4 and proceed to Iteration 5.

ITERATION 5. If $\forall\ n\epsilon S_0$, $n>N_1$, then $N_2=N_1+2$, otherwise $N_2=N_1-2$. Perform Steps 2 through 4 and proceed to Iteration 6.

ITERATION 6. There are 3 values that remain, one on each side of the present value of N_1. Set N_2 equal to N_1+1. Let S_2 consist only of the N_2. S_3 is null. S_1 consists of the remaining two values. Perform Steps 3 and 4 and proceed to Iteration 7.

ITERATION 7. If S_0 contains one value of n, go to Step 5. If S_0 contains 2 values, let $N_2=N_1-1$. Perform Steps 3 and 4 and proceed to step 5.

STEP 5. Only one value remains. All other values of n yield higher costs, thus the present value of N_1 is the optimal value.

Table A1 displays the rate at which values are eliminated as the search progresses.

Table A2 presents the results of application of the search procedure to the example problem presented in Section 7.

TABLE A1

j	0	1	2	3	4	5	6	7
Values eliminated at iteration j	24	14	10	4	4	2	1 or 2	1 or 0
Total values eliminated	24	38	48	52	56	58	59 or 60	60

TABLE A2

Iteration	N_1	N_2	S_1	S_3	S_2	K_1	K_2	Eliminate
0	24	38	1–24	25–37	38–61	2789.90	2181.01	S_1
1	38	48	25–38	39–47	48–61	2181.01	2180.08	S_1
2	48	52	39–48	49–51	52–61	2180.08	2241.04	S_2
3	48	42	48–51	43–47	39–42	2180.08	2156.55	S_1
4	42	44	39–42	43	44–47	2156.55	2158.88	S_2
5	42	40	42, 43	41	39, 40	2156.55	2162.96	S_2
6	42	43	41, 42	-------	43	2156.55	2156.76	S_2
7	42	41	42	-------	41	2156.55	2158.49	S_2

Thus, the *UOS* is $Q^{**}=20.465$ and $T^{**}=9.1$.

$T-\bar{S}_L$	$\bar{S}_L$																								
	2.0	2.5	3.0	3.5	4.0	5.0	6.0	7.0	8.0	9.0	10.0	12.0	14.0	16.0	18.0	20.0	25.0	30.0	35.0	40.0	50.0	60.0	70.0	85.0	100.
−3.0	2.320050	1.812819	1.500188	1.290055	1.135037	.918395	.772826	.667747	.588134	.525647	.475257	.398931	.343812	.302118	.269464	.243193	.195558	.163541	.140539	.123212	.098844	.082525	.070832	.058417	.049705
−2.9	2.159283	1.690287	1.402333	1.207582	1.063348	.861257	.725192	.626853	.552285	.493724	.446477	.374877	.323146	.283999	.253332	.228655	.183897	.153807	.132184	.115894	.092981	.077634	.066636	.054959	.046764
−2.8	2.004933	1.572805	1.308426	1.128261	.994337	.806201	.679267	.587413	.517703	.462923	.418705	.351660	.303195	.266506	.237757	.214617	.172637	.144405	.124114	.105826	.087317	.072909	.062584	.051618	.043923
−2.7	1.857008	1.460393	1.218415	1.052083	.927999	.753224	.635052	.549427	.484387	.433242	.391940	.329280	.283960	.249639	.222737	.201079	.161776	.135336	.116330	.102006	.081853	.068351	.058673	.048395	.041182
−2.6	1.715520	1.352970	1.132264	.979035	.864326	.702321	.592542	.512892	.452334	.404682	.366180	.307735	.265440	.233396	.208272	.188040	.151314	.126600	.108830	.095437	.076589	.063959	.054905	.045290	.038540
−2.5	1.580489	1.251000	1.049939	.909102	.803309	.653488	.551734	.477804	.421542	.377239	.341423	.287025	.247634	.217778	.194361	.175500	.141250	.118195	.101615	.089115	.071523	.059732	.051280	.042301	.035998
−2.4	1.451941	1.154319	.971404	.842270	.744938	.606717	.512624	.444162	.392009	.360911	.317668	.267146	.230540	.202782	.181003	.163457	.131585	.110123	.094683	.083043	.066656	.055672	.047796	.039429	.033556
−2.3	1.329924	1.062761	.896623	.778519	.689200	.562001	.475206	.411959	.363730	.235695	.294912	.248098	.214157	.188408	.168198	.151911	.122316	.102380	.088036	.077218	.061987	.051776	.044454	.036674	.031212
−2.2	1.214511	.976136	.825559	.717826	.636079	.519332	.439473	.381192	.336702	.301588	.273152	.229878	.198482	.174652	.155942	.140859	.113443	.094968	.081670	.071641	.057517	.048045	.041253	.034035	.028968
−2.1	1.105469	.394314	.758168	.660166	.585557	.478696	.405147	.351853	.310918	.278585	.252383	.212482	.183513	.161514	.144235	.130301	.104965	.870884	.075587	.066310	.053243	.044479	.038193	.031513	.026822
−2.0	1.004244	.817181	.694403	.605511	.537613	.440080	.373027	.323934	.286373	.256679	.232600	.195907	.169247	.148990	.133073	.120234	.096879	.801127	.069784	.061225	.049166	.041076	.035273	.029105	.024774
−1.9	.910358	.744629	.634213	.553826	.492222	.403468	.342291	.297425	.263057	.235864	.213798	.180147	.155679	.137077	.122455	.110656	.089184	.074696	.064259	.056383	.045384	.037836	.032493	.026813	.022824
−1.8	.823325	.676547	.577542	.505072	.449354	.368840	.313194	.272314	.240962	.216132	.195969	.165197	.142804	.125770	.112375	.101562	.081876	.068588	.059012	.051784	.041596	.034758	.029851	.024635	.020970
−1.7	.742603	.612826	.524327	.459207	.408977	.336173	.285719	.248587	.220074	.197471	.179104	.151049	.130616	.115065	.102829	.092949	.074954	.062801	.054040	.047426	.038100	.031840	.027347	.022570	.019214
−1.6	.667834	.553350	.474498	.416180	.371051	.305440	.259845	.226228	.200381	.179872	.163193	.137695	.119109	.104955	.093813	.084813	.068413	.057331	.049341	.043306	.034796	.029081	.024979	.020617	.017552
−1.5	.598724	.497998	.427980	.375937	.335533	.276612	.235549	.205218	.181866	.163318	.148222	.125125	.108274	.095433	.085319	.077147	.062248	.052175	.044910	.039422	.031679	.026480	.022746	.018776	.015985
−1.4	.535005	.446645	.384690	.338416	.302375	.249652	.212804	.185534	.164510	.147795	.134179	.113328	.098102	.086491	.077342	.069946	.056455	.047330	.040745	.035770	.038749	.024033	.020646	.017644	.014512
−1.3	.476422	.399157	.344538	.303548	.271521	.224522	.191579	.167152	.148293	.133283	.121047	.102290	.088581	.078119	.069871	.063201	.051028	.042789	.036842	.032347	.026003	.021739	.018677	.015420	.013130
−1.2	.422728	.355396	.307427	.271260	.242910	.201178	.171840	.150043	.133190	.119762	.108806	.091996	.079698	.070306	.062898	.056904	.045959	.038548	.033196	.029149	.023436	.019596	.016837	.013902	.011838
−1.1	.373680	.315214	.273253	.241470	.216478	.179571	.153548	.134175	.119174	.107209	.097437	.082430	.071439	.063040	.056411	.051045	.041242	.034599	.029801	.026171	.021046	.017599	.015123	.012487	.010634
−1.0	.329034	.278459	.241904	.214090	.192150	.159648	.136661	.119513	.106216	.095596	.086916	.073572	.063789	.056307	.050399	.045614	.036867	.030937	.026651	.023408	.018827	.015746	.013532	.010174	.009517
−.9	.288550	.244974	.213262	.189027	.169850	.141350	.121133	.106019	.904281	.084896	.077218	.065402	.056730	.050092	.044847	.040598	.032825	.027552	.023739	.020854	.016776	.014032	.012060	.009960	.008483
−.8	.251985	.214594	.187204	.166181	.140494	.124615	.106912	.093650	.083334	.075076	.068313	.057895	.050241	.044378	.039742	.035984	.029106	.024437	.021059	.018502	.014887	.012454	.010705	.008842	.007531
−.7	.219097	.187151	.163600	.145447	.130993	.109376	.093964	.082361	.073336	.066102	.060173	.051028	.044302	.039146	.035066	.031757	.025697	.021581	.018601	.016345	.013154	.011006	.009461	.007815	.006657
−.6	.189645	.162472	.142317	.126716	.114257	.095562	.082176	.072104	.064245	.067938	.052764	.044774	.038891	.034376	.030802	.027902	.022586	.018974	.016358	.014376	.011572	.009684	.008325	.006878	.005859
−.5	.163391	.140384	.123217	.109875	.099187	.083099	.071543	.062829	.056018	.050546	.046052	.039104	.033982	.030049	.026932	.024402	.019761	.016606	.014319	.012586	.010134	.008481	.007292	.006025	.005133
−.4	.140099	.120711	.106161	.094807	.085685	.071909	.061983	.054482	.048609	.043885	.040000	.033989	.029551	.026140	.023436	.021240	.017209	.014465	.012475	.010967	.008832	.007393	.006357	.005253	.004476
−.3	.119535	.103276	.091007	.081395	.073649	.061915	.053433	.047008	.041970	.037912	.034571	.029396	.025571	.022623	.020294	.018396	.014912	.012538	.010816	.009510	.007660	.006413	.005515	.004558	.003884
−.2	.101472	.087907	.077614	.069510	.062977	.053034	.045824	.040351	.036052	.032584	.029727	.025295	.022015	.019489	.017434	.015853	.012856	.010813	.009330	.008205	.006611	.005535	.004761	.003935	.003353
−.1	.085690	.074430	.065842	.059061	.053565	.045186	.039091	.034454	.030805	.027858	.025427	.021651	.018854	.016697	.014984	.013590	.011022	.009276	.008006	.007042	.005675	.004753	.004088	.003380	.002881
−.0	.071975	.062678	.055551	.049903	.045312	.038290	.033166	.029259	.026179	.023688	.021631	.018432	.016059	.014228	.012773	.011587	.009406	.007916	.006834	.006012	.004846	.004059	.003492	.002387	.002461
.1	.060124	.052490	.046608	.041930	.038116	.032265	.027982	.024709	.022124	.020030	.018300	.015605	.013604	.012058	.010828	.009826	.007980	.006718	.005801	.005104	.004116	.003448	.002967	.002453	.002091
.2	.049942	.043708	.038882	.035029	.031879	.027033	.023474	.020747	.018590	.016841	.015393	.013137	.011458	.010160	.009127	.008285	.006732	.005669	.004896	.004309	.003475	.002912	.002506	.002072	.001767
.3	.041246	.036184	.032247	.029093	.026507	.022516	.019576	.017318	.015529	.014077	.012873	.010994	.009595	.008512	.007649	.006945	.005646	.004756	.004109	.003617	.002918	.002445	.002105	.001741	.001484
.4	.033864	.029777	.026585	.024018	.021903	.018542	.016228	.014370	.012895	.011698	.010701	.009146	.007987	.007088	.006372	.005787	.004707	.003967	.003428	.003018	.002435	.002041	.001757	.001454	.001239
.5	.027635	.024356	.021783	.019707	.017996	.015339	.013370	.011850	.010642	.009658	.008841	.007563	.006608	.005867	.005276	.004793	.003901	.003288	.002842	.002503	.002020	.001694	.001458	.001206	.001029
.6	.022415	.019799	.017738	.016070	.014691	.012543	.010946	.009712	.008728	.007926	.007259	.006214	.005433	.004826	.004371	.003945	.003212	.002789	.002842	.002063	.001666	.001397	.001203	.000995	.000849
.7	.018066	.015993	.014352	.013020	.011916	.010191	.008905	.007908	.007113	.006463	.005922	.005074	.004438	.003944	.003556	.003227	.002629	.002218	.001918	.001690	.001365	.001145	.000986	.000816	.000696
.8	.014469	.012836	.011538	.010481	.009603	.008227	.007197	.006398	.005759	.005236	.004800	.004116	.003602	.003203	.002883	.002622	.002137	.001804	.001560	.001375	.001111	.000932	.000803	.000664	.000567
.9	.011512	.010235	.009215	.008381	.007687	.006597	.005779	.005142	.004632	.004214	.003865	.003317	.002905	.002584	.062327	.002116	.001726	.001458	.001261	.001112	.000898	.000754	.000649	.000537	.000459
1.0	.009100	.008106	.007310	.006658	.006113	.005255	.004608	.004104	.003700	.003368	.003091	.002655	.002326	.002070	.001865	.001697	.001385	.001170	.001013	.000893	.000722	.000606	.000522	.000432	.000369
1.1	.007144	.006377	.005760	.005252	.004828	.004157	.003650	.003254	.002936	.002674	.002455	.002110	.001851	.001648	.001485	.001351	.001104	.000932	.000807	.000714	.000576	.000483	.000416	.000345	.000294
1.2	.005571	.004982	.004507	.004115	.003786	.003265	.002871	.002562	.002313	.002108	.001937	.001666	.001462	.001302	.001174	.001069	.000873	.000738	.000639	.000564	.000456	.000383	.000330	.000273	.000233
1.3	.004314	.003865	.003502	.003201	.002949	.002547	.002242	.002003	.001809	.001650	.001517	.001306	.001147	.001022	.000922	.000839	.000686	.000580	.000503	.000443	.000359	.000301	.000260	.000215	.000184
1.4	.003317	.002978	.002702	.002473	.002280	.001973	.001739	.001554	.001405	.001283	.001180	.001016	.000893	.000796	.000718	.000654	.000535	.000453	.000392	.000346	.000280	.000235	.000203	.000168	.000144
1.5	.002533	.002278	.002070	.001897	.001750	.001517	.001338	.001198	.001084	.000996	.000911	.000785	.000690	.000616	.000556	.000507	.000415	.000351	.000304	.000268	.000217	.000183	.000157	.000130	.000111
1.6	.001920	.001729	.001574	.001444	.001334	.001158	.001023	.000916	.000830	.000758	.000698	.000602	.000530	.000473	.000427	.000389	.000319	.000270	.000234	.000207	.000167	.000141	.000121	.000100	.000086
1.7	.001445	.001304	.001188	.001091	.001009	.000877	.000776	.000695	.000630	.000576	.000531	.000458	.000403	.000360	.000325	.000297	.000243	.000206	.000179	.000158	.000128	.000107	.000093	.000077	.000066
1.8	.001079	.000975	.000890	.000818	.000757	.000659	.000584	.000524	.000475	.000436	.000401	.000346	.000305	.000273	.000246	.000225	.000184	.000156	.000135	.000126	.000097	.000082	.000070	.000058	.000050
1.9	.000800	.000724	.000662	.000609	.000564	.000492	.000436	.000392	.000356	.000326	.000300	.000260	.000229	.000205	.000185	.000169	.000138	.000117	.000102	.000090	.000073	.000061	.000053	.000044	.000037
2.0	.000589	.000534	.000488	.000450	.000417	.000364	.000323	.000291	.000264	.000242	.000223	.000193	.000170	.000152	.000138	.000126	.000103	.000088	.000076	.000067	.000054	.000046	.000040	.000033	.000026
2.1	.000430	.000390	.000357	.000330	.000306	.000268	.000238	.000214	.000195	.000178	.000165	.000143	.000126	.000113	.000102	.000093	.000076	.000065	.000056	.000050	.000040	.000034	.000029	.000024	.000021
2.2	.000311	.000283	.000260	.000240	.000223	.000195	.000174	.000156	.000142	.000130	.000120	.000104	.000092	.000083	.000075	.000068	.000056	.000048	.000041	.000037	.000030	.000025	.000022	.000018	.000015
2.3	.000224	.000204	.000187	.000173	.000161	.000141	.000126	.000113	.000103	.000095	.000087	.000076	.000067	.000060	.000054	.000050	.000041	.000035	.000030	.000027	.000022	.000018	.000016	.000013	.000011
2.4	.000160	.000146	.000134	.000124	.000115	.000101	.000090	.000081	.000074	.000068	.000063	.000055	.000048	.000043	.000039	.000036	.000030	.000025	.000022	.000019	.000016	.000013	.000011	.000009	.000008
2.5	.000113	.000103	.000095	.000088	.000082	.000072	.000064	.000058	.000053	.000049	.000045	.000039	.000035	.000031	.000028	.000026	.000021	.000018	.000016	.000014	.000011	.000009	.000008	.000007	.000006
2.6	.000079	.000073	.000067	.000062	.000058	.000051	.000045	.000041	.000037	.000034	.000032	.000028	.000025	.000022	.000020	.000018	.000015	.000013	.000011	.000010	.000008	.000007	.000006	.000005	.000004
2.7	.000055	.000051	.000047	.000043	.000040	.000036	.000032	.000029	.000026	.000024	.000022	.000020	.000017	.000016	.000014	.000013	.000011	.000009	.000008	.000007	.000006	.000005	.000004	.000003	.000003
2.8	.000038	.000035	.000032	.000030	.000028	.000025	.000022	.000020	.000018	.000017	.000016	.000014	.000012	.000011	.000010	.000009	.000007	.000006	.000005	.000005	.000004	.000003	.000003	.000002	.000002
2.9	.000026	.000024	.000022	.000021	.000019	.000017	.000015	.000014	.000013	.000012	.000011	.000009	.000008	.000007	.000007	.000006	.000005	.000004	.000004	.000003	.000003	.000002	.000002	.000002	.000001
3.0	.000018	.000016	.000015	.000014	.000013	.000012	.000010	.000009	.000009	.000008	.000007	.000006	.000006	.000005	.000005	.000004	.000004	.000003	.000003	.000002	.000002	.000002	.000001	.000001	.000001