

1972

# Capital investment planning: an application of stochastic nonlinear programming

Edwin Lee Hullander  
*Iowa State University*

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Industrial Engineering Commons](#)

## Recommended Citation

Hullander, Edwin Lee, "Capital investment planning: an application of stochastic nonlinear programming" (1972). *Retrospective Theses and Dissertations*. 5256.  
<https://lib.dr.iastate.edu/rtd/5256>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

## INFORMATION TO USERS

This dissertation was produced from a microfilm copy of the original document. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the original submitted.

The following explanation of techniques is provided to help you understand markings or patterns which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting thru an image and duplicating adjacent pages to insure you complete continuity.
2. When an image on the film is obliterated with a large round black mark, it is an indication that the photographer suspected that the copy may have moved during exposure and thus cause a blurred image. You will find a good image of the page in the adjacent frame.
3. When a map, drawing or chart, etc., was part of the material being photographed the photographer followed a definite method in "sectioning" the material. It is customary to begin photoing at the upper left hand corner of a large sheet and to continue photoing from left to right in equal sections with a small overlap. If necessary, sectioning is continued again — beginning below the first row and continuing on until complete.
4. The majority of users indicate that the textual content is of greatest value, however, a somewhat higher quality reproduction could be made from "photographs" if essential to the understanding of the dissertation. Silver prints of "photographs" may be ordered at additional charge by writing the Order Department, giving the catalog number, title, author and specific pages you wish reproduced.

### **University Microfilms**

300 North Zeeb Road  
Ann Arbor, Michigan 48106

A Xerox Education Company

72-26,921

HULLANDER, Edwin Lee, 1937-  
CAPITAL INVESTMENT PLANNING: AN APPLICATION  
OF STOCHASTIC NONLINEAR PROGRAMMING.

Iowa State University, Ph.D., 1972  
Engineering, industrial

University Microfilms, A XEROX Company, Ann Arbor, Michigan

**Capital investment planning:  
An application of  
stochastic nonlinear programming**

**by**

**Edwin Lee Hullander**

**A Dissertation Submitted to the  
Graduate Faculty in Partial Fulfillment of  
The Requirements for the Degree of  
DOCTOR OF PHILOSOPHY**

**Major: Engineering Valuation**

**Approved:**

Signature was redacted for privacy.

**In Charge of Major Work**

Signature was redacted for privacy.

**For the Major Department**

Signature was redacted for privacy.

**For the Graduate College**

**Iowa State University  
Ames, Iowa**

**1972**

PLEASE NOTE:

Some pages may have  
indistinct print.

Filmed as received.

University Microfilms, A Xerox Education Company

## TABLE OF CONTENTS

	<b>Page</b>
INTRODUCTION	1
REVIEW OF LITERATURE	3
CAPITAL INVESTMENT PLANNING MODELS	41
SUMMARY AND CONCLUSIONS	75
LITERATURE CITED	77
ACKNOWLEDGMENTS	83
APPENDIX	85

## INTRODUCTION

The planning of capital investments should consider three factors: (1) the proper timing of cash flows, (2) the interrelationship between projects, and (3) the risk associated with committing capital funds. Traditionally, capital budgeting consisted of finding a rate of return or present worth for each project ranking them and selecting only those that fall within the acceptable rate of return or budget constraints. The procedure avoided the effects of interrelationship of projects by forcing the assumption of independence. Therefore, those interdependent projects were grouped as one large project. Secondly, the risk associated with an investment was treated in the context of the investor's utility function. Finally, capital budgeting has been static in nature, planning for capital expenditures one year at a time. The primary restriction of the development of new and more comprehensive techniques was the limitation of computation. The characteristics of investment planning models are generally complex nonlinear functions.

With the advent of large-scale computer systems, the computational restrictions have been relieved. The development of new methods to analyze risky interrelated investments was pioneered by Hillier, who employs chance-constrained programming for the analysis of risk. However, one of the problems of chance-constrained programming is the assumption

that the random variable is normally distributed. This is not always the case in models of capital investments.

This study will be concerned with the development of a chance-constrained model that employs a nonnegative (chi-square) distributional assumption. The computational difficulties usually associated with this approach will be handled by geometric programming.

The study will review the current literature on capital investment planning and the techniques used for analysis. A chance-constrained programming model, using the chi-square assumption, is then developed and illustrated in two problems of investment planning. The first, a portfolio model, is formulated and its solutions compared with previous solutions using other assumptions and procedures. Next a capital budgeting problem is developed to analyze both the risk of actual losses as well as opportunity loss. The study is concluded with a summary and discussion.



## REVIEW OF LITERATURE

The evaluation of risk associated with an investment project is a very important consideration in capital budgeting. This is especially true when the impact of failure could significantly change the financial position of the organization. While the independence of investments can be achieved by diversification in portfolio investments, it is much more difficult to guard against failure in capital budgeting. The desirability of one investment project is often interrelated with the performance of other investment projects. This interrelationship may be of a competitive nature, where the introduction of a new product would compete with existing products in the same market. On the other hand, they may be complementary, where as a new product may share common facilities or technology, thereby sharing the cost. The revenues resulting from each of the investment projects must be correlated because their incomes are affected by the common factors. Those factors could be internal such as shared facilities or external such as the general state of the economy. The interrelationship between investment projects directly affects the total risk to the investment plan. Therefore, any capital investment decisions should give consideration to the interrelationship of projects and their subsequent risk.

The general framework for current analysis of risky investments was developed by Lutz and Lutz (42) and later synthesized by Farrar (26) for testing particular investment models as well as providing a rigorous survey of the work done up to that time.

Markowitz (46) treated a special case of interrelated risky investments in his analysis of portfolios containing a large number of securities. The portfolio model was formulated as a static model that assumed a deterministic equivalent of a risky or uncertain model. Markowitz illustrated how to determine the portfolio configuration that provided the most suitable combination of rate of return and standard deviation of rate of return. His work subsequently motivated the work of Cheng (19), Sharpe (63), Baumal (4), Fama (25), and Mao and Sarndal (45). Extension of the portfolio idea was developed by Naslund and Whinston (53) based on the risk programming concept of Charnes and Cooper (10).

Weingartner's (72) treatment of the capital budgeting model, under certainty conditions but in a capital rationing and imperfect capital market, provided the foundation of later work by Naslund (51), Byrne (8), and others. Weingartner's (71) survey of papers on evaluation of interrelated investments provides a good cross section of current developments.

The concept of incorporating the interrelationship of investment opportunities with the subsequent risk involved was put forth by Hillier (36) in which he formulated an investment

model that would generate one or more mutually independent series of cash flows that were assumed to be normally distributed. The cash flow within each series was assumed to be either mutually independent or perfectly correlated. The work was extended to the development of the probability distribution of the present value and suggestions for how it might be utilized in a decision process. This approach was carried on by Hertz (31), Hillier (37), Horowitz (39), and Hespos and Strassmann (32).

The two primary characteristics of capital investment planning, according to Hillier (36), are the interrelationship of investment proposals and their subsequent risk. The following discussion of these two areas will provide a foundation for the models to be developed in later sections.

#### Consideration of Interrelationship of Investments

Consider the case where a number of capital investment proposals are presented to management for consideration. The decisions made will most likely affect the long-term growth of the organization; therefore, a good deal of thought goes into the planning of capital investments. The criterion for project selection must be one that incorporates the effects of uncertainty or risk, the goals and objectives of the organization, and the interrelationship of the investment proposals.

The decision to accept or reject a project at a given time is more of a "go or no-go" type of decision in capital

budgeting; whereas, the decision in portfolio selection is "how much" or what percentage of the portfolio should be of a certain type of stock. The former decision rule should not be viewed as restrictive since projects can be postponed and decomposed into phases (pilot plant, product plant, and etc.) where the latter is dependent upon the former. Also, alternative strategies could be formulated, such as, start a new product in a number of different configurations. Then if losses drop below a given level, drop the losers and gear up for the others.

Now consider an investment decision  $x_{ij}$  to be of the form

$$x_{ij} = \begin{cases} 1, & \text{if the } i^{\text{th}} \text{ investment is accepted at time } j \\ 0, & \text{if the } i^{\text{th}} \text{ investment is rejected at time } j \end{cases} \quad (2.0)$$

where the number of investment projects are  $i = 1, 2, \dots, I$  and the decision periods are  $j = 1, 2, \dots, J$  or the planning horizon. For those combinations of investments that are mutually exclusive either by design or by chance, the constraint on the decision variables can be expressed as

$$\sum_i x_i \leq 1, \quad \text{for all } i \in k. \quad (2.1)$$

On the other hand, if one investment project is contingent upon another being approved, then the constraint is

$$x_{ij} \leq x_{ij-1}, \quad (2.2)$$

where the decision at  $x_{ij-1}$  must be affirmative before the latter decision  $x_{ij}$  can be considered. By imposing this type of side condition, investment planning can be linked together

in a sequential decision process. Hillier (36) has discussed the use of dynamic programming for planning investment programs based on this type of decomposition.

The primary purpose of adding the decision structure to investment planning models is to evaluate the subsequent cash flow that is generated. Assume that the immediate cash flow starts at some period  $j$  and is evaluated at the end of each subsequent period. The number of periods that the cash flow is evaluated ( $k = 0, 1, 2, \dots$ ) from the time of investment to the present is the total net cash flow  $X_k(x)$  (total positive inflows minus total negative outflows). Thus, we can think of the net cash flow  $X_k(x)$  as the result of a sequence of decisions concerning a project  $i$  or set of projects that occurred at various times ( $j = 1, 2, \dots$ ). The decision at each point in time initiated a cash flow stream which occurred over  $k$  intervals of times (say years). The net value of the cash flows is evaluated at the present as

$$X_k(x) = \sum_k X_k(x), \quad k = 1, 2, \dots, K. \quad (2.3)$$

Hillier points out two important facts about the net cash flow. First, that the cash flow stream resulting from a decision is actually an aggregation of many distinct cash flow streams some of which are interrelated. Also, the decision at some point in the process may itself be dependent on previous cash flow streams that are aggregations. Secondly, the resulting cash flow streams usually are random variables giving the

model the inherent risk or uncertainty characteristics. Therefore, the cash flow that results from an affirmative decision may take on a range of values causing the net cash flow to be described in terms of a probability distribution. Hillier continues this approach by developing the distribution of the discounted cash flow or the present worth and employs this criteria in a utility maximization model.

Before we can look at the distribution of the total net cash flow, we should investigate the cash flow of the individual investment. First, consider an investment project independent of others and that the decision to initiate the project occurred at some time  $j_0$ . The actual investment cost at  $j_0$  and all subsequent investment costs (negative cash flow) may be random variables. Likewise, the returns from the investment can also be viewed as a random variable at each period  $k$ . If we estimate the cash flow by its mean  $\mu_{ij}$  and variance  $\sigma_{ij}^2$  of some probability distribution, then the net cash flow is the difference of two random variables estimated by their mean and variance.

For the present worth case, the expected present worth is merely the sum of the discounted mean cash flows. However, the variance presents problems in the correlation between cash flows from the same source but in different periods. Various methods of handling the correlation between cash flows have been described in (31 and 35).

Suppose one has estimated the mean  $\mu_{ij}$  and variance  $\sigma_{ij}^2$  of the net cash flows for all investment proposals independently. If there is no interrelationship between the proposals, then we have no problem in determining the mean and variance of the present worth for each proposal. If, on the other hand, the presence of some kind of interaction between proposals exists, it can invalidate the simple additivity assumption. For example, two proposals were estimated independently and both found attractive. However, in combination they were found to be competitive, thus, either or both became no longer attractive. Conversely, a proposal may be unattractive by itself, but in conjunction with another project, may be very attractive. In both cases, the analysis of the investment proposal in isolation can result in misleading decisions. Thus, the criteria for investment decisions should be modified to incorporate the effects of interaction.

To formulate a cash flow that includes this effect, let  $h(x)$  be defined as the net amount by which the individual cash flows  $X_k(x)$  will be adjusted due to complementarity (positive adjustment) or competitive interaction (negative adjustment). Since the cash flows are random variables, it is logical to make the assumption that  $h(x)$  will also be a random variable. Also, the effect of a proposal interacting with more than one other proposal can be expressed in a "pairwise" combination such that the total effect is the sum of the pairwise effects.

Therefore, we can define

$$h(x) = \sum_1 \sum_1 \mu_{11}(x_1)(x_1), \quad i = 1, 2, \dots, I, \quad (2.4)$$

$$1 = 1, 2, \dots, I,$$

where  $i \neq 1$  and  $(\mu_{i1} + \mu_{1i})$  is the net addition (positive or negative) to the total net cash flow due to complementarity between the two proposals ( $i$  and  $1$ ) if both are accepted.

Therefore, the total cash flow for the two proposals ( $i$  and  $1$ ) can be expressed as  $(\mu_i + \mu_{i1})$  and  $(\mu_1 + \mu_{1i})$ , respectively.

The complementary effect of each proposal can be thought of as their equal share in the total effect ( $\mu_{i1} = \mu_{1i}$ ). Thus, the net cash flow can be generalized as

$$X_k(x_{ij}) = \sum_1 \sum_k X_k(x_{ij}) + h(x), \quad i = 1, 2, \dots, I, \quad (2.5)$$

$$k = 1, 2, \dots, K.$$

Expressing the cash flow in terms of its mean and variance, we have

$$\bar{X}_k(x) = \sum_1 \sum_k (\mu_1 + \sum_1 \mu_{11} x_1) x_1, \quad (2.6)$$

and

$$V_k(x) = \sum_1 \sum_k (\sigma_1^2 + \sum_1 \sigma_{11}^2 x_1) x_1, \quad (2.7)$$

where  $i = 1, 2, \dots, I$  and  $k = 1, 2, \dots, K$ .

In general, the above expressions only mean that the estimated cash flow of a proposal at each time interval must be augmented by the interaction effect. Now that the mean and variance of the total net cash flow can be found, the next question is to determine its probability distribution. In the more general case put forth by Hillier (35, 36, and 37) based



on the central limit theorem, he makes a strong argument for assuming the random variables to be normally distributed. One primary reason for making this assumption is that linear combinations of normal random variables are also distributed normally. Also, the central limit theorem indicates that the sum of a series of random variables, having distributions other than normal, can be approximated as a normal distribution under certain conditions.

For certain types of investment models, the normality assumption is very sound. In others, however, the conditions under which the normality assumption is made are not so readily acceptable. In a later section, we shall discuss the effects of using the deterministic equivalent of the random variable in a stochastic programming model.

The development of estimates for interrelated cash flows and the introduction of the investment decision function allows for a great deal more flexibility in planning capital investments. However, this decomposition of the model brings with it more difficulty in computation. This computational difficulty is compounded when considering the problems of risk analysis. The next section will discuss briefly three methods of risk analysis generally referred to as stochastic programming which have been employed in the analysis of capital investments.

### Consideration of Risk in Investments

Mathematical programming can be thought of as stochastic if one or more of the coefficients in the set  $(A, b, c)$  are random variables with a specific probability distribution.

When the probability distribution of the parameters is known or a priori specified, then an important class of decision problems can be formulated to answer such questions as:

(1) how to decide on a decision vector which is in some sense optimal and (2) how to characterize the sensitivity of the decision vector to variation of the parameters.

These questions and others have been approached in the literature on stochastic programming. The research, to date, can be divided into three major areas: (1) stochastic linear programming (SLP), (2) two-stage linear programming under uncertainty (LPUU), and (3) chance-constraint programming (CCP). Generally speaking, all three approaches have the following common characteristics; that is, they incorporate the initial probability distribution of the parameters in order to convert a probabilistic linear program into a deterministic form and then define a set of decision rules having some optimality properties. Of course, the methods by which they incorporate the probability distribution and specify the decision rules are different for each approach.

If the distribution of the parameters is unknown, the problem of defining the characteristics of the optimal vector becomes very difficult. Cases of this nature have been

treated by simulation techniques (74) or in the context of game theory (41 and 49). However, when the probability distribution of the parameters is known or specified, there are three basic approaches to incorporating the random variable into the framework of mathematical programming.

If we assume that sample information is available, however the sample statistics of the parameters are unknown at the time of the decision, then the sample distribution of the activity vector  $x$  becomes dependent upon: (1) the restrictions of the random elements imposed by feasibility, (2) the sample design, and (3) the form of the population distribution. Problems of this nature have been treated in the general framework of stochastic linear programming (69).

If we consider the decision vector  $x$  to be nonstochastic in the sense that we must determine the optimal solution for the vector  $x$  given the random variation of the parameters  $(A, b, c)$ , then the specification of the decision maker's attitude towards risk becomes very important (48). This general area has been approached as chance-constraint programming (CCP) and safety-first programming (SFP).

On the other hand, if we decompose the problem into two stages to obtain an approximation, the first stage employs the certainty equivalent of the random variable in the context of an ordinary linear programming model. Then, the second stage defines a penalty function that modifies the deterministic approximation to incorporate the effect of the random

variable (70). This approach is usually termed two-stage linear programming under uncertainty (LPUU).

A brief review of these methods and their application to risk analysis in general and to capital investment models in particular follows.

Stochastic linear programming was first suggested by Tintner (65 and 66) and was concerned with finding the statistical distribution of the optimal solution of a model such as

$$\max Z = c'x, \quad (2.8)$$

subject to

$$Ax \leq b, \quad (2.9)$$

$$x \geq 0. \quad (2.10)$$

Assuming the multivariate probability distribution for the elements  $A$ ,  $b$ ,  $c$  is known, then the probability of simultaneous occurrence of specified values of the matrix  $A$  and the vectors  $b$  and  $c$  can be expressed as

$$\text{prob} (A, b, c). \quad (2.11)$$

Tintner developed both a passive approach and an active approach to finding the distribution of the optimal solution from the multivariate probability function (2.11).

The passive approach assumes that all combinations of the random variables producing an optimal activity can be found. Then it is possible to derive the probability distribution of the optimal solution that is comprised of the set of optimal activities  $x$ . Since the assumption of independence of the

coefficients is implicit in the programming model, only the linear terms of the Taylor expansion is needed to find the distribution of  $Z_{\max}^*$ . The confidence interval for the expected value of the function has been developed by Tintner (65), Babbar (2 and 3), and extended by Sengupta (60 and 62) and others. From a computational viewpoint, simulation has been used to generate the values of the random variables which are used to solve an ordinary linear program. By repeated simulation runs, a density function for  $Z_{\max}^*$  can be developed.

The active approach to stochastic linear programming transforms the problem into a decision or policy model. If we modify the above model to the following form

$$\max Z = c'x, \quad (2.12)$$

subject to

$$Ax \leq bD, \quad (2.13)$$

$$x \geq 0, \quad (2.14)$$

where  $D$  is a matrix with all elements

$$0 \leq d_{ij} \leq 1 \text{ and } \sum_j d_{ij} = 1, \quad j = 1, \dots, J. \quad (2.15)$$

The decision matrix  $D$  is composed of decision variables  $d_{ij}$  which denote the proportional allocation of the  $i^{\text{th}}$  resource assigned to  $j^{\text{th}}$  activity assuming that all resources are fully utilized. The objective is to choose a best set of  $d_{ij}$  values for the matrix  $D$  that maximizes the objective function in accordance with the preference function of the decision maker.

Application of stochastic linear programming to problems of resource allocation can be found in (50 and 51). The problem of decision analysis has been treated in (66). For a critical appraisal of stochastic linear programming, see Sengupta and Tintner (61).

Certain types of stochastic programming problems, when examined closely, can be decomposed into two or more stages. By separating the problem into stages, a decision rule or strategy can be employed to govern the reaction to any given value of the uncertain event. This approach has been termed two-stage linear programming under uncertainty (LPUU). The basic approach to LPUU is to approximate the optimal solution to the problem by assuming the parameters are deterministic or assigned; this is the first stage. The second stage incorporates the effect of the random variable by modifying the first stage solution. The model can be expressed as follows (70)

$$\max \sum_j E(c_j)x_j + \sum_q \text{prob}_q (\sum_l c_{ql}x_{ql}), \quad (2.16)$$

subject to

$$\sum_j a_{ij}x_j = b_i, \quad (\text{first stage constraint}) \quad (2.17)$$

$$\sum_j a_{qij}x_j + \sum_l a_{qml}x_{ql} = b_{qm}, \quad (2.18)$$

(second-stage decision rule)

$$x_j, x_{ql} \geq 0, \quad (2.19)$$

where  $q = 1, 2, \dots, Q$  (no. of stages),  $j = 1, \dots, K$  and  $l = k + 1, \dots, K$  (variable set), and  $i = 1, \dots, q$  and  $m = q + 1, \dots, G$  (constraint set).

The values of  $x_j$  are fixed in the first stage before the exact value of the random variable is known. The constraints (2.17) contain only the first stage terms with the parameters assumed to be known. For the second stage and all subsequent stages, there always exists a feasible level that can be determined after all the random variables are known. Also, there is a finite number of stages or possible sets of values of the parameters  $(c_{q1}, a_{qnl}, b_{qm})$ . Each set of the parameters' values can be weighed by the probability of their occurrence  $\text{prob}_q$ . Notice in the second-stage decision rule constraint (2.18) there are  $(G - q) Q$  equations. Thus, as the number of stages increases, the problem becomes computationally more difficult.

Linear programming under uncertainty has been applied to many areas where a decision rule is highly desirable for planning purposes (20, 24, 27, and 43). Because of the decomposition principle explicit to its formulation, LPUU has been combined with other techniques that offset its computational disadvantages. Avriel and Wilde (1) combined geometric programming and two-stage linear programming under uncertainty to handle a broad class of nonlinear stochastic problems. Hillier's (36) work on interrelated risky investments also employed LPUU in conjunction with chance-constraint programming to handle multi-stage investment planning models. Byrne et al. (3) proposed a similar approach to capital budgeting problems.

The concept of chance-constraint programming was first introduced by Charnes and Cooper (11) as part of a model for scheduling the production of heating oil to meet an uncertain demand. While the statistical distribution of demand was known, its high degree of variation exceeded the bounds of the scheduling constraints making deterministic programming unsuitable. A new approach was needed which would replace the precise deterministic constraints by one that embodied the intent of the management policy, not the hard and fast rule. Thus, this new approach needed to represent bounds inside of which management would like to operate "most of the time" but not exactly "all of the time".

The resulting chance-constraint concept requires a constraint to hold with at least a specified level of probability but not necessarily with probability of one. This characteristic distinguishes chance-constraint programming from the previously mentioned linear programming under uncertainty. The latter requires that all possible combinations of values of the random variables must have a probability of one of occurrence. The concept of decision rules which result from solving a chance-constrained problem are designed to present a plan of action that is good most of the time but not all of the time.

The exact nature of the decision rule is dependent, in part, on the possibility of sample points inconsistent with the constraints. In general, our object is to find an optimal



vector of stochastic decision rules

$$X = \phi(A, b, c), \quad (2.20)$$

of the generalized chance-constrained model

$$\max Z = c'x, \quad (2.21)$$

subject to

$$\text{prob}(A_x \leq b) \geq \alpha, \quad (2.22)$$

where  $\alpha$  is the specified tolerance limit of the constraint such that it may be violated as more than  $100(1 - \alpha)\%$  of the time. The parameters  $A$ ,  $b$ ,  $c$  are defined as before.

The linear decision rule  $X$  is based on the premise that the function  $\phi$  is selected from a prescribed class of functions in which the matrix  $A$  and vector  $c$  contain only constant elements and  $X$  is restricted to being a linear function of the random variables in  $b$ . Much of the earlier work was based on this type of linear decision rules (10, 11, and 12).

Another type of decision rule which often arises in budgetary planning models is called the zero order decision rule. In this type of rule, the decision vector is not permitted to be an explicit function of any of the random variables involved in the model. In such cases, the decision maker wants to know all of his program values in advance of any observations being made on the random variable. Applications of this decision rule are numerous; for example, see (14, 17, and 18).

A more recent decision rule that, in part, follows the decomposition concept of two-stage linear programming under

uncertainty is the general n-period decision rule. Specifically, the rule contends that the decision required at the  $j^{\text{th}}$  period does not have to be made until the beginning of that period. Thus, it is desirable to have decision rule  $X_j$  determined in a conditional manner or the experience accumulated through all previous periods, as well as implicitly reflecting future possible states of the system. Therefore,  $X_j$  is allowed to be a function of random variables observed in previous periods but not of the  $j^{\text{th}}$  or subsequent periods. In this way,  $X_j$  maximizes the use of information accumulated up to the time the decision rule must be implemented. Specific examples of this decision rule can be found in (15).

The basic core of literature on chance-constraint programming centers around the use of linear decision rules and normal random variables. The basic objective of the program is to convert the chance-constraint model into a deterministic equivalent linear or nonlinear programming model. The rationale for these assumptions is that they led to a compatible linear or nonlinear problem. The general method for obtaining the deterministic equivalent for a chance-constrained problem can be developed as follows.

Determine the decision vector  $x$  that

$$\min \sum_j c_j x_j, \quad (2.23)$$

subject to

$$\sum_j a_{ij} x_j \geq b_i, \quad i = 1, \dots, I, \quad (2.24)$$

$$x_j \geq 0, \quad j = 1, \dots, J, \quad (2.25)$$

where  $a_{1j}$ 's are the constraint coefficients, the  $b_1$ 's are the resources available, and  $c_j$ 's are the elements of the objective function. The chance-constraint formulation can be developed from the above general form of the mathematical programming model by assuming the constraint coefficients  $a_{1j}$  are random variables with normal distribution. The probability that the constraint inequality containing the random variable must be satisfied is denoted as  $\alpha_1$ . Thus, the constraint set (2.24) can be stated as

$$\text{prob} (\sum_j a_{1j} x_j \geq b_1) \geq \alpha_1, \quad i = 1, \dots, I, \quad (2.26)$$

$$j = 1, \dots, J.$$

Let us assume for the  $i^{\text{th}}$  constraint that the  $a_{1j}$ 's are independent random variables with means  $\bar{a}_{11}, \dots, \bar{a}_{1J}$  with variances  $\sigma^2(a_{11}), \dots, \sigma^2(a_{1J})$ . Thus, we can redefine the  $i^{\text{th}}$  constraint as

$$u_1 = \sum_j a_{1j} x_j, \quad j = 1, \dots, J. \quad (2.27)$$

This variable is normally distributed with mean

$$\bar{\mu}_1 = \sum_j \bar{a}_{1j} x_j, \quad (2.28)$$

with variance

$$\sigma^2(u_1) = \sum_j \sigma^2(a_{1j}) x_j^2, \quad j = 1, \dots, J. \quad (2.29)$$

Each constraint can be restated as

$$\text{prob} (u_1 \geq b_1) \geq \alpha_1. \quad (2.30)$$

By expressing the left side of the inequality in terms of the standard function, we have

$$\text{prob } (u_i \geq b_i) = \int_{b_i}^{\infty} h(u_i) du_i, \quad (2.31)$$

where  $h(u_i)$  is the normal density function of  $u_i$ . By setting

$$Z_i = \frac{u_i - \bar{\mu}_i}{(\sigma^2(u_i))^{1/2}} = \frac{u_i - \sum_j \bar{a}_{ij} x_j}{(\sum_j \sigma^2(a_{ij}) x_j^2)^{1/2}}, \quad j = 1, \dots, J, \quad (2.32)$$

and substituting the above in the lower limit of integration of (2.31), we obtain

$$\text{prob } (u_i \geq b_i) = \int_r^{\infty} f(Z_i) dZ, \quad (2.33)$$

where

$$r = \frac{b_i - \sum_j \bar{a}_{ij} x_j}{(\sum_j \sigma^2(a_{ij}) x_j^2)^{1/2}}, \quad j = 1, \dots, J, \quad (2.34)$$

and  $f(Z)$  is the standardized normal density function

$$f(Z) = \frac{1}{(\sigma\pi)^{1/2}} \cdot \exp(-\frac{1}{2}Z^2). \quad (2.35)$$

In terms of the standardized normal left-tail cumulative function

$$\text{prob } (u_i \geq b_i) = 1 - F\left[\frac{b_i - \sum_j \bar{a}_{ij} x_j}{(\sum_j \sigma^2(a_{ij}) x_j^2)^{1/2}}\right], \quad (2.36)$$

returning to the constraint form (2.30), we can express the left-hand side as

$$F\left[\frac{b_i - \sum_j \bar{a}_{ij} x_j}{(\sum_j \sigma^2(a_{ij}) x_j^2)^{1/2}}\right] \leq 1 - \alpha, \quad j = 1, \dots, J, \quad (2.37)$$

employing the inverse function, we obtain

$$\frac{b_1 - \sum_j \bar{a}_{1j} x_j}{(\sum_j \sigma^2 (a_{1j}) x_j)^{\frac{1}{2}}} \leq F^{-1}(1 - \alpha). \quad (2.38)$$

For simplicity, let the expression (2.38) be equivalent to  $\phi(\alpha)$  where we can define  $\phi(\alpha)$  as the percentage or fractile of tolerance for each constraint, e.g., if  $\alpha = .95$ , then  $\phi(\alpha) = .05$ . We can now write the deterministic equivalent constraint as

$$\sum_j \bar{a}_{1j} x_j + \phi(\alpha) (\sum_j \sigma^2 (a_{1j}) x_j^2)^{\frac{1}{2}} \geq b_1. \quad (2.39)$$

To illustrate the use of chance-constraint programming, we can use the problem of determining the optimal mix of cattle feed at minimal cost. This problem is well known in the literature of linear programming (6). The problem is concerned with finding the optimal mix of raw materials that meets the nutrient requirements at minimal cost. The data for the problem is given below:

	$x_1$	$x_2$	$x_3$	$x_4$	$b_1$
	Barley	Oats	Sesame flakes	Ground-nut meal	Requirement
Percent protein ( $\bar{a}_{1j}$ )	12.00	11.90	41.80	52.10	21
$(\sigma^2(a_{1j}))$	0.28	0.19	20.50	0.62	
Percent fat ( $a_{1j}$ )	2.30	5.60	11.10	1.30	5
Cost per ton (guilders)	24.55	26.75	35.00	40.50	

The problem can be formulated as

$$\min Z = 24.55x_1 + 26.75x_2 + 39.00x_3 + 40.50x_4, \quad (2.40)$$

subject to

$$12.00x_1 + 11.90x_2 + 41.80x_3 + 52.10x_4 + (-1.645) \\ (.28x_1^2 + .19x_2^2 + 20.5x_3^2 + .62x_4^2)^{\frac{1}{2}} \geq 21, \quad (2.41)$$

$$2.30x_1 + 5.60x_2 + 20.50x_3 + 1.30x_4 \geq 5, \quad (2.42)$$

$$x_1 + x_2 + x_3 + x_4 = 1, \quad (2.43)$$

$$x_1, x_2, x_3, x_4 \geq 0. \quad (2.44)$$

For more detail of the problem, see (6).

Comparing the solution of the linear form and the stochastic form below, we can see that by relaxing the constraints the optimal cost is changed as well as the values of the optimal variables. Note that the optimal cost increased in the stochastic case. This increase can be attributed to the consideration of risk or uncertainty in the problem. The previous linear model, by ignoring the uncertainty aspect, compromised the solution and any subsequent decisions.

<u>Linear case</u>	<u>Stochastic case</u>
$Z^*_{\max} = 28.94$	$Z^*_{\max} = 29.89$
$x_1 = .6852$	$x_1 = .6359$
$x_2 = .0127$	$x_2 = 0$
$x_3 = .3021$	$x_3 = .3127$
$x_4 = 0$	$x_4 = .0515$

The applications of chance-constraint programming to problems of decision analysis and resources allocation are summarized in (40). Since the concept of chance-constraint is particularly applicable to the decision problems in financial planning, it is worth mentioning a few typical models found in the literature.

One primary problem in investment planning is the liquidity requirements. Models, where the liquidity condition is chance constrained, can be seen in (4, 7, and 8). In conjunction with the liquidity conditions, a group of problems dealing with the extent that borrowing and lending can take place in an investment model have been examined in (18). The classical portfolio model has been extended to chance-constraint programming (51, 52, and 53). The primary characteristics of this type of model are the loss constraint and the capital available constraints. Research and development planning have also been investigated using this concept (17). Here the main feature is the way in which the model takes into account the possibility of a "breakthrough".

Finally, capital budgeting problems have been explored by (7, 8, 52, and 53) under many assumptions. Byrne et al. (8) examines the use of payback methods as being chance constrained to study the recovery rate of the initial investment. Other financial planning models can be found in (40).

The three methods of stochastic programming, discussed in this section, have a common problem, that of difficulty in

computation. The linear programming under uncertainty (LPUU) presents problems in evaluating the second stage where all possible combinations of the random variables are expressed. Stochastic linear programming (SLP) and chance-constraint programming (CCP) present computational difficulties by the introduction of nonlinear terms in the model. Computational procedures for nonlinear programming, until recently, have been lacking in their ability to handle complex problems such as those present in investment planning. Recently, however, the development of two computational methods of handling general nonlinear programming problems shows a great deal of promise. The two methods, sequential unconstrained minimization technique (SUMT) and geometric programming, are discussed in the next section.

#### Computational Aspects of Risk Analysis

Mathematical programming, in general, has been said to be the logical extension of classical optimization theory, formulated in such a manner as to facilitate the use of digital computer systems. A general statement of the mathematical programming problem is to find a vector  $x$  that solves the problem

$$\min f(x), \quad (2.45)$$

subject to

$$g_i(x) \geq 0, \quad i = 1, 2, \dots, I. \quad (2.46)$$



This constrained optimization problem can be solved by means of the Lagrange multiplier technique (23 and 75). This method, however, requires that all constraints be exact equalities. Therefore, the constraint equation must be modified, without loss of generality, to include a slack variable that will convert the inequality into an equality constraint

$$g_i(x) - x_s = 0. \quad (2.47)$$

Now the problem can be formulated as a Lagrangian function

$$L(x, \lambda) = f(x) - \sum_i \lambda_i (g_i(x)), \quad i = 1, \dots, I, \quad (2.48)$$

where  $\lambda_i$  is the Lagrange multiplier of the constraint function. By solving this function for all values of  $x$  and  $\lambda$ , a local and global optimal can be found. This problem, when transformed into the Lagrangian form, becomes an unconstrained optimization problem that can be solved by ordinary calculus.

The computational problem with the Lagrangian method is that all feasible combinations of  $x$  and  $\lambda$  must be found before the global optimal is ascertained. Computational difficulties are compounded by the increasing size of the model, i.e., the number of variables and constraints and also by the introduction of nonlinearity. Therefore, direct Lagrangian solutions are, to date, not computationally feasible for large problems.

Fiacco and McCormick (29) state that the Lagrange method is inextricably associated with every computational method of mathematical programming. For this reason, computational

techniques to handle complex nonlinear optimization models have been based on the Lagrangian theory. Two current methods which will be discussed here are the sequential unconstrained minimization technique (SUMT), developed by Fiacco and McCormick (29), and geometric programming, developed by Duffin, Peterson, and Zener (23).

Fiacco and McCormick developed their computational technique on an idea proposed by Carroll (9). The general form of SUMT can be expressed in the following way. Find a vector  $x$  that will

$$\min f(x), \quad (2.49)$$

subject to

$$g_i(x) \geq 0, \quad i = 1, 2, \dots, m, \quad (2.50)$$

$$h_j(x) \geq 0, \quad j = m+1, \dots, M, \quad (2.51)$$

where there exists at least one point  $x$  such that  $g_i(x) > 0$  for  $i = 1, \dots, m$ . The algorithm defines an unconstrained auxiliary function

$$\text{prob}(x, r_1) \equiv f(x) + r_1 \sum_i (g_i(x))^{-1} + r_1^{-\frac{1}{2}} \sum_j g_j^2(x), \quad (2.52)$$

where  $r_1 \geq 0$ ,  $i = 1, \dots, m$ , and  $j = m+1, \dots, M$ .

The auxiliary function is the same as the Lagrangian function given above. They differ, however, in their computational procedure. For example, as a starting point, let  $x_0$  satisfy the condition of (2.50); proceed from  $x_0$  to a point  $\bar{x}(r_1)$  that approximates the minimum of  $\text{prob}(x, r_1)$  within the set of points satisfying  $g_i(x) \geq 0$ ,  $i = 1, \dots, m$ . Next form a

new function

$$\text{prob}(x, r_2) \equiv f(x) + r_2 \sum_i (g_i(x))^{-1} + r_2^{-k} \sum_j g_j^2(x), \quad (2.53)$$

where  $r_1 > r_2 > 0$ ,  $i = 1, \dots, m$ , and  $j = m+1, \dots, M$ .

Starting from  $x(r_1)$ , approximate the minimum value of  $\text{prob}(x, r_2)$ . By continuing this procedure, a sequence of points  $(x(r_k))$ ,  $k = 1, 2, 3, \dots$ , can be generated that respectively minimizes the auxiliary function  $\text{prob}(x, r_k)$  where  $r_k$  is monotonically decreasing to zero. The basic postulate proven by Fiacco and McCormick (29) is that the sequence of unconstrained minima  $(x(r_k))$  will approach an optimal solution to a mathematical programming problem of the form defined above. The rationale for SUMT is given by Bracken and McCormick (6) and is as follows.

The second term in (2.53) can be thought of as a penalty factor attached to the objective function  $f(x)$  and assures that a minimum of the auxiliary function is achieved in the interior of the inequality-constrained region. This is accomplished by balancing the avoidance of boundaries and minimization of  $f(x)$ . To illustrate, consider the trajectory of points that tend to minimize  $\text{prob}(x, r_1)$  starting at  $x_0$ . The locus of these minima define a curve on which the  $\text{prob}(x, r_k)$  is continually decreasing; therefore, no point on the trajectory can exceed the initial value of  $\text{prob}(x_0, r_1)$ . The feasible boundary is defined by one or more of the  $g_i(x) = 0$ . It can be shown that the value of the auxiliary

function goes to positive infinity as the boundary is approached from the interior region. Consequently, the boundary can never be pierced by the trajectory and the minimum of  $\text{prob}(x, r_1)$  must be a feasible interior point. Along the same line, the third term in (2.53) can be thought of as a barrier function.

As  $r_k$  goes to zero, the third term would go to infinity unless each  $g_1(x(r_k))$  is zero, in which case the auxiliary function would force the  $g_1$ 's to zero. Therefore, we can say that a global minimum can be found in a compact set containing every limit point of any sub-sequence of  $x$  when the following conditions hold

$$(1) \quad \lim_{k \rightarrow \infty} r_k \sum_1 g_1(x)^{-1} = 0, \quad i = 1, \dots, m, \quad (2.54)$$

$$(2) \quad \lim_{k \rightarrow \infty} r_k^{-\frac{1}{2}} \sum_j g_j^2(x) = 0, \quad j = m + 1, \dots, M, \quad (2.55)$$

$$(3) \quad \lim_{k \rightarrow \infty} \text{prob}(x, r_k) = V^*, \quad (2.56)$$

where  $k$  is the iteration number and  $V^*$  is the optimal.

Clearly, the computation of the SUMT method is easier than the direct application of the Lagrange multiplier method. Other motivations for using this transformation is that constraints satisfied at any iteration can be dropped, thereby reducing the size of the model. The theoretical development of the technique is given in (28 and 29) for "well-behaved" convex problems. For non-convex problems, Strong (64) has shown that the existence of a global minima of the auxiliary function converges to the global solution to the programming problem.

The use of SUMT as a computational procedure for the nonlinear programming problem has been documented by Bracken and McCormick (6). The use of the technique in risk analysis is demonstrated by Portillo-Campbell (58). Other applications and information are described in (75).

Geometric programming is another mathematical optimization procedure for dealing with nonlinear functions. The theory establishes the existence theorems characterizing optimal solutions and the framework for computational algorithms. An important feature of geometric programming is that it seeks optimal solutions without knowing the corresponding policy variables. Instead of seeking the optimal values of the independent variables first, it finds the optimal distribution of the total (cost) among the terms of the objective function. The optimal distribution of cost can be formulated in a constrained minimization problem referred to as the primal. The duality theorem developed by Duffin, Peterson, and Zener (23) relates the primal to a computationally attractive maximization problem called the dual. Within this context, we can discuss the use of geometric programming to solve nonlinear programming problems that arise in capital investment planning.

Geometric programming derives its name from the geometric inequality which states that the arithmetic mean is at least as great as the geometric mean. The most important feature of this concept is the orthogonality of its vectors. To illustrate the concept, let

$$(y_1 - y_2)^2 \geq 0, \quad (2.57)$$

thus

$$(y_1^2 - 2y_1y_2 + y_2^2) \geq 0, \quad (2.58)$$

adding  $4y_1y_2$  to both sides

$$y_1^2 + 2y_1y_2 + y_2^2 \geq 4y_1y_2, \quad (2.59)$$

taking the square root and dividing by 2 results in

$$\frac{1}{2}y_1 + \frac{1}{2}y_2 \geq y_1^{\frac{1}{2}}y_2^{\frac{1}{2}}. \quad (2.60)$$

The problem is now expressed in the geometric inequality form.

A more general expression is

$$\sum_i \xi_i y_i \geq \pi_i y_i^{\xi_i}, \quad (2.61)$$

where

$$y_i, \xi_i \geq 0 \text{ and } \sum_i \xi_i = 1, \quad (2.62)$$

rearranging the variables by letting  $y_i = \xi_i y_i$

$$\sum y_i \geq \pi_i \left(\frac{y_i}{\xi_i}\right)^{\xi_i}, \quad (2.63)$$

now let

$$g(y) = \sum_i v_i \text{ where } y_i = c_i (t_1^{a_{i1}} \cdot t_2^{a_{i2}} \cdot \dots \cdot t_n^{a_{in}}). \quad (2.64)$$

By finding a minimum of  $g(y)$ , the set of optimal weights ( $\xi_i$ ) will be found that satisfies the geometric inequality. Therefore, the primal program can be expressed as

$$\min_t g_0(t) = \sum_i c_i \pi_j t_j^{a_{ij}}, \quad \begin{array}{l} i = 1, \dots, n_0, \\ j = 1, \dots, n, \end{array} \quad (2.65)$$

subject to

$$g_k(t) = \sum_1 c_i \pi_j t_j^{a_{ij}} \leq 1, \quad k = 1, \dots, p, \quad (2.66)$$

$$i = m_k, \dots, n_k,$$

where  $t_j > 0$ , for  $i = 1, \dots, n$ , which forces  $g_k(t) \leq 1$ , for  $k = 0, 1, \dots, p$ ,

where

$$m_0 = 1, \quad m_k = n_{k-1} + 1, \quad \text{and } k = 1, \dots, p. \quad (2.67)$$

The exponents  $a_{ij}$  are arbitrary real numbers but the  $c_j$  coefficients are required to be positive, thus requiring  $g_k(t)$  to be a positive polynomial termed posynomials. The dual program is formulated from the right-hand side of the geometric inequality in which the weighing function  $\xi_i$  is found that yields an optimal solution to the problem. The general form can be expressed as

$$\max v(\xi) = \left( \prod_1 \left( \frac{c_i}{v_i} \right)^{\xi_i} \right) \prod_k \lambda_k(\xi)^{\lambda_k(\xi)}, \quad (2.68)$$

subject to

$$\sum_1 \xi_i = 1, \quad i = 1, \dots, n_0, \quad (2.69)$$

$$\sum a_{ij} \xi_i = 0, \quad i = 1, \dots, n_p, \quad (2.70)$$

$$\lambda_k(\xi) = \sum_1 \xi_i, \quad i = m_k, \dots, n_k, \quad (2.71)$$

where  $a_{ij}$ ,  $c_j$ ,  $m_k$ , and  $n_k$  are the same as for the primal program. The first constraint is the normalizing condition, while the second is the orthogonality condition.

While the concept is challenging, some computational limitations restricted the early application of geometric programming. For example, the function  $g_k(t)$  in the primal program is generally non-convex for which no existing computational procedure existed at that time. Secondly, the limitation of only using posynomials restricted its general application. Some of the extensions of geometric programming to overcome these difficulties and produce an efficient computational procedure for handling general nonlinear programming problems will be briefly described.

Geometric programming requires positive coefficients since they are raised to a fractional power in the geometric inequality form; thus, negative numbers are not allowed. This restriction was relaxed by Passy and Wilde (57) in their development of a quasi duality theory for geometric programming called generalized polynomial programming. Passy introduced a signum function to the polynomial term, such that every term yields a program very similar to a geometric program. The general form of this program is

$$\min g_0(x), \quad (2.72)$$

subject to

$$g_m(x) \leq \sigma_m (\equiv \pm 1), \quad m = 1, \dots, M, \quad (2.73)$$

where

$$g_m(x) = \sum_t \sigma_{mt} c_{mt} \prod_n x_n^{a_{mnt}}, \quad \begin{array}{l} t = 1, \dots, T_m, \\ n = 1, \dots, N. \end{array} \quad (2.74)$$



In a similar manner, the dual program can be constructed. While this dual program is not of a constrained maximization form, it does have linear constraints which have made it computationally attractive. Blau (5), using the theoretical results of Passy (56), developed an algorithm to solve generalized polynomial programs. Blau's Lagrangian formulation made certain assumptions: (1) the constrained signum function  $\sigma_i$  and the sign of the objective function are known, (2) the primal Lagrangian function is of the form

$$L(x, \lambda) = g_0(x) + \sum_m \lambda_m (g_m(x) - \sigma_m), \quad m = 1, \dots, M, \quad (2.75)$$

and (3) at the local minima, the optimal values of the Lagrange multipliers  $\lambda_m$  are strictly positive. This means that all constraints are tight or active at the optimal point. This condition restricted the use of geometric programming less than the original case but left something to be desired.

One aspect of Blau's algorithm was his use of the separability of the linear-logarithmic system that gives a solution to the dual vector  $\xi$  from a given vector of Lagrange multipliers. Based on the linearization, Duffin (22) has shown that a geometric program can be defined as a set of linear programs. Computationally, this meant that geometric programming has the potential of becoming as efficient a nonlinear algorithm as linear programming has become for linear systems. Oleson (54) extended Duffin's analytical use of the linearization principle to develop an algorithm that uses the efficiency of simplex linear programming to solve geometric programming

problems. The procedure is based on parametrically changing the objective function of a linearized geometric program. This method, based on the theory of condensation of polynomials (22), consists of three phases. The first converts the geometric program into a log-linear program where all terms are expressed as polynomials and all constraints are converted to monomial inequalities by means of a weighing function. Also, the conversion is made so that all constraints are tight or active at the optimal. The second phase solves the log-linear program (LPA) to find a feasible solution that allows a geometric program to be consistent. This step locates the region in which the optimal may be found. Finally, another linear program (LPB) is formulated consisting of the weights required to convert the constraints to monomials. The solution of this linear program parametrically changes the weights of the optimal variable until the objective function of the first linear program (LPA) is found, such that all geometric program constraints are tight.

Oleson states that the advantages of this procedure are: (1) the degree of difficulty or the size of the problem is not increased and (2) the procedure utilizes the simplex linear programming routine in a parametric fashion. He also points out that its limitation lies in the lack of proof of global optimality which is yet to be developed.

According to Wilde (73), geometric programming has great potential; most of it is as yet unrealized. However, in its

brief history, the method has been successfully employed in several engineering design problems (15, 23, 44, and 56). Economic application to resource allocation can be found in (21, 23, 33, and 58). Other economic applications, such as economic growth models, is treated in (58). Avriel and Wilde (1) applied geometric programming to the nonlinear problems found in stochastic programming.

In the above discussion, it has been shown how the traditional cash flow can be modified by the introduction of a set of decisions that either permit or prevent a cash flow to be realized. Also, the traditional mutually exclusive cash flow estimate was extended to include the complementary or competitive effects of other investments. Once the mean and variance of the cash flow estimate is determined, the next problem is to find the probability distribution of the risk associated with the investment. The treatment of risk or uncertainty in the estimate of cash flows was illustrated in the three methods of stochastic programming. Specifically, the chance-constraint programming method (CCP) assumes the probability distribution of the random variable to be normally distributed and converts the stochastic model into its deterministic equivalent. However, the deterministic form also introduces nonlinearity into the model. Generally, nonlinear problems have presented computational difficulties. However, recent developments in this area allow for efficient solutions to

large-scale complex problems such as are found in investment planning.

Now let us examine the above method of handling uncertainty or risk in investment models from the standpoint of the distributional assumptions. Hillier presents a model based on the present worth of a discounted cash flow. Since it is known that the net cash flow at any point in time may be positive or negative, the assumption was made that the net cash flows may be assumed to be normally distributed. Hillier also illustrates how the individual cash flows may be non-normal, but the present worth may be approximated as a normal distribution based on the central limit theorem. The idea of a normally distributed net cash flow has been used in several capital budgeting and portfolio models. This normality assumption is also consistent with the present assumption of chance-constraint programming in regard to the distribution of the random variables. Consequently, it has been used frequently for the analysis of risk in investment planning.

At this point, two basic problems exist in regard to the net cash flow concept and its normality assumption. The net cash flow at the end of any period is the difference between the investment cost during that period (outflow) and the receipts or yield from the investment received during the period (inflow). It would be logical to assume that the net cash flow would be normally distributed if the two components were distributed normally. Unfortunately, this is not the

case. If we assume investment costs and receipts are random variables, the range over which they could vary would be restricted to the nonnegative domain. Since both investments and receipts constitute tangible cash flows, they must be expressed in positive terms. Therefore, it would be desirable to have a distribution that is wholly contained in the nonnegative domain yet retains as many of the features of the normal distribution as possible.

Sengupta (59B) has discussed the use of nonnegative distributions in conjunction with stochastic programming. In particular, his discussion of the use of the chi-square distribution in chance-constraint programming has application in modeling capital investment plans. Two aspects of this approach are of particular interest. First, the restriction of the linear decision rule in chance-constraint programming may be replaced with more general functional forms that considerably enhance the scope of application in dynamic models that result in nonlinear objective functions. Secondly, it is no longer necessary to assume that the decision maker's utility function is quadratic or of a specific form as was required in the Markowitz (46) study.

In the following sections, the concept of a chi-square distribution for the random variables in certain investment models will be explored in the context of chance-constrained programming. Two investment models will be developed. A portfolio model, originally developed by Naslund (52), will be

formulated under the chi-square assumption and the results compared to those of Portillo-Campbell (58) who obtained a direct solution to the model. Next, a capital budgeting model will be developed employing the payback constraint along the line of Byrne (7) and Weingartner (72). This model will also employ the chi-square assumption in a chance-constraint program.

## CAPITAL INVESTMENT PLANNING MODELS

Planning of capital investments is concerned not only with the facts that risk and interdependence of investment proposals exist and should be accounted for in the decision process, but planning must also be aware of the way in which the model handles these factors. The example of finding the optimal cattle feed mix illustrated how the decision maker could be misled by following the advice of a model that neglected the effects of uncertainty. It is also possible to mislead the decision maker with recommendations that consider the effects of uncertainty in an inappropriate manner.

There exists a wide class of problems in engineering and economics where the input coefficients and the resources available are random variables, but are nonnegative. This characteristic calls for a class of probability distributions that are wholly contained in the nonnegative range. The normality assumption of chance-constraint programming would not be appropriate in cases such as these. Consequently, if the chance-constraint method is to be used for risk analysis, then the normality assumption must be replaced with a nonnegative distribution.

A nonnegative distribution to replace the normal must be selected in such a manner as to retain as many of the desirable characteristics of the normal distribution, while satisfying the nonnegativity condition. Several distributions fall into

this class with varying degrees of attractiveness. Sengupta (59A) has discussed the potential use of several nonnegative distributions in conjunction with stochastic programming. The choice of the chi-square distribution to replace the normality assumption is a logical selection since the chi-square is in fact a squared standard normal and retains the reproductive properties similar to those of the normal distribution. Secondly, other nonnegative distributions, such as the exponential, gamma, and poisson can be closely approximated by the chi-square distribution. However, the replacement of the normal by the chi-square makes the chance-constraint program computationally more difficult. This problem, however, can be transformed into a generalized polynomial programming problem that can be computed very efficiently (54 and 59A).

The choice of the chi-square distribution to be used in the formulation of chance-constrained investment models was motivated by two reasons. First, the computational difficulties are partially offset by the availability of numerical tables for the central and non-central chi-square distributions; therefore, the extension of risk analysis to consideration of various confidence intervals is facilitated. Secondly, the reproductive properties mentioned earlier are extremely useful when examining a series of cash flows that are independent random variables, i.e., the property of a variate by which the sum of a number of variates having a



fixed distribution reproduces the same distribution in form. To illustrate the use of chi-square in chance-constrained programming, consider the model developed by Sengupta in (59A and 59B).

$$\max Z = \sum_j c_j x_j, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (3.0)$$

subject to

$$\text{prob} (\sum_j a_{ij} x_j \leq b_i) \geq \alpha_i, \quad (3.1)$$

$$x_j \geq 0, \quad (3.2)$$

where the parameter  $a_{ij}$ ,  $x_j$ , and  $c_j$  are defined as before. Assume the resource vector  $b_i$  is composed of mutually independent random variables distributed chi-square. The degree by which each constraint must hold is preassigned by the decision maker (i.e., the condition must hold 95% of the time, thus  $\alpha = .95$ ).

First, consider each  $b_i$  in the resource vector  $b$  to be distributed chi-square with known or estimable mean  $E(b_i) = \bar{b}_i$  and variance  $V(b_i) = 2\bar{b}_i$ , thus, the nonnegative frequency function of  $(b_i)$  can be expressed as

$$f(b_i) = \frac{(\bar{b}_i/2)^{\bar{b}_i/2}}{\Gamma(\bar{b}_i/2)} e^{-b_i/2} b_i^{\bar{b}_i/2 - 1} \exp(-b_i/2) dt. \quad (3.3)$$

Since we know that the available resources must be greater than or equal to their allocation, then

$$\text{prob} (X^2(\bar{b}_i) \geq \sum_j a_{ij} x_j) = 1, \quad j = 1, \dots, J. \quad (3.4)$$

Now the only remaining problem is to find the upper bound on  $f(b_1)$  that satisfies the confidence limits  $\alpha_1$ . Recalling that the upper limit on the integral of  $f(b_1)$  meets this condition, we can define (3.4) as

$$\text{prob } (X^2(\bar{b}_1) \leq w) = \alpha_1, \quad (3.5)$$

where  $w$  is the upper bound of  $b_1$  that satisfies the confidence limits  $\alpha_1$ .

To illustrate, let the mean value of  $b_1$  be 10 and the confidence limit be  $\alpha_1 = 0.99$ . Therefore

$$\text{prob } (X^2(10) \leq w) = 0.99, \quad (3.6)$$

using the chi-square tables for the confidence interval of 0.99 and degrees of freedom 10, we find  $w$  to be equal to 2.56. Thus, the deterministic equivalent linear program of the chance-constraint model is

$$\max Z = \sum_j c_j x_j, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (3.7)$$

subject to

$$\sum_j a_{1j} x_j \leq 2.56, \quad (3.8)$$

$$x \geq 0. \quad (3.9)$$

By comparison, assume the random variable  $b_1$  is normally distributed with the same mean and variance

$$\text{mean: } E(b_1) = \bar{b}_1 = 10,$$

$$\text{variance: } V(b_1) = 2(\bar{b}_1) = 20.$$

Recalling that from the cumulative standard normal  $F(w)$ , we found that

$$F^{-1}(1 - \alpha) = F^{-1}(1 - 0.99) = -2.33, \quad (3.10)$$

and the deterministic equivalent of  $b_1$  would be

$$\bar{b}_1 + V(b_1)^{\frac{1}{2}} \cdot F^{-1}(0.01) = 10 + (20)^{\frac{1}{2}} \cdot -2.33 = -0.415, \quad (3.11)$$

thus, the constraint equation corresponding to (3.8) is

$$\sum_j a_{ij} \leq -0.415. \quad (3.12)$$

The optimal solution for this problem is  $x = 0$  where the solution to the chi-square formulation could obviously be better. This simple comparison illustrates the impact of the normality assumption on models that require a nonnegative distribution.

In the next two sections, capital investment models will be discussed in which the chi-square assumption will be used in the context of chance-constraint programming.

### Portfolio Expansion Model

In this section, we will develop a portfolio expansion model along the lines of Naslund (51). The model will be concerned with the optimal timing or planning of investments in order to maximize the total expected value of the portfolio at some future horizon point (planning interval). This model differs from the more traditional portfolio selection model that is static in nature and is concerned with determining the optimal proportion of the portfolio that should be invested in various types of securities. The portfolio expansion model is dynamic in nature and can generally be described as the problem

of determining the optimal amount to invest in the portfolio at each point over the planning interval. The model is constrained by the amount of risk the investor is willing to take, expressed as a loss constraint, and the availability of capital. The interrelated risky nature of the portfolio expansion model is well suited to chance-constraint programming employing the zero order rule for investment planning. Before describing the model, we will discuss briefly some of the research on portfolio investment.

There exist several discussions in the literature of the choice between holding risky assets, such as in a portfolio versus holding money. Tobin (68), for example, makes the assumption that the investor will venture some proportion of his investment dollar in risky assets. That proportion is subject to many things, such as risk, taxes, interest rates, and etc. Tobin's model develops an indifference map between the proportion of the investor's venture capital held in cash versus in his portfolio. The indifference curves are based on the mean and variance of the return on investments. Tobin suggests that, by the use of such an indifference map, it would be possible to study the effects of changes in interest rates, taxes, and risk level on the proportion of capital allocated to risky investments. The risk involved in portfolio selections is derived from the stock market prices over time. These prices are only known probabilistically.

The stock market is generally assumed to be a perfectly competitive market, in that, if certain stocks appeared to be too low at some point in time, investors would start buying causing the stock price to increase. In general, situations of this nature are caused by the availability of information and there is no reason to assume that information is made available to all investors in a systematic way. If the changes in price, due to the availability of information, are assumed to be independent and identically distributed random variables with finite mean and variance, the central limit theorem would suggest that the price change over time may be normally distributed.

Various modifications of the normality assumption have been suggested to find decision rules for the investor. Naslund based his dynamic portfolio model on the normality assumption developed by Osborne (55). However, a strong criticism of the normality assumption was put forth by Fama (25) based on empirical data. He found that the empirical distribution had a larger area under the extreme end of the tails than the normal distribution. From this, he postulated that investors cannot respond fast enough to take advantage of every price change; therefore, he will always pay a little more and obtain a little less than an optimal. The empirical distribution put forth by Fama is called the stable Paretian distribution (25).

If the intent of Fama's argument in respect to the selection of a probability distribution to represent the random variables (stock prices) is examined, a strong case for the chi-square distribution can be put forth. In addition to the nonnegative characteristics mentioned previously, the ability to approximate other distributions with the chi-square would enhance its operational desirability.

The portfolio expansion model developed in this section will assume the random variables are distributed as independent chi-square. The risk associated with the portfolio decisions will be encompassed in a chance-constraint program. The resulting nonlinear programming problem will be solved by means of generalized polynomial programming (geometric programming).

Consider a rational investor who wishes to maximize his expected gain in the stock market that will increase the value of his portfolio at the end of a specified horizon. Assume the investor has allocated funds for his consumption needs up to the horizon and knows what funds will be available for investing at each period. However, the decision to invest the available funds in risky stock at some period or to hold the funds in cash for later investment is dependent upon two constraints. First, the loss constraint which sets a probabilistic limit on the possible losses beyond a specified amount; and, secondly, the capital constraint which also specifies a probabilistic limit that investments at some point should not exceed

the available funds which varies according to accumulated capital gains.

The problem can be expressed in the following manner

$$\max Z = E(\sum_i x_i (\frac{P_i - P_{i-1}}{P_{i-1}})), \quad (3.13)$$

subject to

$$\text{prob} (x_i (\frac{P_i - P_{i-1}}{P_{i-1}}) \geq -L_i) \geq \alpha_i, \quad (\text{loss constraint}) \quad (3.14)$$

$$\text{prob} (x_i \leq k_i + \sum_j x_{j-1} \frac{P_{j-1} - P_{j-2}}{P_{j-2}}) \geq n_i, \quad (3.15)$$

(capital constraint)

$$x_j \geq 0, \quad (3.16)$$

where

$x_i$  is the accumulated amount (\$) invested in stock or stock group in period  $i$ ,

$P_i$  is the stock price or group of stocks priced in period  $i$ ,

$L_i$  is the maximum loss that the investor is willing to accept  $100(\alpha)\%$  of the time,

$\alpha_i$  is the risk level for losses at period  $i$ , set by the investor,

$k_i$  is the capital accumulation other than from returns from earlier investments,

$n_i$  is the risk level for the capital constraint in period  $i$ .

For simplicity, let  $a_i = \frac{P_i - P_{i-1}}{P_{i-1}}$  for the change in stock

prices over the interval  $i-1$  to  $i$ . Also assume that the change in stock prices observed in previous periods

$(\frac{P_{j-1} - P_{j-2}}{P_{j-2}})$  also be denoted as  $a_j$  as ( $j$  goes to  $i$ ).

Assume that the change in stock price is a random variable distributed as an independent chi-square with mean  $(\bar{a}_i)$  and denoted by  $X^2(\bar{a}_i)$ .

The loss constraint (3.14) is specified by the investor to be the minimum value he will allow his portfolio to assume at some point in time. Thus, we can express the value of the portfolio of the  $i^{\text{th}}$  period as the sum of the value gained  $a_i x_i$  during the interval plus the min value set by the investor  $L_i$  or

$$a_i x_i - L_i \geq 0. \quad (3.17)$$

By adding  $L_i$  to both sides, the loss constraint is of the form defined in (3.14). Likewise, the capital constraint can be thought of as limiting the stock buying to be within the funds available. Here too, the investor has set the amount of money  $k_i$  he is willing to invest in the portfolio at some time  $i$ . The total funds available for investing is the sum of the money received from previous stock trading and the new allocation of funds from the investor at time  $i$ .

$$k_i + \sum_j a_j x_j, \quad (3.18)$$

where  $j$  denotes previous activity in the portfolio. By using



(3.25) as the resource vector, then we have the constraint defined in (3.14).

To simplify computation, we will consider the same three-year planning horizon as did Naslund, as well as the data, in order to compare the solutions. Next, we can expand the model for the three-period planning horizons

$$\max Z = E(a_1x_1 + a_2x_2 + a_3x_3), \quad (3.19)$$

subject to the probabilistic loss constraint for each period

$$\text{prob}(a_1x_1 \geq L_1) \geq \alpha_1, \quad (3.20)$$

$$\text{prob}(a_2x_2 \geq L_2) \geq \alpha_2, \quad (3.21)$$

$$\text{prob}(a_3x_3 \geq L_3) \geq \alpha_3, \quad (3.22)$$

also subject to the capital constraints that are also probabilistic in the second and third periods. The initial cash endowment in the first period is not subject to uncertainty.

$$x_1 \leq k_1, \quad (3.23)$$

$$\text{prob}(x_2 \leq k_2 + a_1x_1) \geq n_2, \quad (3.24)$$

$$\text{prob}(x_3 \leq k_3 + a_1x_1 + a_2x_2) \geq n_3, \quad (3.25)$$

$$x_1x_2x_3 \geq 0. \quad (3.26)$$

Table 1 contains the initial values assigned by Naslund (51).

The chance-constrained programming model with chi-square variates was first developed by Sengupta in (59A and 59B). This development will be used in the formulation of the portfolio model.

Table 1. Initial values of Naslund's portfolio model

---

	Values	Confidence level
<b>Loss limitation:</b>		
period 1	$L_1 = -1300$	$\alpha_1 = 0.95$
period 2	$L_2 = -1000$	$\alpha_2 = 0.95$
period 3	$L_3 = -1000$	$\alpha_3 = 0.95$
<b>Capital limitation:</b>		
period i	$k_1 = 7000$	
period 2	$k_2 = 5500$	$n_2 = 0.99$
period 3	$k_3 = 9000$	$n_3 = 0.99$
<b>Change in stock prices:</b>		
mean value	$\bar{a}_1 = 0.05$	
	$\bar{a}_2 = 0.05$	
	$\bar{a}_3 = 0.05$	

---

Now consider the distribution of the quantity

$$Y_1 = \sum_j a_{1j} x_j, \quad j = 1, \dots, J. \quad (3.27)$$

If the mean value of each  $a_{1j}$  can be approximated by an even integer  $2g_{1j} = \bar{a}_{1j}$ , then the exact distribution can be defined as

$$\text{prob}(Y_1 > y_0) = \sum_j \sum_s d_{js} \text{prob}(X^2(2s) > \frac{y_0}{x_j}), \quad (3.28)$$

where  $j = 1, \dots, J$ ,  $s = 1, \dots, g_{1j}$ , and  $d_{js}$  is a constant involving only the  $x_j$ 's defined as

$$d_{js} = \frac{f_j(0)^{(g_{1j} - s)}}{(g_{1j} - s)!}, \quad (3.29)$$

where

$$f_j(0) = \pi \left( \frac{x_j - x_1}{x_j} + \frac{x_1 y}{x_j} - \bar{a}_{1j} \right)^2 / 2. \quad (3.30)$$

Fortunately, simpler approximations of the distribution of  $Y_1$  are available (59) which can be employed in most cases. For example, let  $Y_1$  be approximated by

$$Y_1 \approx k_1 X^2(h), \quad (3.31)$$

where the degree of freedom of  $Y_1$  is

$$h = \frac{(\sum_j a_{1j} x_j)^2}{(\sum_j \bar{a}_{1j} x_j^2)}, \quad (3.32)$$

and the weighing function of noncentrality of  $Y_1$  is

$$k_1 = \frac{\sum_j \bar{a}_{1j} x_j^2}{\sum_j \bar{a}_{1j} x_j}. \quad (3.33)$$

We can now express the general chance-constraint equation as

$$\text{prob } (k_1 X^2(h) \leq b_1) \geq \alpha_1. \quad (3.34)$$

An approximation of (3.34) often referred to in mathematical statistics (59) where equality models must be utilized on the basis of the inequality

$$\frac{(\sum_j \bar{a}_{1j} x_j)^2}{\sum_j \bar{a}_{1j} x_j^2} \leq \sum_j \bar{a}_{1j}, \quad j = 1, \dots, J. \quad (3.35)$$

By using the upper bound of (3.35) for the approximate distribution of  $Y_1$  can be expressed as

$$\left( \frac{\sum_j \bar{a}_{1j} x_j^2}{\sum_j \bar{a}_{1j} x_j} \right) X^2(\sum_j \bar{a}_{1j}). \quad (3.36)$$

Now we can redefine (3.36) in terms of the approximation of  $Y_1$  as

$$\text{prob } (X^2(\sum_j \bar{a}_{1j}) \leq \frac{b_1 (\sum_j \bar{a}_{1j} x_j)}{\sum_j \bar{a}_{1j} x_j^2}) \geq \alpha_1. \quad (3.37)$$

The cumulative distribution of (3.37) is of a central chi-square variate with degrees of freedom  $h = \sum_j \bar{a}_{1j}$

$$F(w) = (2^{h/2} \Gamma(J/2)^{-1} \int_0^w t^{(J/2)-1} \cdot \exp(-t/2) dt, \quad (3.38)$$

where

$$w = \left( \frac{b_1 \sum_j \bar{a}_{1j}}{\sum_j \bar{a}_{1j} x_j^2} \right). \quad (3.39)$$

By using the chi-square tables for various combinations of  $w$  and  $h$ , the value of  $w$  can be readily found. For example,

if  $\alpha_1 = .99$  and  $h = 2.0$ , the chance-constraint equation would be

$$\text{prob } (X^2(2.0) \leq w) = .99, \quad (3.40)$$

the value of  $w$  can be easily found to be 9.21. Thus, for any combination of the tolerance measure  $\alpha_1$  and the degrees of freedom  $h$ , the upper bound of the cumulative distribution (3.40) can be found. Therefore, the chance-constraint equation can be expressed in its deterministic equivalent

$$b_1 \sum_j \bar{a}_{1j} x_j - w_1 \sum_j \bar{a}_{1j} x_j^2 \geq 0. \quad (3.41)$$

Since the above is a concave function of the vector  $x$  for all  $w_1 > 0$ , the final deterministic model for the chance-constraint programming can be expressed as a convex programming problem

$$\min w = -\sum_j c_j x_j, \quad (3.42)$$

subject to

$$b_i \sum_j \bar{a}_{ij} x_j - w_i \sum_j \bar{a}_{ij} x_j^2 \geq 0, \quad i = 1, \dots, I. \quad (3.43)$$

$$x_j, w_i > 0, \quad j = 1, \dots, J. \quad (3.44)$$

Returning to the portfolio model, (3.17 through 3.23), we can rearrange (3.22) and (3.23) to correspond to the form in (3.37) by multiplying through by  $-a_1 x_1$  and  $-x_2$ , respectively, resulting in

$$\text{prob } (-a_1 x_1 \leq k_2 - x_2) \geq n_2. \quad (3.45)$$

Following the same procedure

$$\text{prob } (-a_1 x_1 \leq a_2 x_2 - k_2 - x_3) \geq n_3. \quad (3.46)$$

Multiplying both equations by minus one, the portfolio model (3.18 through 3.20) can be formulated as

$$\max Z = E(a_1x_1 + a_2x_2 + a_3x_3), \quad (3.47)$$

subject to

$$\text{prob}(a_1x_1 \geq L_1) \geq \alpha_1, \quad (3.48)$$

$$\text{prob}(a_2x_2 \geq L_2) \geq \alpha_2, \quad (3.49)$$

$$\text{prob}(a_3x_3 \geq L_3) \geq \alpha_3, \quad (3.50)$$

$$x_1 \leq k, \quad (3.51)$$

$$\text{prob}(a_1x_1 \geq x_2 - k_2) \geq \alpha_2, \quad (3.52)$$

$$\text{prob}(a_1x_1 + a_2x_2 \geq x_3 - k_3) \geq \alpha_3. \quad (3.53)$$

To transform the portfolio model to the convex programming form, we employ the chi-square approximation developed above. First, find the value of  $w_i$  for each probabilistic constraint as illustrated in (3.40). For the first three constraints, the sum of the mean  $h$  is .05 and the

$$\text{prob}(X^2(.05) \leq w_i) = .95, \text{ for } w_1, w_2, \text{ and } w_3. \quad (3.54)$$

Using the tables for a control chi-square variate, we find  $w_1 = 0.192$ . The last two constraints can be handled in a similar manner. For (3.51)  $h_5 = .05$  and the tolerance measure  $\alpha_5 = .999$ , thus

$$\text{prob}(X^2(.05) \leq w_5) = .999, \quad (3.55)$$

find  $w_5$  to be 0.615. The final constraint has an  $h_6 = .10$  with  $\alpha_6 = .99$ , thus

$$\text{prob } (X^2(.10) \leq w_6) = .99, \quad (3.56)$$

resulting in  $w_6 = .663$ . The values of  $w_1$  can now be employed in the above chi-square constraint (3.43).

The portfolio model (3.47 through 3.53) can now be formulated as a convex programming problem

$$\min w = -.05x_1 - .05x_2 - .05x_3, \quad (3.57)$$

subject to

$$-1300(.05)x_1 + .192(.05)x_1^2 \geq 0, \quad (3.58)$$

$$-1000(.05)x_2 + .192(.05)x_2^2 \geq 0, \quad (3.59)$$

$$-1000(.05)x_3 + .192(.05)x_3^2 \geq 0, \quad (3.60)$$

$$7000 - x_1 \geq 0, \quad (3.61)$$

$$(x_2 - 5000)(.05)x_1 - .605(.05)x_1^2 \geq 0, \quad (3.62)$$

$$(x_3 - 9000)(.05x_1 + .05x_2) - .663(.05x_1^2 + .05x_2^2) \geq 0, \quad (3.63)$$

$$x_i \geq 0. \quad (3.64)$$

Clearing terms in the model, we have

$$\min w = -.05x_1 - .05x_2 - .05x_3, \quad (3.65)$$

subject to

$$-65x_1 + .01x_1^2 \geq 0, \quad (3.66)$$

$$-50x_2 + .01x_2^2 \geq 0, \quad (3.67)$$

$$-50x_3 + .01x_3^2 \geq 0, \quad (3.68)$$

$$7000 - x_1 \geq 0, \quad (3.69)$$

$$250x_1 - .05x_1x_2 - .03x_1^2 \geq 0, \quad (3.70)$$

$$450x_1 + 450x_2 - .05x_1x_3 - .05x_2x_3 - .033x_1^2 - .033x_2^2 \geq 0, \quad (3.71)$$

$$x_1 \geq 0. \quad (3.72)$$

In the first three constraints, if we rearrange the terms by adding the left-hand term to both sides then dividing through by the same term, we have

$$x_1 \geq 6500, \quad (3.73)$$

$$x_2 \geq 5000, \quad (3.74)$$

$$x_3 \geq 5000. \quad (3.75)$$

The only deterministic constraint can be expressed in its original form

$$x_1 \leq 7000. \quad (3.76)$$

The last two constraints can be simplified in a similar manner

$$.0002x_1 - .0001x_2 \geq 1, \quad (3.77)$$

$$-x_1^{-1}x_2 + .011x_3 + .011x_1^{-1}x_2x_3 + .00007x_1 + .00007x_1^{-1}x_2^2 \geq 1. \quad (3.78)$$

The portfolio model is now in the convex programming form. Nonlinearity is introduced in the last constraint equation. The problem can now be solved by geometric programming, or more specifically, generalized polynomial programming (54). The procedure for the transformation of the model into a form



compatible with the geometric programming algorithm will be summarized at each step.

The model will be converted back to a maximization problem so that the dual will be a minimization problem consistent with the discussion earlier. Likewise, the constraints will be transformed so they will be less than or equal to unity. The first four constraints are all monomials which can be converted in a like manner. For example

$$x_1 \geq 6500, \quad (3.79)$$

can be divided by the right-hand side to give

$$\frac{x_1}{6500} \geq 1, \quad (3.80)$$

which is the same as

$$1 \leq \frac{x_1}{6500}, \quad (3.81)$$

therefore, dividing through by the right-hand side again

$$6500x_1^{-1} \leq 1. \quad (3.82)$$

The last two constraints can be transformed by dividing through by minus one. The problem can again be presented as

$$\max Z = .05x_1 + .05x_2 + .05x_3, \quad (3.83)$$

subject to

$$6500x_1^{-1} \leq 1, \quad (3.84)$$

$$5000x_2^{-1} \leq 1, \quad (3.85)$$

$$5000x_3^{-1} \leq 1, \quad (3.86)$$

$$.007x_1 \leq 1, \quad (3.87)$$

$$-.0002x_1 - .0001x_2 \leq 1, \quad (3.88)$$

$$x_1^{-1}x_2 - .011x_3 - .011x_1^{-1}x_2x_3 - .00007x_1 \\ - .00007x_1^{-1}x_2^2 \leq 1. \quad (3.89)$$

The first step in the formulation process is to express the coefficients in terms of a new variable, say  $x_4$ . The value of the variable is chosen such that its exponent is equal to the ratio of the logarithm of the coefficient to the logarithm of some normalizing constant. Computational results have indicated that the use of 1000 as the normalizing constant tends to reduce the number of iterations. For example, the variable to replace the coefficient in the first constraint could be found by

$$x_4^{(\ln 6500 / \ln 1000)}. \quad (3.90)$$

In a similar manner, the coefficients in each of the equations can be expressed in terms of  $x_4$ . The constraints can now be expressed as

$$x_4^{1.271}x_1^{-1} \leq 1, \quad (3.91)$$

$$x_4^{-1.233}x_2^{-1} \leq 1, \quad (3.92)$$

$$x_4^{1.233}x_3^{-1} \leq 1, \quad (3.93)$$

$$x_4^{-0.786}x_1 \leq 1, \quad (3.94)$$

$$x_4^{-0.647}x_1 - x_4^{-0.750}x_2 \leq 1, \quad (3.95)$$

$$x_1^{-1}x_2 - x_4^{-1.469}x_3 - x_4^{-1.469}x_2x_3 - x_4^{-0.620}x_1 - x_4^{-0.620}x_1^{-1}x_2 \leq 1. \quad (3.96)$$

This completes the normalization of the coefficients.

The next step is to convert the objective function into a constraint. This is done to facilitate parametric changes of its corresponding dual variable as indicated earlier. Again, introduce a new variable  $x_5$  such that the new objective function is

$$\min w = \frac{1}{x_5}, \quad (3.97)$$

subject to

$$x_5 \leq x_4^{-0.569}x_1 + x_4^{-0.569}x_2 + x_4^{-0.569}x_3, \quad (3.98)$$

dividing both sides by  $x_1x_2x_3$  gives

$$\frac{x_5}{x_1x_2x_3} \leq \frac{x_4^{-0.569}}{x_2x_3} + \frac{x_4^{-0.569}}{x_1x_3} + \frac{x_4^{-0.569}}{x_1x_2}, \quad (3.99)$$

select another variable  $x_6$  such that

$$\frac{x_5}{x_1x_2x_3} \leq \frac{x_4^{-0.569}}{x_2x_3} + \frac{x_4^{-0.569}}{x_1x_3} + \frac{x_4^{-0.569}}{x_1x_2} \leq x_6. \quad (3.100)$$

The above inequality is used to form

$$\frac{x_5}{x_1x_2x_3x_6} \leq 1 \text{ and } \frac{x_4^{-0.569}}{x_2x_3x_6} + \frac{x_4^{-0.569}}{x_1x_3x_6} + \frac{x_4^{-0.569}}{x_1x_2x_6} \leq 1. \quad (3.101)$$

By introducing yet another variable  $x_7$  that will satisfy

$$\frac{x_4^{-0.569}}{x_2x_3x_6} + \frac{x_4^{-0.569}}{x_1x_3x_6} \leq \frac{x_4^{-0.569}}{x_1x_2x_6} + x_7 \leq 1, \quad (3.102)$$

we can now convert the two left terms to monomials by introducing the weighing variable  $e$

$$\frac{x_4^{-0.569}}{e_1 x_2 x_3 x_6} \leq 1 \text{ and } \frac{x_4^{-0.569}}{(1 - e_1) x_2 x_3 x_6} \leq 1, \quad (3.103)$$

where  $0 < e_1 < 1$ . We can also express the terms on the right of (3.102) as

$$\frac{x_7}{e_2} \leq 1 \text{ and } \frac{x_4^{-0.569}}{(1 - e_2) x_1 x_2 x_6} \leq 1, \quad (3.104)$$

where  $0 < e_2 < 1$ . The first expression in (3.101) is amended by the weighing variable  $e_3$

$$\frac{x_5}{e_3 x_1 x_2 x_3 x_6} \leq 1, \quad (3.105)$$

where  $0 < e_3 < 1$ . The tightness constraint is added for the variable  $x_6$

$$\frac{x_6}{e_4} \leq 1, \quad (3.106)$$

where  $0 < e_4 < 1$ . This completes the conversion of the objective function.

The third step is to convert the constraint equations into monomials that can later be transformed into a log-linear program. The writer is grateful that the first four equations are already monomials requiring only the addition of the weighing variables

$$e_5^{-1} x_4^{1.271} x_1^{-1} \leq 1, \quad (3.107)$$

$$e_6^{-1} x_4^{1.233} x_2^{-1} \leq 1, \quad (3.108)$$

$$e_7^{-1} x_4^{1.233} x_3^{-1} \leq 1, \quad (3.109)$$

$$e_8^{-1} x_4^{-0.786} x_1 \leq 1. \quad (3.110)$$

The fifth equation can be treated as before

$$x_4^{-0.697} x_1 - x_4^{-0.750} x_2 \leq x_8, \quad (3.111)$$

such that

$$\frac{x_4^{-0.697} x_1}{e_9 x_8} \leq 1 \text{ and } \frac{x_4^{-0.750} x_2}{(1 - e_9) x_8} \leq 1. \quad (3.112)$$

The last constraint can be converted to a set of monomial equations by the addition of new variables, one for every pairwise decomposition. Start by adding the last two terms to both sides of constraint (3.96) and introduce  $x_9$

$$\begin{aligned} x_1^{-1} x_2 - x_4^{-1.469} x_3 - x_4^{-1.469} x_2 x_3 \leq x_4^{-0.620} x_1^{-1} \\ + x_4^{-0.620} x_1^{-1} x_2^2 \leq x_9, \end{aligned} \quad (3.113)$$

dividing through by  $x_9$  gives

$$\frac{x_1^{-1} x_2}{x_9} - \frac{x_4^{-1.469} x_3}{x_9} - \frac{x_4^{-1.469} x_2 x_3}{x_9} \leq 1, \quad (3.114)$$

and

$$\frac{x_4^{-0.620} x_1^{-1}}{x_9} + \frac{x_4^{-0.620} x_1^{-1} x_2^2}{x_9} \leq 1, \quad (3.115)$$

converting the latter, we have

$$\frac{x_4^{-0.620} x_1^{-1}}{e_{10} x_9} \leq 1 \text{ and } \frac{x_4^{-0.620} x_1^{-1} x_2}{(1 - e_{10}) x_9} \leq 1. \quad (3.116)$$

Again adding the last term in (3.114) to both sides and adding the new variable  $x_{10}$

$$\frac{x_1^{-1} x_2}{x_9} - \frac{x_4^{-1.469} x_3}{x_9} \leq \frac{x_4^{-1.469} x_2 x_3}{x_9} + x_{10} \leq 1, \quad (3.117)$$

the two sets of terms can be converted to monomials as before

$$\frac{x_1^{-1} x_2}{e_{11} x_9} \leq 1 \text{ and } \frac{x_4^{-1.469} x_3}{(1 - e_{11}) x_9} \leq 1, \quad (3.118)$$

$$\frac{x_{10}}{e_{12}} \leq 1 \text{ and } \frac{x_4^{-1.469} x_2 x_3}{(1 - e_{12}) x_9} \leq 1, \quad (3.119)$$

adding the tightness constraint for  $x_9$

$$\frac{x_9}{e_{13}} \leq 1, \quad (3.120)$$

where

$$0 < e_5 < 1 \text{ and } 0 < e_9 < \infty, \quad (3.121)$$

$$0 < e_6 < 1 \text{ and } 0 < e_{10} < \infty, \quad (3.122)$$

$$0 < e_7 < 1 \text{ and } 0 < e_{11} < \infty, \quad (3.123)$$

$$0 < e_8 < 1 \text{ and } 0 < e_{12} < \infty, \quad (3.124)$$

$$0 < e_{13} < 1. \quad (3.125)$$

The resulting geometric programming problem in monomial form is

$$\min w = \frac{1}{x_5}, \quad (3.126)$$

subject to

(normalizing constraint)

$$.001x_4 \leq 1, \quad (3.127)$$

(objective function)

$$e_1^{-1} x_2^{-1} x_3^{-1} x_6^{-1} x_4^{-0.569} \leq 1, \quad (3.128)$$

$$(1 - e_1)^{-1} x_2^{-1} x_3^{-1} x_6^{-1} x_4^{-0.565} \leq 1, \quad (3.129)$$

$$e_2^{-1} x_7 \leq 1, \quad (3.130)$$

$$(1 - e_2)^{-1} x_1^{-1} x_2^{-1} x_6^{-1} x_4^{-0.569} \leq 1, \quad (3.131)$$

$$e_3^{-1} x_1^{-1} x_2^{-1} x_5 x_6^{-1} \leq 1, \quad (3.132)$$

$$e_4^{-1} x_6 \leq 1, \quad (3.133)$$

(loss constraints)

$$e_5^{-1} x_1^{-1} x_4^{1.271} \leq 1, \quad (3.134)$$

$$e_6^{-1} x_2^{-1} x_4^{1.233} \leq 1, \quad (3.135)$$

$$e_7^{-1} x_3^{-1} x_4^{1.233} \leq 1, \quad (3.136)$$

(capital constraints)

$$e_8^{-1} x_1 x_4^{-0.786} \leq 1, \quad (3.137)$$

$$e_9^{-1} x_1 x_8^{-1} x_4^{-0.697} \leq 1, \quad (3.138)$$

$$(1 - e_9)^{-1} x_2 x_8^{-1} x_4^{-0.750} \leq 1, \quad (3.139)$$

$$e_{10}^{-1} x_1^{-1} x_9^{-1} x_4^{-0.620} \leq 1, \quad (3.140)$$

$$(1 - e_{10})^{-1} x_1^{-1} x_2^{-1} x_9^{-1} x_4^{-0.620} \leq 1, \quad (3.141)$$

$$e_{11}^{-1} x_1^{-1} x_2 x_9^{-1} \leq 1, \quad (3.142)$$

$$(1 - e_{11})^{-1} x_3 x_9^{-1} x_4^{-1.469} \leq 1, \quad (3.143)$$

$$e_{12}^{-1} x_{10} \leq 1, \quad (3.144)$$

$$(1 - e_{12})^{-1} x_2 x_3 x_9 x_4^{-1.469} \leq 1, \quad (3.145)$$

$$e_{13}^{-1} x_8 \leq 1, \quad (3.146)$$

(geometric programming weighing constraints)

$$0 < e_1 < \infty, \quad (3.147)$$

$$0 < e_3 < 1, \quad (3.148)$$

$$0 < e_4 < 1, \quad 0 < e_2 < \infty, \quad (3.149)$$

$$0 < e_5 < 1, \quad 0 < e_9 < \infty, \quad (3.150)$$

$$0 < e_6 < 1, \quad 0 < e_{10} < \infty, \quad (3.151)$$

$$0 < e_7 < 1, \quad 0 < e_{11} < \infty, \quad (3.152)$$

$$0 < e_8 < 1, \quad (3.153)$$

$$0 < e_{13} < 1, \quad 0 < e_{12} < \infty. \quad (3.154)$$

The computational dual of the above problem can be expressed as a log-linear program



$$\begin{aligned}
\max Z = & \ln .001y_1 + \ln e_1^{-1}y_2 + \ln (1 - e_1)^{-1}y_3 \\
& + \ln e_2^{-1}y_4 + \ln (1 - e_2)^{-1}y_5 + \ln e_3^{-1}y_6 \\
& + \ln e_4^{-1}y_7 + \ln e_5^{-1}y_8 + \ln e_6^{-1}y_9 + \ln e_7^{-1}y_{10} \\
& + \ln e_8^{-1}y_{11} + \ln e_9^{-1}y_{12} + \ln (1 - e_9)^{-1}y_{13} \\
& + \ln e_{10}^{-1}y_{14} + \ln (1 - e_{10})^{-1}y_{15} + \ln e_{11}^{-1}y_{16} \\
& + \ln (1 - e_{11})^{-1}y_{17} + \ln e_{12}^{-1}y_{18} \\
& + \ln (1 - e_{12})^{-1}y_{19} + \ln e_{13}^{-1}y_{20}, \quad (3.155)
\end{aligned}$$

subject to the log-linear constraints

$$y_0 = 1, \quad (3.156)$$

$$-y_5 - y_6 - y_8 + y_{11} + y_{12} - y_{14} - y_{15} - y_{16} = 0, \quad (3.157)$$

$$\begin{aligned}
-y_2 - y_3 - y_5 - y_6 - y_9 + y_{13} + y_{15} + 2y_{16} \\
+ y_{17} = 0, \quad (3.158)
\end{aligned}$$

$$-y_2 - y_3 - y_{10} + y_{17} + y_{18} = 0, \quad (3.159)$$

$$\begin{aligned}
6.907y_1 - 0.569y_2 - 0.569y_3 - 0.569y_5 + 1.271y_8 \\
+ 1.233y_9 + 1.233y_{10} - 0.786y_{11} - 0.697y_{12} - 0.750y_{13} \\
- 0.620y_{14} - 0.620y_{16} - 1.469y_{17} - 1.469y_{19} = 0, \quad (3.160)
\end{aligned}$$

$$-y_2 - y_3 - y_5 - y_6 + y_7 = 0, \quad (3.161)$$

$$-y_{12} - y_{13} + y_{20} = 0, \quad (3.162)$$

$$-y_{14} - y_{15} - y_{16} - y_{17} + y_{19} = 0, \quad (3.163)$$

$$e_1, e_9, e_{10}, e_{11}, e_{12} = .5, \quad (3.164)$$

$$e_3, e_4, e_5, e_6, e_7, e_8, e_{13} = 1. \quad (3.165)$$

The solution of the portfolio model is given below along with the solution of Portillo-Campbell (58), who used the normality assumption of chance-constraint programming and solved the problem by SUMT. Also, the original (decision rule) solution of Naslund (51), who only solved the linear portion of the model, is given in Table 2.

The nonlinear solutions for the chi-square case is consistent with that of the nonlinear normal, in respect to  $x_1$  and  $x_3$ . The value of  $x_2$  is in line with that of the linear case. The difference between the values of  $x_2$  for the two nonlinear cases is not apparent. In general, the solutions for the chi-square and the normal cases were expected to be quite close since a chi-square can approximate a normal distribution truncated at zero and the mean of the random variable was close to zero.

It should be noted that the procedure provides only an approximate solution. The geometric programming algorithm employed does not insure a global solution; thus, a stopping rule is used to determine the point of termination. The stopping rule used for this problem was set at sum of squares  $Z_j - C_j$  less than 0.001.

Table 2. Solutions to portfolio model

Decision rule linear case (norm. ass.)	Direct solution nonlinear (SUMT) (norm. ass.)	Direct solution nonlinear (G.P.) (chi-square ass.)
$x_1 = 5900$	$x_1 = 6581$	$x_1 = 6597$
$x_2 = 5500$	$x_2 = 3528$	$x_2 = 5588$
$x_3 = 5000$	$x_3 = 5024$	$x_3 = 5028$
$Z^* = 820$	$Z^* = 756$	$Z^* = 861$

Table 3. Convergence of geometric program solution

Iteration number	Computed Z max	Sum $S_j - C_j^q$
0	2.1813	4.6651
1	0.4733	3.4092
2	-1.4393	2.2435
3	-4.5842	1.2173
4	-6.6662	0.6022
5	-6.6921	0.0459
6	-6.7431	0.0017
7	-6.7581	$0.1183 \times 10^{-3}$
8	-6.7581	$0.5101 \times 10^{-5}$

Table 4. Values at the end of the eight iterations

Basic variables	Non-basic variables	Log value at $e_i$
$y_1 = 0$	$y_3 = 3.238 \times 10^{-10}$	$e_1 = -2.00248$
$y_2 = 0$	$y_5 = 6.730 \times 10^{-4}$	$e_2 = -0.74444$
$y_4 = 0$	$y_6 = 2.615 \times 10^{-9}$	$e_3 = -0.45413$
$y_6 = 0$	$y_7 = 5.002 \times 10^{-11}$	$e_4 = -0.47804$
$y_7 = 0$	$y_{13} = -6.357 \times 10^{-6}$	$e_5 = 0.08616$
$y_8 = 0$	$y_{15} = 4.011 \times 10^{-12}$	$e_7 = 0.03922$
$y_{11} = 0$	$y_{16} = 3.924 \times 10^{-8}$	$e_8 = -0.83326$
$y_{12} = 0$	$y_{19} = 5.560 \times 10^{-10}$	$e_9 = -0.15900$
$y_{14} = 0$		$e_{10} = 0.76572$
$y_{17} = 0$		$e_{11} = -0.95129$
$y_{18} = 0$		$e_{12} = -0.20702$
$y_{20} = 0$		$e_{13} = 0.24686$

In the next section, the chi-square assumption will be used in the formulation of a capital investment planning model, in which the timing of investments is dependent somewhat on the cash flows.

### Capital Budgeting Model

In this section, we will formulate a capital budgeting model to allocate capital investments over a finite planning horizon. The concept of risk will be divided into two classes. The risk associated with "opportunity loss" will be structured as a payback constraint in which the investor wants to recover his initial invested capital within a specified period in order to take advantage of new investment opportunities. The concept of payback as a constraint is discussed in (7 and 71). To protect against "actual losses" or cash flow shortages, a liquidity constraint is introduced that permits borrowing to meet the liquidity condition. To guard against misuse of borrowed funds, a penalty cost is added to the objective function. The liquidity constraint is developed along the lines of linear programming under uncertainty. The model is formulated in the framework of a chance-constrained program utilizing the zero order rule. The random variable will be assumed to be distributed as an independent chi-square. The two-stage decision rule of LPUU is incorporated within the structure of the model. Each capital investment will consist of a point-input and stream-output to generate the two cash

flows. The investment cost  $c_{ij}$  for each project  $i$  at each point  $j$  is known or estimated. The flow of funds both into and out of the firm is controlled by the decision parameter  $x$ . In general, we can think of the variable  $x_{ij}$  as the proportion of the total capital expenditure to be committed to the  $i^{\text{th}}$  group at time  $j$ . Therefore, the value of  $x_{ij}$  for each project will be one if accepted or zero if not accepted at time  $j$ . Correspondingly, the decision governing the inflows  $x_{ik}$  can be thought of in a similar manner. The returns from an investment  $x_{ij}$  can be denoted as  $r_{ij}$  when observed at the  $j^{\text{th}}$  point in time or  $r_{ik}$  when the return is measured  $k$  periods after the start of the project. This case would correspond to the  $k^{\text{th}}$  payback period or the discounting interval in traditional cash flow analysis. The symmetry of the cash flows can be seen in Figure 1.

First, we can structure the payback constraint that will allow for additional profit opportunities by increasing the available funds for reinvestment. In general, this is accomplished by minimizing the payback period thereby increasing the velocity of capital funds. To provide additional freedom of selection, we will assume that any project may be slid both forward or backward to take advantage of available capital. This assumption may raise some question but for simplicity we will assume that all projects can be moved without penalty. Now the payback constraint can be expressed as

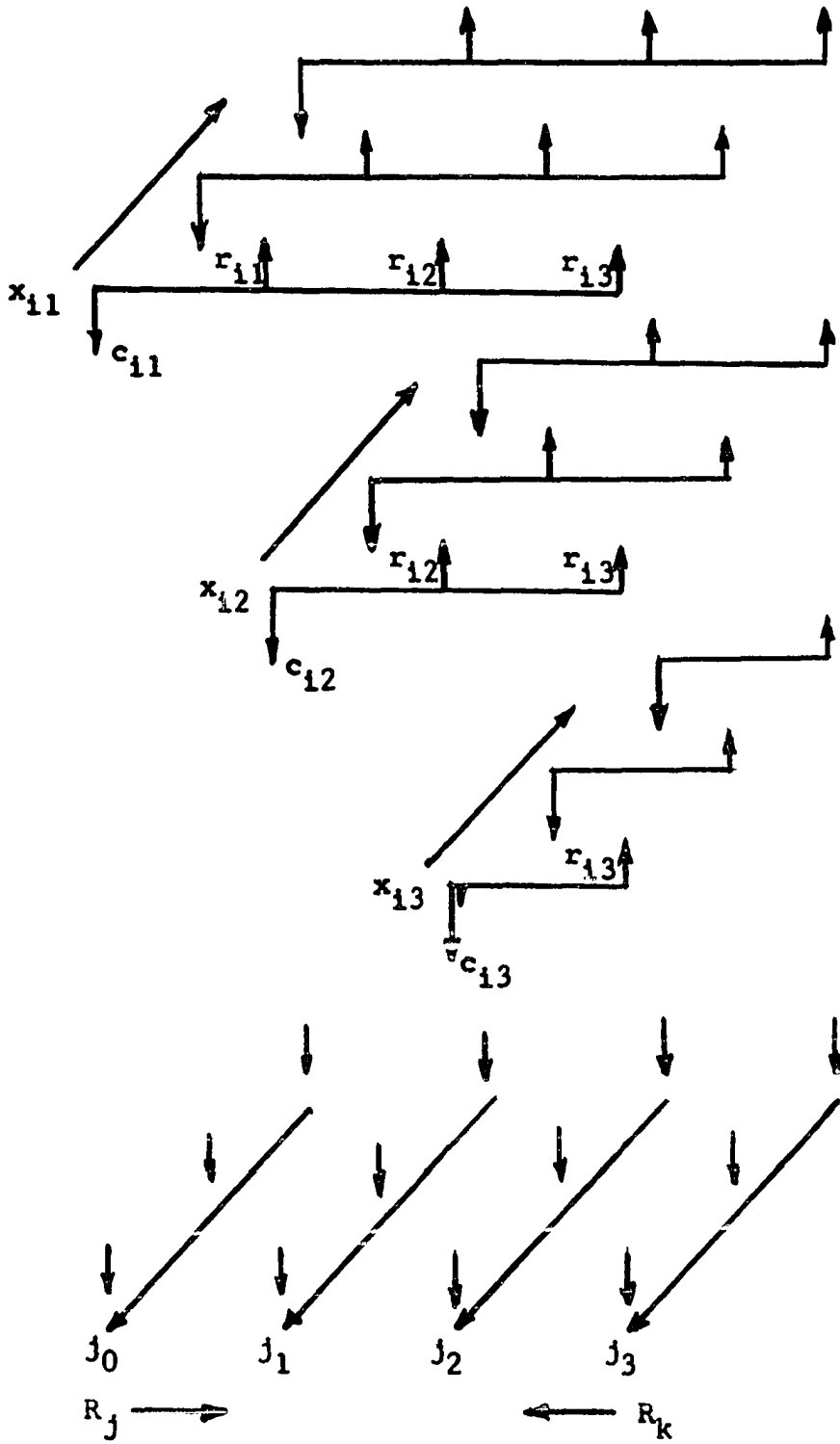


Figure 1. Cash flow diagram

$$\begin{aligned} \text{prob } (\Sigma_1 (\Sigma_k r_{ik}) x_{i1} + (\Sigma_{k'} r_{ik'}) x_{i2} + \dots + r_{i1} x_{ik}) \\ \geq \Sigma_1 \Sigma_j c_{ij} x_{ij} \geq \alpha, \end{aligned} \quad (3.166)$$

where  $k = 1, \dots, K$ ,  $k' = 1, \dots, K-1$ ,  $k = k'$ ,  $i = 1, \dots, I$ , and  $j = 1, \dots, J$ . For simplicity, we can denote the above as

$$\text{prob } (R_k \geq C_j) \geq \alpha. \quad (3.167)$$

This will set the upper bound on the investments undertaken at each period. The payback constraint serves the same purpose as the capital constraint in the portfolio model. Byrne et al. (8) has noted that the flow of returns is from aggregated sources that may or may not be interrelated but may be approximated as a series of independent random variables (normal or, in our case, chi-square).

Next we will formulate the liquidity constraint and its two-stage decision rule. The liquidity condition can be structured from a set of "balance sheet" variables. First, we can define  $M_{j-1}$  as the total available funds at the end of each period. This is equal to the starting balance of the next period. Let  $M_0$  be the initial capital available at the start of the planning period. The amount of cash on hand (or equivalent) to retain liquidity can be denoted as  $L_j$ . This is analogous to the loss constraint in the portfolio model. Since we can borrow funds to meet our liquidity level, we can denote  $W_j$  as the amount borrowed at the end of the period. We will assume excess funds will be used to pay back loans incurred at earlier periods. The cash flow of investment cost can be



denoted as  $C_j$  as seen in the payback constraint. We can first define cash flow of returns as the amount received at  $j$  from all on-going projects  $i$

$$R_j = \sum_i r_{ij} x_{ij}. \quad (3.168)$$

We can now examine the workings of the liquidity constraint by letting

$$j = 1: M_0 - \sum_i c_{i1} = M_1 \text{ if } M_1 < L_1$$

$$\text{then } W_1 = L_1 - M_1, \quad (3.169)$$

$$j = 2: M_1 - \sum_i r_{i2} x_{i2} - \sum_i c_{i2} x_{i2} = M_2, \quad (3.170)$$

$$(W_2 \text{ found as above})$$

$$j = 3: M_2 - \sum_i (r_{i2} x_{i1} + r_{i1} x_{i2}) - \sum_i c_{i3} x_{i3} = M_3. \quad (3.171)$$

Since  $M_{j-1}$  is composed of  $M_0 + R_j - C_j$  then we can rewrite the cash flow of returns as

$$R_j = \sum_j (r_{ij} x_{i1} + r_{ij-1} x_{i2} + \dots + r_{i1} x_{ij}), \quad (3.172)$$

also we can combine terms by letting

$$K_j = L_j - M_0, \quad (3.173)$$

since both  $M_0$  and all levels of  $L_j$  are known. We can now express the liquidity constraint

$$R_j - C_j - K_j + W_j \geq 0, \quad (3.174)$$

where  $R_j - C_j$  indicates the balance of the two cash flows or the remaining funds ( $\pm$ ) after committing investments at  $j$ .  $K_j$  sets the lower bound of the treasury. If the difference between  $R_j - C_j$  is less than  $K_j$ , then the firm must borrow funds to cover the difference. Since the use of borrowed

capital is not free, we will add a penalty cost to the objective function to discourage over borrowing. Thus, the objective function can be written as

$$\max Z = E(R_t - C_t - \sum_j p_j W_j), \quad (3.175)$$

where  $p_j$  is the penalty cost for borrowing funds. The terminal cash flow is denoted as  $D_t - C_t$ . The liquidity constraint is structured as a two-stage linear programming under uncertainty model. The constraint, while probabilistic, operates differently than the other chance-constrained equations.

To examine the mechanics of the model, first let us express the model in the CCP form

$$\max Z = E(R_t - C_t - \sum_j p_j W_j), \quad (3.176)$$

subject to

$$\text{prob}(R_k \geq C_j) \geq \alpha, \text{ (payback constraint)} \quad (3.177)$$

$$\text{prob}(R_j - C_j - K_j + W_j \geq 0) \geq \beta. \quad (3.178)$$

(liquidity constraint)

Notice that  $C_j$  is the same in both constraints and that  $R_k$  and  $R_j$  are random variables composed of a series of independent chi-square variates. Next, we can describe the process of the combined model. The first step is to find an estimate of  $C_j$  by letting

$$\text{prob}(X^2(R_k) \leq \bar{C}_j) = 1 - \alpha, \quad (3.179)$$

thus,  $\bar{C}_j$  is the maximum allocation of investment funds at period  $j$  that satisfies the payback constraint. Next, we can find

$$\text{prob } (X^2(R_j) \leq S_{ckw}) = 1 - \beta, \quad (3.180)$$

where

$$S_{ckw} = C_j - K_j + W_j. \quad (3.181)$$

Since  $C_j$  is estimated at  $\bar{C}_j$  and  $K_j$  is known, then  $W_j$  can be found. The only remaining step is to determine the penalty cost for borrowing that will constitute the second-stage decision rule for the liquidity condition.

A sample problem was structured that consisted of three projects and three time periods. The resulting deterministic equivalent and its geometric program follows the procedure discussed earlier and will not be stated.

A sample problem was developed using arbitrary data to illustrate the procedure. The results indicated that all but one project was selected in the first period and the remaining project in the second. Borrowed funds were needed to meet the first period liquidity constraint but none thereafter. The results, in general, are consistent with Byrne's model (7 and 13) where normally distributed random variables were used. However, the estimation and computation procedures were much simpler.

## SUMMARY AND CONCLUSIONS

The decision maker concerned with the planning of capital expenditures must consider both the timing of capital investments and the risk associated with committing funds. This investigation has put forth the idea that the analysis of risk of an interrelated investment plan can be accomplished by chance-constrained programming (CCP) using the computational procedures of geometric programming. The basic assumption for the analysis has been made that the chi-square distribution approximates the occurrence of the chance variable. The procedure was illustrated in a portfolio expansion model and the results compared to previous solutions under different assumptions and procedures. Also, a capital budgeting model was developed which incorporated the two-stage decision rule of linear programming under uncertainty. The model employed a payback constraint to handle "opportunity loss" type of risk, while a liquidity constraint is included to handle the more traditional accounting losses.

In light of the investigation just completed, the following conclusions may be stated:

- i. When the chance variable is nonnegative with a positive finite mean, the procedure developed herein will yield a more precise solution than from those methods previously available.

2. The computational problems associated with approximating a nonnegative parameter with a standard normal distribution are relieved by the use of the chi-square distribution.
3. Geometric programming procedures reduce the computational difficulties normally associated with chance-constrained programming.

Some suggestions for future research are:

1. The procedure could be expanded to include the solution of the tolerance or confidence limit directly as a function of the sample size and its related cost.
2. Application of safety-first programming as an adjunct to the procedure developed herein would be useful in capital investment planning.
3. The phasing of capital expenditures demonstrated by this procedure could be employed in a model for capacity expansion and growth of an organization.

In conclusion, the procedure and the models have illustrated the potential application of stochastic nonlinear programming as a method of analyzing some of the problems that confront the capital investment planner.

## LITERATURE CITED

1. Avriel, M. and D. Wilde. Stochastic geometric programming. Princeton Symposium on Mathematical Programming. Princeton, New Jersey, Princeton University Press. 1971.
2. Babbar, M. Distribution of solutions of a set of linear equations with application to linear programming. Journal of American Statistical Association 50:854-869. 1955.
3. Babbar, M. Statistical approach in planning production programs for interdependent activities. Unpublished Ph.D. thesis. Ames, Iowa, Iowa State University Library. 1953.
4. Baumal, W. An expected gain-confidence limit criterion for portfolio selection. Management Science 10:174-184. 1963.
5. Blau, G. Generalized polynomial programming: extensions and applications. Unpublished Ph.D. thesis. Stanford, California, Stanford University Library. 1968.
6. Bracken, J. and G. McCormick. Selected application of nonlinear programming. New York, New York, Wiley and Sons, Inc. c1968.
7. Byrne, R. Chance-constrained programming and related approaches to risk control in capital budgeting. NONR Report 760(24) Nr. 047-048. Pittsburgh, Pennsylvania, Carnegie-Mellon University Library. 1968.
8. Byrne, R., A. Charnes, W. Cooper, and K. Kortanek. A chance-constrained programming approach to capital budgeting with portfolio type payback constraints and horizon posture control. Journal of Financial and Quantitative Analysis 4:339-364. 1967.
9. Carroll, C. The created response surface technique for optimizing nonlinear restrained systems. Operations Research 9:169-184. 1961.
10. Charnes, A. and W. Cooper. Chance-constrained programs with normal deviates and linear decision rule. The Technological Institute. Evanston, Illinois, Northwestern University Library. 1959.

11. Charnes, A. and W. Cooper. Deterministic equivalent for optimizing and satisfactions under chance constraints. *Operations Research* 11:18-39. 1963.
12. Charnes, A., W. Cooper, and G. Symonds. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Management Science* 4:235-263. 1958.
13. Charnes, A., W. Cooper, and G. Thompson. Constrained generalized medians and hypermedians as deterministic equivalents for two-stage linear programming under uncertainty. *Management Science* 12:83-112. 1965.
14. Charnes, A., W. Cooper, and G. Thompson. Critical path analysis via chance-constrained and stochastic programming. *Operations Research* 12:460-470. 1964.
15. Charnes, A. and M. Kirby. Optimal decision rules for the E-model of chance-constrained programming. *Cahiers du Centre d'Etudes de Recherche Operationelle* 1:5-44. 1966.
16. Charnes, A., M. Kirby, and W. Raika. Chance-constrained generalized network. *Operations Research* 14:463-475. 1966.
17. Charnes, A. and A. Stedry. Chance-constrained model for real-time control in research and development management. *Management Science* 8:353-362. 1966.
18. Charnes, A. and S. Thore. Planning for liquidity in savings and loan associations. *Journal of Finance* 12. 1966.
19. Cheng, Pao-Lun. Optimum bond portfolio selection. *Management Science* 8:490-499. 1962.
20. Dantzig, G. Linear programming under uncertainty. *Management Science* 1:197-206. 1955.
21. Duffin, R. Cost minimization problems treated by geometric means. *Operations Research* 10:668-675. 1962.
22. Duffin, R. Linearizing geometric programs. *SIAM Review* 12:211-217. 1970.
23. Duffin, R., E. Peterson, and C. Zener. *Geometric programming: theory and application*. New York, New York, Wiley and Sons, Inc. c1967.

24. Elmaghraby, S. An approach to linear programming under uncertainty. *Operations Research* 7:208-216. 1959.
25. Fama, E. Portfolio analysis in the stable paretian market. *Management Science* 11:404-419. 1965.
26. Farrar, D. The investment decision under uncertainty. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. 1962.
27. Ferguson, A. and G. Dantzig. The allocation of aircraft to routes--an example of linear programming under uncertain demands. *Management Science* 3:45-73. 1956.
28. Fiacco, A. Sequential unconstrained minimization methods for nonlinear programming. Unpublished Ph.D. thesis. Evanston, Illinois, Northwestern University Library. 1967.
29. Fiacco, A. and G. McCormick. Nonlinear programming: sequential unconstrained minimization techniques. New York, New York, Wiley and Sons, Inc. c1968.
30. Gordon, M. The payoff period and the rate of profit. *Journal of Business* 28:253-261. 1955.
31. Hertz, D. Risk analysis in capital investment. *Harvard Business Review* 42:95-106. 1964.
32. Hespos, R. and P. Strassmann. Stochastic decision tree for the analysis of investment decisions. *Management Science* 11:B249-B259. 1965.
33. Heyman, M. and M. Avriel. On a decomposition for a special class of geometric programming problems. *Optimization Theory and Applications Journal* 3:392-409. 1969.
34. Hillier, F. Chance-constrained programming with 0 - 1 or bounded continuous decision variables. *Management Science* 14:34-57. 1963.
35. Hillier, F. Derivation of probabilistic information for the evaluation of risky investments. *Management Science* 9:443-457. 1963.
36. Hillier, F. The evaluation of risky interrelated investments. Amsterdam, Holland, North-Holland Publishing Co. c1969.
37. Hillier, F. Supplement to: Derivation of probabilistic information for the evaluation of risky investments. *Management Science* 11:485-487. 1965.



38. Hofschmidt, M. Simulating the behavior of a multi-unit multi-purpose water resource system. In A. C. Hoggatt and F. E. Balderston (eds.) Symposium on simulation models. Chapter 6. Chicago, Illinois, South-Western Publishing Co. c1963.
39. Horowitz, I. The plant investment decision revisited. Journal of Industrial Engineering 17:416-422. 1966.
40. Kirby, M. The current state of chance-constrained programming. Princeton Symposium on Mathematical Programming. Princeton, New Jersey, Princeton University Press. 1971.
41. Luce, R. and H. Raiffa. Games and decisions. New York, New York, Wiley and Sons, Inc. c1960.
42. Lutz, F. and V. Lutz. The theory of investment of the firm. Princeton, New Jersey, Princeton University Press. 1951.
43. Madansky, A. Dual variables in two-stage linear programming under uncertainty. Journal of Mathematic Analysis and Applications 6:98-108. 1963.
44. Mangasarian, O. Nonlinear programming. New York, New York, McGraw-Hill Book Co. c1969.
45. Mao, J. and C. Sarndal. A decision theory approach to portfolio selection. Management Science 12:B323-B333. 1966.
46. Markowitz, H. Portfolio selection: efficient diversification of investments. New York, New York, Wiley and Sons, Inc. 1959.
47. Masse, P. Optimal investment decisions. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. 1962.
48. Miller, B. and H. Wagner. Chance-constrained programming with joint constraints. Operations Research 13:930-945. 1965.
49. Milnor, J. Games against nature. In R. M. Thrall, C. H. Coombs, and R. L. Davis (eds.) Decision process. Chapter 4. New York, New York, Wiley and Sons, Inc. c1954.
50. Moeseke, P. Stochastic linear programming. Unpublished M.S. thesis. Ames, Iowa, Iowa State University Library. 1960.

51. Naslund, B. Mathematical programming under risk. *The Swedish Journal of Economics* 67:240-255. 1965.
52. Naslund, B. A model of capital budgeting under risk. *Journal of Business* 39:257-271. 1966.
53. Naslund, B. and A. Whinston. A model of multi-period investment under uncertainty. *Management Science* 8: 184-207. 1962.
54. Olsson, G. Computational aspects of geometric programming. Unpublished Ph.D. thesis. Ames, Iowa, Iowa State University Library. 1971.
55. Osborne, M. Brownian motion in the stock market. *Operations Research* 7:145-173. 1959.
56. Passy, U. Generalization of geometric programming: partial control of linear inventory systems. Unpublished Ph.D. thesis. Stanford, California, Stanford University Library. 1966.
57. Passy, U. and D. Wilde. Generalized polynomial optimization. *Journal of Applied Mathematics* 15:1345-1356. 1967.
58. Portillo-Campbell, J. Decision under risk programming with economic application. Unpublished Ph.D. thesis. Ames, Iowa, Iowa State University Library. c1969.
- 59A. Sengupta, J. Chance-constrained linear programming with chi-square deviates. To be published in *Management Science* ca. 1972.
- 59B. Sengupta, J. A generalization of some distribution aspects of chance-constraint linear programming. *International Economic Review* 11:287-303. 1970
60. Sengupta, J. On the sensitivity of optimal solutions under investment planning and programming. *Atharities* 8:1-23. 1965.
61. Sengupta, J. and G. Tintner. A review of stochastic programming. *Review of the International Statistical Institute* 39:197-223. 1971.
62. Sengupta, J., G. Tintner, and B. Morrison. Stochastic linear programming with application to economic models. *Economica* 30:262-270. 1963.

63. Sharpe, W. A simplified model for portfolio analysis. *Management Science* 4:277-293. 1963.
64. Strong, R. A note on sequential unconstrained minimization technique for nonlinear programming. *Management Science* 12:142-144. 1965.
65. Tintner, G. and J. Sengupta. Stochastic linear programming and its application to economic planning. *Essays in Honor of Oscar Lange*. New York, New York, Pergamon Press. 1965.
66. Tintner, G., J. Sengupta, and V. Rau. An application of stochastic linear programming to development planning. *Metroeconomica* 14:25-41. 1962.
67. Tobin, J. An aggregative dynamic model. *Journal of Political Economics* 63:103-115. 1955.
68. Tobin, J. Liquidity preference as behavior towards risk. *Review of Economics Statistics* February:18-39. 1958.
69. Wagner, H. On the distribution of solutions in linear programming problems. *Journal of the American Statistical Association* 53:161-163. 1958.
70. Wagner, H. Principles of operations research with applications to managerial decisions. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. c1969.
71. Weingartner, H. Capital budgeting of interrelated projects. *Management Science* 12:465-576. 1966.
72. Weingartner, H. Mathematical programming and the analysis of capital budgeting problems. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. c1963.
73. Wilde, D. and C. Beightler. Foundations of optimization. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. c1967.
74. Wolfe, R. and L. Cutter. Experiments in linear programming. In R. L. Graves and P. Wolfe (eds.) *Recent advances in mathematical programming*. Chapter 2. New York, New York, McGraw-Hill Book Co. c1963.
75. Zangwill, W. Nonlinear programming, a unified approach. Englewood Cliffs, New Jersey, Prentice-Hall, Inc. c1969.

76. Zener, C. Minimization of system cost in terms of sub-system cost. National Academy of Sciences Proceedings 51:162-164. 1964.

## ACKNOWLEDGMENTS

The writer would like to dedicate this thesis to those "good people of the Graduate Office", whose timely counsel has aided generations of my fellow students over the hurdles of graduate study.

A very special debt of gratitude is owed my wife, Joan, who has weathered the tides of a "husband in grad school", and, in addition to providing encouragement, has done an excellent job of editing and typing this manuscript.

Thanks are also due to my colleagues of the Department of Industrial Engineering, in particular, Dr. Keith L. McRoberts and Dr. John Even. Also, I would like to acknowledge Professor J. K. Walkup and the College of Engineering for providing the financial support that facilitated my graduate work. I am indebted to Dr. Lee Fletcher and Dr. Herbert David, whose assistance and direction have far exceeded that normally expected of a committee member.

Finally, I would like to express my deepest gratitude to Dr. Jati K. Sengupta and Dr. Arthur C. Kleinschmidt, who I have had the honor of having as major professors. Both have offered continuous encouragement and assistance during the course of my graduate work. Dr. Sengupta has provided the stimulus and the direction to investigate areas heretofore unapproachable. The influence of Dr. Sengupta can be seen in the research leading to this dissertation. It has been a

great honor to work with both Dr. Sengupta and Dr. Kleinschmidt; to the writer, it is an even greater honor to count them as my friends.

**APPENDIX**

```

INPUT = 5
OUTPUT = 6
C   GEOMETRIC PROGRAM
C   SOLVED BY USING LINEAR PROGRAMS
C   INPUT DATA MUST BE A CONSISTENT LINEAR PROGRAM FORMED
C   BY CONVERTING A TIGHT MONOMIAL CONSTRAINED GEOMETRIC
C   PROGRAM.
C   INPUT DATA REQUIRED
C   FIRST CARD   NUMBER OF L.P.CONSTRAINTS
C   SECOND CARD NUMBER OF L.P. COLUMNS
C   AN ARRAY OF SUBSCRIPTS OF EPSILONS ASSOCIATED WITH EACH COLUMN
C   AN ARRAY OF POINTERS TO PAIRED EPSILONS
C   SUBROUTINE INTIAL READS IN THE LINEAR PROGRAM AND SOLVES FOR
C   A BASIC FEASIBLE SOLUTION AND THEN SAVES THIS FEASIBLE
C   SOLUTION FOR USE WITH LINEAR PROGRAM B
CALL INTIAL
100 FORMAT (' ',5D19.7)
C   COMPUTE SUM OF SQUARES OF ZJ -CJ
CALL ZJCJ (XX)
JJ = 0
11 Z = EX (M1T,1)
IF ( XX .LT. 0.1D-10) XX = 0.1D-10
1 YY = XX
IPHASE = 1
WRITE ( OUTPUT,21) JJ
21 FORMAT ( '1', ' ' ITERATION NUMBER ' ',I3)
JJ = JJ + 1
WRITE ( OUTPUT,20) EX(M1T,1)
20 FORMAT ('0','COMPUTED VALUE OF THE OBJECTIVE FUNCTION = ',D19.7)
JJJ = 1
70 CONTINUE
DO 22 I = 1,NT
IF ( IBS (I) .NE. JJJ) GO TO 22

```



```

      IF ( IT(I) .LE. 0) GO TO 23
      IF ( IT(I) .LT. I) GO TO 22
23  WRITE (OUTPUT, 101) JJJ, E(I), JJJ, DE(I)
      JJJ = JJJ + 1
22  CONTINUE
      IF ( JJJ .LE. KKK) GO TO 70
101 FORMAT (' ', 'E(', I3, ') = ', D19.7, 10X, 'DELTA E(', I4, ') = ', D19.7)
      DO16 I = 1, NT
      X=DEXP(EX(M1T, I))
16  WRITE (OUTPUT, 105) I, I, EX(M1T, I) , X
      DO 200 I = 1, MT
      K = IIB (I)
      DE ( I) = 0.0D0
200 CONTINUE
      DO 201 I = 1, MT
      K = NT + I
      P = 0.0D0
      DO 202 J = 1, MT
202 P = P + EX (J, K) *DF(J)
      P= DEXP (P)
      WRITE ( OUTPUT, 203) I, P
203 FORMAT (' ', ' X (', I5, ') = ', D19.5)
201 CONTINUE
105 FORMAT (' ', 'Z(', I3, ') -C(', I3, ')', D19.7, 'EXP', D19.7)
C   FORM LINEAR PROGRAM B USING BASIC FEASIBLE SOLUTION OF A
C   CALL FORM
C   COMPUTE ZJ -CJ FOR LINEAR PROGRAM B
C   CALL ZJ
C   SOLVE LINEAR PROGRAM B
C   CALL LPP
C   CHECK IF LINEAR PROGRAM B IS UNBOUNDED, IF SO CALL BOUND
C   IF ( ICHECK (1) .NE. 1) GO TO 2
C   COMPUTE FEASIBLE DIRECTION FOR CHANGE OF EPSILONS

```

```

2 CALL DELTA
6 CONTINUE
C   COMPUTE THE SUM OF SQUARES OF ZJ - CJ
   CALL ZJCJ (XX)
C   IF THE NEW SUM OF SQUARES IS NOT LOWER THAN PREVIOUS
C   DIVIDE THE CHANGE IN EPSILON BY TWO
   IF ( XX.LT. YY) GO TO 4
   IF ( IPHASE .NE. 2) GO TO 10
   DO 40 I = 1,N
40  E (I) = E(I) + DE(I)
   CALL ZJCJ (XX)
   GO TO 44
C THRU ... 3 DIVIDES DELTA E BY TWO
10 CONTINUE
   DO 3 I = 1,N
   DE (I) = DE(I) / 2.000
   3 E(I) = E(I) - DE(I)
   GO TO 6
C   CHECK FOR TERMINAL CONDITION
4 IPHASE = 2
   YY = XX
   GO TO 10
44 IF ( XX .GT. 0.1D-10) GO TO 1
   IF ( Z .LT. EX(M1T,1)) GO TO 11
   IF ( Z .GT. EX(M1T,1)) GO TO 10
1000 STOP
END
SUBROUTINE LPP
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(20,40),IB(40),IBT(40)
DIMENSION E(40),DE(40),IBS(40),IT(40)
DIMENSION ICHECK(40)
DIMENSION C(40),CB(40)

```

```

        DIMENSION UPPER ( 40) ,IMARK(40)
        INTEGER OUTPUT
        COMMON /A1/ INPUT,OUTPUT
        COMMON /A2/ A,IBT,IB
        COMMON /A4/ E,DE,IBS,IT
        COMMON /A5/ C,CB,XXXX,ICOL
        COMMON /A6/ ICHECK
        COMMON /A7/ N,M,M1,MN,KKK
        COMMON /A8/ NT,MT,M1T,MNT
C   SUBROUTINE LP IS A STANDARD SIMPLEX METHOD OF SOLVING L.P.'S
C   THRU   ... 60 INITIALIZES UNBOUNDED CHECK
        DO 300 I = 1,N
        UPPER (I) = 0.0D0
300   IMARK (I) = 0
        DO 200 JJJ = 1,KKK
        DO 200 I = 1,NT
        IF ( IBS (I) .NE. JJJ) GO TO 200
        IF ( IT(I) .EQ. 0) GO TO 201
        IF ( IT(I) .LT. 0) GO TO 202
        IF ( IT (I) .LT. I) GO TO 200
        UPPER(JJJ+1)=(1.0D0 - E(I))* .95D0
        UPPER ( JJJ + KKK + 1 ) = .95D0* E(I)
        GO TO 200
201   UPPER ( JJJ + 1) = 10000.D0
        GO TO 204
202   UPPER (JJJ + 1) = 1.0D0 - E(I)
204   UPPER ( JJJ + KKK + 1 ) = .95D0* E(I)
200   CONTINUE
        65 DO 60 I = 1 ,N
        60 ICHECK (I) = 0
C   SELECTS COLUMN TO ENTER BASIS
70   IJ= 0
        X =-0.1D-10

```

```

C   THRU ... 51 FINDS MOST NEGATIVE A(J) - C(J)
      DO 51 I= 2,N
      Z = A (M1,I)
      IF ( IMARK (I) .NE. 0) Z = - Z
      IF ( Z .GE. X) GO TO 51
      IF ( ICHECK (I).EQ.1 ) GO TO 51
      L=I-KKK
      IF (L .LT. 2) L=I+KKK
      IF (IMARK (L) .NE.0) GO TO 51
      IJ = I
      X=Z
51  CONTINUE
      IF (IJ .EQ. 0) GO TO 100
      JJ = IJ
      K = 0
      ITHETA = 3
      X = UPPER (IJ)
      Y = X
      DO 501 I = 1,M
      IF ( A(I,IJ) .LT. 0.1D-10) GO TO 501
      Y = A(I,1) /A(I,IJ)
      IF ( X.LE. Y) GO TO 501
      ITHETA = 1
      X = Y
      K = I
501 CONTINUE
      DO 502 I = 1,M
      IF ( A(I,IJ) .GT.-0.1D-10) GO TO 502
      IF ( IB (I) .GT. N) GO TO 502
      Y = -(UPPER (IB(I))-A(I,1)) /A(I,IJ)
      IF ( X.LE. Y) GO TO 502
      ITHETA = 2
      X = Y

```

```

100 FORMAT (' ', 'THE SUBSCRIPTS OF EPSILON FOR EACH COLUMN ARE')
    DO 12 I = 1, N
    12 WRITE ( OUTPUT, 101) I, IBS(I)
101 FORMAT (' ', 'SUBSCRIPT (' , I4, ') = ' , I4)
    WRITE (OUTPUT, 102)
102 FORMAT (' ', 'POINTERS FOR EPSILON')
    DO 13 I = 1, N
    13 WRITE (OUTPUT, 103) I, IT(I)
103 FORMAT (' ', 'POINTER (' , I4, ') = ' , I4)
C  SUBROUTINE LP SOLVES THE LINEAR PROGRAM
    CALL LP
C  THRU ... 11 SAVES THE BASIC FEASIBLE SOLUTION FO LINEAR PROGRAM A
C      IIBT (J) CONTAINS POINTER TO ROW IF BASIC
C                                     ZERO IF NON BASIC
C      IIB (I) CONTAINS POINTER TO BASIC COLUMN
C  THE REMAINDER SAVES PARAMETERS FOR LINEAR PROGRAM A
C      INITIALIZES PARAMETER FOR LINEAR PROGRAM B
C          L.P. A          L.P. B
C          N              NT      NUMBER OF COLUMNS
C          M              MT      NUMBER OF VARIABLES
C          M1             MIT     INDEX OF ROW OF Z(J) -C(J)
C          MN             MNT     COLUMN SIZE WITH ARTIFICIAL VARIABLES
    DO 10 J = 1, M
    IIB (J) = IB(J)
    DO 10 I = 1, MN
10  EX  (J,I) = A (J,I)
    DO 11 I = 1, MN
11  IIBT (I) = IBT (I)
    MT = M
    NT = N
    MIT = M1
    MNT = MN
    M = N - 1 - M

```

```

C ICOL IS THE COLUMN OF NORMAL C VALUE USED IN NORMALIZATION
C IVAR IS VARIABLE USED TO NORMALIZE L.P.'S COEFFICIENTS TO ONE
  READ ( INPUT,7) IVAR,ICOL
C THRU ... 8 NORMALIZE CONSTRAINT COSTS TO ONE
  XXXX = DLOG ( A (IVAR,ICOL))
  DO 8 I = 2,N
    8 A ( IVAR,I) = DLOG ( A (IVAR,I))/XXXX
  KKK = 0
C THRU ... 9 INTIALIZES ALL PAIRED EPSILON TO 0.5
C INTIALIZES ALL OTHERS TO ONE
C KKK IS TOTAL NUMBER OF EPSILONS USED
  DO 9 I = 1,N
    K = IBS (I)
    IF ( KKK .LT. K) KKK = K
    IF ( K .EQ. 0) GO TO 9
    E (I) = 0.5D0
    IF ( IT (I) .LE. 0) E (I) = 1.0D0
  9 CONTINUE
C SUBROUTINE COST COMPUTES COST COEFFICIENTS DEPENDENT ON
  CALL COST
C SUBROUTINE ZJ CALCULATES Z AND Z (J) -C (J) FOR LINEAR PROGRAM A
  CALL ZJ
  WRITE ( OUTPUT,104)
104 FORMAT (' ',' NORMALIZED INPUT MATRIX')
  DO 14 I = 1,M1
    14 WRITE (OUTPUT, 105) I,(A (I,J),J= 1,MN)
105 FORMAT (' ','ROW NUMBER OF LINEAR PROGRAM ',I10,20 (/5D19.7))
  WRITE ( OUTPUT ,106) (IB (I),I =1,M)
106 FORMAT(' ','BASIC COLUMNS APE',15I5)
  DO 15 I = 1,N
    15 WRITE (OUTPUT,107) I, C (I)
107 FORMAT (' ','COST COEFFICIENT (' ,I4,') =' ,D19.7)
  WRITE ( OUTPUT ,100)

```

```

      READ (INPUT,7) (IBS(I),I = 1,M)
C      ZERO IF EPSILON (I) IS RESTRICTED TO .GT. 0
C      -1 IF EPSILON (I) IS RESTRICTED TO .LE. 1
      READ ( INPUT,7) (IT (I),I = 1,N)
C M + N = SIZE OF LINEAR PROGRAM A WITH ARTIFICAL VARIABLES ADDED
      MN = M + N
      L = N + 1
C THRU ... 2 INITIALIZES ARRAY TO ZERO
      DO 1 I = L,MN
        IBS (I) = 0
      1 IT (I) = 0
        M1 = M + 1
        DO 2 I = 1,MN
          IBT (I) = 0
          E ( I) = 0.000
          DE (I) = 0.000
        2 C (I) = 0.000
C THRU ... 3 INITIALIZES ARTIFIAL VARIABLES AND THEIR COST FUNCTIONS
      DO 3 I = 1,M1
        IB (I) = L
        IBT (L) = I
        CB (I) = - 1000.000
        C (L) = - 1000.000
      DO 4 J = 1,MN
      4 A ( I,J) = 0.000
        A (I,L) = 1.000
      3 L = L + 1
C THRU ... 5 READ IN LINEAR PROGRAM A ARRAY
C      EACH COLUMN CORRESPONDS TO A GEOMETRIC PROGRAM
C      MONOMIAL CONSTRAINT
C      EACH ROW CORRESPONDS TO A PRIMAL VARIABLE
      DO 5 I = 1,M
      5 READ ( INPUT,6) ( A(I,J),J = 1,N)

```

```

      DO 531 I = 1, M
531  A(I, 1) = A(I, 1) + A(I, IJ) * UPPER (IJ)
      GO TO 65
100  CONTINUE
      L = M + 1
      DO 600 J = 1, N
      IF ( IBT (J) .NE. 0) GO TO 600
      IF ( IMARK (J) .E2. 0) GO TO 600
      A(L, 1) = UPPER (J)
      IBT (J) = L
      L = L + 1
600  CONTINUE
      RETURN
      END
      SUBROUTINE INTIAL
      IMPLICIT REAL*8 (A-H, O-Z)
      DIMENSION E(40), DE(40), IBS(40), IF(40)
      DIMENSION EX(20,40), IIB(40), IIBT(40)
      DIMENSION C(40), CB(40)
      DIMENSION A(20,40), IB(40), IBT(40)
      INTEGER OUTPUT
      COMMON /A1/ INPUT, OUTPUT
      COMMON /A2/ A, IBT, IB
      COMMON /A3/ EX, IIBT, IIB
      COMMON /A4/ E, DE, IBS, IT
      COMMON /A5/ C, CB, XXXX, ICOL
      COMMON /A7/ N, M, M1, MN, KKK
      COMMON /A8/ NT, MT, M1T, MNT
C   N  EQUALS NUMBER OF MONOMIAL CONSTRAINTS
      READ (INPUT, 7) M
C   M  EQUALS NUMBER OF CONVERTED PRIMAL CONSTRAINTS
      READ ( INPUT, 7) N
C   IBS (I) CONTAINS SUBSCRIPT OF EPSILON ASSOCIATED WITH MONOMIAL

```



```

      K = I
502 CONTINUE
      IF ( IMARK (IJ) .NE. 0) GO TO 510
      IF ( ITHETA .NE. 1) GO TO 505
      CALL BASIS ( IJ,K)
      GO TO 65
505 IF ( ITHETA .NE. 2) GO TO 506
      IJJ = IB (K)
      CALL BASIS (IJ,K)
      IJ = IJJ
      IMARK ( IJ) = 1
      DO 507 I = 1,M
507  A ( I, 1) = A (I, 1) - UPPER (IJ) * A (I,IJ)
      GO TO 65
506 IMARK (IJ) = 1
      DO 508 I = 1,M
508  A (I, 1) = A (I, 1) - UPPER (IJ) * A (I,IJ)
      GO TO 65
510 IF ( ITHETA .NE. 1) GO TO 520
      CALL BASIS (IJ,K)
      A (K, 1) = A (K, 1) + UPPER (IJ)
      IMARK (IJ) = 0
      GO TO 65
520 IF ( ITHETA .NE. 2) GO TO 530
      L = IB (K)
      CALL BASIS (IJ,K)
      IMARK (IJ) = 0
      IMARK (L) = 1
      A (K, 1) = A (K, 1) + UPPER (IJ)
      DO 521 I = 1,M
521  A ( I, 1) = A (I, 1)
      GO TO 65
530 IMARK (IJ) = 0

```

```

C CHECKS FOR PAIR CHANGE
  IF ( IT (J) .GT. 0) DE (IT(J)) = - DE(J)
  4 CONTINUE
C THRU ... 5 CHECKS FOR VIOLATED CONSTRAINTS ON EPSILON
C           IT (I) POSITIVE IF PAIR MUST EQUAL ONE
C           ZERO IF ONLY POSITIVE RESTRICTION
C           NEGATIVE IF O.LE. 1
C
  DO 5 I = 1,NT
  IF ( E(I) .LT. 0.1D-6) GO TO 5
  X = E(I) + DE (I)
  IF ( X .GT. 1.0D0) GO TO 6
  IF ( X .GT. 0.0D0) GO TO 5
  X = - E(I) * .95D0/DE(I)
  GO TO 7
  6 IF ( IT(I) .EQ. 0) GO TO 5
  X = (1.0D0 - E(I)) * .95D0/ DE(I)
  7 DO 8 J = 2,N
  8 DE(J) = DE(J) * X
  5 CONTINUE
C COMPUTE NEW SET OF EPSILONS
  DO 9 I = 1,NT
  9 E(I) = E(I) + DE(I)
  RETURN
  END

```

```

      IF ( DABS (A(M1,I)) .LE. 1.0D-10 ) A(M1,I) = 0.0D0
505 CONTINUE
      A(II,JJ) = 1.0D0
100 RETURN
      END
      SUBROUTINE DELTA
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(20,40),IB(40),IBT(40)
      DIMENSION E(40),DE(40),IBS(40),IT(40)
      INTEGER OUTPUT
      COMMON /A1/ INPUT,OUTPUT
      COMMON /A2/ A,IBT,IB
      COMMON /A4/ E,DE,IBS,IT
      COMMON /A8/ NT,MT,M1T,MNT
      COMMON /A7/ N,M,M1,MN,KKK
C     SUBROUTINE DELTA EXTRACTS A FEASIBLE DIRECTION FOR CHANGE OF
C     EPSILON
C     THRU ... 3 ZEROES CHANGE
      DO 1 I = 1,NT
      1 DE (I) = 0.0D0
C     THRU ... 4 SEARCHES FOR BASIC VARIABLE OF DELTA E
      DO 4 L = 1,KKK
C     THRU ... 3 FINDS SUBSCRIPT OF DELTA EPSILON
      DO 3 I = 2,NT
      IF ( IBS (I) .NE. L ) GO TO 3
      J= I
      GO TO 2
      3 CONTINUE
      2 JJ = L + 1
C     CHECKS OF POSITIVE CHANGE
      IF ( IBT (JJ) .NE. 0) DE(J) = A (IBT(JJ),1)
C     CHECKS FOR NEGATIVE
      IF ( IBT (JJ + KKK) .NE. 0) DE (J) = -A(IBT(JJ+KKK),1)

```

```

DO 52 I = 1, M
IF (A (I, JJ) .LE. 0.1D-7) GO TO 52
IF (X .EQ. -1.0D0 ) X = A (I, 1) / A (I, JJ)
IF (X .LT. (A (I, 1) / A (I, JJ))) GO TO 52
X = A (I, 1) / A (I, JJ)
II = I
52 CONTINUE
C RETURN IF COLUMN JJ CANNOT BE ADDED TO BASIS
IF (II.EQ. 0 ) GO TO 100
53 CONTINUE
C CHANGE OF BASIS
C MARKS CHANGES IN POINTERS
I = IB (II)
IBT(I) = 0
IBT (JJ ) = II
IB (II) = JJ
CB (II) = C(JJ)
C CALCULATE NEW TABLEAU
C THRU ... 31 COMPUTES NEW ROWS EXCEPT PIVOT ROW
DO 31 I= 1, M1
IF (I.EQ. II) GO TO 31
X = A (I, JJ) / A (II, JJ)
DO 32 J= 1, MN
32 A(I, J) = A (I, J) - A(II, J) * X
31 CONTINUE
X = A (II, JJ)
C THRU ... 35 COMPUTES NEW PIVOT ROW
DO 35 J= 1, MN
35 A(II, J) = A (II, J) / X
DO 40 I = 1, M1
40 A(I, JJ) = 0.0D0
C THRU ... 505 CLEAN -UP PROCEDURE
DO 505 I = 1, N

```

```

107 FORMAT (' ', 'COST COEFFICIENT (' , I4, ') = ' , D19.7)
RETURN
END
SUBROUTINE ZJ
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION C(40),CB(40)
DIMENSION A(20,40),IB(40),IBT(40)
INTEGER OUTPUT
COMMON /A1/ INPUT,OUTPUT
COMMON /A2/ A,IBT,IB
COMMON /A5/ C,CB,XXXX,ICOL
COMMON /A7/ N,M,M1,MN,KKK
C SUBROUTINE ZJ COMPUTES Z(J) - C(J) GIVEN COST COEFFICIENTS OF LINEAR
C PROGRAM
DO 55 J= 1,N
A(M1,J) = - C(J)
DO 55 I= 1,M
55 A(M1,J) = A(M1,J) + CB(I) * A(I,J)
RETURN
END
SUBROUTINE BASIS (JJ,II)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(20,40),IB(40),IBT(40)
DIMENSION C(40),CB(40)
INTEGER OUTPUT
COMMON /A1/ INPUT,OUTPUT
COMMON /A2/ A,IBT,IB
COMMON /A5/ C,CB,XXXX,ICOL
COMMON /A7/ N,M,M1,MN,KKK
C SUBROUTINE BASIS IS STANDARD SIMPLEX TABLEAU CHANGE
IF ( II.NE. 0) GO TO 53
X =-1.0D0
C THRU ... 52 FINDS PIVOT ROW

```

```

      K = 0
C   SUBROUTINE SASIS TRIES TO ADD JJ COLUMN TO BASIS
      CALL BASIS ( JJ,K)
      IF (K.NE. 0) GO TO 65
C   YES CONTINUE ON
C   NO MARK AS UNBOUNDED AN TRY OTHERS
      ICHECK (JJ) = 1
      ICHECK (1) = 1
      GO TO 70
100  CONTINUE
      DO 200 I=1,M
      J=I
      IF (IB(I).LE.N) GO TO 200
      DO 240 III=2,N
      K=III
      IF (IBT(III).NE.0) GO TO 240
      IF (A(J,K).LE. 0.1D-6) GO TO 240
      X= A(J,1)/A(J,K) - 0.1D-6
      DO 250 II=1,M
      IF (A(II,K).LE. 0.1D-6) GO TO 250
      IF (A(II,1)/A(II,K) .LT.X) GO TO 240
250  CONTINUE
      CALL BASIS (K,J)
      GO TO 100
240  CONTINUE
200  CONTINUE
      DO 14 I = 1,M1
      14 WRITE (OUTPUT, 105) I, (A(I,J),J= 1,MN)
105  FORMAT (' ', 'ROW NUMBER OF LINEAR PROGRAM ',I10,20 (/5D19.7))
      WRITE ( OUTPUT ,106) (IB(I),I =1,M)
106  FORMAT(' ', 'BASIC COLUMNS ARE',15I5)
      DO 15 I = 1,N
      15 WRITE (OUTPUT,107) I, C(I)

```

```

      KK = KK + 1
      C (KK) = C (KK) - EX (I, 1) * DE (IIB (I))
      C (KK + KKK) = - C (KK)
10  CONTINUE
      RETURN
      END
      SUBROUTINE LP
      IMPLICIT REAL*8 (A-H, O-Z)
      DIMENSION A (20, 40), IB (40), IBT (40)
      DIMENSION C (40), CB (40)
      DIMENSION ICHECK (40)
      INTEGER OUTPUT
      COMMON /A1/ INPUT, OUTPUT
      COMMON /A2/ A, IBT, IB
      COMMON /A5/ C, CB, XXXX, ICOL
      COMMON /A6/ ICHECK
      COMMON /A7/ N, M, M1, MN, KKK
C   SUBROUTINE LP IS A STANDARD SIMPLEX METHOD OF SOLVING L.P.'S
C   THRU ... 60 INITIALIZES UNBOUNDED CHECK
      65 DO 60 I = 1, N
      60 ICHECK (I) = 0
C   SELECTS COLUMN TO ENTER BASIS
      70 IJ = 0
      X = -0.1D-10
C   THRU ... 51 FINDS MOST NEGATIVE Z (J) - C (J)
      DO 51 I = 2, N
      IF ( A (M1, I) .GE. X) GO TO 51
      IF ( ICHECK (I) .EQ. 1 ) GO TO 51
      IJ = I
      X = A (M1, I)
51  CONTINUE
      IF (IJ .EQ. 0) GO TO 100
      JJ = IJ

```

```

      K = K + 1
      A (K, 1) = -EX (M1T, J)
      KK = IBS (J)
      IF ( KK.NE. 0 ) A (K, KK+ 1 ) = A(K, KK+ 1) - DE (J)
      DO 5 I = 1, MT
      KK = IBS (IIB (I))
      IF ( KK.NE. 0) A (K, KK+1) = A (K, KK+1) + EX (I, J) *DE (IIB (I))
5 CONTINUE
4 CONTINUE
C THRU ... 6 ADDS COLUMNS TO INSURE POSITIVE DELTA EPSILONS
  DO 6 I = 1, KKK
  KK = I + KKK + 1
  DO 6 J = 1, M
5 A (J, KK) = - A (J, I + 1)
C THRU ... 7 FORCE INTIAL SOLUTION TO BE BASIC FEASIBLE
  DO 7 I = 1, M
  IF ( A (I, 1) .GT. 0.0D0) GO TO 7
  DO 8 J = 1, N
  A (I, J) = -A (I, J)
  7 CONTINUE
  J = N
C THRU ... 9 ADDS LARGE NEGATIVE COST COEFFICIENTS TO INITIAL SOLUTI
  DO 9 I = 1, M
  J = J + 1
  CB (I ) = - 10000.0D0
  C (J) = -10000.0D0
  IBT (J) = I
  9 IB (I) = J
C THRU ... 10 USE L.P. AS OBJECTIVE FUNCTION COST PARTIALS
C FOR FORMING OBJECTIVE FUNCTION FOR L.P. B
  DO 10 I = 1, MT
  KK = IBS (IIB (I))
  IF (KK.EQ. 0) GO TO 10

```



```

DIMENSION A(20,40),IB(40),IBT(40)
DIMENSION E(40),DE(40),IBS(40),IT(40)
DIMENSION EX(20,40),IIB(40),IIBT(40)
INTEGER OUTPUT
COMMON /A1/ INPUT,OUTPUT
COMMON /A2/ A,IBT,IB
COMMON /A3/ EX,IIBT,IIP
COMMON /A4/ E,DE,IBS,IT
COMMON /A5/ C,CB,XXXX,ICOL
COMMON /A7/ N,M,M1,MN,KKK
COMMON /A8/ NT,MT,M1T,MNT
C SUBROUTINE FORM SETS UP LINEAR PROGRAM TO COMPUTE DELTA EPSILON
C THRU ... 2 INITIALIZES ARRAYS
  DO 1 I = 1,M
    CB(I) = 0.0D0
  1 IB(I) = 0
    DO 2 J = 1,MN
      IBT(J) = 0
      C(J) = 0.0D0
    DO 2 I = 1,M1
      2 A(I,J) = 0.0D0
C THRU ...3 COMPUTE THE PARTIAL DEPRIVATIVE OF C(J)
  DO 3 I= 2,NT
    KK = IBS(I)
    IF ( KK.EQ. 0 ) GO TO 3
    DE(I) = - 1.0D0/E(I)
    IF ( IT(I) .LE. 0 ) GO TO 3
    IF ( IT(I) .LT. I) DE(I) = - DE(I)
  3 CONTINUE
  K = 0
C THRU ... 4 SETS UP KKK COLUMNS OF L.P. B ONE FOR EACH EPSILON
  DO 4 J = 2,NT
    IF ( IIBT(J) .NE. 0) GO TO 4

```

```

DO 4 J = 2,NT
4 XX = XX + EX(M1T,J)*EX(M1T,J)
WRITE ( OUTPUT,5) XX
5 FORMAT (' ', 'THE SUM OF SQUARES OF ZJ -CJ  =',D19.9)
RETURN
END
SUBROUTINE COST
C SUBROUTINE COST USED THE EPSILONS TO CALCULATE COST COEFFICIENTS
C EPSILON AND INITIALIZES THE BASIC COST VECTOR
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(20,40),IB(40),IBT(40)
DIMENSION E(40),DE(40),IBS(40),IT(40)
DIMENSION C(40),CB(40)
INTEGER OUTPUT
COMMON /A1/ INPUT,OUTPUT
COMMON /A2/ A,IBT,IB
COMMON /A4/ E,DE,IBS,IT
COMMON /A5/ C,CB,XXXX,ICOL
COMMON /A7/ N,M,NT,XXX
COMMON /A8/ NF,NT,INT
DO 1 I = 1,N
C(I) = 0.000
IF ( IBS(I) .EQ. 0) GO TO 1
C(I) = -DLOG (E(I))
1 CONTINUE
DO 2 I = 1,M
2 CB(I) = C( IB(I))
C(ICOL) = -XXXX
RETURN
END
SUBROUTINE FORM
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION C(40),CB(40)

```

```

M1 = M + 1
N = KKK* 2 + 1
MN = N
RETURN
END
SUBROUTINE ZJCJ (XX)
C SUBROUTINE ZJCJ (XX) CALCULATES COST COEFFICIENTS DEPENDENT
C ON EPSILON
C USES THE COST COEFFICIENTS TO COMPUTE Z(J) -C(J)
C COMPUTES THE SUM OF SQUARES OF Z(J) -C(J)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION C(40),CB(40)
DIMENSION E(40),DE(40),IBS(40),IT(40)
INTEGER OUTPUT
DIMENSION EX(20,40),IIB(40),IIBT(40)
COMMON /A1/ INPUT,OUTPUT
COMMON /A3/ EX,IIBT,IIB
COMMON /A4/ E,DE,IBS,IT
COMMON /A5/ C,CB,XXXX,ICOL
COMMON /A8/ NT,MT,M1I,1NT
DO 1 I = 1,NT
C(I) = 0.0D0
IF ( IBS (I) .EQ. 0) GO TO 1
C(I) = - DLOG (E(I))
1 CONTINUE
C (ICOL) = -XXXX
DO 2 I = 1,MT
2 CB (I) = C (IIB(I))
DO 3 J = 1,NT
EX (M1I,J) = -C(J)
DO 3 I = 1,MT
3 EX (M1I,J) = EX (M1I,J) + CB(I)*EX(I,J)
XX = 0.0D0

```