NASA TECHNICAL MEMORANDUM

NASA TM X-73372

(NASA-TM-X-73372) OPERATIONS RESEARCH INVESTIGATIONS OF SATELLITE POWER STATIONS (NASA) 41 p HC A03/MF A01 CSCL 10A

N77-21547

Unclas

G3/44 22916

OPERATIONS RESEARCH INVESTIGATIONS OF SATELLITE POWER STATIONS

By John W. Cole and John L. Ballard
Program Development

December 1976

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
NASA TM X-73372		

4. TITLE AND SUBTITLE	5. REPORT DATE
Operations Research Investigations of Satellite Power Stations	December 1976
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT #
John W. Cole and John L. Ballard*	
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. WORK UNIT NO.
George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812	11. CONTRACT OR GRANT NO.
	13. TYPE OF REPORT & PERIOD COVERED
12. SPONSORING AGENCY NAME AND ADDRESS	
National Aeronautics and Space Administration Washington, D.C. 20546	14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES *Dr. Ballard is an Assistant Professor, Department of Industrial and Management Systems Engineering, University of Nebraska, Lincoln, Nebraska. He was a member of the 1976 NASA-ASEE Summer Faculty Fellowship Program.

16. ABSTRACT

A systems model reflecting the current "in-house" design concepts of Satellite Power Stations (SPS) was developed. The model is of sufficient scope to include the interrelationships of the following major design parameters: the transportation to and between orbits; assembly of the SPS; and maintenance of the SPS.

The systems model is composed of a set of equations that are nonlinear with respect to the system parameters and decision variables. The model determines a "figure of merit" from which alternative concepts concerning transportation, assembly, and maintenance of satellite power stations can be studied. A hybrid optimization model was developed to optimize the system's decision variables. The optimization model consists of a random search procedure and the optimal-steepest descent method. A FORTRAN computer program was developed to enable the user to optimize nonlinear functions using the model. Specifically, the computer program was used to optimize Satellite Power Station system components.

17. KEY WORDS	18. DISTRIBUTION STATEMENT
	Unclassified-Unlimited

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	42	NTIS

TABLE OF CONTENTS

Technical Memorandum X-73372

OPERATIONS RESEARCH INVESTIGATIONS OF SATELLITE POWER STATIONS

INTRODUCTION

Several concepts have been proposed for generating electric power in space, transmitting the energy to Earth, and using the energy as useful power. Initial analyses of these concepts indicate that they may be competitive with future commercial power rates; however, advances in technology are required well beyond the current state-of-the-art to make the concepts cost effective.

Three basic concepts have been identified as possible cost effective candidates: the photovoltaic, the thermal concentrator, and the nuclear. The photovoltaic designs typically consist of solar cells arranged with a lightweight concentrator into a large, essentially flat array of 10 by 20 kilometers producing on the order of 18 gigawatts of electricity. Such a satellite will weigh in excess of 100 million kilograms. To be of significant benefit to the U.S. energy requirements, at least one must be placed in synchronous orbit each year for 30 years.

The thermal concentrator system typically consists of many large concentrating mirrors built of smaller flat facets which concentrate the solar flux onto thermionic diodes, thermal absorbers for some working fluid, or a combination of both. This concept is typically one-third the dimension of the photovoltaic, but twice the weight.

The nuclear concept, using high temperature gas reactors, seems to provide the best nuclear option and is considerably smaller than the other concepts, but is much heavier.

Many concept and design questions are still open. The economic availability of the SPS program will be strongly dependent on the technical design, logistics, assembly, maintenance, and operations philosophies selected. There is a desperate need for techniques that will search out the optimum answers to complex and involved relationships of design, construction, and operation. To this end the following research was proposed.

DESCRIPTION OF RESEARCH

Develop a systems model of the current in house design of the satellite power stations of both the photovoltaic type and the solar concentrator with a thermal engine type. The models should be of sufficient scope to include the interrelationships of the major design parameters, the transportation to and between orbits, assembly and maintenance, and power benefits throughout the useful life of the system. Define a figure of merit describing the power benefits, and develop a method for finding the benefit partial derivatives with respect to the significant design variables. Investigate nonlinear programming methods for optimizing the model design for maximum benefit subject to linear design constraints. Implement an appropriate optimization method.

Develop a systems model of a reasonably equivalent ground-based solar power station and apply the above techniques to optimize the design. Evaluate and compare the power concepts investigated.

The level of depth of model fidelity should be limited to the extent necessary to prove the analysis technique. Sufficient depth should be included, however, to facilitate expansion of the models for more detailed in-house investigations.

This research is intended to be performed during two 10-week terms of activity, specifically, summer 1976 and summer 1977.

STUDY STATUS

During the first term, investigation of contractor descriptions and NASA descriptions of Satellite Power Systems indicated that the model equations could be described by nonlinear equations constrained by bounded variables. An optimization procedure was developed to solve a set of equations subject to such conditions and was applied to an expanded version of the ECON sizing equations (reference 1). The program was debugged and applied to the low

Earth orbit (LEO) vs. the geosynchronous Earth orbit (GEO) assembly questions and to the photovoltaic and thermal concentrator design. The fidelity and extent of the model equation was not sufficient, however, to adequately investigate the pertinent question, but was quite adequate to verify the optimization techniques and procedures.

The following term will bring the model equation into consistency with current concepts and will expand them to be able to adequately address some of the critical problem areas previously mentioned. Comparative analyses of alternative concepts will be conducted and, if time permits, an equivalent ground-based solar concept will be modeled to provide a more firm basis of comparison.

RESEARCH OBJECTIVES AND PROCEDURES

The objective of this research was to investigate the potential of using operation research techniques in planning the logistic requirements for the construction of a Satellite Power Station (SPS). As in most operations research studies development of a mathematical system model was a necessity. Specific attention was given to developing a model of the transportation to and from orbit and of the assembly subsystems. The modeling approach taken was to define the pertinent decision variables in the system. The values of these variables are of prime interest and will be directly determined through the solution procedure. An existing mathematical model was modified in order to integrate the decision variables, system parameters, and system restrictions into one model. The final product of the modeling was the determination of an objective function that defines a measure of the effectiveness of the system. This objective function provides a means of comparing alternative feasible solutions.

The second step in the research activity was the investigation of optimization techniques that could be applicable to the analysis of the existing mathematical model. Optimization techniques fall into two major classes - linear and nonlinear optimization methods. If a mathematical model contains only linear interrelationships between the decision variables and system parameters in both the objective function, as well as, in all the constraints, the model is classified as a linear optimization model. Otherwise, the model is categorized as a nonlinear optimization model. Solution techniques applicable to practical nonlinear optimization models are not as well developed as those used to solve linear optimization problems.

Most solution techniques used to solve optimization problems are iterative. That is, the optimal solution is found in a step-wise fashion. Each successive iteration provides a new set of decision variable values that produces a superior value of the objective function, and the optimal solution is determined at the final iteration. The final product of this research was the implementation of a computerized algorithm that can be used to numerically solve a bounded nonlinear optimization problem.

Satellite Power System Model

The Satellite Power System model consists of the following subsystems: (1) the satellite sizing subsystem; (2) the assembly equipment sizing subsystem; (3) the transportation subsystem; (4) the ground station support subsystem; and, (5) the cost subsystem. Figure 1 depicts the five subsystems and their interrelationships. The satellite sizing subsystem for the photovoltaic SPS concept consists of the

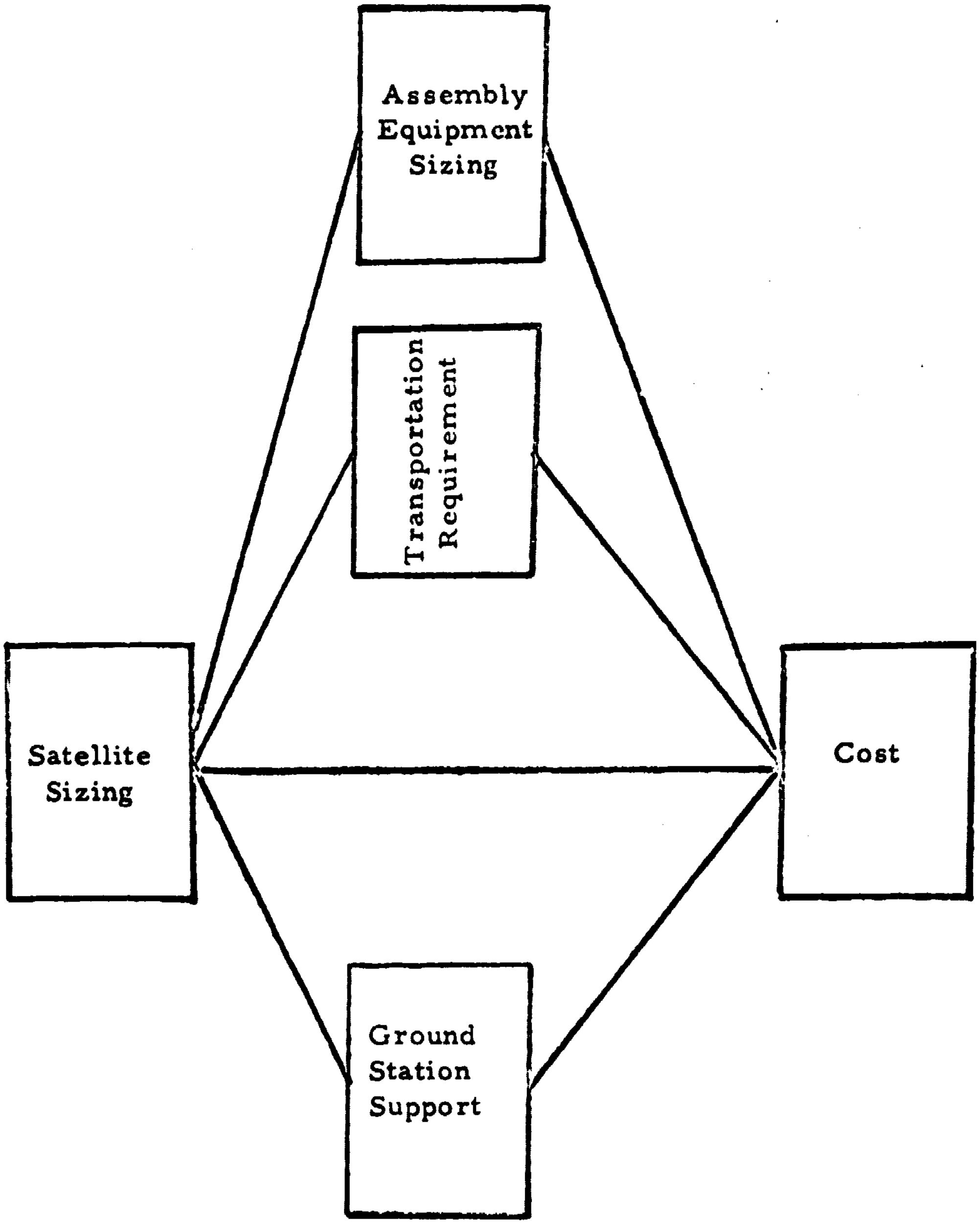


Figure 1.

Satellite Power System Subsystem Schematic following output variables:

(1) Power output at rectenna (kw)
(2) Aera of the solar blanket (km^2)
(3) Aera of the solar concentrator (km^2)
(4) Mass of the solar blankets (kg)
(5) Mass of the solar concentrator (kg)
(6) Mass of the conducting structure (kg)
(7) Mass of the non-conducting structure (kg)
(8) Mass of the central mast (kg)
(9) Total mass of the antenna structure (kg)
(10) Total mass of the dc-rf converters (kg)
(11) Total mass of the antenna interface (kg)
(12) Total mass of the phase control electronics (kg)
(13) Total mass of the antenna (kg)
(14) Miscellaneous mass (kg)
(15) Total mass of the operational satellite (kg)

The fifteen preceeding satellite sizing variables provide inputs to the four other subsystems.

The assembly equipment sizing subsystem determines the individual and total mass of assembly equipment and personnel required for the construction of one SPS. Certain decision variables found in this subsystem are the percentage of total satellite mass to be assembled by man input, total man-days of construction time, rate of manned-assembly, rate of remote controlled assembly, and the productivity of operations in space. Outputs of the assembly equipment sizing subsystem are total mass of the satellite to be assembled by man input, total mass of the satellite to be constructed by remote construction, total man-days of construction time, total machine days of construction time, number of on orbit personnel, number of on-orbit teleoperators, total number of fabrication modules, total number of manned manipulators, total number of LEO space stations, total mass of the fabrication units, total mass of the teleoperator units, total mass of the Low Earth Orbit (LEO) support vehicles, total mass of the extra-vehicular activity, total mass of the manned manupulator units, total mass of the LEO space stations, total mass of the assembly equipment propellant, and total mass of the space station resupply. The outputs of the assembly subsystem provide inputs to the transportation requirements subsystem and the cost subsystem.

The transportation subsystem computes the sizing of the components necessary to transport the crew modules between the LEO and geosynchronous (GEO) space stations. Among the required inputs are the mass of the crew modules, the mass of the orbital transfer vehicles propellants, total construction time, and the time between crew rotations. Also, an advanced ion stage is sized to transport an assembled SPS from LEO to GEO. If other alternatives than LEO assembly are to be considered, this subsystem would be substantially modified. Other significant factors computed by the transportation subsystem are the heavy lift launch vehicle requirements for the construction and equipment support for the assembly of one SPS and the Shuttle requirements for the transportation of personnel to LEO and vehicle requirements for transfer to GEO.

The cost subsystem utilizes the output of all the previous subsystems and the ground station support subsystem. The final product of the model is the output of the cost subsystem, and is an expression for total production cost of one SPS. The cost expression is composed of the total LEO launch cost, total space station and assembly cost, total satellite procurement cost, and the total ground station procurement cost. Presently, the cost model is an aggregation of an earlier model developed for NASA by ECON (1) and current MSFC concepts. The model has been transformed into a FORTRAN subroutine consisting of 169 decision variables and parameters. Many of the interrelationships between the variables are nonlinear.

NONLINEAR OPTIMIZATION METHODS

Classical Optimization Methods

Classical nonlinear optimization techniques are based upon theoretical mathematical analyses that involve an application of the principles of calculus to problems involving maxima and minima. In order to apply the classical optimization techniques to the minimization (maximization) of a function, the function must be shown to be continuous and differentiable within a region (R) and to have a minimum (maximum) within the region. The well-known theorem of Weierstrass (3) states:

"Every function which is continuous in a closed region R of variables (X_1, X_2,..., X_n) possesses a largest and a smallest value within the interior or on the boundary of that region." Therefore, this theorem asserts that an extreme point exists within or on the boundary of a region R. Gottfried and Weisman (4), Hadley (5), and Taha (6) among others present discussions of the application of classical optimization techniques to single-dimensional and multi-dimensional unconstrained functions. These techniques are based upon satisfying certain necessary and sufficient conditions. The necessary condition for a function, $f(X_1, X_2, \ldots, X_n)$, to pass through an extremum at the point (X_{10}, X_{20},..., X_{no}) is that the partial derivate of $f(X_1, X_2, \ldots, X_n)$ vanishes at (X_{10}, X_{20},..., X_{no}). The extremum may be a relative maximum, relative minimum, or a saddle point. The sufficient condition for the characterization of an extremum as a relative maximum, or a saddle point. The sufficient condition for the characterization of an extremum as a relative maximum or minimum is restated by Gottfried and Weisman (4) as follows:

Let $f(X_1, X_2, \ldots, X_n)$ vary continuously in an open region R. Consider the set of determinants D_i, i= 1, 2,..., n, where

$$D_i = \begin{vmatrix} \frac{\partial f}{\partial X_1{}^2} & \frac{\partial f}{\partial X_1 2X_2} & \cdots & \frac{\partial f}{\partial X_1 2X_n} \\ \frac{\partial f}{\partial X_2 2X_1} & \frac{\partial f}{X_2{}^2} & \cdots & \frac{\partial f}{X_2 2X_n} \\ \vdots & & \cdots & \vdots \end{vmatrix}$$

$$\left| \frac{\partial f}{\partial X_n 2X_1} \quad \frac{\partial f}{\partial X_n 2X_2} \quad \cdots \quad \frac{\partial f}{\partial X_n^2} \right|$$

evaluated at $(X_{10}, X_{20}, \ldots, X_{no})$

If $\partial f / \partial X_1 = \frac{\partial f}{\partial X_2} = \ldots = \frac{\partial f}{\partial X_n} = 0$ at $(X_{10}, X_{20}, \ldots, X_{no})$ then,

(1) D_i less than 0 for $i = 1, 3, 5, \ldots$ and D_i greater than 0 for $i = 2, 4, 6, \ldots$ indicate the presence of a relative maximum at $(X_{10}, X_{20}, \ldots, X_{no})$.

(2) D_i greater than 0 for $i = 1, 2, \ldots n$ indicates the presence of a relative minimum at $(X_{10}, X_{20}, \ldots, X_{no})$.

(3) The failure to satisfy conditions (1) or (2) indicates a saddle point at $(X_{10}, X_{20}, \ldots X_{no})$.

Although the preceding conditions are satisfied, the classical approach to solving maxima and minima problems can only guarantee local minima and maxima and does not provide a direct means of finding the global or absolute minimum (maximum).

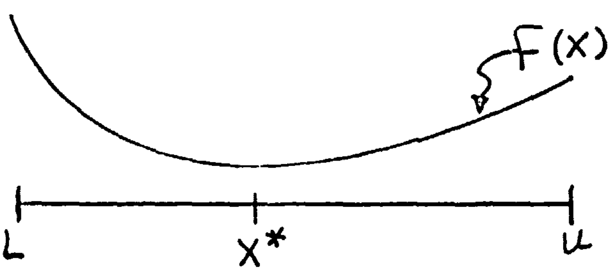
Classical optimization theory has been extended to minimizing (maximizing) a function $f(X_1, X_2, \ldots, X_n)$ subject to n equality constraints of the form $g_j (X_1, X_2, \ldots, X_n) = 0$. The technique employed is the method of Langrangian Multipliers. Kuhn and Tucker (7) derived the necessary and sufficient conditions for the Langrangian function to possess a saddle point at $(X_{10}, X_{20}, \ldots, X_{no}, \lambda_{10}, \lambda_{20}, \ldots, \lambda_{mo})$.

In principle, classical optimization methods may be applied to a general class of nonlinear problems (either constrained or unconstrained); however, severe computational difficulties arise when solving high-dimensional problems. In fact, Hadley (5) asserts that classical methods are best suited for theoretical analyses or especially simple situations. They are not suited for numerical computations. Gottfried and Weisman (4) state that while classical theory serves to provide insight into the characteristics and problems associated with extremizing continuous functions, it does not provide efficient computational procedures for optimizing practical problems. However, classical theory provides a basis for the development of more efficient computational algorithms.

Unconstrained Optimization Search Techniques

Since classical optimization methods have been proven an inadequate means of solving practical nonlinear optimization problems, several numerical searching algorithms will be discussed as potential problem solving methods. Many numerical techniques operate in a sequential fashion. The algorithms search for the optimum by generating a succession of search points, and most use past information (previous search points) to determine a new search point with a corresponding improvement in the objective function. If the objective function is unimodal, sequential search techniques will yield an absolute optimum; otherwise, the procedure may only yield a local minimum (maximum) or a saddle point. Gottfried and Weisman (4) state that although many practical engineering problems contain multi-modal objective functions, one can usually determine a subregion over which the function is unimodal and sequential search techniques provide a useful means for locating the optimum.

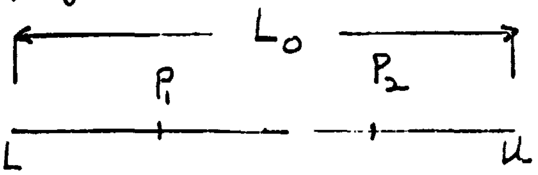
The simplest forms of search techniques are known as direct-search techniques. Such methods evaluate a function at several data points within a region in order to estimate the location of the minimum (maximum). A typical one-dimensional function is depicted below:



The function f(X) is unimodal on the interval (L, U). The minimum of f(X) lies at X^*. The goal of a direct search technique is to isolate the absolute minimum of f(X) in the interval L_n after the evaluation of seven data points. The more powerful of two search techniques is the one that produces the smallest interval of uncertainty, L_n. Typical examples of one dimensional search techniques are the half-interval method, symmetrical two-point search, three-point search, Fibonacci search, and the golden-ratio search. Gottfried and Weisman (4) suggest that the Fibonacci and golden-ratio search techniques are among the best available one-dimensional algorithms.

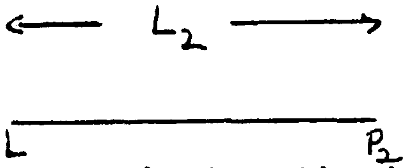
The golden-ratio search is based upon the golden ratio (P=1.618034). The procedure used by the golden-ratio search on the interval width L_0 is as follows:

(1) Locate two search points a distance L_0/P from the end of the orginal interval, L_0.



$f(P_1) < f(P_2)$

(2) The new search interval becomes:



(3) Locate two search points within the interval L_2, 1/P units from the new end points and evaluate the function at each point.

(4) Continue the procedure outlined in step 3 for M iterations.

(5) The estimation of the value of X that provides the optimal value of f(X) lies at the center of L_M, the last interval of uncertainty. The golden-ratio search procedure is an efficient technique and possesses decided computational advantages over Fibonacci search method. The algorithm is easily programmed on a digital computer and can become one of the components of a multi-dimensional gradient algorithm.

In order to minimize (maximize) a multi-dimensional continuous function $f(X_1, X_2, \ldots, X_n)$ that is differentiable and unimodal, a class of numerical techniques known as gradient methods can be utilized. These methods are based upon classical optimization theory and employ numerical procedures to locate the point $(X_1, X_2, \ldots, X_n)$ that optimizes $f(X_1, X_2, \ldots, X_n)$. Among these procedures are the method of steepest descent, the conjugate gradient procedure, and the variable-metric algorithm.

The method of steepest descent utilizes numerical techniques for minimizing the function $f(X_1, X_2, \ldots, X_n)$. An algorithm for the method of steepest descent outlined by Gottfried and Weisman (4) is as follows:

(1) Find an initial point $(X_{10}, X_{20}, \ldots, X_{no})$ within the region and evaluate $f(X_{10}, X_{20}, \ldots, X_{no})$.

(2) Evaluate the gradient vector $\nabla f(X_1, X_2, \ldots, X_{no})$ at the point $(X_{10}, X_{20}, \ldots, X_{no})$. The partial derivate evaluated numerically is as follows:

$$\frac{\partial f}{\partial X_i} \sim \frac{f(X_1, X_2, \ldots, X_i + D/2, \ldots) - f(X_1, X_2, \ldots, X_i - D/2, \ldots)}{D}$$

$$i = 1, 2, \ldots, n$$

(3) A new point $(X_{11}, X_{21}, \ldots, X_{n1})$ is found by

$$(X_{11}, X_{21}, \ldots, X_{n1}) = (X_{10}, X_{20}, \ldots, X_{no}) - f(X_{10}, X_{20}, \ldots, X_{no})\, T$$

The new point is found by proceeding in the direction of the negative gradient an arbitrarily small distance indicated by T. (T may be a scalar or a vector of dimension n).

(4) Let $(X_{10}, X_{20}, \ldots, X_{no}) = (X_{11}, X_{21}, \ldots, X_{n1})$ and return to Step 2.

(5) The procedure ends when:

$$\frac{\partial f}{\partial Xi} \leq \epsilon, \quad \epsilon \sim 0 \text{ for all } i$$

and the last point determined is the stationary value of $f(X_1, X_2, \ldots, X_{no})$. The method of steepest descent may lead to a saddle point rather than an extremum, although this is unlikely (8). Nevertheless, the characteristics of the stationary point can be analyzed by using random search techniques.

The steepest-decent algorithm can be improved if T is chosen in an optimal fashion, such that, $F(X_1, X_2, \ldots, X_n)$ possesses a relative minimum along the line joining X_K and X^*_{K+1}. Where,

$$X_K = (X_{1K}, X_{2K}, \ldots, X_{NK})$$

$$X^*_{K+1} = X_K - \nabla fT$$

A one-dimensional search technique can be utilized to find the optimal distance to move along the line joining X_K and X_{K+1}. The point is X_{K+1} where, $X_{K+1} = \theta_K X_K + (1 - \theta_K) X^*_{K+1}$. The value of θ_K is found using a one-dimensional optimal search technique.

$f(X_{K+1})$ less than $f(X_K)$ and,

$f(X_{K+1})$ less than $f(X^*_{K+1})$.

The steepest descent method works well if the computation occurs on the interior of the regions; however, if the search region is bounded $a \leq X \leq b$ and the gradient vector is directed out of the region, the one-dimensional search may proceed to move outside of the region. The move should terminate at the region boundary. Another difficulty arises when the gradient vector is calculated at the boundary and some components point outside the boundary. Gottfried and Weisman (4) state when this occurs, it is generally satisfactory to set these components equal to zero and search in the direction of the modified search vector. The optimal steepest-descent procedure terminates when the modified gradient vector is sufficiently small.

Although the method of steepest-descent is one of the most straightforward of all the gradient techniques, it still possesses some numerical difficulties. The number of computations required to extremize a function depends upon the degree of the function's sensitivity to changes in the independent variables. Also, the steepest descent method may "zig-zag" toward the optimum and require many steps of decreasing size as the optimum is approached.

In contrast to sequential optimization techniques, random search techniques are not based upon classical theory and can be applied to a more general classification of optimization problems. The functions need not be continuous, differentiable, or unimodal; therefore, the rationale behind the random search technique is not mathematically sophisticated. A point within the region of interest is chosen at random and the function is evaluated. The procedure continues until n points have been evaluated. At the termination of the search, the point found yielding the best value of the function is the extreme point. Random search techniques are useful in evaluating discontinuous functions and for terminal explorations when using sequential optimization techniques. Gottfried and Weisman (4) note that random search procedures offer a practical approach to the initial exploration of a function that may be multimodal and that their use in combination with sequential methods is often highly effective.

A Computerized Optimization Program

In order to achieve a flexible nonlinear optimization routine, a computer program that combines a random search procedure and an optimal - steepest descent algorithm was written in FORTRAN. The theoretical background for these numerical procedures employed by the program was presented in the previous section of this report. The program was designed to be a modular program consisting of a generalized main program and collection of specialized subroutines. A simplified chart for the main program is depicted in Figure 2. The main program performs input and output activities. Specific input requirements are discussed in detail at the end of this section. Also, it conducts the random search, and monitors the sequential search procedure.

The random search segment examines a specified number (NINT) of points within the region of interest. The procedure is to determine at random a value for each bounded independent variable and evaluate a user defined objective function called FUNCTN at this point. Upon completion of this segment, a current "best" set of values for the independent variables has been found. This point is an estimate of the extremum and serves as the initial search point for the steepest - descent algorithm.

As in the random search segment, the optimal - steepest descent segment minimizes a user defined function that is provided to the main program through the subroutine FUNCTN. All independent variables used by FUNCTN have their values stored as elements of the array X. Also, system parameter values may be stored as elements of X. In this research the form of FUNCTN is the modified ECON cost model (discussed previously) consisting of 169 independent variables and parameters. Specifically, the steps taken in the optimal - steepest descent segment are as follows:

(1) At the point X evaluate numerically the partial derivatives of FUNCTN with respect to the independent variables.

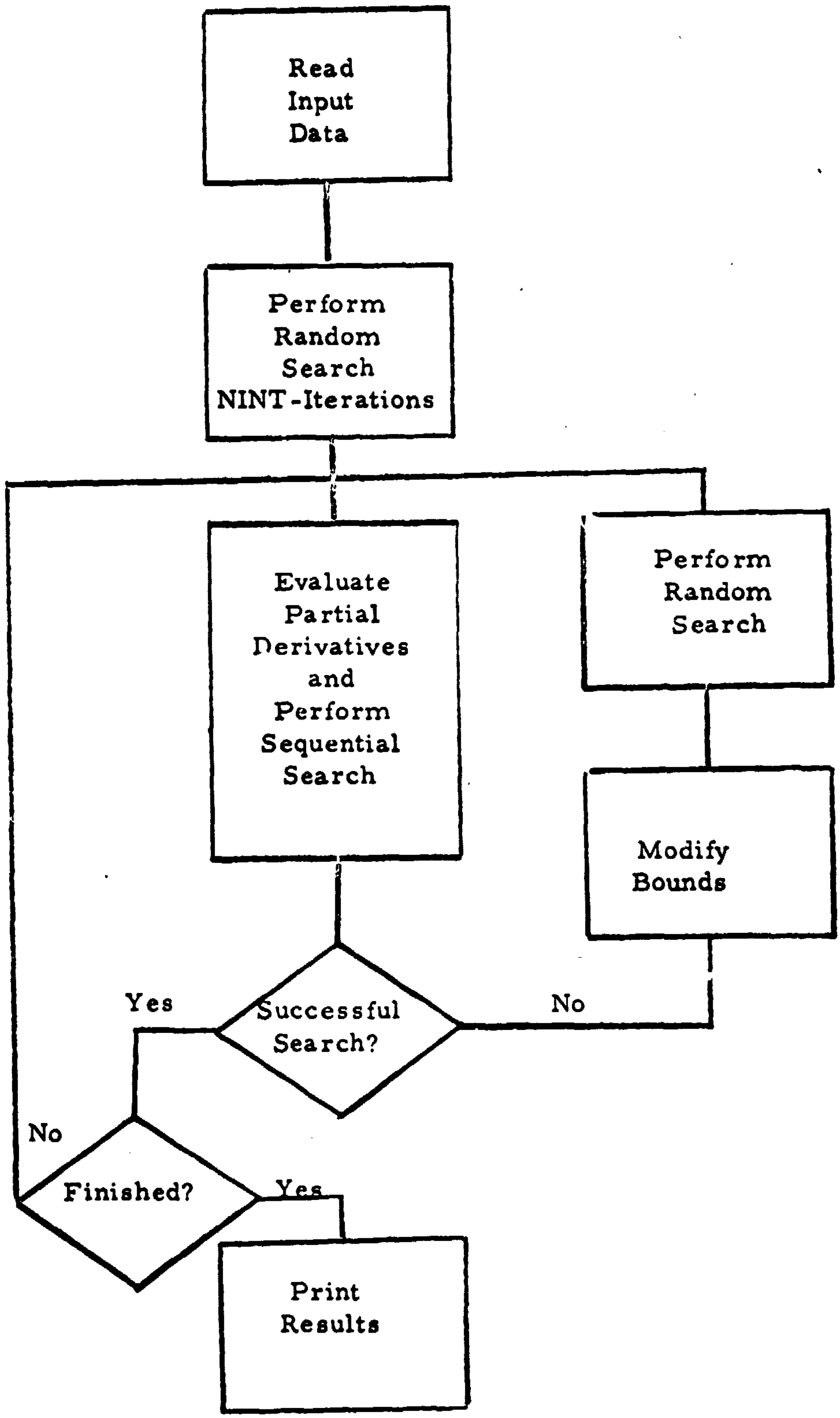


Figure 2.

Simplified Main Program Flow-Chart

(2) Normalize the vector of partial derivatives.

(3) Find a point XNEW in the direction such that the value returned by FUNCTN can be improved.

(4) Perform a one-dimensional search for the point that provides the minimum value of the objective function and lies on the line connecting X and XNEW.

(5) Continue steps 1 through 4 for a specified number (NMAX) of iterations. A complete FORTRAN listing of this program can be found in the Appendix to this report.

Example - Preliminary Results

Decision Variable	Initial Point	Extremum found by Random Search	Extremum found by 100 Point Sequential Search
Total Construction Time (Days)	330	330	330
Time Between Crew Rotation (Days)	90	177	330
Turn Around Time for HLLV (Days)	14	14	14
No. of Personnel that can be carried Per Shuttle Flight	68	55	99
Turn Around Time for Shuttle (Days)	14	19	14
Fraction of Total Satellite Mass to be Assembled by Manned Input	.20	.66	.79
Total System Cost	$69G	$64.3G	$60G

Input Variable Definitions

Variable	Description
IX	Initial random number seed - any odd integer.
KIN	Number of independent variables and parameters, i. e., total number of active X array elements.
KI	Number of independent variables.
NMAX	Total number of optimal - steepest descent iterations.
NINT	Total number of initial random search iterations.
NPRINT	Intermediate printout factor, i. e., print every NPRINT iterations.
X	Array of independent variables and parameters, KIN elements.
INVPT	Array of the subscripts of the independent variables, KI elements. INVPT(L) indicates the location in X of the L^{th} independent variable.
BNDLW	Array of lower bounds of the independent variables. BNDLW (INVPT(L)) is the lower bound of the L^{th} independent variable.
BNDUP	Array of upper bounds of the independent variable with subscripts determined as in BNDLW.
SCALE	Array of scaling factor for the independent variables. Scale (INVPT(L)) should possess a value between zero and one.

ENUF	The minimum improvement in the objective function that is acceptable between successive iterations for the optimal - steepest descent search to continue.

Input Data Cards

Card Type 1

Variable	Columns	Format Type
KIN	1-5	I
NMAX	6-10	I
NINT	11-15	I
IX	16-20	I
NPRINT	21-25	I

Card Type 2

ENUF Punched in E10.6 format.

Card Type 3

X 5 entries per card in E15.8 format and a total of KIN entries.

Card Type 4

KI Number of independent variables pundhed in interger format in columns 1-3.

Card Type 5

INVPT 25 entries per card in I3 format and a total of KI entries.

Card Type 6

BNDUP 5 entries per card in E15.8 format and a total of KI data entries.

Card Type 7

BNDLW 5 entries per card in E15.8 format and a total of KI data entries.

Card Type 8

SCALE 5 entries per card in E15.8 format and a total of KI data entries.

Program Subroutine Descriptions

NEWPT (X, XNEW, DIFF, EPS): Locates a point XNEW (J) a distance EPS(J)*DIFF(J) from X(J) in the direction of DIFF(J), the partial derivative of FUNCTN with respect to X(J).

GRAD (X, XNEW, DIFF, Y): Performs a sequential search using the optimal - steepest descent method on the line joining X and XNEW. Returns to the calling routine the current estimate of the extremum (XNEW) and Y, the value of FUNCTN determined at XNEW.

POINT (X, X1, D, XO): Determines a point (XO) that lies on the line joining the points X and XI. D (O less than D less than 1) and provides a means of locating the point.

RANDU (IX, IY, YFL): Returns a random number, YFL, on the interval between zero and one. IX is the preceding "seed" number and IY is succeeding "seed" number.

INITIAL: Provides a nominal upper and lower bound for all non-independent variables.

FUNCTN (X, DELTA, ICOL, COSTMD): This is the user defined function that is to be optimized. This subroutine operates with two options.

Option 1 - ICOL equals 0. The function is evaluated at the point defined by the array X and the value is returned to the calling routine as COSTMD.

Option 2 - ICOL greater than 0. The function is evaluated at the point X(1), X(2),..., X(ICOL-1), X(ICOL) + DELTA, X(ICOL + 1),..., and the value is returned by COSTMD. This option allows the user to numerically evaluate partial derivatives of FUNCTN with respect to the independent variable represented in array location ICOL.

CONCLUSIONS AND RECOMMENDATIONS

As a result of this research, the following can be concluded:

(1) A systems model describing the transportation and assembly requirements for the construction of a Satellite Power System can take the form of a multidimensional cost function consisting of bounded decision variables.

(2) The characteristics of the decision variables at a "point design" can be analyzed by evaluating the partial derivatives. This information is one method of determining the significant variables and can provide valuable information to system planners and designers.

(3) The controllable variables can be adjusted within the appropriate bounds such that the total system cost can be minimized using a general computerized routine that was written to minimize a nonlinear function in the presence of bounded variables. The procedure uses random and sequential search methods.

It is recommended that future research be directed toward correlation with improved cost models with special attention given to the definition of the interrelationships between system variables and parameters. Further work should include the study of the appropriate systems model using the nonlinear optimization program developed as a result of this research. A logical extension of this research would be the development of an algorithm for the optimization of a nonlinear objective function in the presence of linear constraints.

REFERENCES

1. "Space-Based Solar Power Conversion and Delivery Systems Study," ECON, April 1976.

2. "Satellite Power System," MSFC Status Report, July 1976.

3. Couvant, R., Differential and Integral Calculus, Volumns I and II, Wiley-Interscience, New York, 1936.

4. Gottfried, Byron S. and Weisman, Joel, Introduction to Optimization Theory, Printice-Hall, Englewood Cliffs, N.J., 1973.

5. Hadley, G., Nonlinear and Dynamic Programming, Addison-Wesley, Reading, Mass.: 1964.

6. Taha, Hamdy A., Operations Research an Introduction, Macmillian Company, New York: 1971.

7. Kuhn, H.W. and Tucker, A.W., "Nonlinear Programming," Proceedings of Second Berkley Symposium on Mathematical Statistics and Probability, University of California Press, Berkley, 1951.

8. Wilde, D.J., Optimum Seeking Methods, Prentice Hall, Englewood Cliffs, N.J., 1964.

```
                         3200 FORTRAN (3.0)/RTS

      PROGRAM MAIN
C                 APPENDIX
C             PROGRAM LISTING
      REAL NTEMP
      DIMENSION X(200),DIFF(200),XNEW(200),Y(2) ,NTEMP(200)
      DIMENSION ALTH(200),SCALE(200)
      DIMENSION ALOW(200),XUP(200)
      COMMON KI,KIN,INVPT(200),BNDUP(200),BNDLW(200)
      DATA (DIFF = 200(1.0))
C ****************************************************************************
C ****************************************************************************
C **              DESCRIPTION OF SIGNIFICANT PROGRAM VARIABLES              **
C **          KI - NUMBER OF INDEPENDT VARIABLES                            **
C **               AN UPPER AND LOWER BOUND MUST BE PROVIDED                **
C **          KIN - TOTAL NUMBER OF INDEPENDENT VARIABLES AND PARAMETERS    **
C **          X - ARRAY OF INDEPENDENT VARIABLES  AND PARAMETERS            **
C **          BNDUP - ARRAY OF UPPER BOUNDS FOR INDEPENDENT VARIABLES       **
C **          BNDLW - ARRAY OF LOWER BOUNDS FOR INDEPENDENT VARIABLES       **
C **          INVPT - ARRAY THAT CONTAINS THE SUBSCRIPTS OF THE             **
C **               INDEPENDENT VARIABLES  --   INVPT(J) IS THE              **
C **               SUBSCRIPT OF THE J IND. VARIABLE                         **
C **          IX - INITIAL RANDOM NUMBER SEED    - ANY ODD INTEGER          **
C **          SCALE - ARRAY OF STEP SIZES FOR INDEPENDENT VARIABLES         **
C **               SCALE(J)  - = 1.                                         **
C **          NPRINT - ALLOWS THE USER TO RECIEVE INTERMEDIATE PRINT        **
C **               PRINT OCCURS EVERY NPRINT ITERATIONS                     **
C **          NMAX - MAXIMUM NUMBER OF OPTIMAL - STEEPEST DESCENT ITERATIONS **
C **          NINT - MAXIMUM NUMBER OF INITIAL RANDOM SEARCHES              **
C **          ENUF - WHEN OBJECTIVE FUNCTION IMPROVEMENT IS LESS THAN ENUF STOP
C ****************************************************************************
C ****************************************************************************
      DELTAO = 1.0E06
C                    INPUT SEGMENT
C
      READ 220,KIN,NMAX,NINT,IX,NPRINT
  220 FORMAT(6I5)
  225 FORMAT(8E10.6)
      READ 225,ENUF
  205 FORMAT(5X,5(I6,3X))
      READ 5, (X(I),I=1,KIN)
      CALL INITAL
    5 FORMAT(5E15.8)
      READ 25,KI
      READ 25,(INVPT(I),I=1,KI)
C
      PRINT 1000,KIN,NMAX,NINT,NPRINT,IX
 1000 FORMAT(1H1,9X,28HTOTAL NUMBER OF VARIABLES = ,I6,/,
     A10X,49HMAXIMUM NUMBER OF SEQUENTIAL SEARCH ITERATIONS = ,I6,/,
     B10X,45HMAXIMUM NUMBER OF RANDOM SEARCH ITERATIONS = ,I6,/,
     C10X,25HINTERMEDIATE PRINT EVERY ,I6,10H ITERATION,/,
     D10X,29HINITIAL RANDOM NUMBER SEED = ,I6)
      PRINT 1005,KI
 1005 FORMAT(//,9X,34HNUMBER OF INDEPENDENT VARIABLES = ,I6)
      PRINT 1010
 1010 FORMAT(//,9X,35HSUBSCRIPTS OF INDEPENDENT VARIABLES)
      PRINT 1015,(INVPT(I),I=1,KI)
 1015 FORMAT(10X,20I4)
C
      READ 5,(NTEMP(I),I=1,KI)
      DO 80 I=1,KI
      J=INVPT(I)
      BNDUP(J) = NTEMP(I)
   80 CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
      READ 5,(NTEMP(I),I=1,KI)
      DO 85 I=1,KI
      J=INVPT(I)
      BNDLW(J) = NTEMP(I)
 85   CONTINUE
C
C
C
      READ 5,(NTEMP(I),I=1,KI)
      DO 86 I=1,KI
      J=INVPT(I)
      SCALE(J) = NTEMP(I)
 86   CONTINUE
C
C
 25   FORMAT(25I3)
      CALL FUNCTN(X,0.0,0,YLOW)
      PRINT 300,(JH,X(JH),JH=1,KIN)
      DO 140 I=1,KIN
      XNEW(I) = X(I)
 140  CONTINUE
C                      RANDOM SEARCH SEGMENT
C
      IF(NINT) 2005,2005,2000
 2000 CONTINUE
      DO 115 JLOOP=1,NINT
      DO 120 I=1,KI
      IS = INVPT(I)
 240  CALL RANDU(IX,IY,YFL)
 245  XNEW(IS) = BNDLW(IS) +(BNDUP(IS)-BNDLW(IS))*YFL
 120  CONTINUE
      CALL FUNCTN(XNEW,0.0,1,YNOW)
 400  FORMAT(5X,E15.8,5X,E15.8)
      IF(YNOW-YLOW) 125,135,135
 125  YLOW = YNOW
      DO 130 I=1,KIN
      X(I) = XNEW(I)
 130  CONTINUE
 135  CONTINUE
 115  CONTINUE
 2005 CONTINUE
      NCNT=0
      CALL FUNCTN(X,0.0,0,YO)
      DO 55 I=1,KIN
 55   XNEW(I) = X(I)
      PRINT 300,(JH,X(JH),JH=1,KIN)
C                      OPTIMAL - STEEPEST DESCENT SEGMENT
C
 35   CONTINUE
      NI =0
       NCNT=NCNT+1
      IF(NCNT-NMAX) 100,100,40
 100  SUM=0.
      ZERO =0.
      DIFNRM =0.
      DO 30 I=1,KI
      IS = INVPT(I)
      DELTA =(BNDUP(IS)-BNDLW(IS))/DELTAO
      DELTA1 = (-1.0*DELTA)
      CALL FUNCTN(X,DELTA,IS,D)
      CALL FUNCTN(X,DELTA1,IS,E)
      DIFF(IS) = (D-E)/2.0/DELTA
      ALTH(IS)=0.
```

```
      IF(DIFF(IS)) 420,430,425
  420 XLTH(IS) =(BNDUP(IS)-X(IS))*SCALE(IS)
      XLOW(IS) = X(IS)
      XUP(IS) = XLOW(IS)+XLTH(IS)
      IF(XLTH(IS)-1.0E-05) 430,430,505
  505 DIFNRM = DIFNRM+DIFF(IS)**2
      SUM=SUM+ ABS(DIFF(IS))
      GO TO 515
  425 XLTH(IS) =(X(IS)-BNDLW(IS))*SCALE(IS)
      XLOW(IS) = BNDLW(IS)
      XUP(IS) = XLOW(IS)+XLTH(IS)
      IF(XLTH(IS)-1.0E-05) 430,430,510
  510 DIFNRM = DIFNRM+DIFF(IS)**2
      SUM=SUM+ ABS(DIFF(IS))
      GO TO 515
  430 DIFF(IS) =0.
  515 ZERO = ZERO+XLTH(IS)
   30 CONTINUE
      IF(SUM-1.0E-05) 40,40,45
   45 CONTINUE
      IF(ZERO-0.) 50,50,415
  415 DIFNRM = SQRT(DIFNRM)
      DO 200 I=1,KI
      IS = INVPT(I)
      DIFF(IS) = DIFF(IS)/DIFNRM
  200 CONTINUE
C
C
  410 CALL NEWPNT(X,XNEW,DIFF,XLTH)
      CALL GRAD(X,XNEW,DIFF,ZO)
      DF=ZO-YO
      IF(DF-ENUF) 305,51,51
   51 CONTINUE
      YLOW = YO
      DO 520 JJ=1,KI
      IS = INVPT(JJ)
      CALL RANDU(IX,IY,YFL)
      XNEW(IS) = XLOW(IS) +(XUP(IS)-XLOW(IS))*YFL
      CALL FUNCTN(XNEW,0,0,0,YNOW)
      IF(YNOW-YLOW-ENUF) 530,526,526
  530 ZO = YNOW
      DF=ZO-YO
      GO TO 305
  526 XNEW(IS) =X(IS)
  520 CONTINUE
      DO 525 II=1,50
      DO 540 JJ=1,KI
      IS =INVPT(JJ)
      CALL RANDU(IX,IY,YFL)
      XNEW(IS) = XLOW(IS) +(XUP(IS)-XLOW(IS))*YFL
  540 CONTINUE
      CALL FUNCTN(XNEW,0,0,0,YNOW)
      IF(YNOW-YLOW-ENUF) 530,525,525
  525 CONTINUE
      GO TO 50
  305 IF(NCNT/NPRINT*NPRINT-NCNT) 210,215,210
  215 PRINT 95,NCNT,DF,YO
   95 FORMAT(5X,I5,5X,E15.8,5X,E15.8)
  210 IF(ABS(YO-ZO)-ENUF    ) 50,50,54
   54 YO = ZO
      DO 53 IB =1,KIN
   53 X(IB) = XNEW(IB)
      IF(NCNT/NPRINT*NPRINT-NCNT) 165,160,165
  165 CONTINUE
   75 GO TO 35
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
160   CONTINUE
      PRINT 500,(JH,X(JH),DIFF(JW),JH=1,KIN)
500   FORMAT(3(2X,I5,2X,E15.8,2X,E15.8))
300   FORMAT(5(3X,I5,2X,E15.8))
170   FORMAT(6(5X,E15.8))
      GO TO 165
40    CONTINUE
50    CONTINUE
C **** OUTPUT ****
      PRINT 1020
1020 FORMAT(1H1,3X,6HNUMBER,6X,5HVALUE,12X,11HLOWER BOUND,
    A9X,11HUPPER BOUND,9X,18HPARTIAL DERIVATIVE)
      PRINT 1025,NCNT,YC
1025 FORMAT(//,9X,17HITERATION NUMBER ,I5,2X,
    A19HOBJECTIVE FUNCTION ,E15.8)
      DO 10 I=1,KIN
      PRINT 15,I,X(I),BNDLW(I),BNDUP(I),DIFF(I)
15    FORMAT(/,5X,I5,4(5X,E15.8))
10    CONTINUE
      PRINT 1025,NCNT,YO
20    CONTINUE
      STOP
      END
```

3200 FORTRAN (3.0)/RTS / /

```
      SUBROUTINE POINT(X,X1,D,XO)
      COMMON KI,KIN,INVPT(200),BNDUP(200),BNDLW(200)
      DIMENSION X(200),X1(200),XO(200)
      DO 10 I=1,KIN
10    XO(I) = X(I)
      DO 5 I=1,KI
      J = INVPT(I)
      XO(J) =X(J)*D +(1.0-D)*X1(J)
5     CONTINUE
      RETURN
      END
```

3200 FORTRAN (3.0)/RTS / /

```
      SUBROUTINE NEWPNT(X,XNEW,DIFF,EPS)
      DIMENSION X(200),XNEW(200),DIFF(200)
      DIMENSION EPS(200)
      COMMON KI,KIN,INVPT(200),BNDUP(200),BNDLW(200)
      DO 5 I=1,KI
      J = INVPT(I)
      XNEW(J) = X(J) -EPS(J)*DIFF(J)
      IF(XNEW(J)-BNDUP(J)) 15,15,10
10    XNEW(J) = BNDUP(J)
      GO TO 5
15    IF(XNEW(J)-BNDLW(J)) 20,5,5
20    XNEW(J) = BNDLW(J)
5     CONTINUE
30    RETURN
      END
```

```
      SUBROUTINE GRAD(X,XNEW,DIFF,Y)
      COMMON KI,KIN,INVPT(200),BNDUP(200),BNDLW(200)
      DIMENSION X(200),XNEW(200),DIFF(200),XPT(200),XO(200)
      THL=0.
      THR=1.0
      CALL FUNCTN(X,0.0,1,YLFT)
      CALL FUNCTN(XNEW,0.0,1,YRT)
      XL=1.0
      THL1=THR-XL/1.618
      THR1=THL+XL/1.618
      DO 5 KI=1,14
      CALL POINT(X,XNEW,THR1,XPT)
      CALL FUNCTN(XPT,0.0,1,YR1)
      CALL POINT(X,XNEW,THL1,XPT)
      CALL FUNCTN(XPT,0.0,1,YL1)
      IF(YL1-YR1) 10,10,15
10    XL=THR1-THL
      THR = THR1
      THR1 = THL1
      THL1 = THR-XL/1.618
      GO TO 5
15    XL = THR-THL1
      THL = THL1
      THL1 = THR1
      THR1 = THL + XL/1.618
5     CONTINUE
      DO 50 IO=1,KIN
      XO(IO) = X(IO)
50    CONTINUE
      THMIN = (THL1+THR1)/2.
      CALL POINT(X,XNEW,THMIN,XO)
      DO 55 IO=1,KIN
      XNEW(IO) = XO(IO)
55    CONTINUE
      CALL FUNCTN(XNEW,0.0,0,Y)
      RETURN
      END
```

3200 FORTRAN (3.0)/RTS

```
      SUBROUTINE RANDU(IX,IY,YFL)
      IY = IX*35567
      IF(IY) 5,10,10
5     IY = IY+8388607 +1
10    YFL =IY
      YFL = YFL/8388608.
      IX = IY
      RETURN
      END
```

```
      SUBROUTINE FUNCTN (X,DELTA,ICOL,COSTMD)
      DIMENSION X(200)
      COMMON NI,NIN,INVPT(200),BNDUP(200),BNDLW(200)
      IF(ICOL) 100,100,110
C ******************************************************************************
C ******************************************************************************
C **                         DIRECTORY OF VARIABLES                           **
C **                                                                          **
C **                                                                          **
C **        X(1)        = ALCT                                                **
C **                      RATIO OF TOTAL INITIAL-TO-FINAL MASS OF THE         **
C **                      LARGE CRYO PLUS CREW MODULE                         **
C **        X(2) = DELTA VLCT = TOTAL LEO-GEO MISSION DELTA V (M/SEC)         **
C **        X(3) = VJLCT = ROCKET EXHAUST JET VELOCITY (M/SEC)                **
C **        X(4) = M LCT PROP = MASS OF CRYO PROPELLANTS REQUIRED FOR         **
C **               ONE ROUND-TRIP TO GEO (KG)                                 **
C **        X(5) = LAMDA LCT = PROPELLANT MASS-FRACTION OF THE CRYO TUG       **
C **        X(6) = MLCT = MASS OF THE LARGE CRYO TUG (DRY) (KG)               **
C **        X(7) = MLCT PROP (BIG M) = TOTAL MASS OF CRYO PROPELLANTS         **
C **               USED DURING THE CONSTRUCTION OF ONE SSPS (KG)              **
C **        X(8) = TCONST = TOTAL CONSTRUCTION TIME (DAYS)                    **
C **        X(9) = TRCT = TIME PERIOD BETWEEN CREW ROTATIONS (DAYS)           **
C **        X(10) = ALPHA AIS = RATIO OF TOTAL INITIAL-TO-FINAL MASS OF THE   **
C **               ADVANCED ION STAGE + PAYLOAD                               **
C **        X(11) = DELTA AIS = TOTAL LEO-GEO MISSION DELTA V OF THE          **
C **               ION STAGE (M/SEC)                                          **
C **        X(12) = VJ AIS = EXHAUST JET VELOCITY OF THE ION STAGE (M/SEC)    **
C **        X(13) = M AIS PROP = TOTAL MASS OF ION PROPELLANT (KG)            **
C **        X(14) = M GEO S/S = MASS OF GEO SPACE STATION (KG)                **
C **        X(15) = M TOT SAT = TOTAL MASS OF THE OPERATIONAL SATELLITE (KG)  **
C **        X(16) = LAMDA AIS = PROPELLANT MASS-FRACTION OF THE ION STAGE     **
C **        X(17) = M AIS = TOTAL MASS OF THE ION STAGE (DRY) (KG)            **
C **        X(18) = M PROP DEPOT = TOTAL MASS OF THE TANKS USED AS A          **
C **               PROPELLANT DEPOT IN LOW-EARTH ORBIT (KG)                   **
C **        X(19) = MHT = MASS OF SINGLE LIQUID HYDROGEN TANK (KG)            **
C **        X(20) = MLH = TOTAL MASS OF LIQUID HYDROGEN TO BE STORED(KG)      **
C **        X(21) = CHT = CAPACITY OF A LIQUID HYDROGEN STORAGE TANK CKG)     **
C **        X(22) = M LOXT = MASS OF A SINGLE LIQUID OXYGEN STORAGE TANK (KG) *
C **        X(23) = M LOX = TOTAL MASS OF LIQUID OXYGEN TO BE STORED (KG)     **
C **        X(24) = C LOXT = CAPACITY OF A LIQUID OXYGEN STORAGE TANK         **
C **        X(25) = M IT = MASS OF A SINGLE ION PROPELLANT STORAGE TANK       **
C **        X(26) = CIT = CAPACITY OF SINGLE ION PROPELLANT STORAGE TANK (KG**
C **        X(27) = M UMAE = TOTAL MASS OF UNMANNED ASSEMBLY EQUIPMENT (KG)   **
C **        X(28) = M FAB = TOTAL MASS OF THE FABRICATION MODULES (KG)        **
C **        X(29) = M TELE = TOTAL MASS OF THE TELEOPERATORS (KG)             **
C **        X(30) = M AE PROP = TOTAL MASS OF ASSEMBLY EQUIPMENT PROPELLANT   **
C **        X(31) = M TUG = TOTAL MASS OF THE LEO SUPPORT TUGS (KG)           **
C **        X(32)= M MAE = TOTAL MASS OF THE MANNED ASSEMBLY EQUIPMENT (KG)   **
C **        X(33) = M EVA = TOTAL MASS OF THE EVA EQUIPMENT (KG)              **
C **        X(34) = M MANIP = TOTAL MASS OF THE MANNED MANIPULATORS (KG)      **
C **        X(35) = M LEO S/S = TOTAL MASS OF THE LEO SPACE STATIONS (KG)     **
C **        X(36)= M GEO S/S =  TOTAL MASS OF THE GEO SPACE STATIONS (KG)     **
C **        X(37) = M S/S RES = TOTAL MASS OF SPACE RESUPPLY (KG)             **
C **        X(38) = M IOVP = TOTAL MASS OF THE INTER-ORBIT                    **
C **               VEHICLES AND PROPELLANT (KG)                               **
C **        X(39) = M CREW = MASS OF THE CREW MODULES (KG)                    **
C **        X(40) = M LEO = TOTAL MASS LAUNCHED TO LEO FOR THE                **
C **               CONSTRUCTION OF ONE SSPS (KG)                              **
C **        X(41) = N HLLV = TOTAL NUMBER OF HEAVY LIFT LAUNCH                **
C **               VEHICLE FLIGHTS .                                          **
C **        X(42) = M P/L = THE PAYLOAD TO LEO OF AN HLLV (KG)                **
C **        X(43) = F LOAD = AVERAGE LOAD FACTOR FOR AN HLLV                  **
C **               (WHAT PERCENTAGE OF PAYLOAD IS USED)                       **
```

```
C **          X(44) - N H UNITS = NUMBER OF HLLV UNITS ACQUIRED FOR             **
C **                  THE CONSTRUCTION OF ONE SSPS                              **
C **          X(45) - T H TURN = TURN AROUND TIME FOR EACH HLLV UNIT (DAYS)     **
C **          X(46) - N SHUTTLE = TOTAL NUMBER OF SHUTTLE FLIGHTS               **
C **          X(47) - N LEO = TOTAL NUMBER OF LOW-EARTH ORBIT PERSONNEL         **
C **          X(48) - N GEO = TOTAL NUMBER OF GEO PERSONNEL                     **
C **          X(49) - F SHUTTLE = NUMBER OF PERSONNEL THAT CAN BE               **
C **                  CARRIED PER SHUTTLE FLIGHT                                **
C **          X(50) - N S UNITS = TOTAL NUMBER OF SHUTTLES ACQUIRED             **
C **          X(51) - T S TURN = TURN AROUND TIME OF EACH SHUTTLE (DAYS)        **
C **          X(52) - M MANNED = TOTAL MASS OF SATELLITE TO BE                  **
C **                  CONSTRUCTED BY ON-ORBIT PERSONNEL (KG)                    **
C **          X(53) - BETA = PERCENTAGE OF TOTAL SATELLITE MASS TO BE           **
C **                  ASSEMBLED BY MAN INPUT                                    **
C **          X(54) - R MANNED = RATE OF MANNED ASSEMBLY (KG/MAN-DAY)           **
C **          X(55) - T MANNED = TOTAL MAN-DAYS OF CONSTRUCTION TIME            **
C **          X(56) - F S = NUMBER OF SHIFTS PER DAY                            **
C **          X(57) - F M = FACTOR OF PRODUCTIVITY ACCOUNT FOR                  **
C **                  OPERATIONS IN SPACE                                       **
C **                  (PRODUCTIVE TIME/ TOTAL WORK TIME)                        **
C **          X(58) - C HLLV (BIG C) = TOTAL COST OF HLLV ACTIVITY              **
C **          X(59) - C HLLV = COST PER HLLV FLIGHT (OPERATIONS)                **
C **          X(60) - C H UNIT = COST PER HLLV UNIT                             **
C **          X(61) - C SHUTTLE (BIG C ) = TOTAL COST OF SHUTTLE ACTIVITY       **
C **          X(62) - C SHUTTLE = COST PER SHUTTLE FLIGHT (OPERATIONS)          **
C **          X(63) - C S UNIT = COST PER SHUTTLE UNIT                          **
C **          X(64) - C LLC = TOTAL LOW- EARTH ORBIT LAUNCH COST                **
C **                                                                            **
C **                                                                            **
C **                                                                            **
C **          X(65) - C UMAE = TOTAL COST OF UNMANNED ASSEMBLY EQUOPMENT        **
C **          X(66) - VALUE NOT USED                                            **
C **          X(67) - C FAB = UNIT COST OF FABRICATION MODULE ($)               **
C **          X(68) - N FAB = NUMBER OF FABRICATION MODULES                     **
C **          X(69) - D FAB = DESIGN LIFE OF FABRICATION MODULE (DAYS)          **
C **          X(70) - C TELE = UNIT COST OF TELEOPERATOR ($)                    **
C **          X(71) - N TELE = NUMBER OF TELEOPERATORS                          **
C **          X(72) - D TELE = DESIGN LIFE OF TELEOPERATOR (DAYS)               **
C **          X(73) - C AE PROP = SPECIFIC COST OF ASSEMBLY EQUIMENT            **
C **                  PROPELLENT ($/ KG)                                        **
C **          X(74) - C TUG = UNIT COST OF LEO SUPPORT TUG($)                   **
C **          X(75) - N TUG = TOTAL NUMBER OF SUPPORT TUGS                      **
C **          X(76) - D TUG = DESIGN LIFE OF LEO SUPPORT TUG ($)                **
C **          X(76) - C GRD OP = COST PER GROUND OPERATOR ($)                   **
C **                  ( FOR TELEOPERATORS)                                      **
C **          X(78) - F GRD = NUMBER OF SHIFTS FOR GROUND OPERATORS             **
C **          X(79) - C MAE = TOTAL COST OF MANNED ASSEMBLY EQUIPMENT ($)       **
C **          X(80) - C EVA = UNIT COST OF EVA EQUIPMENT ($)                    **
C **          X(81) - F EVA = FACTOR TO ACCOUNT FOR WHETHER OR NOT EVA UNITS    **
C **                  MUST BE TAILORED TO INDIVIDUALS OR CAN BE                 **
C **                  USED REPETITIVELY AND FOR HOW LONG                        **
C **          X(82) - C MANIP = UNIT COST OF MANNED MANIPULATOR ($)             **
C **          X(83) - N MANIP = TOTAL NUMBER OF MANNED MANIPULATORS             **
C **          X(84) - D MANIP = DESIGN LIFE FOR MANNED MANIPULATOR              **
C **          X(85) - F MANIP = FACTOR TO ACCOUNT FOR MANIPULATOR DOWNTIME      **
C **                  ( I. E) . THE PERCENTAGE OF TIME THE UNITS ARE            **
C **                  AVAILABLE)                                                **
C **          X(86) - M MANIP = MASS OF A SINGLE MANNED MANIPULATOR (KG)        **
C **          X(87) - N LEO S/S = TOTAL NUMBER OF LEO SPACE STATIONS            **
C **          X(88) - F LEO S/S = NUMBER OF PERSONNEL THAT CAN BE HOUSED IN     **
C **                  EACH STATION                                              **
C **          X(89) - D LEO S/S = DESIGN LIFE OF A LEO SPACE STATION (DAYS)     **
C **          X(90) - C LEO S/S = UNIT COST OF LEO SPACE STATION ($)            **
C **          X(91) - C GEO S/S = UNIT COST OF GEO SPACE STATION ($)            **
C **          X(92) - M LEO S/S = MASS OF A SINGLE LEO STATION (KG)             **
```

```
C **        X(93)  - N GEO S/S = TOTAL NUMBER OF GEO SPACE STATIONS                **
C **        X(94)  - D GEO S/S = DESIGN LIFE OF GEO STATION (DAYS)                 **
C **        X(95)  - C S/S RES = SPECIFIC COST OF SPACE STATION RESUPPLY ($)       **
C **        X(96)  - C S/S + A = TOTAL COST OF SPACE STATIONS AND ASSEMBLY         **
C **                 FOR ONE SPS ($)                                               **
C **        X(97)  - C LEO-GEO = TOTAL COST OF LEO - GEO TRANSPORTATION            **
C **        X(98)  - C LCT =UNIT COST OF LARGE CRYO TUG ($/KG)                     **
C **        X(99)  - C AIS = UNIT COST OF ADVANCED ION STAGE ($/KG)                **
C **        X(100) - C LCT PROP = SPECIFIC COST OF CRYO TUG PROPELLANT($/KG)       **
C **        X(101) - C AIS PROP = SPECIFIC COST OF ION PROPELLANTS($/KG)           **
C **        X(102) - C CREW = UNIT COST OF CREW MODULE ($)                         **
C **        X(103) - D CREW = DESIGN LIFE OF CREW MODULE                           **
C **        X(104) - C LHT = UNIT COST OF LIQUID HYDROGEN STORAGE TANK ($)         **
C **        X(105) - C LOXT = UNIT COST OF LIQUID OXYGEN STORAGE TANK ($)          **
C **        X(106) - C IT = UNIT COST OF ION PROPELLANT STORAGE TANK ($)           **
C **        X(107) - D LOXT = DESIGN LIFE OF LOX STORAGE TANK (DAYS)               **
C **        X(108) - C IT = DESIGN LIFE OF ONE ION PROP STORAGE TANK               **
C **        X(109) - D LCT = DESIGN LIFE OF CRYO TUG (DAYS)                        **
C **        X(110) - DESIGN LIFE OF THE ION STAGE (DAYS)                           **
C **        X(112) - C UPC = TOTAL UNIT PRODUCTION COST                            **
C **                                                                               **
C **                                                                               **
C **                                                                               **
C **        X(113) - M REMOTE = TOTAL MASS OF SATELLITE TO BE CONSTRUCTED          **
C **                 BY REMOTE CONSTRUCTION                                        **
C **        X(114) - R REMOTE = RATE OF REMOTE CONTROLLED ASSEMBLY                 **
C **                 (KG/ MACHINE DAY)                                             **
C **        X(115) - T REMOTE = TOTAL MACHINE DAYS OF CONSTRUCTION TIME            **
C **        X(116) - F TELE AV = FACTOR TO ACCOUNT FOR DOWNTIME OF                 **
C **                 TELEOPERATORS                                                 **
C **        X(117) - F T = FACTOR TO ACCOUNT FOR PERCENTAGE OF TIME THAT           **
C **                 TELEOPERATORS CAN BE DOING USEFUL                             **
C **        X(118) - F FAB = FACTOR TO ACCOUNT FOR FABRICATION MODULE              **
C **                 DOWNTIME                                                      **
C **        X(119) - R FAB = RATE OF FABRICATION MODULES (KG/DAYS)                 **
C **        X(120) - M FAB = MASS OF A SINGLE FABRICATION MODULE (KG)              **
C **        X(121) - M TELE = MASS OF A SINGLE TELEOPERATOR                        **
C **        X(122) - M TUG = MASS OF A SINGLE LEO SUPPORT TUG (KG)                 **
C **        X(123) - M EVA = MASS OF A SINGLE EVA UNIT (KG)                        **
C **        X(124) - F EVA FACTOR TO ACCOUNT FOR WHETHER OR NOT EVA UNITS          **
C **                 MUST BE TAILORED TO INDIVIDUALS                               **
C **        X(125) - CF = CONTIGENCY FACTOR                                        **
C **        X(126) - F DEG = FACTOR TO ACCOUNT FOR BLANKET                         **
C **                 DEGRADATION DURING ORBITAL TRANSFER                           **
C **        X(127) - C ANT = TOTAL PROCUREMENT COST OF                             **
C **                 TRANSMITTING ANTENNA ($)                                      **
C **        X(128) -C PD = SPECIFIC COST OF ANTENNA POWER                          **
C **                 DISTRIBUTION ($ / KW)                                         **
C **        X(129) - C PC = SPECIFIC COST OF PHASE CONTROL ($/KW)                  **
C **        X(130) - C WG = SPECIFIC COST OF WAVEGUIDE ($/KW)                      **
C **        X(131) - C DC-RF = SPECIFIC COST OF DC-RF CONVERTORS ($/KW)            **
C **        X(132) - C ST = SPECIFIC COST OF ANTENNA STRUCTURE ($/KW)              **
C **        X(133) - C SAT = TOTAL PROCUREMENT COST OF AN                          **
C **                 OPERATIONAL SATELLITE ($)                                     **
C **        X(134) - C SAB = SPECIFIC COST OF SOLAR ARRAY BLANKET($/KM**2)         **
C **        X(135) - C SAC = SPECIFIC COST OF SOLAR CONCENTRATOR ($/KM**2)         **
C **        X(136) - C STC = SPECIFIC COST OF CONDUCTING STRUCTURE ($/KG)          **
C **        X(137) - C STNC = SPECIFIC COST OF NON - CONDUCTING                    **
C **                 STRUCTURE ($/KG)                                              **
C **        X(138) - C STCM = SPECIFIC COST OF CENTRAL MAST ($/KG)                 **
C **        X(139) - C MISC = SPECIFIC COST OF MISCELLANEOUS                       **
C **                 EQUIPMENT ($/KG)                                              **
C **        X(140) - C GRD STAT = TOTAL PROCUREMENT COST OF THE                    **
C **                 GROUND STATION ($)                                            **
C **        X(141) - C RE = SPECIFIC COST OF REAL ESTATE AND SITE                  **
```

```
C **                    PREPARATION ($/KW)                                       **
C **          X(142) = C STRUCT = SPECIFIC COST OF RECTENNA                     **
C **          X(143) = C RF-DC = SPECIFIC COST OF RF-DC CONVERTERS ($/KW)       **
C **          X(144) = SPECIFIC COST OF POWER INTERFACE ($/KW)                  **
C **          X(145) = C PC = SPECIFIC COST OF PHASE FRONT CONTROL ($/KW)       **
C **          X(146) = M ANTS = TOTAL MASS OF THE ANTENNA STRUCTURE (KG)        **
C **          X(147) = A B = AREA OF SOLAR BLANKET (KM **2)                     **
C **          X(148) = TOTAL MASS OF THE DC-RF CONVERTERS (KG)                  **
C **          X(149) = P = POWER OUTPUT AT THE RECTENNA BUSBAR (KW )            **
C **                    (BEGINNING OF LIFE, B. O. L.)                           **
C **          X(150) = M WG = TOTAL MASS OF THE WAVEGUIDES                      **
C **          X(151) = M SAB = TOTAL MASS OF THE SOLAR BLANKET (KG)             **
C **          X(152) = M ANT-INT = TOTAL MASS OF THE ANTENNA INTERFACE          **
C **          X(153) = A C = AREA OF SOLAR CONCENTRATOR AS SEEN BY THE SUN (    **
C **                    KM **2)                                                 **
C **          X(154) = M PCE = TOTAL MASS OF THE PHASE CONTROL ELECTRONICS (KG**
C **          X(155) = M ANT = TOTAL MASS OF THE ANTENNA                        **
C **          X(156) = M MISC = TOTAL MASS OF MISCELLANEOUS COMPONENTS          **
C **          X(157) = BETA = PERCENTAGE OF TOTAL SATELLITE MASS TO BE          **
C **                    ASSEMBLED BY MAN INPUT                                  **
C **          X(158) = M SAC = TOTAL MASS OF THE SOLAR CONCENTRATOR)            **
C **          X(159) = M STC = TOTAL MASS OF THE CONDUCTING STRUCTURE(KG)       **
C **          X(160) = M STNC = TOTAL MASS OF THE NON-CONDUCICTING STRUCTURE    **
C **          X(161) = M STCM = TOTAL MASS OF THE CENTRAL MAST (KG)             **
C **          X(162) = N MW = MICROWAVE EFFICIENCY                              **
C **          X(163) = N DC-RF = DC-RF CONVERTER EFFICIENCY                     **
C **          X(164) = N PC = PHASE CONTROL EFFICIENCY                          **
C **          X(165) = N ION PROP = IONOSPHERIC PROPAGATION EFFICIENCY          **
C **          X(166) = N ATM PROP = ATMOSPHERIC PROPAGATION EFFICIENCY          **
C **          X(167) = N BC = BEAM COLLECTION EFFICIENCY                        **
C **          X(168) = N RF-DC = RF-DC CONVERTER EFFICIENCY                     **
C **          X(169) N RECT PD = RECTENNA POWER DISTRIBUTION EFFICIENCY         **
C ****************************************************************************
C ****************************************************************************
 110  Z = X(ICOL)
      X(ICOL) = X(ICOL) + DELTA
 100  CONTINUE
C
C
C
C
C
C
C        ECON ROUTINE TC SIZE ONE SPS
C
C
      X(162) = 1.
      DC 10 I=163,169
      X(162) = X(162)*X(I)
  10  CONTINUE
C
      X(155) = X(146) + X(148) +X(150)+ X(152) +X(154)
C
C
       X(15) = X(151)+X(158)+X(159)+X(160)+X(161)+X(156)+X(155)
      X(15) =X(15)*X(125)
C
C
      X(52) = X(53)*X(15)
      X(113) = X(15)*(1.-X(53))
C
      X(55) = X(52)/X(54)
C
      X(47) = (X(55)*X(56))/(X(8)*X(57))
       J=X(47)
      X(47) = J+1.
C
```

```
C
      X(115) =X(113)/X(114)
C
      X(71) = X(115)/X(H)/X(116)/X(117)
C
      X(68) = (X(159)+X(160)+X(150))/X(119)/X(118)/X(8)
C
      X(83) = X(47)/X(56)*X(85)
C
      X(34) = X(83)*X(86)
C
      X(87) = X(47)/X(88)
      J=X(87) +1.
      X(87) = J
C
      X(35) = X(87)*X(92)
C
      X(28) = X(68)*X(120)
C
      X(29) = X(71)*X(121)
C
      X(31) = X(75) * X(122)
C
      X(33) = X(124)*X(123)*(X(47)+X(48))
C
      X(1) = EXP(X(2)/X(3))
C
      X(4) = X(5)*(X(1)-1.)/(X(5)-(X(1)-1.)*(1.-X(5)))
C
      X(6)= X(4)*(1.-X(5))/X(5)
C
      X(7)= X(4)*X(8)/X(9)
C
      X(10)= EXP(X(11)/X(12))
C
      X(13)= (X(14)+X(15))*X(16)*(X(10)-1.)/(X(16)-(X(10)-1.)
     A*(1.-X(16)))
C
      X(17)= X(13)*(1.-X(16))/X(16)
C
      X(20)= X(7)/9.
C
      X(23)= X(7)*(8./9.)
C
      X(18)= X(19)*X(20)/X(21)+X(22)*X(23)/X(24)+X(25)*X(13)/X(26)
C
      X(30)= .01*X(15)
C
      X(27)= X(28)+X(29)+X(30)+X(31)
C
      X(37)= .043*X(15)
C
      X(32)= X(33)+X(34)+X(35)+X(36)+X(37)
C
      X(38)= X(6)+X(17)+X(7)+X(13)+X(39)+X(18)
C
      X(40)= X(27)+X(32)+X(38)+X(15)
C
      X(41) = X(40)/(X(42)*X(43))
C
      X(44) = X(41)*X(45)/X(8)
C
      X(46) = (X(47)+X(48))*X(8)/X(9)/X(49)
C
      X(50) = X(46)*X(51)/X(8)
```

```
C
      X(58) = X(59)*X(41)*X(60)*X(44)
C
      X(62) =9.0E06+X(49)*.25E06
      X(63) =1.425E08+X(49)*.04E08
      X(61) = X(62)*X(46) + X(63)*X(50)
C
      X(64) = X(58) *X(61)
C
      X(65) = X(67)*X(68)*X(8)/X(69)                 +
     A X(73)*X(30) +X(74)*X(75)*X(8)/X(76) + X(77)*X(71)*X(78)
C
      X(79) = X(80)*(X(47)+X(48))*X(81) +X(82)*X(83)*X(8)/X(84) +
     A X(90)*X(87)*X(8)/X(89) + X(91)*X(93)*X(8)/X(89) +X(95)*X(37)
C
      X(96) = X(65)+X(79)
C
      X(97) = X(98)*X(6)*X(8)/X(109) + X(99)*X(17)*X(8)/X(110)
     A   X(100)*X(7) +X(102)*X(8)/X(103)
     B   + X(104)*(FIXT(X(20)/X(21))+1.)*X(8)/X(111)
     C   + X(105)*(FIXT(X(23)/X(24))+1.0)*X(8)/X(107)
     D   + X(106)*X(13)/X(26)*X(8)/X(108)
C
      SUM =0.
      DO 150 I=128,132
      SUM = SUM+X(I)
  150 CONTINUE
       X(127) = SUM*X(149)/X(144)
C
      X(133) = X(134)*X(147)+X(135)*X(153)+X(136)*X(159)+
     AX(160)*X(137)+X(138)*X(161)+X(127)+X(139)*X(150)
C
      SUM =0.
      DO 155 I=141,145
      SUM = SUM+X(I)
  155 CONTINUE
      X(140) = SUM*X(149)
C
      X(112) = X(64) +X(97) +X(96)+ X(133) +X(140)
C
      COSTMD = X(112)
      IF(ICOL) 119,119,120
  120 X(ICOL) = 4
  119 RETURN
      END


      SUBROUTINE INITAL
      COMMON KI,KIN,INVPT(200),BNDUP(200),BNDLW(200)
      DO 5 I=1,KIN
      BNDUP(I) = 1.0E100
      BNDLW(I) = 0.0
    5 CONTINUE
      RETURN
      END
```