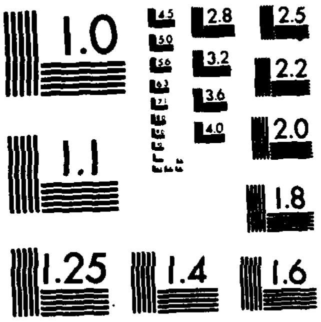
AD-A136 792   ANALYSIS OF INVENTORY MODELS WITH BUDGET CONSTRAINT(U)   1/1
NAVAL POSTGRADUATE SCHOOL MONTEREY CA   S J KANG SEP 83

UNCLASSIFIED   F/G 15/5   NL

END
DATE
FILMED
2-84
DTIC


1.0
4.5
5.0
5.6
6.3
7.1
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

②

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ANALYSIS OF INVENTORY MODELS
WITH BUDGET CONSTRAINT

by

Sung Jin, Kang

September 1983

Thesis Advisor: F.R. Richards

Approved for public release; distribution unlimited.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. A136792	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Analysis of Inventory Models with Budget Constraint		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Sung Jin, Kang		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 75
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Units Short
Time-Weighted Units Short
Availability
Lagrangian Multiplier Search
Marginal Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis addresses the problem of determining the optimal number of spares for a multi-item inventory system with a procurement budget constraint. Various inventory models are considered with objective functions like time-weighted units short, units short, essentiality-weighted units short and pseudo-availability. Solution algorithms

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(20. ABSTRACT Continued)

are derived using the generalized Lagrange multiplier approach and a marginal analysis approach.

Sample data and output results are provided and comparisons of the alternative models are given. Finally, a discussion and example is given of the use of the models as a means of estimating the budget required to attain a specified level of performance.

Analysis of Inventory Models with Budget Constraint

by

Sung Jin, Kang
Major, Republic of Korea Army
B.S., Republic of Korea Military Academy, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL

September 1983

Author: Kang Sung Jin

Approved by:
Thesis Advisor

Gilbert T. Howard
Second Reader

Chairman, Department of Operations Research

K.T. Marshall
Dean of Information and Policy Sciences

ABSTRACT

This thesis addresses the problem of determining the optimal number of spares for a multi-item inventory system with a procurement budget constraint. Various inventory models are considered with objective functions like time-weighted units short, units short, essentiality-weighted units short and pseudo-availability. Solution algorithms are derived using the generalized Lagrange multiplier approach and a marginal analysis approach.

Sample data and output results are provided and comparisons of the alternative models are given. Finally, a discussion and example is given of the use of the models as a means of estimating the budget required to attain a specified level of performance.

TABLE OF CONTENTS

LIST OF TABLES

I. INTRODUCTION

In today's world, while all systems are becoming more and more sophisticated, the control and maintenance of inventories of these systems is a problem common to all enterprises and military services. In private and commercial concerns the effective control of inventories can result in decreased costs, increased sales and profits and consumer satisfaction. In the military proper management of inventories may contribute to increased availability and readiness, decreased inventory investment and system costs.

For each component of each weapon system two fundamental questions must be answered:

(1) When to replenish the inventory;

(2) How much to buy for the replenishment.

In order to answer these questions, many inventory models have been developed in the past 30 years. See for example, Hadley and Whitin [Ref. 1], Muckstadt [Ref. 2] and Eriksson [Ref. 3]. Most previous work solves a variety of cost minimization problems considering expected values of steady state variable costs associated with shortage cost, ordering cost and storage cost.

Such models may be appropriate for the commercial sector, but are not always appropriate in the military world.

In the commercial sector, the objective function of the inventory model is to maximize profit or minimize the average annual costs. Non-cost oriented objective functions frequently are used in the military inventory systems. For example, attempts are often made to maximize availability or fill rate, or minimize the number of backorders or expected time weighted stockouts, or minimize the probability of a stockout with a budget constraint.

Obviously costs are important in every inventory model. However, many real-world inventory problems are so complicated, one cannot represent accurately the real situation. Thus, some simplifications and approximations are used when constructing a mathematical model of any real world system. If this is not done, the results obtained by use of the model can easily lead to operating rules which are worse than those currently in use, worse than those which could be derived from simple heuristic intuitive considerations.

Many of the inventory problems are viewed as single period problems. For example, initial provisioning, allowance list determination and the fly-away kit problem are single period problems. These models are perhaps the simplest of the models in which demand is treated as a stochastic variable.

Reasonable objective functions in these models are to maximize performance subject to a constraint on the resources. Typical measures of performance might be availability, time-weighted units short, fill rate, the number of backorders, and mean supply response time.

This thesis considers various single period models which attempt to maximize performance subject to budget constraint.

Chapter II describes the general single period problem and introduces the method used in this thesis of solving those problems.

Chapters III and IV develop the time-weighted units short model and the availability model, and explain the solution procedure. Sample data runs for both models are provided.

Chapter V provides a comparison of models considered in the thesis and discusses some of the properties of each model, and Chapter VI discusses the use of models for purposes of determining the budget required.

Chapter VII summarizes the results of the research and concludes with some suggestions for additional research.

II. THE GENERAL PROBLEM

In this chapter, we consider the general single period model as a process for transforming resources into new distributions of inventory positions over the line items in the inventory.

The essential problems of control in a line item-inventory control system with multiple line items are:

(1) How much resources to commit at a point in time;

(2) How shall these resources be allocated to achieve system objectives.

In a typical continuous review inventory system, we can determine the optimal order quantity (Q) and reorder point (r) for a given item by minimizing the average annual variable costs. But in applying this theory to the real world inventory systems which consist of multiple line items, it is frequently the case that resulting minimum cost solutions are not feasible because of a budget limitation or some other constraint. Thus, in a constrained multi-item inventory system, the typical continuous-review policy is sometimes inappropriate. In the following section we discuss several objective functions to guide the line item inventory control system in determining how to allocate availabile procurement funds at a particular replenishment epoch.

A. GENERAL FORM OF OBJECTIVES WITH CONSTRAINTS

Consider the case in which an administrator, responsible for the replenishment decisions, determines replenishment of stocks of various line items on a periodic basis. Suppose that a fixed amount of procurement budget has been allocated to the replenishment epoch at hand and that a target number of reorder actions has been established as a working constraint for the allocation epoch. The administrator's task is to transform the available resources into replenishment orders for different items.

1. Measure of Effectiveness and Objective Function

Daeschner [Ref. 5] examined the constrained line-item allocation problems. He considered several possible objective functions which can be adapted to the case where unsatisfied demands are backordered and to the case where unsatisfied demands are lost sales. Let $\pi_j > 0$ be the penalty (reward) per unit for item j and let D_j be the demand for item j in a period. Let X_j be the inventory position for item j after ordering in a period. Let $D_j = d_j$. Then the number of sales for item j in the period is given by

$$\begin{array}{lll} d_i & \text{if} & d_j \le X_j \\ X_j & \text{if} & d_j > X_j \end{array}$$

The expected sales for item j is therefore

$$\sum_{d_j=1}^{X_j} d_j \; p(D_j = d_j) + X_j \; p(D_j > X_j)$$

which is equivalent to

$$E(D_j) - \sum_{d_j=X_j+1}^{\infty} (d_j - X_j) p(D_j = d_j) \; .$$

We assume that the inventory system seeks to minimize the expected penalty incurred, or, equivalently, to maximize the expected penalty avoided. Mathematically, the objective is to maximize

$$Z(\underline{X}) = \sum_{j=1}^{N} \pi_j (E(D_j) - \sum_{d_j=X_j+1}^{\infty} (d_j - X_j) p(D_j = d_j))$$

Several interpretations and uses of the penalty coefficient π_j are possible. Four are illustrated in Table I below. Each reflects a formulation of system objectives which has been adopted or considered by the Navy Supply System. Daeschner [Ref. 5] also considered π_j as a linear combination of various coefficients in his line item allocation model.

There are many other types of objective functions which are currently used in the military. For example,

(1) Minimize units short in a given period

$$Z(\underline{X}) = \sum_{j=1}^{N} \sum_{d_j=X_j+1}^{\infty} (d_j - X_j) \; p(D_j = d_j)$$

TABLE I

INTERPRETATIONS AND USES OF π_j

Penalty Coefficients	Objective
$\pi_j = c_j$	Maximize expected sales from stock.
$\pi_j = 1/\mu_j$	Maximize the expected requisitions filled (μ_j = average quantity of item j demanded per requisition).
$\pi_j = 1$	Maximize the expected number of units issued from stock.
$\pi_j = LT_j + TMNIS - TMISS$	Maximize expected customer waiting time per unit avoided by issue from stock, where LT_j is the lead time for item j, TMNIS is the calendar time anticipated to process a request and TMISS is the time to affect issue from stock of a demand, available item.

(2) Minimize time-weighted units short

$$Z(\underline{X}) = \sum_{j=1}^{N} TWUS_i(X_i) \cdot E_i$$

where:

$TWUS(X_i)$ = time weighted units short

E_i = essentiality

(3) Maximize system availability

$$Z(\underline{X}) = \prod_{i=1}^{N} A_i(X_i)$$

where:

A_i = Availability for each item.

The objective functions (2) and (3) will be explained in Chapters III and IV.

2. <u>The Line-Item Allocation Model and Solution Procedure</u>

In the previous sections, many kinds of objective functions are introduced. If we define them correctly, those objective functions can be solved by various techniques. It is evident that an actual inventory system with limited resources might be unable to carry out a prescribed inventory policy if either the amount of procurement funds available or the number of replenishment actions exceed the available resources. The problem is made more complicated by the fact that the objective functions are "non-linear" and the requirement that the X_j's must be integers. The problem is stated mathematically as

$$\max \quad Z(\underline{s})$$

$$\text{(A1)} \quad \text{s.t.} \quad \sum_{j=1}^{N} c_j s_j \le B$$

$$\sum_{j=1}^{N} H(s_j) \le R$$

$\underline{s}$ = $(s_1, s_2, \ldots, N)$: integers

c_j = the unit price of item j

s_j = the number of buys of item j

$$H(s_j) = 1 \quad \text{if} \quad s_j > 0$$
$$= 0 \quad \text{otherwise}$$

B = the procurement budget limit at the reallocation epoch

R = the maximum number of individual procurement activities allowed in the present allocation.

To solve the problem (A1), the generalized Lagrange multiplier (GLM) method of Everett [Ref. 4] can be used. Using this method, the problem can be reexpressed as

$$\text{(A2)} \qquad \max_{\underline{s}} \; L(\underline{S},\underline{\lambda}) = Z(\underline{s}) - \lambda_1((\sum_{j=1}^{N} c_j S_j) - B)$$
$$- \lambda_2((\sum_{j=1}^{N} H(s_j)) - R)$$

$\underline{S} \in s$ and $\lambda_1, \lambda_2 \geq 0$ with optimal solution $\underline{S}^*(\underline{\lambda})$.

Problem (A2) is the Lagrangian problem associated with (A1). Using Everett's theorem, one can determine a bound on the optimal solution, $Z(\underline{s}^*)$ to be

$$Z(\underline{s}^*) \leq Z(\underline{s}^*(\underline{\lambda})) - \lambda_1(B(\underline{\lambda}) - B) - \lambda_2(R(\underline{\lambda}) - R)$$

where

$$B(\underline{\lambda}) = \sum_{j=1}^{N} c_j S_j^*(\underline{\lambda})$$

$$R(\underline{\lambda}) = \sum_{j=1}^{N} H(S_j^*(\underline{\lambda}))$$

and $\underline{s}^*(\lambda)$ is the optimal solution vector for a given pair (λ_1,λ_2). In solving problem (A2), we can separate the N-variable optimization problem into N one-variable problems. Choosing trial values of λ_1 and λ_2 we maximize

$$\text{(A3)} \qquad L_j(s_j,\underline{\lambda}) = z_j(s_j) - \lambda_1 c_j s_j - \lambda_2 H(s_j) \ .$$

When considering the integer nature of decision variables, the optimal solutions for (A3) are determined by finding the values s_j^* such that

$$L_j(s_j^*+1,\underline{\lambda}) - L_j(s_j^*,\underline{\lambda}) \leq 0 \quad \text{and} \quad L_j(s_j^*,\underline{\lambda}) - L_j(s_j^*-1,\underline{\lambda}) > 0.$$

Thus s_j^* is the smallest value such that

$$\Delta L_j(s_j,\underline{\lambda}) = L_j(s_j+1,\underline{\lambda}) - L_j(s_j,\underline{\lambda}) \leq 0$$

In order to get an optimal solution, Daeshner [Ref. 5] used an interactive computer program, which evaluates the current optimal solutions with λ_1 and λ_2. Each time, the user can select a pair λ_1, λ_2 and objective function type to be considered. Then the user is provided with output which indicates the budget consumed, the number of stock replenishments generated, the achieved objective function value and a maximum attainable value for the objective function.

After examining the output, the user can modify the input parameters and continue or terminate the run. Decreasing the non-negative multiplier values tends to use more of the corresponding resources, increasing the values used, less. When the replenishment actions generated by a pair of values (λ_1, λ_2) exactly consume the available resources, B and R, the solution is optimal. Frequently exact equality may be impossible because of integer nature of the problem. Thus the solution obtained may not be optimal, but the difference is not likely to be significant.

B. AUTOMATING SEARCH ON THE LAGRANGE MULTIPLIER

The interactive search method cannot guarantee an optimal solution and it requires trial and error to get the approximate optimal solution. Consider the case in which there is only a single constraint, with the same type of objective functions. The mathematical program is then

$$\text{(B1)} \qquad \begin{array}{l} \max\limits_{\underline{s}} \; Z(\underline{s}) \\ \text{s.t.} \quad \sum\limits_{j=1}^{N} c_i s_i \le B \end{array}$$

$\underline{s} = (s_1, s_2, \ldots, s_N) =$ integer number of buys

$B =$ budget limit

$c_i =$ price of item i.

We can rewrite the above equation using a Lagrange multiplier, as:

$$(1) \qquad L(\underline{s},\theta) = Z(\underline{s}) - \theta[\sum_{j=1}^{N} c_i s_i - B]$$

Then separate the equation.

$$(2) \qquad L(s_1, s_2, \ldots, s_N) = \sum_{j=1}^{N} (Z(s_i) - \theta c_i s_i) + \theta B$$

Equation (2) can be maximized by maximizing each subobjective function. If $Z(s_i)$ is differentiable with respect to each s_i, the optimal solution is obtained by

$$\frac{\partial L}{\partial s_i} = \frac{dZ(s_i)}{ds_i} - \theta c_i, \qquad i = 1,2,\ldots,N$$

Thus set $\frac{\partial L}{\partial s_i} = 0$ and get

$$\theta = \frac{dZ_i(s_i)}{ds_i} / C_i$$

where θ is such that $\sum_{j=1}^{N} c_i s_i = B$. Everett [Ref. 4] shows that θ can also be interpreted as a shadow price for the objective function: i.e., $\theta = \partial Z/\partial B$. Due to the integer nature of s_i, it is often impossible to get an exact optimal solution. Difference equations must be used because the region in which the solution is desired consists of a set of discrete points. Therefore, let

$$\text{(3)} \quad \begin{aligned} \Delta L_i(s_i,\theta) &= L_i(s_i,\theta) - L_i(s_i-1,\theta) \\ &= Z_i(s_i) - \theta c_i s_i - Z_i(s_i-1) + \theta c_i(s_i-1) \\ &= \Delta Z_i(s_i) - c_i\theta \end{aligned}$$

We know that Equation (3) is a concave function at the point $\underline{s} \geq 0$. The optimal solution must satisfy $\Delta L_i(s_i,\theta) \geq 0$ and $\Delta L_i(s_i+1,\theta) < 0$. Thus the optimal solutions are given by finding the largest s_i's such that

$$\Delta L_i(s_i,\theta) \geq 0$$

or equivalently

$$\text{(4)} \qquad \Delta Z_i(s_i) - c_i\theta \geq 0 \qquad i = 1,\ldots,N.$$

The Lagrangian multiplier θ can be found by the following search algorithm.

STEP 1. Find an initial upper bound θ_u. Let all s_i be assigned zero at the beginning and find the change of objective function per unit dollar as a result of increasing to one unit.

$$\theta_1 = \frac{\Delta Z_1(1)}{c_1}$$

$$\theta_2 = \frac{\Delta Z_2(1)}{c_2}$$

$$\vdots$$

$$\theta_n = \frac{\Delta Z_n(1)}{c_N}$$

where $\Delta Z_i(1) = Z_i(1) - Z_i(0)$.

Because of decreasing marginal returns or objective function values and because of the interpretation of θ, an upper bound on θ is given by: $\theta_u = \max[\theta_1, \theta_2, \ldots, \theta_n]$.

STEP 2. The initial θ_o will be

$$\theta_o = \frac{\theta_L + \theta_u}{2}$$

where $\theta_L = 0$.

Find for each i, the largest s_i so that

$$\frac{\Delta Z_i(s_i)}{c_i} \geq \theta_o$$

and evaluate the objective function and the budget required.

STEP 3. If the budget used is greater than the given budget, let

$$\theta_1 = \frac{(\theta_o + \theta_u)}{2}$$

otherwise

$$\theta_1 = \frac{\theta_o + \theta_L}{2}$$

Each time update the S vector, the objective function values, the upper bound of the objective, and the amount of budget consumed.

STEP 4. Stopping rule.

Stop when the used budget is equal to the given budget or the difference between the current upper bound and the objective function value is less than some limit (ε). Otherwise go to step 2, and continue until the above conditions are satisfied.

A FORTRAN program for this algorithm is given in Appendix B.

C. MARGINAL ANALYSIS PROCESS

The theory of marginal analysis has been used in many inventory models when resource constraints are active. In an economic sense, $\Delta Z_i(s_i)/c_i$ can be interpreted as the marginal increase in the objective function per dollar spent achieved by adding one more unit of stock. It is reasonable for an inventory controller who has a scarce resource such as a procurement budget to buy an item which gives the maximum benefit per dollar spent.

By using a simple computerized algorithm, the line item allocation problem can be solved easily. The first step is to set all $s_i = 0$ and compute

$$\max_{i} \left[\frac{\Delta Z_1(s_1+1)}{c_1}, \frac{\Delta Z_2(s_2+1)}{c_2}, \ldots, \frac{\Delta Z_n(s_n+1)}{c_n}\right]$$

If the maximum is taken on for item j, set $s_j = 1$ and deduct the unit price for unit j from the budget. The second step is then to recompute ΔZ_j and then find

$$\max\{\max_{i \neq j}\{\frac{\Delta Z_i(s_i)}{c_i}\}, \frac{\Delta Z_j(s_j)}{c_j}\} .$$

The next unit is assigned to the index j where the maximum is taken on, etc. This is continued until adding an additional unit exceeds the budget constraint. It should be noted, however, that the method described does not insure optimality [Ref. 1]. Specifically the method may stop too soon. If the item i selected from the marginal analysis has a c_i value greater than the remaining budget, the procedure terminates even though some other item j may have a c_j value less than the remaining budget. An obvious improvement in this area could be the inclusion of a subroutine that would select from the remaining items the best one from those having c_j's smaller than the remaining budget. A FORTRAN program for performing this marginal analysis is provided in Appendix C.

D. SAMPLE DATA RUNS OF UNITS SHORT MODEL

A weapon system consists of 10 components. The system manager wants to minimize the number of units short by supplying spare parts to support the weapon system. Suppose that the demand rate, lead time, price and essentiality code for each item i are known. The objective function can be expressed by

$$\text{(C1)} \qquad \text{minimize } Z(\underline{s}) = \sum_{i=1}^{10} \sum_{d_i=s_i+1}^{\infty} (d_i-s_i)\,p(D_i=d_i)\,E_i$$

$$\text{subject to} \quad \sum_{i=1}^{10} s_i c_i \leq B$$

where:

E_i = essentiality code

B = budget limit

$$p(D_i=d_i) = \frac{e^{-\lambda_i T_i}(\lambda_i T_i)^{d_j}}{(d_i)!}$$

The approximate solution of (C1) can be obtained by the marginal analysis method. Table II shows the computational results for this system with known input data.

TABLE II

THE RESULT OF UNITS SHORT MODEL

No.	λ_i	Lead Time	Price ($)	Essen.	Allocation	Units Short
1	1.0	1.0	10.2	1.0	5.0	0.0007
2	0.1	1.0	20.0	1.0	1.0	0.0048
3	3.0	1.0	100.0	1.0	2.0	1.2489
4	25.0	1.0	2.0	3.0	42.0	0.0013
5	1.0	1.0	5.0	1.0	5.0	0.0007
6	0.5	1.0	5.0	3.0	4.0	0.0002
7	10.0	1.0	1.0	1.0	21.0	0.0012
8	5.0	1.0	100.0	1.0	4.0	1.4368
9	1.0	1.0	50.0	1.0	3.0	0.0233
10	2.0	1.0	100.0	1.0	2.0	0.5413

Table II shows several properties of the units short model. First of all, more than one unit short in a year occurs in the high cost items (items 3 and 8). Second, low demands and low price items are allocated enough. Items 2, 5 and 6 are allocated more than five times their mean demand. Also this model tends to stock more of the high demands and low price items.

Finally, the essentiality weights cause greater allocations to be provided to those items with high essentiality than would be provided with equal weights.

Other results include:

Total objective value	3.26
Shadow price	0.001899
Budget limit	\$1170
Budget left	\$0.0

The shadow price is the last maximum value of $\frac{\Delta Z(s_i)E_i}{c_i}$. It can be interpreted approximately as the amount of decrease in the objective function achieved by adding one more dollar.

III. TIME WEIGHTED UNITS SHORT MODEL

A. DESCRIPTION OF MODEL

In the previous chapter we have discussed various objective functions and solution methods for the single period inventory problem. In the military services, many measures of effectiveness have been used to indicate system performance. Among these measures are fill rate, availability, mean supply response time, the number of stockouts, and time-weighted units short. In this chapter we consider a model which minimizes time-weighted-units-short (TWUS).

Suppose that a weapon system consists of n components and the objective is to allocate a given budget for spare parts so as to minimize time-weighted-units short for the entire system. Assume that

(1) procurement lead time and repair lead time are known constants.

(2) demands for each installed unit have a known distribution.

(3) the total amount of procurement budget available to spend on all components is fixed.

(4) the objective is to minimize essentiality weighted TWUS. Mathematically, the model can be written as

$$\min_{\underline{s}} \sum_{i=1}^{n} \text{TWUS}_i(s_i)E_i \cdot \frac{1}{\text{SLT}}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} c_i s_i \leq B$$

where:

$\text{TWUS}_i(s_i)$ = time weighted units short when there are s_i units for item i

SLT = total sum of lead time demand $(\sum_{i=1}^{n} \lambda_i T_i)$

E_i = essentiality code for item i

B = budget limit in a given period

C_i = price of each item.

In the above problem, if the TWUS is properly defined, this model will be solved easily by using the methods explained in Chapter II.

B. POISSON DEMAND CASE

We shall now determine an exact expression for the $\text{TWUS}_i(s_i)$ for the case in which demands are Poisson distributed. Let the mean rate of demand be λ_i units per year and the lead time be a constant T_i. In addition to treating the demand variable as being discrete, the number of buys s_i also will be treated as a discrete variable. Thus if D_i is the lead time demand item j:

$$p(D_i = s_i) = \frac{e^{-\lambda_i T_i}(\lambda_i T_i)^{s_i}}{(s_i)!} \tag{1}$$

$$= p(s_i;\lambda_i T_i)$$

Let

$$\overline{p}(s_i) = \text{prob}(D_i \geq s_i) = \sum_{d_i = s_i}^{\infty} p(d_i;\lambda_i T_i) \tag{2}$$

If there are s_i units of stock for item i, Richards and McMasters [Ref. 8] show that the expected time-weighted units short in $(0,T_i)$ is given by

$$E[TWUS_i(s_i)] = \frac{T_i}{2}\{\overline{p}(s_i+1)[\lambda_i T_i - 2s_i + \frac{s_i(s_i+1)}{\lambda_i T_i}]$$

$$+ p(s_i;\lambda_i T_i)(\lambda_i T_i - s_i)\} \tag{3}$$

For those cases where the expected lead time demand is large, the Poisson probabilities in (3) can be approximated by a normal distribution with mean $\lambda_i T_i$ and variance $\sigma_i^2 = \lambda_i T_i$. Let

$$\phi(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2)$$

be the standard normal probability density function and let $\Phi(x) = \int_x^{\infty} \phi(u)du$ be the complementary cumulative distribution function for the standard normal. Then expression (3) can be rewritten in terms of the normal probability function as follows:

$$E[TWUS_i(s_i)] = \frac{T_i}{2}\{\Phi(\frac{s_i+1-\lambda_i T_i}{\sigma_i})[\lambda_i T_i - 2s_i + \frac{s_i(s_i+1)}{\lambda_i T_i}] + \frac{1}{\sigma_i}\,\phi(\frac{s_i-\lambda_i T_i}{\sigma_i})(\lambda_i T_i - s_i)\} \qquad (4)$$

This expression should be used in those cases in which $\lambda_i T_i$ is large. We have developed the expression for the expected time-weighted units short in a period of length T_i when there are s_i units of stock for item i.

In the next section we write the expression for the total essentiality-weighted time weighted units short over all items, and we provide a solution procedure for allocating the given budget optimally.

C. SOLUTION PROCEDURE

The mathematical program for the time-weighted-units short problem is:

$$\text{(C1)} \quad \min Z(\underline{s}) = \frac{1}{SLT}\sum_{i=1}^{N} \frac{E_i T_i}{2}\{\bar{p}(s_i+1)[\lambda_i T_i - 2s_i + \frac{s_i(s_i+1)}{\lambda_i T_i}] + p(s_i;\lambda_i T_i)(\lambda_i T_i - s_i)\}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} c_i s_i \leq B$$

To solve this problem we can use the Lagrangian multiplier technique. Let

$$L(s_1,s_2,\ldots,s_n;\theta) = \sum_{i=1}^{N} Z_i(s_i) + \theta(B - \sum_{i=1}^{N} c_i s_i) \qquad (5)$$

Here Equation (5) is separable in the items, and minimization of the total objective function is accomplished by minimizing the individual functions $Z_i(s_i)$ subject to budget constraints. Consider a single item i. Let

$$\begin{aligned} \Delta L_i(s_i) &= L_i(s_i-1) - L(s_i) \\ &= Z_i(s_i-1) - Z_i(s_i) + \theta c_i s_i - \theta c_i(s_i-1) \\ &= \Delta Z_i(s_i) + \theta c_i \qquad (6) \end{aligned}$$

where

$$\Delta Z_i(s_i) = E_i[\mathrm{TWUS}_i(s_i-1) - \mathrm{TWUS}_i(s_i)] \qquad (7)$$

As shown earlier

$$\begin{aligned} \mathrm{TWUS}(s-1) &= \frac{T}{2}\{\overline{P}(s)[\lambda T-2(s-1)+\frac{s(s-1)}{\lambda T}]+p(s-1;\lambda T)(\lambda T-s+1)\} \\ &= \frac{T}{2}\{\overline{P}(s)[\lambda T-2s+\frac{s^2-s}{\lambda T}+2] + \frac{s}{\lambda T}p(s;\lambda T)(\lambda T-s+1)\} \end{aligned}$$

$$\text{TWUS}(s) = \frac{T}{2}\{\overline{P}(s+1)[\lambda T-2s+\frac{s(s+1)}{\lambda T}] + p(s;\lambda T)(\lambda T-s)\}$$

$$= \frac{T}{2}\{\overline{P}(s)[\lambda T-2s+\frac{s(s+1)}{\lambda T}]-p(s;\lambda T)(\lambda T-2s+\frac{s(s+1)}{\lambda T})$$

$$+ p(s;\lambda T)(\lambda T-s)\}$$

so that

$$\text{TWUS}(s-1) - \text{TWUS}(s)$$

$$= \frac{T}{2}\{\overline{P}(s)[2 + \frac{s^2-s-s^2-s}{\lambda T}]+p(s;\lambda T)[s - \frac{s^2}{\lambda T} + \frac{s}{\lambda T} + \lambda T-2s$$

$$+ \frac{s^2+s}{\lambda T} - \lambda T + s]\}$$

$$= \frac{T}{2}[\overline{P}(s)[2 - \frac{2s}{\lambda T}] + \frac{2s}{\lambda T}\, p(s;\lambda T)]$$

$$= \overline{P}(s)[T - \frac{s}{\lambda}) + \frac{s}{\lambda}\, p(s;\lambda T) \qquad (8)$$

Substitute Equations (7) and (8) into (6). Then

$$\Delta L_i(s_i) = E_i[(T_i - \frac{s_i}{\lambda_i})\overline{P}(s_i;\lambda_i T_i) + \frac{s_i}{\lambda_i}\, p(s_i;\lambda_i T_i)]$$

$$+ \theta c_i \qquad (9)$$

The optimum solution s_i^* is the largest s_i such that

$$\Delta L_i(s_i) \geq 0$$

or equivalently,

$$\frac{E_i(Z_i(s_i-1)-Z_i(s_i)}{c_i} = \frac{E_i}{c_i}[(T_i - \frac{s_i}{\lambda_i})\overline{P}(s_i;\lambda_i T_i) + \frac{s_i}{\lambda_i} p(s_i)] \geq -\theta$$

The basic algorithm for solving this problem was explained in the previous chapter. A computer program for searching for θ is provided in Appendix B.

D. SAMPLE DATA RUNS

Consider a weapon system which consists of 10 components. Suppose that the demand for each component is Poisson distributed with parameter λ_i and the lead time is known constant T_i. Let the budget available for procurement be $19224. Table III shows the optimal allocations provided by the TWUS model.

The allocation given when the demand distribution is approximated by the normal distribution is also provided in Table III for comparison. (For comparability, the variance for the normal distribution is taken to be the same as the mean). Comparing the results, we observe that the normal case buys more of the high demands low cost items. There is a small difference in the allocation for items 8 and 9 which are more expensive than the others. For demand rates less than 10, the usefulness of the approximation is questionable.

TABLE III

OPTIMAL ALLOCATION FOR TWUS MODEL

Item	λ_i	T_i	c_i ($)	E_i	Poisson	Normal
1	10	1	10	1	17	20
2	100	1	20	1	113	120
3	15	1	80	1	19	21
4	20	1	2	1	32	36
5	50	1	5	1	65	71
6	80	1	30	1	90	96
7	20	1	1	1	35	37
8	15	1	200	1	17	15
9	75	1	100	1	77	74
10	10	1	75	1	14	16

The resulting values of the objective function for the optimal solutions are:

	Poisson Case	Normal Case
$Z(\underline{s}^*)$	0.00094	0.0015
Shadow price (θ^*)	0.00015	0.00055
Budget limit	$19224	$19224
Budget left	0	0

The objective function value for the Poisson demand case is less than the normal demand case. The main reason for the difference is due to items 8 and 9.

IV. PSEUDO AVAILABILITY MODEL

A. DESCRIPTION OF MODEL

In the previous chapter, the TWUS objective function was introduced as a means for allocating a limited budget. Operational availability is a widely stated measure of the operational readiness of military forces and weapon systems. Thus, it is appropriate to consider stockage models with an availability objective as a means of allocating limited resources.

> The most direct and meaningful measure of the influence of peace time operating stocks on readiness is weapon system (or end item) availability. We use the terms availability, end-item availability, and weapon system availability interchangeably to mean the probability that an end item, such as a tank or an aircraft, selected at random, is not waiting for a component to be repaired or shipped to it. [Ref. 6]

Many authors have attempted to determine stockage levels for components by maximizing equipment operational availability, subject to a budget constraint. See, for example, Jee [Ref. 7]. Usually, the availability for component i is defined by the ratio

$$A_i = \frac{MTBF_i}{MTBF_i + MTTR_i + MSRT_i(s_i)}$$

where:

$MTBF_i$ = the Mean Time Between Failure of item i;

$MTTR_i$ = the Mean Time To Repair for item i;

$MSRT_i(s_i)$ = the Mean Supply Response Time for item i.

If the weapon system is assumed to consist of the n components all arranged in series, then the system availability is the product of the individual item availabilities. (This assumption means that the system will fail if any of the components fails.) With this assumption, the allocation problem, stated in terms of system availability is:

$$\text{(P1)} \qquad \max_{\underline{s}} \prod_{i=1}^{n} A_i(s_i)$$

$$\text{s.t.} \sum_{i=1}^{n} c_i s_i \leq B \qquad i = 1,2,\ldots,N$$

where:

$A_i(s_i)$ = the availability of item i having s_i units in stock;

B = the budget limit;

C_i = the price for each item i,

In the expression of $A_i(s_i)$, the term $MTBF_i$ is the reciprocal of the failure rate i, $MTTR_i$ is assumed to be independent of the decision variables and the available funds and $MSRT_i(s_i)$ can be expressed in terms of $TWUS(s_i)$ as

$$MSRT_i(s_i) = \frac{1}{\lambda_i T_i} TWUS_i(s_i).$$

Thus, the main determination of availability from the point of view of the supply system is $MSRT_i(s_i)$. Many techniques for solving this model have been developed. In the next section we represent an algorithm for solving the availability model by using the marginal analysis method.

B. SOLUTION PROCEDURE

The model (P1) is not additive in the individual component availabilities but is converted into an additive function by transforming the objective function. Taking the natural log of the objective function, the model can be expressed in the following way.

$$(P2) \quad \max_{\underline{s}} \sum_{i=1}^{n} \ln A_i(s_i)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} c_i s_i \leq B \qquad i = 1,2,\ldots,N$$

Now the model (P2) is separable for all i and maximization of (P2) yields the same solution as maximization of (P1). The marginal analysis method selects an item which gives at each step the greatest increase in $\log(A_i(s_i))$ per dollar spent.

STEP 1. Start with zero units for all items.

STEP 2. Compute the increase in log availability per dollar spent as a result of purchasing one additional unit.

$$\frac{\Delta\ \ell n\ A_i(s_i)}{c_i} = \frac{1}{c_i}[\ell n\ A_i(s_i) - \ell n\ A_i(s_i)]$$

where $i = 1,2,\ldots,N$

$$\ell n\ A_i(s_i) = \ell n[\frac{MTBF_i}{MTBF_i + MTTR_i + MSRT_i(s_i)}]$$

STEP 3. Select that item i corresponding to the maximum ratio.

$$\max_{\text{all } s_i} [\frac{\Delta \ell n\ A_1(s_1)}{c_1}, \frac{\Delta \ell n\ A_2(s_2)}{c_2}, \ldots, \frac{\Delta \ell n\ A_n(s_n)}{c_n}]$$

STEP 4. Increase the number of units stocked for the item selected at step 3 by one additional unit if the unit price is less than the amount of budget remaining.

STEP 5. Update the S vector, the MSRT(s) expression and decrement the available budget. If the remaining budget is greater than the cost of the cheapest item, Go to Step 3. Otherwise, Stop.

In the following section, we will illustrate this procedure with a sample system. The computer program is provided in Appendix C.

C. A NUMERIC EXAMPLE FOR THE MSRT MODEL AND THE AVAILABILITY MODEL

In the expression for availability, the MTBF and MTTR terms are not functions of the number of spare parts. Therefore it is commonly believed that maximization of system availability is equivalent to minimization of mean supply response time. However, this is not the case, as shown below.

Suppose a weapon system consists of three components and the demands are Poisson distributed with parameters λ_1, λ_2 and λ_3, respectively. The lead time is a known constant and the components have essentiality codes E_i. The unit price and MTTR are known and the budget is limited to 20 dollars. This information is summarized in Table IV.

TABLE IV

INPUT DATA FOR EXAMPLE

ITEM	λ_i	c_i	$MTTR_i$	E_i	T_i
1	1	5	0.0274	1	1.0
2	0.1	5	0.0027	3	1.0
3	10	1	0.0054	1	1.0

To solve MSRT minimization problems, we first determine the MSRT's for all possible cases. These values are provided in Table V.

TABLE V

MSRT DATA FOR ALL FEASIBLE SOLUTIONS S_i

MSRT(s_i)	ITEM 1	ITEM 2	ITEM 3
MSRT(0)	0.9482	0.9837	0.6
MSRT(1)	0.3161	0.1967	0.5
MSRT(2)	0.0708	0.02	0.4099
MSRT(3)	0.0132	0.00227	0.3298
MSRT(4)	0.0021	0.0002	0.2596
MSRT(5)	0.0003		0.1992
MSRT(6)			0.1485
MSRT(7)			0.1072
MSRT(8)			0.0746
MSRT(9)			0.0500
MSRT(10)			0.0322
MSRT(11)			0.0199

Using the solution procedure described in the previous chapter we determine the optimal solution to be as shown in Table VI.

TABLE VI

THE ALLOCATION OF SPARE PARTS FOR MSRT MODEL

ITEM	1	2	3	
ALLOCATION	$\Delta Z_1(s_1)$	$\Delta Z_2(s_2)$	$\Delta Z_3(s_3)$	USED BUDGET ($)
(0,0,0)	0.12462	0.47216	0.1	0
(0,1,0)	0.12462	0.1040	0.1	5
(1,1,0)	0.04905	0.1040	0.1	10
(1,2,0)	0.04905	0.01265	0.1	15
(1,2,1)	0.04905	0.01265	0.09	16
(1,2,2)	0.04905	0.01265	0.08012	17
(1,2,3)	0.04905	0.01265	0.07022	18
(1,2,4)	0.04905	0.01265	0.06039	19
(1,2,5)	0.04905	0.01265	0.0507	20

The optimal solution for MSRT model is (1,2,5). Repeating the analysis for the availability objective function we obtain the results provided in Table VII from the marginal analysis procedure.

$$\begin{aligned} \Delta Z_i(s_i) &= \frac{1}{c_i}[Z_i(s_i+1) - Z_i(s_i)] \\ &= \frac{1}{c_i}[\ell n\ A_i(s_i+1) - \ell n\ A_i(s_i)] \end{aligned}$$

TABLE VII

THE ALLOCATION OF SPARE PARTS FOR AVAILABILITY MODEL

ITEM	1	2	3	
ALLOCATION	$\Delta Z_1(s_1)$	$\Delta Z_2(s_2)$	$\Delta Z_3(s_3)$	USED BUDGET
(0,0,0)	0.07712	0.1294	0.1529	0
(0,0,1)	0.07712	0.1294	0.1611	1
(0,0,2)	0.07712	0.1294	0.1689	2
(0,0,3)	0.07712	0.1294	0.1759	3
(0,0,4)	0.07712	0.1294	0.1808	4
(0,0,5)	0.07712	0.1294	0.1821	5
(0,0,6)	0.07712	0.1294	0.1777	6
(0,0,7)	0.07712	0.1294	0.1660	7
(0,0,8)	0.07712	0.1294	0.1469	8
(0,0,9)	0.07712	0.1294	0.1217	9
(0,1,9)	0.07712	0.0447	0.1217	14
(0,1,10)	0.07712	0.0447	0.0938	15
(0,1,11)	0.07712	0.0447	0.06702	16
⋮			⋮	
(0,1,15)				20

The optimal solution for the availability model is (0,1,15). Comparing the results of the two models, we see that the availability model allocates more units to the high demand lower cost items than the MSRT model.

D. SAMPLE DATA RUNS

Suppose that a weapon system consists of 10 components and the demand of each component is Poisson distributed with parameter λ_i, and lead time T_i, mean time to repair $MTTR_i$ are known constants. In order to maximize the availability of spare parts with budget constraint, we can use the modified Availability model (P2) instead of (P1). By using the computer program in Appendix C, this problem can be solved. Table VIII provides the allocations of spare parts in the Availability model when the budget is 1170 dollars.

TABLE VIII

THE ALLOCATION OF SPARE PARTS FOR THE AVAILABILITY MODEL

ITEM	λ_i	T_i	c_i ($)	E_i	$MTTR_i$	ALLOCATION	$A_i(s_i)$
1	1.0	1.0	10.0	1.0	0.0137	4.0	0.986
2	0.1	1.0	20.0	1.0	0.0274	2.0	0.997
3	3.0	1.0	100.0	1.0	0.0137	3.0	0.821
4	25.0	1.0	2.0	3.0	0.0822	37.0	0.327
5	1.0	1.0	5.0	1.0	0.0274	5.0	0.973
6	0.5	1.0	5.0	3.0	0.0027	4.0	0.999
7	10.0	1.0	1.0	1.0	0.0054	21.0	0.949
8	5.0	1.0	100.0	1.0	0.0411	3.0	0.538
9	1.0	1.0	50.0	1.0	0.0082	3.0	0.987
10	2.0	1.0	100.0	1.0	0.1370	2.0	0.697

From the table, one can see that the availability of an item is greatly influenced by the MTTR term (see item 4). The availability for that item never exceeds 0.333 even if the MSRT is zero. We also observe that the availability model tends to stock the high demand low cost items.

The objective function for the optimal solution is given by:

Total obejctive value	0.08999
Shadow price	0.000261
Budget limit	$1170
Budget left	$0.0

A comparison of the above results with the allocation given in Table VIII shows that the total availability is relatively low even though most of the items have high availabilities. Also as mentioned above, when the MTTR data for an item is large relative to the MTBF, a high availability cannot be achieved.

V. COMPARISON OF MODELS

A. ANALYSIS FOR SAME DATA

In this chapter, we continue to consider the allocation of spare parts to maximize the system performance in the different allocation models. In this thesis, we have looked at three models: the units short model, the time-weighted units short model, and the availability model. Since each model attempts to reduce stockouts as much as possible the allocations generated by the models are strongly correlated. This is especially true for the availability model and the MSRT model since availability is a function of MSRT. However, we saw earlier that the allocations from the models are not necessarily the same.

Assume that a weapon system consists of 10 items, the demands are Poisson distributed and $MTTR_i$, c_i, T_i, E_i are known constants and a budget constraint of the weapon system is $1170. The optimal allocations for the three models are shown in Table IX. As can be seen, the TWUS model is more sensitive to the lead times than are the other two models (see items 5, 6, and 7).

The units short model is more sensitive to the price of the item than are the other two models. For item 9 the units short model bought nothing, but the TWUS model and the availability model allocated 2 and 3 items respectively. All

TABLE IX

THE ALLOCATIONS OF SPARE PARTS FOR THE THREE DIFFERENT MODELS

Item	λ_i (yr)	cost ($)	Ess.	T_i (yr)	MTTR (yr)	Units Short Model	TWUS Model	Avail. Model
1	1.0	10	1	1	0.0137	3	3	4
2	0.1	10	1	1	0.0137	2	1	2
3	15.0	3	1	1	0.0137	24	20	22
4	15.0	3	3	1	0.0274	26	23	24
5	3.0	10	1	0.5	0.0274	4	3	4
6	3.0	5	3	0.5	0.0274	6	4	6
7	10.0	50	1	0.2	0.0054	0	0	3
8	10.0	50	1	1	0.0411	8	7	2
9	2.0	50	1	1	0.0137	0	2	3
10	2.0	100	4	2	0.1370	5	5	5

three models are highly affected by the essentiality code. This is illustrated by a comparison of items 9 and 10. Table X presents the corresponding values of the three objective functions.

TABLE X

THE COMPARISON OF OBJECTIVE VALUES FOR THREE MODELS

OBJ FN MODEL	UNITS SHORT	TWUS	AVAILABILITY
UNITS SHORT	0.1523	0.0458	0.0483
TWUS	0.1527	0.0353	0.0687
AVAILABILITY	0.1969	0.0775	0.0709

The above table was established by computing each objective function for the allocations determined by the three different procedures. Comparing the results of the three models, the TWUS model seems to do the best job considering all three objective functions. However, no general conclusions can be drawn about the preference of the TWUS model for other situations.

One needs to determine which objective function most closely matches a servicers' feeling about how operational readiness is affected by stockouts and delays in satisfying stockouts.

B. DISCUSSION OF SIMILARITIES

In the budget allocation problem there are many factors which affect the allocation such as demand, lead time, cost, time to repair and essentiality. The three models share similar properties. First of all, as can be seen in the above example, all models tend to stock the cheap, high demand items in favor of expensive low demands items. This is because of the models attempts to get the biggest benefit per dollar spent. Potential benefit per additional unit increases with an items demand rate. Second, items having high essentiality code are given preference, as is the intent of essentiality assignment schemes. Essentiality weighting is one way to counter the preference given the high demand low cost item observed earlier. It is frequently the case that the most critical items are low demand expensive items. Without the essentiality weighting such items would be neglected by the type of models examined in this thesis.

VI. USE OF THE MODELS FOR BUDGET DETERMINATION

A. EFFECTIVENESS VS. BUDGET

The models that we have discussed have attempted to optimize performance subject to a budget constraint. We have assumed that the budget was given. There are many ways in which budgets are determined. However, budgeting people and inventory managers alike often express the desire to have a methodology that they can use to determine the amount of money that should be provided.

In most cases the amount is determined historically by giving an amount equal to what has been provided in the past for similar systems or perhaps by giving a little more or less based on judgement or financial constraints. There is, however, a strong interest brought about by Congressional pressures to relate resources to readiness. Congress wants to know "how much money is needed to support our weapon systems at a specified level of performance." In this chapter we show how the models developed earlier in this thesis can be used in just this manner.

Specifically, we show how the models that we have developed can be modified easily to determine the minimum budget required to provide a specified level of logistics performance.

The models developed earlier can each be run for a range of budget levels producing for each given budget an allocation and a predicted overall level of performance. Figure 3 illustrates this for the case in which the performance measure is pseudo-availability. As expected, the curve shows that availability is a non-decreasing function of budget with decreasing marginal returns. This can be done also for the time-weighted units short model or any of the other models discussed in this thesis. In all cases we would obtain a similar display. Performance is a monotonic function of budget with decreasing marginal returns.

Figure 4 displays a similar result for the case in which the performance measure is MSRT. Each point on the curve represents an optimal level of performance for a given budget. For this example displayed, Figure 3, there is a little benefit to be gained by increasing the budget above $2500. However there is a dramatic increase in effectiveness obtained by increasing the budget from $1000 to $2000. This is precisely the sort of information needed to make intelligent budgeting decisions. Of course some decision maker must decide if the increase in effectiveness is worth the additional expenditure.

If a specific level of effectiveness is specified, one can graphically determine the amount of budget required by simply moving horizontally across the graph from the specified level of effectiveness until the curve is intersected and then down to the budget axis.

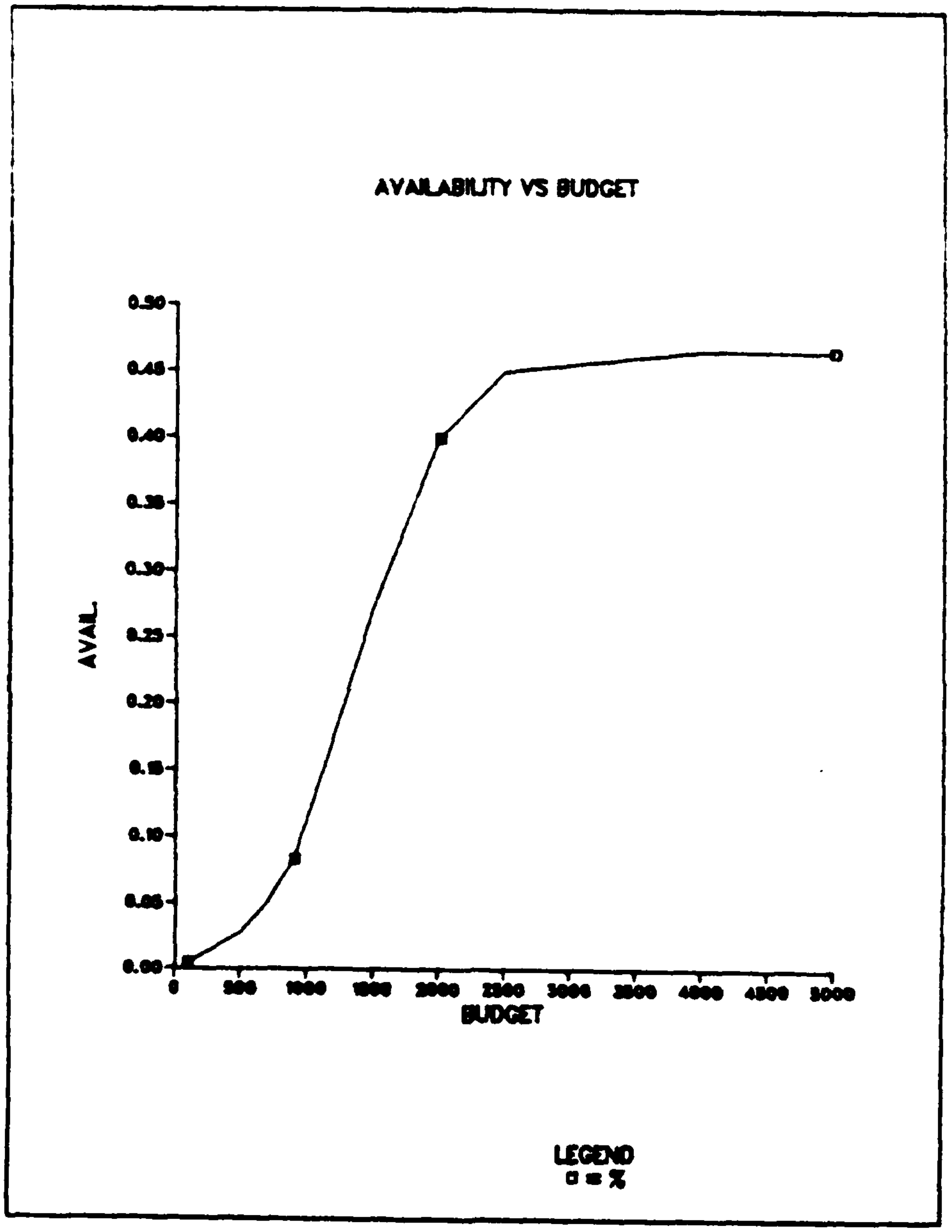


Figure 3. Total Availability Vs. Budget Curve

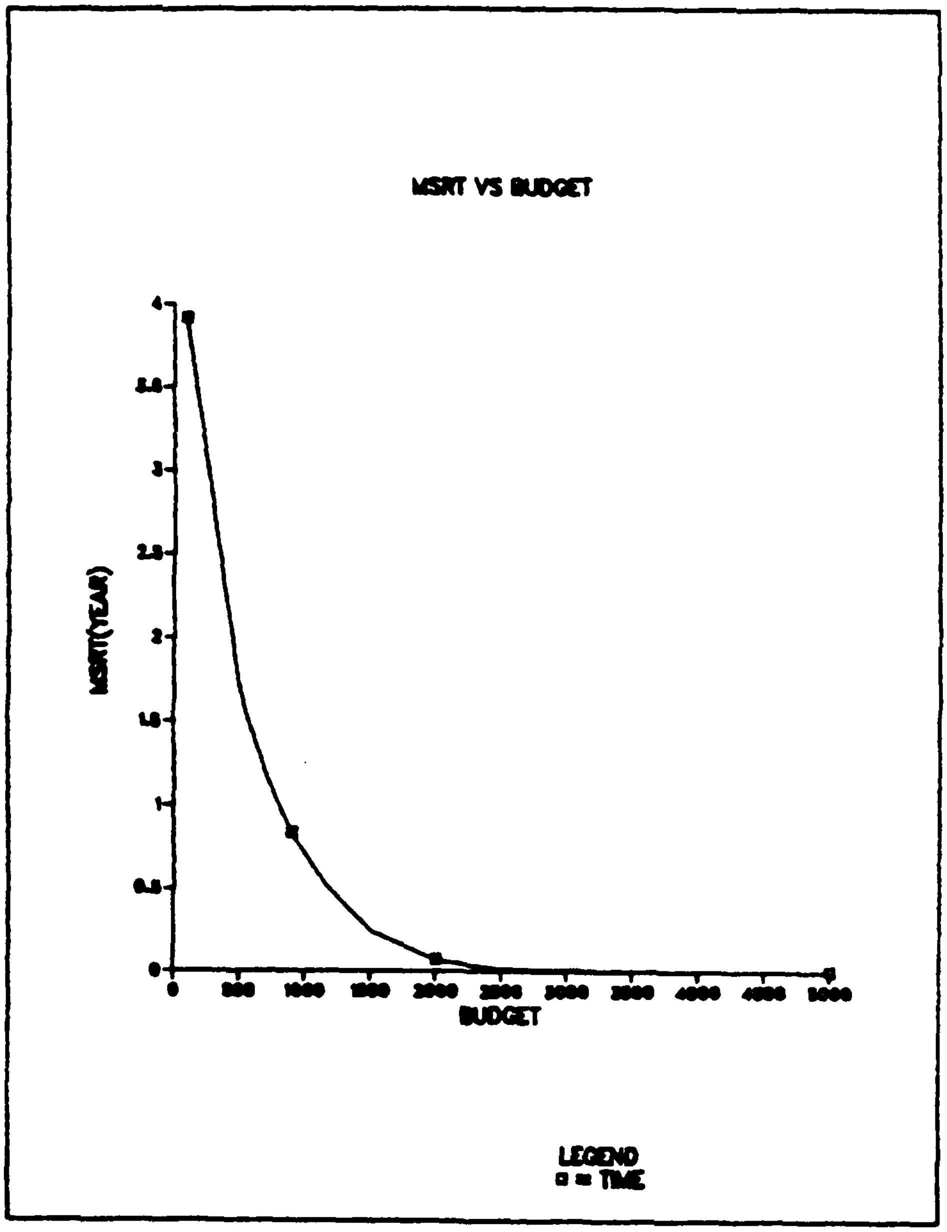


Figure 4. MSRT Vs. Budget Curve

The next section determines analytically the minimum amount budget required by solving a companion problem to the problems discussed earlier in this thesis.

B. COMPANION PROBLEMS

In the previous chapters we have concentrated on the optimization of system effectiveness with a budget constraint. For many weapon systems such as air detection radars, missiles and nuclear delivery systems, the system performance is so important that the necessary budget will be provided to attain whatever performance is deemed necessary.

For such systems it is reasonable to restate the optimization problem to determine the minimum budget required to satisfy a specified level of performance. Consider, for example, the availability optimization problem and the companion problem:

$$\text{(D1)} \qquad \min \sum_{i=1}^{n} c_i s_i$$

$$\text{s.t.} \quad \prod_{i=1}^{n} A_i \geq L \qquad i = 1,2,\ldots,n$$

where L is the minimum performance level for a weapon system.

For the MSRT model case the corresponding problem is:

$$\text{(D2)} \qquad \min \sum c_i s_i$$

$$\text{s.t.} \quad \sum_{i=1}^{n} \text{MSRT}_i(s_i) \leq R \qquad i = 1,2,\ldots,n$$

where R is the maximum allowable cumulative supply response time for the weapon system.

Problems (D1) and (D2) can be solved using the same methods explained in Chapters III and IV. In the above models the budget is determined so that the system requirement for availability or main supply response time can be achieved.

For problem (D2) the total cost is minimized when

$$\sum_{i=1}^{n} MSRT_i(s_i) = R .$$

Sometimes a minimum allowable supply response time is required for each item. In such a case multiple constraints could be specified. This is illustrated below:

$$\text{(D3)} \qquad \min \sum_{i=1}^{n} c_i s_i$$

$$\begin{aligned} MSRT_1(s_1) &\leq R_1 \\ MSRT_2(s_2) &\leq R_2 \\ &\vdots \\ MSRT_n(s_n) &\leq R_n \end{aligned}$$

where R_i is the maximum allowable supply response time, $i = 1,2,\ldots,N$.

To solve (D3), find the smallest s_i such that

$$\begin{aligned} MSRT_1(s_1) &= R_1 \\ MSRT_2(s_2) &= R_2 \\ &\vdots \\ MSRT_n(s_n) &= R_n \end{aligned}$$

This problem is solved easily using the same procedures which we discussed.

So far we have discussed many different ways to apply the theoretical models to practical use of models for budget determination decisions. There is no unique method which gives us an optimal result. So the user of these models should choose one of possible methods so as to maximize the system performance or minimize the total cost.

VII. CONCLUSIONS

It is concluded that the various measures of effectiveness can be used in the budget constrained multi-item inventory system with stochastic demands. We have examined some of the more reasonable measures like minimization of units short, minimization of time-weighted units short and maximization of system availability. We have also looked at models which incorporate essentiality weights into each of the models.

In order to solve budget allocation stockage problems a feasible, efficient method of effecting line item inventory control is available using an adaptation of Everett's Generalized Lagrangian Multiplier method. Further, the use of a G.L.M. procedure provides valuable information for system managers as to relative effectiveness of additional procurement funds, versus additional transition processing capability. The final value of Lagrangian multipliers can be interpreted as the amount of improvement of the objective function per unit dollar spent.

The models discussed in this thesis are all more likely to stock cheap, high demand items than expensive, low demand items. Such is the nature of budget constrained optimization problems. If a system manager wishes to maintain enough stock for an item having low demand, high cost, his only alternative in our models is to assign a high essentiality code for the item. The essentiality code has the effect of reducing the ratio C/E as opposed to C. In the solution procedure for each model, the assigned essentiality code directly affects the allocation for the item.

We have shown how the models can be used as a tool to determine the amount of budget. A simple graphical procedure allows a decision maker to determine the minimum budget required to search a specified level of performance. The optimization model is run several times to generate a plot of performance vs. budget. Each point of the curve represents the effectiveness for the optimal allocation of a given budget. A manager can, first, determine an appropriate system performance level and read from the curve the budget required to achieve the effectiveness.

Further analysis to improve these models may be possible. For instance, it would be useful to have an automatic search algorithm for the Lagrangian multipliers for a multiple constrained problem. It may also be possible to relax the assumptions for a constant lead time or a constant mean time to repair. These single period inventory may expand to time-dependent multi-item, multi-echelon, multi-indenture inventory systems.

APPENDIX A

COMPUTER PROGRAM FOR INTERACTIVE SEARCH

```
      DIMENSICN KDEM(5,100),TMLD(100),IDUES(100),IONHD(100),
     1PRICE(1C0),SZRQN(100),INVNEW(100),PI(100),ZWT(100),TMSVD(
     2100),QUAL(1C0),KFREQ(5,100)
      COMMON CEMFCT(100), DEVFCT(100),IPOS(100)
      LOGICAL QUAL
      N = 100
      NORDER = 0
      CSTBUD = 0.
      DATA  IC1/'YES'/,IC2/ 'NO'/
      FLAG = 0.
      WRITE(6,137)
  137 FORMAT('  READ IN INITIAL VALUES.')
      READ(3,10) N,(ZWT(I),I=1,4),SHADB,SHADW,TMISS,TMNIS,J,R,WRKLD,
     1BUDGET
      DO 731 L=1,N
  731 READ(4,20) (KDEM(I,L),I=1,5),DEMFCT(L),PRICE(L),(KFREQ(I,L),I=1,5)
     1,DEVFCT(L),TMLD(L),IDUES(L),IONHD(L)
10    FORMAT(I4, EF5.2,3F6.2,F10.2)
20    FORMAT(5I3,1X,2F5.2,1X,5I2,F6.2,5X,F2.1,7X,I2,I3)
C1    FORMAT(5I3,1X,2F6.2,1X,5I2,F7.2,5X,F3.1,7X,I2,I3)
      DO 22 J=1,N
      DEM= 0.
      DNUM=0.
      DO 23 K=1,5
      DDEM=KDEM(K,J)
      DFREQ=KFREQ(K,J)
      DNUM=DNUM+DDEM
   23 DEM=DEM+DFREQ
      SZRQN(J)=1.
      IF(DEM.LE.0.) GO TO 967
      SZRQN(J)=DNUM/DEM
      IF(SZRQN(J).LE.0.) SZRQN(J)=1.
  967 IPOS(J)= IDUES(J)+IONHD(J)
      TMSVD(J)=AMAX1(TMLD(J)+TMNIS-TMISS,0.)
   22 PI(J)=ZWT(1)*PRICE(J)+ZWT(2)*U+ZWT(3)*R/SZRQN(J)+ZWT(4)*TMSVD(J)
      ZSTARM = 0.
       ZS = 0.
      ZR = 0.
      ZU = 0.
      ZT = 0.
      ZMAXS = 0.
      ZSTAR = 0.
      ZMAXT = 0.
      ZMAXU = 0.
      FLAG=0.
      ZMAXR = 0.
  313 DO 100  J=1,N
      QTY1 = DEMFCT(J)
```

```
      IQTY2 = IPOS(J)                                                   INV00490
      ZMAXS = ZMAXS + PRICE(J)*QTY1                                     INV00500
      VALDSC=DSCVAL(IQTY2,J)                                            INV00510
      ZS=VALDSC*PRICE(J)+ZS                                             INV00520
      ZU=ZU+U*VALDSC                                                    INV00530
      ZMAXU=ZMAXU+U*QTY1                                                INV00540
      ZMAXR=ZMAXR + R*QTY1/SZRQN(J)                                     INV00550
      ZR = ZR + R*VALDSC/SZRQN(J)                                       INV00560
      ZMAXT=ZMAXT + QTY1*TMSVD(J)                                       INV00570
      ZT = ZT + VALDSC*TMSVD(J)                                         INV00580
      ZSTAR = ZSTAR + VALDSC*PI(J)                                      INV00590
  100 ZSTARM = ZSTARM +(PI(J))*QTY1                                     INV00600
      WRITE(6,40)     (ZWT(I),I=1,4),ZSTAR,ZSTARM,ZS,ZU,ZR,ZT,ZMAXS,ZMAXINV00610
     1U,ZMAXR,ZMAXT                                                     INV00620
   40 FORMAT(7X,'VALUE OF OBJECTIVE FUNCTION EVALUATED AT LEGACY FOR CHOINV00630
     1ICE OF WEIGHTS',//,14X,F6.4,2X,'FOR SALES',/,14X,F6.4,2X,'FOR UNITINV00640
     2S ISSUED',/,14X,F6.4,2X,'FOR REQNS ISSUED',/,14X,F6.4,2X,'FOR CUSTINV00650
     3OMER TIME SAVED',//,6X,'IS',2X,F10.2,'.',//,6X,'IF INFINITE RESOURINV00660
     4CES WERE AVAILABLE THIS VALUE MIGHT BE INCREASED TO',F11.2,'.',//,INV00670
     56X,'VALUES OF LEGACY AND GREATEST POSSIBLE VALUE OF THE PURE OBJECINV00680
     6TIVE FUNCTIONS ARE',//,10X,'SALES',16X,'UNITS',16X,'REQNS',15X,'TIINV00690
     7ME',/4F20.2,/4F20.2)                                              INV00700
      ZSTAR = 0.                                                        INV00710
      ZS = 0.                                                           INV00720
      ZU = 0.                                                           INV00730
      ZT = 0.                                                           INV00740
      ZR = 0.                                                           INV00750
  312 DO 200 J=1,N                                                      INV00760
      ITRIAL = IPOS(J) + 1                                              INV00770
      RUSS= SHADB * PRICE(J)/ PI(J)                                     INV00780
      D= DEMFCT(J)                                                      INV00790
      V = DEVFCT(J)                                                     INV00800
      BILL = ANDTRY(RUSS)                                               INV00810
      DEB = BILL*DEVFCT(J) + DEMFCT(J)                                  INV00820
      APOS = IPOS(J)                                                    INV00830
      IF(DEB.LT.APOS) DEB=APOS                                          INV00840
      MIKE = DEB + .5                                                   INV00850
      ITRIAL = MIKE                                                     INV00860
  201 VALDSC = DSCVAL(ITRIAL,J)                                         INV00870
      IBUYQ = ITRIAL - IPOS(J)                                          INV00880
      BUYQ = IBUYQ                                                      INV00890
      TSTBY = PI(J)* VALDSC - (SHADB * PRICE(J) * BUYQ) - SHADW         INV00900
      ICE = IPOS(J)                                                     INV00910
      ALICE = DSCVAL(ICE,J)                                             INV00920
      TSTNBY = PI(J)* ALICE                                             INV00930
      IF(TSTBY.GT.TSTNBY) GO TO 204                                     INV00940
      WINNER = ALICE                                                    INV00950
      INVNEW(J) = IPOS(J)                                               INV00960
```

```
  199 ZSTAR= ZSTAR + PI(J)*WINNER                                           INV00970
      ZS = ZS + WINNER*PRICE(J)                                             INV00980
      ZU = ZU + WINNER * U                                                  INV00990
      ZR = ZR + WINNER * R / SZRON(J)                                       INV01000
      ZT = ZT + WINNER * TMSVD(J)                                           INV01010
      GO TO 200                                                             INV01020
  204 WINNER = VALDSC                                                       INV01030
      INVNEW(J)=ITRIAL                                                      INV01040
      NORDER = NORDER + 1                                                   INV01050
      CSTBUD = CSTBUD +  BUYQ * PRICE(J)                                    INV01060
      GO TO 199                                                             INV01070
  200 CONTINUE                                                              INV01080
      UBOUND = ZSTAR - (CSTBUD - BUDGET)*SHADB -(NORDER- WRKLD)*SHADW        INV01090
      WRITE(6,300) BUDGET, WRKLD                                            INV01100
  300 FORMAT(10X,'BUDGET =',F10.2,10X,'WORKLOAD LIMIT IS',F5.0)             INV01110
  301 WRITE(6,302) CSTBUD, NORDER, UBOUND                                   INV01120
  302 FORMAT(5X,'COST IS',F10.2,5X,'NORDER =',I4,5X,'UBOUND =',F10.2)       INV01130
      WRITE(6,329) ZSTAR, ZS, ZU, ZR, ZT                                    INV01140
 329  FORMAT(5X,' ZSTAR = ', F8.2,'ZS = ',F8.2,'    ZU =',F8.2,'   ZR     INV01150
     1=',F8.2,'ZT=',F8.2)                                                   INV01160
      WRITE(6,303)                                                          INV01170
  303 FORMAT(3X,'   TRY AGAIN?')                                            INV01180
      READ(5,304) IANS                                                      INV01190
  304 FORMAT(A3)                                                            INV01200
      IF(IANS.EQ.IC2) GO TO 400                                             INV01210
      ZSTAR = 0.                                                            INV01220
      ZU = 0.                                                               INV01230
      CSTBUD = 0.                                                           INV01240
      ZS = 0.                                                               INV01250
      NORDER = 0.                                                           INV01260
      ZR = 0.                                                               INV01270
      ZT = 0.                                                               INV01280
      ZMAXS = 0.                                                            INV01290
      ZMAXT = 0.                                                            INV01300
      ZMAXR = 0.                                                            INV01310
      ZMAXU = 0.                                                            INV01320
      FLAG = 0.                                                             INV01330
      ZSTARM = 0.                                                           INV01340
      WRITE(6,305)                                                          INV01350
  305 FORMAT(3X,'   SAME WEIGHTS?')                                         INV01360
      READ(5,304) IANS                                                      INV01370
      IF(IANS.EQ.IC1) GO TO 350                                             INV01380
      FLAG=1.0                                                              INV01390
  310 WRITE(6,307)                                                          INV01400
  307 FORMAT(3X,'  INPUT ZS,ZU,ZR,ZT VIA 4F5.3')                            INV01410
      READ(5,306) ZWT(1),ZWT(2),ZWT(3),ZWT(4)                               INV01420
  306 FORMAT(4F5.3)                                                         INV01430
      WRITE(6,308)(ZWT(I),I=1,4)                                            INV01440
```

```
  308 FORMAT(3X,' WEIGHTS ARE',3X,4(F6.3,2X))                           INV01450
      WRITE(6,309)                                                      INV01460
  309 FORMAT(3X,'  CORRECT?')                                           INV01470
      READ(5,304) IANS                                                  INV01480
      IF(IANS.EQ.IC2) GO TO 310                                         INV01490
      DO  800 J=1,N                                                     INV01500
 800  PI(J)=ZWT(1)*PRICE(J)+ZWT(2)*U+ZWT(3)*R/SZRQN(J)+ZWT(4)*TMSVD(J)  INV01510
 350  WRITE(6,363) SHADB,SHADW                                          INV01520
  363 FORMAT(3X,'SHADB =',F10.3,5X,'SHADW =',F10.3,/,'   SAME MULTIPLIERINV01530
     1S?')                                                              INV01540
      READ(5,304) IANS                                                  INV01550
      IF(IANS.EQ.IC1) GO TO 312                                         INV01560
      WRITE(6,315)                                                      INV01570
  315 FORMAT(3X,' INPUT SHADB AND SHADW VIA 2F10.3')                    INV01580
  316 FORMAT(2F10.3)                                                    INV01590
      READ(5,316) SHADB,SHADW                                           INV01600
      GO TO 312                                                         INV01610
  400 WRITE(6,401)                                                      INV01620
  401 FORMAT(3X,' LOOK AT LINE ITEMS?')                                 INV01630
      READ(5,304) IANS                                                  INV01640
      IF(IANS.EQ.IC2) GO TO 500                                         INV01650
      WRITE(6,981)                                                      INV01660
 981  FORMAT('   HOW MANY?',/,' XXX')                                   INV01670
      READ(5,982)NUMI                                                   INV01680
 982  FORMAT(I3)                                                        INV01690
      WRITE(6,403)                                                      INV01700
 403  FORMAT('   ITEM  LEGACY  NEW QTY')                                INV01710
      DO 402 J=1,NUMI                                                   INV01720
      IF(IPOS(J).EQ.INVNEW(J) ) GO TO 402                               INV01730
      WRITE(6,404) J, IPOS(J), INVNEW(J)                                INV01740
 404  FORMAT(2X,I4,3X,I5,3X,I5)                                         INV01750
 402  CONTINUE                                                          INV01760
      WRITE(6,403)                                                      INV01770
      DO 405 J=1,N                                                      INV01780
      WRITE(6,404) J, IPOS(J),INVNEW(J)                                 INV01790
 405  CONTINUE                                                          INV01800
  500 STOP                                                              INV01810
      END                                                               INV01820
      FUNCTION DSCVAL(I,J)                                              INV01830
      COMMON DEMFCT(100), DEVFCT(100), IPOS(100)                        INV01840
      K=1                                                               INV01850
      DSCVAL = 0.                                                       INV01860
      IF(I.LE.J) GO TO 182                                              INV01870
      D = DEMFCT(J)                                                     INV01880
      V = DEVFCT(J)                                                     INV01890
180   DSCVAL = DSCVAL + DLTAZJ(K,D,V)                                   INV01900
      IF(K.GE.I) GO TO 182                                              INV01910
      K=K+1                                                             INV01920
```

```
      GO TO 180                                                         INV01930
  182 RETURN                                                            INV01940
      END                                                               INV01950
      FUNCTION DLTAZJ(I,D,V)                                            INV01960
      COMMON DEMFCT(100),DEVFCT(100),IPOS(100)                          INV01970
      SIGMA = 1.25*V                                                    INV01980
      AMIN = D* .5                                                      INV01990
      IF(SIGMA.LT.AMIN) SIGMA = AMIN                                    INV02000
      IF(SIGMA.LT.1.) SIGMA = 1.                                        INV02010
      AI = I                                                            INV02020
      X =(AI - D)/SIGMA                                                 INV02030
      AX = ABS(X)                                                       INV02040
      T = 1.0/(1.0 + .2316419*AX)                                       INV02050
      D = 0.3989423 * EXP(-X*X/2.0)                                     INV02060
      DLTAZJ = 1.0 - D*T*((((1.330274*T - 1.821256)*T + 1.781478)*T     INV02070
     1 - 0.3565638)*T + 0.3193815)                                      INV02080
      DLTAZJ = 1.0 - DLTAZJ                                             INV02090
      IF(X) 1,2,2                                                       INV02100
    1 DLTAZJ= 1.0 - DLTAZJ                                              INV02110
    2 RETURN                                                            INV02120
      END                                                               INV02130
      FUNCTION ANDTRY(APROB)                                            INV02140
      IF(APROB.GT..995)APROB=.995                                       INV02150
      IF(APROB.LT..005)APROB=.005                                       INV02160
    7 D = APROB                                                         INV02170
      IF(D - 0.5) 9,9,8                                                 INV02180
    8 D = 1.0 - D                                                       INV02190
    9 T2 = ALOG(1.0 /(D*D))                                             INV02200
      T = SQRT (T2)                                                     INV02210
      X = T - (2.515517+0.802853*T+0.010328*T2)/(1.0+1.432788*          INV02220
     1T+0.189269*T2 + 0.001308*T*T2)                                    INV02230
      IF(APROB - .5) 10,10,11                                           INV02240
   10 X = -X                                                            INV02250
   11 D = 0.3989423*EXP(-X*X/2.0)                                       INV02260
      X = -X                                                            INV02270
      ANDTRY = X                                                        INV02280
   12 RETURN                                                            INV02290
      END                                                               INV02300
```

APPENDIX B

COMPUTER PROGRAM FOR AUTOMATING SEARCH

```
C      VARIABLE DEFINITION
C
C      TMLD      : LEAD TIME  (100)
C      TMEAN     : MEAN DEMAND (100)
C      TMLDM     : LEAD TIME DEMAND (100)
C      PRICE     : PRICE(100)
C      DELZ      : (Z(I)-Z(0))/PRICE(I)  (100)
C      S         : THE NUMBER OF ITEM REQUIRED (100)
C      PBAR      : P(S) COMPLEMENTRY CUMULATIVE DISTRIBUTION  (100)
C      ZSTAR     : THE VALUE OF OBJECTIVE FUNTION
C      CSTBUD    : USED BUDGET
C      BUDGET    : GIVEN BUDGET
C      UB        : UPPER BOUND
C      CUB       : CURRENT UPPER BOUND
C      THETA     : LAGLANGIAN MULTIPLIER
C      MTBF      : MEAN TIME BETWEEN FAILURE
C      MSRT      : MEAN SUPPLY RESPONSE TIME
C      MTTR      : MEAN TIME TO REPAIR
C      ESS       : ESSENTIALITY CODE
C
C
C    ====================================================================
       DIMENSICN DELZ(100),S(100),PBAR(100),MTTR(100),ESS(100),AAV(100),
      /TMLD(100),TMEAN(100),TMLDM(100),PRICE(100),MTBF(100),MSRT(100)
       REAL*4 DELZ,S,PBAR,ZSTAR,CSTBUD,UB,CUB,THETA,THETAM,THETAL,
      1Z1,SS,SNN,SN,SSS,OLD,X1,X2,X3,X4,X5,X6,X7,X8,CCDF,PO,ZS1,ZS2,
      1Z2,Z3,Z0,MTBF,MTTR,ESS,MSRT,AAV
       INTEGER*4 I,J,K,N,KK
C
C
C    ====================================================================
C      INPUT DATA
C    ====================================================================
       N=10
       BUDGET=19224.0
       DO 11 I=1,N
          READ(5,12) TMEAN(I),TMLD(I),PRICE(I),MTTR(I),ESS(I)
          MTBF(I)=1.0/TMEAN(I)
   11  CONTINUE
   12  FORMAT(5F10.5)
C    ===================================================================
C      FIND Z(1)-Z(0) ARRAY
C    ===================================================================
       DO 13 I=1,N
        TMLDM(I)=TMEAN(I)*TMLD(I)
       CALL SUB (TMLDM(I),1.0,CCDF,CDF,PO)
          X1=CCDF
          X3=PO
```

```
      CALL SUB (TMLDM(I),0.0,CCDF,CDF,PO)                               TH200490
         X2=CCDF                                                        TH200500
      Z1=(TMLC(I)-(1.0/TMLDM(I)))*X2                                    TH200510
      Z1=Z1+(1.0/TMLDM(I))*X3                                           TH200520
      DELZ(I)=Z1*ESS(I)/PRICE(I)                                        TH200530
   13 CONTINUE                                                          TH200540
C    ===============================================================    TH200550
C                                                                       TH200560
C     FIND UPPER VALUE OF THETA                                         TH200570
C                                                                       TH200580
C     ==============================================================    TH200590
      THETAM=DELZ(1)                                                    TH200600
      DO 14 K=2,N                                                       TH200610
         IF(THETAM.GT.DELZ(K)) GO TO 14                                 TH200620
            THETAM=DELZ(K)                                              TH200630
   14 CONTINUE                                                          TH200640
C     ================================================================  TH200650
C                                                                       TH200660
      THETAL=0.0                                                        TH200670
      THETA=THETAM/2.0                                                  TH200680
      OLD=20.                                                           TH200690
      WRITE(6,906) THETAM                                               TH200700
  906 FORMAT(1X,' THE MAXIMUM VALUE OF THETA = ',F10.2)                 TH200710
C                                                                       TH200720
C    ================================================================   TH200730
C     FIND S(1),S(2), , , , S(N)                                        TH200740
C    ================================================================   TH200750
  800 CONTINUE                                                          TH200760
      SUM=0.0                                                           TH200770
      DO 901 J=1,N                                                      TH200780
         SSS=1.0                                                        TH200790
  903    S1=SSS-1.0                                                     TH200800
         S2=SSS+1.0                                                     TH200810
         S3=SSS+2.0                                                     TH200820
      CALL SUB (TMLDM(J),S1,CCDF,CDF,PO)                                TH200830
         X1=CCDF                                                        TH200840
      CALL SUB (TMLDM(J),SSS,CCDF,CDF,PO)                               TH200850
         X2=CCDF                                                        TH200860
         X5=PO                                                          TH200870
          ZS1=(TMLC(J)-(SSS/TMLDM(J)))*X1                               TH200880
          ZS1=(ZS1+(SSS/TMLDM(J))*X5)/PRICE(J)                          TH200890
          ZS1=ESS(J)*ZS1                                                TH200900
      IF(X2.LE.0.0001) GO TO 902                                        TH200910
      IF( ZS1.LE.(THETA)) GO TO 902                                     TH200920
      SSS=SSS+1.0                                                       TH200930
       GO TO 903                                                        TH200940
  902  S(J)=SSS-1.0                                                     TH200950
       SUM=SUM+S(J)*PRICE(J)                                            TH200960
```

```
  901 CONTINUE                                                          TH200970
      WRITE(6,905) (S(I),I=1,10),THETA,SUM                              TH200980
  905 FORMAT(1X,' S=',10F5.1,' THETA= ',F9.5,'CST= ',F9.1)              TH200990
C                                                                       TH201000
C     ================================================================  TH201010
C         COMPUTE Z(S) BY USING S(1),S(2),,,,,,S(N)                     TH201020
C     ================================================================  TH201030
      UB=99999.0                                                        TH201040
      ZSTAR=0.0                                                         TH201050
      ZSTARL=0.0                                                        TH201060
      CSTBUD=0.0                                                        TH201070
      SLT=0.0                                                           TH201080
      DO 15 J=1,N                                                       TH201090
         SLT=SLT+TMLDM(J)                                               TH201100
C     CALL SUB (TMEAN(J),S(J)-1.0,CCDF,CDF,PO)                          TH201110
C        X5=CCDF                                                        TH201120
      CALL SUB (TMLDM(J),S(J),CCDF,CDF,PO)                              TH201130
         X6=CCDF                                                        TH201140
      CALL SUB (TMLDM(J),S(J)+1.0,CCDF,CDF,PO)                          TH201150
         X7=CCDF                                                        TH201160
      CALL SUB (TMLDM(J),S(J)+2.0,CCDF,CDF,PO)                          TH201170
         X8=CCDF                                                        TH201180
         PBAR(J)=X6                                                     TH201190
         Z1=TMLDM(J)*X6                                                 TH201200
         Z2=2.0*S(J)*X7                                                 TH201210
         Z3=S(J)*(S(J)+1.0)*X8/TMLDM(J)                                 TH201220
         Z0=0.5*TMLD(J)*(Z1-Z2+Z3)                                      TH201230
         MSRT(J)=ZC/TMLDM(J)                                            TH201240
         AAV(J)=MTBF(J)/(MTBF(J)+MTTR(J)+MSRT(J))                       TH201250
         ZSTAR=ZSTAR+Z0*ESS(J)                                          TH201260
          CSTBUD=CSTBUD+S(J)*PRICE(J)                                   TH201270
   15 CONTINUE                                                          TH201280
      ZSTAR=ZSTAR/SLT                                                   TH201290
      CUB=ZSTAR-THETA*(CSTBUD-BUDGET)                                   TH201300
C     ================================================================  TH201310
C     SEARCH OPTIMAL THETA                                              TH201320
C     ================================================================  TH201330
C     IF ((CUB-ZSTAR)/CUB.LE.0.00001.OR.(CSTBUD-BUDGET).LT.1.0) GO TO   TH201340
C    / 99                                                               TH201350
      IF (ABS(THETA-OLD).LE.0.00001.AND.(CSTBUD.LE.BUDGET)) GO TO 99    TH201360
C     IF(ABS((CUB-UB)/CUB).LE. 0.001) GO TO 99                          TH201370
C     IF(((BUDGET-CSTBUD).LE.1.).AND.(CSTBUD.GT.BUDGET)) GO TO 99       TH201380
      IF(CUB.LE.UB) UB=CUB                                              TH201390
      IF(CSTBUD.GT.BUDGET) GO TO 21                                     TH201400
         THETAM=THETA                                                   TH201410
         THETA=(THETAL+THETAM)/2.0                                      TH201420
         OLD=THETAM                                                     TH201430
         GO TO 800                                                      TH201440
```

```
21        THETAL=THETA                                                      TH201450
          OLD=THETA                                                         TH201460
          THETA=(THETAL+THETAM)/2.0                                         TH201470
          GO TO 80C                                                         TH201480
C                                                                           TH201490
C                                                                           TH201500
C     ===============================================================      TH201510
C      OUTPUT PRINT                                                         TH201520
C     ===============================================================      TH201530
C                                                                           TH201540
 99      WRITE(6,100)                                                       TH201550
         WRITE(6,101) THETA, ZSTAR,CUB,BUDGET,CSTBUD                        TH201560
         WRITE(6,102)                                                       TH201570
         DO 200 I=1,N                                                       TH201580
          WRITE(6,103) I,PRICE(I),TMEAN(I),S(I),PBAR(I),AAV(I),MSRT(I)      TH201590
  200 CONTINUE                                                              TH201600
  100 FORMAT(1H1,///20X,' ******* THE RESULT OF SEARCH *******'///)         TH201610
  101 FORMAT(10X,' THETA                =  ',F10.4/                         TH201620
     /          10X,' ZSTAR                =  ',F10.4/                      TH201630
     /          10X,' CURRENT UPPER BOUND =  ',F10.3/                       TH201640
     /          10X,' BUDGET LIMIT         =  ',F10.3/                      TH201650
     /          10X,' USED BUDGET          =  ',F10.3//)                    TH201660
  102 FORMAT(6X,' NO. ',5X,'  PRICE   ',5X,'M DEMAND ',5X,'  BUY S ',5X,    TH201670
     /' PBAR (S) ',' AVIL (S) ',5X,' MSRT(S)  '//)                          TH201680
  103 FORMAT(6X,I4,7X,F9.2,5X,F9.2,5X,F8.2,5X,F8.3,5X,F8.3,5X,F9.5)         TH201690
         STOP                                                               TH201700
         END                                                                TH201710
         SUBROUTINE SUB(TT,SS,CCDF,CDF,PO)                                  TH201720
         REAL*4 TT,SN,SS,PO,CDF,CCDF                                        TH201730
         INTEGER*4 I                                                        TH201740
         SN=0.0                                                             TH201750
         PO=EXP(-TT)                                                        TH201760
         CDF=PO                                                             TH201770
         CCDF=1.0-CDF                                                       TH201780
C        WRITE(6,10) TT,SN,PO,CDF,CCDF                                      TH201790
C   10 FORMAT(1X,' MEAN,SN,PO,CDF,CCDF ',5F9.5)                             TH201800
C        IF(CDF.GE.0.005) CCDF=0.005                                        TH201810
         IF(SS.LE.0.C) GO TO 26                                             TH201820
         SN=SN+1.0                                                          TH201830
   22    PO=PO*TT/SN                                                        TH201840
         CDF=CDF+PO                                                         TH201850
         CCDF=1.0-CDF                                                       TH201860
C        WRITE(6,11) TT,SN,PO,CDF,CCDF                                      TH201870
C   11 FORMAT(1X,' MEAN,SN,PO,CDF,CCDF, '5F9.5)                             TH201880
         IF(CDF.GT.0.9999) GO TO 26                                         TH201890
         IF(SN.GE.SS) GO TO 26                                              TH201900
         SN=SN+1.0                                                          TH201910
         GO TO 22                                                           TH201920
```

```
   26 CONTINUE                                                          TH201930
      RETURN                                                            TH201940
      END                                                               TH201950
```

APPENDIX C

COMPUTER PROGRAM FOR MARGINAL ANALYSIS

```
C
C ====    TEST FOR THE THREE INVENTORY MODELS ======
C
C ==================
C  ** OBJECTIVES **
C ==================
C
C                1. MINIMIZE UNITS SHORT MODEL
C                2. MINIMIZE TWUS        MODEL
C                3. MAXIMIZE PSEUDO AVAILABILITY MODEL
C
C  ** CONSTRAINT **
C                -- 2UDGET ----
C
C  ** METHOD **
C                -- MARGINAL ANALYSIS TECHNIQE --
C
C ** FUNCTIONS ***
C                1.PCCDF:  FIND C.C.D.F FOR POISSON DISTRIBUTION
C                2.PDEN :  FIND PROB(X=N) POSSION DISTRIBJTION
C                3.TWS  :  FIND THE OBJECT FUNCTION FOR TWUS MODEL
C                4.UNITS:  FIND THE OBJECT FUNCTION
C                5.DELU :  FIND THE (Z(S-1)-Z(S))*E/C FOR MODEL 1
C                6.DELTWS: FIND THE (Z(S-1)-Z(S))*E/C FOR MODEL 2
C                7.DELAV : FIND THE (Z(S)-Z(S-1))*E/C FOR MODEL 3
C
C ** VARIABLE DEFINITION
C                LAM(I)  : POISSON ANNUAL DEMAND
C                T(I)    : LEAD TIME
C                C(I)    : PRICE FOR AN ITEM
C                ESS(I)  : ESSENTIALITY CODE
C                MTTR(I) : MEAN TIME TO REPAIR
C                MSRT(I) : MEAN SUPPLY RESPONSE TIME
C                AAV(I)  : AVAILABILITY
C                US (I)  : UNIT SHORT
C                S(I)    : STOCK VECTOR
C                RATIO(I): Z(S)-Z(S+1)
C
C
      REAL LAM(100),T(100),C(100),ESS(100),MTTR(100),MSRT(100),
     /AAV(100),US(100),S(100),RATIO(100),
     /B,BR,RR,OBJ1,OBJ2,OBT3,LTM,MTBF,X
      INTEGER N,I,K J,KK
C
C     INPUT DATA
C
      WRITE(6,9)
      N=10
```

```
      DO 1 I=1,N                                                        UM000490
         READ(5,12) LAM(I),T(I),C(I),MTTR(I),ESS(I)                     UM000500
         WRITE(6,13) I,LAM(I),T(I),C(I),ESS(I),MTTR(I)                  UM000510
    1 CONTINUE                                                          UM000520
    9 FORMAT(1H1,10X,'    NO',5X,'LAMDA',5X,'LEAD T',5X,'PRICE ',5X,    UM000530
     /' ESS.',5X,' MTTR   '//)                                          UM000540
   12 FORMAT(5F10.4)                                                    UM000550
   13 FORMAT(10X,I5,5X,F5.1,5X,F5.1,5X,F6.1,5X,F5.1,5X,F10.4)           UM000560
      B=1170.                                                           UM000570
C-----------------------------------------------------------------------UM000580
C                                                                       UM000590
C     INITIALIZE                                                        UM000600
C                                                                       UM000610
      DO 15 I=1,N                                                       UM000620
         S(I)=0.0                                                       UM000630
C        RATIC(I)=DELU(LAM(I),T(I),1.0,C(I),ESS(I))                     UM000640
C        RATIO(I)=DELTWS(LAM(I),T(I),1.0,C(I),ESS(I))                   UM000650
         RATIO(I)=DELAV(LAM(I),T(I),1.0,C(I),ESS(I),MTTR(I))            UM000660
   15 CONTINUE                                                          UM000670
C                                                                       UM000680
C-----------------------------------------------------------------------UM000690
C                                                                       UM000700
C    ALLOCATE ITEMS WITH GIVEN BUDGET                                   UM000710
C                                                                       UM000720
C                                                                       UM000730
      BR=B                                                              UM000740
   19 KK=0                                                              UM000750
      RR=-1.                                                            UM000760
      DO 16 K=1,N                                                       UM000770
         IF(C(K).GT.BR) GO TO 16                                        UM000780
         IF(RATIO(K).LE.RR) GO TO 17                                    UM000790
         RR=RATIO(K)                                                    UM000800
         KK=K                                                           UM000810
   17    CONTINUE                                                       UM000820
   16 CONTINUE                                                          UM000830
      IF(KK.EQ.0) GO TO 18                                              UM000840
C                                                                       UM000850
C     UPDATE RATIC(K),S(K),BR                                           UM000860
C                                                                       UM000870
      S(KK)=S(KK)+1.0                                                   UM000880
C     RATIO(KK)=DELU(LAM(KK),T(KK),S(KK),C(KK),ESS(KK))                 UM000890
C     RATIO(KK)=DELTWS(LAM(KK),T(KK),S(KK),C(KK),ESS(KK))               UM000900
      RATIO(KK)=DELAV(LAM(KK),T(KK),S(KK),C(KK),ESS(KK),MTTR(KK))       UM000910
      BR=BR-C(KK)                                                       UM000920
      THETA=RR                                                          UM000930
      GO TO 19                                                          UM000940
C                                                                       UM000950
C-----------------------------------------------------------------------UM000960
```

```
C
C     CALCULATE OBJECTIVE FUNTIONS WITH S VECTORS OBTAINED FORM THE
C        ABOVE STEP
C
  18  OBJ1=0.0
      OBJ2=0.C
      OBJ3=1.0
      SUMLTM=0.0
      DO  20  J=1,N
         LTM=LAM(J)*T(J)
         X=S(J)
         MTBF=1.0/LAM(J)
         SUMLTM=SUMLTM+LTM
         US(J)=UNITS(LAM(J),T(J),X)
         MSRT(J)=TWS(LAM(J),T(J),X)/LTM
         AAV(J)=MTBF/(MTBF+MTTR(J)+MSRT(J))
         OBJ1=OBJ1+ESS(J)*US(J)
         OBJ2=OBJ2+ESS(J)*MSRT(J)*LTM
         OBJ3=OBJ3*AAV(J)
  20     CONTINUE
         OBJ1=OBJ1/SUMLTM
         OBJ2=OBJ2/SUMLTM
C
C-----------------------------------------------------------------------
C
C     OUTPUT FCR THIS TEST
C
C
      WRITE(6,25)
      DO  26  I=1,N
         WRITE(6,27)  I,S(I),US(I),MSRT(I),AAV(I)
  26  CONTINUE
  25  FORMAT(///1X,'  ==== ALLOCATION OF ITEMS  ======'///
     /5X,'  I   ',10X,' S(I)  ',5X,'  US(I)   ',5X,'MSRT(I)',5X,' AVIL ')
  27  FORMAT(5X,I3,12X,F6.1,5X,F9.4,3X,F9.4,5X,F9.5)
      WRITE(6,30) B,BR,THETA,OBJ1,OBJ2,OBJ3
  30  FORMAT(//10X,'BUDGET LIMIT =',F10.2/
     /          10X,'BUDGET LEFT =',F10.2/
     /          10X,'THETA        =',F10.6/
     /          10X,'OBJECT VALUE OF UNITS SHORT MODEL  =',F10.6/
     /          10X,'OBJECT VALUE OF TWUS        MODEL  =',F10.6/
     /          10X,'OBJECT VALUE OF AVAIL.      MODEL  =',F10.6//)
      STOP
      END
C    POISSON COMPLMENTRY CUMULATIVE DISTRIBUTION FUNTIJN
C
      REAL FUNCTICN PCCDF(AL,TT,SS)
      REAL AL,TT,SS,PG,CDF,SN,AA
```

```
         AA=AL*TT                                                       UM001450
         SN=0.0                                                         UM001460
         PCCDF=1.C                                                      UM001470
         IF(SS.LE.0.0) GO TO 26                                         UM001480
         PO=EXP(-AA)                                                    UM001490
         CDF=PO                                                         UM001500
         PCCDF=1.-CDF                                                   UM001510
         IF(SS.EQ.1.0) GO TO 26                                         UM001520
C        IF(CCF.GT.0.9999) PCCDF=0.0001                                 UM001530
C        IF((SS-1.0).EQ.0.0) GO TO 26                                   UM001540
  22     SN=SN+1.C                                                      UM001550
         PO=PO*AA/SN                                                    UM001560
         CDF=CDF+FO                                                     UM001570
         PCCDF=1.C-CDF                                                  UM001580
         IF(PCCDF.LE.0.00001) GO TO 26                                  UM001590
         IF(SN.GE.(SS-1.0)) GO TO 26                                    UM001600
C        IF(CCF.GT.0.99999) PCCDF=0.0001                                UM001610
C        SN=SN+1.0                                                      UM001620
         GO TO 22                                                       UM001630
  26  CONTINUE                                                          UM001640
      RETURN                                                            UM001650
      END                                                               UM001660
C                                                                       UM001670
C    POISSON MASS FUNTION                                               UM001680
C                                                                       UM001690
      REAL FUNCTICN PDEN (AL,TT,SS)                                     UM001700
      REAL AL,TT,SS,AA                                                  UM001710
         SN=1.0                                                         UM001720
         AA=AL*TT                                                       UM001730
         PDEN=EXP(-AA)                                                  UM001740
         IF(SS.LE.0.0) GO TO 27                                         UM001750
  28     PDEN=PDEN*AA/SN                                                UM001760
         IF(SS.EQ.SN) GO TO 27                                          UM001770
         SN=SN+1.0                                                      UM001780
         GO TC 28                                                       UM001790
  27     CONTINUE                                                       UM001800
      RETURN                                                            UM001810
      END                                                               UM001820
C                                                                       UM001830
C     TIME WAITED UNITS SHORT OBJECTIVE FUNTICN                         UM001840
C                                                                       UM001850
      REAL FUNCTICN TWS(AL,TT,S1)                                       UM001860
      REAL AL,TT,S1,Z1,Z2,Z3,AA                                         UM001870
      AA=AL*TT                                                          UM001880
      Z1=AA*PCCDF(AL,TT,S1)                                             UM001890
      Z2=2.0*S1*PCCDF(AL,TT,S1+1.0)                                     UM001900
      Z3=S1*(S1+1.0)*PCCDF(AL,TT,S1+2.0)/AA                             UM001910
      TWS=0.5*TT*(Z1-Z2+Z3)                                             UM001920
```

```
      RETURN                                                            UM001930
      END                                                               UM001940
C                                                                       UM001950
C     CALCULATE THE OBJECTIVE FUNTION FOR UNIT SHORT MODEL              UM001960
C                                                                       UM001970
      REAL FUNCTION UNITS(AL,TT,S2)                                     UM001980
      REAL AL,TT,S2                                                     UM001990
      UNITS=AL*TT*PCCDF(AL,TT,S2-1.0)-S2*PCCDF(AL,TT,S2)                UM002000
C     UNITS=AL*TT*PDEN(AL,TT,S2)+(AL*TT-S2)*PCCDF(AL,TT,S2+1.)          UM002010
      RETURN                                                            UM002020
      END                                                               UM002030
C                                                                       UM002040
C    THIS FUNCTION 'DELU' CALCULATES (E/C(Z(S-1)-Z(S)) FOR UNITS        UM002050
C    SHORT MODEL                                                        UM002060
C                                                                       UM002070
      REAL FUNCTION DELU(AL,TT,S2,CC,E)                                 UM002080
      REAL AL,S2,TT,CC,E                                                UM002090
      DELU=PCCDF(AL,TT,S2)                                              UM002100
C     DEL1=UNITS(AL,TT,S2-1.0)                                          UM002110
C     DEL2=UNITS(AL,TT,S2)                                              UM002120
C     DELU=DEL1-DEL2                                                    UM002130
      DELU=(E/CC)*DELU                                                  UM002140
      RETURN                                                            UM002150
      END                                                               UM002160
C                                                                       UM002170
C   THIS FUNTION COMPUTES  THE MARGINAL EFFECTAS INCREASE ONE MORE      UM002180
C    ITEMS FOR TWUS MODEL CASE                                          UM002190
                                                                        UM002200
      REAL FUNCTION DELTWS(AL,TT,SS,CC,E)                               UM002210
      REAL AL,TT,SS,CC,E,DEL1,DEL2,DEL3                                 UM002220
      DEL1=TWS(AL,TT,SS)                                                UM002230
      DEL2=TWS(AL,TT,SS-1.0)                                            UM002240
      DELTWS=(E/CC)*(DEL2-DEL1)                                         UM002250
C     DEL3=(TT-SS/AL)*PCCDF(AL,TT,SS)-(SS/AL)*PDEN(AL,TT,SS)            UM002260
C     DELTWS=(E/CC)*DEL3                                                UM002270
C                                                                       UM002280
C                                                                       UM002290
      RETURN                                                            UM002300
      END                                                               UM002310
C                                                                       UM002320
C    THIS FUNCTION COMPUTE THE MARGINAL EFFECT  AS INCREASE ONE MORE ITEUM002330
C         ( PSEUDO AVAILABILITY MODEL CASE)                             UM002340
C                                                                       UM002350
      REAL FUNCTION DELAV(AL,TT,S4,CC,E,MTR)                            UM002360
      REAL AL,TT,CC,E,MTR,S4,A1,A2,MST1,MST2,MTF                        UM002370
      MST1=TWS(AL,TT,S4)/(AL*TT)                                        UM002380
      MST2=TWS(AL,TT,S4-1.0)/(AL*TT)                                    UM002390
      MTF=1.0/AL                                                        UM002400
```

```
A1=MTF/(MTF+MTR+MST1)                                             UM002410
A2=MTF/(MTF+MTR+MST2)                                             UM002420
DELAV=(ALOG(A1)-ALOG(A2))*E/CC                                    UM002430
RETURN                                                            UM002440
END                                                               UM002450
```

LIST OF REFERENCES

1. Hadley, G. and Whitin, T.M., Analysis of Inventory Systems, Prentice-Hall, 1963.

2. Muckstadt, John A., "A Model for a Multi-Item, Multi-Echelon, Multi-Indenture Inventory System," Management Science (U.S), V. 20, p. 472-481, December 1973.

3. Eriksson, B., OPUS-VII Manual, Systecon AB, Stockholm, 1981.

4. Everett, H. III, "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," Operations Research, V. 11, p. 393-417, November 1963.

5. Daeschner, William Edward, Models for Multi-Item Inventory Systems with Constraints, Ph.D. Thesis, Naval Postgraduate School, June 1975.

6. Abell, John B. and Lengel, Joan E., Toward the Use of Availability Models for Spares Computations in the Department of Defense, Logistics Management Institute, June 1982.

7. Jee, Man-Won, Mathematical Models for Operational Availability, Ph.D. Thesis, Naval Postgraduate School, September 1980.

8. Naval Postgraduate School NPS Technical Report NPS55-83-026, Wholesale Provisioning Models by Richards, F. Russell and A.W. McMasters, p. 26-29, September 1983.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93943	1
4.	Professor F. Russell Richards, Code 55Rh Naval Postgraduate School Monterey, California 93943	5
5.	Professor Gilbert T. Howard, Code 55Hk Naval Postgraduate School Monterey, California 93943	1
6.	Chung, Won-Hwa, 102ho, Na-dong, Sinan A.P.T. 10-10 ho Sinlim-dong Kwanak-ku Seoul, Korea	2
7.	Kang, Sung-Jin, 107-ho, 3-dong Saehan Manson A.P.T. 1702 Daemyung 5-dong Nam-ku Deagu-Si Seoul, Korea	2
8.	Kang, Sung-Jin, 265 Game-dong Waryong-myun Andong-Kun Kyungbuk-do Seoul, Korea	5