ON LINEAR, DYNAMIC, AND STOCHASTIC PROGRAMMING

by 557

DON BRIAN CAMPBELL

B.S., Kansas State University, 1965

B.S., Kansas State University, 1967

A MASTER'S REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Statistics

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1970

Approved by:

H. C. Fryer
Major Professor

LD
2668
R4
1971
C3
c.2

CONTENTS

page
0. INTRODUCTION.. 1

1. LINEAR PROGRAMMING.. 3

1.1 Problem Formulation.. 3
1.2 Simplex Method... 14
1.3 Example of Simplex Method.. 21

2. DYNAMIC PROGRAMMING... 26

2.1 Technique Formulation.. 26
2.2 Computational Procedure.. 39

3. STOCHASTIC PROGRAMMING.. 46

3.1 Problem Formulation.. 46
3.2 Dynamic Programming and Markov Processes........................... 60
3.3 Stochastic Linear Programming..................................... 77
3.4 Reliability in Linear Programming.................................. 85
3.5 Example of Reliability with Demand as Random Variable.............. 89
3.5.1 Minimum Expectation Policy....................................... 92
3.5.2 Computational Procedures for the Minimum Expectation Policy...... 96
3.5.3 Application of Kataoka Method of Reliability..................... 99

4. SUMMARY...108

0. INTRODUCTION

The last thirty years has seen the incorporation of mathematical methods within operations research. The earliest operations research studies are attributed to the British Military in the early part of World War II (Trefethen, 1954). The formal work in operations research began in the United States in the early 1940's, also with regard to military problems.

Within the area of operations research today are mathematical problems which contain factors that cannot be predicted with certainty, such as economic demand factors. The mathematical problems with random variables within operations research now are classified according to the function to be optimized and computational techniques to solve the formulations. Problems dealing with a linear function which is to be optimized over a constrained set of variables are widely used as economic models. The best known and most widely used procedure for solving linear programming problems is called the simplex method, developed by George Dantzig in 1947 (Hadley, 1962). The simplex method is an algebraic iterative procedure which exactly solves any linear programming problem in a finite number of steps, or gives an indication of an unbounded solution.

With further development of mathematical methods in operations research various computational techniques were devised. Dynamic programming is an approach to optimization which can be more generally employed than a technique such as the simplex method which applies only to linear objective functions. Dynamic programming is a term which Bellman coined in the early 1950's for a recursive optimization technique developed by Bellman and his associates at the RAND Corporation (Hadley, 1964). The technique developed really refers to the types of problems to which it can be applied. The application of dynamic programming to Markovian processes is one area where considerable work has been done (Howard, 1960). This use of dynamic programming naturally concerns processes with random variables.

The purpose of this paper is to consider various optimization problems which have random variates within their formulation and to indicate some of the available solutions.

1. LINEAR PROGRAMMING

1.1 *Problem Formulation*

The general problem of linear programming, (LP), is to optimize a linear function of variables constrained by linear relations (equalities or inequalities). A distinction is made with regard to constraints. Part or all of the variables are either nonnegative or nonpositive and all other variables are completely arbitrary. Linear programs are also divided into various types. Two general types are Integer Programs where variables are defined over the integers, and those programs which are defined over the reals. Of interest here will be those (LP) programs where variables take on real values.

The following is an algebraic formulation of an (LP) problem of the most general form:

$$\min \text{ (or max) } z = \sum_{j=1}^{n} c_j \, x_j$$

subject to (s.t.) the constraints:

$$\left.\begin{aligned}
&\text{(i)} \quad \sum_{j=1}^{n} a_{ij} \, x_j \geq d_i, && i = 1, \ldots, p \\
&\text{(ii)} \quad \sum_{j=p+1}^{m} a_{ij} \, x_j = d_i, && i = p+1, \ldots, m \\
&\text{(iii)} \quad x_j \geq 0, && j = 1, \ldots, q \\
&\text{(iv)} \quad x_j \text{ arbitrary}, && j = q+1, \ldots, n,
\end{aligned}\right\} \tag{1.1}$$

where all c_j, a_{ij}, and d_i have numerical values.

The following notation will be used for (LP) problems (Simonnard, 1966) unless otherwise specified:

$M = \{1, \ldots, m\}$: to be a set of constraint indices

$N = \{1, \ldots, n\}$: to be a set of variable indices

$M_1 \subset M$

$N_1 \subset N$

$\underset{\sim}{A} = (a_{ij})$, where $i \varepsilon M$, $j \varepsilon N$: to be a coefficient matrix of dimension $m \times n$

$\underset{\sim}{a}_j$: to be the jth column vector of $\underset{\sim}{A}$

$\underset{\sim}{a}_i$: to be the ith row vector of $\underset{\sim}{A}$

$\underset{\sim}{x} = (x_j)$, $j \varepsilon N$: to be a column vector of constants of n components

$\underset{\sim}{c} = (c_j)$, $j \varepsilon N$: to be a row vector of constants of n components

$\underset{\sim}{d} = (d_i)$, $i \varepsilon M$: to be a column vector of m components.

Then the general (LP) problem can be stated in matrix form as

$$\left.\begin{array}{lll} \min\ (\max)\ z = \underset{\sim}{c}\,\underset{\sim}{x} & & \\ (\text{s.t.}):\ \underset{\sim}{a}_i\,\underset{\sim}{x} \geq d_i, & & i \varepsilon M_1 \\ \qquad\qquad \underset{\sim}{a}_i\,\underset{\sim}{x} = d_i, & & i \varepsilon M - M_1 \\ \qquad\qquad\ \ x_j \geq 0\ , & & j \varepsilon N_1 \\ \qquad\qquad\ \ x_j\ \text{arbitrary}, & & j \varepsilon N - N_1\ . \end{array}\right\} \tag{1.2}$$

Except for the dual problem the development of (LP) problems is made under the constraint $x_j \geq 0$ for every j, rather than under the constraints (iii) and (iv) in (1.1).

Therefore, the general (LP) problem becomes

$$\left.\begin{aligned} &\min\ (\max)\ z = \underset{\sim}{c}\,\underset{\sim}{x} \\ &(\text{s.t}):\ \underset{\sim}{A}\,\underset{\sim}{x} \geq \underset{\sim}{d} \\ &\qquad\quad \underset{\sim}{x} \geq \underset{\sim}{0}\ . \end{aligned}\right\} \tag{1.3}$$

Now every inequality $\underset{\sim}{a}_i\,\underset{\sim}{x} \geq \underset{\sim}{d}_i$, or $\underset{\sim}{a}_i\,\underset{\sim}{x} \leq \underset{\sim}{d}_i$ can respectively be replaced by the relations:

$$\underset{\sim}{a}_i\,\underset{\sim}{x} - x_i^s = d_i,\quad x_i^s \geq 0 \tag{1.4}$$

or

$$\underset{\sim}{a}_i\,\underset{\sim}{x} + x_i^s = d_i,\quad x_i^s \geq 0 \tag{1.5}$$

The subtraction or addition of a supplementary non-negative variable, denoted as a slack variable, is given a coefficient zero in the function to be optimized, termed the objective function. Also, the *standard form of an (LP) problem* generally preferred is the minimization of the objective function. This minimization can be obtained from the following relation should the (LP) problem be stated as the maximum of objective function:

$$\text{minimum } f(x) = -\text{ maximum } [-\,f(x)] \tag{1.6}$$

where f(x) is the linear function to be optimized. Thus, the standard form for an (LP) problem becomes:

$$\left.\begin{aligned} &\min\ z = \underset{\sim}{c}\,\underset{\sim}{x} \\ &(\text{s.t.}):\ \underset{\sim}{A}\,\underset{\sim}{x} = \underset{\sim}{d} \\ &\qquad\quad\ \underset{\sim}{x} \geq \underset{\sim}{0} \end{aligned}\right\} \tag{1.7}$$

as presented by Erdelyi (1968).

Certain terminology is necessary for further development of the (LP) problem. Any set of $\underset{\sim}{x}$ which satisfies $\underset{\sim}{A}\ \underset{\sim}{x} = \underset{\sim}{d}$ is termed a "solution" to the (LP) problem. Any solution which satisfies $\underset{\sim}{x} \geq \underset{\sim}{0}$ is termed a "feasible solution". Any feasible solution which optimizes (minimizes in our standard form) $z = \underset{\sim}{c}\ \underset{\sim}{x}$ is called an "optimal feasible solution" (Hadley, 1962).

The system of equations $\underset{\sim}{A}\ \underset{\sim}{x} = \underset{\sim}{d}$ is taken to be nonredundant, that is, $\rho(A) = m<n$. Since $m<n$ we have at least two solutions. (LP) problems in the cases where $m = n$ and there is a unique solution, or where there is no solution at all are of no interest.

A "basis" $\underset{\sim}{B}$ of the standard (LP) problem is a set of m linearly independent $\underset{\sim}{a}_j$ vectors. Therefore, $\underset{\sim}{B}$ is a nonsingular submatrix of $\underset{\sim}{A}$ with $\rho(\underset{\sim}{B}) = m$. The m x_j's associated with $\underset{\sim}{B}$ are termed "basic variables," denoted in matrix form as $\underset{\sim}{x}^B$, an m x 1 column vector. The remaining n-m x_j's not associated with the basis are denoted as "secondary variables," shown in matrix form as $\underset{\sim}{x}^R$, an (n-m) x 1 column vector. If $\underset{\sim}{B}$ is a basis of the standard (LP) problem and $\underset{\sim}{x}^R$ is set to $\underset{\sim}{0}$, then there exists a unique solution to

$$\underset{\sim}{B}\underset{\sim}{x}^B = \underset{\sim}{d} \tag{1.8}$$

namely,

$$\underset{\sim}{x}^B = \underset{\sim}{B}^{-1}\ \underset{\sim}{d}\ .$$

The "basic solution" associated with B is

$$\left.\begin{aligned} \underset{\sim}{x}^B &= \underset{\sim}{B}^{-1}\ \underset{\sim}{d} \\ \underset{\sim}{x}^R &= \underset{\sim}{0} \quad . \end{aligned}\right\} \tag{1.9}$$

A basic solution is termed "degenerate" if there are not m strictly positive x_j components of $\underset{\sim}{x}^B$. From the basic solution, techniques have been developed which make changes of bases until the condition for a feasible solution is satisfied, namely $\underset{\sim}{x} \geq 0$, and then to finally have an optimal feasible solution. These methods depend upon an initial basic solution.

(LP) problems are N-dimensional Euclidean geometry problems. The n independent variables of the standard (LP) problem are a set of n elements $(x_1, x_2, \ldots, x_n)$ which generate the n-dimensional Euclidean space. The subspace generated by the set of all points $(x_1, \ldots, x_n)$ which satisfy $\sum_{j=1}^{n} a_{ij}\, x_j - d_i$, $i = 1, \ldots, m \leq n$ is the solution space. Development of techniques can then either follow arguments based upon finite Euclidean geometry or follow the notion of linear algebra and matrix theory.

The fundamental theorem of linear programming can now be stated.

Theorem 1.1: Given an (LP) problem in standard form (1.7),

(i) if it has at least one finite feasible solution, it has at least one basic feasible solution.

(ii) if it has at least one finite optimal feasible solution, it has at least one optimal basic feasible solution.

The proof can be either from n-dimensional geometry which appeals to the theory of convex polyhedra (Simonnard, 1966) or by using linear algebra and matrix theory. The proof given here is the latter method.

Proof of Theorem 1.1. (Simonnard, 1966):

(i) Let $\mathbf{A}\mathbf{x} = \mathbf{d}$ be the linear system in standard form (1.7); $\mathbf{A}$ is m x n. Consider an arbitrary finite feasible solution, and further suppose the variables have been ordered such that those which have positive values are the first k ($k \leq n$), and the last (n-k) are all of zero value. Let $\mathbf{A}_1$ be the matrix formed by the k first columns of $\mathbf{A}$, i.e. $\mathbf{A}_1$ contains all the positive valued variables,

$$A_1 = (\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k).$$

Then the finite feasible solution can be written

$$\sum_{j=1}^{k} x_j \mathbf{a}_j = \mathbf{d}.$$

Two and only two cases are possible:

Case 1: $\rho(\mathbf{A}_1) = k$.

This requires $k \leq m$ for a solution. The column vectors $\mathbf{a}_1, \ldots, \mathbf{a}_k$ *are therefore linearly independent.* Let $\mathbf{A}^{(m)} = (\mathbf{a}_{\alpha_1}, \mathbf{a}_{\alpha_2}, \ldots, \mathbf{a}_{\alpha_m})$ be a nonsingular submatirx of $\mathbf{A}$ with rank m. There exists at least one such matrix since $\rho(\mathbf{A}) = m$. The m columns of $\mathbf{A}^{(m)}$ form a basis of the m-dimensional space R^m. The vector $\mathbf{a}_j$, j = 1, ..., k can be expressed *as a linear function of the vectors* $\mathbf{a}_{\alpha_1}$, i = 1, ..., m,

$$\mathbf{a}_j = \sum_{i=1}^{m} \lambda_{ij} \mathbf{a}_{\alpha_i}, \quad j = 1, \ldots, k; \; \lambda_{jj} \neq 0.$$

Therefore, $\mathbf{a}_{\alpha_1}, \mathbf{a}_{\alpha_2}, \ldots, \mathbf{a}_{\alpha(j-1)}, \mathbf{a}_j, \mathbf{a}_{\alpha_{j+1}}, \ldots, \mathbf{a}_{\alpha_m}$ also form a basis of R^m. So $\mathbf{a}_{j'}$, $j \neq j'$ can be expressed as a linear function of the new basis

$$\underset{\sim}{a}_{j'} = \sum_{i=1}^{j-1} \lambda'_{ij'} \underset{\sim}{a}_{\alpha_1} + \mu_j \underset{\sim}{a}_j + \sum_{i=j+1}^{m} \lambda'_{ij'} \underset{\sim}{a}_{\alpha_i}, \quad j' = 1, \ldots, k; \ j' \neq j;$$

and one of the $\lambda'_{ij'} \neq 0$, for $\underset{\sim}{a}_{j'}$ and $\underset{\sim}{a}_j$ are linearly independent. If $\lambda'_{j'j'} \neq 0$ then $\underset{\sim}{a}_{j'}$ can replace $\underset{\sim}{a}_{\alpha_{j'}}$ in the basis. So, another new basis is formed for R.

Repeating the process will eventually replace the initial basis of R , $\underset{\sim}{a}_{\alpha_1}, \underset{\sim}{a}_{\alpha_2}, \ldots, \underset{\sim}{a}_{\alpha_m}$, with the k vectors $\underset{\sim}{a}_1, \underset{\sim}{a}_2, \ldots, \underset{\sim}{a}_k$ in the new basis, i.e. $(\underset{\sim}{a}_1, \ldots, \underset{\sim}{a}_k, \underset{\sim}{a}_{\alpha_{(k+1)}}, \ldots, \underset{\sim}{a}_{\alpha_m})$. The feasible solution $x_1, \ldots, x_k, x_{k+1} = 0, \ldots, x_m = 0$ is a basic feasible solution. This basic feasible solution is degenerate if $k<m$. If $k=m$ the basic feasible solution is nondegenerate.

Case 2: $\rho(\underset{\sim}{A}_1) < k$.

When $k>m$ this certainly is the case. The vectors $\underset{\sim}{a}_1, \underset{\sim}{a}_2, \ldots, \underset{\sim}{a}_k$ are then linearly dependent by

$$\sum_{j=1}^{k} \lambda_j \underset{\sim}{a}_j = \underset{\sim}{0}, \quad \text{some } \lambda_j \neq 0 \ .$$

Suppose some $\lambda_j > 0$. Then choose r such that

$$\frac{x_r}{\lambda_r} = \min_{j,\lambda_j>0} \left(\frac{x_j}{\lambda_j}\right), \quad j\varepsilon\{1, \ldots, k\}.$$

Using $\sum_{j=1}^{k} x_j \underset{\sim}{a}_j = \underset{\sim}{d}$, the following relationship can be made:

$$\sum_{j=1}^{k} (x_j - \frac{\lambda_j}{\lambda_r} x_r) \underset{\sim}{a}_j = \sum_{j=1}^{k} x_j \underset{\sim}{a}_j - \frac{x_r}{\lambda_r} \sum_{j=1}^{k} \lambda_j \underset{\sim}{a}_j = \underset{\sim}{d} \ .$$

So, let $x_j' = x_j - \frac{\lambda_j}{\lambda_r} x_r$, $j = 1, \ldots, k$, which forms a new feasible solution of not more than (k-1) positive variables, since $x_r' = 0$.

If the vectors associated with these positive variables are still linearly dependent, the operation is repeated. After at most p operations where $p \leq k - 1$ since $\underset{\sim}{d} \neq \underset{\sim}{0}$, the (k-p) vectors associated with the positive variables are linearly independent. These (k-p) linearly independent vectors revert us back to case 1. If $(k-p) < m$ then the basic feasible solution is degenerate, and if $(k-p) = m$ the basic feasible solution is nondegenerate. This completes the proof of the part (i) Theorem 1.1.

(ii) Consider now an arbitrary but finite optimal feasible solution, using the notation of the proof of part (i).

Case 1: $\rho(\underset{\sim}{A}_1) = k$. As shown in proof of Part (i) Case 1, the feasible solution is a basic feasible solution regardless of being degenerate or nondegenerate. Therefore, considering an arbitrary optimal feasible solution under case 1 immediately gives the conclusion that the optimal feasible solution is an optimal basic feasible solution.

Case 2: $\rho(\underset{\sim}{A}_1) = \ell \leq m<k$. Then there exist at least one set of ℓ column vectors $\underset{\sim}{a}_j$ of $\underset{\sim}{A}_1$ which are linearly independent. Let the column vectors $\underset{\sim}{a}_j$ be numbered such that the ℓ linearly independent vectors are the first columns of $\underset{\sim}{A}$ and therefore also of $\underset{\sim}{A}_1$. The linear system of equations $\underset{\sim}{A}\underset{\sim}{x} = \underset{\sim}{d}$ can be reduced to a linear system containing $x_{k+1}, x_{k+2}, \ldots, x_n$ equal to zero and considering

$x_{\ell+1}$, $x_{\ell+2}$, ..., x_k as parameters. This linear system has a solution in the variables x_1, x_2, ..., x_ℓ, since the rank of the matrix $(\underset{\sim}{a}_1, \underset{\sim}{a}_2, \ldots, \underset{\sim}{a}_\ell)$ of dimension m by ℓ is ℓ. The solutions for $\underset{\sim}{x}$, of dimension ℓ by 1 in the reduced system can be written as

$$x_j = \alpha_j + \sum_{i=1}^{k-\ell} \beta_{ji}\, x_{\ell+i}, \quad j = 1, 2, \ldots, \ell. \tag{1.10}$$

Note that α_j is determined from the ℓ-dimensional space where the matrix $(\underset{\sim}{a}_1, \underset{\sim}{a}_2, \ldots, \underset{\sim}{a}_\ell)$ is nonsingular of order ℓ; obtained from the m by ℓ matrix $(\underset{\sim}{a}_1, \underset{\sim}{a}_2, \ldots, \underset{\sim}{a}_\ell)$ by setting $a_{ij} = 0$, $i = \ell+1, \ldots, m$; $j = 1, 2, \ldots, \ell$. The reduced system $\underset{\sim}{A}\underset{\sim}{x} = \underset{\sim}{d}$ is a system of ℓ equations in ℓ unknowns. Therefore, $\underset{\sim}{x} = \underset{\sim}{A}^{-1}\,\underset{\sim}{d}$ and $\underset{\sim}{A}^{-1}\,\underset{\sim}{d}$ is composed of elements α_j, $j = 1, \ldots, \ell$.

The positive values of x_i, $i = 1, 2, \ldots, k$, in the optimal program satisfy (1.10). By substituting (1.10) for x_i, $i = 1, 2, \ldots, \ell$; and zero for x_i, $i = k+1, \ldots, n$, in the objective function $z = \underset{\sim}{c}\underset{\sim}{x}$, or $z = \sum_{j=1}^{n} c_j x_j$, the relationship

$$z = \alpha_0 + \sum_{i=1}^{k-\ell} \beta_j\, x_{\ell+i}$$

is obtained as follows:

$$z = \sum_{j=1}^{\ell} c_j\, x_j + \sum_{j=\ell+1}^{k} c_j\, x_j$$

$$= \sum_{j=1}^{\ell} c_j(\alpha_j + \sum_{i=1}^{k-\ell} \beta_{ji}\, x_{\ell+i}) + \sum_{j=\ell+1}^{k} c_j\, x_j$$

$$z = \sum_{j=1}^{\ell} c_j \alpha_j + \sum_{j=1}^{\ell} c_j \sum_{i=1}^{k-\ell} \beta_{ji} \, x_{\ell+i} + \sum_{j=\ell+1}^{k} c_j \, x_j$$

$$= \alpha_0 + \sum_{j=1}^{\ell} c_j \sum_{i=1}^{k-\ell} \beta_{ji} \, x_{\ell+i} + \sum_{i=1}^{k-\ell} c_{\ell+i} \, x_{\ell+i}$$

$$= \alpha_0 + \sum_{j=1}^{\ell} \sum_{i=1}^{k-\ell} c_j \beta_{ji} x_{\ell+i} + \sum_{i=1}^{k-\ell} c_{\ell+i} x_{\ell+i}$$

$$= \alpha_0 + \sum_{i=1}^{k-\ell} \sum_{j=1}^{\ell} c_j \beta_{ji} x_{\ell+i} + \sum_{i=1}^{k-\ell} c_{\ell+i} x_{\ell+i}$$

$$= \alpha_0 + \sum_{i=1}^{k-\ell} x_{\ell+i} \left(\sum_{j=1}^{\ell} c_j \beta_{ji} + c_{\ell+i} \right)$$

$$= \alpha_0 + \sum_{i=1}^{k-\ell} \beta_j x_{\ell+i} \, . \qquad (1.11)$$

The objective function then takes on its optimal feasible solution when the variables $x_{\ell+i}$ assume the values for the optimal feasible solution.

Examining (1.10), if the $x_{\ell+i}$ assume a value $\delta x_{\ell+i}$, new values of x_j result. The $\delta x_{\ell+i}$ can always be chosen sufficiently small in order for the new x_j values to remain strictly positive since the initial values of the $x_{\ell+i}$ and of the x_j are strictly positive. These new values then are a new feasible solution along with the values $x_{k+1} = x_{k+2} = \dots = x_n = 0$. So, δz from (1.11) has the sign of β_i or of $-\beta_i$, depending on $\delta x_{\ell+i}$ being positive or negative. Since z is optimal for $\delta x_{\ell+i} = 0$, $\ell+i = \ell+1, \dots, k$ implies $\beta_i = 0$, $i = 1, 2, \dots, k-\ell$, the optimal feasible solution is $z = \alpha_0$. Thus, the new feasible solution is still an optimal feasible solution.

From the above argument the scheme is to decrease the variable $x_{\ell+i}$ until it or one of the variables x_j vanishes, which gives a new optimal feasible solution with at most (k-1) non-zero variables. The operations are then repeated until after p or less operations, the (k-p) vectors associated with the (k-p) positive variables are linearly independent so that the matrix formed by these vectors is of rank (k-p). This will certainly occur for $p \leq k-1$ for $\underset{\sim}{d} \neq 0$. Thus, the method reverts to Case 1 and an optimal feasible solution which is degenerate if $k-p < m$, since again by hypothesis $\rho(\underset{\sim}{A}) = m$.

The proof of Theorem 1.1 being now complete solves from the theoretical viewpoint the (LP) problem. The number of bases and therefore the number of basic feasible solutions is finite, and from matrix theory, it is known how to calculate these basic feasible solutions. The maximal number of basic solutions is the number of square submatrices of order k which can be extracted from $\underset{\sim}{A}_1$, where $\underset{\sim}{A}_1$ is k by n (suppressing the redundant (m-k) equations), and is given by $\binom{n}{k}$. So it is sufficient to compute the value of z corresponding to each basic feasible solution in order to deduce the optimal feasible solution.

The geometrical approach to solving (LP) problems is so quickly limited by the dimensions of the problem that there is little to be gained by presentation here. Presented without proof is an n-dimensional analogue to Theorem 1.1.

If the set of feasible programs is bounded, i.e. (a convex polyhedron K), there exists at least one extremal point of K whose coordinates constitute an optimal feasible solution.

1.2 Simplex Method. Returning to the maximal number of basic feasible solutions for a system of m nonredundant equations in n unknowns where all submatrices of order m are nonsingular, it was noted that the total number of basic feasible solutions is $\binom{n}{m}$. Since (LP) problems can have a large number of equations and many unknowns, the exhaustion of all basic feasible solutions in order to find the optimal feasible solution becomes prohibitive. For example, as cited in Simonnard (1966) a problem of ten equations in twenty unknowns would require solution of 250,000 systems of ten equations in ten unknowns in order to exhaust all possible basic feasible solutions. For this reason the "Simplex Method" was developed by G. B. Dantzig to explore directly the set of basic feasible solutions. Further, the computational experience with the simplex method shows that for the usual problems where $n>m$, the optimal feasible solution is attained with considerably less calculation than $\binom{n}{m}$. Another aspect of the simplex method is that initially where nothing is known about the compatibility or redundancy of the equations the method determines the feasibility of the problem, and if feasible, finds an initial basic feasible solution. The method also shows the absence of a finite optimal feasible solution.

Before presentation of the simplex method certain notation will be given. Consider a general (LP) problem. As previously given it is always possible to put the problem in standard form (1.7). The initial conditions of $\rho(\underset{\sim}{A}) = m<n$ which is necessary and sufficient for a consistent system with an infinite number of solutions can no longer be assumed as a given condition when solving practical problems. As will be shown it is always possible to avoid calculating the rank of $\underset{\sim}{A}$ and $(\underset{\sim}{A}, \underset{\sim}{d})$ by artifically creating an initial basic program.

When considering a basis $\underset{\sim}{B}$ of the standard (LP) problem, the m column vectors of $\underset{\sim}{A}$ constituting such a basis will keep the column-index they originally had in $\underset{\sim}{A}$ disregarding the order they are arranged in $\underset{\sim}{B}$. The set of indices in the order of columns of $\underset{\sim}{B}$ are given as $I = \{j_1, \ldots, j_m\}$. So,

$$B = (a_{j_1}, \ldots, a_{j_m}) = (a_s),\ s \varepsilon I,\ I \varepsilon N,\ N = \{1, 2, \ldots, n\}.$$

The (n-m) other columns of $\underset{\sim}{A}$ are denoted as $\underset{\sim}{R} = (\underset{\sim}{a}_j)$, $j \varepsilon J = N-I$. The m basic variables associated with the columns $\underset{\sim}{a}_s$, form a column vector with m elements $\underset{\sim}{x}^B = (x_s)$, $s \varepsilon I$; the m associated elements of $\underset{\sim}{c}$ with the m basic variables are denoted as an m by 1 column vector $\underset{\sim}{c}^B = (c_s)$, $s \varepsilon I$, the secondary variables constitute a column vector of (n-m) elements $\underset{\sim}{x}^R = (x_j)$, $j \varepsilon J$ and the remaining (n-m) elements of $\underset{\sim}{c}$ form a row vector $\underset{\sim}{c}^R = (c_j)$, $j \varepsilon J$.

After rearranging the columns of $\underset{\sim}{A}$ and the rows of $\underset{\sim}{x}$ the system of linear equations can be written as

$$\min z = [\underset{\sim}{c}^B, \underset{\sim}{c}^R]\begin{pmatrix}\underset{\sim}{x}^B\\ \underset{\sim}{x}^R\end{pmatrix}$$

$$\text{s.t.:} [\underset{\sim}{B}, \underset{\sim}{R}]\begin{pmatrix}\underset{\sim}{x}^B\\ \underset{\sim}{x}^R\end{pmatrix} = \underset{\sim}{d}, \quad \begin{pmatrix}\underset{\sim}{x}^B\\ \underset{\sim}{x}^R\end{pmatrix} \geq \underset{\sim}{0} \quad .$$

A particular value of z or of the vector $\underset{\sim}{x}$ will be denoted as $\bar{z}$ or $\bar{\underset{\sim}{x}}$, respectively.

If $[\underset{\sim}{B}, \underset{\sim}{R}]\begin{pmatrix}\underset{\sim}{x}^B\\ \underset{\sim}{x}^R\end{pmatrix} = \underset{\sim}{d}$ is written as $\underset{\sim}{B}\underset{\sim}{x}^B + \underset{\sim}{R}\underset{\sim}{x}^R = \underset{\sim}{d}$ and pre-multiplied by $\underset{\sim}{B}^{-1}$, the result is

$$\underset{\sim}{x}^{B} = \underset{\sim}{B}^{-1}\,\underset{\sim}{d} - \underset{\sim}{B}^{-1}\,\underset{\sim}{R}\,\underset{\sim}{x}^{R}. \qquad (1.12)$$

A corresponding decomposition of the objective function gives

$$\min z = \underset{\sim}{c}^{B}\,\underset{\sim}{x}^{B} + \underset{\sim}{c}^{R}\,\underset{\sim}{x}^{R} \qquad (1.13)$$

or substituting (1.12) into (1.13) gives

$$\begin{aligned}\min z &= \underset{\sim}{c}^{B}(\underset{\sim}{B}^{-1}\,\underset{\sim}{d} - \underset{\sim}{B}^{-1}\underset{\sim}{R}\,\underset{\sim}{x}^{R}) + \underset{\sim}{c}^{R}\,\underset{\sim}{x}^{R} \\ &= \underset{\sim}{c}^{B}\,\underset{\sim}{B}^{-1}\,\underset{\sim}{d} - (\underset{\sim}{c}^{B}\,\underset{\sim}{B}^{-1}\underset{\sim}{R} - \underset{\sim}{c}^{R})\,\underset{\sim}{x}^{R}\ . \qquad (1.14)\end{aligned}$$

For a basic solution, setting $\underset{\sim}{x}^{R} = 0$ then

$$\underset{\sim}{\bar{x}}^{B} = \underset{\sim}{B}^{-1}\,\underset{\sim}{d}$$

$$\bar{z} = \underset{\sim}{c}^{B}\,\underset{\sim}{B}^{-1}\,\underset{\sim}{d}\ ,$$

and then (1.12) and (1.14) can be written as

$$\underset{\sim}{x}^{B} = \underset{\sim}{\bar{x}}^{B} - \underset{\sim}{B}^{-1}\,\underset{\sim}{R}\underset{\sim}{x}^{R} \qquad (1.15)$$

$$z = \bar{z} - (\underset{\sim}{c}^{B}\,\underset{\sim}{B}^{-1}\underset{\sim}{R} - \underset{\sim}{c}^{R})\underset{\sim}{x}^{R}. \qquad (1.16)$$

An equivalent expression of $\underset{\sim}{x}^{B}$ and z is

$$\underset{\sim}{x}^{B} = \underset{\sim}{\bar{x}}^{B} - \sum_{j\varepsilon J} x_j\,\underset{\sim}{y}_j \qquad (1.17)$$

$$z = \bar{z} - (\underset{\sim}{c}^{B} \sum_{j\varepsilon J} x_j\underset{\sim}{y}_j - \sum_{j\varepsilon J} c_j x_j), \qquad (1.18)$$

where $\underset{\sim}{B}^{-1} \underset{\sim}{R} = \underset{\sim}{Y} = [\underset{\sim}{y}_1, \underset{\sim}{y}_2, \ldots, \underset{\sim}{y}_{n-m}] = (y_{ij})$, $i \varepsilon I$, $j \varepsilon J$.

Denoting

$$z_j = \underset{\sim}{c}^B \underset{\sim}{y}_j, \; j \varepsilon J,$$

another expression for (1.18) becomes

$$z = \bar{z} - (\sum_{j \varepsilon J} \underset{\sim}{c}^B \underset{\sim}{y}_j x_j - \sum_{j \varepsilon J} c_j x_j)$$

$$= \bar{z} - \sum_{j \varepsilon J} (z_j - c_j) x_j \; . \qquad (1.19)$$

The fundamental theorems of the simplex method are now given.

Theorem 1.2: Given a basic feasible solution associated with a basis $\underset{\sim}{B}$, if $z_k - c_k > 0$ and $y_k \leq 0$ for some $k \varepsilon J$, then no finite optimal feasible solution exists.

Proof: Since $\underset{\sim}{y}_k \leq 0$, it follows from (1.15) of which the starting basis was $\underset{\sim}{x}^B = \bar{\underset{\sim}{x}}^B$, another program is obtained by giving x_k a value > 0, $\bar{x}_k$, with the other secondary variables remaining zero. Therefore, the value of $\underset{\sim}{x}^B$ must be altered to satisfy $\bar{\underset{\sim}{x}}'^B = \bar{\underset{\sim}{x}}^B - \bar{x}_k \underset{\sim}{y}_k \geq \bar{\underset{\sim}{x}}^B$, which is no longer a basic solution since $x_k \neq 0$. The objective function assumes a new value

$$\bar{z}' = \bar{z} - (z_k - c_k)\bar{x}_k$$

with the result that $z \to -\infty$ as $\bar{x}_k \to +\infty$.

Theorem 1.3: Given a basic feasible solution associated with a basis $\underset{\sim}{B}$, if for $k \varepsilon J$, $z_k - c_k > 0$, and if $y_{sk} > 0$ for at least one $s \varepsilon I$, then a new basic feasible solution may be obtained by substituting $\underset{\sim}{B}'$ for $\underset{\sim}{B}$ by exchanging $\underset{\sim}{a}_k$ for $\underset{\sim}{a}_h$, h being defined by

$$\bar{x}_k = \frac{\bar{x}_h}{y_{hk}} = \min_{y_{sk}>0} \left\{\frac{\bar{x}_s}{y_{sk}}\right\} , \quad s\varepsilon I$$

and is a new basic program giving z a new value $\bar{z}' \leq \bar{z}$.

<u>Proof</u>: If the secondary variable x_k is given the value $\bar{x}_k$ and the other secondary variables are zero, the new values of the basic variables are

$$\bar{x}'_s = \bar{x}_s - y_{sk}\,\bar{x}_k \geq 0, \quad s\varepsilon I,\ k\varepsilon J.$$

From the condition for determining $\bar{x}_k$, $\bar{x}_h$, the variable which is to leave the basis, it is readily seen that $\bar{x}'_h = 0$ as follows:

$$\bar{x}'_h = \bar{x}_h - y_{hk}\frac{\bar{x}_h}{y_{hk}} = 0 .$$

Therefore, the variable column vectors of $\underset{\sim}{A}$, $\underset{\sim}{a}_s$, sεI and s ≠ h, associated with the variables $\{\bar{X}'_s\}$, sεI and s ≠ h and the column vector $\underset{\sim}{a}_k$ associated with $\bar{x}_k$ variable form a new basis $\underset{\sim}{B}'$.

The preceding operations then give a new basic feasible solution in which all variables not associated with $\underset{\sim}{B}'$ are zero. The new value of z is

$$\bar{z}' = \bar{z} - (z_k - c_k)\bar{x}_k \leq \bar{z} .$$

<u>Theorem 1.4</u>: Given a basic feasible solution associated with a basis $\underset{\sim}{B}$, a necessary and sufficient condition for this basic feasible solution

to be optimal is that $z_j - c_j \leq 0$ for every $j \varepsilon J$.

This follows from (1.19) where $\bar{z}$ would be the optimal solution to the objective function since any new value z would be greater than or at the least equal to $\bar{z}$.

When the basis is changed, $z_j - c_j > 0$ is usually satisfied for a subset of J, J_1. Therefore, it is most helpful to chose k so as to maximize the absolute change in z. Since this change is

$$- (z_k - c_k) \frac{\bar{x}_h}{y_{hk}}$$

with

$$\bar{x}_k = \frac{\bar{x}_h}{y_{hk}} = \min_{s/y_{sk}>0} \left(\frac{\bar{x}_s}{y_{sk}}\right) , \qquad (1.20)$$

the maximum (absolute value) of the change in z for $k \varepsilon J_1$ is desired. In the use of the simplex algorithm a simplex criterion is employed: k is chosen so that

$$z_k - c_k = \max_{j \varepsilon J_1} [z_j - c_j] . \qquad (1.21)$$

This does not produce the maximal variation but is simple and in practice works well. Equation (1.21) is known as the <u>entry criterion</u>.

The simplex algorithm can now be stated:

(1) Determine an initial basic feasible solution $\underset{\sim}{x}^B$. Let I be the index set of columns of $\underset{\sim}{A}$ associated with $\underset{\sim}{x}^B$, and J = N - I. Compute $\underset{\sim}{B}^{-1}$ and $\underset{\sim}{Y} = \underset{\sim}{B}^{-1}\underset{\sim}{R} = \underset{\sim}{y}_j$, $j \varepsilon J$.

(2) Compute and test $z_j - c_j$, $j \varepsilon J$, where $z_j = \underset{\sim}{c}^B \underset{\sim}{y}_j$:

(i) if $z_j - c_j < 0$ for every $j \varepsilon J$, the feasible solution is optimal;

(ii) if $z_j - c_j > 0$, and if $\underset{\sim}{y}_j \leq \underset{\sim}{0}$ for at least one $j \varepsilon J$, a finite minimal feasible solution does not exist;

(iii) if $z_j - c_j > 0$, and if $\underset{\sim}{y}_j \nleq \underset{\sim}{0}$, then choose k from the entry criterion, (1.21), and determine h from the exit criterion, (1.20).

(3) Let p be the column order of a_h in the basis $\underset{\sim}{B}$.

Compute:

(i) $$\underset{\sim}{v}'_p = \left(-\frac{y_{k1}}{y_{kp}}, -\frac{y_{k2}}{y_{kp}}, \ldots, -\frac{y_{kp-1}}{y_{kp}}, \frac{1}{y_{kp}}, -\frac{y_{kp+1}}{y_{kp}}, \ldots, -\frac{y_{km}}{y_{kp}}\right)$$

(ii) $$\underset{\sim}{J}_p = [\underset{\sim}{e}_1, \underset{\sim}{e}_2, \ldots, \underset{\sim}{e}_{p-1}, \underset{\sim}{v}_p, \underset{\sim}{e}_{p+1}, \ldots, \underset{\sim}{e}_m]$$

where $\underset{\sim}{e}'_i = [0, 0, \ldots, 0, 1, 0, \ldots, 0]$, i.e., 1 is the ith element in the vector.

(iii) $$(\underset{\sim}{B}')^{-1} = \underset{\sim}{J}_p \underset{\sim}{B}^{-1}$$

(iv) $$\underset{\sim}{x}^{B'} = (\underset{\sim}{B}')^{-1} \underset{\sim}{d}$$

(v) $$\underset{\sim}{Y}' = (\underset{\sim}{B}')^{-1} \underset{\sim}{R}' \, .$$

So the preceding formulae give the change of basis, the new basis, and the new basic feasible solution. The primes can then be deleted and the process repeated from (2).

The example to follow and the above development of the simplex method come from Simonnard (1966) and Erdelyi (1968).

1.3. Example of the Simplex Method.

$$\text{minimize } z = \underset{\sim}{c}\underset{\sim}{x} = [-3\ 3\ 1]\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

$$\text{s.t.:}\quad \underset{\sim}{A}\underset{\sim}{x} = \begin{pmatrix} 3 & 4 & -1 \\ 8 & 2 & 0 \\ 2 & 1 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \le \underset{\sim}{d} = \begin{pmatrix} 10 \\ 30 \\ 8 \end{pmatrix}$$

$$\underset{\sim}{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \ge \underset{\sim}{0}\ .$$

In standard form after introducing slack variables the (LP) problem becomes:

$$\min z = \underset{\sim}{c}\underset{\sim}{x} = [-3\ 3\ 1\ 0\ 0\ 0]\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix}$$

$$\text{s.t.:}\quad \begin{pmatrix} 3 & 4 & -1 & 1 & 0 & 0 \\ 8 & 2 & 0 & 0 & 1 & 0 \\ 2 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 10 \\ 30 \\ 8 \end{pmatrix}$$

$$\underset{\sim}{x} \ge \underset{\sim}{0}\ .$$

A basic feasible solution with which to start is $\underset{\sim}{x}' = [0\ 0\ 0\ 10\ 30\ 8]$.

The initial $\underset{\sim}{B}$ is then $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ which is also $\underset{\sim}{B}^{-1}$. Next, the calculation of $\underset{\sim}{Y} = \underset{\sim}{B}^{-1}\ \underset{\sim}{R}$ gives $\underset{\sim}{Y} = \begin{pmatrix} 3 & 4 & -1 \\ 8 & 2 & 0 \\ 2 & 1 & 1 \end{pmatrix}$. Then

$$\underset{\sim}{\bar{z}} = \underset{\sim}{c}^B \underset{\sim}{Y} = [0\ 0\ 0]\ \underset{\sim}{Y} = [0\ 0\ 0] = [\bar{z}_1\ \bar{z}_2\ \bar{z}_3]$$

since the coefficients of the secondary variables are zero in the objective function. To apply the entry criterion each $z_j - c_j$ is determined:

$$z_1 - c_1 = 0 - (-3) = 3$$

$$z_2 - c_2 = 0 - (3) = -3$$

$$z_3 - c_3 = 0 - (-1) = 1,$$

which implies, by (1.21) since $y_j \not\leq 0$ for each $z_j - c_j > 0$, that x_1 enters the basis. For the exit criterion, (1.20),

$$\bar{x}_1 = \frac{\bar{x}_h}{y_{h1}} = \min_{\substack{s=(4,5,6) \\ y_{s1}>0}} \left\{\frac{\bar{x}_s}{y_{s1}}\right\},$$

y_{sk} is y_{s1} from $\underset{\sim}{y}_1 = \begin{pmatrix} 3 \\ 8 \\ 2 \end{pmatrix}$ since x_1 enters the basis $\underset{\sim}{B}$; so that

$$\frac{\bar{x}_4}{y_{41}} = \frac{10}{3}, \frac{\bar{x}_5}{y_{51}} = \frac{30}{8}, \frac{\bar{x}_6}{y_{61}} = \frac{8}{2} = 4 .$$

Therefore, $\bar{x}_1 = 10/3$ and x_4 leaves the basis. To calculate the new column vector associated with x_1 in the basis the column order of $\underset{\sim}{a}_4$ in $\underset{\sim}{B}$ is seen to be 1, so computing

$$\underset{\sim}{v}_1' = \left(\frac{1}{y_{11}} - \frac{y_{21}}{y_{11}} \sim \frac{y_{31}}{y_{11}}\right)$$

gives $\underset{\sim}{v}_1' = [1/3 - 8/3 \sim 2/3]$

The new value of J_1 is

$$\begin{pmatrix} 1/3 & 0 & 0 \\ -8/3 & 1 & 0 \\ -2/3 & 0 & 1 \end{pmatrix} .$$

So $(\underset{\sim}{B}')^{-1} = \underset{\sim}{J}_1 \underset{\sim}{B}^{-1} = \underset{\sim}{J}_1$ and the first iteration is complete. The second iteration would proceed in like manner and it becomes evident that some kind of tableau would facilitate presentation of the iterations. The simplex tableau for the example is given in Table 1.1 to illustrate as representative of the structure of a tableau. The optimal basic feasible *solution is* $\underset{\sim}{\bar{x}}' = [\frac{18}{5}\ \frac{4}{5}\ \frac{6}{5}\ 0\ 0\ 0]$ *and the minimum value of the objective* function is $z = -\ 58/5$. It has been noted that the simplex algorithm requires a knowledge of an initial basic feasible solution. Where there is no knowledge concerning an initial basic feasible solution the rank of the matrices $\underset{\sim}{A}$ and $(\underset{\sim}{A}, \underset{\sim}{d})$ should be found, which could quite easily be lengthy. Methods by which an initial basic feasible solution can be found are known which avoid determining the rank of the matrix. One method consists of artifically creating an initial

TABLE 1.1

Simplex Tableau

	c^B	x^B	$\bar{x}^B$	B^{-1}			y_k	x_3/y_{sk}	z_k-c_k
Starting Solution	0	x_4	10	1	0	0	3	$\underline{10/3}$	1
	0	x_5	30	0	1	0	8	30/8	
	0	x_6	8	0	0	1	2	8/2	
First Iteration	-3	x_1	10/3	1/3	0	0	-1/3	———	
	0	x_5	10/3	-8/3	1	0	8/3	10/8	
	0	x_6	4/3	-2/3	0	1	5/3	4/5	2
Second Iteration	-3	x_1	18/5	1/5	0	1/5			
	0	x_5	6/5	-8/5	1	-8/5			<0
	-1	x_3	4/5	-2/5	0	3/5			

basic feasible solution. This method is known as "The Method of Penalties" (Simonnard, 1966; Hadley, 1962) or the "-M technique" (Gue & Thomas, 1968). Another method is known as the "Two-Phase Method" (Simonnard, 1966; Hadley, 1962) where phase I determines if any feasible solutions exist, and if so calculates a feasible solution. Phase II improves the feasible solution obtained in the first phase to the optimal feasible solution. The two-phase method was developed to overcome difficulties inherent to the method of penalties which is round off error and accuracy of the optimal solution.

The (LP) problem has been defined and the development of the simplex algorithm has been presented as a method of solving the problem. There are a number of further considerations of (LP) problems which could be developed, but defining the problem so that later consideration of $\underset{\sim}{A}$ as a matrix containing random variables a_{ij} is all that is necessary here. Therefore, dynamic programming will now be developed.

2. DYNAMIC PROGRAMMING

2.1. Technique Formulation.

Dynamic programming (DP) refers to a computational method rather than a particular form of a nonlinear programming problem. The development of (DP) will follow that of Nemhauser (1966). (DP) originated as a result of studying certain forms of sequential decision problems arising from inventory theory. Developed by Bellman in the early 1950's who coined the name of the computation technique, dynamic programming reduces a problem of n decisions to n problems of one decision each.

The following notation and terminology will be used:

(i) $D = (d_1\ d_2\ \ldots\ d_n)$: variables which are termed the independent or decision variables;

(ii) $Y = (y_1\ y_2\ \ldots\ y_p)$: parameters which affect the objective function but are uncontrollable;

(iii) R: Dependent variables which are functions of the decision variables and parameters, i.e. R = R(D,Y), and is termed the return function.

(iv) S: The region of feasibility or constraint set generally represented by

$$g_i(\underset{\sim}{D}) \left.\begin{matrix} \leq \\ = \\ \geq \end{matrix}\right\} 0, \quad i = 1, \ldots, m.$$

As with the (LP) problem, any D satisfying the constraints is a feasible solution. Any optimal solution (D*) is defined as a feasible solution yielding the greatest possible return, i.e.

$$R(Y) = R(D^*, Y) \geq R(D, Y), \ D\varepsilon S$$

$$= \max_{D} \ R(D, Y), \ D\varepsilon S.$$

Basically, (DP) is a transformation from a sequential or multistage decision process containing many interdependent variables which converts the process into a series of single-stage problems with only a few variables. The transformation is based on Bellman's "Principle of Optimality" which Nemhauser (1966) states:

> "an optimal set of decisions has the property that whatever the first decision is, the remaining decisions must be optimal with respect to the outcome which results from the first decision."

Bellman's work is generally acknowledged as the foundation of dynamic programming. From this general approach to (DP) a specific development is now given.

Consider a system which can be described by a state vector $\underset{\sim}{X}_o$ and the system is to be changed so that it can be described by a state vector $\underset{\sim}{X}_N$ different from $\underset{\sim}{X}_o$. So the transformation from $\underset{\sim}{X}_o$ to $\underset{\sim}{X}_N$ is desired, or $X_N = T_N\ (X_o)$. If there is known a transformation t_N which will change the state of the system X_{N-1} to X_N, then it can be represented as $X_N = t_N(X_{N-1})$. The systems can be represented as flow diagrams. $X_N = T_N\ (X_o)$ can be represented as

$$\xrightarrow{X_o} \boxed{T_N} \xrightarrow{X_N} \qquad (2.1)$$

and $X_N = t_N(X_{N-1})$ as

$$\xrightarrow{X_{N-1}} \boxed{t_N} \xrightarrow{X_N} . \qquad (2.2)$$

If the system is $X_N = t_N(X_{N-1})$ then to solve the original problem $X_N = T_N(X_o)$ a transformation is needed to change the system from X_o to X_{N-1}. If T_{N-1} is such a transformation then the original problem (2.1) can be represented as

$$\xrightarrow{X_o}\boxed{T_{N-1}}\xrightarrow{X_{N-1}}\boxed{t_N}\xrightarrow{X_N} \quad .$$

So the equivalence is established by

$$X_N = t_N(T_{N-1}(X_o)) = T_N(X_o),$$

It is apparent that the decomposition of (2.1) may possibly continue if appropriate transformations can be found. Thus, the final result may be obtained by decomposing (2.1), a problem of N state variables into N subproblems of one state variable each:

$$\left.\begin{aligned}
&1. && X_N = t_N(X_{N-1})\\
&2. && X_{N-1} = t_{N-1}(X_{N-2})\\
& && \cdots\cdots\cdots\cdots\\
&N\text{-}n. && X_{n+1} = t_{n+1}(X_n)\\
&N\text{-}n\text{+}1. && X_n = t_n(X_{n-1})\\
& && \cdots\cdots\cdots\cdots\\
&N. && X_1 = t_1(X_o).
\end{aligned}\right\} \qquad (2.3)$$

The flow diagram for (2.3) is

$$\xrightarrow{X_o}\boxed{t_1}\xrightarrow{X_1}\Big| \quad \Big|\xrightarrow{X_{n-1}}\boxed{t_n}\xrightarrow{X_n}\boxed{t_{n+1}}\xrightarrow{X_{n+1}}\Big| \quad \Big|\xrightarrow{X_{N-2}}\boxed{t_{N-1}}\xrightarrow{X_{N-1}}\boxed{t_N}\xrightarrow{X_N}.$$

Note, however, that the multistage transformations started with the final state X_N and proceeded to the initial state X_o through the transformations t_N, t_{N-1}, ..., t_1. This is known as "backward" recursion. To avoid confusion there is a change of notation so that the transformation indices agree with the order of performing the required transformation. This is done simply by renumbering the transformations in reverse order. Thus X_o becomes the final state and X_N become the initial state. The N subproblems then become

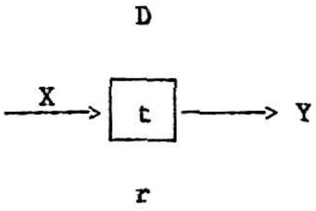
$$\begin{array}{rl}
1. & X_o = t_1(X_1) \\
2. & X_1 = t_2(X_2) \\
 & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot \\
n. & X_{n-1} = t_n(X_n) \\
n+1. & X_n = t_{n+1}(X_{n+1}) \\
 & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot \\
N. & X_{N-1} = t_N(X_N)
\end{array} \tag{2.4}$$

with a corresponding flow diagram

$$\xrightarrow{X_N} \boxed{t_N} \xrightarrow{X_{N-1}} \big\} \quad \big\{ \xrightarrow{X_{n+1}} \boxed{t_{n+1}} \xrightarrow{X_n} \boxed{t_n} \xrightarrow{X_{n-1}} \big\} \quad \big\{ \xrightarrow{X_1} \boxed{t_1} \xrightarrow{X_o} .$$

The N state variable system has been replaced by an equivalent system containing N one state problems. Methods for finding the transformations for the decomposition of the original system and conditions under which the decomposition can be performed will be stated. Before these considerations, however, a more general form of problems suited to dynamic programming is presented. If Equations (2.3) have more than one solution, then all solutions are equally satisfactory. If a decision variable is added to the system which contains more than one feasible solution then a dependent variable measuring the effectiveness of a decision is also added to the system. The problem then is one of choosing optimal decisions which yield maximum returns.

A one stage representation of the system is as follows:



(2.5)

with the factors

(i) X: Input state variable giving a description of the system at the beginning of stage.

(ii) Y: Output state variable giving description of system at termination of stage.

(iii) D: Decision variable which is independent variable controlling the operation of the box.

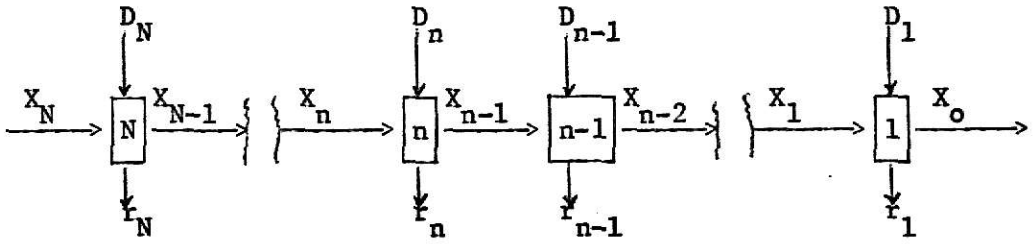
(iv) r: Stage return dependent variable measuring the effectiveness of decisions and is a single-valued function $r = r(X,D,Y)$.

(v) t: Stage transformation - a single-valued transformation expressing each element of the output state as a function of the input state and decisions, that is, $Y = t(X,D)$. Since $Y = t(X,D)$, Nemhauser (1966) writes the stage return either as $r = r(X,D,t(X,D))$ or as $r = r(X,D)$.

The one-stage optimization problem is to maximize the stage return as a function of the input state. Denoting $f(X)$ as the optimal stage return and $D^* = D(X)$ as the optimal decisions, then

$$f(X) = r(X,D(X)) = r(X,D^*) = \max_{D} r(X,D) \geq r(X,D).$$

Consider a system containing a set of stages joined in series so that the output of one stage is input for the next stage. The system is termed a serial multistage system and is represented in a flow diagram as



when using the renumbering of transformation indices as in (2.4).

For the general stage $n(n = 1, 2, \ldots, N)$ of the N-stage system, the stage transformation is $X_{n-1} = t_n(X_n, D_n)$ and the corresponding stage return is $r_n = r_n(X_n, D_n)$. Certain relationships become apparent from the structure of the serial multistage decision system. From the transformations X_n depends only on the decisions made prior to stage n, i.e., $D_{n+1}, \ldots, D_N$ and X_N. This can be written as

$$\begin{aligned}
X_n &= t_{n+1}(X_{n+1}, D_{n+1}) = t_{n+1}(t_{n+2}(X_{n+2}, D_{n+2}), D_{n+1}) \\
&= t'_{n+1}(X_{n+2}, D_{n+2}, D_{n+1}) = t'_{n+1}(t_{n+3}(X_{n+3}, D_{n+3}), D_{n+2}, D_{n+1}) \\
&= t''_{n+1}(X_{n+3}, D_{n+3}, D_{n+2}, D_{n+1}) \\
&= \ldots\ldots\ldots\ldots\ldots\ldots \qquad (2.6) \\
&= t_{n+1}^{''\cdots'}(X_N, D_N, \ldots, D_{n+1}) \\
&= t_{n+1}(X_N, D_N, \ldots, D_{n+1}) ,
\end{aligned}$$

where t_{n+1} becomes a different transformation over X_N, D_N, ..., D_{n+1} from the initial t_{n+1} over X_{n+1} and D_{n+1}. Since the return function depends on X_n and D_n (2.6) gives the result that r_n depends on X_N and D_N, ..., D_n, i.e.

$$r_n = r(X_n, D_n) = r_n(X_N, D_N, \ldots, D_n), \tag{2.7}$$

where r_n over X_N, D_N, ..., D_n is a different function from r_n over X_n and D_n.

The total return function R_n over all stages is a function of the individual stage returns, i.e.,

$$R_N(X_N, X_{N-1}, \ldots, X_1; D_N, D_{N-1}, \ldots, D_1) =$$
$$g[r_N(X_N,D_N), r_{N-1}(X_{N-1},D_{N-1}), \ldots, r_1(X_1,D_1)].$$

From the elimination of X_{N-1}, ..., X_1 as in (2.6), (2.7) reduces to

$$R_N(X_N,D_N,D_{N-1},\ldots,D_1) = g[r_N(X_N,D_N),r_{N-1}(X_N,D_N,D_{N-1}),\ldots,r_1(X_N,D_N,\ldots,D_1)]. \tag{2.8}$$

The N-stage initial optimization problem is to maximize the N-stage return R_n over the variables D_1, ..., D_N which is to find an optimal return as a function of the initial state X_N. Let $f_N(X_N)$ be the maximum N-stage return, and $D_n^* = \bar{D}_n(X_n)$, $X_n^* = t_n(X_n)$ as the optimal decisions and states. Then $f_N(X_N)$ can be expressed as either

$$\text{(i)}\quad f_N(X_N) = g[r_N(X_N,D_N^*), r_{N-1}(X_N,D_N^*,D_{N-1}^*),\ldots,r_1(X_N,D_N^*,\ldots,D_1^*)]$$

$$= \max_{D_N,\ldots,D_1} g[r_N(X_N,D_N),r_{N-1}(X_N,D_N,D_{N-1}),\ldots,$$
$$r_1(X_N,D_N,\ldots,D_1)] \tag{2.9}$$

or

$$\text{(ii)} \quad f_N(X_N) = g[r_N(X_N, D_N^*),\ r_{N-1}(X_{N-1}^*,\ D_{N-2}^*),\ \ldots,\ r_1(X_1^*,\ D_1^*)]$$

$$= \max_{D_N, \ldots, D_1} g[r_N(X_N, D_N), r_{N-1}(X_{N-1}^*, D_{N-1}^*), \ldots, r_1(X_1, D_1)] \tag{2.10}$$

subject to, (s.t): $X_{n-1} = t_n(X_n, D_n)$, $n = 1, 2, \ldots, N.$

Note that (2.9) contains N decision variables and one state variable while (2.10) contains N decision variables, N state variables, and N constraints. It would seem that (2.9) would then be preferred since optimization techniques decrease in efficiency as the number of variables increase. However, (2.10) can under certain conditions be transformed into N optimization problems, each containing one decision variable and one state variable.

Consider the decomposition of the N-stage return function (2.10). To achieve decomposition the function g must be such that the maximization with respect to D_{N-1}, ..., D_1 can be moved inside the Nth stage return, i.e.,

$$f_N(X_N) = \max_{D_N} g_1[r_N(X_N, D_N),\ \max_{D_{N-1}, \ldots, D_1} g_2(r_{N-1}(X_{N-1}, D_{N-1}), \ldots, r_1(X_1, D_1))].$$

Sufficient conditions for decomposition of the maximization of stage returns as given by Mitten (1964) are as follows:

If the return function g satisfies the conditions of

(i) Separability

$$g[r_N(X_N, D_N), r_{N-1}(X_{N-1}, D_{N-1}), \ldots, r_1(X_1, D_1)]$$

$$= g_1[r_N(X_N, D_N), g_2(r_{N-1}(X_{N-1}, D_{N-1}), \ldots, r_1(X_1, D_1))],$$

where g_1 and g_2 are real valued functions, and

(ii) Monotonicity

g_1 is a monotonically nondecreasing function of g_2 for every r_N, then

(iii) g can be decomposed as

$$\max_{D_N,D_{N-1},\ldots,D_1} g[r_N(X_N,D_N),r_{N-1}(X_{N-1},D_{N-1}),\ldots,r_1(X_1,D_1)]$$

$$= \max_{D_N} g_1[r_N(X_N,D_N), \max_{D_{N-1},\ldots,D_1} g(r_{N-1}(X_{N-1},D_{N-1}),\ldots, r_1(X_1,D_1))].$$

The proof of decomposition for an N-stage system is identical to the proof for a two-stage problem. Therefore, the following proof of decomposition is for a two-stage problem. This proof follows the development of Nemhauser (1966).

Proof: Let $f_2(X_2) = \max_{D_2,D_1} g(r_2(X_2, D_2), r_1(X_1,D_1))$

s.t.: $X_1 = t_2(X_2,D_2)$.

Substituting in for X_1 gives $f_2(X_2) = \max_{D_2,D_1} g(r_2(X_2,D_2), r_1(X_2,D_2,D_1))$.

Let $f_2'(X_2) = \max_{D_2} [g(r_2(X_2,D_2), \max_{D_1} r_1(X_2,D_2,D_1))]$. The definition of a maximum gives $f_2(X_2) \geq f_2'(X_2)$. Of interest is when equality holds. A sufficient condition for equality is that g be a monotically nondecreasing function of r_1 for every feasible value of r_2. From the definition of monotonicity if

$$r_1(X_2,D_2,D_1') \geq r_1(X_2,D_2,D_1''), \quad \text{(fixed } X_2,D_2)$$

Then

$$g(r_2(X_2,D_2),r_1(X_2,D_2,D_1')) \geq g(r_2(X_2,D_2),r_1(X_2,D_2,D_1'')).$$

However, for each value of X_2,D_2

$$r_1(X_2,D_2,D_1^*) = \max_{D_1} r_1(X_2,D_2,D_1) \geq r_1(X_2,D_2,D_1).$$

Thus, from the monotonicity with X_2, D_2 fixed,

$$g(r_2(X_2,D_2),r_1(X_2,D_2,D_1^*)) \geq \max_{D_1} g(r_2(X_2,D_2),\ r_1(X_2,D_2,D_1)).$$

Therefore,

$$\begin{aligned} f_2'(X_2) &= \max_{D_2}\ g(r_2(X_2,D_2),\ \max_{D_1} r_1(X_2,D_2,D_1)) \\ &\geq \max_{D_2}\max_{D_1} g(r_2(X_2,D_2),r_1(X_2,D_2,D_1) = f(X_2)\ . \qquad (2.11) \end{aligned}$$

The given inequality from the definition of maximum $f_2'(X_2) \leq f_2(X_2)$ with the derived inequality (2.11) gives the desired result, namely $f_2'(X_2) = f_2(X_2)$. So, if g is a monotonically nondecreasing function of r_1 for every r_2 then the position of maximization with respect to D_1 can be changed with no possibility of missing the optimal solution.

The condition of separability is added to decompose N-stage problems. It can be readily shown that problems with additive stage returns are always decomposable.

Consider identifying total return functions not involving strictly additive individual stage returns and yet satisfying Mitten's condition. Indicating the separability condition, the total return functions are of the form:

$$R_N = r_N(X_N,D_N) \ 0 \ r_{N-1}(X_{N-1}, D_{N-1}) \ 0 \cdots 0 \ r_1(X_1,D_1)$$

$$= r_N(X_N,D_N) \ 0 \ R_{N-1}. \qquad (2.12)$$

The symbol "0" is taken as a composition operator with the purpose of stipulating separability. If the total stage return function can be written as (2.12) then it is possible to express

$$R_N = g(r_N,r_{N-1},\ldots,r_1)$$

as

$$R_N = g_1 \ (r_N, \ g_2(r_{N-1},\ldots,r_1)).$$

Multiplication of stage returns readily gives the decomposition of the return function as long as $r_n(X_n,D_n)$ only takes on non-negative real values for all n. Consider the decomposition of

$$R_3 = r_3(X_3,D_3) + r_2(X_2,D_2) \cdot r_1(X_1,D_1).$$

Readily

$$R_2 = r_2(X_2,D_2) \cdot r_1(X_1,D_1)$$

and thus

$$R_3 = r_3(X_3,D_3) + R_2$$

and is separable. Now let

$$R_3 = r_3(X_3,D_3) \cdot r_2(X_2,D_2) + r_1(X_1,D_1).$$

If

$$R_2 = r_2(X_2,D_2) + r_1(X_1,D_1)$$

then

$$R_3 \neq r_3(X_3,D_3) \cdot R_2$$

and there is no backward solution for R_2 to obtain separability. However, the problem can be separated from the forward direction by letting

$$R_2 = r_3(X_3,D_3) \cdot r_2(X_2,D_2)$$

then

$$R_3 = R_2 + r_1(X_1,D_1).$$

To conclude examples of separability consider

$$R_4 = r_4 + r_3r_2 + r_1 \ .$$

There is no direction from which this total return function is separable. Thus, it would not meet Mitten's conditions for decomposition.

Generally, if the composition operator has the property that

$$f_{N-1}(X_{N-1}) = \max_{D_{N-1},\ldots,D_1} [r_{N-1}(X_{N-1},D_{N-1}) \ 0 \cdots 0 \ r_1(X_1,D_1)]$$

$$r_N(X_N,D_N) \ 0 \ f_{N-1}(X_{N-1}) \geq r_N(X_N,D_N) \ 0 \ [r_{N-1}(X_{N-1},D_{N-1}) \ 0 \ \ldots \ 0 \ r_1(X_1,D_1)].$$

for all values of r_N and f_{N-1}, decomposition is possible. This condition is slightly weaker than Mitten's conditions but does not seem to enlarge the decomposable problem class.

The recursive equations for the general composition operater are as follows:

Let

$$f_N(X_N) = \max_{D_N, \ldots, D_1} [r_N(X_N,D_N) \; 0 \cdots 0 \; r_1(X_1,D_1)]$$

$$\text{s.t.:} \quad X_{n-1} = t_n(X_n,D_n), \quad n = 1, \ldots, N.$$

If monotonicity is satisfied the maximization with respect to $D_{N-1}, \ldots, D_1$ can be moved inside as

$$f_N(X_N) = \max_{D_N} [r_N(X_N,D_N) \; 0 \max_{D_{N-1}, \ldots, D_1} (r_{N-1}(X_{N-1},D_{N-1}) \; 0 \cdots 0 \; r_1(X_1,D_1))]$$

$$\text{s.t.:} \quad X_{n-1} = t_n(X_n,D_n), \quad n = 1, \ldots, N.$$

Since

$$f_{N-1}(X_{N-1}) = \max_{D_{N-1}, \ldots, D_1} [r_{N-1}(X_{N-1},D_{N-1}) \; 0 \cdots 0 \; r_1(X_1,D_1)]$$

a substitution can be made with the result

$$f_N(X_N) = \max_{D_N} [r_N(X_N,D_N) \; 0 \; f_{N-1} \, (X_{N-1})]$$

$$\text{s.t.:} \quad X_{n-1} = t_n(X_n,D_n), \; n = 1, \ldots, N.$$

Let

$$Q_N(X_N,D_N) = r_N(X_N,D_N) \; 0 \; f_{N-1} \, (t_N(X_N,D_N)),$$

then determination of $f_N(X_N)$ is a one stage optimization, i.e.,

$$f_N(X_N) = \max_{D_N} (Q_N(X_N,D_N)).$$

Repeating the above argument on $f_{N-1}(X_{N-1})$, ..., $f_2(X_2)$ the following recursion equations are obtained:

$$f_n(X_n) = \max_{D_n} [Q_n(X_n, D_n)], \quad n = 1, \ldots, N$$

$$Q_n(X_n, D_n) = r_n(X_n, D_n), \qquad n = 1 \tag{2.13}$$

$$= r_n(X_n, D_n) \; 0 \; f_{n-1}(t_n(X_n, D_n)), \quad n = 2, \ldots, N$$

Note that the previous derivation of the recursive equations under the operator + follows the same approach. Any operator satisfying the conditions of Mitten will result in recursion equations of the form (2.13).

It may be of some concern whether a distinction need be made between *forward and backward recursion; between the initial state* problem $f(X_o)$ and the final state problem $f(X_N)$. For most multistage decision problems these distinctions need not be made when the choice between inputs and outputs is arbitrary from a mathematical standpoint. In these problems the transformations are constructed so that output states are functions of input states and decisions. Then the optimal return is found as a function of the input state to stage N using backward recursion. An example of multistage decision systems where direction of analysis is not arbitrary is in non-serial systems, where stages are not connected in series by state variables (Nemhauser, 1966).

2.2. Computational Procedure.

The theoretical development of dynamic programming (DP) is now complete. The next steps are the formulation of a general system into a multistage form and the determination of an efficient method for solving the recursive equations. As with the (LP) problem there are different computing methods depending on the requirements of the problem. Nowever, with (DP) there is no fundamental algorithm such as the simplex method for (LP). Yet, a general scheme for the (DP) technique of solving classes of problems meeting conditions stated previously can be formulated.

The computational aspects of (DP) concern the solution of the following recursive equations from Nemhauser (1966):

$$f_n(X_n) = \max_{D_n} Q_n(X_n, D_n), \quad n = 1, \ldots, N \tag{2.14}$$

with

$$Q_n(X_n, D_n) = r_n(X_n, D_n), \quad n = 1 \tag{2.15}$$

$$Q_n(X_n, D_n) = r_n(X_n, D_n) \ 0 \ f_{n-1}), \quad n = 2, \ldots, N \tag{2.16}$$

for

$$X_{n-1} = t_n(X_n, D_n), \quad n = 2, \ldots, N \tag{2.17}$$

Computation proceeds as

(i) $n = 1$: calculate $r_1(X_1, D_1)$. Since $n = 1$,

$Q_1(X_1, D_1) = r_1(X_1, D_1)$.

Equation (2.14) is used to find $f_1(X_1)$ and $D_1(X_1)$.

(ii) $n + 1$: calculate $r_{n+1}(X_{n+1}, D_{n+1})$. For $n + 1$, Q_{n+1} is calculated from (2.16) by appropriately combining r_{n+1} and f_n. Thus, the optimal return from the N-stage system $f_N(X_N)$ is obtained, and $D_n(X_n)$, $(n = 1, \ldots, N)$, the optimal nth stage decision functions of the nth stage inputs.

The optimal inputs X_n^*, $(n = 1, \ldots, N-1)$ and the optimal decisions D_n^*, $(n = 1, \ldots, N)$ are determined as follows:

(iii) D_N^* is obtained immediately from $D_N(X_N)$.

(iv) X_{N-1}^* is calculated from $X_{N-1}^* = t_N(X_N^*, D_N^*)$.

(v) D_{N-1}^* is obtained from X_{N-1}^* and $D_{N-1}(X_{N-1})$.

(vi) Repeat (iv) and (v) until D_1^* is established and then the optimal solution is obtained.

The method for optimizing any multistage decision system is then complete. The characteristics distinguishing problems from one another are

(i) the return functions $r_n(X_n, D_n)$ and transformations $t_n(X_n, D_n)$;

(ii) the interpretation of the operator "0" which specifies how r_n and f_{n-1} are combined to yield Q_n;

(iii) the technique used to maximize $Q_n(X_n, D_n)$ to obtain $f_n(X_n)$ and $D_n(X_n)$.

As in (LP) problems the maximization and minimization of a function have the following relationship:

$$\min f(x) = -[\max(-f(x))], \tag{2.18}$$

since f(x) is a real valued function defined over the nonnegative values of x, maximization and minimization can be interchanged through (2.18).

To illustrate the (DP) technique, consider an analytical problem

$$\text{minimize} \sum_{n=1}^{N=3} d_n^2$$

$$\text{s.t.:} \sum_{n=1}^{3} d_n \geq k, \quad k>0$$

$$d_n \geq 0, \quad n = 1, 2, 3.$$

To put the problem in the form appropriate for the technique

$$\text{minimize } r_1(x_1,d_1) + r_2(x_2,d_2) + r_3(x_3,d_3)$$

$$\text{s.t.: } x_{n-1} = t_n(x_n,d_n), \quad n = 1, 2, 3.$$

The state variables are introduced and replace

$$\sum_{n=1}^{3} d_n \geq k$$

by $\quad x_3 \geq k$

$$x_2 = x_3 - d_3$$

$$x_1 = x_2 - d_2$$

and $\quad x_o = x_1 - d_1$.

This is acceptable since adding all four equations gives

$$d_1 + d_2 + d_3 \geq k - x_o$$

and to be consistent with the requirements $x_o = 0$. This implies $d_1 = x_1 \geq 0$, which further gives $0 \leq d_2 \leq x_2$ and finally $0 \leq d_3 \leq x_3$. The problem can now be restated as

$$\text{minimize } \sum_{n=1}^{3} d_n^2$$

$$\text{s.t.: } \left.\begin{array}{ll} x_1 = x_2 - d_2, & d_1 = x_1 \geq 0 \\ x_2 = x_3 - d_3, & 0 \leq d_2 \leq x_2 \\ x_3 \geq k \quad , & 0 \leq d_3 \leq x_3 \end{array}\right\} \qquad (2.19)$$

Since

$$r_n(x_n, d_n) = d_n^2$$

$$x_{n-1} = t_n(x_n, d_n) = x_n - d_n, \quad n = 1, 2, 3$$

the problem is in proper form for the use of the technique with

$$R_3 = d_3^2 + d_2^2 + d_1^2 .$$

Stating the problem in terms of the recursive equations of (DP):

$$f_1(x_1) = \min_{d_1 = x_1} d_1^2$$

$$f_n(x_n) = \min_{0 \leq d_n \leq x_n} [d_n^2 + f_{n-1}(x_n - d_n)], \quad n = 2, 3, \text{ with } x_3 \geq k.$$

The solution procedure then begins:

(i) $n = 1$: calculating $r_1(x_1, d_1)$ gives d_1^2. Since $n = 1$, $Q_1(x_1, d_1) = d_1^2$. Finding $d_1(x_1) = x_1$ gives $f_1(x_1) = x_1^2$.

(ii) $n + 1 = 2$: calculating $r_2(x_2, d_2)$ gives d_2^2. $Q_2(x_2, d_2)$ is found by expressing the optimal one-stage return as a function of x_2 and d_2.

Since $x_1 = x_2 - d_2$, $f_1(x_1) = (x_2 - d_2)^2$,

$Q_2(x_2, d_2)$ becomes $d_2^2 + (x_2 - d_2)^2$.

Thus,

$$f_2(x_2) = \min_{0 \leq d_2 \leq x_2} [d_2^2 + (x_2 - d_2)^2].$$

Optimization of $f_2(x_2)$ is done as follows:

$$\frac{\partial Q_2}{\partial d_2} = 2\ d_2 - 2\ (x_2 - d_2) \overset{\text{set}}{=} 0$$

implies the unique solution $d_2 = x_2/2$. It is readily seen that this partial derivative of Q_2 gives a minimum since $\left.\frac{\partial^2 Q_2}{\partial d_2^2}\right|_{d_2 = \frac{x_2}{2}} > 0$.

So, $f_2(x_2) = x_2^2/2 = (x_3 - d_3)^2/2$.

(iii) For n = 3 the procedure is the same and

$$f_3(x_3) = \min_{0 \leq d_3 \leq x_3} \left(d_3^2 + \frac{(x_3 - d_3)^2}{2} \right) .$$

The optimization of $f_3(x_3)$ is found as before by setting $\frac{\partial Q_3}{\partial d_3} = 0$, which yields $d_3 = x_3/3$, and then $f_3(x_3) = x_3^2/3$. Since $x_3 \geq k$ the $f_3(x_3)$ is clearly optimized by $x_3 = k$, thus

$$f_3(x_3) = k^2/3, \quad d_3^* = k/3;$$

$$x_2^* = x_3^* - d_3 = k - k/3,\ d_2^* = x_2^2/2 = (k-k/3)/2 = k/3,$$

and $x_1^* = x_2^* - d_2^* = k/3 = d_1^*$.

More difficult problems are solved in the same manner as this simple example. This particular problem easily extends to N stages or can be extended to different return functions, transformations, and composition operators. For detailed presentation see Nemhauser (1966).

Still, it remains to be seen how much of an advantage there is to (DP) compared to exhaustive search as would be done with X_n being integer values, i.e., $X_n = k$, $(k = 1, \ldots, k_n)$. Consider an optimization problem with N stages, K_n and J_n as values for the state and decision variables respectively at stage n = 1, ..., N. Since a feasible solution is specified by a particular value of the vector $(x_N, d_N, \ldots, d_1)$, there are $K_N \prod_{n=1}^{N} J_n$ feasible solutions. To determine the return from each feasible solution requires the addition of N numbers (since for this problem total return is the sum of all stage returns). Since the numbers are added two at a time there will be $(N-1)\, K_N \prod_{n=1}^{N} J_n$ additions. Determination of a maximum from these will require $K_N \prod_{n=1}^{N} J_n - 1$ comparisons

By comparison, (DP) requires addition for each combination of (x_n, d_n) at stages two through N, or $\sum_{n=2}^{N} K_n J_n$ additions. For each value of the state variable at all stages there are J_{n-1} comparisons. At the last stage there are an additional $K_n - 1$ comparisons to determine the maximum of $f_N(x_N)$. Therefore, a total of $\sum_{n=1}^{N} K_n(J_n-1) + K_n - 1$ comparisons. To make comparisons more obvious, let $J = K_m = J_n$ for all m and n. Then the number of additions plus comparisons for direct search is $NJ^{N+1} - 1$ and for dynamic programming $(2N-1)J^2 - (N-1)J-1$. For example, if there are ten decision variable values and stage variable values for a fifty stage problem, then direct search makes 5×10^{52} comparisons and additions while (DP) makes 9409. This is quite a substantial savings. The example comes from Nemhauser (1966).

The preceeding developments of (LP) and (DP) give a general acquaintance to these types of problems. Further, the objective of these sections has been to supply background for the main consideration of this report which is the introduction of random variables rather than deterministic variables into the constrained optimization problems. The procedure then becomes one of optimizing the <u>expected value</u> of the objective function.

3. STOCHASTIC PROGRAMMING

3.1 Problem Formulation.

Consider first the (DP) technique with random variables introduced into the problem. The "randomness" of the variables can be categorized into "decision making under risk", "decision making under uncertainty", and "adaptive decision making". Decision making under risk is where the probability of occurrence for each return is known. Decision making under uncertainty is the case of complete ignorance concerning the probabilities of the different possible returns. The intermediate case is adaptive decision making where previous information from the process generates estimates of probabilities shifting the problem from the area of uncertainty to the area of risk. A basic assumption of the problem is that there is one decision maker in contrast to "game theory" which contains more than one functionally distinct decision maker. "Decision making under risk" is also known as "stochastic decision making". The following development is from Nemhauser (1966).

Consider a (DP) problem and decision making under risk with a single stage stochastic return function r(D,k) where D is the usual decision variable and k is a discrete random variable. The probability distribution of k is denoted as p(k). Then for a fixed set of D the expected value of the return function is

$$\bar{r}(D) = \sum_{k} p(k)\ r(D,k).$$

Similarly, for a continuous random variable z for a stochastic return function r(D,z), the expected value of the return function is given by

$$\bar{r}(D) = \int p(z)\ r(D,z)dz,$$

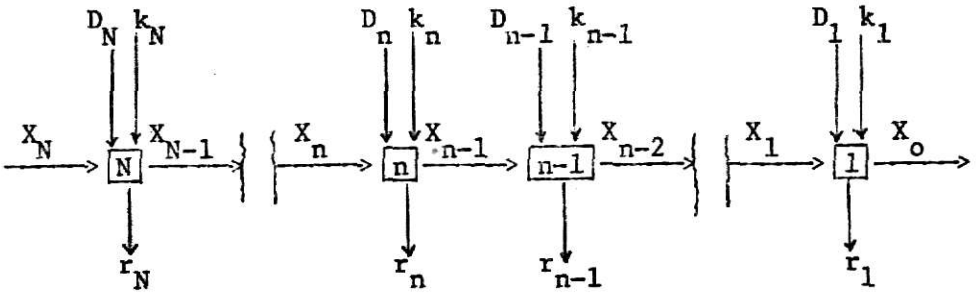
with p(z) being the probability density function of z. Under this scheme a decision policy D* is a global maximum under risk if and only if $\bar{r}(D^*) \geq \bar{r}(D)$ for all feasible D.

Maximization of the expected return function under risk has created certain objections which led to the introduction of the concept called "utility". Savage (1954) defines "utility" as a function that quantifies the relation of preference among several courses of action. A simple example offered by Daniel Bernoulli (cited in Stingler (1950)), who was among the first to develop the idea of utility, states the objection of maximizing the return function:

> "Suppose a pauper happens to acquire a lottery ticket by which he may with equal probability win either nothing or 20,000 ducats. Will he have to evaluate the ticket as 10,000 ducats; and would he be acting foolishly, if he sold it for 9,000 ducats?"

Bernoulli claimed that a dollar which is worthless to a millionare would be precious to a pauper. Therefore, the returns may be measured in ducats, but the number of ducats is not equivalent to the utility of ducats. So, the return function may contain factors other than the return of the number of ducats. These objections do not concern the use of the expected return, but rather the measurement of the return. It will be assumed in the further development of stochastic programming (SP) that the measure is correct, thereby equating "value" and "utility" of the return function. A development of optimization under risk using the (DP) technique for a multistage problem will now be considered.

The N-stage stochastic (DP) system is similar to the N-stage deterministic (DP) system, only at each stage there is an additional variable containing the stochastic element which affects the stage transformation and return function.



At stage n = 1, ..., N there is an input stage variable X_n, a decision variable D_n, and a random variable k_n, which determine the return

$$r_n = r_n(X_n, D_n, k_n). \tag{3.1}$$

The transformation becomes

$$X_{n-1} = t_n(X_n, D_n, k_n), \quad n = 1, \ldots, N.$$

Further, the k_i, i = 1, ..., N are independently distributed with probability density function $p_i(k_i)$, i = 1, ..., N respectively. Also, the total return function R_N is the sum of the individual stage returns, r_n. That is,

$$\left.\begin{aligned} &R_N(X_N, \ldots, X_1; D_N, \ldots, D_1; k_N, \ldots, k_1) = \sum_{n=1}^{N} r_n(X_n, D_n, k_n) \\ &\text{s.t.:} \quad X_{n-1} = t_n(X_n, D_n, k_n). \end{aligned}\right\} \tag{3.2}$$

Note that (3.1) is dependent upon the random variables $k_N, \ldots, k_{n+1}$ in addition to k_n, since X_n depends upon $k_N, \ldots, k_{n+1}$. Recall that in the deterministic case, $r_n(X_n, D_n)$ was dependent only on $X_N, D_N, \ldots, D_n$. However, for the stochastic system X_n depends upon previously observed random variables, $k_N, \ldots, k_{n+1}$ as well as $D_N, \ldots, D_{n+1}$. Thus, even if a decision policy is given, the input for the nth stage is unknown until $k_N, \ldots, k_{n+1}$ are realized.

The expected value of the total return function R_N requires the expectation over more than one random variable. Given a function $r(D, k_1, \dots, k_N)$, where $k_1, \dots, k_N$ are independently distributed random variables with probability density functions $p_1(k_1), \dots, p_N(k_N)$ respectively, the expected value of $r(D, k_1, \dots, k_N)$ is

$$\bar{r}(D) = \sum_{k_1} \cdots \sum_{k_N} \left(\left\{ \prod_{n=1}^{N} p_n(k_n) \right\} r(D, k_1, \dots, k_N) \right). \tag{3.3}$$

The expected value of the total return function $R_N(X_N;D_N,\dots,D_1;k_N,\dots,k_1)$ is given by

$$\begin{aligned} \bar{R}_N(X_N;D_N,\dots,D_1) &= \sum_{k_1} \cdots \sum_{k_N} \left(\prod_{n=1}^{N} p_n(k_n) \right) \left(\sum_{n=1}^{N} r_n(X_n,D_n,k_n) \right) \\ &= \sum_{k_1} \cdots \sum_{k_N} \left(\prod_{n=1}^{N} p_n(k_n) \right) \left(r_N(X_N,D_N,k_N) \right) + \cdots \\ &\quad + \sum_{k_1} \cdots \sum_{k_N} \left(\prod_{n=1}^{N} p_n(k_n) \right) \left(r_1(X_1,D_1,k_1) \right) \end{aligned} \tag{3.4}$$

$$\text{s.t.:} \quad X_{n-1} = t_n(X_n,D_n,k_n), \quad n = 1, \dots, N.$$

Since the Nth stage return does not depend on $k_{N-1}, \dots, k_1$,

$$\begin{aligned} \bar{R}_N(X_N;D_N,\dots,D_1) &= \sum_{k_N} \left\{ p_N(k_n) r_n(X_n,D_n,k_n) \left\{ \sum_{k_{N-1}} p_{N-1}(k_{N-1}) \cdots \sum_{k_1} p_1(k_1) \right\} \cdots \right\} \\ &\quad + \sum_{k_N} \left\{ p_N(k_N) \cdots \sum_{k_2} \left\{ p_2(k_2) \sum_{k_1} [p_1(k_1) r_1(X_1,D_1,k_1)] \right\} \cdots \right\}. \end{aligned} \tag{3.5}$$

$$\text{s.t.:}\quad X_{n-1} = t_n(X_n,D_n,k_n),\quad n = 1, \ldots, N.$$

The intention is to maximize the expected N stage return over $D_N, \ldots, D_1$. Let $\bar{f}_N(X_N)$ be the maximum expected return as a function of X_N. Then

$$\bar{f}_N(X_N) = \max_{D_N,\ldots,D_1} \bar{R}_N(X_N;D_N,\ldots,D_1) \tag{3.6}$$

$$= \max_{D_N,\ldots,D_1} \Bigg\{ \sum_{k_N} p_N(k_N) r_N(X_N,D_N,k_N)$$

$$+ \sum_{k_N} \Bigg\{ p_N(k_N) \sum_{k_{N-1}} \Big\{ p_{N-1}(k_{N-1}) r_{N-1}(X_{N-1},D_{N-1},k_{N-1}) \Big\}\Bigg\} + \cdots$$

$$+ \sum_{k_N} \Bigg\{ p_N(k_N) \cdots \sum_{k_2} \Big\{ p_2(k_2) \sum_{k_1} p_1(k_1) r_1(X_1,D_1,k_1) \Big\} \cdots \Bigg\} . \tag{3.7}$$

Since $p_N(k_N)$ is common to every term, factoring gives

$$\bar{f}_N(X_N) = \max_{D_N,\ldots,D_1} \Bigg\{ \sum_{k_N} p_N(k_N) \Bigg\{ r_N(X_N,D_N,k_N)$$

$$+ \sum_{k_{N-1}} \Big\{ p_{N-1}(k_{N-1}) r_{N-1}(X_{N-1},D_{N-1},k_{N-1}) \Big\} + \cdots$$

$$+ \sum_{k_{N-1}} \Big\{ p_{N-1}(k_{N-1}) \cdots \sum_{k_2} \Big\{ p_2(k_2) \sum_{k_1} p_1(k_1) r_1(X_1,D_1,k_1) \Big\} \cdots \Big\}\Bigg\}\Bigg\} \tag{3.8}$$

$$\text{s.t.:}\quad X_{n-1} = t_n(X_n,D_n,k_n),\quad n = 1, \ldots, N. \tag{3.9}$$

Using the procedure of the deterministic recursive equations, (3.6) is replaced by

$$\bar{f}_N(X_N) = \max_{D_N} \max_{D_{N-1}} \bar{R}_N(X_N;D_N,\ldots,D_1). \tag{3.10}$$

Since the Nth stage return is not a function of $D_{N-1},\ldots,D_1$, it can be removed from the inner maximization to give

$$\begin{aligned}
\bar{f}_N(X_N) &= \max_{D_N} \max_{D_{N-1}} \bar{R}(X_N;D_N,\ldots,D_1) \\
&= \max_{D_N} \Bigg\{ \sum_{k_N} p_N(k_N) \Bigg(r_N(X_N,D_N,k_N) \\
&\quad + \max_{D_{N-1},\ldots,D_1} \Bigg\{ \sum_{k_{N-1}} \Bigg(p_{N-1}(k_{N-1}) r_{N-1}(X_{N-1},D_{N-1},k_{N-1}) \Bigg) \\
&\quad + \cdots + \sum_{k_{N-1}} \Bigg(p_{N-1}(k_{N-1}) \cdots \sum_{k_1} \Bigg(p_2(k_2) \sum_{k_1} \Bigg(p_1(k_1) r_1(X_1,D_1,k_1) \Bigg)\Bigg) \cdots \Bigg)\Bigg)\Bigg)\Bigg).
\end{aligned} \tag{3.11}$$

Now,

$$\begin{aligned}
\bar{f}_{N-1}(X_{N-1}) &= \bar{f}_{N-1}(t_N(X_N,D_N,k_N)) \\
&= \max_{D_{N-1},\ldots D_1} \Bigg\{ \sum_{k_{N-1}} \Bigg(p_{N-1}(k_{N-1}) r_{N-1}(X_{N-1},D_{N-1},k_{N-1}) \Bigg) + \cdots \\
&\quad + \sum_{k_{N-1}} \Bigg(p_{N-1}(k_{N-1}) \cdots \sum_{k_2} \Bigg(p_2(k_2) \sum_{k_1} \Bigg(p_1(k_1) r_1(X_1,D_1,k_1) \Bigg)\Bigg) \cdots \Bigg)\Bigg) .
\end{aligned}$$

So, the recursive equation is

$$\bar{f}_N(X_N) = \max_{D_N} \left\{ \sum_{k_N} p_N(k_N) \left[r_N(X_N,D_N,k_N) + \bar{f}_{N-1}(t_N(X_N,D_N,k_N)) \right] \right\}.$$

Inductively, for any stage $1 \leq n \leq N$, the fundamental stochastic recursive equations are

$$\bar{f}_n(X_n) = \max_{D_N} \left\{ \sum_{k_n} p_n(k_n) \; Q_n(X_n,D_n,k_n) \right\}, \quad 1 \leq n \leq N$$

with

$$Q_n(X_n,D_n,k_n) = r_n(X_n,D_n,k_n) + \bar{f}_{n-1}(t_n(X_n,D_n,k_n), \quad 2 \leq n \leq N$$

and

$$Q_1(X_1,D_1,k_1) = r_1(X_1,D_1,k_1).$$

The optimal decision policy resulting from multistage optimization under the risk is itself stochastic with the exception of the first optimal decision $D_N^*(X_N)$. The remaining optimal decisions from the recursive analysis $D_{N-1}(X_{N-1}),\ldots, D_1(X_1)$ cannot be expressed deterministically in terms of X_N until the values of the preceeding random variables are known. So in substituing $D_N^*(X_N)$ into (3.9) the optimal value of X_{N-1}^* is known only probabilistically as

$$X_{N-1}^* = t_N(X_N,D_N^*(X_N),k_N) = t_N(X_N,k_N).$$

Thus, the optimal value of D_{N-1} will be known only probabilistically also, since

$$D_{N-1}(X_{N-1}^*) = D_{N-1}(t_N(X_N,k_N)) = D_{N-1}(X_N,k_N) \; .$$

Therefore, this N stage stochastic optimization technique gives incomplete results, since only the first optimal decision is found. The remaining optimal decisions can be found one at a time as the random variables take on values. Thus, the policy of assigning the random variables their expected values in order to obtain an optimal decision policy seems appropriate. In a special case of quadratic stage returns and linear transformations, Tou (1963) showed that independent random variables k_n, $n = 1, \ldots, N$ could be replaced by their expected values $(\bar{k}_n)$,

$$\bar{k}_n = \sum_{k_n} p_n(k_n)k_n, \quad n = 1, \ldots, N.$$

Then $D_n^*(X_n)$ could be obtained from the deterministic problem

$$\bar{f}_n(X_n) = \max_{D_n} Q_n(X_n, D_n, \bar{k}_n), \quad 1 \leq n \leq N$$

where

$$Q_n(X_n, D_n, \bar{k}_n) = r_n(X_n, D_n, \bar{k}_n) + f_{n-1}(t_n(X_n, D_n, \bar{k}_n)), \quad 2 \leq n \leq N$$

and

$$Q_1(X_1, D_1, \bar{k}_n) = r_1(X_1, D_1, \bar{k}_1) .$$

However, it should be noted that in a general stochastic problem replacing the random variables by their expected values may lead to erroneous results.

The composition operator "0" was used before in the general scheme of combining stage returns to derive the basic recursive equation for deterministic multistage optimization problems. This general operator is not possible for stochastic optimization as becomes evident in letting "0" be the product of stage returns. The recursive equations still apply for maximizing the product of stage returns, but do not apply to maximizing the minimum individual stage returns. This comes about from

$$\bar{R}_2(X_2,X_1;D_2,D_1) = \sum_{k_2}\sum_{k_1} p_2(k_2)p_1(k_1) \cdot \min\,(r_2(X_2,D_2,k_2),r_1(X_1,D_1,k_1))$$

$$\text{s.t.:}\quad X_1 = t_2(X_2,D_2,k_2)$$

instead of what is required from the recursive equations

$$\bar{R}_2 = (X_2,X_1;D_2,D_1) = \sum_{k_2} p_2(k_2)\Big(\min\Big(r_2(X_2,D_2,k_2),\sum_{k_1} p_1(k_1)r_1(X_1,D_1,k_1)\Big)\Big).$$

Thus, decomposition is not possible for multiplication of stage returns for the stochastic optimization problem.

The development of (DP) under risk by Nemhauser (1966) considers the random variables to be identically distributed with known discrete probability density functions. If the random variables are continuous identically distributed independent random variables, what modifications would be necessary to the development under the discrete random variable case? First, the total return function from N stages would continue to be as before, i.e., (3.2). The expected return would be given by

$$\bar{r}(D) = \int_{k_1}\cdots\int_{k_N}\left(\prod_{n=1}^{N}\phi_n(k_n)\right)r(D,\ k_1,\ \ldots,\ k_N)\ dk_N\ dk_{N-1}\cdots dk_1,$$

where $\phi_n(k_n)$ is the probability density function of k_n and the integrals are assumed to exist. By replacing the sums over k_i, $i = 1, \ldots, N$ by integrals over k_i, $i = 1, \ldots, N$, the whole development from the discrete case extends to the final result

$$\bar{f}_n(X_n) = \max_{D_n} \int_{k_n} \phi_n(k_n)\ \phi_n(X_n,D_n,k_n)dk_n, \quad 1 \leq n \leq N$$

where

$$\phi_n(X_n,D_n,k_n) = r_n(X_n,D_n,k_n) + \bar{f}_{n-1}(t_n(X_n,D_n,k_n), \quad 2 \leq n \leq N$$

and

$$\phi_1(X_1,D_1,k_1) = r_1(X_1,D_1,k_1) \ .$$

If the random variables are continuously distributed a recursive relationship can be developed, assuming the existence of the integral, but other computational methods must be used to either find the optimum or obtain a satisfactory discrete approximation.

Now that the formulation of (DP) under risk has been introduced it is of importance to further amplify the different classes of problems that can be considered in the (DP) framework.

Consider first the reason for even having the (DP) technique at all. The ordinary calculus methods of maximizing functions are dependent upon the continuous variation of the independent variables and at best give relative maxima or minima over a closed interval and does not consider the boundary points. Where constraints on the continuous function are concerned the Lagrange multiplier method will account for these problems. Since many of the problems considered will be over closed intervals of the independent variables and a global maximum is desired the methods of calculus may be unsatisfactory. Further if the function considered is discrete generally new tools of optimization are needed. Finally, problems of interest are many times of high dimensionality which causes difficulties in calculus from having to evaluate all combinations of feasible values of the independent variables to find the solution to optimizing the objective function.

Dynamic programming obtains the global optimum. Any constraints to (DP) simplify the search process and limits the number of possibilities at each stage, thereby reducing the computations. Dynamic programming formulations usually lead to equations which cannot generally be solved analytically but which are well suited to numerical solution using a digital computer (Jacobs, 1967). The procedure for reducing multistage decision problems to equations that may be solved numerically is fundamental to (DP).

Multistage decision problems constitute an area where (DP) is used quite extensively. It is possible to consider multistage decision processes where the number of stages essentially is infinite. These problems are known as infinite stage systems and can arise in two fundamentally different ways:

(i) by simply letting $N \to \infty$ which can be represented by the model

$$\left.\begin{aligned} &\max \sum_{n=1}^{\infty} r_n(X_n, D_n) \\ &\text{s.t.:}\quad X_{n-1} = t_n(X_n, D_n),\ n = 1,\ 2,\ \ldots, \end{aligned}\right\} \qquad (3.12)$$

which is called an "infinite planning horizon (Gue & Thomas, 1968),

(ii) assuming the stages to correspond to time periods, the horizon is finite, but the time periods are infinitesimally small. Thus, the time between successive decisions is negligible compared with the horizon. In the limit, therefore, the decisions are made continuously, and for any finite horizon there are an infinite number of decisions.

Regarding the model (3.12), it is meaningful if for all feasible combinations of the decisions the total return function is bounded from above. An example from Bellman & Dreyfus (1962) of a specific (DP) approach to a problem under risk lends insight to the infinite stage system.

Consider a transport plane dispatching to an overseas base with a cargo consisting of replacement parts for airplanes. Suppose there are N types of replacement parts, and associated with each is a cost incurred if the part is needed at the base, but not available. Let the demand for each part be Poisson distributed with known λ. The weight capacity W and available space S of the cargo vehicle become constraints of the problem. The problem is to determine the number of items of each type to be dispatched so as to minimize the expected cost due to shortages at the base.

The following notation will be used for the problem:

(i) w_i: to be the weight of each item of the ith type,

(ii) s_i: to be the volume of each item of the ith type,

(iii) c_i: the cost peritem for not fulfilling the demand,

(iv) λ_i: the mean value of the Poisson distribution representing the demand for items of the ith type.

Let x_i be the number of items of the ith type which are loaded and p(z) be the probability of a demand for z items of the ith type. Then the expected cost for items of the ith type due to unfulfilled demand will be

$c_i \sum_{z=x_i+1}^{\infty} (z-x_i)\, p(z)$. Since demand is distributed Poisson with mean λ, denoted by $p(z,\lambda_i)$ for the ith item, the total expected cost, E_N, becomes

$$E_N = \sum_{i=1}^{N} i \left(\sum_{z=x_i+1}^{\infty} (z-x_i)\ p(z,\lambda_i) \right).$$ Therefore, the problem is to minimize E_N overall x_i

$$\text{s.t.} \quad x = 0, 1, 2, \ldots,$$

$$\sum_{i=1}^{N} x_i w_i \leq W$$

$$\sum_{i=1}^{N} x_i s_i \leq S\ .$$

The (DP) formulation of the problem follows:

Define $f_k(w',s')$ as cost associated with an optimal decision policy of items of the first k types, with a cargo - vehicle restriction of w' and space limitation s', $0 \leq w' \leq w$ and $0 \leq s' \leq s$. The basic recurrent relation is then

$$f_k(w',s') = \min_{x_k} \left\{ k \sum_{z=x_{k+1}}^{\infty} (z-x_k)\ p(z,\lambda_k) + f_{k-1}(w' - x_k w_k,\ s'-x_k s_k) \right\}$$

$$\text{s.t.} \quad 0 \leq x_k \leq \min \left\{ \left[\frac{w'}{w_k}\right],\ \left[\frac{s'}{s_k}\right] \right\}, \text{ where}$$

[x] denotes the greatest integer $\leq$ x.

$$x = 0, 1, 2, \ldots$$

$$\sum_{i=1}^{N} x_i w_i \leq W$$

$$\sum_{i=1}^{N} x_i s_i \leq S\ .$$

The solution to this example is obtained by considering feasible combinations in integer programming and can be found in Bellman & Dreyfus (1962).

The problems which make use of (DP) have essentially been discrete relationships such as step functions. A model of a continuous multistage process where the horizon is finite but the time periods are infintestimally small, analogous to the discrete multistage process

$$\max_{D_n} \sum_{n=1}^{N} r_n(x_n, D_n)$$

$$\text{s.t.} \quad x_{n-1} = t_n(x_n, D_n), \quad n = 1, \ldots, N$$

$$x_N = k$$

is

$$\max_{U(t)} \int_{t_1}^{t_2} F(t,x,U)dt$$

$$\text{s.t.} \; \frac{dx}{dt} = g(t,x,U), \; t_1 \leq t \leq t_2$$

$$x_1 = x(t_1) = k.$$

The determination of a function which optimizes an integral is a problem in the calculus of variations. Dynamic programming and the calculus of variations have a close relationship. The formal relationship is that a dynamic programming approach can be used to derive the necessary conditions for an optimum; and in addition, discrete recursive optimization can be used to find approximate numerical solutions to computationally difficult variational problems.

The computational technique of (DP) has no general algorithm of problem solution as does the (LP) approach with the simplex method. Further, the purpose of this report is to present the general formulation of the technique of (DP) and the introduction of random variates into the multistage decision process. From a review of the literature there has been little work on the general introduction of random variates into the (DP) scheme. Many specific problems with many different conditions have been considered. Yet a general development of stochastic programming with emphasis on the role of the random variable seems to be lacking.

3.2 Dynamic Programming and Markov Processes

One area where considerable development has been done with emphasis on the role of the random variates in the (DP) technique using Markov processes as a system model is given in a monograph by Howard (1960). Markovian decision processes are considered an important class of stochastic optimization problems. Markov processes are defined as a collection of random variables $\{x(t), t\varepsilon T\}$, where the random variables are defined over the index set T, and if, for any set of n points

$t_1 < t_2 < \cdots < t_n$, $t_i \varepsilon T$, $i = 1, \ldots, n$, the conditional distribution of $x(t_n)$, for given values of $x(t_1), \ldots, x(t_{n-1})$, depends only on $x(t_{n-1})$. The classification of Markov processes depends upon the index set being continuous or discrete, and the nature of the state space of the process. The set of possible values of a stochastic process is the state space and is usually taken to be either discrete or continuous. If the state space contains a finite or countably infinite number of values it is called discrete and the stochastic process is called a a Markov chain. If the index set is the nonnegative integers the Markov chain is a discrete parameter Markov chain (Parzen, 1962). The development of (DP) and Markov processes considers initially the discrete Markov chain as a systems model.

A Markov process is described by a *transition probability function*, $P(x,t_o;E,t)$ or $P(E,t|x,t_o)$, which represents the conditional probability that the state of the system will at time t belong to the set E, given that at time $t_o<t$ the system is in state x. The Markov process is defined to have stationary transition probabilities if $P(x,t_o;E,t)$ depends only on t and t_o only through the difference $(t-t_o)$ (Parzen, 1962). Stationarity in the transition probabilities is also incorporated in the initial consideration of (DP) and Markov processes. Therefore, the system model is a finite stochastic process in which the state of the system at any stage depends only on the state of the system at the previous stage and on a known probability transition matrix. Denoting the states at stage n by i, $i = 1, \ldots, M$, $n = 1, \ldots, N$, the probability of going from state i at stage n to state j at stage n-1 is given by P_{ij}. The set of transition probabilities can be represented in matrix form as

$$\underset{\sim}{P} = \begin{bmatrix} p_{11} & \cdots & p_{1j} & \cdots & p_{1M} \\ \vdots & & \vdots & & \vdots \\ p_{i1} & \cdots & p_{ij} & \cdots & p_{iM} \\ \vdots & & \vdots & & \vdots \\ p_{M1} & \cdots & p_{Mj} & \cdots & p_{MM} \end{bmatrix} ,$$

called the probability transition matrix. The properties of the transition matrix are $0 \leq p_{ij} \leq 1$ and $\sum_j p_{ij} = 1$. The probability of being in state j at stage n-1, denoted by $\Pi_{n-1}(j)$ is found by multiplying the probability of being in state i at stage n ($\Pi_n(i)$) by the transition probability p_{ij} and summing over all states at stage n, i.e.

$$\Pi_{n-1}(j) = \sum_{i=1}^{M} p_{ij} \, \Pi_n(i), \quad j = 1, \ldots, M \qquad (3.13)$$
$$n = 1, \ldots, N .$$

The solution of (3.13) gives the state probabilities after n stages as a function of the initial state, and the limiting state probabilites (steady state probabilities) if they exist as n increases without bound. The functional relationship on the initial state and steady state probabilities is made more obvious in vector form. Define a row vector of state probabilities $\underset{\sim}{\Pi}_n$ with components $\Pi_n(i)$, then $\underset{\sim}{\Pi}_{n-1} = \underset{\sim}{\Pi}_n \underset{\sim}{P}$, n = N, N-1,...,1. Since by recursion $\underset{\sim}{\Pi}_{N-1} = \underset{\sim}{\Pi}_N \underset{\sim}{P}$ $\underset{\sim}{\Pi}_{N-2} = \underset{\sim}{\Pi}_{N-1} \underset{\sim}{P} = \underset{\sim}{\Pi}_N \underset{\sim}{P}^2$ - - - - - - - $\underset{\sim}{\Pi}_0 = \underset{\sim}{\Pi}_1 \underset{\sim}{P} = \Pi_N P^N$. In general the form would be $\underset{\sim}{\Pi}_{n-1} = \Pi_N \cdot P^{N-(n-1)}$, n = N, ..., 1 which is not as simple in form as the forward recursion.

Suppose that an n-state Markov process has a return r_{ij} associated with transitions from state i to state j. Howard (1960) calls $\underset{\sim}{R} = (r_{ij})$, i = 1, ..., M; j = 1, ..., M, the "reward" matrix which corresponds to the transition matrix $\underset{\sim}{P}$. The total expected return from an n-stage Markov process starting in state i, recursively is the expected value of the return from stage n plus the expected (n-1) stage return from the resulting state, summed over all states. For a one-stage process

$$\bar{R}_1(i) = \sum_{j=1}^{M} p_{ij}\ r_{ij},$$

or in matrix form by denoting $\underset{\sim}{p}(i)$ as a row vector of $\underset{\sim}{P}$, $\underset{\sim}{r}(i)$ as a row vector of $\underset{\sim}{R}$, then

$$\underset{\sim}{\bar{R}}_1 = \bar{R}_1(i)_{i=1,\ldots,M} = \underset{\sim}{p}(i)\underset{\sim}{r}'(i), \quad i = 1, \ldots, M; \text{ gives diagonal elements of } \underset{\sim}{P}\underset{\sim}{R}'.$$

For an n-stage process

$$\bar{R}_n(i) = \sum_{j=1}^{M} p_{ij}[r_{ij} + \bar{R}_{n-1}(j)] \quad n = 2, \ldots, N$$

is the total expected return. A matrix form for the recursive expression can be found as follows:

$$\bar{R}_n(i) = \sum_{j=1}^{M} p_{ij}\ r_{ij} + \sum_{j=1}^{M} p_{ij}\ \bar{R}_{n-1}(j)$$

$$= \underset{\sim}{p}(i)\ \underset{\sim}{r}'(i) + \underset{\sim}{p}(i)\ \underset{\sim}{\bar{R}}'_{n-1}, \qquad (3.14)$$

where $\underset{\sim}{\bar{R}}_{n-1} = (\underset{\sim}{\bar{R}}_{n-1}(j))$, j = 1, ..., M.

$$= q(i) + \underset{\sim}{p}(i)\ \underset{\sim}{R}'_{n-1}, \quad i = 1, \ldots, M; \quad n = 2, \ldots, N. \qquad (3.15)$$

$$\underset{\sim}{\bar{R}}_n = \underset{\sim}{q} + \underset{\sim}{P}\ \underset{\sim}{R}'_{n-1}, \quad n = 2, \ldots, N. \qquad (3.16)$$

The quantity q_i can be interpreted as the reward to be expected in the next transition out of state i. It is not necessary to specify both a $\underset{\sim}{P}$ matrix and an $\underset{\sim}{R}$ matrix in order to determine the expected return from the system. All that is required is a $\underset{\sim}{P}$ matrix and a $\underset{\sim}{q}$ column vector with components q_i. When large problems are to be solved the data storage is reduced considerably by a q column vector rather than the $\underset{\sim}{R}$ matrix. Also, the index of n varies in Howard (1960) and Nemhauser (1966). Nemhauser assumes that $\underset{\sim}{\bar{R}}_o$ is arbitrarily zero, and can therefore set the index set for n as {2, ..., N}. On the other hand Howard (1960) sets the lower boundary of n as 1 and $\underset{\sim}{\bar{R}}_o$, the boundary expected return when the system ceases operation may represent an expected return from selling the system after N-stages have been completed.

To incorporate decision making into the system a set of transition matrices and return matrices are possible at each stage, and to optimize the system a decision as to what $\underset{\sim}{P}$ and $\underset{\sim}{R}$ to use at each stage must be made. A decision variable $d_n = k$, $k = 1, \ldots, K$ denotes a choice of the kth transition matrix and ith return matrix at the nth stage. Specifically, if the state of the system is i, $d_n = k$ means that the ith row of the kth $\underset{\sim}{P}$ and the ith row of the kth $\underset{\sim}{R}$ give the relevant transition probabilities and returns at stage n. In general the assumption that the same set of $\underset{\sim}{P}$ and $\underset{\sim}{R}$ matrices are available for each stage or that $\underset{\sim}{P}$ and $\underset{\sim}{R}$ are chosen in pairs rather than seperately need not be made. Only for simplicity are these assumptions made here. The transition probability is denoted by $p_{ij}(d_n)$ and the associated return by $r_{ij}(d_n)$. The expected return from n stages, starting in state i, is

$$\bar{R}_1(i,d_1) = \sum_{j=1}^{M} p_{ij}(d_1)\ r_{ij}(d_1) = \underset{\sim}{p}(i,d_1)\underset{\sim}{r}'(i,d_1)$$

and

$$\bar{R}_n(i,d_n,\ldots,d_1) = \sum_{j=1}^{M} p_{ij}(d_n)[r_{ij}(d_n) + \bar{R}_{n-1}(j,d_n,\ldots,d_1)],n=2,\ldots,N,$$

or

$$\bar{R}_n(i,d_n,\ldots,d_1) = q(i;d_n) + \underset{\sim}{p}(i)\ \underset{\sim}{\bar{R}}'_{n-1}(\ \ d_n,\ldots,d_1)],i=1,\ldots,M;n=2,\ldots,N,$$

or

$$\underset{\sim}{\bar{R}}_n(d_n,\ldots,d_1) = \underset{\sim}{q}(d_n) + \underset{\sim}{P}(d_n)\ \underset{\sim}{\bar{R}}'_{n-1}(\ \ d_n,\ldots,d_1),\ n = 2,\ldots,N\ .$$

The expected return matrix $\underset{\sim}{\bar{R}}_n(d_n,\ldots,d_2)$ is a column vector with each component being $\bar{R}_n(i,d_n,\ldots,d_1)$.

Following the development of multistage stochastic optimization models presented previously, the following notation of Nemhauser (1966) will be used:

(i) Markovian return matrix $r_{ij}(d_n) = r_n(x_n,D_n,k_n)$

(ii) Markovian transition matrix $p_{ij}(d_n) = t_n(x_n,D_n,k_n)$, where in the Markovian model

(i') The state variable x_n is represented by i, the state of the Markovian system at the nth stage,

(ii') The decision variable D_n is represented by d_n=k,

$k = 1, \ldots, K$, and represents the choice of particular transition and return matrices, and

(iii') The random variable k_n is completely hidden in the new notation. It is a chance factor which determines the output state j, given the input state and decision. The distribution of the random variable for the ith input state is given by the ith row of the transition matrix.

To maximize the expected return from N stages as a function of the initial state, let

$$\bar{f}_N(i) = \max_{d_N,\ldots,d_1} \sum_{j=1}^{M} p_{ij}(d_N)[r_{ij}(d_N) + \bar{R}_{N-1}(j,d_N,\ldots,d_1)].$$

Following the usual recursive scheme,

$$\bar{f}_1(i) = \max_{d_1=1,\ldots,K} \sum_{j=1}^{M} p_{ij}(d_1)\ r_{ij}(d_1)$$

and

$$\bar{f}_n(i) = \max_{d_n=1,\ldots,K} \sum_{j=1}^{M} p_{ij}(d_n)[r_{ij}(d_n) + \bar{f}_{n-1}(j)], \quad n = 2,\ldots,N.$$

Since $\sum_{j=1}^{M} p_{ij}(d_n)\ r_{ij}(d_n)$ is the ith element of $\underset{\sim}{q}(d_n)$ and has previously been denoted $q(i;d_n)$, the recursive relation becomes

$$\bar{f}_1(i) = \max_{d_1=1,\ldots,K} q(i;d_1), \quad i = 1, \ldots, M$$

and

$$\bar{f}_n(i) = \max_{d_n=1,\ldots,K} \left\{ q(i;d_n) + \sum_{j=1}^{M} p_{ij}(d_n)\, \bar{f}_{n-1}(j) \right\}, \quad i = 1, \ldots, M, \quad n = 1, \ldots, N$$

or

$$\bar{f}_n(i) = \max_{d_n=1,\ldots,K} \left\{ q(i;d_n) + \underset{\sim}{P}(i;d_n)\, \bar{f}'_{n-1} \right\} \quad i = 1, \ldots, M, \quad n = 1, \ldots, N$$

where $\underset{\sim}{\bar{f}}_{n-1}$ is a row vector with elements $\bar{f}_{n-1}(j)$, $j = 1, \ldots, M$.

This method for solution of a sequential process may be called the value-iteration process (Howard, 1960) because the $\bar{f}_n(i)$ (Nemhauser, 1966) or $v_i(n)$ (Howard, 1960) are "values" which are determined iteratively. This method has important limitations. Not many processes operate with the specter of termination so imminent. For the most part, systems operate on an indefinite basis with no clearly defined end point (upper bound on n)(Howard, 1960). Further, it does not seem efficient to have to iterate $\bar{f}_n(i)$ for n = 1, 2, 3, ..., until such a sufficiently large n so that termination is very remote. A method of analyzing processes of indefinite iteration would be more satisfactory. This implies the development of some Markov processes possessing long run or steady state distributions into the (DP) scheme. Even if the long-duration process is solved by value iteration, the convergence on the best alternative in each state is asymptotic and difficult to measure analytically.

Consider a system which has an indefinite number of stages where a Markov chain is used as a model. Again, decision making arises when a choice is made between various transition and return matrices.

Following Howard, now consider the system with an infinite number of stages. A Markov chain with an infinite number of transitions serves as a model for this case. Further, the model is to be a completely ergodic Markov chain, which means that the limiting state probability distribution is independent of starting conditions.

For completely ergodic Markov processes, a quantity $\Pi(i)$ is defined as the probability that the system occupies the ith state after a large number of moves regardless of the initial state. Recall that the probability of being in state j at stage n-1, denoted by $\Pi_{n-1}(j)$, is given by the simultaneous linear difference equations,

$$\Pi_{n-1}(j) = \sum_{i=1}^{M} p_{ij}\Pi_n(i), \quad j = 1, \ldots, M.$$

Suppose there are N stages, so the vector Π_N denotes the starting state; then under special conditions on the transition matrix $\underset{\sim}{P}$

$$\lim_{n,N\to\infty} \Pi_{N-n}(i) = \Pi(i)$$

exists independently of the starting state Π_N. Markov chains with this property are called completely ergodic and are quite common in practice. The special condition on the transition matrix $\underset{\sim}{P}$ to have a completely ergodic Markov chain is the following:

A necessary and sufficient condition for complete ergodicity is that some power of $\underset{\sim}{P}$, i.e. $\underset{\sim}{P}^n$ where n is any integer, contain only positive elements. So, if $\underset{\sim}{P}$ contains only positive elements, the Markov chain is completely ergodic.

If the limit of the state probability equations is taken with respect to n, the equations become

$$\Pi(j) = \sum_{i=1}^{M} p_{ij}\, \Pi(i), \quad j = 1, \ldots, M.$$

There are M linear, homogeneous equations in M unknowns. The system of equations has M-1 linearly independent equations. Thus, M-1 of the equations with the condition that $\sum_{j=1}^{M} \Pi(j) = 1$ determines the steady state probabilities.

To proceed recursively, returns are introduced into the infinite stage Markov process. Let the return from a transition from state i to j be r_{ij}. For a completely ergodic Markov process, the expected return per stage in terms of steady state transitions is

$$g = \sum_{i=1}^{M} \sum_{j=1}^{M} r_{ij}\, p_{ij}\, \Pi(i), \tag{3.17}$$

or

$$g = \sum_{i=1}^{M} \Pi(i)\ q_i, \tag{3.18}$$

where q_1 is defined by (3.14) and (3.15). The reasoning for the equations is the following:

(i) The probability of being in state i in the steady state (long run distribution) is $\Pi(i)$.

(ii) Thus, the steady state probability of a transition from state i to state j is $p_{ij}\ \Pi(i)$,

(iii) and the expected return from a transition from state i to state j, in a steady state, is $r_{ij}\ p_{ij}\ \Pi(i)$.

(iv) Thus summing over i and j gives (3.17). This reasoning is a result of asymptotic behavior of a z-transform of the Markov process.

Therefore, the expected return from a single transition is found by multiplying the immediate expected return from a state (q_i) by the steady state probability of being in that state ($\Pi(i)$) and then summing over all states.

Since g is in terms of steady state probabilities it is independent of the starting state. However, the total expected return is not independent of the starting state. The total expected return for a system in state i with n transitions remaining, denoted by $v_n(i)$, is given by

$$\left.\begin{array}{l} v_1(i) = q_i, \quad i = 1, \ldots, M \\ \text{and} \\ v_n(i) = q_i + \sum_{j=1}^{M} p_{ij}\ v_{n-1}(j), \quad i = 1, \ldots, M. \end{array}\right\} \tag{3.20}$$

Since the expected return from a single transition is g for large n, $v_n(i)$ is a linear function of n with slope g; i.e.

$$v_n(i) = ng + v_i, \quad i = 1, \ldots, M,$$

or in matrix form

$$\underset{\sim}{v}(n) = n\underset{\sim}{g} + \underset{\sim}{v}.$$

Note that for the completely ergodic Markov process all states have the same gain. The intercept v_i depends only on the starting state. Also,

$$v_n(i) - v_n(j) = v_i - v_j$$

so that $v_i - v_j$ is a measure of the relative advantage or disadvantage of starting in state i rather than j.

The above development of the recursive equations (Nemhauser, 1966) state the end results of applying a z-transformation to the Markov process and noting the asymptotic behavior of the process in terms of the generating function, (the z-transformation) (Howard, 1960).

Using the approximation for $v_n(i)$ for large n in the defining equations for $v_n(i)$ gives

$$ng + v_i = q_i + \sum_{j=1}^{M} p_{ij} \, [(n-1)\, g + v_j]$$

or

$$v_i = q_i + \sum_{j=1}^{M} p_{ij} \, v_j - g, \quad i = 1, \ldots, M. \tag{3.21}$$

There are M simultaneous linear equations in M+1 unknowns $(v_1, \ldots v_M, g)$. The value of g can be determined independently of

(3.21) by (3.17), and (3.18). The equations reduce to having M unknowns, but still cannot be solved for absolute values of v_i. The M unknowns have only M-1 linearly independent equations. By fixing a v_i, say $v_M = 0$, the remaining v_i's will be determined relative to v_M, thereby differing from the actual values of v_i by a constant.

The end result of an infinite-stage decision process is to maximize the expected return per stage, denoted by g(d). As developed previously, d = k, k = 1, ..., K, implies the transition probabilities and return from state i are p_{ij} (d = k) and r_{ij}(d = k). The subscript n has been omitted from d for a steady state policy that maximizes g, independent of the stage. When the g's are equal, the object is to maximize the v_i's, the total expected return.

Consider the maximization of the gain. One way in which that g(d) can be maximized is to compute g for all decision policies. However, with M states and K feasible decisions from each state, there are K^M policies, a rather large number for relatively small values of M and K. A considerably more efficient way to find the policy which maximizes g(d) is to use an approximation in policy space to solve an infinite-stage recursion equation.

Let $\bar{f}$ be the optimal expected return per stage as the number of stages becomes infinitely large, i.e.

$$\bar{f} = \lim_{n \to \infty} \frac{\bar{f}_n(i)}{n}, \quad i = 1, \ldots, M.$$

Note $\bar{f}$ is independent of the starting state and

$$\bar{f} = \max_{d=1,\ldots,K} g(d),$$

where

$$g(d) = q_i(d) + \sum_{j=1}^{M} p_{ij}(d)\ v_j(d) - v_i(d)\ .$$

The approximation in policy space begins by guessing a policy d_o and calculating the corresponding expected return g_o and the relative values v_{io}. This procedure is called the "value determination operation" (Howard, 1960). Then to determine a better policy should one exist, the maximum of g is found using the current values of v_i, i.e. v_{io}, by solving

$$\max_{d=1,\ldots,K} [q_i(d) + \sum_{j=1}^{M} p_{ij}(d)(v_{jo} - v_{io})],\quad i = 1, \ldots., M. \qquad (3.22)$$

Equation (3.22) is called the "policy improvement routine" (Howard, 1960). A constant added to the v_i will not affect the maximization but will be only a test-quantity component independent of d = K.

The value of d which maximizes g is used as a next guess for an optimal policy (d_1). Then using d_1 the value determination operation is performed to determine corresponding values of g_1 and v_{i1}. In general, $g_1 \geq g_o$ and if $g_1 = g_o$, then $g_1 = \bar{f}$, and d_1 is an optimal policy. Otherwise, the policy improvement routine is again performed using v_{i1} to determine a new decision policy. The value determination operation together with the policy improvement routine is called the "policy iteration method" (Howard, 1960).

It has been proven that the policy iteration method yields monotone convergence to the optimal g, i.e. $g_M > g_{M-1}$, or when there is no improvement and $g_M = g_{M-1}$, the optimal average expected return and the optimal decision policy have been found.

An illustration of the policy iteration method concludes the discussion of infinite stage systems. There are two alternative decision and return matrices: These are

$$\underset{\sim}{P}_1 = \begin{pmatrix} 1/2 & 1/2 \\ 3/4 & 1/4 \end{pmatrix} \quad \text{and} \quad \underset{\sim}{R}_1 = \begin{pmatrix} 0 & 6 \\ -3 & 8 \end{pmatrix}$$

for d = 1

and

$$P_2 = \begin{pmatrix} 1 & 0 \\ 1/2 & 1/2 \end{pmatrix} \quad \text{and} \quad R_2 = \begin{pmatrix} 2 & 4 \\ 1 & -1 \end{pmatrix} \quad .$$

for d = 2.

The steady state solution for this problem is that the optimal decision if the system is in state two is d = 1, and if the system is in state one the optimal decision is d = 2, and also g = 2.

To verify the steady state solution with the policy iteration method, let the beginning solution be $d_o = 1$ for both states. The relevant probabilities and returns are then $\underset{\sim}{P}_1$ and $\underset{\sim}{R}_1$. The value of g_o would be found as follows:

$$\Pi(j) = \sum_{i=1}^{M} p_{ij}\ \Pi(i), \quad j = 1, \ldots, M$$

$$\sum_{j=1}^{M} \Pi(j) = 1$$

$$\Pi(1) = 1/2\ \Pi(1) + 3/4\ \Pi(2)$$

$$\Pi(2) = 1/2\ \Pi(1) + 1/4\ \Pi(2)$$

$$\Pi(1) + \Pi(2) = 1$$

Taking one of the first two equations and the third and solving simultaneously, the solution is found to be $\Pi(1) = 3/5$ and $\Pi(2) = 2/5$. Now

$$q_i = \sum_{j=1}^{M} r_{ij}\ p_{ij}$$

$$q_1 = 1/2\ (0) + 1/2\ (6) = 3$$

$$q_2 = 3/4\ (-3) + 1/4\ (8) = -\ 1/4$$

$g_o = \sum_{i=1}^{M} q_i\ \Pi(i) = 3\ (3/5) - 1/4\ (2/5) = 17/10$. Recalling from Equation (3.21) that M - 1 of the equations are independent, a v_i is arbitrarily fixed, say $v_{20} = 0$. Then v_{10} can be found as

$$v_1 = 3 + 1/2\ v_1 + 1/2\ v_2 - g$$

$$v_2 = -1/4 + 3/4\ v_1 + 1/4\ v_2 - g$$

or $\left.\begin{array}{l} g + 1/2\ v_1 = 3 \\ g - 3/4\ v_1 = -\ 1/4 \end{array}\right\}$ which has solution $g = 17/10$ and $v_1 = 13/5$.

This completes the first value determination operation, and now the policy improvement routine should be performed. For state one,

$$g(1) = 3 + 1/2\ (13/5) - 13/5 = 17/10$$

$g(2) = 2 + 1\ (13/5) - 13/5 = 2.$ Thus,

$d_1 = 2.$

For state two,

$g(1) = -1/4 + 3/4\ (13/5) = 17/10$

$g(2) = 0 + 1/2\ (13/5) = 13/10$

so $d_1 = 1.$

Turning again to the value determination operation, and using d_1, the relevant probabilities and returns are

$$\underset{\sim}{P} = \begin{pmatrix} 1 & 0 \\ 3/4 & 1/4 \end{pmatrix} \quad \underset{\sim}{R} = \begin{pmatrix} 2 & 4 \\ -3 & 8 \end{pmatrix} .$$

Setting $v_{21} = 0$, the equations from (3.21) become

$$v_{11} = 2 + v_{11} - g_1$$

and $$0 = -\ 1/4 + 3/4\ v_{11} - g_1.$$

So $g_1 = 2$ and $v_{11} = 3$. Using $v_{11} = 3$ and $v_{21} = 0$ in the policy improvement routine yields for state one,

$g(1) = 3 + 1/2\ (3) - 3 = 3/2$

and $g(2) = 2 + 1(3) - 3 = 2$

so $d_2 = 2.$

For stage two,

$g(1) = -\ 1/4 + 3/4\ (3) = 2$

and $g(2) = 0 + 1/2\ (3) = 3/2$

so $d_2 = 1.$

The last two iterations produced the same decisions, consequently the optimal decision from state one is $d = 2$ and from stage two, $d = 1$, with $g = 2$. Thus, the conjecture that the solution determined is steady state is verified.

3.3. Stochastic Linear Programming

Consider now the introduction of random variables into the linear programming model. Assume initially that the probability distributions of any random variables introduced into the problem are completely known. This would be called linear programming under "risk", (Nemhauser, 1966; Madansky, 1963). The following development is from Madansky (1963). Let the standard (LP) problem be given by

$$\min \quad \underset{\sim}{z} = \underset{\sim}{c}'\underset{\sim}{x} \tag{3.22}$$

$$\text{s.t.} \quad \underset{\sim}{A}\underset{\sim}{x} \geq \underset{\sim}{b} \tag{3.23}$$

$$\underset{\sim}{x} \geq 0 \tag{3.24}$$

where $\underset{\sim}{b}$ is mx1, $\underset{\sim}{x}$ and $\underset{\sim}{c}$ are n_1 - vectors, and $\underset{\sim}{A}$ is mxn_1 matrix.

The introduction of risk into the problem can occur in either the coefficients of the objective function, i.e. $\underset{\sim}{c}$; or in the constraints - either the right-hand side $\underset{\sim}{b}$, or the matrix $\underset{\sim}{A}$, or both.

First let the stochastic element be in the objective function (3.22). In this problem, the optimal vector $\underset{\sim}{x}$ lies in the convex polyhedron defined by the inequalities (3.23) and (3.24), and the problem is just one of trying to find a vector in this polyhedron which minimizes an appropriate objective function for the risky situation. The appropriate objective function is obtained as follows: Consider the utility of each possible value of the objective function, that is, the utility of $\underset{\sim}{c}'\underset{\sim}{x}$ for each possible $\underset{\sim}{c}$. Then take the expected value of the utility of $\underset{\sim}{c}'\underset{\sim}{x}$ over the distribution of the random vector $\underset{\sim}{c}$ as the objective function to be minimized (Madansky, 1963). If the utility function is linear in the objective function, then the problem reduces to one of looking at the inner product of the expected value of $\underset{\sim}{c}$ with $\underset{\sim}{x}$, and this now becomes a non-stochastic linear programming problem. But whatever the nature of the utility function, the problem has been converted to one which is nonstochastic.

A problem of quite different nature arises when risk is introduced into either the matrix $\underset{\sim}{A}$ or the right-hand side $\underset{\sim}{b}$. In this situation difficulty stems from how to carry over the notion of feasibility, inherent in linear programming, to a "linear program" whose matrix and right-hand side are random variables. Various formulations in this area have been directed at different ways of resolving the difficulty. The simplest answer is embodied in the "fat" formulation given by Madansky, 1963) and characterized by the following reasoning: The decision maker has to decide on some vector $\underset{\sim}{x}$ of activities before he can observe values of $\underset{\sim}{A}$ and $\underset{\sim}{b}$. After he has made his choice, he is confronted with some particular $\underset{\sim}{A}$ and $\underset{\sim}{b}$ and can see whether or not $\underset{\sim}{x}$ has satisfied the constraints. The difficulty is that his pre-chosen $\underset{\sim}{x}$ may not be feasible for the observed $\underset{\sim}{A}$ and $\underset{\sim}{b}$. What the "fat" formulation prescribes is that one restrict oneself to the convex set of those $\underset{\sim}{x}$ which are feasible, no matter what values of $\underset{\sim}{A}$ and $\underset{\sim}{b}$ will subsequently be observed. Thus, the intersection for all $\underset{\sim}{A}$ and $\underset{\sim}{b}$ of the polyhedra given by the constraints $\underset{\sim}{A}\underset{\sim}{x} \geq \underset{\sim}{b}$, $\underset{\sim}{x} \geq 0$ is the feasible region, and the problem is to find the $\underset{\sim}{x}$ in this set

$S = \bigcap_{\underset{\sim}{A},\underset{\sim}{b}} \{x \mid \underset{\sim}{x} \geq 0, \underset{\sim}{A}\underset{\sim}{x} \geq \underset{\sim}{b}\}$, the "permanently feasible" set, such that $z = \underset{\sim}{c}'\underset{\sim}{x}$ is minimum.

To characterize an optimal solution for the "fat" formulation does not present any difficulty. If $\underset{\sim}{x}$ belongs to S, and is optimal for any particular programming problem, Equations (3.22), (3.23), and (3.24) for some possible value of $\underset{\sim}{A}$ and of $\underset{\sim}{b}$, then $\underset{\sim}{x}$ is also optimal for the "fat" formulation. A difficulty with this formulation is that it may not lead to a decision because the permanently feasible set S may be empty. Madansky (1963) points out that problems where the probability distribution of either $\underset{\sim}{A}$ or $\underset{\sim}{b}$ is defined over the entire real line are likely to have S as the null set. Thus the "fat" formulation would not be of any help even in formulating the problem to be solved. A variant of the "fat" formulation which may not preserve feasibility requires that being feasible be only 100 P % sure. Then the set of nonnegative $\underset{\sim}{x}$'s are considered such that the probability that $\underset{\sim}{A}\underset{\sim}{x} \geq \underset{\sim}{b}$ is P and min $z = \underset{\sim}{c}'\underset{\sim}{x}$ for $\underset{\sim}{x}$ in this set.

A more realistic statement of the problem is the "slack" formulation also considered by Madansky (1963). It involves converting the problem to a two-stage problem, which can be roughly described as follows: The decision maker is supposed to choose a nonnegative $\underset{\sim}{x}$, then observe a value of the random matrix $\underset{\sim}{A}$ and the random vector $\underset{\sim}{b}$, and finally compare $\underset{\sim}{A}\underset{\sim}{x}$ with $\underset{\sim}{b}$. The vector $\underset{\sim}{x}$ may or may not be feasible. But feasible or not the decision maker is allowed to make another decision $\underset{\sim}{y}$ to compensate for any discrepancies between $\underset{\sim}{A}\underset{\sim}{x}$ and $\underset{\sim}{b}$, based on the original decision $\underset{\sim}{x}$ and the later observed $\underset{\sim}{A}$ and $\underset{\sim}{b}$, but at a penalty cost. An example of this formulation is the linear inventory problem (Madansky, 1963). Here $\underset{\sim}{x}$ is the amount of inventory which must be on hand, $\underset{\sim}{b}$ is the random variable of demand; to be observed later $\underset{\sim}{A}$ is a nonrandom matrix of relevant technology coefficients, and $\underset{\sim}{y}$ is the second-stage decision, embodying two kinds of activities. If the demand exceeds the inventory, the goods must be bought on the open market and at a penalty cost to meet the excess demand over supply. If the inventory exceeds the demand, the excess must be scrapped as a penalty, reflecting the loss for not having made a better choice of $\underset{\sim}{x}$. This formulation is more realistic than the "fat" formulation in that it keeps the decision maker in business after he has made a choice of $\underset{\sim}{x}$ and the random variables have been observed. The constraints for the two-stage problem are given by $\underset{\sim}{A}\underset{\sim}{x} + \underset{\sim}{B}\underset{\sim}{y} = \underset{\sim}{b}$. Note that "slack" variables have been introduced into the n_2 - vector $\underset{\sim}{y}$ so that the inequality constraints $\underset{\sim}{A}\underset{\sim}{x} \geq \underset{\sim}{b}$ are equalities. Typically, the mxn_2 matrix $\underset{\sim}{B}$ is going to be a matrix of zeros and plus or minus ones (Madansky, 1963). Also required is the nonnegativity of $\underset{\sim}{x}$ and $\underset{\sim}{y}$.

The objective function for this two stage problem is constructed as follows: Let $\underset{\sim}{f}$ be the nonrandom penalty cost vector for the second-stage decision vector $\underset{\sim}{y}$. For given $\underset{\sim}{A}$, $\underset{\sim}{b}$, and $\underset{\sim}{x}$ the best second-stage decision is found, that is, the $\underset{\sim}{y}$ which is optimal for

$$\left.\begin{aligned} \underset{\sim}{B}\underset{\sim}{y} &= \underset{\sim}{b} - \underset{\sim}{A}\underset{\sim}{x} \\ \underset{\sim}{y} &\geq 0 \\ \min z_1 &= f'y\ . \end{aligned}\right\} \qquad (3.25)$$

Now, assuming that the utility of the objective function is linear, the appropriate objective for the two-stage problem is

$$c'x + E \min_{\underset{\sim}{y}} \underset{\sim}{f}'\underset{\sim}{y} \ . \qquad (3.26)$$

It is also assumed that for every possible $\underset{\sim}{x}$ and $(\underset{\sim}{A}, \underset{\sim}{b})$ there exists a $\underset{\sim}{y}$ which will compensate for any discrepancy between $\underset{\sim}{A}\underset{\sim}{x}$ and $\underset{\sim}{b}$, given that the decision $\underset{\sim}{x}$ has been made and a particular $(\underset{\sim}{A}, \underset{\sim}{b})$ has been observed. This "slack" formulation reduces to the "fat" formulation in case any component of $\underset{\sim}{f}$ is infinite, that is, in case it costs so much for certain types of discrepancies and therefore the decision in the first stage is considered as permanently feasible.

To find a solution of the "slack" formulation, the problem where $\underset{\sim}{b}$ is random is easier to solve than the case where $\underset{\sim}{A}$ is random. (Madansky, 1963). One approach to a solution where $\underset{\sim}{b}$ is the random vector is to search for a "certainty equivalent", that is, a nonrandom vector which replaces the random vector $\underset{\sim}{b}$ so that the solution of the resulting nonrandom problem will also be the solution to the two-stage problem. As could be expected the expected value of $\underset{\sim}{b}$ is not always a certainty equivalent, but there are some situations where it would be. One circumstance where the expected value of $\underset{\sim}{b}$ would be a certainty equivalent is as follows:

Let $C(\underset{\sim}{b}, \underset{\sim}{x}) = \underset{\sim}{c}'\underset{\sim}{x} + \min_{\underset{\sim}{y}} \underset{\sim}{f}'\underset{\sim}{y}$

and if $C(\underset{\sim}{b},\underset{\sim}{x})$ has the form $C(\underset{\sim}{b}, \underset{\sim}{x}) = \underset{\sim}{A}_1(\underset{\sim}{x}) + \underset{\sim}{A}_2(\underset{\sim}{b}) + \underset{\sim}{A}_3(\underset{\sim}{x})\underset{\sim}{b}$ then replacing $\underset{\sim}{b}$ by its expected value and solving that nonstochastic problem will yield the solution to the two-stage problem (Madansky, 1960). An example of such a function $C(\underset{\sim}{b}, \underset{\sim}{x})$ is a quadratic in both $\underset{\sim}{x}$ and $\underset{\sim}{b}$. In general, when the components of the vector $\underset{\sim}{b}$ are each independent and have uniform distributions over some finite range, then the function $EC(\underset{\sim}{b}, \underset{\sim}{x})$ which is essential in the minimization is under fairly wide circumstances going to be of this quadratic nature (Beale, 1961) and the expected value solution will be the solution of this problem.

Another formulation of the stochastic (LP) problem is as a "chance-constrained" program. Each of the constraining equations of the original problem, (3.22), (3.23) and (3.24), are specified with regard to the probability with which each is to be achieved. Subject to these probabilistic constraints the minimization of the objective function is performed. There is similarity here with the "fat" formulation except that the probabilities of each possible infeasibility are explicitly stated.

The difference between the chance-constrained formulation and the "slack" formulation is that in the latter the specific plans of the decision maker for each possible infeasibility are explicitly spelled out, as are the explicit costs for all possible infeasibilities, where as in the former these explicit costs of the various types of infeasibility are reflected in the probabilities associated with each constraint. Consider the violation of a particular constraint to be costly. In the"slack" formulation one would have to think hard about what the actual costs of the specific plan under infeasibility would be. However, in the chance-constrained formulation a high probability of satisfying the constraint could be assigned if violation of the constraint would be very costly.

Aside from searching for certainty equivalents and reducing the stochastic problem to a nonrandom problem, algorithms for minimizing

E C($\underset{\sim}{b}$, $\underset{\sim}{x}$) have been developed (Dantzig, Madansky, 1961). Part of the two-stage problem is the second-stage problem, the problem that the decision maker has once he has made the initial decision $\underset{\sim}{x}$ and observed the random vector $\underset{\sim}{b}$. The form of the problem is Equation (3.25). For a given $\underset{\sim}{b}$ and $\underset{\sim}{x}$ there is an optimal set of dual variables, $\underset{\sim}{\bar{\Pi}}$ ($\underset{\sim}{b}$, $\underset{\sim}{x}$). One way of characterizing the solution of the two-stage problem is in terms of the expected optimal value of the dual variables of the second-stage problem.

Specifically, the following three results led to a particular algorithm (Dantzig, Madansky, 1961):

(i) Suppose $\underset{\sim}{\bar{x}}$ is the optimal first-stage decision, i.e., it minimizes E C($\underset{\sim}{b}$, $\underset{\sim}{x}$) and satisfied the constraints, and let x_1 be feasible. Then

$$[\underset{\sim}{c}' - E\ \underset{\sim}{\bar{\Pi}}'(\underset{\sim}{b},\ \underset{\sim}{x}_1)\underset{\sim}{A}]\underset{\sim}{\bar{x}} \leq [\underset{\sim}{c}' - E\ \underset{\sim}{\bar{\Pi}}'(\underset{\sim}{b},\ \underset{\sim}{x}_1)]\ \underset{\sim}{x}_1, \text{ i.e.,}$$

given any other feasible vector $\underset{\sim}{x}_1$, the optimum $\bar{x}$ for the two-stage problem provides a smaller value than does $\underset{\sim}{x}_1$ for the linear form $[\underset{\sim}{c}' - E\ \underset{\sim}{\Pi}'(\underset{\sim}{b},\ \underset{\sim}{x})\underset{\sim}{A}]\underset{\sim}{x}$. By generating linear forms based upon a particular choice of $\underset{\sim}{x}_1$ and evaluating the expected optimum for the second-stage problem given $\underset{\sim}{x}_1$ and then determining where there exists a vector which makes the above linear form smaller than when evaluated at x_1 is the way to proceed with the problem.

(ii) E C(b, x) is convex in $\underset{\sim}{x}$. Stated otherwise, the "slack" formulation of the problem is in reality a recasting of the problem as a convex programming problem. Although E C($\underset{\sim}{b}$, $\underset{\sim}{x}$) is convex it is not necessarily differentiable everywhere in the interior of the region of definition of $\underset{\sim}{x}$, so Lagrangian methods or calculus methods cannot be used to find solutions. However, the supports of this convex function can be constructed in terms of the expected optimal prices for the second-stage problem.

(iii) The plane given by $[\underset{\sim}{c}' - E\,\underset{\sim}{\bar{\Pi}}'(\underset{\sim}{b},\underset{\sim}{x}_1)\underset{\sim}{A}]\underset{\sim}{x} + \underset{\sim}{E}\,\underset{\sim}{\bar{\Pi}}'(\underset{\sim}{b},\underset{\sim}{x}_1)\underset{\sim}{b}$ is a support to $EC(\underset{\sim}{b},\underset{\sim}{x})$ at $\underset{\sim}{x} = \underset{\sim}{x}_1$. That is, the term $\underset{\sim}{c}' - E\,\underset{\sim}{\bar{\Pi}}'(\underset{\sim}{b},\ \underset{\sim}{x}_1)\underset{\sim}{A}$ behaves as a gradient of this convex function at $\underset{\sim}{x}_1$. This gives a combination of two results: one a result about the convexity of $EC(\underset{\sim}{b},\ \underset{\sim}{x}\ )$ and a characterization of the support planes, and the other a necessary condition for optimality of $\underset{\sim}{x}$. These results used in conjunction have been the foundation for algorithms for minimizing the convex function (Dantzig, Madansky, 1961).

Other efforts have been in determining optimizing algorithms for the case where $\underset{\sim}{b}$ takes on only a finite number of possible values. Here, the problem can be written out in full as the following large (LP) problem:

$$\begin{array}{lllll}
\underset{\sim}{A}\underset{\sim}{x} + \underset{\sim}{B}\underset{\sim}{y}_1 & & & & = \underset{\sim}{b}_1 \\
\underset{\sim}{A}\underset{\sim}{x} + & \underset{\sim}{B}\underset{\sim}{y}_2 & & & = \underset{\sim}{b}_2 \\
\hdashline
\underset{\sim}{A}\underset{\sim}{x} + & & & \underset{\sim}{B}\underset{\sim}{y}_N & = \underset{\sim}{b}_N \\
\underset{\sim}{c}'\underset{\sim}{x} + p_1\underset{\sim}{f}'\underset{\sim}{y}_1 + p_2\underset{\sim}{f}'\underset{\sim}{y}_2 + \cdots + & & p_N\underset{\sim}{f}'\underset{\sim}{Y}_N & & = \min
\end{array} \qquad (3.27)$$

where the p's are the probabilities of the various b's. In this format the solution is not just for the optimal $\underset{\sim}{x}$, but for the whole set of optimal plans $y_1, \ldots, y_N$, where as in the formulation as a convex programming problem of interest is only the optimum first-stage decision. A method of computer solution by a dual problem formulation has been developed (Dantzig, Wolfe, 1961).

3.4 Reliability in Linear Programming. With the general approach to stochastic (LP) problem formulations given above and the development of the conditions whereby algorithms could be used to solve the problem a more detailed development of the stochastic aspects of the programming problem will now be given. Certainly the most available principle to lead to an optimal policy is to optimize the expected value of the objective function. In the development of the preceeding formulations by Madansky (3.22 - 3.24) the expectation was with regard to the objective function (3.22) and with "chance constrained" problems, (3.23 - 3.24). However, it seems that few theories discuss "reliability" of the obtained optimal policy as is usually seen in mathematical statistics (Kataoka, 1962). Discussions of reliability are encountered in another approach to stochastic (LP) problems which does not consider directly problems of decision making under risk. Rather, of interest are questions of the form: What is the distribution — or at least what are the expected value and variance — of the objective function if the value of the random $\tilde{A}$ and $\tilde{b}$ are realized, and then solve the nonrandom (deterministic) problem. (Tintner, 1955; Madansky, 1963; Kataoka, 1962).

In order to define a criterion of reliability of a policy, the distribution function of the value of the objective function must be known, which usually means "cost". Once the distribution function is determined, a natural definition of the reliability criterion is as follows: For a given α $(0 < \alpha < 1)$ obtain a limit η such that

$$\Pr(\text{cost of the policy} \leq \eta) = \alpha$$

The term η is called "α percent confidence limit of cost" of the policy. This means that when that policy is used, and letting $\alpha = .95$, the cost will be under η with a confidence of 95%.

Let two decision policies be defined as follows:

(i) minimum expectation policy: $\hat{x}$

(ii) deterministic minimum policy: $\bar{x}$.

The minimum expectation policy $\hat{x}$ is determined by solving

$\min\limits_{\underset{\sim}{x}} E_{\underset{\sim}{s}}\, f(\underset{\sim}{x}, \underset{\sim}{s})$ where $\underset{\sim}{s}$ is a vector of random variables and $f(\underset{\sim}{x}, \underset{\sim}{s})$ is a cost function. The deterministic minimum policy $\bar{x}$ is obtained by solving $\min\limits_{\underset{\sim}{x}} f(\underset{\sim}{x}, \underset{\sim}{\bar{s}})$, where $\underset{\sim}{\bar{s}}$ is the vector of the expected values of the random variables. Then to be considered are relationships between $\hat{x}$ and $\bar{x}$, and the derivation of inequalities which are useful for estimating $\hat{x}$.

As a first step, assume that

(i) data of demand in m time periods are given, and that linear regression analysis can be applied to the data, and

(ii) the sample size m is large enough for a Student's t distribution to be well approximated by a normal distribution. Then, the distribution of demand is taken as a normal distribution. These assumptions will be of use in a production horizon problem to be developed now.

Certain definitions and assumptions are necessary for presentation of the problem.

(A) Definitions and Assumptions:

s: vector of random variables which can be production coefficients of demand

$\bar{s}$: $E\ s$

S: set of all possible values of s

x: vector of controllable variables (deterministic amounts of production)

X: set of feasible x's

$f(x,s)$: objective function which can be cost, profit, etc.

Assumption (1): Let $f(x,s)$ be a cost function and convex in s. The reliability criterion is

$Pr(f(x,s) \leq \eta) = \alpha$, where

η is the α confidence limit of cost.

Assumption (2): The set of feasible x's is not dependent on S. In other words, the conditions which x's should fulfill do not contain the random variable s.

In addition, several different kinds of policies are defined:

$\min\limits_{x \varepsilon X} \ \underset{s \varepsilon S}{E} \ f(x,s) = E\ f(\hat{x},s)$: minimum value of expectation

where

$\hat{x}$ is the minimum expectation policy (M.E.P.)

$\underset{s \varepsilon S}{E} \ \min\limits_{x \varepsilon X} f(x,s)$: expectation of minimum value

$$\min_{x\varepsilon X} f(x,\bar{s}) = f(\bar{x},\bar{s}):\quad \text{deterministic minimum value}$$

where

$\bar{x}$ is the deterministic minimum policy (D.M.P.).

Madansky (1960) shows that

$$\min_{x} f(x,s) \leq f(x,s).$$

Therefore $$\mathop{E}_{s} \min_{x} f(x,s) \leq \mathop{E}_{s} f(x,s)$$

and $$\mathop{E}_{s} \min_{x} f(x,s) \leq \min_{x} E f(x,s) \tag{3.28}$$

Since f(x,s) is a convex function in s, we can apply Jensen's inequality (Feller, 1966) to obtain

$$\min_{x} f(x,\bar{s}) \leq \mathop{E}_{s} \min_{x} f(x,s), \tag{3.29}$$

Also, for x as a (M.E.P.),

$$\min_{x} \mathop{E}_{s} f(x,s) = \mathop{E}_{s} f(\hat{x},s) \leq \mathop{E}_{s} f(\bar{x},s). \tag{3.30}$$

Then, from Equations (3.28) (3.29) and (3.30) the following inequalities hold:

$$f(\bar{x},\bar{s}) \leq \min_{x} \mathop{E}_{s} f(x,s) \leq E f(\bar{x},s). \tag{3.31}$$

Thus, the following statements can be made:

(i) Equation (3.28) shows that the minimum value of expectation is not smaller than the expectation of minimum value.

(ii) If f(x,s) is a convex function of s the minimum value of expectation exists between a lower bound of the deterministic minimum value and the upper bound of the expectation of the cost of deterministic minimization policy. These inequalities give a simple method for evaluating $\bar{x}$ by the following relationship;

$$E\ f(\hat{x},s) - f(\bar{x},\bar{s}) \leq E\ f(\bar{x},s) - f(\bar{x},\bar{s})$$

and $$\frac{E\ f(\hat{x},s) - f(\bar{x},\bar{s})}{f(\bar{x},\bar{s})} \leq \frac{E\ f(\bar{x},s) - f(\bar{x},\bar{s})}{f(\bar{x},\bar{s})}, \tag{3.31}$$

then

$$\frac{E\ f(\hat{x},s) - f(\bar{x},\bar{s})}{E\ f(\hat{x},s)} \leq \frac{E\ f(\bar{x},s) - f(\bar{x},\bar{s})}{f(\bar{x},\bar{s})} \tag{3.32}$$

The left-hand side of Equation (3.32) is the relative error of $f(\bar{x},\bar{s})$, which is limited by the right hand side of (3.32). If the right hand side is small when compared to unity, the D.M.P., $\bar{x}$, would be a good approximation for $\hat{x}$.

3.5. Example of Reliability with Demand as Random Variable.

As an application of the theoretical development of reliability thus far, consider a production horizon problem in the case where demand in each period is a random variable. The model is one of simple commodity production over n periods with no time lag between production and selling. The most important problem in the production horizon problem is how to predict amounts of demand in the future. Depending on the property and quantity of the concerned commodity, there would be a variety of methods for forcasting. Adopted here is the most popular one: linear regression as presented by Kataoka (1962).

Let the amounts of demand, $s^{(k)}$, be given at time t_k, $(k=1,\ldots,m)$. Further, the amount of demand at time t, s_t, is a random variable which is normally distributed with mean value $\beta_1 + \beta_2 t$ and variance σ^2, or in other words

$$s_t = \beta_1 + \beta_2 t + \varepsilon\sigma, \tag{3.33}$$

where ε is a normal random variable with $E\,\varepsilon = 0$, and $E\,\varepsilon^2 = 1$, with β_1, β_2, and σ^2as unknownparameters. Let further definitions be:

$$\bar{s} = \frac{1}{m}\sum_{k=1}^{m} s^{(k)}, \tag{3.34}$$

as the sample mean of demand,

$$\bar{t} = \frac{1}{m}\sum_{k=1}^{m} t_k, \tag{3.35}$$

as the mean time,

$\hat{\beta}_1$ as the least squares estimate of β_1,

$\hat{\beta}_2$ as the least squares estimate of β_2,

$\hat{\sigma}_2$ as the least squares estimate of σ^2.

Then the prediction of demand by linear regression from the method of least squares is as follows:

$$\hat{\beta}_2 = \frac{\sum(s^{(k)} - \bar{s})(t_k - \bar{t})}{\sum(t_k - \bar{t})^2} \tag{3.36}$$

$$\hat{\beta}_1 = \bar{s} - \hat{\beta}_2\,\bar{t} \tag{3.37}$$

$$\hat{\sigma}^2 = \frac{1}{m}\sum_{k=1}^{m}(s^{(k)} - \hat{\beta}_1 - \hat{\beta}_2\, t_k)^2. \tag{3.38}$$

The least squares estimate of s_τ at τ is

$$\hat{s}_\tau = \hat{\beta}_1 + \hat{\beta}_2\, \tau .$$

Further, define

$$z_\tau = s_\tau - \hat{s}_\tau$$

then the variance of z_τ is given by

$$\text{var } z_\tau = \sigma^2\left(\frac{m+1}{m} + \frac{(\tau - \bar{t})^2}{\sum(t_k - \bar{t})^2}\right) ,$$

which can be found in Fryer (1966).

If the random variable T is defined as

$$T = \frac{z_\tau/\sqrt{\text{var } z_\tau}}{m\hat{\sigma}^2/(m-2)\sigma^2} = \frac{z_\tau/\left(\frac{m+1}{m} + \frac{(\tau - \bar{t})^2}{\sum(t_k - \bar{t})^2}\right)^{1/2}}{\left[\sum(s^{(k)} - \hat{\beta}_1 - \hat{\beta}_2\, t_k)^2\right]^{1/2}}\sqrt{m-2} , \tag{3.39}$$

then the distribution of the random variable T is the Student's t-distribution with (m-2) degrees of freedom.

Since it is known that for a sufficiently large m, the distribution function of a random variable

$$u = T\left(\frac{m-4}{m-2}\right)^{1/2}$$

is well approximated by the normal distribution with $E(u) = 0$, $E(u^2) = 1$, the density function of s will be approximated by

$$f(s) = \frac{1}{\sqrt{2\Pi}\ \sigma_\tau} \exp\left\{- \frac{(s_\tau - m_\tau)^2}{2\ \sigma_\tau^2}\right\} \tag{3.40}$$

where

$$m_\tau = \hat{\beta}_1 + \hat{\beta}_2 \tau \tag{3.41}$$

and

$$\sigma_\tau^2 = \left\{\sum_k (s^{(k)} - \hat{\beta}_1 - \hat{\beta}_2\ t_k)^2\right\}\left\{\frac{m+1}{m} + \frac{(\tau - \bar{t})^2}{\sum (t_k - \bar{t})^2}\right\} \quad / \ (m-4)$$

from Kataoka (1962).

Definition (3.41) gives an interesting fact that the variance of the expected amount of demand is a monotone increasing function of the prediction time interval if $\hat{\beta}_2 > 0$. The normal distribution function (3.40) will now be used as the demand function at a time period. Putting $\tau = t_m + i\tau_o$, then m_i and σ_{si}^2 are defined as

$$m_i = \hat{\beta}_1 + \hat{\beta}_2(t_m + i\tau_o)$$

and

$$\sigma_{si}^2 = \left\{\sum_{k=1}^{m} (s^{(k)} - \hat{\beta}_1 - \hat{\beta}_2\ t_k)^2\right\}\left\{\frac{m+1}{m} + \frac{(t_m + i\tau_o - \bar{t})^2}{\sum (t_k - \bar{t})^2}\right\} ,$$

where τ_o is the time interval of one period, and $i = 1, 2, 3, \ldots$.

3.5.1 Minimum Expectation Policy

Let the following definitions be made:

n: total number of periods

$$\phi(s;m;\sigma^2) = \frac{1}{\sqrt{2\Pi}\,\sigma} e^{-\frac{(s-m)^2}{2\sigma^2}}$$

s_i: a random demand for the commodity during the ith period

m_i: $E\, s_i$

$\sigma^2_{s_i}$: $E(s_i - m_i)^2$

$\phi(s_i;m_i,\sigma^2_{s_i})$: density function of the amount of demand during ith period and denoted also as $\phi(s_i)$

S_i: $\sum_{j=1}^{i} s_j = S_i$

M_i: $E\, S_i$.

An assumption that the amount of demand, s_1, s_2, ..., s_n are independent is also made. Then

$$\sigma_i^2 = E(S_i - M_i)^2 = E(\sum_{j=1}^{i} (s_j-m_j))^2 = \sum_{j=1}^{i} \sigma^2_{s_j}$$

To develop the cost function further definitions need be made:

x_i: amount of the commodity produced during the ith period and is a controllable variable.

c_i: capacity of production where

$$0 \leq x_i \leq c_i \tag{3.42}$$

y_o: initial inventory which is constant

$C_i = \sum_{j=1}^{i} c_i + y_o$

y_i: inventory at the end of ith period

X_i: $\sum_{j=1}^{i} x_j + y_o$.

Then the cost function consists of two parts, the production cost and the inventory-penalty cost. Let the notation for the cost function be:

$p_i(x_i)$: production cost of x_i

linear production cost: $p_i(x_i) = a_i\ x_i$

convex production cost: $p_i(x_i) = a_i\ x_i + b_i\ x_i^2,\ a_i \geq 0,\ b_i > 0$

concave production cost: $a_i\ \ell og\ (1 + x_i)$.

The inventory-penalty cost contains two parts:

an inventory cost of ith period for $y_i \geq 0$

a penalty cost of th period for $y_i < 0$.

Now, suppose this portion of the complete cost function is given by

$$G(y) = \begin{cases} ay \text{ for } y \geq 0 \\ -by \text{ for } y < 0 \end{cases} ,$$

where a, b > 0, then the cost of the ith period f_i is

$$f_i = p_i(x_i) + G(y_i) \tag{3.43}$$

and the following relationships hold:

$$\left.\begin{array}{l} y_1 = X_1 - S_1 \\ y_2 = X_2 - S_2 \\ - - - - - - - \\ y_n = X_n - S_n \end{array}\right\} \qquad (3.44)$$

and

$$\left.\begin{array}{l} x_1 = X_1 - y_o \geq 0 \\ x_2 = X_2 - X_1 \geq 0 \\ - - - - - - - - - \\ x_n = X_n - X_{n-1} \geq 0 \end{array}\right\} \qquad (3.45)$$

Inserting (3.44) and (3.45) into (3.43) gives

$$f_i = f_i(X_i, X_{i-1}, S_i) = p_i(X_i - X_{i-1}) + G(X_i - S_i) \qquad (3.46)$$

and the total cost becomes

$$f(x,s) = \sum_{i=1}^{n} f_i(X_{i-1}, S_i), \; (X_o = y_o). \qquad (3.47)$$

Kataoko (1962) shows that the expectation of the inventory-penalty cost, G_i is

$$Q_i(X_i) = \int_{-\infty}^{\infty} G_i(X_i - S_i) \; \phi(S_i;M_i,\sigma_i^2) \; D \; S_i, \qquad (3.48)$$

and the expectation of the total cost is

$$E \; f(x,s) = \sum_{i=1}^{n} \{p_i(X_i - X_{i-1}) + Q_i \; (X_i)\}. \qquad (3.49)$$

Let further be defined the quantities:

$$\left.\begin{array}{l} f_1(y_o,X_1) = p_1(X_1 - y_o) + Q_1(X_1) \\ f_2(X_1,X_2) = p_2(X_2 - X_1) + Q_2(X_2) \\ - - - - - - - - - - - - - - - - - \\ f_n(X_{n-1},X_n) = p_n(X_n - X_{n-1}) + Q_n(X_n) \end{array}\right\} \quad . \qquad (3.50)$$

Then, the problem of interest will be

$$\min \{f_1(y_o,X_1) + f_2(X_1,X_2) + \cdots + f_n(X_{n-1},X_n)\}$$

s.t.: Equations (3.42) and (3.45)

This is a dynamic programming problem which can be solved by successive minimization. Computational procedures are given by Kataoka (1962). For a deterministic solution $G_i(X_i - M_i)$ is substituted for $Q_i(X_i)$ in (3.50).

3.5.2. Computational Procedures for the Minimum Expectation Policy.

The procedure to follow is an outline of a more detailed development by Kataoka (1962).

To begin the following definitions are made:

$$F_{n-1}(X_{n-1}) = \min f_n(X_{n-1}, X_n)$$

$$X_{n-1} \le X_n \le C_n$$

$$0 \le X_{n-1} \le C_{n-1}$$

$$F_{n-2}(X_{n-2}) = \min \{f_{n-1}(X_{n-2}, X_{n-1}) + F_{n-1}(X_{n-1})\}$$

$$X_{n-2} \le X_{n-1} \le C_{n-1}$$

$$0 \le X_{n-2} \le C_{n-2}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots \qquad (3.51)$$

$$F_1(X_1) = \min \{f_2(X_1, X_2) + F_2(X_2)\}$$

$$X_1 \le X_2 \le C_2$$

$$0 \le X_1 \le C_1$$

$$\min E\, f(x,s) = \min \{f_1(y_o, X_1) + F_1(X_1)\}$$

$$0 \le X_1 \le C_1$$

Simultaneously, the optimal policy at each step as a function of the preceeding variable can be found:

$$\hat{X}_n = X_n(X_{n-1})$$

$$\hat{X}_{n-1} = X_{n-1}(X_{n-2})$$

$$\cdots\cdots\cdots \qquad (3.52)$$

$$\hat{X}_2 = X_2(X_1)$$

$$\hat{X}_1 = \text{constant}$$

and from the last step of (3.51), the optimal $\hat{X}_1$ can be obtained as

$$y_o + \hat{x}_1 = \hat{X}_1 = \text{constant.}$$

Then, by solving (3.52) by backward recursion for $X_1, X_2, \ldots, X_n$, the optimal policy $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n$ will be obtained and is given as

$$\begin{aligned} \hat{x}_1 &= \hat{X}_1 - y_o \\ \hat{x}_2 &= \hat{X}_2 - \hat{X}_1 \\ &\cdots\cdots\cdots \\ \hat{x}_n &= \hat{X}_n - \hat{X}_{n-1}. \end{aligned} \tag{3.53}$$

Then the approximate distribution function of f(x,s) and the reliability criterion can be found. The results as given by Kataoka (1962) are as follows: Let

$$x = \frac{f(x,s) - Ef(x,s)}{\sigma_f} = \frac{\sum(G_i - Q_i)}{\sigma_f},$$

where Ef(x,s) is given by (3.49) and

$$\sigma_f^2 = E[f(x,s) - Ef(x,s)]^2 .$$

Then the distribution function of x

$$P(X \leq x) = \psi(x) = \psi_o(x) + \frac{\xi_f^3}{3\sqrt{\Pi}} e^{-\frac{x^2}{2}(1-x^2)} \tag{3.54}$$

where

$$\psi_o(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\Pi}} e^{-\frac{x^2}{2}} dx \tag{3.55}$$

and

$$\xi_f^3 = E[f(x,s) - Ef(x,s)]^3$$

The reliability criterion, α limit η, will be obtained as follows:

For each given value of α, let $x(\alpha)$ be such that $\psi(x(\alpha)) = \alpha$. Then $P(X \leq x(\alpha)) = \alpha$ implies

$$P\left(\left(\frac{f(x,s) - Ef(x,s)}{\sigma_f}\right) \leq x(\alpha)\right) = \alpha$$

$$P(f(x,s) \leq \eta) = \alpha$$

where

$$\eta = \sum_{i=1}^{n} p_i(x_i) + Q^{(1)} + \sigma_f \, x(\alpha),$$

in which

$$Q^{(1)} = \sum_i E \, G_i = Q_i^{(1)}$$

and

$$G_i = G(X_i - S_i).$$

The derivation of the reliability criterion depends upon fitting a distribution function to f(x,s) by the method of moments.

Therefore, a reliability criterion for the optimal policy has been developed, incorporating considerable statistical theory in the method. The following example illustrates the complexity of computation encountered even for simple linear production costs and normally distributed demands.

3.5.3 Application of Kataoka Method of Reliability

Assuming that production has been continuing over a sufficient interval of time such that the least squares estimates for the distribution parameters of the random variable of demand have been found then the problem needs certain definitions before minimization of the expected value of the cost function can be found.

Definitions:

n: total number of periods of prediction of random demand: 3 Days.

$\phi(s,m,\sigma^2) = N(s;50,25)$

s_i: random demand during ith day

m_i: $E(s_i) = 50$

$\sigma^2_{s_i}$: $E(s_i - m_i)^2 = 25$

$\phi(s_i;m_i,\sigma^2_{s_i}) = \phi(s,m,\sigma^2)$: density function of the amount of demand during ith day is N(50,25), i.e. $\phi(s_i;m_i,\sigma^2_{s_i}) = \phi(s,m,\sigma^2) = \frac{1}{\sqrt{50\Pi}} e^{-\frac{1}{50}(s-50)^2}$

S_i: $\sum_{j=1}^{i} s_j$: demand over first i days

M_i: ES_i: expected demand over first i days

$$E(S_i) = E(\sum_{j=1}^{i} s_j) = \sum_{j=1}^{i} E(s_j) = i(50)$$

Assumption: The amounts of demand s_1,s_2,s_3 are independent.

x_i: Amount of production during ith day

c_i: capacity of production during ith day: 60, i.e. constant capacity over days.

$0 \leq x_i \leq 60$ over each day

y_o: initial inventory of 15 items

$C_i = \sum_{j=1}^{i} c_i + y_o = i(60) + 15$ is capacity for first i days

y_i: inventory at the end of the ith day.

$X_i = \sum_{j=1}^{i} x_j + y_o$: total production over first i days and the initial inventory.

Production Cost: Assume a linear production cost on each day

$$p_i(x_i) = a + bx_i = 75 + 1x_i$$

Inventory penalty cost:

$$G(y) = \left\{ \begin{array}{ll} 2y & \text{for } y \geq 0 \\ -3y & \text{for } y < 0 \; . \end{array} \right\} \tag{3.56}$$

Then the cost of the ith day is

$$f_i = 75 + 1x_i + G(y_i).$$

The relations (3.44) and (3.45) are then made. Insertion of these relations into (3.43) gives

$$f_1 = 75 + 1(X_1 - y_o) + G(X_1 - S_1)$$

$$f_2 = 75 + 1(X_2 - X_1) + G(X_2 - S_2)$$

$$f_3 = 75 + 1(X_3 - X_2) + G(X_3 - S_3).$$

Now Equation (3.48) needs to be found. For this production problem example (3.48) is given as

$$\phi(S_3;M_3,\sigma_3^2) = N(S_3;150,75)$$

$$\phi(S_2;M_2,\sigma_2^2) = N(S_2;100,50)$$

$$\phi(S_1;M_1,\sigma_1^2) = N(S_1;50,25).$$

Then

$$\phi_i(X_i) = 2 \int_{-\infty}^{X_i} (X_i - S_i)\ \phi(S_i;i\cdot 50,i\cdot 25)\ dS_i - 3 \int_{X_i}^{\infty} \phi(S_i;i\cdot 50,i\cdot 25) dS_i.$$

The expectation of total cost is given by (3.49) as

$$Ef(x,s) = 3.75 - y_o + X_3 + \sum_{i=1}^{3} Q_i(X_i).$$

Equation (3.50) is defined as

$$f_1(y_o,X_1) = 75 + (X_1 - y_o) + Q_1(X_1)$$

$$f_2(X_1,X_2) = 75 + (X_2 - X_1) + Q_2(X_2)$$

$$f_3(X_3,X_2) = 75 + (X_3 - X_2) + Q_3(X_3) \; .$$

Therefore, the problem of interest is to minimize

$$\{f_1(y_o, X_1) + f_2(X_1, X_2) + f_3(X_2, X_1)\}$$

subject to the constraints (3.42) and (3.45). Equations (3.51) are defined and (3.52) becomes the method of finding the optimal policy of each step as a function of the preceeding variable. The solutions of (3.52) are found by differentiation with respect to a limit of an integral as found in Buck(1956). To get $\hat{X}_3 = X_3(X_2)$

$$F_2(X_2) = \min f_3(X_2, X_3) = \min \{75 + (X_3 - X_2) + Q_3(X_3)\}$$

$$X_2 \le X_3 \le C_3 = 60 \cdot 3 + 15 = 195$$

$$0 \le X_2 \le C_2 = 60 \cdot 2 + 15 = 135$$

$$\frac{\partial}{\partial X_3} f_3(X_2, X_3) = \frac{\partial}{\partial X_3} [75 + (X_3 - X_2) + Q_3(X_3)] = 1 + \frac{\partial}{\partial X_3} Q_3(X_3)$$

Set to zero

$$1 + \frac{2}{\sqrt{150\pi}} \left\{\frac{\partial}{\partial X_3} [X_3 \int_{-\infty}^{X_3} e^{-\frac{1}{150}(S_3-150)^2} dS_3 - \int_{-\infty}^{X_3} S_3 e^{-\frac{1}{150}(S_3-150)^2} dS_3]\right\}$$

$$- \frac{3}{\sqrt{150}} \left\{\frac{\partial}{\partial X_3} [X_3 \int_{X_3}^{\infty} e^{-\frac{1}{150}(S_3-150)^2} dS_3 - \int_{X_3}^{\infty} S_3 e^{-\frac{1}{150}(S_3-150)^2} dS_3]\right\} .$$

Letting $y = \dfrac{S_3 - 150}{\sqrt{75}}$, then

$$0 = 1 + \frac{2}{\sqrt{2\pi}} \int_{-\infty}^{\frac{X_3-150}{\sqrt{75}}} e^{-y^2/2} d_y - \frac{3}{\sqrt{2\pi}} \int_{\frac{X_3-150}{\sqrt{75}}}^{\infty} e^{-y^2/2} dy$$

$$0 = 1 + 2\Phi\ (\frac{X_3-150}{\sqrt{75}}) - 3[1 - \Phi(\frac{X_3-150}{\sqrt{75}})]$$

$$\Phi(\frac{X_3-150}{\sqrt{75}}) = .4$$

$$\hat{X}_3 = 150 - 8.66(.254) = 147.8$$

$$\frac{\partial^2 f_3(X_2,X_3)}{\partial X_3^2} = \frac{\partial}{\partial X_3}\ [-\ 2 + 5\ \Phi(\frac{X_3-150}{\sqrt{75}})] \geq 0\ .$$

Therefore, $\hat{X}_3$ is minimum of $F_2(X_2)$. Since

$$X_2 \leq X_3 \leq 195$$

$$0 \leq X_2 \leq 135$$

$\hat{X}_3$ is feasible.

For $F_1(X_1)$ the value of $\hat{X}_2$ is found as

$$F_1(X_1) = \min\ \{75 + (X_2-X_1) + Q_2(X_2) + 75 + (147.8 - X_2) + Q_3(147.8)\}$$

$$\frac{\partial}{\partial X_2}\ (f) = 1 + \frac{\partial}{\partial X_2}\ (X_2) - 1 = \frac{\partial}{\partial X_2}\ Q_2(X_2).$$

Setting to zero gives

$$0 = 2\Phi(\frac{X_2-100}{\sqrt{50}}) - 3[1 - \Phi(\frac{X_2-100}{\sqrt{50}})]$$ from previous argument for $\hat{X}_3$.

Therefore,

$$\Phi(\frac{X_2-100}{\sqrt{50}}) = .6$$

$$\frac{X_2-100}{\sqrt{50}} = .254$$

$\hat{X}_2 = 100 + 7.07\ (.254) = 101.8$ and checking the constraints for X_2 shows $\hat{X}_2$ is feasible.

So

$$F_1(\hat{X}_1) = 75 + (101.8 - X_1) + Q_2(101.8) + 75 + (147.8 - 101.8) + Q_3(147.8).$$

Then the

$$Ef(x,s) = 75 + (X_1-y_o) + Q_1(X_1) + 75 - (101.8 - X_1) +$$

$$Q_2(101.8) + 75 + (147.8 - 101.8) + Q_3(147.8)$$

and the

$$\frac{\partial f}{\partial X_1} = \frac{\partial Q_1(X_1)}{\partial X_1} = -3 + 5\Phi(\frac{X_1-50}{\sqrt{25}}) \ .$$

Setting to zero gives

$$\Phi(\frac{X_1-50}{\sqrt{25}}) = .6$$

$$\frac{X_2-50}{5} = .254$$

$$\hat{X}_1 = 50 + 1.27 = 51.27$$

and checking the constraints for X_1 shows $\hat{X}_1$ feasible.

Therefore, the optimal policy for minimizing the expectation of the cost function is

$$y_o = 15$$

$$\hat{x}_1 = \hat{X}_1 - \hat{y}_o = 51.27 - 15 \simeq 36$$

$$\hat{x}_2 = \hat{X}_2 - \hat{X}_1 = 101.8 - 51 \simeq 102 - 51 = 51$$

$$\hat{x}_3 = \hat{X}_3 - \hat{X}_2 = 147.8 - 101.8 \simeq 148 - 102 = 46.$$

Now to obtain an upper bound on the cost function its distribution must be found. Kataoka (1962) uses the method of moments to fit a distribution to the cost function f(x,s). The first three central moments need be obtained:

$$f(x,s) = \sum_{i=1}^{3} p_i(X_i - X_{i-1}) + \sum_{i=1}^{3} G(X_i - S_i)$$

$$Ef(x,s) = \sum_{i=1}^{3} P_i(X_i - X_{i-1}) + \sum_{i=1}^{3} Q_i(X_i)$$

$$\text{var }(f(x,s)) = E(f(x,s) - Ef(x,s))^2 = E[\sum G_i(X_i - S_i) - \sum_{i=1}^{3} Q_i(X_i)]^2$$

$$= E[\sum G_i - \sum Q_i]^2 = E(\sum G_i)^2 - (\sum Q_i)^2 = T^{(2)} - Q^{(1)^2}, \text{ where}$$

for the production example

$$T^{(2)} = E(G_i)^2 = E[\sum G_i^2 + 2 \sum_{i<j}\sum G_i G_j] = \sum EG_i^2 + 2 \sum_{i<j}\sum E(G_i G_j)$$

$$= \sum Q_i^2 + 2 \sum_{i<j}\sum Q_i Q_j$$

From (3.54) $T^{(2)}$ can be written as

$$T^{(2)} = \sum_{i=1}^{3} \{ 4 \int_{-\infty}^{X_i} (X_i - S_i)^2 \, \phi(S_i; i \cdot 50, i \cdot 25)\, dS_i + 9 \int_{X_i}^{\infty} (X_i - S_i)^2 \, \phi(S_i; i \cdot 50, i \cdot 25)\, dS_i$$

$$+ 2 \{ \sum_{i<j}\sum [(2 \int_{-\infty}^{X_i} (X_i - S_i) \phi(S_i; i \cdot 50, i.25)\, dS_i - 3 \int_{X_i}^{\infty} (X_i - S_i) \, \phi(S_i; i \cdot 50, i.25)\, dS_i)$$

$$(2 \int_{-\infty}^{X_j} (X_j - S_i) \, \phi(S_j; j \cdot 50, j.25)\, dS_j - 3 \int_{X_j}^{\infty} (X_j - S_j) \, \phi(S_j; j \cdot 50, j.25)\, dS_j)] \}$$

and

$$Q^{(1)^2} = (\sum_{i=1}^{3} Q_i)^2 = \left\{ \sum_{i=1}^{3} [2 \int_{-\infty}^{X_i} (X_i - S_i) \ \phi(S_i; i.50, i.25) dS_i \right.$$

$$\left. - 3 \int_{X_i}^{\infty} (X_i - S_i) \ \phi(S_i; i\cdot 50, i\cdot 25) dS_i] \right\}^2 .$$

The third central moment of f(x,s) needs to be found as

$$\xi^3 = E[f(x,s) - Ef(x,s)]^3 = E(\sum G_i - \sum Q_i]^3$$

$$= E[(\sum G_i)^3 = 3(\sum Q_i)(\sum G_i)^2 + 3(\sum Q_i)^2(\sum G_i) - (\sum Q_i)^3]$$

$$= T^{(3)} - 3Q^{(1)}T^{(2)} + 3Q^{(1)^3} - Q^{(1)^3}$$

$$= T^{(3)} - 3Q^{(1)}T^{(2)} + 2Q^{(1)^3} .$$

where

$$T^{(3)} = E(\sum G_i)^3 = E(\sum G_i^3 + 3\sum_{i<j}\sum G_i G_j^2 + 6 \sum_{i<j<k}\sum\sum G_i G_j G_k)$$

$$= \sum EG_i^3 + 3\sum_{i<j}\sum EG_i G_j^2 + 6 \sum_{i<j<k}\sum\sum EG_i G_j G_k$$

$$= \sum_{i=1}^{3} [8 \int_{-\infty}^{X_i} (X_i - S_i)^3 \phi(S_i; i.50, i.25) dS_i - 27 \int_{X_i}^{\infty} (X_i - S_i)^3 \phi(S_i; i.50, i.25) dS_i]$$

$$+ 3\left\{\sum_{i<j}\sum [(2 \int_{-\infty}^{X_i} (X_i - S_i) \ \phi(S_i; i\cdot 50, i\cdot 25) dS_i - 3 \int_{X_i}^{\infty} (X_i - S_i) \ \phi(S_i; i\cdot 50, i\cdot 25) dS_i)\right.$$

$$\left.(4 \int_{\infty}^{X_j} (X_j - S_j)^2 \ \phi(S_j; j\cdot 50, j\cdot 25) dS_j - 9 \int_{X_j}^{\infty} (X_j - S_j)^2 \ \phi(S_j; j\cdot 50, j\cdot 25) dS_j)]\right\}$$

$$+ 6\left\{\prod_{i=1}^{3} [2 \int_{-\infty}^{X_i} (X_i - S_i) \ \phi(S_i; i\cdot 50, i\cdot 25) dS_i - 3 \int_{X_i}^{\infty} (X_i - S_i) \phi(S_i; i\cdot 50, i.25) dS_i]\right\}.$$

In order then to find η such that Pr $(f(x,s) \leq \eta) = \alpha$, (3.54) is used. Then

$$Pr(X \leq X(\alpha)) = \alpha$$

implies

$$Pr(f(x,s) \leq Ef(x,s) + \sigma_f X(\alpha)) = \alpha$$

implies

$$Pr(f(x,s) \leq \sum_{i=1}^{3} P_i(X_i - X_{i-1}) + \sum_{i=1}^{3} Q_i(X_i) + \sigma_f X(\alpha)) = \alpha,$$

where

$$\sum_{i=1}^{3} P_i(X_i - X_{i-1}) = 3.75 - y_o + \hat{X}_3 = 388$$

then

$$\eta = 388 + Q^{(1)} + \sigma_f X(\alpha)$$

where (3.54) gives distribution from which the value $X(\alpha)$ is found.

Thus, the complexity of $T^{(2)}$ and $T^{(3)}$ in finding the first three moments to fit a probability distribution to f(x,s) introduces great complexity of computation even for a simple linear production cost function.

4. SUMMARY

The report has been concerned with the formulation of problems in mathematical programming. Specifically, those mathematical programs which have a linear objective function, those which are decomposable where the dynamic programming technique is appropriate, and the introduction of random variables into the above problems.

Linear programming was developed as a foundation for random variables to be introduced into the objective function or the constraints. Then a method of introducing reliability of the minimized function was developed with an outline of the computational procedure.

Dynamic programming was also developed with the purpose of introducing random variables into the system. Then Markov processes were presented within the dynamic programming theory.

It is certainly realized that there are other approaches to solving mathematical programming problems than those methods presented here. Some of these methods are the continuous maximum principle, the discrete maximum principle, the Lagrangian method, the Kuhn-Tucker method; some of which are included in the calculus of variation.

REFERENCES

Beale, E. M. L., "The Use of Quadratic Programming in Stochastic Linear Programming," RAND Corporation Paper P-2404, 1961.

Bellman, R. (1963). Mathematical Optimization Techniques. University of California Press.

Bellman, R. (1968). Some Vistas of Modern Mathematics. University of Kentucky Press.

Bellman, R. and Dreyfus, S. E. (1962). Applied Dynamic Programming. The RAND Corporation.

Buck, R. C. (1956). Advanced Calculus. McGraw-Hill Book Company, Inc.

Chung, An-min (1963). Linear Programming. Charles E. Merril Books, Inc., Columbus, Ohio.

Dantzig, G. B. and Madansky, A. "On the Solution of Two-Stage Linear Programs under Uncertainty", Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Vol. I, (J. Neyman, ed.), Berkeley, University of California Press, 1961.

Dantzig, B. G. and Wolfe, P. "The Decomposition Algorithm for Linear Programs," Econometrica, Vol. 29, 1961.

Dreyfus, S. E. (1965). Dynamic Programming and the Calculus of Variations. Academic Press, New York.

Erdelyi, I. (1968). Unpublished notes for "Topics in Computer Science".

Fan, L. T. (1966). The Continuous Maximum Principle. John Wiley & Sons, Inc., New York.

Fryer, H. C. (1966). Concepts and Methods of Experimental Statistics. Allyn and Bacon, Inc.

Fuller, L. E. (1962). Basic Matrix Theory. Prentice-Hall Inc.

Graves, R. L. and Wolfe, P. (1963). Recent Advances in Mathematical Programming. McGraw-Hill, Inc.

Gue, R. L. and Thomas, M. E. (1968). Mathematical Methods in Operations Research. The Macmillan Company, New York.

Hadley, G. (1962). Linear Programming, Addison-Wesley Publishing Company, Inc.

Hadley, G. (1964). Nonlinear and Dynamic Programming. Addison-Wesley Publishing Company, Inc.

Howard, R. A. (1960). Dynamic Programming and Markov Processes. The M.I.T. Press, Cambridge, Mass.

Jacobs, O. L. R. (1967). An Introduction to Dynamic Programming. Chapman and Hall LTD., London.

Kataoka, S. "On Stochastic Programming I. Stochastic Programming and its Application to Production Horizon Problem", Hitotsubashi J. Arts & Sciences, Vol. 2, 1962.

Lavi, A. and Vogl, T. P. (1966). Recent Advances in Optimization Techniques. John Wiley & Sons, Inc. New York.

Levine, K. C. and Thomas, W. R. (1968). Optimization of Continuous Expected Value Models. Bureau of Business and Economic Research, Georgia State College, Atlanta, Georgia.

Madansky, A. (1963) contained in Graves & Wolfe.

Madansky, A., "Inequalities for Stochastic Linear Programming Problems", Management Science, Vol. 6, 1960.

Madansky, A., "Methods for Solutions of Linear Programs under Uncertainty", Operations Research, Vol. 10, 1962.

Nemhauser, G. L. (1966). Introduction to Dynamic Programming. John Wiley & Sons, Inc., New York.

Parzen, E. (1962). Stochastic Processes. Holden-Day.

Savage, J. (1954). The Foundation of Statists. John Wiley & Sons, New York.

Simonnard, M. Translated by Jewell, W. S. (1966). Linear Programming. Prentice-Hall.

Stingler, G. J., "The Development of Utility Theory", Journal Political Economics, Vol. 58 (1950).

Tintner, G., "Stochastic Linear Programming with Application to Agricultural Economics", Second Symposium on Linear Programming, National Bureau of Standards, Washington, D.C., Vol. 1, 1955.

Tou, J. T. (1963). Optimum Design of Digital Control Systems. Academic Press, New York.

Trefethen, F. N. (1954). cited in McClaskey, J. F. and Trefethen, F. N. (1954) Operations Research for Management, Vol. 1. John Hopkins Press.

van der Veen, B. (1967). Introduction to the Theory of Operational Research. Cleaver-Hume Press. LTD., London.

Zukhovitskiy, S. I. and Avdeyeva, L. I., Translated by Scripta Technica, Inc. (1966). Linear and Convex Programming. W. B. Saunders Company.

ON LINEAR, DYNAMIC, AND STOCHASTIC PROGRAMMING

by

DON BRIAN CAMBELL

B.S., KANSAS STATE UNIVERSITY, 1965

B.S., KANSAS STATE UNIVERSITY, 1967

AN ABSTRACT OF A MASTER'S REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Statistics

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1969

ABSTRACT

The report is concerned with the formulation of problems in mathematical programming. Specifically, those mathematical programs which have a linear objective function, those which are decomposable where the dynamic programming technique is appropriate, and the introduction of random variables into the problems. Linear programming is developed as a foundation for random variables to be introduced into the objective function or the constraints. Then a method of introducing reliability of the minimized objective function is developed. Dynamic programming is also developed with the purpose of introducing random variables into the system. Then Markov processes are presented within the dynamic programming theory.