OPTIMUM N-STAGE PRODUCTION POLICIES
UNDER A LEARNING EFFECT

By

BOBBY CARROLL SPRADLIN

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1961

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1963

Submitted to the Faculty of the Graduate School
of the Oklahoma State University
in partial fulfillment of
the requirements for
the degree of
DOCTOR OF PHILOSOPHY
May, 1965

OPTIMUM N-STAGE PRODUCTION POLICIES

UNDER A LEARNING EFFECT

Thesis Approved:

Thesis Adviser

Dean of the Graduate School

PREFACE

This investigation is based upon the assumption that a learning effect is present in a production system. The primary objective of this dissertation is to show that dynamic programming can be used as an effective technique to cope with the effects of learning as it pertains to an operational system. By the use of this technique, optimum N-stage production policies are derived that span a predetermined planning period and take into account variable regression on the manufacturing progress function. The number of stages need not be specified beforehand since the technique also determines the optimum number of production quantities.

The text of this study is somewhat theoretical and a basic knowledge of dynamic programming is helpful for its understanding. The Appendix contains a non-computer example that is also helpful in understanding the actual mechanics of the procedures presented in the text. For a rigorous development of the mathematical techniques used in this analysis, it is suggested that the reader consult references (1) and (2) of the Bibliography.

Indebtedness is acknowledged to the members of my Advisory Committee: Professors W. J. Bentley,

P. E. Torgersen, W. J. Fabrycky, J. L. Folks, and J. E. Shamblin, for their helpful suggestions and guidance throughout my doctoral program and this investigation.

I am also grateful to the School of Industrial Engineering and Management for providing an assistantship, thus making graduate study possible for me.

D. A. Pierce deserves special mention for suggesting this problem and for his helpful criticism and comments throughout the study.

Also, I wish to thank Miss Velda Davis for doing an excellent job of typing this dissertation. Finally, my wife, Stona, deserves great credit for her financial contribution and role as a morale builder during this effort.

TABLE OF CONTENTS

LIST OF FIGURES

Figure | Page

CHAPTER I

INTRODUCTION

The derivation of optimum production policies under a learning effect gives rise to a multi-stage decision process which may be described as follows. A system exists whose state at any time may be specified by a vector. The components of this vector may be the production quantities of a multi-stage decision process. Over a period of time, the variables describing the system may undergo a transformation. If the transformation may be chosen, then a decision process exists. If a sequence of decisions are to be made, the decision process may be called a multi-stage decision process.

Multi-stage decision processes are encountered in investment programs in production operations, and in military operations. In production operations, these processes may be found in the specific areas of sequential testing, inventory control, or production scheduling. A large number of these problems are of a degree of difficulty that precludes formal analysis. In these cases, the required decision must be based entirely on intuition and judgment; a few may be analytically treated by classical procedures.

Each decision of a multi-stage decision process may be thought of as a choice of variables that will have a particular effect on the system. A sequence of such choices constitutes a policy. If all of these choices are considered together, a multi-stage decision process is reduced to a classical problem of determining the maximum or minimum of a specific function. This function, which is developed in the course of measuring some quantitative property of the system, serves as a means for evaluating policies.

To maximize or minimize a well behaved function does not seem too overwhelming as it merely encompasses taking partial derivatives and, hence, the solution of a set of equations. However, as the number of equations increases, the difficulty in obtaining a solution increases very rapidly. In addition, if the solution is a boundry point in the region of variation, then the calculus is not a sufficient method of analysis. This is a result of the fact that certain decision processes have an all or nothing characteristic about them. The result of these falacies in traditional analysis for the purpose of multi-stage decision processes forces the use of combinations of analytic and search techniques to obtain solutions.

Any procedure which employs a search technique can only become more difficult as the number of variables and equations become larger. Also, any solution that simply employs enumeration is not satisfactory in that it does not provide insight into how sensitive a system is to change. In modeling the particular production systems with which this work is concerned, a method of analysis termed Dynamic Programming by Richard Bellman (1) is used. According to Bellman (1), it is the structure of the policy which is essential. This means that instead of determining the optimal sequence of decisions from some fixed state of the system it is desirable to determine the optimal decision to be made at any state of the system. This formulation allows a reduction in the dimension of the decision to one that is acceptable; that is, a particular stage at a time.

To illustrate this principle, a particular problem of maximizing the function,

$$P(X_1, \ldots, X_n) = f_1(X_1) + f_2(X_2) + \ldots + f_n(X_n)$$

over the region $X_i \geq 0$, $\sum_i^n X_i = X$ is discussed. To treat this problem, it is imbeded within a family of allocation processes. Instead of considering a particular quantity of resources and a fixed number of activities, an entire family of such problems, where X will range over a grid of positive values and n may be any positive integer, is considered. The problem is given a timelike quality by requiring that resources be allocated to each of the activities $f_i(X_i)$ one at a time. A dynamic n stage allocation process is created by starting with $f_n(X_n)$ and proceeding to $f_1(X_1)$.

The dependence of the maximum of $P(X_1, \ldots, X_n)$ on X and n is specified by introducing a sequence of functions $\{g_n(X)\}$ defined for $n = 1, 2, \ldots, X \geq 0$. Let

$$g_n(X) = \max_{\{X_i\}} P(X_1, \ldots, X_n)$$

over the same region

$$X_i \geq 0 \;, \sum_{i}^{n} X_i = X$$

as above.

The function $g_n(X)$ is the maximum return from an allocation of X resources to n activities. It is easily seen that

$$g_n(0) = 0 \qquad n = 1, 2, \ldots\ldots,$$

since it is true that $f_i(0) = 0$, $i = 1, 2, \ldots, n$, and, also, $g_1(X) = f_1(X) \qquad \forall X \geq 0$.

It is now desirable to have a recurrence relation connecting $g_n(X)$ and $g_{n-1}(X)$ for an arbitrary n and X. Let X_n, $0 \leq X_n \leq X$ be the amount of the resource lent to the n^{th} activity. Disregarding the exact value of X_n, it is known that the remaining quantity of resources is used to obtain a maximum return from the remaining $(n-1)$ activities.

By definition, the maximum return from $(n-1)$ activities starting with $X - X_n$ resources is $g_{n-1}(X - X_n)$, and, thus, a total return resulting from allocating X_n resources to the n^{th} activity is

$$\text{(a)} \quad f_n(X_n) + g_{n-1}(X - X_n).$$

It is clear that the optimal choice of X_n is one that maximizes (a). This leads to the basic functional equation

$$g_n(X) = \max_{0 \leq X_n \leq X} [f_n(X_n) + g_{n-1}(X - X_n)]$$

for $\qquad n = 1, 2, \ldots .$

The Principle of Optimality (2), which is used to obtain the previous functional equation, is now stated.

> An optimal policy has the property that, whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

A direct derivation of the preceding functional equation is as follows:

(a) observing that

$$\max_{\substack{X_1+X_2+\ldots+X_n=X \\ X_i \geq 0}} = \max_{0 \leq X_n \leq X} \left[\max_{\substack{X_1+X_2+\ldots+X_{n-1}=X-X_n \\ X_i > 0}}\right].$$

(b) It can be written

$$g_n(X) = \max_{\substack{X_1+X_2+\ldots+X_n \\ X_i \geq 0}} \left[f_n(X_n) + f_{n-1}(X_{n-1}) + \ldots + f_1(X_1)\right]$$

$$= \max_{0 \le X_n \le X} \left[\max_{\substack{X_1+X_2+\dots+X_{n-1}=X-X_n \\ X_i \ge 0}} (f_n(X_n)+f_{n-1}(X_{n-1})+\dots+f_1(X_1)) \right]$$

$$= \max_{0 \le X_n \le X} \left[f_n(X_n) + \max_{X_1+X_2+\dots+X_{n-1}= \;-X_n} (f_{n-1}(X_{n-1})+\dots+f_1(X_1)) \right]$$

$$= \max_{0 \le X_n \le X} \left[f_n(X_n) + g_{n-1}(X-X_n) \right] \quad ,(2).$$

All of the models derived in this dissertation are based on this property of multi-stage decision processes.

CHAPTER II

AN OPTIMUM PRODUCTION SCHEDULE UNDER THE LEARNING EFFECT

Basically, the situation which exists under the learning effect is a variable production rate. Looking at the situation from the viewpoint of raw materials supply, there is, of course, a variable demand rate. This is an equally important problem but is not considered here. A major problem exists in determining a production schedule even if the demand for the finished product is constant. It is desirable frequently to produce until an inventory is built up and then terminate production until this inven- ventory reaches a certain minimum level, at which time the cycle is repeated. In the more usual situation where the potential production rate is constant, this problem is merely a one dimensional one which consists of finding an economic order quantity. This technique is the classical introduction to total value analysis and is presented in many basic texts on economic analysis (4).

When the potential production rate is affected by manufacturing progress, however, the determination of an economic lot size is more difficult. First, the unit costs of production are higher at the beginning of the production run. More important, however, is the fact that whenever production is terminated for a period of time, the learning function regresses toward its ignorant state. It stands to reason that there must also be a balance between the cost of carrying higher inventories and the cost of regression in learning. The smaller the production lots, the lower the average inventory; but, then the process must be interrupted more often causing a greater "loss of learning."

Before going further into the analysis of this particular problem, it is pointed out that this is only one of the many operational problems which are created by the learning effect. There are problems associated with raw materials supply, manpower requirements, research and training programs, and many other facets of operational systems. In subsequent chapters, the production scheduling problem is expanded to include probabilistic demand, alternative demand for facilities, and the problem of inventory supply.

It seems that the first thing to be attacked in this problem is, "What are the characteristics of the regression in learning?" With some reflection, it is seen that the regression is not a constant value each time the process is interrupted. This is because the learning per unit diminishes with the number of units made. For constant regression, the total learning curve including regressions reverses itself after a certain number of units are made.

It is assumed, therefore, that the regression is related to the absolute rate of learning at the point at which the interruption occurs. It is assumed that the regression is equal to the amount learned during the production of the last M units.

By "the amount learned" is meant the change in the ordinate on the manufacturing progress function. The appropriate value for M is chosen to reflect the particular situation to which this analysis is being applied.

The next immediate problem is whether the optimal lot sizes are equal, resulting again in a one dimensional problem, or if they are variable, resulting in an n dimensional problem -- n being the number of production runs. It is believed that since the regressions in learning are variable, the optimal lot sizes are also variable.

In order to solve the problem for a variable lot size, some predetermined finite time period is chosen over which to plan production. The approach to this variable lot size problem is to determine a vector, $(Q_1, Q_2, \ldots, Q_n)$, of lot sizes which fulfill the requirements for the interval being filled, both the Q_i's and n being unknown.

Bellman (1) points out that in this type of problem there exists a choice of solving one n dimensional problem or n one (or two) dimensional problems. Since the computations involved in solving a programming problem usually increase exponentially with dimension, the alternative of solving n one dimensional problems is nearly always more feasible. This alternative, termed dynamic programming, is used here. Note also that if the problem is approached as an n dimensional one it is required to determine n beforehand. It is found that in using the dynamic programming approach, the optimum value of n as well as the Q vector is determined.

An outline of the dynamic programming approach to this problem is as follows:

Let the production quantities be $Q_i (i = n, n-1, \ldots, 1)$.

X be the number of units made since the initial unit of production.

Y be the total number of units for which production is being planned.

$f_n(X)$ be the cost of an optimal policy in which n orders are placed beginning Q_n with unit X and finishing Q_1 with unit Y.

$C_X(Q_i)$ be the cost of producing and storing Q_i units beginning at unit X until they are sold.

Note that the first order quantity is Q_n, the next Q_{n-1}, and so on until Q_1. It is assumed that production is very rapid relative to consumption and the production of lot Q_{i-1} does not begin until lot Q_i is depleted.

Application of the Principle of Optimality gives:

$$f_n(X) = \min_{0 \leq Q_n \leq Y-X} \left\{ C_X(Q_n) + f_{n-1}(X + Q_n - M) \right\} (n = 2, 3, \ldots)$$

$$f_1(X) = C_X(Y - X).$$

This functional relationship can best be explained by returning to the example under consideration.

The first step is to determine an optimal two-stage policy spanning from any unit X until the end of the planning period. These optimal policies are denoted by $f_2(X)$. They consist simply of the best combination of Q_1 and Q_2 which meets the demand from any point in the planning period until the end of the planning period. When starting within the planning period, it is assumed for purposes of the learning curve that enough units are already made to meet the requirements from the beginning of the period up until that time.

The next step is to determine the optimal three-stage policies. These policies likewise span from any point in the planning period until the end of the period. This determination of three-stage policies is effectively reduced, however, to the determination of two-stage policies because of the values of $f_2(X)$ already determined and the principle of optimality. This is done as follows:

$$f_3(X) = \min_{0 \leq Q_3 \leq Y-X} \left\{ C_X(Q_3) + f_2(X + Q_3 - M) \right\} .$$

The only variable which needs to be manipulated is Q_3, since once it is determined the remaining optimal policy $f_2(X + Q_3 - M)$, is already known and tabulated.

Using the general relationship:

$$f_n(X) = \min_{0 \leq Q_n \leq X-Y} \left\{ C_X(Q_n) + f_{n-1}(X + Q_n - M) \right\}.$$

This procedure is continued until $f_{n-1}(1) \leq f_n(1)$, which means that the cost of the policy is not reduced by adding another stage. An optimal n stage policy is now determined which spans from the first unit produced until the end of the planning period.

The best, although seldom possible, technique for determing the optimal policies at each stage is as follows: Set

$$\frac{\partial f_2(X)}{\partial Q_2} = \frac{\partial \{C_X(Q_2) + f_1(X + Q_2 - M)\}}{\partial Q_2} = 0$$

to minimize $f_2(X)$ with respect to Q_2. This expression is then used to solve:

$$\frac{\partial f_3(X)}{\partial Q_3} = \frac{\partial \{C_X(Q_3) + f_2(X + Q_3 - M)\}}{\partial Q_3} = 0$$

in order to minimize $f_3(X)$ with respect to Q_3.

This procedure is carried out for larger values of n until the noted termination occurs. The equations obtained when the derivatives are set equal to zero nearly always become unmanageable, however, leading one to use a search technique to find the minimum of the functions. The procedure is as follows. Try values of Q_2 for a given X until $f_2(X)$ is found. Repeat this process for a grid of values of X and tabulate all these values. Then $f_3(X)$ for the grid of X values is found from the relationship:

$$f_3(X) = \min_{0 \leq Q_3 \leq Y-X} \left\{ C_X(Q_3) + f_2(X + Q_3 - M) \right\}$$

by trying different values of Q_3 and the tabulated values of $f_2(X)$. This procedure is carried out until termination as described above.

The specific method for solving the problem at hand is now outlined. The equation for the exponential learning function was derived in Appendix A and is:

$$T_X = KX^n$$

where T_X = time to make unit X

K = time to make unit one

$n = \log_{10}N / \log_{10} 2$

where N is the rate of time to make unit j to the time to make unit 2j for any j.

Since the direct labor costs are approximately proportional to the production time per unit, T_X is replaced with C_X, the direct labor cost of unit X, and the K is the direct labor cost of the first unit. For this analysis, all other production cost will be assumed constant and neglected. The costs which will be included are the direct labor costs, setup costs, and inventory storage costs.

The direct labor costs for a lot size of Q_n are cost per unit times the number of units or

$$\sum_{X}^{X+Q_n} KX^n dX$$

where X is the number of the first unit in the lot.

The storage costs per unit time are expressed as a percentage of the average number of units stored during that unit of time. If this per cent is noted by C, the storage costs for a lot of size Q_n from the time it is received until the time it is depleted are:

$$C \cdot \frac{Q_n}{d} \cdot \frac{Q_n}{2} = C \cdot \frac{Q_n^2}{2d}$$

where d is the demand rate per unit time, assumed to be constant here. The setup costs are S dollars per order.

As discussed before, the regression in learning from the time production is stopped on one order until it is begun again on the next is taken as the amount learned on the last M units. This is accounted for in the algorithm by moving back M units toward the origin the integral representing direct labor cost every time production is interrupted.

$f_2(X)$ is now found as follows:

$$f_2(X) = \min_{0 \leq Q_2 \leq Y-X} \left\{ S + \frac{C \cdot Q_2^2}{2d} + \sum_{X-M}^{X+Q_2-M} KX^n + S + \frac{C \cdot Q_1^2}{2d} + \sum_{X+Q_2-2M}^{X+Q_1+Q_2-2M} KX^n \right\}$$

which simplifies to:

$$f_2(X) \doteq \min_{0 \leq Q_2 \leq Y-X} \left\{ 2S + C(Q_1^2 + Q_2^2)/2d + \frac{KX^{n+1}}{n+1} \Big|_{X-M}^{X+Q_2-M} + \frac{KX^{n+1}}{n+1} \Big|_{X+Q_2-2M}^{X+Q_1+Q_2-M} \right\}$$

which is solved for a series of X values by using a search technique.

Now to solve for $f_3(X)$:

$$f_3(X) = \min_{0 \leq Q_3 \leq Y-X} \left\{ S + \frac{C \cdot Q_3^2}{2d} + \sum_{X-M}^{X+Q_3-M} KX^n + f_2(X + Q_3 - M) \right\}$$

$$\doteq \min_{0 \leq Q_3 \leq Y-X} \left\{ S + \frac{C \cdot Q_3^2}{2d} + \frac{KX^{n+1}}{n+1} \Big|_{X-M}^{X+Q_3-M} + f_2(X+Q_3-M) \right\}$$

which is also solved for a series of X values by using the values of $f_2(X)$ that are already tabulated. This procedure is carried out until the termination noted before occurs (see Figure 1).

The equations $f_n(X)$ do two things in this analysis. They:

1. Obtain the total production cost from the first unit produced to the end of the planning period, and
2. They are used to obtain the values of $f_{n-1}(X)$ that are used in the functional equations.

The following derivations and illustrations serve to establish the role of the previous functional relationship.

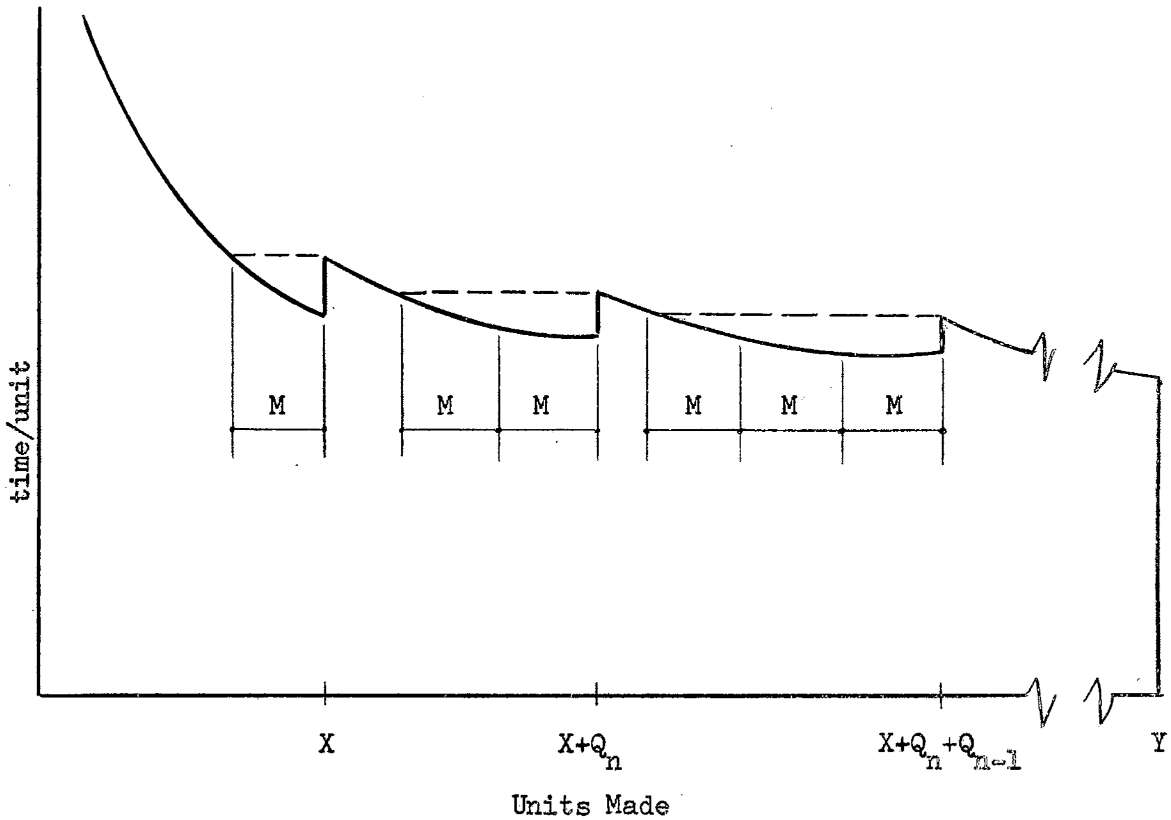


Figure 1. Obtaining Structures of Optimal n-1 Stage Policies From Any Point in the Planning Period

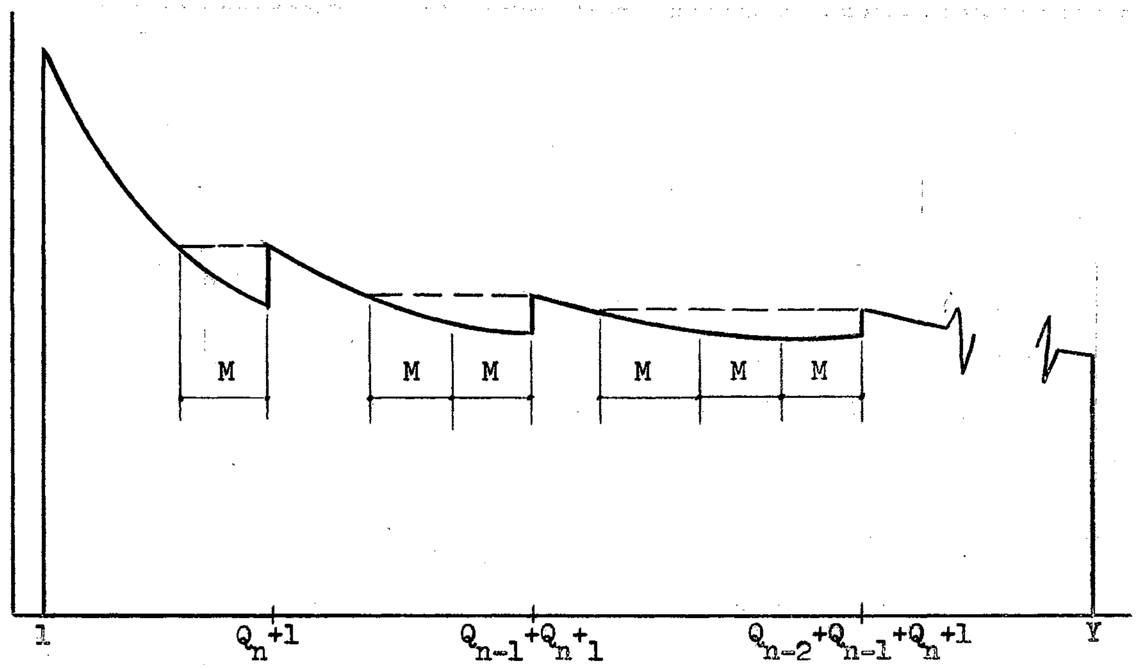


Figure 2. The Total Direct Labor Cost From the First Unit of Production

To obtain the total direct labor cost from the first unit of production:

$$f_n(1) = \sum_{1}^{1+Q_n} f(X) + \sum_{1+Q_n-M}^{1+Q_n+Q_{n-1}-M} f(X) + \ldots +$$

$$\sum_{1+Q_n+\ldots+Q_{n-K}-(K+1)M}^{1+Q_n+Q_{n-1}+\ldots+Q_{n-(K+1)}-(K+1)M} f(X)$$

where $0 \leq K \leq n-1$. Now to obtain $f_{n-1}(X)$, only the direct labor cost is considered to illustrate how to obtain $f_{n-1}(X)$ from any point X in the planning period.

$$f_n(X) = \min_{Q_n}\left\{\sum_{X-M}^{X-M+Q_n} (f_n X) + f_{n-1}(X + Q_n - M)\right\}$$

$$= \min\left\{\sum_{X-M}^{X+Q_n-M} f(X) + \sum_{X+Q_n-2M}^{X+Q_n+Q_{n-1}-2M} f(X) + \right.$$

$$+ \sum_{X+Q_n+Q_{n-1}-3M}^{X+Q_n+Q_{n-1}+Q_{n-2}-3M} f(X) + \ldots +$$

$$+ \sum_{X+Q_n+Q_{n-1}+\ldots+Q_{n-K}-(K+2)M}^{X+Q_n+\ldots+Q_{n-(K+1)}-(K+2)M} f(X).$$

$(X - M)$ is substituted for (1) in the equation $f_n(1)$ to see if the total direct labor cost equation works for every X as well as for X equal one.

If $(X - M)$ is used in $f_n(1)$ instead of (1), one obtains

$$f_n(X-M) = \sum_{X-M}^{X+Q_n-M} f(X) + \sum_{X+Q_n-M}^{X+Q_n+Q_{n-1}-2M} f(X) + \sum_{X+Q_n+Q_{n-1}-3M}^{X+Q_n+Q_{n-1}+Q_{n-2}-3M} f(X)$$

$$+ \ldots + \sum_{X+Q_n+Q_{n-1}+\ldots+Q_{n-K}-(K+2)M}^{X+Q_n+\ldots+Q_{n-(K+1)}-(K+2)M} f(X) + \ldots .$$

This is the result obtained from the functional equation using $f_n(X)$ to obtain $f_{n-1}(X)$.

Now the functional equation in which $f_n(X)$ is used to obtain $f_{n-1}(X)$ for any X in the planning period is checked to see if it gives the total direct labor cost starting with the first unit produced and spanning the entire planning period.

To do this, substitute (1) for (X) in the functional equation,

$$f_n(X) = \min_{Q_n}\left\{\sum_{X-M}^{X-M+Q_n} f(X) + f_{n-1}(X+Q_n-M)\right\}.$$

Substituting (1) for (X) gives

$$f_n(1) = \min_{Q_n}\left\{\sum_{1}^{1+Q_n} f(X) + f_{n-1}(1+Q_n-M)\right\}$$

$$= \min_{Q_n}\left\{\sum_{1}^{1+Q_n} f(X) + \sum_{1+Q_n-M}^{1+Q_n+Q_{n-1}-M} f(X) + f_{n-2}(1+Q_n+Q_{n-1}-2M)\right\}$$

$$= \min_{Q_n}\left\{\sum_{1}^{1+Q_n} f(X) + \sum_{1+Q_n-M}^{1+Q_n+Q_{n-1}-M} + \sum_{1+Q_n+Q_{n-1}-2M}^{1+Q_n+Q_{n-1}+Q_{n-2}-2M} f(X) \ldots\right\}.$$

Since this is the total direct labor cost obtained by summing the entire learning function from unit one to the end of the planning period, it is clear that the functional equation gives the desired total direct labor cost.

A short numerical example of the optimizing process described in this chapter is presented in Appendix B. The purpose of this example is to demonstrate the manner in which the process is carried out. For ease of computation, the summation notation is replaced by integrals.

In Appendix C, a computer program for this procedure, along with optimum policies that arise when various loss of learning and storage cost are applicable, is presented.

CHAPTER III

AN OPTIMUM PRODUCTION SCHEDULE UNDER THE LEARNING EFFECT WHEN DEMAND VARIES

In some production processes, it is reasonable to assume that demand is constant. An example of this is the aircraft industry where a contract specifies the number of aircraft to be manufactured. However, there are industrial situations where a learning effect is clearly present and the demand for the product is a random variable. A situation of this nature is exemplified by the automobile industry. Formulation of this problem into the framework of a functional equation and a recurrence relationship is somewhat analogous to that of the preceding chapter except that now expected values are used.

The assumption of stochastic demand makes it necessary to redefine some of the equations and symbols previously used in the following manner:

Let X be defined as in the previous chapter

Y be defined as in the previous chapter

$C_X(Q_i)$ be the expected cost of producing and storing Q_i units beginning at unit X until they are sold.

$f_n(x)$ be the optimum expected cost of a policy in which n orders are placed beginning Q_n with unit X and finishing Q_1 with unit Y.

The assumption is made that after a production lot is completed, the process will shut down and begin again upon depletion of that lot.

The amount of regression M_i, i = 1, 2, ..., n is assumed to be a function of the demand over a time period and the amount of product produced to be used or sold in that period. This is reasonable since, if the production quantity is large and the demand small, a long wait is required before beginning the next production lot. This causes a greater regression toward the ignorant state as a result of having been away from the process longer. Therefore, let

$$M_i = \left[\frac{pQ_i}{d_i}\right] \qquad i = 1, 2, \ldots, n.$$

where d_i is the demand over a period of time, p is some proportionality constant and Q_i is the production lot to be dispensed during this period.

The bracketed quantity $\left[\frac{pQ_i}{d_i}\right]$ means that M_i is actually the greatest integer less than or equal to this amount. This is so that the value $(X + Q_i - M_i)$ in the functional equation $f_{i-1}(X + Q_i - M_i)$ will coincide with one of the grid values of X for i = 2, 3, 4, ... See Figure 3.

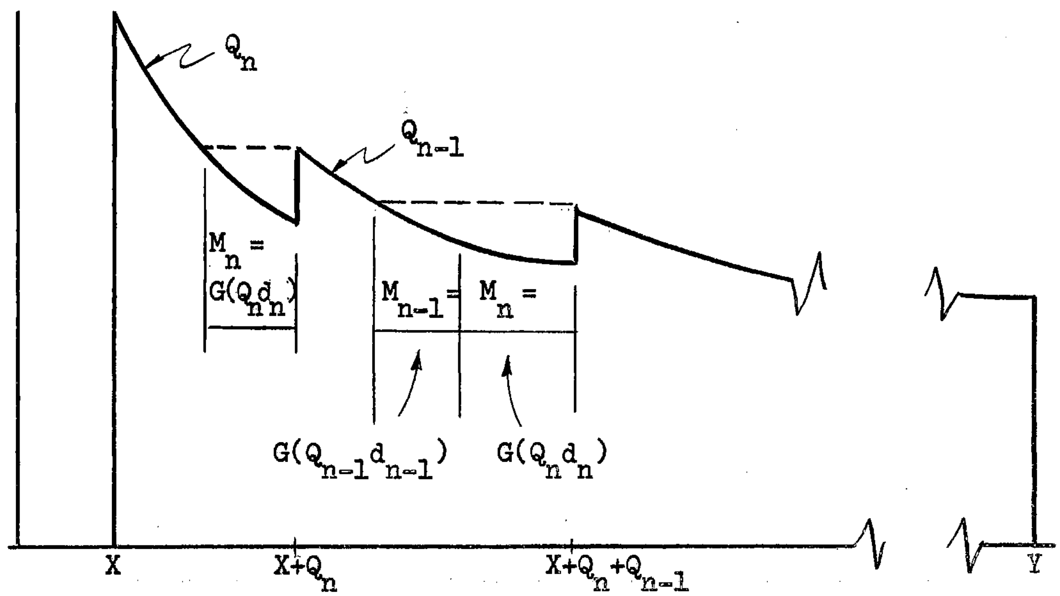


Figure 3. An Illustration of Regression in Learning When Demand is a Random Variable

The one-stage policies are formed as follows:

$$f_1(X) = E\left\{S + \frac{C \cdot Q_1^2}{2D} + \sum_{X-M_1}^{X+Q_1-M_1} KX^n\right\} = \left\{S + \frac{C \cdot Q_1^2}{2} E(1/d) + E \sum_{X-M_1}^{X+Q_1-M_1} KX^n\right\}$$

$$\approx \left\{S + \frac{C \cdot Q_1^2}{2} \cdot E(1/d) + E\left(\frac{KX^{n+1}}{n+1} \Big|_{X-M_1}^{X+Q_1-M_1}\right)\right\} = \left\{S + \frac{C \cdot Q_1^2}{2} E(1/d) + \right.$$

$$\left. \frac{K}{n+1} E\left[(X+Q_1-M_1)^{n+1} - (X-M_1)^{n+1}\right]\right\} .$$

Letting $A = (X+Q_1)$, $B = X$, $C = M_1$.

$$(A-C)^{n+1} - (B-C)^{n+1} = \sum_{i=0}^{n+1} \binom{N+1}{i} A^i (-1)^{n+1-i} C^{n+1-i} -$$

$$- \sum_{j=0}^{n+1} \binom{n+1}{j} B^j (-1)^{n+1-j} C^{n+1-j}$$

$$= \sum_{k=0}^{n+1} \binom{n+1}{k} (A-B)^k (-1)^{n+1-k} C^{n+1-k} .$$

$$f_1(X) \doteq \left\{ S + \frac{C \cdot Q_1^2}{2} E(1/d) + \frac{K}{n+1} E\left[\sum_{k=0}^{n+1} \binom{n+1}{k} \left(Q_1^k\right) (-1)^{n+1-k} (M_1)^{n+1-k} \right] \right\}$$

$$\doteq \left\{ S + \frac{C \cdot Q_1^2}{2} E(1/d) + \frac{K}{n+1} \left[\sum_{k=0}^{n+1} \binom{n+1}{k} Q_1^k (-1)^{n+1-k} E\left(\frac{PQ_1}{D}\right)^{n+1-k} \right] \right\}$$

$$\doteq \left\{ S + \frac{C \cdot Q_1^2}{2} E(1/d) + \frac{K}{k+1} \sum_{k=0}^{n+1} \binom{n+1}{k} Q_1^{n+1} (-P)^{n+1-k} E(1/d)^{n+1-k} \right\} .$$

These one-stage policies are used to compute the optimal two-stage policies.

$$f_2(X) = \min_{Q_2} E \left\{ S + \frac{C \cdot Q_2^2}{2d} + \sum_{X-M_2}^{X+Q_2-M_2} KX^n + f_1(X+Q_2-M_2) \right\}$$

$$\doteq \min_{Q_2} \left\{ S + \frac{C \cdot Q_2^2}{2} E(1/d) + \frac{K}{n+1} E\left[(X+Q_2-M_2)^{n+1} - (X-M_2)^{n+1} \right] + f_1(X + Q_2 - M_2) \right\}.$$

Again letting: $(X + Q_2) = A$, $C = M_2$, $B = X$

$$f_2(X) \doteq \min_{Q_2} \left\{ S + \frac{C \cdot Q_2^2}{2} E(1/d) + \frac{K}{n+1} \sum_{k=0}^{n+1} \binom{n+1}{k} Q_2 (-1)^{n+1-k} P^{n+1-k} \right.$$

$$Q_2^{n+1-k} E(1/d)^{n+1-k} + f_1(X+Q_2-M_2)\Big\}$$

$$f_2(X) \doteq \min_{Q_2}\Big\{S + \frac{C \cdot Q_2^2}{2} E(1/d) + \frac{K}{n+1} \sum_{k=0}^{n+1} \binom{n+1}{k} (-P)^{n+1-k}$$

$$Q_2^{n+1} E(1/d)^{n+1-k} + f_1(X+Q_2-M_2)\Big\} .$$

Proceeding in the same manner, the two-stage policies are used to determine the three-stage policies.

$$f_3(X) = \min_{Q_3} E\Big\{S + \frac{C \cdot Q_3^2}{2d} + \sum_{X-M_3}^{X+Q_3-M_3} KX^n + f_2(X+Q_3-M_3)\Big\}$$

$$\doteq \min_{Q_3}\Big\{S + \frac{C \cdot Q_3^2}{2d} E(1/d) + \frac{K}{n+1} E\Big[(X+Q_3-M_3)^{n+1} - (X-M_3)^{n+1}\Big]$$

$$+ f_2(X+Q_3-M_3)\Big\}$$

$$\doteq \min_{Q_3}\Big\{S + \frac{C \cdot Q_3^2}{2d} E(1/d) + \frac{K}{n+1} \sum_{k=0}^{n+1} \binom{n+1}{k} Q_3^k (-1)^{n+1-k}$$

$$E(M_3)^{n+1-k} + f_2(X+Q_3-M_3)\Big\}$$

$$f_3(X) \doteq \min_{Q_3}\Big\{S + \frac{CQ_3^2}{2d} \cdot E(1/d) + \frac{K}{n+1} \sum_{k=0}^{n+1} \binom{n+1}{k} Q_3^{n+1} (-P)^{n+1-k}$$

$$E(1/d)^{n+1-k} + f_2(X+Q_3-M_3)\Big\}.$$

Proceeding again in this same manner, the optimal n stage policies are determined from the functional equation:

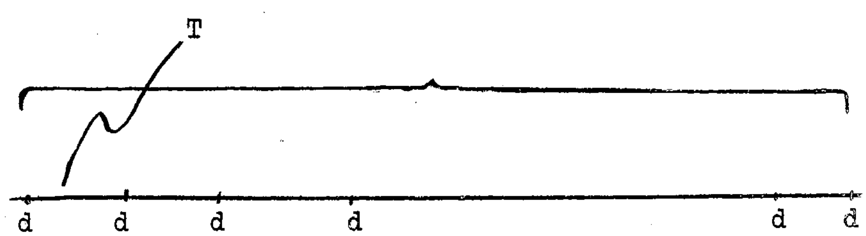
$$f_n(X) = \min_{0 \leq Q_n \leq Y-X} E\left\{S + \frac{C \cdot Q_N^2}{d} + \sum_{X-M_n}^{X+Q_n-M_n} KX^n + f_{n-1}(X+Q_n-M_n)\right\}.$$

The procedure is continued until $f_{n-1}(1) \leq f_n(1)$ which again means that the expected cost of the policy is not reduced by adding another stage. An optimal n stage policy is now determined which spans the entire production period.

There are certain distributions which, if applicable as either the demand distribution or an approximation to it, will make the actual computations of these expected values and, hence, the optimal policies much simplier.

If the demand distribution is approximated by the log normal distribution, then $1/d$ where $d \sim$ (log normal), is distributed as (-log normal). Then $E(1/d) = -E(d)$ also the higher moments may be readily found.

Frequently, if the demand is distributed as a Poisson variable, then T, the time between demand, is distributed exponentially.



M_i is expressed as the sum of m of the independent variables, T; therefore:

$$MGF(M_i) = [MGF(T)]^m.$$

The higher moments that are required in the functional equations are then found.

As a final example, if the demand distribution is approximated by some form of the F distribution, that is if $d \sim F(n_1, n_2)$, then $1/d \sim F(n_2, n_1)$. In this case, the expected values in the functional equations are found from the generating function of a particular F distribution.

The ease with which these functional equations are evaluated will depend on such things as the length of the planning period and the applicability of assuming a demand distribution G(d) that simplifies the finding of higher moments of the distribution of $1/d$.

CHAPTER IV

OPTIMUM POLICIES UNDER LEARNING FOR INVENTORY AND PRODUCTION SCHEDULING SIMULTANEOUSLY

In this phase of the study, the problem of obtaining an optimum policy for ordering inventory simultaneously with an optimum production schedule is considered. The method of attack is that of reducing the dimension of the problem. This time, instead of reduction from an n dimensional problem to n one dimensional problems, n two dimensional problems are considered.

A single commodity raw material is assumed, and it is suggested that if the dimension were increased at each stage more raw materials could be included. Situations do exist where a certain raw material used in a production process is of such importance costwise when compared to other raw materials used in the process that it justifies application of an optimizing process on its behalf. Also, the instant filling of raw material orders and a constant demand on production items is assumed. The costs associated with raw materials are procurement cost and holding cost.

In this particular study, shortage cost is neglected. This is due to an assumption about raw materials in the cost model that is made later on.

As a result of having two dimensions, instead of one at each stage, the functional equation differs somewhat from the previous ones.

The following notation is used:

Let	$f_n(X,Z)$	be the cost of an optimal policy given (1.) a location X on the learning curve and (2.) a raw material inventory.
	X	be the location on the learning curve.
	Z	be the given amount of raw material inventory.
	$P_i (i = 1, 2, \ldots, n)$	be the raw material order quantities.
	$Q_i (i = 1, 2, \ldots, n)$	again be the production quantities.

It is assumed that the raw material concerned is connected to the production unit in such a manner that one unit of production requires one unit of raw material. The following schematic diagrams are helpful in describing the situation as it is assumed.

For a one stage policy, there is but one choice for a production quantity and only one choice for the inventory lot other than 0 (see Figure 4).

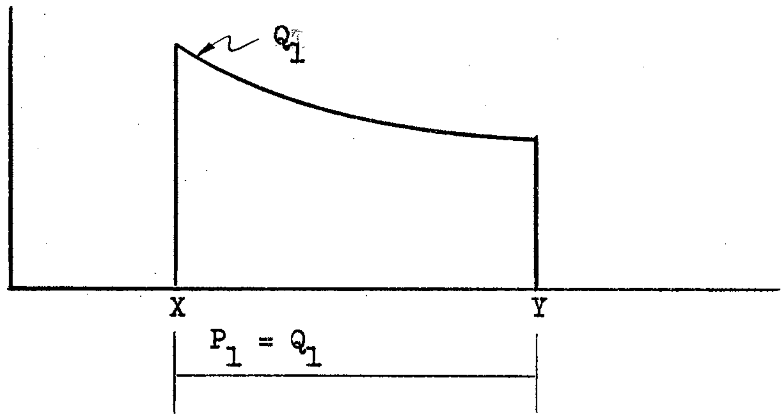


Figure 4. A One Stage Policy for Production Illustrating Alternatives for Ordering Inventory

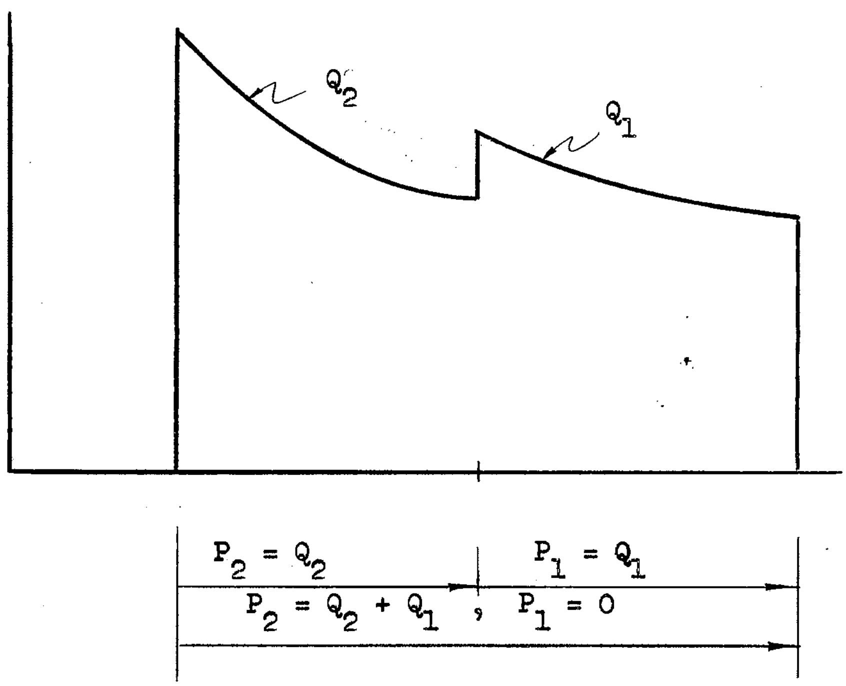


Figure 5. A Two Stage Policy for Production Illustrating Alternatives for Ordering Inventory

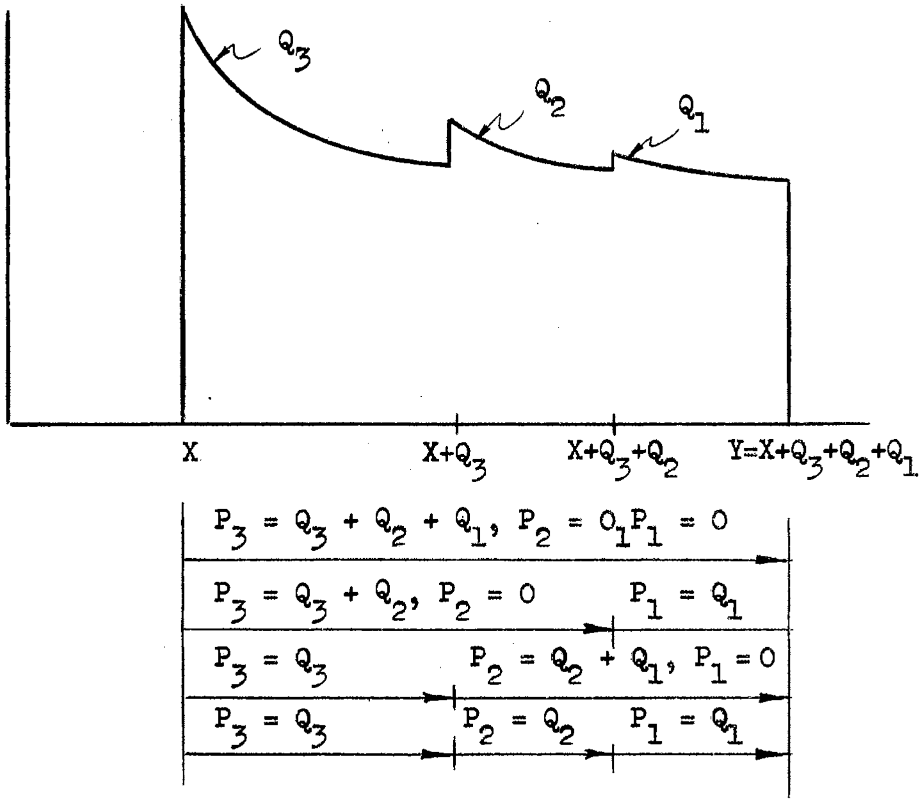


Figure 6. A Three Stage Production Policy Illustrating the Alernatives for Ordering Inventory

These figures indicate the fact that the order quantity of raw materials is restricted to be such that it coincides itemwise to the sum of a portion or all of the production lots.

The given amount of raw materials in the functional equation is under a similar restriction. Z is required to be of the following nature:

$$Z = \sum_{i=K}^{n} Q_i \quad K = 1,\ 2,\ \ldots,\ N \text{ for } Z \varepsilon f_n(X,Z) \text{ or } Z = 0.$$

For example, if $Z \varepsilon f_3(X,Z)$, then $Z = Q_3 + Q_2 + Q_1$, Q_3+Q_2, Q_3, or 0. This eliminates shortage cost on raw materials. $\sum_{i=n}^{n} Q_i$ is taken to be equal to Q_n.

The holding cost for raw materials is a constant times the average amount of inventory on hand for the period during which a particular batch of inventory is being used. If A is the constant, the holding cost for the i^{th} stage of the production schedule is

$$A \cdot f(Q_i)$$

where $f(Q_i)$ is the average amount of raw material on hand for the i^{th} production period.

In addition to this, if $Z \varepsilon f_i(X,Z) > Q_i$, there is the additional holding cost of

$$A \cdot [Z - Q_i]$$

for the i^{th} production period, since the entire amount $[Z - Q_i]$ is carried for the i^{th} period.

The procurement cost is D dollars per order. The following rule is implemented for procurement cost:

if $Z \varepsilon f_n(X,Z) = 0$, then $D = D$ for $D \varepsilon f_n(X,Z)$

if $Z \varepsilon f_n(X,Z) \neq 0$, then $D = 0$ for $D \varepsilon f_n(X,Z)$.

The algorithm is begun by finding the one stage policies in the following manner:

$$f_1(X, Z) = \underset{\substack{Q_1 = Y-X \\ P_1 = \begin{cases} Q_1 \\ 0 \end{cases}}}{} \left\{ D + A[Z - Q_1] + A \cdot f(Q_1) + \frac{C \cdot Q_1^2}{2d} + \sum_{X-M}^{X+Q_1-M} KX^n \right\}$$

where $Z \varepsilon f_1(X, Z) = Q_1, 0.$

These policies are used to determine the optimum two stage policies:

$$f_2(X, Z) = \min_{0 \leq Q_2 \leq Y-X} \left\{ D + A[Z - Q_2] + A \cdot f(Q_2) + S + \frac{C \cdot Q_2^2}{2d} + \sum_{X-M}^{X+Q_2-M} KX^n + f_1(X + Q_2 - M, Z + P_2 - Q_2) \right\}$$

$$P_2 = \begin{cases} Q_2 + Q_1 \\ Q_2 \\ 0 \end{cases}$$

where $Z \varepsilon f_2(X, Z) = Q_2 + Q_1, Q_2, 0.$

Using the optimal two stage policies, the three stage policies are found as indicated below:

$$f_3(X, Z) = \min_{0 \leq Q_3 \leq Y-X} \left\{ D + A \cdot [Z - Q_3] + A \cdot f(Q_3) + S + \frac{C \cdot Q_3^2}{2d} + \sum_{X-M}^{X+Q_3-M} KX^n + f_2(X + Q_3 - M, Z + P_3 - Q_3)^{X-M} \right\}$$

$$P_3 = \begin{cases} Q_3 + Q_2 + Q_1 \\ Q_3 + Q_2 \\ Q_3 \\ 0 \end{cases}$$

where $Z \varepsilon f_3(X, Z) = \sum_{i=1}^{3} Q_i, \ldots, \sum_{i=3}^{3} Q_i, 0.$

The general formulation is now seen for any number of stages.

$$f_n(X, Z) = \min_{0 \le Q_n \le Y-X} \left\{ D + A \cdot [Z - Q_n] + A \cdot f(Q_n) + S + \frac{C \cdot Q_n^2}{2d} \right.$$

$$P_n = \begin{cases} Q_n + Q_{n-1} + \dots + Q_1 \\ Q_n + \dots + Q_2 \\ \vdots \\ Q_n + Q_{n-1} \\ Q_n \\ 0 \end{cases} \qquad \sum_{X-M}^{X+Q_n-M} KX^n + f_{n-1}(X + Q_n - M, \left. , Z + P_n - Q_n) \right\}$$

where $Z \varepsilon f_n(X, Z) = \sum_{i=1}^{n} Q_i, \sum_{i=2}^{n} Q_i, \dots, \sum_{i=n}^{n} Q_i, 0.$

Although it seems at first that the grid of values for Z is rather coarse, with some reflection one sees that Z actually takes on all values. This is because as X moves from Y to the value 1 in the planning period, the production quantities Q_i take on all integer values and Z, defined as a sum or partial sum of a production schedule from X to the end of the planning period, also takes on all integer values.

This procedure is again carried out until

$$f_{n-1}(1, Z) \le f_n(1, Z) \quad \forall Z.$$

At this point, the number of stages n is determined also a vector of production quantities and, in addition, a vector of order quantities for raw materials, all which span the planning period from any point X within the planning period.

CHAPTER V

OPTIMUM PRODUCTION SCHEDULES UNDER LEARNING FOR TWO PRODUCTION FACILITIES

The incorporation of a second production process into the operational situation greatly enhances the realism of the problem of production scheduling. This means that facilities and manpower do not lie idle waiting for depletion of a previous production quantity, but instead go directly to producing a different item.

The analysis of this particular problem is carried out by reducing an n dimensional problem to n three dimensional problems. In this problem, it is assumed that both processes have a learning effect present. In addition to working from a point X to the end of the planning period under one manufacturing progress function, work is also started at a point W and proceeds to the end of the planning period under a second function. Both progress functions are assumed to be exponential functions of the type previously used and derived in Appendix A. The process containing the point X is called process A and the process containing the point W, process B. The assumption is made that there is sufficient demand for the production units of both processes to keep them both busy for the length of the planning period, and that the work crew can run either production system.

A new concern in this problem is that of running short or over with one product while engaged in the production of the other. A way to overcome this is to set up the rule that the work crew will begin on process A, produce a certain quantity of product and then switch to process B while this quantity produced on A is being depleted. Also, it is assumed that the demand and production rates of the two processes are known. The main problem with the rule for disallowing shortages and overages is that once the production quantity for the n^{th} stage is set, this automatically determines the size of the production quantities in each of the successive stages for both processes (see Figure 5). With some reflection, it is evident that depending on the nature of the demand on production rates for the two processes, this does not necessarily allow the desired balance between loss of learning and storage of finished product to be obtained. Another problem that results from the production quantities being determined for all the successive stages of production is that in determining the optimum policy at the n^{th} stage for a particular value of Q_n, all of the optimum (n-1) stage policies for this Q_n are not necessarily computed. This fact renders the method of analysis used on the previous problems useless. Therefore, in addition to a balance between loss of learning and storage of finished product, the shortages and overages are incorporated into the scheme of things to obtain a satisfactory over-all balance of cost.

The notation used in determining this algorithm is as follows. Let:

P_A	be the production rate for Process A
P_B	be the production rate for Process B
d_A	be the demand rate for finished product of Process A
d_B	be the demand rate for finished product of Process B
X	be the location on the learning curve of Process A
W	be the location on the learning curve of Process B
$Q_i (i = 1, 2, ..., n)$	be a production quantity, which if begun on either process, determines the schedule on both processes for the entire planning period.
$D_i (i = 1, 2, ..., n)$	be the production quantities for Process A
$Q_i (i = 1, 2, ..., n)$	be the production quantities for Process B
$f_n(X, W, Q_n)$	be the cost of an optimal policy, when shortages and overages are allowed, given a point X in the planning period for Process A, a point W

in the planning period for Process B, and a beginning production quantity Q_n for Process A that fixes subsequent production quantities of both processes so as to prevent shortage or overage (Figure 7).

Since the actual policy will be to allow shortages and overages, Figure 8 will illustrate how the schedule appears.

The setup cost for the two processes is S_1 for Process A and S_2 for Process B. The storage cost for Process A is

$$C \cdot D/d_A \cdot D/2 = C \cdot D^2/2d_A,$$

and the storage cost for Process B is

$$H \cdot O/d_B \cdot O/2 = H \cdot O^2/2d_B,$$

where H and C are constant cost per unit for each of the two Processes B and A, respectively.

The cost associated with the shortages and overages is as follows:

G dollars per unit for units over demand for Process A.

-D dollars per unit for units short of demand for Process A.

E dollars per unit for units over demand for Process B.

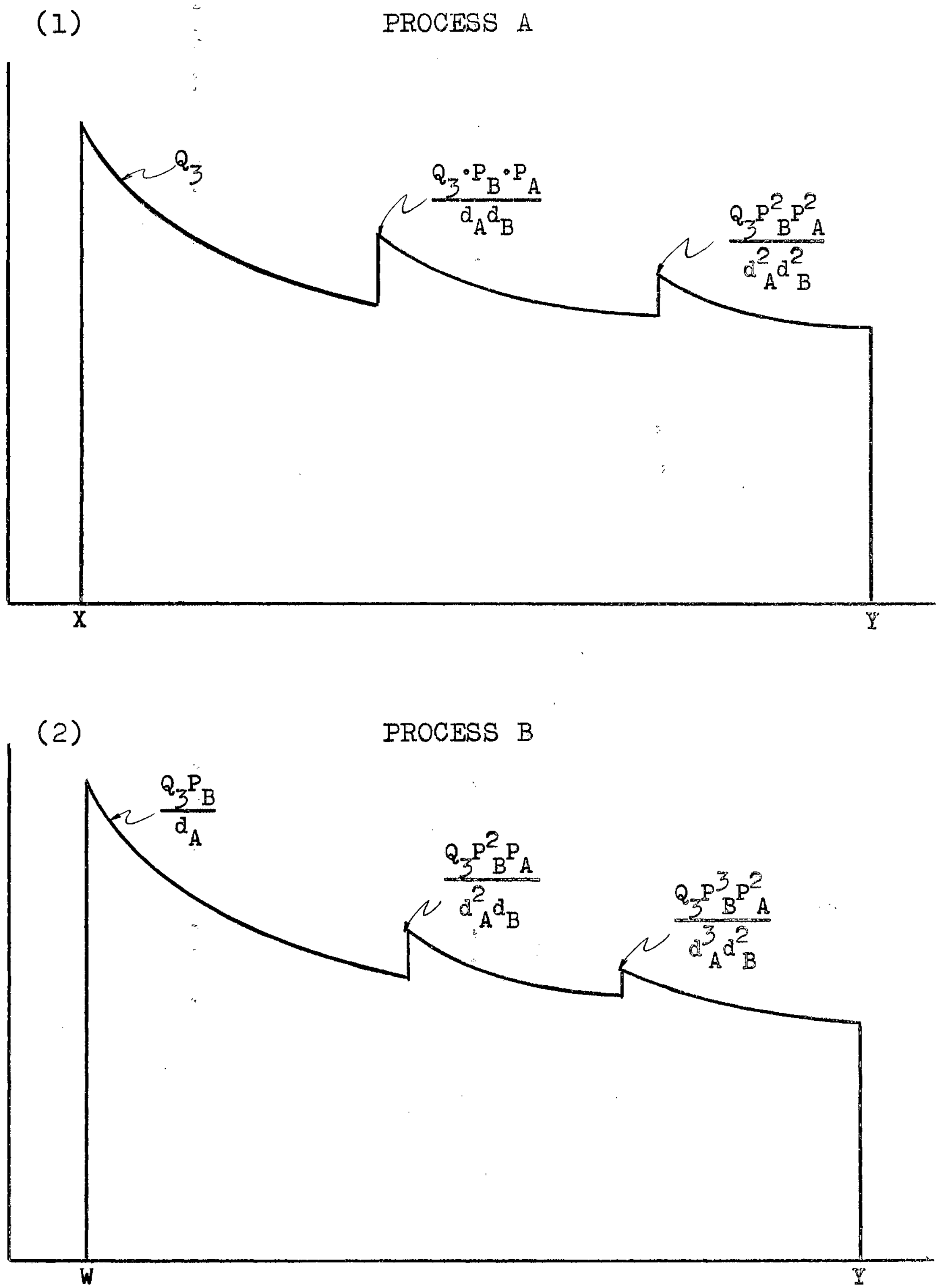


Figure 7. Illustration of How an Arbitrary Production Quantity Q_n Fixes the Entire Production Schedule for a Three Stage Process When Shortages and Overages are not Allowed

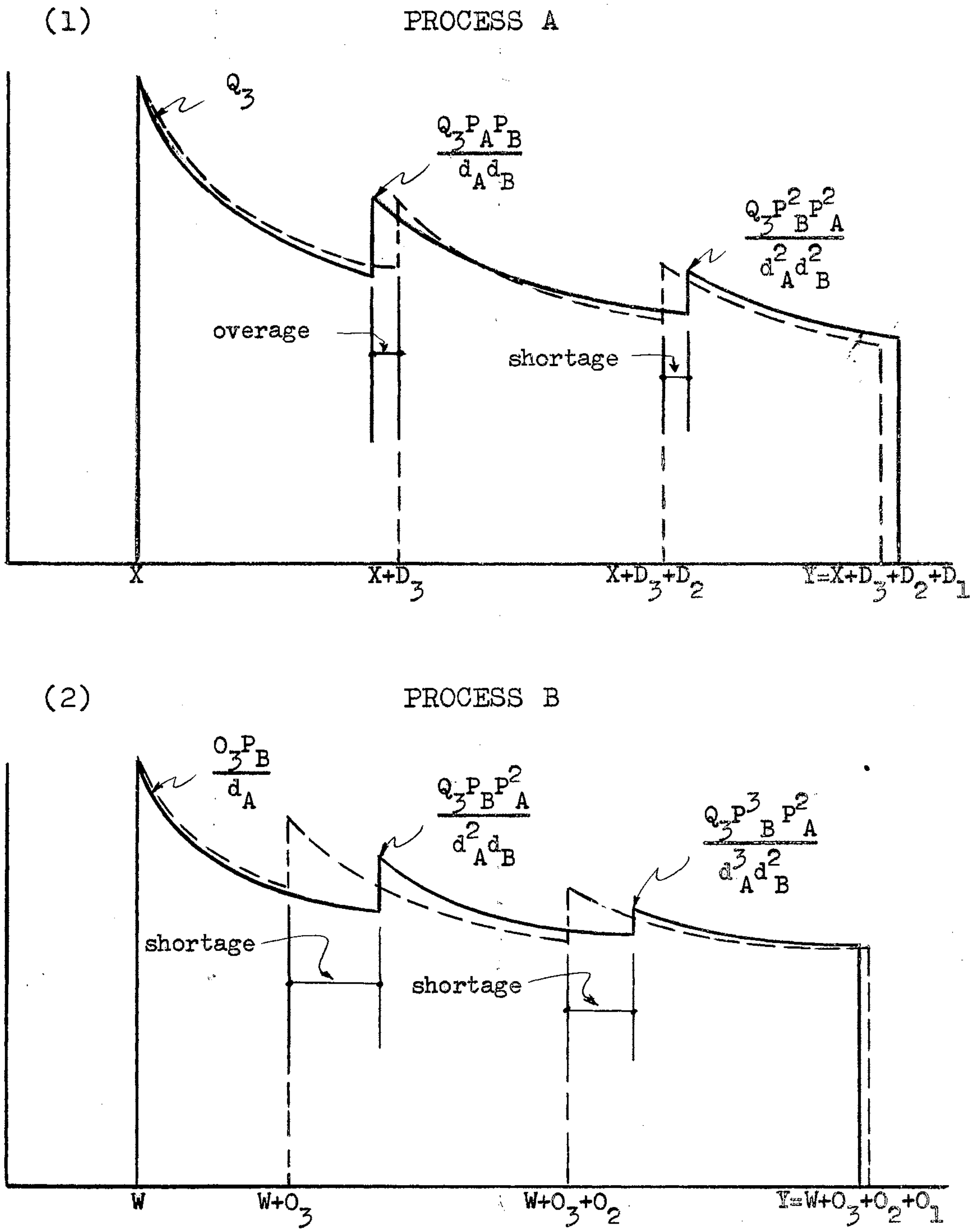


Figure 8. Illustration of How Loss of Learning and Storage of Finished Items are Incorporated With Shortages and Overages to Obtain an Over-all Balance of Cost

-F dollars per unit for units short of demand for Process B.

The symbol $\delta_i (i = 1, 2)$ is defined as follows:

$$\delta_1 = G \text{ if } (D_i - Q_i) > 0 \quad i = 1, 2, \ldots, n$$

$$= -D \text{ if } (D_i - Q_i) < 0 \quad i = 1, 2, \ldots, n$$

$$\delta_2 = E \text{ if } \left(O_i - \left[\frac{Q_i \cdot P_B}{d_A}\right]\right) > 0 \quad i = 1, 2, \ldots, n$$

$$= -F \text{ if } \left(O_i - \left[\frac{Q_i \cdot P_B}{d_A}\right]\right) < 0 \quad i = 1, 2, \ldots, n$$

The formulation of the algorithm is begun by finding one stage policies.

$$f_1(X, W, Q_1) = \left\{\delta_1(D_1 - Q_1) + \delta_2\left(O_1 - \left[\frac{Q_1 \cdot P_B}{d_A}\right]\right) + S_A + S_B + \frac{C \cdot D_1^2}{2d_A} + \frac{H \cdot O_1^2}{2d_B} + \sum_{X-M}^{X+D_1-M} KX^n + \sum_{W-N}^{W+O_1-N} LW^n\right\}.$$

Where L and N are defined for the learning function of Process B in a similar manner that K and M are for the learning function of Process A, and the quantity $\left[\frac{Q_1 \cdot P_B}{d_A}\right]$ is taken to mean the greatest integer less than or equal to $\frac{O_1 \cdot P_B}{d_A}$.

The one stage policies are used to obtain the optimal two stage policies.

$$f_2(X, W, Q_2) = \min_{\substack{0 \leq D_2 \leq Y-X \\ 0 \leq O_2 \leq Y-W}} \left\{\delta_1(D_2 - Q_2) + \delta_2\left(O_2 - \left[\frac{Q_2 \cdot P_B}{d_A}\right]\right) + \right.$$

$$+ S_A + S_B + \frac{C \cdot D_2^2}{2d_A} + \frac{H \cdot O_2^2}{2d_B} + \sum_{X-M}^{X+D_2-M} KX^n + \sum_{W-N}^{W+O_2-N} LW^n + f_1\left(X+D_2-M, W+O_2-N, \left[\frac{Q_2 \cdot P_A P_B}{d_A d_B}\right]\right)\Bigg\}$$

which are now used to obtain the optimal three stage policies as follows:

$$f_3(X, W, Q_3) = \min_{\substack{0 \leq D_3 \leq Y-X \\ 0 \leq O_3 \leq Y-W}} \Bigg\{ \delta_1(D_3 - Q_3) + \delta_2\left(O_3 - \left[\frac{Q_A P_B}{d_A}\right]\right) + S_A + S_B + \frac{C \cdot D_3^2}{2d_A} + \frac{H \cdot O_3^2}{2d_B} + \sum_{X-M}^{X+D_3-M} KX^n + \sum_{W-N}^{W+O_3-N} LW^n + f_2\left(X+D_3-M, W+O_3-N, \left[\frac{Q_3 \cdot P_A P_B}{d_A d_B}\right]\right)\Bigg\}.$$

The general form of the algorithm is now stated as follows:

$$f_n(X, W, Q_n) = \min_{\substack{0 \leq D_n \leq Y-X \\ 0 \leq O_n \leq Y-W}} \Bigg\{ \delta_1(D_n - Q_n) + \delta_2\left(O_n - \left[\frac{Q_n P_B}{d_A}\right]\right) + S_A + S_B + \frac{C \cdot D_n^2}{2d_A} + \frac{H \cdot O_n^2}{2d_B} + \sum_{X-M}^{X+D_n-M} KX^n + \sum_{W-N}^{W+O_n-N} LW^n + f_{n-1}\left(X+D_n-M, W + O_n - N, \left[\frac{Q_n P_A P_B}{d_B d_A}\right]\right)\Bigg\}.$$

The process is continued until $f_{n-1}(1, 1, Q_n) \leq f_n(1, 1, Q_n)$ for each value of Q_n. The value that will be of interest is then the $\min_{Q_n}\{f_n(1, 1, Q_n)\}$. This value is associated with the desired n stage schedule for both processes, which is the optimal policy in light of the cost considered.

It is noted that if the value of Q_n is such that the schedule determined by it does not span the planning period for the two processes, then this lends weight to the shortage and overage cost, since the actual schedule does span the period and is compared stage-by-stage to the schedule predetermined By Q_n. In effect, this would rule out a particular value of Q_n for the $\min_{Q_n}\{f_n(1, 1, Q_n)\}$. For this reason, the only values of Q_n that are instrumental in determining the optimal schedule for the two processes are those values which cause the predetermined schedule to closely approximate the planning period for both processes.

CHAPTER VI

SUMMARY AND CONCLUSIONS

This dissertation employs an analytical technique called dynamic programming to make possible the optimization of n-state production systems under the effect of learning with variable regression. Classical optimization procedures, except direct enumeration, have been unsuccessful in dealing with the learning effect where variable regression is present. Four specific production situations are modeled to indicate the range of applicability of the algorithm.

Chapter II deals with a constant demand production situation with setup cost and storage cost incorporated into the model as well as the cost due to loss of learning. The model presented in this chapter permits an optimum n-stage production policy on behalf of these production costs.

In Chapter III the same basic model presented in Chapter II is expanded to include stochastic demand on the production item. The parameters of the model are again setup cost, storage cost, and cost due to loss of learning. In this model it is assumed that the demand distribution is known. Certain examples of distributions that might be applicable as demand distributions or approximations of demand distributions are presented. These examples serve to simplify the actual numerical work of the algorithm in this chapter.

Chapter IV expands the basic constant demand model to include raw materials supply. The new cost introduced to the optimizing process by this expansion include order cost and inventory storage cost. The algorithm presented in this chapter produces an optimum n-stage production policy simultaneously with an optimum n-stage inventory schedule for a single commodity, raw material. Both of these policies span a predetermined planning period.

Finally, in Chapter V a model is presented to obtain optimum n-stage production policies for alternate production facilities. In the derivation of this model, overage and shortage costs are incorporated so the optimum schedule for the two production facilities includes the desired balance of loss of learning cost on two facilities - two different setup costs, storage cost for two different production items, and at last overage and shortage costs for two production items. In this derivation the progress function of the two processes need not be the same.

Each model developed encompasses a portion of the total operational system; that is, each model defines a system bounded by the assumptions stated. The underlying common link in the models developed is the manufacturing progress function, as derived in Appendix A, with variable regression. The variable regression is accomplished by moving back a constant M units on the abscissa each time the process is interrupted. This causes a variable amount of regression on the ordinate.

Appendix B presents a non-computer example to illustrate the actual mechanics of the procedure presented in Chapter II. The procedures presented in the remaining chapters are carried out in a similar manner. The example in Appendix B is a simple case for a planning period of five units. For the specific parameters chosen to work the example, the result was a three-stage policy with two units each in the third and second stages and one unit in the first stage.

Appendix C presents computer solutions to a number of examples, the first being the same as the example of Appendix B. Subsequent examples are presented for more realistic planning periods. The computer program presented may be used for further experimentation.

A general conclusion from this investigation is that dynamic programming provides a feasible means for solution of an n-stage production policy under a learning effect with variable regression.

A special feature of the technique presented here is the sensitivity analysis that is inherent in it. For example should it be decided to shorten the length of the planning period, optimal policy structures have already been determined for the shortened period. In this same manner for a longer planning period, it is not necessary to rework the entire problem.

After working an example by hand using integral approximations, it was noted that while a computer solution using summation notation results in a generally lower optimum policy cost, the policy structures are exactly the same. The numerical computations verify the intuitive conclusion that with high storage cost relative to direct labor cost more production stages result that when the converse is true.

SELECTED BIBLIOGRAPHY

(1) Bellman, R. Dynamic Programming. Princeton, New Jersey: Princeton University Press, 1957.

(2) Bellman, R., and S. Dreyfus. Applied Dynamic Programming. Princeton, New Jersey: Princton University Press, 1962.

(3) Fabrycky, W. J., and P. E. Torgersen. Operations Economy. Englewood Cliffs, New Jersey: Printice-Hall, Inc. (being prepared under contract).

(4) Thuesen, H. G., and W. J. Fabrycky. Engineering Economy, 3rd ed. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1964.

(5) Cochran, E. B. "New Concepts of the Learning Curve," Journal of Industrial Engineering, Vol. XI, No. 4, 1960, p. 317.

(6) Vazsonyi, Andrew. Scientific Programming in Business and Industry. New York: John Wiley and Sons, 1958.

(7) Carr, C., and C. Howe. Quantitative Decision Procedures in Management and Economics. New York: McGraw-Hill, 1964.

(8) Mood, A. Introduction to the Theory of Statistics. New York, McGraw-Hill, 1950.

APPENDIXES

APPENDIX A

THE MANUFACTURING PROGRESS FUNCTION

APPENDIX A

THE MANUFACTURING PROGRESS FUNCTION

The manufacturing progress function rests on the assumption that the direct labor man-hours required to complete a unit of production decreases by a constant percentage each time the production quantity is doubled. This empirical relationship between direct labor man-hours per unit and the quantity of units produced was first noted and accepted by the aircraft industry. It was recognized that the labor hours required to build an airplane declined quite regularly as more such airplanes were built.

A typical rate of improvement in the aircraft industry is twenty per cent between doubled quantities. This is expressed as an eighty per cent progress curve and means that the direct labor man-hours required to build the second aircraft are eighty per cent of those required to build the first. The fourth aircraft requires eighty per cent of the man-hours that the second required, the eighth requires eighty per cent of the fourth, and so forth. Since the production quantity is doubled in each case for the given percentage improvement to occur, the rate of improvement, in relation to time, is actually diminishing.

The development of the unit formula (3) for a progress function is begun by assuming that the following relationship is applicable: <u>As the quantity of units produced is doubled, the number of direct labor man-hours required to produce each of these units is reduced by a constant percentage</u>.

Let:

X = the number of units produced, counting from the first unit.

Y_X = the number of direct labor man-hours required to produce the X^{th} unit.

K = the number of direct labor man-hours required to produce the first unit. ($Y_X = K$ for $X = 1$, the 1^{st} unit.)

N = the per cent improvement expressed as a positive rather than a negative slope, e.g., for an eighty per cent progress curve, $N = 0.80$.

$n = \log_{10} N / \log_{10} 2$.

A general equation for Y and X may be developed as follows:

$$Y_X = KN^0 \quad \text{where} \quad X = 2^0$$

$$Y_X = KN^1 \quad \text{where} \quad X = 2^1$$

$$Y_X = KN^2 \quad \text{where} \quad X = 2^2$$

$$Y_X = KN^3 \quad \text{where} \quad X = 2^3$$

and thus,

$$Y_X = KN^t \quad \text{where} \quad X = 2^t.$$

Taking the common logarithms of both equations gives

$$\log Y_X = \log K + t \log N \text{ and } \log X = t \log 2.$$

Solving both equations for t,

$$t = \frac{\log Y_X - \log K}{\log N} \quad \text{and } t = \frac{\log X}{\log 2}.$$

Equating the result gives

$$\frac{\log Y_X - \log K}{\log N} = \frac{\log X}{\log 2}$$

$$(\log Y_X - \log K)(\log 2) = \log X \log N$$

$$\log Y_X - \log K = \frac{\log X \log N}{\log 2}.$$

By definition, $n = \frac{\log N}{\log 2}$; therefore,

$$\log Y_X - \log K = n \log X$$

$$\log \frac{Y_X}{K} = n \log X.$$

Taking the antilog of both sides,

$$Y_X = KX^n.$$

This progress function is used for the analysis presented in this dissertation. It is mentioned, however, that this presents no loss of generality as there is nothing unique about this function that restricts the techniques applied here to it in particular.

APPENDIX B

A NON-COMPUTER SOLUTION

APPENDIX B

A NON-COMPUTER SOLUTION

The following example serves to illustrate the manner in which the algorithm described in Chapter II is carried out. For ease of computation, integral approximations are used instead of summation. The planning period is chosen to be five units long and an eighty-one per cent exponential manufacturing progress function is assumed.

The variables previously described in Chapter II are assumed to have the following values: $S = 10$, $C = 10$, $K = 2$, $M = 1$, $d = 1$, $n = -.3$. The optimizing process is begun by finding the optimal two stage policies from each point in the planning period to the end of the planning period. The only quantity that is manipulated is Q_2. The quantity Q_1 is automatically set when Q_2 is chosen.

$$f_2(4) = \min_{0 \leq Q_2 \leq 2} \left\{ C_4(Q_2) + f_1(4 + Q_2 - 1) \right\}.$$

The quantity $C_4(Q_2)$ is the cost of producing and storing Q_2 units beginning at unit 4 until they are sold. It is found in the following manner:

$$C_4(Q_2) = S + \frac{C \cdot Q_2^2}{2 \cdot d} + \int_{X-M}^{X+Q_2-M} KX^n dx$$

$$= 10 + \frac{10 \cdot Q_2^2}{2} + 2\int_{4-1}^{4+Q_2-1} X^{-.3}dX.$$

The quantity $f_1(4+Q_2-1)$ is the cost of a one stage policy from the point $(4+Q_2)$ to the end of the planning period. It is found as follows:

$$f_1(4+Q_2-1) = S + \frac{C \cdot Q_1^2}{2 \cdot d} + \int_{X-M}^{X+Q_1-M} KX^n dX$$

but $X = (4 + Q_2 - 1)$; therefore,

$$f_1(4+Q_2-1) = S + \frac{C \cdot Q_1^2}{2 \cdot d} + \int_{4+Q_2-2}^{4+Q_2+Q_1-2} KX^{-.3}dX$$

$$= 10 + \frac{10 \cdot Q_1^2}{2 \cdot d} + 2\int_{4+Q_2-2}^{4+Q_2+Q_1-2} X^{-.3}dX.$$

$f_2(4)$ is now found by manipulating Q_2.

$$f_2(4) = \min \begin{cases} \underline{Q_2 = 1} \\ 10 + 5.1^2 + 2\int_{4-1}^{4+1-1} X^{-.3}dX \\ \qquad + 10 + 5.1^2 + 2\int_{4+1-2}^{4+1+1-2} X^{-.3}dX = 32.8 \\ \underline{Q_2 = 2} \\ 10 + 5.2^2 + 2\int_{4-1}^{4+2-1} X^{-.3}dX \\ \qquad + 10 + 5.0 + 2\int_{4+2-2}^{4+2-2} X^{-.3}dX = 42.66 \end{cases}$$

$$f_2(4) = \min \begin{cases} \underline{Q_2 = 0} \\ 10 + 5.0^2 + 2\int_{4-1}^{4+0-1} X^{-.3}dX \\ \qquad + 10 + 5.2^2 + 2\int_{4+0-2}^{4+0+2-2} X^{-.3}dX > 40 \end{cases}$$

This means that $f_2(4) = 32.8$, $Q_2 = 1$, $Q_1 = 1$.

The two stage policies are now found from the point $X = 3$, or the third unit of production until the end of the planning period.

$$f_2(3) = \min_{0 \leq Q_2 \leq 3} \left\{ C_3(Q_2) + f_1(3+Q_2-1) \right\}.$$

$$f_2(3) = \min \begin{cases} \underline{Q_2 = 0} \\ 10 + 0 + \int_{3-1}^{3+0-1} f(X)dX + 10 + 5.3^2 \int_{3-2}^{3+0+3-2} f(X)dX = 69.2 \\ \underline{Q_2 = 1} \\ 10 + 5.1^2 + \int_{3-1}^{3+1-1} f(X)dX + 10 + 5.2^2 + \int_{3+1-2}^{3+1+2-2} f(X)dX \\ \qquad = 49.60 \\ \underline{Q_2 = 2} \\ 10 + 5.2^2 + \int_{3-1}^{3+2-1} f(X)dX + 10 + 5.1 + \int_{3+2-2}^{3+2+1-2} f(X)dX \\ \qquad = 49.26 \end{cases}$$

$$f_2(3) = \min\begin{cases} \underline{Q_2 = 3} \\ 10 + 5.3^2 + \int_{3-1}^{3+3-1} f(X)dX + 10 + 5.0 + \int_{3+3-2}^{3+3+0-2} f(X)dX \\ > 50. \end{cases}$$

This means that $f_2(3) = 49.26$, $Q_2 = 2$, $Q_1 = 1$.

The same procedure is carried out from the points $X = 2$ and $X = 1$, and yields the results

$$f_2(2) = 65.0 \qquad Q_2 = 2 \qquad Q_1 = 2$$

$$f_2(1) = 91.30 \qquad Q_2 = 3 \qquad Q_1 = 2.$$

$f_2(1)$ is now compared to $f_1(1)$ to see if the additional stage is feasible.

$$f_1(1) = 10 + 5.5^2 + \int_{1}^{1+5} f(X)dX = 141.85.$$

Since $f_2(1) < f_1(1)$, a two stage policy is more desirable than a one stage policy. The cost of a three stage policy is now computed from each point in the planning period.

$$f_3(3) = \min_{0 \leq Q_3 \leq 3}\left\{C_3(Q_3) + f_2(3+Q_3-1)\right\}.$$

$$f_3(3) = \min\begin{cases} \underline{Q_3 = 0} \\ 10 + 5.0 + \int_{3-1}^{3+0-1} f(X)dX + 10 + 5.2^2 + \int_{3+0-2}^{3+0+2-2} f(X)dX \\ \quad + 10 + 5.1^2 \int_{3+0+2-3}^{3+0+2+1-3} f(X)dX = 59.6 \end{cases}$$

$$f_3(3) = \min\begin{cases} \underline{Q_3 = 1} \\ 10 + 5.1^2 + \int_{3-1}^{3+1-1} f(X)dX + 10 + 5.1^2 + \int_{3+1-2}^{3+1+1-2} f(X)dX \\ \quad + 10 + 5.1^2 + \int_{3+0+2-3}^{3+0+2+1-3} f(X)dX = 49.6 \\ \underline{Q_3 = 2} \\ 10 + 5.2^2 + \int_{3-1}^{3+2-1} f(X)dX + 10 + 5.1 + \int_{3-2}^{3+2+1-2} f(X)dX \\ \quad + 10 + 5.1^2 + \int_{3+2+1-3}^{3+2+1-3} f(X)dX > 50. \end{cases}$$

This means that $f_3(3) = 49.6$, $Q_3 = 1$, $Q_2 = 1$, $Q_1 = 1$.

The optimum three stage policy is now computed from the point X = 2 to the end of the planning period.

$$f_3(2) = \min_{0 \le Q_3 \le 4} \left\{ C_2(Q_3) + f_2(2+Q_3-1) \right\}.$$

This is found to be $f_3\ (2) = 66.35$, $Q_3 = 2$, $Q_2 = 1$, $Q_1 = 1$.

It is noted that the only quantity that is manipulated is Q_3. For instance, in the preceding calculations when Q_3 is set equal to 1 this sets X = 3 in the planning period and the structure of the optimal two stage policies from X = 3 determines that $Q_2 = 2$ and $Q_1 = 1$.

In this same manner $f_3(1)$ is determined.

$$f_3(1) = 82.65,\ Q_3 = 2,\ Q_2 = 2,\ Q_1 = 1.$$

Since $f_3(1) < f_2(1)$, this means that a third stage is feasible.

The four stage policies are computed in a similar manner and found to be:

$$f_4(2) = 66 \qquad Q_4 = 1,\ Q_3 = 1,\ Q_2 = 1,\ Q_1 = 1.$$

$$f_4(1) = 83.55 \qquad Q_4 = 2,\ Q_3 = 1,\ Q_2 = 1,\ Q_1 = 1.$$

Since $f_4(1) > f_3(1)$, this means that an optimal policy spanning the production period consist of three stages and is now determined. The policy is $Q_3 = 2$, $Q_2 = 2$, $Q_1 = 1$, and $f_3(1) = 82.65$.

APPENDIX C

SOLUTIONS BY A DIGITAL COMPUTER

APPENDIX C

SOLUTIONS BY A DIGITAL COMPUTER

A Fortran program for the algorithm derived in Chapter II is presented in this Appendix. The program is compatible to any IBM 1410 digital computer that uses the PR-155 operating system. This program was written for an eighty-one per cent progress function. To revise it to a specific function entails changing the statements (AX**3) to (AX**n) where n reflects the desired manufacturing progress function. Also, the constant amount M that is moved back on the abscissa each time the process is interrupted is chosen to be one in this program. This may be revised to a suitable M by changing the limits of the do loop after statement 00041 from ISA = Q(2,X)-V-1 to ISA = Q(2,X)-V-M and 1ST = Q(2,X)+Q(I,INT)-V-1 to IST = Q(2,X)+Q(I,INT)-V-M. The statements 1SA and 1ST after statement 00039 are changed in a similar manner.

The statements ISA = X-IV and 1UP = A(N,X)-V after statement 00021 may be changed to ISA = X-IV(M) and 1UP = A(N,X)-V(M) to be compatible to a chosen M. Also, after statement 0051, ISA = K1+1 and IST = IST+X-(K1+1) may be changed to ISA = (K1+1)M and IST = IST+X-(K1+1)M for a general M.

Format statement 00100 refers to the input and indicates ten columns with four decimal places for each parameter. Format statement 00300 is a carriage control statement and brings the carriage back to the beginning of a page for the print out of each solution. Format statements 00301, 00302, 00303 are print out statements and result in a print out of the type shown in the following examples. Format statement 00310 results in the print out of parameter values preceding each solution.

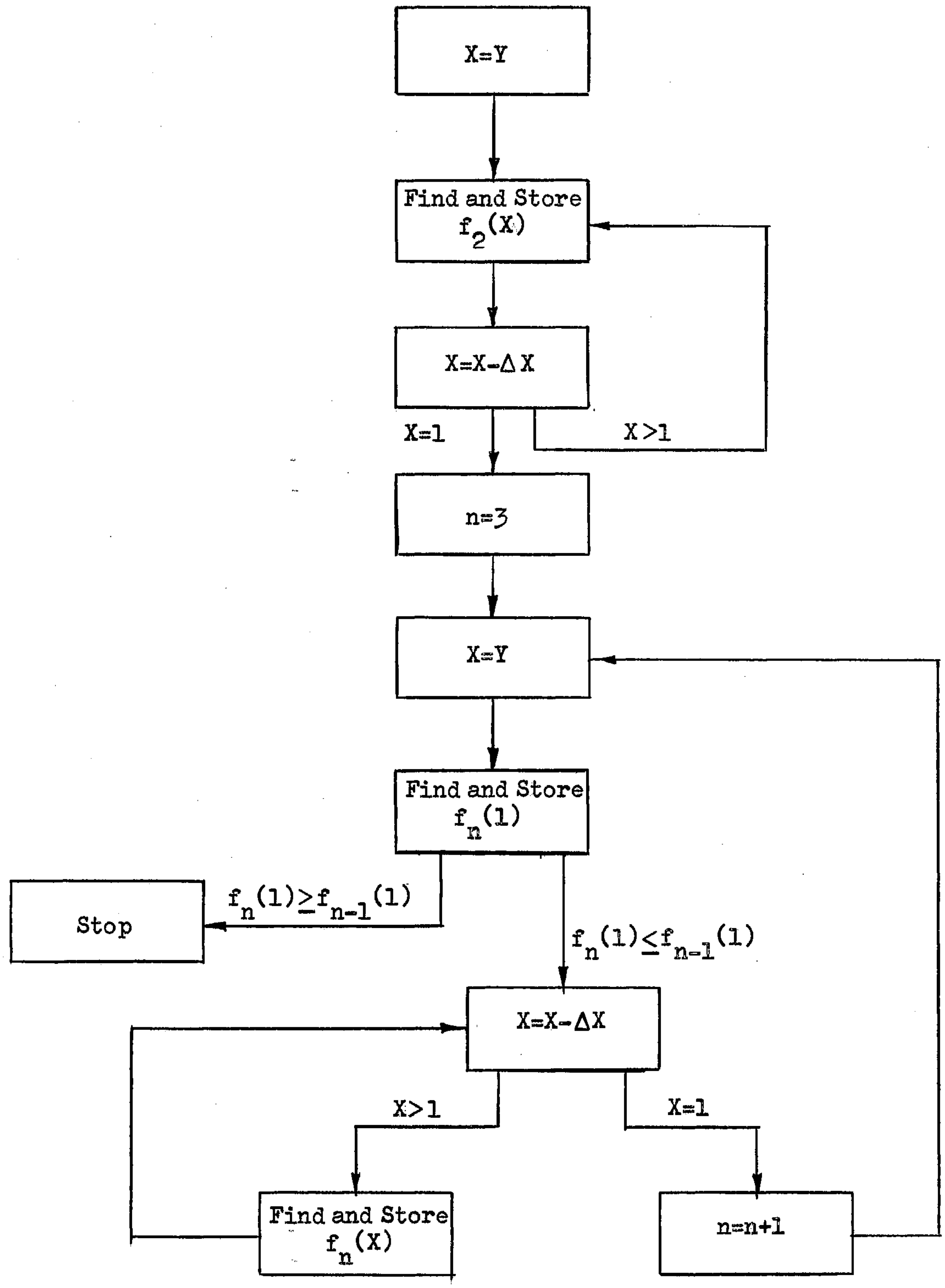
The data is punched on standard eighty column IBM cards in the following manner:

Variable	Columns
S	1-10
C	11-20
Y	21-30
D	31-40
K	41-50.

Four decimal places are allowed in each field and a decimal place occupies one position.

The following illustration is a flow diagram that demonstrates the general logic of the program.

Four computer solutions to examples are presented, with Example One being the non-computer example presented in Appendix B solved by the program in this Appendix. The particular set of variables to which a solution applies is listed above the columns headed F(NX), Q(NX), Q1.



Flow Diagram of Algorithm

```
              FORTRAN LISTING                  1410-FO-970
      INTEGERX,AY,XNQ,XJ
      REALK
      DIMENSIONY2(6,11),Y3(6,11,7),A(6,11),Q(6,11)
00100 FORMAT(F10.4,F10.4,F10.4,F10.4,F10.4)
00300 FORMAT(1H1)
00301 FORMAT(10X,I6,3X,1PE20.8)
00302 FORMAT(10X,I6,3X,1P2E20.8)
00303 FORMAT(10X,I6,3X,1P3E20.8)
00310 FORMAT(3H S=,F8.2,3H C=,F8.2,3H Y=,F8.2,3H D=,F8.2,3H K=,F8.2)
00312 FORMAT( 5X,2HN=,I4)
00313 FORMAT(//15X,1HX,13X,5HF(NX),15X,5HQ(NX),15X,3H Q1,/)
01000 WRITE(3,300)
      READ(1,100)S,C,Y,D,K
      IF(S.EQ.999.)CALLEXIT
      WRITE(3,310)S,C,Y,D,K
      WRITE(3,313)
      Y2(1,1)=S+((C*(Y-1.)**2)/(2.*D))+(K/.632)*((Y**.632)-1.)
      N=2
      WRITE(3,312)N
      X=Y
00005 J=0
      Y2(2,X)=10000.0
00001 A(2,X)=J
      AX=X
      A(1,X)=Y-(A(2,X)+AX)
      V=1.
      IF(X.EQ.1)V=0.0
      IV=V
      ISA=X-IV
      IST=A(N,X)-V
      IST=IST+X
      SUM=0.0
      IX=X
      DO39X=ISA,IST
      AX=X
      SUM=SUM+(1./(AX**3))
00039 CONTINUE
      X=IX
      SUM1=0.0
      ISA=A(2,X)-V-1.
      ISA=ISA+X
      IST=A(2,X)+A(1,X)-V-1.
      IX=X
      DO40X=ISA,IST
      AX=X
      SUM1=SUM1+(1./(AX**3))
00040 CONTINUE
      X=IX
      Y3(2,X,J+1)=S+(C*(A(2,X)*A(2,X))/(2.*D))+K*SUM+S+(C*(A(1,X)*A(1,X))/(2.*D))+K*SUM1
      IF(Y3(2,X,J+1).GE.Y2(2,X))GOTO11
      Y2(2,X)=Y3(2,X,J+1)
      Q(2,X)=A(2,X)
```

```
           FORTRAN LISTING                1410-FO-970
      INT=Q(2,X)
      INT=INT+X
      Q(1,INT)=A(1,INT)
      AY=Y
00004 IF(J-(AY-X).GE.0)GOTO8
      J=J+1
      GOTO1
00011 Q(2,X)=A(2,X)-1.
      AX=X
      INT=Q(2,X)
      INT=INT+X
      Q(1,INT)=Y-(Q(2,X)+AX)
00008 V=1.
      IF(X.EQ.1)V=0.0
      IV=V
      ISA=X-IV
      IST=Q(2,X)-V
      IST=IST+X
      SUM=0.0
      IX=X
      DO41X=ISA,IST
      AX=X
      SUM=SUM+(1./(AX**3))
00041 CONTINUE
      X=IX
      SUM1=0.0
      ISA=Q(2,X)-V-1.
      ISA=ISA+X
      INT=Q(2,X)
      INT=INT+X
      IST=Q(2,X)+Q(1,INT)-V-1.
      IX=X
      DO43X=ISA,IST
      AX=X
      SUM1=SUM1+(1./(AX**3))
00043 CONTINUE
      X=IX
      Y2(2,X)=S+((C*(Q(2,X)*Q(2,X)))/(2.*D))+K*SUM+S+((C*(Q(1,INT)*Q(1,INT)))/(2.*D))+K*SUM1
      WRITE(3,303)X,Y2(2,X),Q(2,X),Q(1,INT)
      X=X-1
      IF(X.LE.0)GOTO12
      GOTO5
00012 IF(Y2(2,1).LT.Y2(1,1))GOTO15
      WRITE(3,301)X,Y2(1,1)
      GOTO1000
00015 N=N+1
      WRITE(3,312)N
      X=Y
00024 J=0
      Y2(N,X)=10000.0
00021 A(N,X)=J
      IXX=X
```

FORTRAN LISTING 1410-FO-970

```
      SUMBN=0.0
      NA=N
      XJ=X+J
      V=1.
      IF(X.EQ.1)V=0.0
      SUM=0.0
      IV=V
      ISA=X-IV
      IUP=A(N,X)-V
      IST=IUP+X
      IX=X
      DO37X=ISA,IST
      AX=X
      SUM=SUM+(1./(AX**3))
00037 CONTINUE
      X=IX
      BN=S+(C*(A(N,X)*A(N,X)))/(2.*D)+K*SUM
      SUMBN=SUMBN+BN
      INT=A(N,X)
      X=X+INT
      N=N-1
      IK=K
      K1=IV
00051 ISA=K1+1
      ISA=X-ISA
      SUM=0.0
      IX=X
      IST=Q(N,X)
      IST=IST+X-(K1+1)
      DO46X=ISA,IST
      AX=X
      SUM=SUM+(1./(AX**3))
00046 CONTINUE
      X=IX
      BN=S+((C*(Q(N,X)*Q(N,X)))/(2.*D))+K*SUM
      SUMBN=SUMBN+BN
      INT=Q(N,X)
      X=X+INT
      N=N-1
      K1=K1+1
      IF(N.GE.1)GOTO51
      N=NA
      X=IXX
      Y3(N,X,J+1)=SUMBN
      IF(Y3(N,X,J+1).GE.Y2(N,X))GOTO18
      Y2(N,X)=Y3(N,X,J+1)
      Q(N,X)=A(N,X)
      AY=Y
      IF(J-(AY-X).GE.0)GOTO20
      J=J+1
      GOTO21
00018 Q(N,X)=A(N,X)-1.
```

```
             FORTRAN LISTING              1410-FO-970
00020 CONTINUE
      WRITE(3,302)X,Y2(N,X),Q(N,X)
      IF(X-1.LE.0)GOTO23
00022 X=X-1
      GOTO24
00023 IF(Y2(N,1).LT.Y2(N-1,1))GOTO25
      WRITE(3,302)X,Y2(N-1,1),Q(N-1,1)
      GOTO1000
00025 AY=Y
      IF(N-(AY-1).LT.0)GOTO28
      WRITE(3,302)X,Y2(N,1),Q(N,1)
      GOTO1000
00028 N=N+1
      WRITE(3,312)N
      X=Y
      GOTO24
      END
```

EXAMPLE ONE

S= 10.00 C= 10.00 Y= 6.00 D= 1.00 K= 2.00

	X	F(NX)	Q(NX)	Q1
N= 2				
	6	2.00849890E 01	.00000000E-00	.00000000E-00
	5	2.51162390E 01	1.00000000E 00	.00000000E-00
	4	3.01793980E 01	1.00000000E 00	1.00000C00E 00
	3	4.54293980E 01	2.00000000E 00	1.00000000E 00
	2	6.25740740E 01	2.00000000E 00	2.00000000E 00
	1	8.74606480E 01	3.00000000E 00	2.00000000E 00
N= 3				
	6	3.01213240E 01	.00000000E-00	
	5	3.51525740E 01	1.00000000E 00	
	4	4.02847220E 01	1.00000000E 00	
	3	4.59722220E 01	1.00000000E 00	
	2	6.29722220E 01	2.00000000E 00	
	1	7.77847220E 01	2.00000000E 00	
N= 4				
	6	4.03713240E 01	.00000000E-00	
	5	4.54025740E 01	1.C0000000E 00	
	4	5.05347220E 01	1.00000000E 00	
	3	5.62222220E 01	1.00000000E 00	
	2	6.90000000E 01	1.0C000000E 00	
	1	7.82962960E 01	2.00000000E 00	
	1	7.77847220E 01	2.00000000E 00	

EXAMPLE TWO

S= 10.00 C= 5.00 Y= 11.00 D= 1.00 K= 25.00

	X	F(NX)	Q(NX)	Q1
N= 2				
	11	2.00592930E 01	.00000000E-00	.00000000E-00
	10	2.25935860E 01	1.00000000E 00	.00000000E-00
	9	2.51319490E 01	1.00000000E 00	1.00000000E 00
	8	3.26946000E 01	1.00000000E 00	2.00000000E 00
	7	4.03103410E 01	2.00000000E 00	2.00000000E 00
	6	5.30043670E 01	2.00000000E 00	3.00000000E 00
	5	6.58949920E 01	3.00000000E 00	3.00000000E 00
	4	8.43209170E 01	4.00000000E 00	3.00000000E 00
	3	1.05073030E 02	4.00000000E 00	4.00000000E 00
	2	1.52645910E 02	5.00000000E 00	4.00000000E 00
	1	1.75229030E 02	5.00000000E 00	5.00000000E 00
N= 3				
	11	3.01081210E 01	.00000000E-00	
	10	3.26424140E 01	1.00000000E 00	
	9	3.52150700E 01	1.00000000E 00	
	8	3.78651420E 01	1.00000000E 00	
	7	4.54808830E 01	2.00000000E 00	
	6	5.33147090E 01	2.00000000E 00	
	5	6.13324470E 01	2.00000000E 00	
	4	7.47583730E 01	3.00000000E 00	
	3	9.07853700E 01	3.00000000E 00	
	2	1.29011290E 02	3.00000000E 00	
	1	1.45692540E 02	4.00000000E 00	
N= 4				
	11	4.01810070E 01	.00000000E-00	
	10	4.27153000E 01	1.00000000E 00	
	9	4.52879560E 01	1.00000000E 00	
	8	4.79380280E 01	1.00000000E 00	
	7	5.07545080E 01	1.00000000E 00	
	6	5.84545080E 01	2.00000000E 00	
	5	6.64722460E 01	2.00000000E 00	
	4	7.54272820E 01	2.00000000E 00	
	3	8.70530920E 01	2.00000000E 00	
	2	1.24553080E 02	3.00000000E 00	
	1	1.36699910E 02	3.00000000E 00	
N= 5				
	11	5.02967470E 01	.00000000E-00	
	10	5.28310400E 01	1.00000000E 00	
	9	5.54036960E 01	1.00000000E 00	
	8	5.80537680E 01	1.00000000E 00	
	7	6.08702480E 01	1.00000000E 00	
	6	6.40787000E 01	1.00000000E 00	
	5	7.19693250E 01	2.00000000E 00	
	4	8.06701350E 01	2.00000000E 00	
	3	9.22959450E 01	2.00000000E 00	
	2	1.28806010E 02	2.00000000E 00	
	1	1.36104010E 02	2.00000000E 00	
N= 6				
	11	6.04967470E 01	.00000000E-00	
	10	6.30310400E 01	1.00000000E 00	
	9	6.56036960E 01	1.00000000E 00	
	8	6.82537680E 01	1.00000000E 00	
	7	7.10702480E 01	1.00000000E 00	
	6	7.42787000E 01	1.00000000E 00	
	5	7.85437500E 01	1.00000000E 00	
	4	8.69696750E 01	2.00000000E 00	
	3	9.83206000E 01	2.00000000E 00	
	2	1.34280890E 02	2.00000000E 00	
	1	1.41346870E 02	2.00000000E 00	
	1	1.36104010E 02	2.00000000E 00	

EXAMPLE THREE

S= 10.00 C= 5.00 Y= 11.00 D= 1.00 K= 50.00

	X	F(NX)	Q(NX)	Q1
N= 2				
	11	2.01185870E 01	.00000000E-00	.00000000E-00
	10	2.26871740E 01	1.00000000E 00	.00000000E-00
	9	2.52638990E 01	1.00000000E 00	1.00000000E 00
	8	3.28892000E 01	1.00000000E 00	2.00000000E 00
	7	4.06206820E 01	2.00000000E 00	2.00000000E 00
	6	5.35087350E 01	2.00000000E 00	3.00000000E 00
	5	6.67899850E 01	3.00000000E 00	3.00000000E 00
	4	8.61418360E 01	4.00000000E 00	3.00000000E 00
	3	1.10146060E 02	4.00000000E 00	4.00000000E 00
	2	1.82791830E 02	5.00000000E 00	4.00000000E 00
	1	2.05458070E 02	5.00000000E 00	5.00000000E 00
N= 3				
	11	3.02162430E 01	.00000000E-00	
	10	3.27848300E 01	1.00000000E 00	
	9	3.54301420E 01	1.00000000E 00	
	8	3.82302840E 01	1.00000000E 00	
	7	4.59617660E 01	2.00000000E 00	
	6	5.41294180E 01	2.00000000E 00	
	5	6.26648950E 01	2.00000000E 00	
	4	7.70167470E 01	3.00000000E 00	
	3	9.65707420E 01	3.00000000E 00	
	2	1.60522590E 02	3.00000000E 00	
	1	1.76385090E 02	4.00000000E 00	
N= 4				
	11	4.03620150E 01	.00000000E-00	
	10	4.29306020E 01	1.00000000E 00	
	9	4.55759140E 01	1.00000000E 00	
	8	4.83760560E 01	1.00000000E 00	
	7	5.15090160E 01	1.00000000E 00	
	6	5.94090160E 01	2.00000000E 00	
	5	6.79444930E 01	2.00000000E 00	
	4	7.83545670E 01	2.00000000E 00	
	3	9.41061870E 01	2.00000000E 00	
	2	1.56606180E 02	3.00000000E 00	
	1	1.68399840E 02	3.00000000E 00	
N= 5				
	11	5.05934960E 01	.00000000E-00	
	10	5.31620830E 01	1.00000000E 00	
	9	5.58073950E 01	1.00000000E 00	
	8	5.86075370E 01	1.00000000E 00	
	7	6.17404970E 01	1.00000000E 00	
	6	6.56574050E 01	1.00000000E 00	
	5	7.39386550E 01	2.00000000E 00	
	4	8.38402750E 01	2.00000000E 00	
	3	9.95918950E 01	2.00000000E 00	
	2	1.62091890E 02	3.00000000E 00	
	1	1.69737660E 02	3.00000000E 00	
	1	1.68399840E 02	3.00000000E 00	

EXAMPLE FOUR

S= 10.00 C= 10.00 Y= 11.00 D= 1.00 K= 500.00

	X	F(NX)	Q(NX)	Q1
N= 2				
	11	2.11858710E 01	.00000000E-00	.00000000E-00
	10	2.68717420E 01	1.00000000E 00	.00000000E-00
	9	3.26389950E 01	1.00000000E 00	1.00000000E 00
	8	4.88920130E 01	1.00000000E 00	2.00000000E 00
	7	6.62068280E 01	2.00000000E 00	2.00000000E 00
	6	9.50873540E 01	2.00000000E 00	3.00000000E 00
	5	1.27899850E 02	3.00000000E 00	3.00000000E 00
	4	1.81418370E 02	4.00000000E 00	3.00000000E 00
	3	2.81460630E 02	4.00000000E 00	4.00000000E 00
	2	8.27918340E 02	5.00000000E 00	4.00000000E 00
	1	8.74580770E 02	5.00000000E 00	5.00000000E 00
N= 3				
	11	3.21624330E 01	.00000000E-00	
	10	3.78483040E 01	1.00000000E 00	
	9	4.43014280E 01	1.00000000E 00	
	8	5.23028640E 01	1.00000000E 00	
	7	6.96176790E 01	2.00000000E 00	
	6	9.12941830E 01	2.00000000E 00	
	5	1.16648950E 02	2.00000000E 00	
	4	1.60167470E 02	3.00000000E 00	
	3	2.55707420E 02	3.00000000E 00	
	2	7.90707410E 02	4.00000000E 00	
	1	8.13851010E 02	4.00000000E 00	
N= 4				
	11	4.36201580E 01	.00000000E-00	
	10	4.93060290E 01	1.00000000E 00	
	9	5.57591530E 01	1.00000000E 00	
	8	6.37605890E 01	1.00000000E 00	
	7	7.50901600E 01	1.00000000E 00	
	6	9.40901600E 01	2.00000000E C0	
	5	1.19444930E 02	2.00000000E 00	
	4	1.62963450E 02	3.00000000E 00	
	3	2.51045670E 02	3.00000000E 00	
	2	7.86045660E 02	4.00000000E 00	
	1	7.99479950E 02	4.00000000E 00	
N= 5				
	11	5.59349720E 01	.00000000E-00	
	10	6.16208430E 01	1.00000000E 00	
	9	6.80739670E 01	1.00000000E 00	
	8	7.60754030E 01	1.00000000E 00	
	7	8.74049740E 01	1.00000000E 00	
	6	1.06404970E 02	2.00000000E 00	
	5	1.29386560E 02	2.00000000E 00	
	4	1.68402760E 02	2.00000000E 00	
	3	2.55902750E 02	3.00000000E 00	
	2	7.90902740E 02	4.00000000E 00	
	1	8.02275930E 02	4.00000000E 00	
	1	7.99479950E 02	4.00000000E 00	

The solutions are under these headings. The numbers noted by N in the left hand column refer to the number of a given stage, and the numbers listed under the column headed X refer to particular units in the planning period.

The last value under the column headed F(NX) is the optimum N stage policy. To find the value of N, search the F(NX) column at the points where X = 1 until this number is found; N is then read from the left hand column.

The number listed under the column QNX is the production quantity for the N^{th} stage. In the solution for Example One where the inputs are S = 10; C = 10; Y = 6; D = 1; K = 2, the optimum production policy is found as follows: N is found to be 3, and Q(3,1) = 2. This sets X = 3 in the planning period and N = 2; Q(23) = 2; Q1 is read directly across from Q(23), Q1 = 1. This gives a production policy of Q(31) = 2, Q(23) = 2, Q(1) = 1, which spans the planning period of five production units. The cost of this policy is F(3,1) = 77.784.

VITA

Bobby Carroll Spradlin

Candidate for the Degree of

Doctor of Philosophy

Thesis: OPTIMUM N-STAGE PRODUCTION POLICIES UNDER A LEARNING EFFECT

Major Field: Engineering

Biographical:

Personal Data: Born in Lawton, Oklahoma, June 16, 1937, the son of Carl L. and Bessie S. Spradlin.

Education: Attended grade school in Lawton, Oklahoma; graduated from Lawton High School in 1955; received the Degree of Bachelor of Science from Oklahoma State University in January, 1961; received the Degree of Master of Science from Oklahoma State University in August, 1963; completed requirements for the Degree of Doctor of Philosophy in August, 1964.

Professional Experience: Computer Science Division of The Transportation Research Command, Ft. Eustis, Virginia, February, 1961-March, 1961; employed as a Quality Control Engineer, Sandia Corporation, Albuquerque, New Mexico, May, 1963-September, 1963; accepted position at General Dynamics, Ft. Worth, Texas, as Senior Operations Analyst, Operations Research Division, effective August, 1964.