ornl

ORNL/TM-10714

OAK RIDGE
NATIONAL
LABORATORY

MARTIN MARIETTA

ORMONTE: An Uncertainty Analysis Code for Use with User-Developed Systems Models on Mainframe or Personal Computers

A User's Guide

K. A. Williams
C. R. Hudson II

OAK RIDGE NATIONAL LABORATORY
CENTRAL RESEARCH LIBRARY
CIRCULATION SECTION
4500N ROOM 175
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON
If you wish someone else to see this report, send in name with report and the library will arrange a loan.

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A10 Microfiche A01

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/TM-10714
Dist. Category UC-05 & UC-32

Contract No. DE-AC05-84OR24100

Engineering Technology Division

ORMONTE: AN UNCERTAINTY ANALYSIS CODE FOR USE WITH USER-DEVELOPED SYSTEMS MODELS ON MAINFRAME OR PERSONAL COMPUTERS

A USER'S GUIDE

K. A. Williams
C. R. Hudson II

Date Published - May 1989

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831
operated by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY

MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES

3 4456 0299506 1

CONTENTS

	Page
LIST OF FIGURES	vii
LIST OF TABLES	ix
ACKNOWLEDGEMENTS	xi
ABSTRACT	1
1. INTRODUCTION	2
1.1 References	3
2. RATIONALE FOR UNCERTAINTY/RISK ANALYSIS	4
2.1 Deterministic Projections	4
2.2 Probabilistic Projections	4
2.3 References	5
3. BACKGROUND OF THE MONTE CARLO UNCERTAINTY ANALYSIS METHODOLOGY	7
3.1 Historical Development	7
3.2 Use of Monte Carlo Based Uncertainty Analysis in Economic Evaluations Performed at Oak Ridge	7
3.3 References	8
4. BRIEF DESCRIPTION OF THE MONTE CARLO TECHNIQUE FOR MULTI-VARIABLE UNCERTAINTY EVALUATION	10
4.1 Mathematical Description of Computational Procedure	10
4.2 Nature of Output from a Monte Carlo MVSS Run	13
4.2.1 The generation of output histograms	13
4.3 Special Features of the ORMONTE Code	15
5. DETAILED DESCRIPTION OF THE ORMONTE CODE	17
5.1 Basic Coding Structure	17
5.2 The MAIN Program	17
5.3 Description of Data Entry Subroutines Utilized for All Three Sensitivity Analysis Options	20
5.3.1 SUBROUTINE READOP	20
5.3.2 SUBROUTINE RDMAST	21
5.3.3 SUBROUTINE WRMAST	23
5.3.4 SUBROUTINE RDMODL: A user-provided subroutine	23
5.4 SUBROUTINE MODEL: A User-Provided Routine	23
5.5 The SVSS Driver: SUBROUTINE SVSSOP	30
5.5.1 Preparation of SVSSOP input data	31
5.5.2 SVSSOP algorithms	34
5.6 A Driver for Elasticity Analysis: SUBROUTINE ELAST	34
5.6.1 Inputs to the ELAST driver	37
5.6.2 Brief description of the ELAST driver algorithms	37

CONTENTS (continued)

			Page
	5.7	SUBROUTINE MVSSOP: The Monte Carlo Driver for Multivariable Uncertainty Analysis	39
		5.7.1 SUBROUTINE RDMVSS	40
		5.7.2 Input distribution descriptions and parameters	45
		5.7.3 SUBROUTINE PRINTX	48
		5.7.4 SUBROUTINE XVALUE: Calculation of return values from the input distribution parameters and the random numbers	48
		5.7.5 Use of MVSSOP for statistical analysis and display of MVSS results	53
6.	SPECIAL CONSIDERATIONS WHEN CONSTRUCTING A MODEL TO BE DRIVEN BY ORMONTE		56
	6.1	Variable Naming	56
	6.2	Internal Loops	56
	6.3	Independence of Input Variables	56
	6.4	Reoptimization within a Model	56
	6.5	Reduction of Output	57
7.	ACQUISITION AND CHARACTERIZATION OF INPUT DATA		58
	7.1	Some DOE Experience-based Guidelines to Assist in the Selection of Numerical Input MVSS Values	58
	7.2	PDA (Probability Data Analysis) Software for Analysis and Characterization of Raw Data	60
		7.2.1 Sources of data	60
		7.2.2 Characterization of the input data	63
	7.3	References	73
8.	INTERPRETATION OF THE OUTPUT STATISTICS AND HISTOGRAMS		75
	8.1	Understanding the Output of MVSSOP	75
	8.2	Location of the Best-Guess Deterministic Figure of Merit Estimate Relative to the Probabilistic Figure of Merit Spectrum	75
	8.3	A Caution Regarding Mode Values in Output Histograms	78
9.	INSTALLATION OF ORMONTE		79
	9.1	Installation of Sensitivity Analysis Program	79
		9.1.1 Installation and execution of ORMONTE on a PC	79
		9.1.2 Installation of ORMONTE on a mainframe computer (VAX 8600)	81
	9.2	Installation of PDA (Probability Data Analysis) Options	81

CONTENTS (continued)

		Page
Appendix A.	MATHEMATICAL FORMS FOR INPUT VARIABLE PROBABILITY DISTRIBUTIONS	85
Appendix B.	DETERMINATION OF NUMBER OF MULTI-VARIABLE SENSITIVITY STUDY ITERATIONS REQUIRED	91
Appendix C.	ANALYTICAL INVERSION OF THE TRIANGULAR DISTRIBUTION	93
Appendix D.	SAMPLE PROBLEM	99
Appendix E.	A RANDOM NUMBER GENERATOR FOR COMPUTERS USING THE INTEL 80X86 MICROPROCESSOR	185
Appendix F.	LIST OF ACRONYMS	191

LIST OF FIGURES

Figure		Page
4.1	Flowchart for Monte Carlo-based multivariable sensitivity analysis procedure for evaluation of a system subject to multiple uncertainties	11
4.2	Generation of a return input value from a random number and a cumulative density function	12
4.3	Sample output probability histogram from ORMONTE MVSS option	14
5.1	Generalized ORMONTE flowsheet	18
5.2	Example of user-provided RDMODL subroutine and input data set thereto	24
5.3	Example and schematic of user-provided MODEL subroutine and INCLUDE file	26
5.4	Sample plot made from data generated by ORMONTE SVSS option and the example model	30
5.5	Sample plot made from data generated by ORMONTE ELAST option and the example model	36
5.6	Graphical interpretation of ELAST algorithms	39
5.7	Available input uncertainty distributions constructed from statistical parameters used for the example model	43
5.8	Nomenclature and printed example for histogram construction portion of ORMONTE MVSS option	55
7.1	Typical cumulative probability S-curve	61
7.2	Example probability wheel display	62
7.3	LOTUS 1-2-3 macro designed for probability questioning (screen display menu)	64
7.4	Screen display showing topic name and variable value for which probability is to be evaluated	65
7.5	Screen display showing complete set of value/probability data obtained by questioning	66
7.6	Data from Fig. 7.5 as drawn by PLOT option	67
7.7	Screen display showing Fig. 7.5 data after automatic sorting	68
7.8	LOTUS 1-2-3 macro designed for data analysis (screen display, menu and example data to be fit to normal distribution	69
7.9	Screen display showing normal fit regression analysis results for data on Fig. 7.8.	71
7.10	Plot of Fig. 7.8 raw data points and normal distribution obtained by using parameters calculated from regression analysis in Fig. 7.9.	72
7.11	Plot of attempt to fit triangular distribution S-curve to normally distributed raw data in Figs 7.8 and 7.10.	73

Figures (continued)

Figure		Page
8.1	Interpreting the output relative probability histograms from an ORMONTE MVSS run	76
8.2	Locating a baseline unit cost on a hypothetical unit cost output histogram	77
C.1	Inversion of the triangular distribution	93
E.1	Random number generator for Ryan McFarland Fortran	187
E.2	Random number generator for Microsoft Fortran	188
E.3	Random number genertor using Pascal	189

LIST OF TABLES

Table		Page
5.1	Sample input data set for SUBROUTINE READOP	21
5.2	Sample input data set for SUBROUTINE RDMAST	22
5.3	Sample input data set for SUBROUTINE SVSSOP	32
5.4	Portion of SVSS summary table produced by SUBROUTINE SVSSOP	35
5.5	Sample input data set for SUBROUTINE MVSSOP	41
5.6	Defining parameters for non-histogram input distributions	44

ACKNOWLEDGMENTS

This work was performed by members of the Engineering Technology Division (ETD) at the Oak Ridge National Laboratory (ORNL) under the sponsorship of the Office of the Assistant Secretary for Defense Programs (DP), Office of the Director of Plans and Resources Management (DP-5) U.S. Department of Energy (DOE). The authors wish to acknowledge K. K. Hudson and E. G. Lazur of DOE/DP for their sponsorship and support of this documentation effort and their support of developing ORMONTE for use with cost models utilized by DP.

Although this is the first detailed documentation of the Oak Ridge Monte Carlo Driver Code (ORMONTE), development and use of the code has spanned a period of ten years and has involved several DOE sponsors and several individuals at both ORNL and the Oak Ridge Gaseous Diffusion Plant (ORGDP). R. L. Hoglund [formerly of the Operations Analysis Division (OA&P)/ORGDP)] was the program manager responsible for introducing and implementing the Monte Carlo methodology for the economic evaluation of advanced uranium enrichment technologies. (Work supported by the DOE/NE Office of Uranium Enrichment and Assessment under Norton Haberman.) Others involved in these early implementation efforts were J. R. Merriman (at that time head of Enrichment Technology Division), J. M. Googin (Y-12 Development Div.), and G. E. Michaels (at that time OA&P/ORGDP). L. M. Rosenbaum (at that time OA&P/ORDGP) was responsible for most of the FORTRAN coding. Code improvements were added by the following individuals in 1985 to enhance its utility in evaluating uranium enrichment technologies: T. J. Burns (Mathematics and Engineering Physics Division), D. Steinert [Computing and Telecommunications Division (C&TD)], A. K. Farrar (at that time OA&P), and W. B. Arthur (at that time Centrifuge Division). The assistance of T. C. Tucker (C&TD) in developing random number generators for various PC microprocessors and PC FORTRAN software is also recognized. Review of this document by several of the above contributors is also appreciated. The authors would also like to acknowledge the assistance of the ETD Word Processing Center, the ETD Publications Office, and C. L. Wagner in manuscript preparation.

ORMONTE: AN UNCERTAINTY ANALYSIS CODE FOR USE WITH USER-DEVELOPED SYSTEMS MODELS ON MAINFRAME OR PERSONAL COMPUTERS

K. A. Williams
C. R. Hudson II

ABSTRACT

Documentation is provided for a general purpose uncertainty analysis code entitled ORMONTE. The code is intended to be linked with FORTRAN-77 models designed by the user. ORMONTE is capable of performing sequential single-variable sensitivity studies (SVSS), elasticity analyses, and multivariable sensitivity studies (MVSS) by acting as a driver on the user's model. The MVSS function utilizes the Monte Carlo simulation method of probabilistic analysis. The model accessed by the ORMONTE driver code and the names and characteristics of the modeler's uncertain variables are provided by the user. ORMONTE is essentially a bookkeeping code which keeps track of the inputs and outputs defined by the user, samples the user-defined uncertainty distributions, repeatedly runs the user's model, and produces statistical data and output probability histograms for the output variables of interest to the user.

ORMONTE has been utilized extensively by the USDOE and its Oak Ridge contractors, Martin Marietta Energy Systems Inc., and its predecessor, Union Carbide Corp.; Nuclear Division, for the economic evaluation of large-scale energy projects for which computer models have been constructed for performance and cost projection purposes. ORMONTE has been used on several computer systems including a Cray, several VAXes, two IBM-360 mainframes, and DOS compatible PCs (along with the appropriate FORTRAN-77 software). The detail documentation provided here deals mainly with the PC version to be utilized with RM/FORTRAN; however, installation on a mainframe is reasonably simple. An example problem and complete ORMONTE input and output therefrom are provided.

Systems models driven by ORMONTE in the past include performance and cost models for new uranium enrichment plants, advanced nuclear power plants, coal fired plants, and defense materials production reactors. Future applications include ORMONTE's use as a driver for parametric costing models such as FAST-E and as a driver for a systems model such as that for the design and cost of a

fusion power plant. Some discussion of how ORMONTE results were used in evaluating advanced energy systems is included in the report. ORMONTE is structured such that it can easily be coupled to existing FORTRAN models and installed on a PC.

1. INTRODUCTION

As the use of probabilistic uncertainty/risk analysis has increased within Martin Marietta Energy Systems and the U.S. Department of Energy, it has become necessary to educate new users in the installation and operation of the ORMONTE Monte Carlo driver code so that it can be run in conjunction with their own mathematical models of energy systems. It should be noted, however, that ORMONTE can be used with any model, not just those that are energy-related. Emphasis in this report is on DOE applications. It is also anticipated that the increasing speed and storage capabilities of personal computers will make the use of uncertainty analysis methodology available to an increasing number of potential users. Since the economic evaluation tasks of interest to DOE/ Defense Programs (DP) involve considerable cost analysis, it was felt that documentation of the sensitivity analysis code (ORMONTE), which will drive their parametric cost models (Ref. 1.1), was an important step in the implementation of the technique for this application. The documentation is also necessary in order that ORMONTE can be linked to proprietary parametric cost estimating codes used by DOE/DP such as those in the FAST (Freiman Analysis of Systems Techniques) series.

Chapter 2 will discuss the rationale for the use of uncertainty/ risk analysis and why it is an important tool for augmenting the usual base case or deterministic estimates generated by a cost or performance model. Chapter 3 describes the background of the ORMONTE code and how it has been successfully applied in previous economic evaluation efforts for the Department of Energy. Chapter 4 gives a brief description of the mathematical basis and special features of the ORMONTE code. Chapter 5 is a detailed discussion of the overall structure and FORTRAN-77 algorithms within the code with emphasis on the mathematical methodology utilized. Chapter 6 discusses the considerations needed in designing or preparing the systems model(s) which are driven by ORMONTE. Chapter 7 discusses the development of input data for ORMONTE and how one can convert uncertainty data obtained from an expert interview into the probability distribution input form required by ORMONTE. Chapter 8 describes the interpretation of ORMONTE output, and Chapter 9 discusses the installation of ORMONTE on both PC and mainframe computers. A sample problem, which includes complete input and output data, is given in Appendix D. A listing of the code is not included in this report. This ORMONTE code is available in electronic form from the DOE National Energy Software Center at Argonne National Laboratory.

Throughout the report the attempt has been made to make the text easily understandable by nonstatisticians or nonmathematicians. The hope is that both

technical and nontechnical persons faced with decision-making tasks will find ORMONTE possibly useful in their analyses. The computer and statistical backgrounds of most technical and management staff should be adequate to at least understand the power of this uncertainty analysis method and the meaning of the results obtained therefrom.

If the reader wishes more detailed guidelines for process and economic modeling of emerging energy technologies, along with a highly detailed example project utilizing an earlier mainframe version of the ORMONTE uncertainly analysis code, Ref. 1.2 or 1.3 is recommended. References 1.2 and 1.3 are the same document and are available from the two sources listed in Sect. 1.1 below.

1.1 References

1.1 Gasperow, L.A., Hackney, J. W. and Hudson, K. K., *Parametric Cost Estimating; A Guide*, U.S. Dept. of Energy, Office of the Assistant Secretary for Defense Programs (DP 5.1), Washington, D.C., February 1987.

1.2 Williams, K. A., *A Methodology for Economic Evaluation of Process Technologies in the Early Research and Development Stages*, PhD Dissertation, The University of Tennessee, Knoxville, August 1984; 475 pages (available from University Microfilms International, 300 N. Zeeb Rd, Ann Arbor, MI 48106, publication # 8429631), or

1.3 Williams, K. A., *A Methodology for Economic Evaluation of Process Technologies in the Early Research and Development Stages*, Report #K/OA-5684; Martin Marietta Energy Systems, Inc.; Oak Ridge Gaseous Diffusion Plant, Oak Ridge, TN 37831; August 1, 1984; 475 pages (prepared for the U.S. Department of Energy under Contract No. DE-AC05-84OR21400; available from National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161; order #DE84015863).

2. RATIONALE FOR UNCERTAINTY/RISK ANALYSIS

2.1 Deterministic Projections

For most research and development (R&D) programs, researchers are required to make projections regarding the future performance and/or cost of their products or systems. This has indeed been the case for energy research programs carried out by the Department of Energy and its contractors in such areas as nuclear weapons, fusion, uranium enrichment, advanced nuclear reactors, synthetic fuels, and renewable energy sources. In evaluating such projects, decision-makers have usually had to rely on what are called "base case," "deterministic," or "single point" performance and cost estimates which rely on one set of input information usually supplied by the project proponents. These performance and cost projections are often produced via a computer model and, in some applications, a spreadsheet. The inputs to the estimating process usually consist of "best guess" numerical projections for those project-related variables that have significant leverage on the figure of merit upon which the project might be judged, such as unit cost or overall reliability.

The projections made for the key project inputs are likely to be significantly different from the values actually realized when the project is carried out through scaleup and deployment. Such disparity between projected and actual data is caused by inaccuracy of early R&D results, unforeseen events, incomplete design details, project scope changes, physical phenomena which are not understood, scaleup difficulties, and changes in the economic climate; therefore, most inputs to a projection have associated with them some degree of uncertainty. The net result is that the uncertainty associated with one or more inputs eventually propagates to an uncertainty in the figure of merit of interest for the project. When a deterministic figure of merit is located within its uncertainty spectrum, one can then describe the "risk" associated with the deterministic value and gauge its degree of optimism or pessimism.

2.2 Probabilistic Projections

Experience shows that deterministic performance and cost estimates usually prove to be optimistic when examined in the light of prevailing uncertainties. This problem is discussed from the standpoint of actual experience in Refs. 2.1-2.3. Uncertainty analysis can answer the following: given what I know about my system model and its uncertain inputs, how far off can my possible figure of merit outcome be from my earlier deterministic projection, and what is the probability of attaining a future value less than or equal to the projected deterministic outcome? The general methodology for examining this question is often called risk analysis or probabilistic analysis. Different mathematical tools for pursuing probabilistic analysis exist; among these are decision tree analysis, Monte Carlo simulation, and the Latin hypercube method. Such methods have been used extensively in the last

several years to evaluate health and safety related risks associated with energy systems. The results of such analyses are often referred to as PRAs (Probabilistic Risk Assessments). Several excellent references (Refs. 2.4-2.10) describe these methods and their applications.

This report will describe a general computer code which uses the Monte Carlo method to drive the user's system model such that a Multivariable Sensitivity Study (MVSS) is performed on the user's model. The user has the advantage that all input variable effects on the figure of merit can be examined simultaneously and the results presented in terms of probabilities and statistical parameters. The computer code described also has the capability to perform automated, sequential single-value sensitivity studies (SVSS). This procedure is a needed predecessor step to the multivariable sensitivity studies (MVSS). The applications described in this report are related to economic projections rather than projections of accident or health/ safety risks.

2.3 References

2.1 Merrow, E. W., et al., *A Review of Cost Estimation in New Technologies: Implications for Energy Process Plants*, The Rand Corporation, R-2481-DOE, July 1979.

2.2 Merrow, E. W., et al., *Understanding Cost Growth and Performance Shortfalls in Pioneer Process Plants*, The Rand Corp., R-2569-DOE, Sept. 1981.

2.3 Merrow, E. W., *Understanding the Outcomes of Megaprojects*, The Rand Corporation, R-3560-PSSP, March 1988.

2.4 Rish, W. R., *Approach to Uncertainty in Risk Analysis*, ORNL/TM-10746, Martin Marietta Energy Systems, Inc., Oak Ridge National Laboratory, August 1988.

2.5 Rish, W. R., *Review of Studies Related to Uncertainty in Risk Analysis*, ORNL/TM-10776, August 1988.

2.6 Dunigan, T. H., *A Portable Hypercube Simulator*, ORNL/TM-10410, July 1987.

2.7 Worley, B. A., *Deterministic Uncertainty Analysis*, ORNL-6428, December 1987.

2.8 Henrion, M., et al; "A Computer Aid for Risk and Other Policy Analysis," *Risk Analysis*; 5 No. 3, p 195-207, (1985).

2.9 Henrion, M., *The Value of Knowing How Little You Know: The Advantages of a Probablistic Treatment of Uncertainty in Policy Analysis*; PhD Thesis; Carnegie-Mellon University, March 1982 (Univ. Microfilms #8314544).

2.10 Cox, D. C., et. al., "Methods for Uncertainty Analysis; A Comparative Survey;" *Risk Analysis* (1) No. 4, pp 251-258, (1981).

3. BACKGROUND OF THE MONTE CARLO UNCERTAINTY ANALYSIS METHODOLOGY

3.1 Historical Development

For very simple linear models it has always been possible to calculate the "propagation of uncertainty" from inputs to outputs via direct analytical means, i.e., using the statistical parameters (the means and variances) describing the uncertainty in each input variable (Ref. 3.1). Most models, however, are more complex and involve nonlinear relationships in their algorithms, thus nonanalytic techniques are needed to perform MVSSs. The availability of high-speed digital computers made feasible the application of Monte Carlo techniques to a wide range of scientific, engineering, and economics problems involving complex models and multiple uncertainties.

Monte Carlo techniques are not new, and their historical use is well documented. Intense use of the techniques began as part of the Manhattan Project during World War II, mostly in the course of studies of the physics and design of the atomic bomb. For example, Von Neumann and Ulam (Ref. 3.2) utilized the method to simulate neutron diffusion in fissile material. Since then the method has been utilized in nuclear reactor design, plasma physics, and many other research applications. In the early 1960s, David B. Hertz (Refs. 3.3-3.5) was the first to apply the method to economic and business analysis and projections. It soon became apparent that the Monte Carlo method, along with modeling, could be utilized as an R&D tool in the assessment of many processes or systems where technical and economic uncertainties abound, such as weapons systems (Ref. 3.6), fusion (Ref. 3.7), and new chemical plant ventures (Ref. 3.8).

3.2 Use of Monte Carlo Based Uncertainty Analysis in Economic Evaluations Performed at Oak Ridge

In Oak Ridge the intensive application of the Monte Carlo technique for cost/risk MVSS had its roots in the Uranium Enrichment process economic evaluation programs carried out in the Operations Analysis and Planning Division of the Oak Ridge Gaseous Diffusion Plant (ORGDP). Simultaneous advancement of both laser technology and plasma science resulted in significant R&D work on highly innovative advanced isotope separation (AIS) methods for the uranium enrichment step of the commercial nuclear fuel cycle. By 1976, three AIS processes were being funded by the Department of Energy: the Molecular Laser Isotope Separation (MLIS) Process, developed by Los Alamos National Laboratory; the Atomic Vapor Laser Isotope Separation (AVLIS) Process, developed by Lawrence Livermore National Laboratory; and the Plasma Separation Process (PSP), developed by TRW

Incorporated. All three of these processes involved high-technology hardware still in the basic laboratory stages, and each had been demonstrated on the proof-of-principle level. A number of the key scientific and cost issues relevant to scale-up of these processes were highly uncertain or left to be determined; hence, there was a need for an economic assessment activity in parallel with R&D suitable to these conditions.

By the late 1970s, budgetary considerations dictated that DOE choose only one of these three processes for further R&D. Each contractor laboratory was asked to explore and contribute for peer review certain critical basic information on its proposed process candidate. Martin Marietta's predecessor, the Union Carbide Corporation-Nuclear Division (UCC-ND) at Oak Ridge was assigned the task of modeling and assessing the technical and economic aspects of the three proposed AIS processes. Along with extensive process models, the Monte Carlo driver code here described played a key role in the May 1982 DOE Headquarters decision to choose the AVLIS process for further engineering and development. By the mid 1980s budgetary problems again dictated that DOE make a process selection, this time between AVLIS and the advanced gas centrifuge (AGC) process. The Martin Marietta Energy Systems Corporation at Oak Ridge was assigned the task of again modeling and assessing the technical and economic aspects of the two proposed processes. Along with extensive process and cost models, the Monte Carlo driver code again played a key role in the June 1985 DOE Headquarters decision to choose AVLIS for further development. Reference 3.9 briefly describes the results of these uncertainty analysis studies.

In 1986 the method was applied to augment deterministic projections of busbar power costs for coal-fired and nuclear plants coming on line in the year 2000 (Ref 3.10). Other Oak Ridge/DOE-sponsored applications of the methodology have been: space power sources for the Strategic Defense Initiative (SDI) (Ref. 3.11), cost estimating for DOE new production reactor (NPR) concepts (Ref. 3.12), and cost estimating for two advanced liquid metal reactor (LMR) concepts (Ref. 3.13). These last two applications were also involved in a DOE decision-support process.

3.3 References

3.1 Cortes, R. H., "How Wrong Can Your Calculations Be?" *Petroleum Refiner*, Vol. 37, No. 10, pp. 129-130 (June 1958).

3.2 Curtiss, J. H., et al., *Monte Carlo Method*, National Bureau of Standards Applied Mathematics Series, Vol. 12 (1951).

3.3 Hertz, D. B., "Risk Analysis in Capital Investment," *Harvard Business Review* (January-February 1964).

3.4 Hertz, D. B., Thomas, H., *Risk Analysis And Its Applications*; Wiley, New York, 1983.

3.5 Hertz, D. B. and Thomas, H., *Practical Risk Analysis: An Approach Through Case Histories*; Wiley, New York, 1984.

3.6 Timson, F. S., "Measurement of Technical Performance in Weapon System Development Programs: A Subjective Probability Approach," Report No. RM-5207-ARPA, The Rand Corp., Santa Monica, Cal (Dec. 1968).

3.7 *The Economics of Fusion Research*, ERDA-COO/4181-1, Econ Inc. (Oct. 15, 1977).

3.8 Williams, K. A. and Holmes, J. M., "Simulation of Process Economics for Pioneer Plants," *Chem. Engr. Progress*, Jan. 1987.

3.9 "Uranium Enrichment Technology Evaluation and Assessment – May 15, 1985," *Nucl. Fuel*, 10(12), p. 18 (June 17, 1985).

3.10 Williams, K. A., et al., *Nuclear Economics 2000: Deterministic and Probabilistic Projections of Nuclear and Coal Electric Power Generation Costs for the Year 2000*, ORNL-6368, Martin Marietta Energy Systems, Inc., Oak Ridge National Laboratory; June 1987.

3.11 Williams, K. A., "A Methodology for Economic Evaluation of Technologies in the Early R&D Stages," *Proceedings of the 1986 Summer Computer Simulation Conference, July 1986*, Reno, Nev.; the Society for Computer Simulation, p. 1094-1099.

3.12 *New Production Reactor (NPR) Capacity Cost Estimate*, DOE/DP-0052; U.S. Dept. of Energy, Office of the Asst. Secy for Defense Programs, July 1988.

3.13 "LMR Advanced Concepts: Special Review of Phase III Cost Estimates for th PRISM and SAFR," prepared by C. R. Hudson for the U.S. DOE Office of Advanced Reactor Programs, March 1988.

4. BRIEF DESCRIPTION OF THE MONTE CARLO TECHNIQUE FOR MULTIVARIABLE UNCERTAINTY EVALUATION

Monte Carlo simulation methods can provide probability distributions in the form of histograms for both uncertain input variables and output figures of merit from a user's model. This is accomplished by making repeated evaluations of the system response (via the model) using input variable values drawn randomly from appropriate distributions that represent the projected statistical properties of each uncertain input variable. After performing a large number of such evaluations, a statistical distribution or histogram of system response is obtained. Meaningful use of the Monte Carlo method requires a user-supplied systems model adapted to the Monte Carlo code here described.

4.1 Mathematical Description of Computational Procedure

The Monte Carlo simulation procedure consists essentially of five steps. Figure 4.1 shows a flowchart for the simulation procedure

1. Input: For each independent input variable, Y_i (i = 1 through NI; where NI = total number of input variables), a distribution must be defined which gives the probability of occurrence for any particular value within the variable range. [It should be noted that most Monte Carlo users utilize the term "distribution" to refer to a probability density function (pdf).] The types of distributions for each variable need not be the same. The subject of determining the distribution type and input values for a given variable is discussed in Chapter 7 and Appendix A.

2. Sampling: Via a random number generator and the input distributions, ORMONTE generates an ensemble or vector, Y_i (i = 1, NI) of randomly selected input values for submission to the user's model. Each ensemble of values is unique and represents a different possible set of occurrences. [In order for a random number (a real number between 0 and 1) to be utilized, the pdf must be integrated by ORMONTE to form a cumulative distribution function (cdf) with a maximum cumulative probability of unity, i.e., the area under the pdf curve. The cdf is then algebraically inverted or transformed such that an input variable value or "return value" can be calculated for each cdf probability value selected between 0 and 1. This inversion procedure is shown graphically in Fig. 4.2 and is described in more detail in Chap. 5 and Appendix C]. The baseline or reference deterministic case will never be exactly reproduced in this sampling process.

3. Simulation: For each iteration the model calculates single values of the figures of merit and other variables internal to the model whose ultimate distributions are desired by the analyst [these variables are designated as Y_i (i = NI + 1, NT); where NT is the total number of input plus output variables. Note that the overall Y_i vector includes both input and output variables]. It is assumed that the

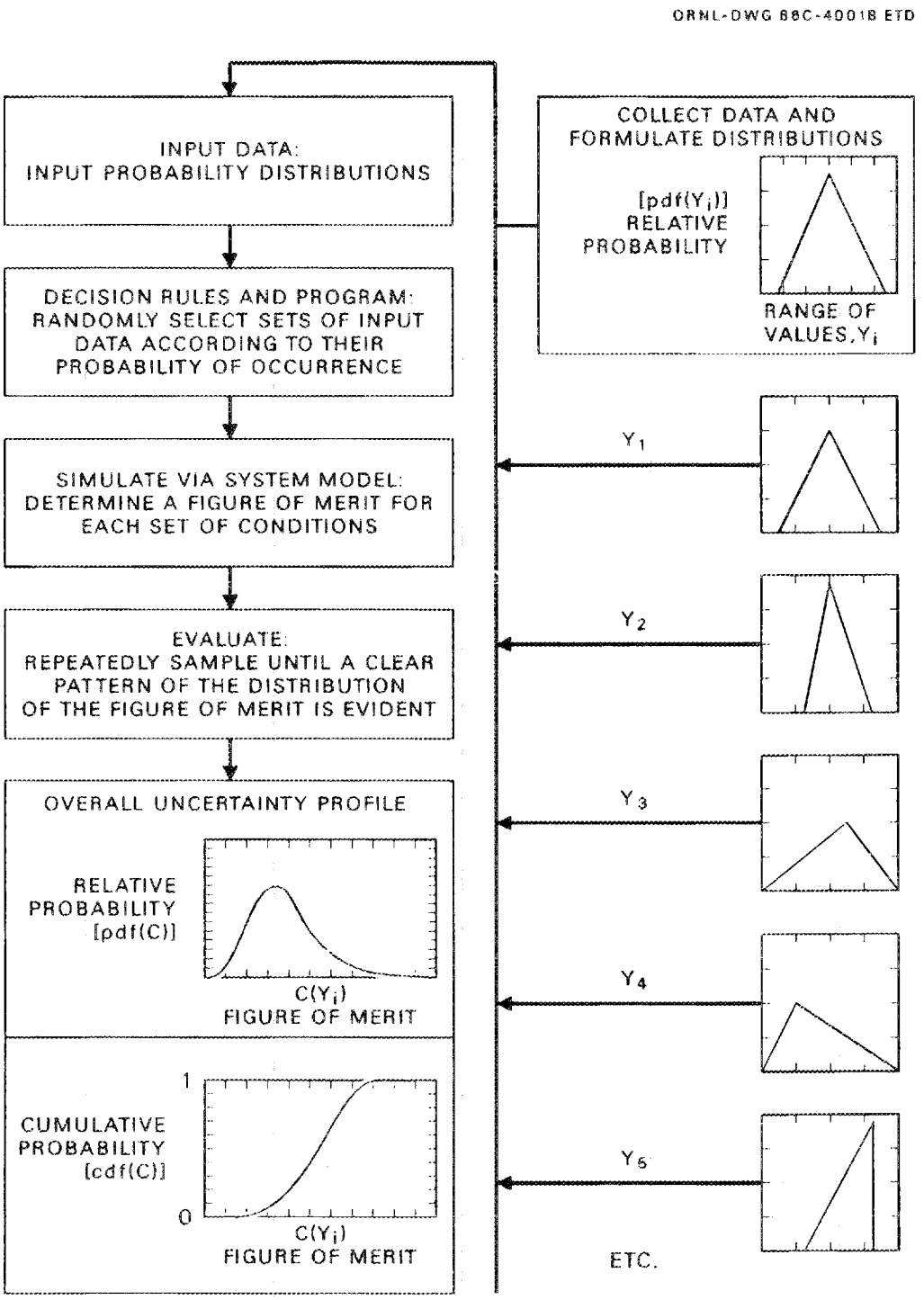


Fig. 4.1. Flowchart for Monte Carlo-based multivariable sensitivity analysis procedure for evaluation of a system subject to multiple uncertainties.

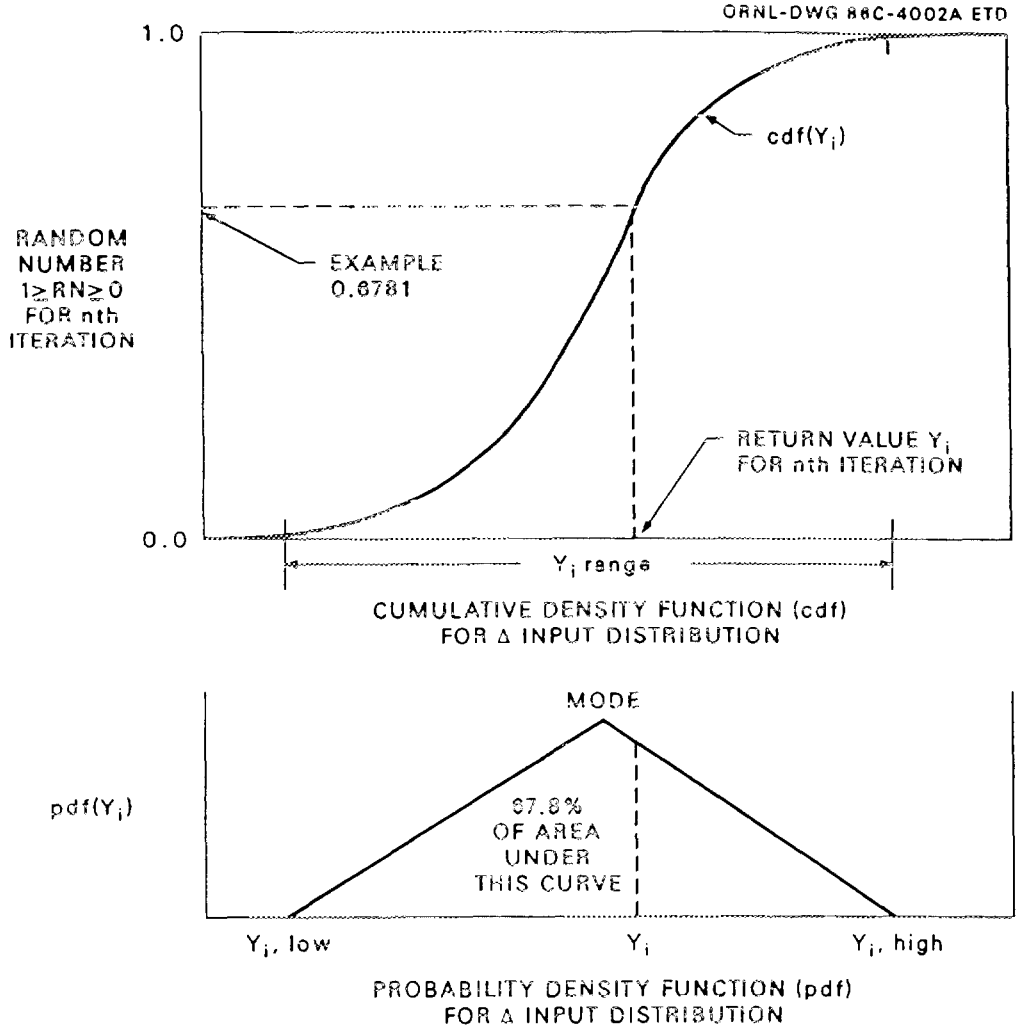


Fig. 4.2. Generation of a return input value from a random number and a cumulative density function.

ensemble of input variables submitted is internally consistent and does not represent a physically or logically impossible or unreasonable situation, especially in the sense of the system performance model. If this problem exists, the user's model should be restructured to avoid the generation of such cases.

4. Iteration and Storage: Once the figures of merit for a given iteration are evaluated, they are stored and another ensemble of input values is submitted to the user's model. The number of iterations or cases needed is dependent somewhat on the model and the user's desired level of accuracy (see Appendix B). If an input ensemble consisting of the means for each input distribution is submitted to the model (a separate calculation from the Monte Carlo simulation), a mean output value results. If during the simulation the output values for each figure of merit are stored and the statistical mean for all iterations is recalculated after each additional

iteration, it will be found that this mean will eventually approach the calculated mean evaluated off-line using the input distribution means. Iteration can be terminated when the difference between the two output means reaches an arbitrarily small fraction of the off-line calculated mean. (One method for determining the required number of iterations is described in Appendix B.) Experience indicates that for most complex models at least a few hundred evaluations or iterations are needed. Using additional evaluations improves the statistics and the geometrical appearance (shapes) of the input and output distributions; however, depending on the complexity of the model, the hardware, and accounting procedures, the price in mainframe machine computation time and computer input-output units (I-Os) can be high.

4.2 Nature of Output from a Monte Carlo MVSS Run

The Monte Carlo simulation technique allows consideration of hundreds of different off-baseline cases. If the analyst wishes, every case could be examined individually by printing full output tables for every iteration or case that represents submission of a new set or ensemble of input values. In reality, the user is probably interested only in output of a limited number of crucial variables or figures-of-merit. The design of ORMONTE allows the user the option of choosing whatever variables he desires for display and analysis. The vector composed of sampled and calculated values, $Y_{i,k}$ [i = 1 to NT; where NT = NI + NO = total number of variables (input plus output), k = a particular iteration] is stored for each iteration, m, of the simulation. At the end of the simulation (m = M iterations), a very large array Y of dimension M × NT is available for statistical analysis and plotting of histograms. The values stored in this array are for both input and figure-of-merit variables.

By saving the actual input values sampled by the random number generation, the user can check how well the total input samples drawn approximate the input probability density function chosen. If the approximation is poor, i.e., the statistical parameters for the sample and input distribution definition differ significantly, then the user must specify more iterations.

4.2.1 The generation of output histograms

The array of Y values (inputs plus outputs) generated by the simulation essentially represents the collection of M (number of iterations) responses per Y_i value obtained by the submission of M sets of deviations or sets of nonbaseline sampled inputs submitted to the overall model.

This output is best understood if plotted in the form of a relative probability distribution or pdf. These distributions take the form of histograms constructed from the M Y-vectors available for analysis. If small equal size bins of width $\Delta Y_{i,m}$ are specified for a given Y_i, each $Y_{i,m}$ value out the total of M can be assigned to the bin which contains the boundaries that encompass its value. The result is a histogram as shown in Fig. 4.3. If the particular Y_i is a function of two or more input

ORNL—DWG 89—4199 ETD

STATISTICS FOR OUTPUT VARIABLE PCOST (13) JTYPE=-99 SAMPL.DIST TYPE: NOT-APPL

MINIMUM 15.301 MAXIMUM 81.777

MEAN 30.676 VARIANCE 65.110 STD.DEVIATION 8.0691

PERCENTILES: 2.5% 18.8 5.0% 20.0 50.0% 29.3 95.0% 45.9 97.5% 49.6

5000 POINT HISTOGRAM * = 8POINTS

PROB	CUME	VALUE		
0.0008	0.0000	14.00	4	*
0.0080	0.0008	16.00	40	*****
0.0410	0.0088	18.00	205	**********
0.0722	0.0498	20.00	361	**********
0.0968	0.1220	22.00	484	**********
0.1072	0.2188	24.00	536	**********
0.1088	0.3260	26.00	544	**********
0.1012	0.4348	28.00	506	**********
0.0874	0.5360	30.00	437	**********
0.0754	0.6234	32.00	377	**********
0.0672	0.6988	34.00	336	**********
0.0526	0.7660	36.00	263	**********
0.0466	0.8186	38.00	233	**********
0.0364	0.8652	40.00	182	**********
0.0282	0.9016	42.00	141	**********
0.0214	0.9298	44.00	107	**********
0.0158	0.9512	46.00	79	**********
0.0098	0.9670	48.00	49	******
0.0092	0.9768	50.00	46	******
0.0060	0.9860	52.00	30	****
0.0032	0.9920	54.00	16	**
0.0016	0.9952	56.00	8	*
0.0008	0.9968	58.00	4	*
0.0010	0.9976	60.00	5	*
0.0006	0.9986	62.00	3	
0.0000	0.9992	64.00	0	
0.0004	0.9992	66.00	2	
0.0000	0.9996	68.00	0	
0.0002	0.9996	70.00	1	
0.0000	0.9998	72.00	0	
0.0000	0.9998	74.00	0	
0.0000	0.9998	76.00	0	
0.0000	0.9998	78.00	0	
0.0002	0.9998	80.00	1	
	1.0000	82.00		

Fig. 4.3 Sample output probability histogram from ORMONTE MVSS option.

variables, the output histogram will usually take the shape of a bell-shaped distribution. This observation is true if the geometric forms of the input distributions submitted to the model have single peaks. The generation of normal or quasi-normal output distribution forms is a consequence of the Central Limit Theorem of statistics.

The ORMONTE driver code is designed to perform the repetitive data submission task and also the histogram-plotting and output data analysis task. ORMONTE calculates the statistical parameters such as standard deviation,

median, mean, and selected percentile probability ranges for all specified variables, Y_i (both input and output).

The 5 and 95 percentiles for each histogram are most useful since the analyst is most often interested in the middle 90% range of a distribution. For a normal distribution these percentiles approximate the $\pm 1.65\sigma$ region (σ = standard deviation). If the user's model is sufficiently robust, there is little danger in discarding the cases representing the 5% "tails" at each end of an output histogram. By dealing with the middle 90%, the user is considering the more representative cases or outcomes.

In many cases discontinuities, raggedness, or "spikes" will occur in the output histograms. Two causes contribute to this appearance, poor input statistics (due to insufficient sample size) and/or the existence of discontinuous functions within the user's model. If a large number of iterations is used, the former factor will have a reduced effect; however, the latter can arise as a result of reoptimization within the model (which will drive the variable being optimized to find a discontinuity) and the fact that some entities in the user's model may occur as discrete or integer quantities.

4.3. Special Features of the ORMONTE Code

The ORMONTE code is very general in nature and in most cases can be run in conjunction with an already existing code, with usually only minor modifications to the latter. Among the salient features are:

1. In addition to performing Monte Carlo MVSS, the driver code can also perform automated single-value sensitivity studies (SVSS), run only the base case, or perform elasticity analysis (% change in an output per % change in an input). The SVSS and elasticity options procedures, when applied to the user's model, can help in selecting which variables are important enough to merit becoming MVSS input variables. Variables having significant leverage on the output figure of merit should be chosen. The Pareto principle implies that out of many possible input variables, only a limited number are likely to exert significant leverage.
2. A large number of input and output variables can be handled in each simulation. The sum of the number of uncertain input and output variables can be up to 150. Minor changes to the source code could increase this sum.
3. A master COMMON block is used to pass input and output data back and forth between the user's model and the ORMONTE driver code. The INCLUDE feature of most FORTRAN-77 compilers allows coding of this master COMMON block only once, rather than coding it into every SUBROUTINE in the user's model.
4. The user of ORMONTE has nine different input probability distributions from which to choose for describing input data uncertainty. A histogram option is also available and can be used to approximate other more complex

types of distributions. Within a given run of ORMONTE, mixed types of distributions can be used for a set of inputs.

5. Following completion of all iterations comprising the simulation, the stored data are analyzed by ORMONTE, and probability histograms are printed for both input and output variables. These histograms are produced for output to the standard printer connected to the user's computer system. ORMONTE can also save the histogram data for use with plot programs such as DISSPLA, CHARTMASTER, HARVARD GRAPHICS, or LOTUS 1-2-3.
6. Also included in this ORMONTE package are new computer tools for the acquisition of raw probabilistic input data and for the characterization of such data. This PDA (Probability Data Analysis) package requires the use of LOTUS 1-2-3 and the software provided along with the ORMONTE FORTRAN source files. A major function of the "probability wheel" feature of the PDA package is to elicit probabilistic responses from an expert.

At Oak Ridge, ORMONTE has been successfully run with FORTRAN-77 compilers and linkers on the following mainframe systems: IBM 3033, VAX (several models), PDP-10, and the Cray XMP. Its use on an IBM PC requires FORTRAN-77 compiler and linker software such as Ryan McFarland or Microsoft FORTRAN.

5. DETAILED DESCRIPTION OF THE ORMONTE CODE

This section describes the MAIN and the various subroutines which constitute the ORMONTE code. (The PDA options are described in Sec. 7.2.) In addition to this description, the FORTRAN source code itself is liberally commented such that the user should have little difficulty following the code's logic structure. The FORTRAN source code is available in electronic form from the National Energy Software Center at Argonne National Laboratory.

5.1 Basic Coding Structure

From the beginning ORMONTE has been designed such that it can be easily integrated with existing user models written in FORTRAN 77 or its predecessor FORTRANs. The principal problem in connecting a driver code such as ORMONTE to an existing code is establishing comunication of input and output variables between the codes without forcing the user to deal with a whole new set of variable names. ORMONTE circumvents this problem by use of a COMMON block named /MASTER/ which appears in both its own drivers and the user's model. In the user's model /MASTER/ contains all of the sensitivity variables of interest, both inputs and outputs, and retains the variable names preferred by the user. In the ORMONTE driver /MASTER/ holds a vector Y(150) in which each element Y(I) corresponds to a variable in the user's model. The input and output to ORMONTE, however, can utilize the variable names appearing in the user's model. COMMON blocks /MASNAM/ and /PARAM/ handle the bookkeeping needed to accomplish this task. The use of the /MASTER/ COMMON block has another advantage in that it allows easier debugging of the user's code if a Monte Carlo simulation fails before all iterations have been completed. By printing the Y-vector passed via /MASTER/ the user can look at the set of inputs and outputs associated with a particular iteration of his model and see if any physically impossible or basically inconsistent sets of inputs were chosen during the sampling process; if so, the problem is with the inputs chosen or the user's model and not with the driver code itself. Use of the /MASTER/ COMMON block is made for all four driver options available in ORMONTE: the baseline or deterministic estimate (DETO), stacked one-at-a-time or single value sensitivity studies (SVSS), multivariable sensitivity studies (MVSS: the probablistic method using Monte Carol methodology), and the elasticity analysis (ELAS), (which is a special type of single-value sensitivity study).

5.2 The MAIN Program

The MAIN consists of a sequence of calls to the various subroutines constituting the code. The following subroutines are called during a run as shown on Fig. 5.1:

READOP -- Reads the driver option desired by the user, the input/output unit specifications, and the number of sensitivity variables to be processed by the code.

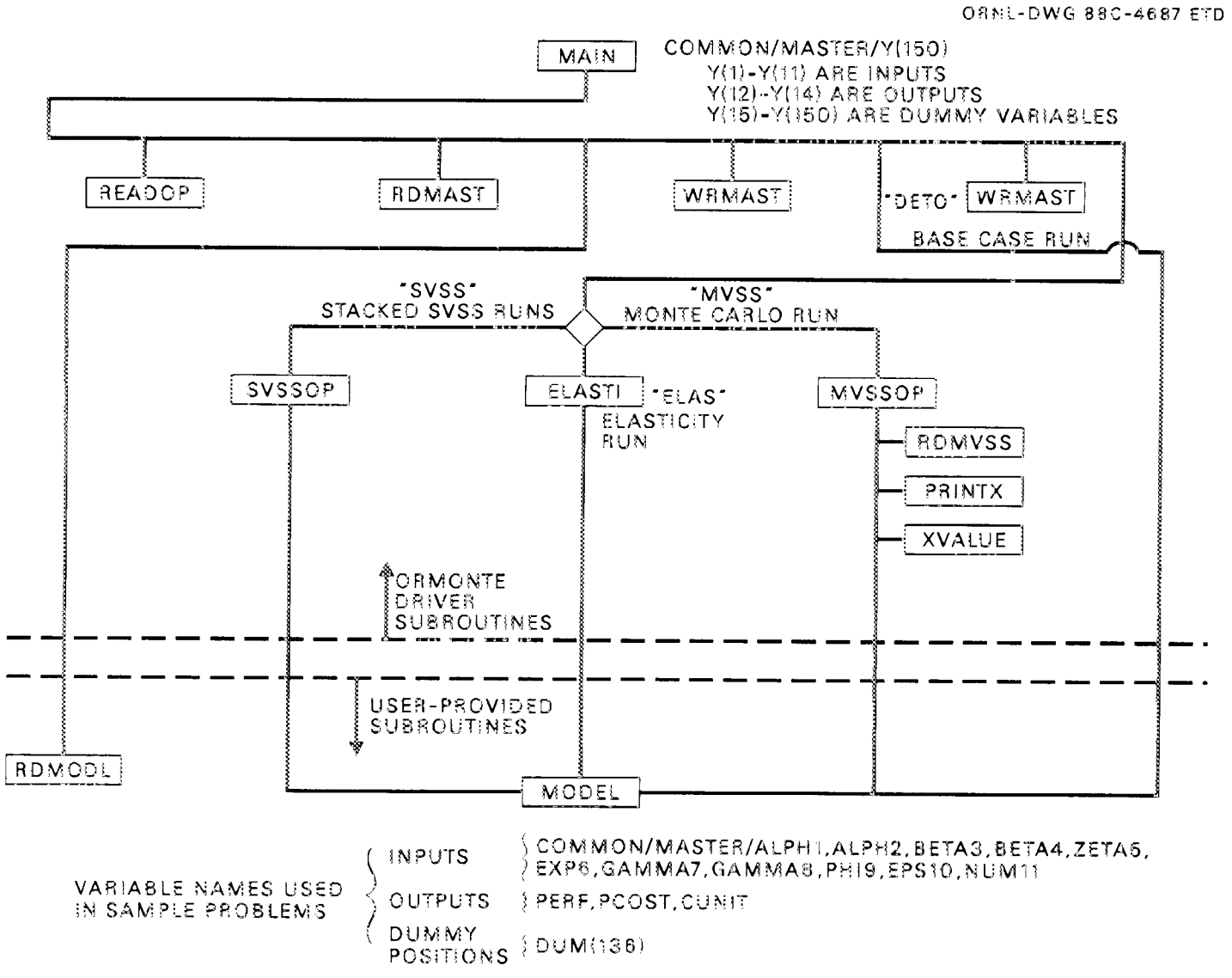


Fig. 5.1. Generalized ORMONTE flowsheet.

RDMAST — Reads the "base case" or deterministic input data for initializing the Y-vector (which is passed via /MASTER/).

RDMODL — Reads the nonsensitivity data specific to the user's model. The user must develop this routine.

WRMAST — Prints out the data read by RDMAST. This routine can also be called anytime it is necessary to ascertain the current values of the variables in the Y-vector.

MODEL — This is the user's model and mathematically defines the desired figures of merit (FOMs) or outputs in terms of the uncertain input variables. This first call to MODEL allows calculation of a single-point, "base," or deterministic case. (For USDOE applications, this model was often several hundred lines of FORTRAN simulating the projected performance, design, and life cycle costs of a scaled-up energy-related process undergoing R&D. Applications of ORMONTE are by no means limited to this type of model.)

WRMAST — This routine is called again to print out both the input and calculated output variables (Y-vector) corresponding to the deterministic case or for a given iteration. Essentially, WRMAST writes out the current contents of /MASTER/.

One of the following subroutines is called depending on the sensitivity analysis option desired by the user. All three of these subroutines call MODEL; MVSSOP calls other subroutines which are described later in this chapter.

SVSSOP — Single-Value Sensitivity Study (SVSS) Option. This driver allows the user to specify alternate or non-baseline values for a particular input variable and have the desired figures of merit calculated for each case. By stacking the input data, several variables can undergo SVSS analysis within a given run. Since these are one-variable-at-a-time sensitivity studies, all other input variables in the Y-vector are held at their deterministic values, i.e., for a study on a particular variable Y_i, $Y_k = Y_{k,base}$ for $k \neq i$.

MVSSOP — Multi-Variable Sensitivity Study (MVSS) Option. This Monte Carlo driver option allows the user to consider multiple uncertainties in his model inputs. Input variable uncertainties must be expressed in probabilistic form. Probability histograms are then plotted for the figures of merit of interest.

ELASTI — This subroutine calculates the elasticity or coefficient-of-sensitivity (COS) for all of the input sensitivity variables handled by the Y-vector. The elasticity is the percent change in a desired figure of merit divided by the percent change in a given input. This analysis is useful for determining which input variables have the most impact on the FOM.

At the beginning of any of the three options above, one deterministic case is run using the base inputs read by RDMAST. At the end of an ORMONTE run, the MAIN prints out a message letting the user know that the run is complete and that normal termination of the run has occurred.

5.3 Description of Data Entry Subroutines Utilized for All Three Sensitivity Analysis Options

Regardless of the sensitivity analysis option chosen, input subroutines READOP, RDMAST, and RDMODL are always called by the MAIN. One data set is associated with each of these routines. These data sets can be opened, read, and closed on either the same or separate input units. In the example problem described in this report, input units 4, 8, and 5 are utilized by READOP, RDMAST, and RDMODL respectively. Short descriptions of the data requirements and logic for each data entry subroutine are given below. Sample data sets for the example problem are given in this chapter and in Appendix D.

5.3.1 SUBROUTINE READOP

READOP reads the data on unit 4 from a dataset named OPTIO.DAT. An example data set is shown on Table 5.1. The following, mostly INTEGER variables are read:

TITLE — A 50 character title identifying the data set

INMAST — The input unit used by SUBROUTINE RDMAST for sensitivity variable initialization (INMAST = 8 for the example problem)

INMOD — The input unit used by SUBROUTINE RDMODL to read in the non-sensitivity data for the user's model (usually INMAST = 5)

ISVSS — The input unit used to read in instructions and data for the single-value sensitivity study option (ISVSS = 11 for the example problem)

IMVSS — The input unit used to read in instructions and data for the multi-variable sensitivity study option (IMVSS = 9 for the example problem)

IELAS — The input unit used to read in instructions and data for the elasticity analysis option (IELAS = 12 for the example problem)

IPRNT — The output unit for normal ORMONTE printing (usually IPRNT = 6)

IPLOT — The output unit for output data which is saved for off-line plotting (IPLOT = 15) for SVSS and MVSS options only

IBUG — The output unit used for detailed debugging printout (IBUG = 14) [Y-vector printout]

Table 5.1. Sample input data set
for SUBROUTINE READOP

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [ELAS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]
```

The following CHARACTER*4 variable allows the user to select a sensitivity analysis option:

OPTION — The character constant read in for OPTION selects the sensitivity option as follows:

- 'DETO' — Run only a deterministic or "base" case using the data in MASINI.DAT
- 'SVSS' — Run a single variable sensitivity study (SVSS)
- 'MVSS' — Run a multivariable sensitivity study (MVSS)
- 'ELAS' — Run an elasticity analysis

The last variable read in is INTEGER*2 form:

NTOT — The total number (input + output) of sensitivity variables to be read in and initialized by RDMAST. This value must be smaller than the size of the Y-vector, i.e., NTOT is less than or equal to 150.

The CHARACTER*n data type variables AL1, AL2, BL4, etc., which are read in along with the above variables, are included to make the data set OPTIO.DAT self-explanatory and to reduce the chance of column misalignment during data set editing. The characters read in for these variables will not affect the code calculations.

It should be noted that the unit designator INOPT must be defined in READOP prior to opening the data set OPTIO.DAT. After all of the data is read READOP prints out a near replica of the data set OPTIO.DAT on unit IPRNT and confirms that all cards (lines of data) were read. COMMON /OPTION/ passes the read in data to the rest of the code.

5.3.2 SUBROUTINE RDMAST

The function of RDMAST is to initialize the Y-vector with deterministic or base case input values for each of the input sensitivity variables or elements in the

Y-vector. RDMAST also reads the user-designated variable names corresponding to each variable in the Y-vector. The COMMON /MASTER/ statement makes the connection between the values designated by the variable names in the user's model and the Y-vector utilized by the ORMONTE sensitivity driver options. The data are read in from data set MASINI.DAT (example shown on Table 5.2; note that each line of a data set is called a "card"). In addition to a title card (50 characters maximum), NTOT initialization data cards are read in via the DO-loop in RDMAST. This fills in vector elements Y(1) through Y(NTOT). For the Y-vector values in this group which represent output variables, a value of -1 is entered. The variables must be ordered the same way as they are in the /MASTER/ COMMON block with input variables followed by output variables. The following variables are read from each card:

NAMEN(I,J) — The mnemonic variable name for variable Y(I) chosen by the user and used for a sensitivity variable name in his model. The index I is the same as that for Y(I). Since NAMEN has a CHARACTER*1 Data type, the index J (incremented from 1 to 6) allows a six character maximum variable name. If longer names are desired, the range of index J can be increased to 8 by altering the source code.

Y(I) — The numerical value for this input element of the Y-vector, for output variables enter -1. (The model will later calculate the correct deterministic values for Y-vector output elements.) If an input variable happens to have a value of -1 the code will know that the variable is still an input rather than an output, since a later data set (MVSS.DAT) actually defines the number of inputs and outputs.

Table 5.2. Sample input data set for SUBROUTINE RDMAST

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
ALPH1     200.
ALPH2     50.
BETA3     10.
BETA4     1.
ZETA5     1.
EXP6      1.
GAMMA7    8.
GAMMA8    10.
PHI9      18.
EPS10     1.
NUM11     1.
PERF      -1.
PCOST     -1.
CUNIT     -1.
```

RDMAST fills the unused Y-vector positions with zeroes for Y(NTOT+1) through Y(150). The unused NAMEN(I,J) positions are filled with '******' characters. The subroutine prints out the Y-vector data, including the index, name, and value of each variable. At the end of RDMAST a message printing out the unit number (INMAST) confirms that all input RDMAST data have been read.

5.3.3 SUBROUTINE WRMAST

WRMAST is predominantly a diagnostic routine used to print the contents of the master Y-vector whenever WRMAST is called. The WRITE format is similar to that of RDMAST. The MAIN calls WRMAST immediately after calling RDMAST as a means of confirming that the initializing Y-vector data have been read correctly and are available through COMMON /MASTER/. Since the print flag IFLAG is equal to zero at this point, WRMAST prints on unit IPRNT. If WRMAST is used as a debugging tool during SVSS or MVSS, the value of IFLAG is greater than zero and WRMAST will print on unit IBUG. If during an MVSS run, execution abnormally terminates prior to completion of all iterations, it is useful to call and print the Y-vector using WRMAST for the failed iteration. In this manner the problem within the user's model can be readily isolated and identified. Usually the problem turns out to be the submission of a set of inputs that are logically or physically inconsistent from the standpoint of the relationships and algorithms within the user's model.

5.3.4 SUBROUTINE RDMODL: A user-provided subroutine

The user of ORMONTE must take his model and separate the sensitivity variables (those uncertain inputs and outputs which will be in the Y-vector) from the non-sensitivity variables (those certain inputs/outputs which will not be manipulated by the ORMONTE driver). The sensitivity variables were read by RDMAST above. RDMODL reads the nonsensitivity variables specific to the users application and passes them to his model via COMMON /USRMOD/. The reason this read subroutine is disconnected from the user's model and called by the ORMONTE driver MAIN is as follows: during the course of an MVSS, the user's model is called hundreds of times by the Monte Carlo driver; it is wasteful of CPU time and I/Os to have the user's model reread its nonsensitivity input data every time it is called. The READ formatting and number of and names of the input variables are solely at the discretion of the user. The source code listing in Fig. 5.2 goes with the SUBROUTINE MODEL example in this chapter and also is the RDMODL subroutine designed for the example problem in Appendix D. The input data set MODL.DAT is also shown on Fig. 5.2.

5.4 SUBROUTINE MODEL: A User-Provided Routine

Most users of the ORMONTE driver code will find it necessary to convert their particular application or code to a subroutine callable by ORMONTE called MODEL. This will mean converting the original user's MAIN to SUBROUTINE MODEL; however, MODEL will continue to call any other subroutines that are part

ORNL-DWG 89-4200 ETD

Source File: RDMODL.FOR

```
C ***************************************************
      SUBROUTINE RDMODL
C ***************************************************
C  DECLARATIONS
      CHARACTER*72 TITLE
      CHARACTER*6 NAMVAR
C  I-O DATA IS PASSED BY THIS COMMON BLOCK
      COMMON/OPTION/INOPT,INMAST,INMOD,ISVSS,IMVSS,IELAS,
     *IPRNT,IPLOT,IBUG,OPTION,NTOT
C  THIS COMMON BLOCK PASSES THE READ IN DATA TO SR. MODEL
C  AND WILL HAVE DIFFERENT ELEMENTS FOR EACH APPLICATION
      COMMON/USRMOD/C1,C2,C3
      DIMENSION NAMVAR(3)
C  DATA IS READ IN ON UNIT "INMOD"
      OPEN(UNIT=INMOD,FILE='MODL.DAT')
C FORMATTING FOR THIS S.R IS AT THE USER'S DISCRETION
      READ(INMOD,11)TITLE
      READ(INMOD,1)NAMVAR(1),C1
      READ(INMOD,1)NAMVAR(2),C2
      READ(INMOD,1)NAMVAR(3),C3
   11 FORMAT(A72)
    1 FORMAT(A6,F20.3)
    7 FORMAT(' ',A6,2X,F20.3)
C  PRINT OUT THE DATA:
      WRITE(IPRNT,22)TITLE
   22 FORMAT('1',A72,/)
      WRITE(IPRNT,7)NAMVAR(1),C1
      WRITE(IPRNT,7)NAMVAR(2),C2
      WRITE(IPRNT,7)NAMVAR(3),C3
      WRITE(IPRNT,3)INMOD
    3 FORMAT(' MODEL NON-SENS. DATA READIN COMPLETED ON UNIT'
     *,' ',I2)
C   CLOSE THE FILE
      CLOSE(UNIT=IMOD,STATUS='KEEP')
      RETURN
      END
```

Data File: MODL.DAT

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT
 C1      1.5
 C2      2.0
 C3      50.
```

Fig. 5.2. Example of user-provided RDMODL subroutine and input data set thereto.

of the user's code. The major changes that must be made to the user's code are as follows:

1. These COMMON blocks must be added to MODEL and many of its constituent SUBROUTINES:

   /MASTER/ The user-defined variable names should be used. Each name corresponds directly to a particular Y(I) variable (element in the Y-vector) in the ORMONTE driver. If the user's model has several subroutines, the INCLUDE statement can be used to place COMMON /MASTER/ near the beginning of each routine while only having to type the COMMON statement once. This FORTRAN-77 feature is especially helpful if a large number of sensitivity variables are used, since it reduces the chance for COMMON variable alignment problems and misspellings.

   /USRMOD/ This COMMON allows transfer of the user's nonsensitivity data from SUBROUTINE RDMODL.

   /OPTION/ This COMMON transfers the unit numbers needed for output (WRITE) statements.

   /PRINT/ Print control information is transferred by COMMON/PRINT/ ITER,INTER,IPR,IFLAG. If an MVSS is requested, the user will probably not want a full MODEL output for each of the hundreds of iterations which are calculated. The following string should precede every WRITE statement in MODEL in order to suppress printout:

   ```
   IF(ITER.LE.IPR)
   ```

   where:

   ITER = The number of the current Monte Carlo iteration
   IPR = The maximum number of iterations for which full printout from MODEL is desired (can be zero)
   INTER = An extra variable saved for special uses (e.g., the programmer could print out every 50th iteration if the proper tests were inserted in MODEL)
   IFLAG = The flag used to designate the WRITE unit for SUBROUTINE WRMAST.

A schematic of an example MODEL, its source listing, and its /MASTER/ COMMON block (in an INCLUDE file) are given in Fig. 5.3 and are used to illustrate use of this code. It is a very simple and hypothetical economic model with 11 input variables and 3 output figures of merit. This ORMONTE driver, however, can be used with very large codes which become SUBROUTINE MODEL. Martin Marietta Energy Systems has used the ORMONTE driver on 10,000+ line codes where each iteration might require optimization procedures within MODEL and several seconds of CPU time on a CRAY-XMP or a few minutes of CPU time on other mainframes.

ORNL—DWG 89—4201 ETD (PART A)

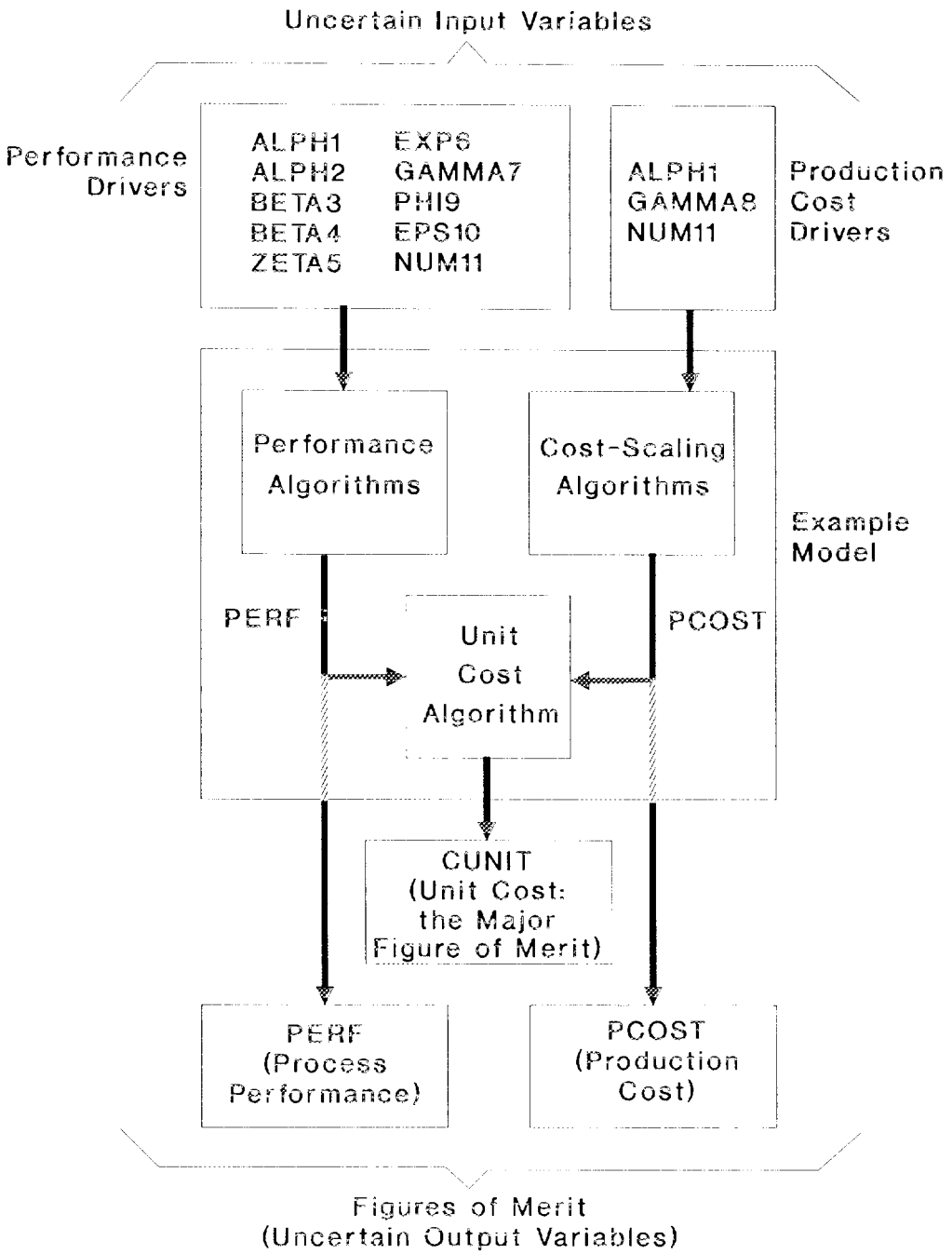


Fig. 5.3. Example and schematic of user-provided MODEL subroutine and INCLUDE file.

ORNL--DWG 89--4201 ETD (PART B)

Source File: MODEL.FOR

```
C ******************************************************
      SUBROUTINE MODEL
C ******************************************************
C EXAMPLE USER'S MODEL TO ILLUSTRATE USE OF ORMONTE CODE
C ******************************************************
C THIS MODEL HAS 11 INPUT AND 3 OUTPUT VARIABLES (FIGURES
C OF MERIT) FOR AN NTOT OF 14. THE CODING IS A FOR A HYPO-
C THETICAL PROCESS WHOSE PERFORMANCE DEPENDS ON 10 UNCER-
C TAIN INPUTS AND WHOSE PRODUCTION COST DEPENDS ON 2 UNCER-
C TAIN VARIABLES. THE DESIRED UNCERTAIN FIGURES-OF-MERIT
C ARE THE PERFORMANCE(UNITS PRODUCED), PRODUCTION COST
C (ANNUAL COST TO PRODUCE THEM), AND THE UNIT PROD'N COST
C (THE QUOTIENT OF PROD'N COST & NO. OF UNITS)
C THE PROCESS AND ALGORITHMS ARE TOTALLY FICTITIOUS.
C THE NON-SENSITIVITY DATA (CONSTANTS C1,C2,&C3 ARE
C READ IN BY S.R. RDMODL PRIOR TO THE FIRST CALL TO
C THIS ROUTINE. /MASTER/ MAKES THE VARIABLE NAMES IN THIS
C ROUTINE EQUIVALENT THE APPROPRIATE Y-ARRAY ELEMENTS
C IN THE REST OF ORMONTE
C ********************************************************
C THE /MASTER/ COMMON BLOCK CONTAINING THE USER'S MODEL
C VARIABLE NAMES RESIDES IN THE FILE MASTER.FOR. WHEN
C THIS FILE (MODEL.FOR) IS COMPILED, MASTER.FOR IS
C AUTOMATICALLY PICKED UP AND INSERTED INTO THE MODEL.OBJ
C FILE. THIS INCLUDE STATEMENT SHOULD ALSO BE USED IN
C SUBROUTINES CALLED BY SUBROUTINE MODEL WHICH UTILIZE
C ANY OF THE /MASTER/ VARIABLES. THE INCLUDE FEATURE
C ELIMINATES THE NEED TO TYPE OUT THE POSSIBLY LARGE
C COMMON/MASTER/ STATEMENT AND ELIMINATES ERRORS DUE TO
C MISALIGNMENT OF COMMON DATA BETWEEN SUBROUTINES.
C*******************************************************
C ALL USER DEFINED SENSITIVITY VARIABLES IN COMMON/MASTER/
C MUST BE REAL*4 !
C*******************************************************
      INCLUDE 'MASTER.FOR'
C I-O AND OPTION DATA READ IN EARLIER ARE PASSED VIA
C THIS COMMON BLOCK:
      COMMON/OPTION/INOPT,INMAST,INMOD,ISVSS,IMVSS,IELAS,
     *IPRNT,IPLOT,IBUG,OPTION,NTOT
C PRINT CONTROL
      COMMON/PRINT/ITER,INTER,IPR,IFLAG
C /USRMOD/ PASSES THE USER'S DATA FROM S.R RDMODL
      COMMON/USRMOD/C1,C2,C3
C ALL INPUTS MUST BE REAL*4
      REAL*4 NUM11
```

Fig. 5.3 (continued)

Source File: MODEL.FOR (Continued)

```
C THIS SIMPLE COST/PERFORMANCE MODEL FOR A FICTITIOUS
C PROCESS CONSISTS OF ONLY THREE ALGORITHMS
C PERFORMANCE (UNITS PRODUCED) AS A FUNCTION OF
C TEN UNCERTAIN VARIABLES
C
      PERF=(C1*ALPH1/BETA3 + C2*ALPH2 + C3*BETA4 +
     ^NUM11*(GAMMA7*EPS10)**(-EXP6))/(ZETA5*PHI9)
C
C THE PRODUCTION COST IS A FUNCTION OF THREE UNCERTAIN
C VARIABLES: NUM11,GAMMA8,&ALPH1
C
      PCOST=C1*NUM11*GAMMA8*ALPH1/200. + 10
C
C CALCULATION OF UNIT PRODUCT COST
C
      CUNIT=PCOST/PERF
C
C FOR AN MVSS THE MODEL KNOWS WHICH ITERATION IS IN
C PROGRESS. IPR IS THE NUMBER OF ITERATION FOR WHICH THE
C USER DESIRES A FULL PRINTOUT OF HIS MODEL'S CALCULA-
C TIONS
C
C ITER IS ALSO PRINTED FOR EVERY 10TH ITERATION AS A DEBUGGING
C AID AND TO ALLOW THE USER TO OBSERVE THE PROGRESS OF A RUN
C ON THE SCREEN IF A VDT IS USED
C
      IF((MOD(ITER,100)).EQ.0)WRITE(IPRNT,22)ITER
   22 FORMAT('-',I5)
      IF(ITER.LE.IPR)WRITE(IPRNT,2)ITER
    2 FORMAT(1H1,'PRINTOUT OF USERS MODEL FOR ITERATION # ',
     *I4,/)
      IF (ITER.LE.IPR)WRITE(IPRNT,3)C1,C2,C3
    3 FORMAT(' CONSTANTS (NON-SENSITIVITY):',//
     *,11X,'C1=',F10.3,/,11X,'C2=',F10.3,/,11X,
     *'C3=',F10.3,//)
      IF(ITER.LE.IPR)WRITE(IPRNT,4)ALPH1,ALPH2,BETA3,BETA4,
     *ZETA5,EXP6,GAMMA7,GAMMA8,PHI9,EPS10,NUM11
    4 FORMAT(' INPUT VARIABLES (UNCERTAIN SENS. VARS.):',//,
     *11X,'ALPH1=',F10.3,10X,'ALPH2=',F10.3,10X,'BETA3=',F10.3,/,
     *11X,'BETA4=',F10.3,10X,'ZETA5=',F10.3,10X,'EXP6 =',F10.3,/,
     *11X,'GAMMA7=',F10.3,9X,'GAMMA8=',F10.3,9X,'PHI9 =',F10.3,/,
     *11X,'EPS10=',F10.3,10X,'NUM11=',F10.3,//)
      IF(ITER.LE.IPR)WRITE(IPRNT,5)PERF,PCOST,CUNIT
    5 FORMAT('  OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):',
     *//,11X,'PERF=',F10.3,10X,'PCOST=',F10.3,10X,'CUNIT=',F10.3,
     *//)
```

Fig. 5.3 (continued)

Source File: MODEL.FOR (Continued)

```
C  FOR THIS EXAMPLE THE VALUES OF PERF,PCOST,&CUNIT WILL HAVE
C THEIR CALC. VALUES STORED IN ELEMENTS Y(12),Y(13), AND Y(14)
C OF THE Y-ARRAY
      RETURN
      END
```

Source File: MASTER.FOR

```
C ***************************************************
C   COMMON BLOCK /MASTER/ IS IN THIS "INCLUDE" FILE
C   "MASTER.FOR". THE COMPILATION STEP PICKS IT UP.
C   THE INCLUDE STATEMENT SHOULD BE INSERTED AT THE
C   BEGINNING OF S.R. MODEL AND MANY OF THE S.R.S
C   THAT IT MIGHT CALL
C ***************************************************
C INPUTS:
      COMMON/MASTER/ALPH1,ALPH2,BETA3,BETA4,ZETA5,
     *EXP6,GAMMA7,GAMMA8,PHI9,EPS10,NUM11,
C OUTPUTS:
     *PERF,PCOST,CUNIT,DUM(136)
C ***************************************************
C   BE VERY CAREFUL OF SPELLING OF VAR. NAMES HERE!
```

Fig. 5.3 (continued)

The sensitivity variables passed through /MASTER/ must be REAL*4 type variables and have names that are eight or fewer characters in length. It is not necessary for the variable names read in earlier as NAMEN(I,J) to be identical to the variable names used in comon block /MASTER/ in subroutine MODEL. The NAMENs are used to label the print output only and are never compared to the variable names in the program. In fact, there is no way to do that; if there were, we would not need to use a common block to set up the variables. It follows from this that the variable names in /MASTER/ may be more than six characters and may even be arrays, although they must be the same type and precision as Y (i.e., REAL*4). The NAMENs should match the variable names if possible, however, understandable abbreviations may be used. Also, the code can be changed to increase the maximum size of NAMEN. Chapter 6 will present considerations needed when constructing a new model to be linked with ORMONTE.

5.5 The SVSS Driver: SUBROUTINE SVSSOP

SUBROUTINE SVSSOP is an automated driver for performing selected one-at-a-time or single variable sensitivity studies (SVSS). This step is usually a precursor to MVSS in that it can be used to establish in advance those input variables which have the most leverage or impact on the desired output figures of merit. The data produced by this subroutine can be saved for later plotting in the form shown by the sample plot on Fig. 5.4. A typical set of plots (for the example model in Fig. 5.3) would be of the form CUNIT vs ALPH1, CUNIT vs ALPH2; CUNIT vs BETA3, etc; where (using the variable names from the example user's model) the unit cost CUNIT is one of the three desired figures of merit, and ALPH1, ALPH2, ... NUM11 are uncertain inputs. [Note that in the driver code, which includes subroutine SVSSOP, CUNIT is really Y(14) and ALPH1 through NUM11 are vector elements Y(1) through Y(11) respectively.]

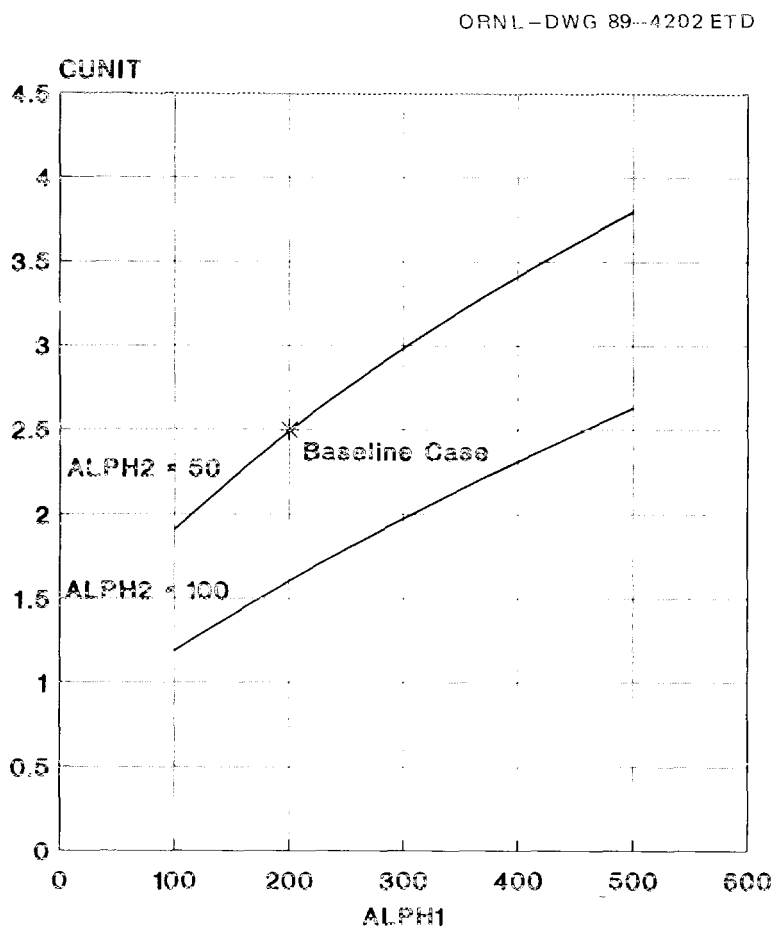


Fig. 5.4. Sample plot made from data generated by ORMONTE SVSS option and the example model.

The user selects one or more values for the given input (ALPH1 for example) and holds all other inputs (e.g., ALPH2 thru NUM11) at their deterministic or baseline values, i.e., at their initialization values read in by RDMAST. The driver then picks up each new value of ALPH1 and runs a deterministic estimate using the new, now-altered Y-vector. New figure of merit values (CUNIT in the example) are calculated corresponding to each new input (ALPH1) value altered from the baseline. SVSSOP has the capability to produce sensitivity tables for up to five outputs or figure of merit variables for each input variable altered. (In the example problem three outputs, PERF (performance), PCOST (production cost), and CUNIT (unit cost) are utilized.)

A special feature of SVSSOP is the capability to produce the data for parametric plots in which one or more input variables are altered from their baseline variables and held constant while another input variable assumes differing values from case to case. (Example: ALPH1 is the sensitivity variable, BETA3 thru NUM11 are held at their baseline values while ALPH2 is held at a constant nonbaseline value for all values of ALPH1. In essence ALPH2 is a parameter as shown in the plot on Fig. 5.4. The upper CUNIT versus ALPH1 curve is for ALPH2 held at its baseline value and the lower curve represents CUNIT versus ALPH1 for ALPH2 at a larger, nonbaseline value. SVSSOP can accommodate several such parametric curves on a given plot if the input instructions are correctly followed.

By stacking cases, several SVSS output tables can be produced in one batch execution of ORMONTE. From these tables, plots similar to Fig. 5.4 can be prepared using "Chartmaster," "Harvard Graphics," "Lotus," or other graphics software. (ORMONTE itself is not a plotting routine. The plots in this document were prepared using "Harvard Graphics.")

5.5.1 Preparation of SVSSOP input data

SVSSOP does not have a read subroutine separate from itself. As the data are read on unit ISVSS (from the dataset SVSS.DAT) it performs the SVSS for each case (or plot) before moving on to read another case (Table 5.3 is a sample input data table). The input cards (lines of data) are as follows:

Card 1: A dataset title card which can have up to 72 character positions (variable name: TITL, type: CHARACTER*72).

Card 2: Descriptive data for the first plot. For the first plot the first sensitivity variable (ALPH1 in the example) would be the abcissa variable. NUMCRV gives the number of curves which will appear on that plot. If NUMCRV > 1 a parametric study as described in paragraph above is to be performed. If NUMCRV=1 the only variable which will assume nonbaseline values is the abscissa variable (ALPH1 in the example) and only one curve would appear on a plot. For the data in Table 5.3 the data for two sensitivity curves on one plot (Fig. 5.4) will be generated for the SVSS on variable Y(1) or ALPH1 [NUMCRV = 00002].

Table 5.3. Sample input data set for SUBROUTINE SVSSOP

	Card #								
	1	TITLE: SVSS DRIVER TEST CASE DSN=SVSS.DAT							
Plot 1	2	00002	003	12	13	14			Curve 1
	3	001	ALPH1	06	100.	150.	200.	300.	
	4				400.	500.			
	5	000							
	5A	001							Curve 2
	6	002		100.					
Plot 2	2	00001	003	12	13	14			one curve per plot
	3	002	ALPH2	04	25.	50.	75.	100.	
	4								
	5	000							
Plot 3	2	00001	003	12	13	14			one curve per plot
	3	003	BETA3	04	6.	8.	10.	12.	
	4								
	5	000							
Plot 4	2	00001	003	12	13	14			one curve per plot
	3	004	BETA4	05	.75	.9	1.00	1.1	
	4				1.3				
	5	000							
Plot 5	2	00001	003	12	13	14			one curve per plot
	3	005	ZETA5	05	.75	0.9	1.	1.1	
	4				1.2				
	5	000							
Plot 6	2	00001	003	12	13	14			one curve per plot
	3	006	EXP6	05	0.5	.75	1.0	1.5	
	4				2.0				
	5	000							
Plot 7	2	00001	003	12	13	14			one curve per plot
	3	007	GAMMA7	05	5.	6.5	8.	9.5	
	4				11.				
	5	000							
Plot 8	2	00001	003	12	13	14			one curve per plot
	3	008	GAMMA8	06	5.	7.5	10.	14.	
	4				18.	22.			
	5	000							
Plot 9	2	00001	003	12	13	14			one curve per plot
	3	009	PHI9	05	5.	12.	18.	24.	
	4				30.				
	5	000							
Plot 10	2	00001	003	12	13	14			one curve per plot
	3	010	EPS10	05	0.8	0.9	1.0	1.1	
	4				1.3				
	5	000							
Plot 11	2	00003	003	12	13	14			Curve 1
	3	011	NUM11	05	0.9	0.95	1.	1.08	
	4				1.15				
	5	000							
	5A	001							Curve 2
	6	002		100.					
	5A	002							Curve 3
	6	001		300.					
	7	002		100.					
	EOF	00000							

NFOM is the number of figures of merit (ordinates) which will be tabulated as a function of the abcissa variable. (In the example there are three figures of merit (FOMs), PERF, PCOST, and CUNIT. In this case NFOM = 003, and all three will be tabulated as a function of ALPH1. (For printer width reasons NFOM is limited to 5.) The user's MODEL may have more than 5 FOMs, but only 5 can be printed on a summary table by SVSSOP for a given case.

INDFOM(I) is the pointer to the NFOM variables to be printed. Each INDFOM represents the Y-vector index of the desired FOM. In the example problem where NFOM=3, PERF=Y(12), PCOST=Y(13) and CUNIT=Y(14); therefore INDFOM(1)=12, INDFOM(2)=13, and INDFOM(3)=14. Up to five INDFOM elements can be specified.

Card 3: This card contains the data for the sensitivity input variable which would appear on the abscissa of a plot.

INDEX is the Y-vector index of that variable. [For the example ALPH1=Y(1); therefore INDEX=001.]

NAMEN is the user's name for the input variable (ALPH1 in the example problem) which can have six or less CHARACTER*1 characters.

NPTS is the number of different Y(INDEX) values or number of abscissa points which will appear on each curve. (In the example problem, six different values for ALPH1 are submitted to MODEL, and six values for the FOMs will be calculated.) Up to eight points can appear on each curve or in each table [NPTS=06].

YREAD(I) is the "points data" or abscissa array containing the 1 to 8 different values of the input variable Y(INDEX) to be submitted by SVSSOP to MODEL. In the example problem NPTS=6; therefore YREAD(1) through YREAD(6) are read in by SVSSOP to replace the baseline value of ALPH1 for each case. Card 3 can accommodate the first four YREAD values.

Card 4: Contains the values for YREAD(5) through YREAD(8). This card is a blank line if no values for YREAD(5 to 8) are used.

Card 5 and 5A: NUMCHG is used mainly for parameter definition for those cases in which more than one curve appears on a FOM vs. Y(INDEX) plot. NUMCHG tells SVSSOP how many input variables in the Y-vector will be altered to nonbaseline values for the requested curves. Usually no more than two Y(I) values (I cannot equal INDEX) are altered for a parametric study. Consider the fifth card in the first plot of the example problem (Table 5.3). NUMCHG=000 on card 5 for the first curve in the ALPH1 sensitivity study, i.e., no Y(I) values other than Y(1) are altered from their baseline values. Card 5A tells the code that for the second curve, one variable Y(K), for which K≠1, will have its value altered, i.e., [NUMCHG = 001].

Card 6 through Card (5+NUMCHG): For each of the NUMCHG Y-array values which are altered the following must be provided:

IND(I): The Y-vector index of the parameter to be altered. IND cannot be equal to INDEX. Card 6 contains IND(1) and Card (5+NUMCHG) contains IND(NUMCHG). In the example problem (plot 1, curve 2) ALPH1 is the sensitivity variable and ALPH2 is the parameter; therefore, since ALPH2=Y(2), IND(1)=002.

VAL(I): This is the nonbaseline value assumed by Y[IND(I)] for this particular curve. For the example problem (plot 1, curve 2), Y [IND(2)]=100.

Next Card: If NUMCRV.GE.3, a sequence of cards starting with a card similar to card 5A but with new data is submitted. This will add a new curve to the plot data. Plot 11 of the example problem (Appendix D) shows such a case.

Once all the data for the NUMCRV curves on a particular plot have been read and the cases calculated, a new sequence of cards can be read which will produce plot data for the next input sensitivity variable. Each new sequence (or plot) will start with a card similar to Card 2 above. Every second or greater curve within a given plot will begin with a card similar to 5A above. At the end of all the stacked data an end of file (EOF) card typed "00000" will cause the program to exit SVSSOP and return to the MAIN.

5.5.2 SVSSOP algorithms

SVSSOP does not perform any mathematically complex tasks. It is merely a driver which keeps track of variable indices and names, and it automates what would be a tedious and cumbersome task involving many separate runs of MODEL. Several DO loops accomplish the driver task and the logic should be easily understood from examination of a source listing. Table 5.4 shows the example problem unit IPLOT output data used to produce the SVSS plot in Fig. 5.4. It should be noted that numerical data produced by MODEL will be printed for every point used to produce the Fig. 5.4 plot. For unit IPRNT output, SVSSOP prints a header for each MODEL output figure of merit, identifying which plot, curve, variable, and point value the case represents (see sample output in Appendix D). The plot data summary (Table 5.4) is printed on unit IPLOT and is stored in a file FORT15 for the example problem. Since the plot data format is uniform, a graphics program could be written to automatically access this data and produce graphs similar to that in Fig. 5.4. The user should note that for every new plot any previously altered Y-vector values are reset to their baseline values prior to resubmission to MODEL.

5.6 A Driver for Elasticity Analysis: SUBROUTINE ELAST

In SVSSOP above the user chooses the nonbaseline values or points for whatever Y-vector input variable he is examining. In ELAST the ORMONTE driver chooses two values for the input Y-vector variable; one 0.5% above the baseline

Table 5.4. Portion of SVSS summary table produced by SUBROUTINE SVSSOP

INDEX,NPTS,NUMCRV
1 6 2
OF PARAMETER CHANGES= 0

Curve 1

PT#	ALPH1	PERF	PCOST	CUNIT
1	1.00000E+02	9.17361E+00	1.75000E+01	1.90765E+00
2	1.50000E+02	9.59028E+00	2.12500E+01	2.21579E+00
3	2.00000E+02	1.00069E+01	2.50000E+01	2.49827E+00
4	3.00000E+02	1.08403E+01	3.25000E+01	2.99808E+00
5	4.00000E+02	1.16736E+01	4.00000E+01	3.42653E+00
6	5.00000E+02	1.25069E+01	4.75000E+01	3.79789E+00

Plotted Data (Fig. 5.4)

Baseline Point

OF PARAMETER CHANGES= 1
2 ALPH2 100.000

Curve 2

PT#	ALPH1	PERF	PCOST	CUNIT
1	1.00000E+02	1.47292E+01	1.75000E+01	1.18812E+00
2	1.50000E+02	1.51458E+01	2.12500E+01	1.40303E+00
3	2.00000E+02	1.55625E+01	2.50000E+01	1.60643E+00
4	3.00000E+02	1.63958E+01	3.25000E+01	1.98221E+00
5	4.00000E+02	1.72292E+01	4.00000E+01	2.32164E+00
6	5.00000E+02	1.80625E+01	4.75000E+01	2.62976E+00

value and one 0.5% below the baseline value. Subroutine ELAST of ORMONTE then calculates the resulting percent changes in the desired figures of merit and reports the elasticity [also known as the coefficient-of-sensitivity (COS)] which is calculated as the average percent change in output divided by the average percent change in the input. For the example problem this would be the same as:

[PERF/PERF(base)]/[ALPH1/ALPH1(base)] .

The ELAST driver performs this calculation for all of the input variables in the Y-vector. For a given input variable Y(I), elasticities can be calculated for up to five FOMs selected by the user. The resulting table of elasticities or table of COSs is useful for studying the relative leverage of model inputs on the important outputs. ELAST can help the analyst determine if his model is working correctly and if figures of merit are behaving as expected when a deviation in an input variable is

experienced. This step and the SVSS procedure discussed earlier are useful precursors to MVSS since they readily identify the inputs having the most leverage on the model outputs. Figure 5.5 shows an elasticity plot for one of the FOMs, (PERF), in the example problem.

ORNL-DWG 89-4903 ETD

Data: ELAST.DAT

```
TITLE: SAMPLE ELASTICITY ANALYSIS DATA DSN=ELAST.DAT
# OF INPUTS: [ 11]    # OF FIGS OF MERIT: [ 3]
/MASTER/ Y( ) INDICES FOR FIGS OF MERIT:12  13  14  00  00
```

PERF Unused Indices

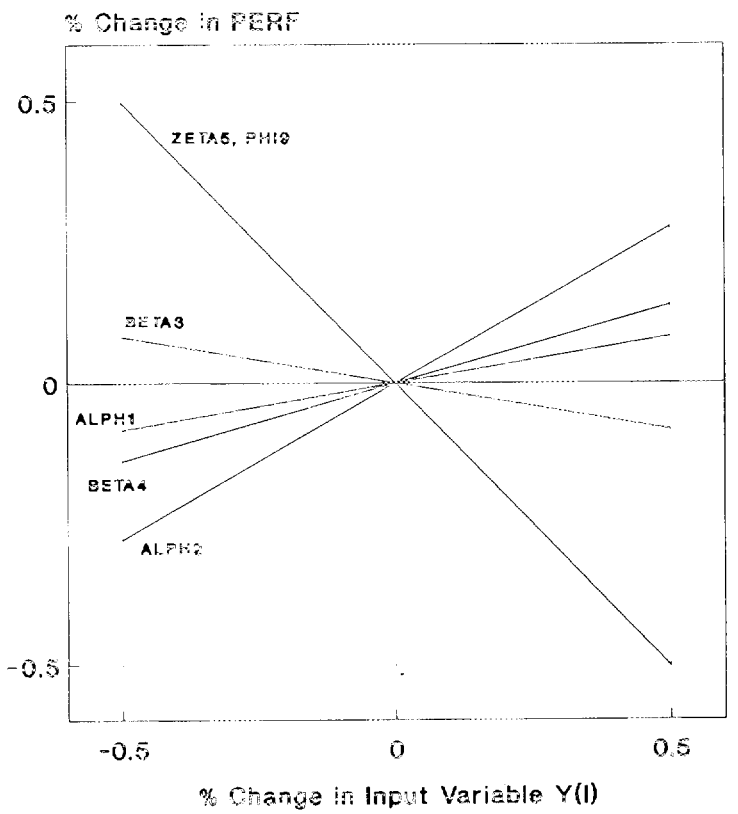


Fig. 5.5. Sample plot made from data generated by ORMONTE ELAST option and the example model.

Note that the elasticity is calculated only around the baseline input variable value. If the figure of merit is not linear with that input, a different elasticity would appear about a nonbaseline input value. The analyst should be aware of nonlinear "cliff" situations in his model, i.e., input ranges for which the elasticity rises or falls abruptly.

5.6.1 Inputs to the ELAST driver

Three lines of data are needed for the ELAST driver. These are stored in file ELAST.DATA and are read on unit IELAST. Each card is described below and a sample data set is shown on Fig. 5.5.

Card 1: The data set can have a 71-character title. The variable name is TITL and is of type CHARACTER*71.

Card 2: NSIN: The number of input variables in the Y-vector (11 in the example problem)
NFOM: The number of figures of merit for which an elasticity will be calculated. (NFOM.LE.5) NFOM = 3 in the example problem)

Card 3: INDFOM(I): The Y-vector for the NFOM figures of merit. Up to five indices may be specified. [For PERF, PCOST, and CUNIT in the example problem, INDFOM(1), INDFOM(2), and INDFOM(3) equal 12, 13, 14 respectively].

5.6.2 Brief Description of the ELAST Driver Algorithms

ELAST considers deviations in both inputs and calculated model outputs from their base case values. The base case inputs were read in earlier by RDMAST, and the base case or deterministic outputs were calculated by the first call to MODEL by the MAIN. COMMON block/MASTER/ transfers these values to this routine. A new vector called YBAS(150) accepts these baseline Y-vector values for future reference by ELAST. IPR is set to -1 to prevent the detailed printout of MODEL results for every nonbaseline case.

After ELAST reads the data, a DO loop is entered in which the following sequence is performed on all NSIN (total number of inputs) input variables in the Y-vector:

1. A slightly lower-than-baseline value for input Y(J) is calculated

   YL(J) = 0.995*YBAS(J) (-0.5% deviation)

   If YBAS(J) = 0. the value YL(J) = -.00005 is substituted.

2. Y(J) is now set to YL(J) and MODEL is called.

3. A DO loop (I=1, NFOM) with the following statement now captures the slightly deviated FOMs:

   YFOML(I) = Y[INDFOM(I)]

4. The same procedure is repeated for a 0.5% upward deviation in Y(J); thus, creating YG(J) and YFOMG(I).

5. The differences or deviations are now calculated:

   Inputs: DELYL=YBAS(J)-YL(J)
   DELYG=YG(J)-YBAS(J)
   Outputs: DLFOML(I)=YBAS[INDFOM(I)]-YFOML(I)
   DLFOMG(I)=YFOMG(I)-YBAS[INDFOM(I)]

6. The fractional deviation in the inputs are now calculated:
   XFRAL=DELYL/YBAS(J)
   XFRAG=DELYG/YBAS(J)

If YBAS(J)=0, a warning message for division by zero is printed and no elasticity is calculated. If YBAS(J)≠0 the average fractional input deviation is calculated:

XFRA=(XFRAL+XFRAG)/2.

7. The same procedure is used to calculate fractional deviations in the figure(s) of merit or outputs:

   YFRAL(I)=DLFOML(I)/YBAS[INDFOM(I)]
   YFRAG(I)=DLFOMG(I)/YBAS[INDFOM(I)]
   YFRA(I)=[YFRAL(I)+YFRAG(I)]/2.

8. The elasticity or COS is calculated for each of the I=1, NFOM figures of merit:

   ELAST(I)=YFRA(I)/XFRA

9. The average slopes for each FOM are also calculated as follows:

   Lower: SLOPEL(I)=DLFOML(I)/DELYL
   Upper: SLOPEG(I)=DLFOMG(I)/DELYG
   Average: SLOPE(I)=[SLOPEL(I)+SLOPEG(I)]/2.

Figure 5.6 shows graphically how the elasticities and slopes are calculated for one FOM and one input. Appendix D contains a partial output of ELAST for the example problem. Figure 5.5 displays in graphical form the partial results of ELAST for one figure of merit, PERF, in the example problem.

ORNL-DWG 89-4904 ETD

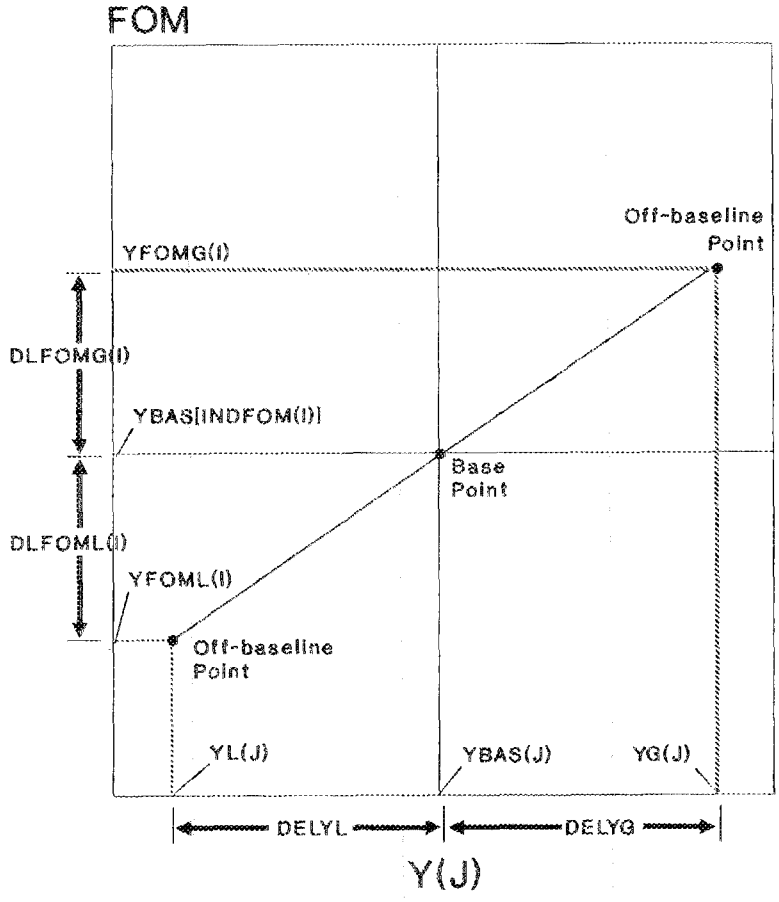


Fig. 5.6. Graphical interpretation of ELAST algorithms.

5.7 SUBROUTINE MVSSOP: The Monte Carlo Driver for Multivariable Uncertainty Analysis

MVSSOP has the capability of handling up to 150 (input plus output) variables and ten different types of probability distributions for the inputs. Basically MVSSOP performs the following tasks in this order:

1. It calls SUBROUTINE RDMVSS to read the parameters defining the type and parameters of each input's probability distribution.
2. It calls SUBROUTINE PRINTX to write out the information read by RDMVSS.
3. For each iteration it calls SUBROUTINE XVALUE, which samples the input distributions (via a random number generator), and fills the input positions of the Y-vector with an ensemble of random values.
4. It submits these values to SUBROUTINE MODEL, which calculates the figures of merit and fills the output positions of the Y-vector.

5. It temporarily stores the Y-vector for every iteration until all iterations are complete; the resulting Y-array contains M Y-vectors, where M is the number of iterations.
6. Using the Y-array data it performs a statistical analysis on every input and output variable.
7. It divides the range of occurrence for each uncertain variable into equal size bins and produces frequency histograms for all variables.
8. It instructs the printer to write out the histograms and statistical analysis for each variable.

The subroutines called by MVSSOP and the various steps listed above will be described in greater detail below.

5.7.1 SUBROUTINE RDMVSS

For deterministic estimates and SVSS studies a single value for each of the Y-vector variables is specified in the input data. In an MVSS a range of possible values and the probabilities of these values within that range are specified in the input data. The choice of the type probability distribution and the choice of its parameters provide the MVSS Monte Carlo driver with the information needed to assign the probabilities with the desired range. The main function of RDMVSS is to read in the distribution type and parameters for each input variable of interest (which must be one of those in the input portion of the Y-vector).

The MVSS data appear on file MVSS.DAT (example from Appendix D shown on Table 5.5) and are read on unit IMVSS. Instructions for data entry appear on COMMENT statements within RDMVSS; however, the instructions will be repeated here. The following lines of data are required:

Card #1: Up to 72 characters can be provided for a title for the dataset. The variable TITLE is of type CHARACTER*72.

Card #2: NITER: The number of Monte Carlo iterations to be completed by the driver. NITER is a right justified INTEGER variable appearing in columns 8-13 and has a range of 1 to 99999. If a negative sign appears in column 8 (in front of NITER), printout of input variable probability histograms is suppressed. NITER is 5000 for the example problem.

IPR: The number of desired iterations for which a full printout from the WRITE statements in MODEL on unit IPRNT is produced. For most applications IPR=0. IPR is 5 for the example problem.

INTER: This variable is not presently used. By use of the correct tests involving ITER and INTER, the user could generate full printout for every nth iteration. (ITER is the iteration counter and has a value between 1 and NITER.)

Table 5.5. Sample input data set for SUBROUTINE MVSSOP

```
Card #
  1   TITLE: 14 VARIABLE MVSS TEST CASE
  2   NITER [  5000]   IPR [   5]  INTER [   1]
  3   DISTRIBUTION DATA:
  4   INDEX  XNAME  JTYPE     X1          X2          X3          X4
  5       V      V      V           V           V           V           V
      1 ALPH1      -1      200.        100.        500.
      2 ALPH2       0      50.
      3 BETA3      -9      6.          12.
      4 BETA4      -2      1.          .08
      5 ZETA5      -3      1.          .08         1.2
      6 EXP6       -4      1.          0.5         2.
      7 GAMMA7     -6      5.          7.          10.         11.
      8 GAMMA8     -5      10.         0.20
      9 PHI9        5
     5.      5.        .1        .2        .4        .2        .1
     10 EPS10      -7      1.          .08         .8
     11 NUM11      -8      1.          .08         .9          1.15
     12 PERF      -99
     13 PCOST     -99
     14 CUNIT     -99
```

Input var. designators

Output var. designators

IPR and INTER are passed to MVSSOP by COMMON/PRINT/. NITER is passed by COMMON/PARAM/. The CHARACTER*n variables AL1, AL2, and AL3 are optional and are used to provide mnemonic variable name information for this data card. Their use simplifies the editing process when altering an existing MVSS data set.

Cards #3-5: These cards contain character data that are not used by the code but are useful for preparing and editing MVSS data sets. Card 3 prints out the title "Distribution Data." Card 4 prints out column headings for the actual distribution parameter cards (Cards 6 and above). Card 5 contains "V" markers to assist the preparation and editing of the following cards. Once these cards are typed (see example data set) they need not be altered.

Input Variable Designation Cards: The following instructions apply to each input variable (one variable per card). These cards contain the parameters that define the probability distributions. A more detailed discussion of each type of distribution will appear in subsection 5.7.1, and Appendix A. COMMON/PARAM/ passes this data to the other parts of the Monte Carlo driver which need it.

INDEX: The Y-vector index for the variable whose distribution parameters will appear on this card. The number of cards of this type must equal the number of sensitivity input variables. INDEX can have an integer value from 1 to 149 and occupies columns 3 through 5 (right justified). These cards should be ordered sequentially with the input variable Y-indices.

XNAME: The user-chosen name (not value) for the input variable Y(INDEX). XNAME should be the same as the variable name NAMEN (I,J) read by SUBROUTINE RDMAST (six characters or less). The same comments appearing in the last paragraph in Section 5.4 apply to XNAME as well as NAMEN. XNAME occupies columns 7-12 (right justified).

JTYPE: JTYPE identifies the type of distribution whose parameters are to follow. If JTYPE is negative, a parameter-defined distribution is called. For such distributions, mathematical procedures in SUBROUTINE XVALUE are invoked to build up the probability/Y-value relationships from the parameters. In essence these are what might be called "canned" or pre-packaged distributions. The integer value of JTYPE, if a negative value from -1 to -9, designates the functional form of the probability function as shown in Table 5.6 below. If JTYPE=0, Y(INDEX) will assume a constant value, and no probability distribution is sampled. Use of this JTYPE=0 feature allows the user to input a deterministic value for Y(INDEX) different than that read in by SUBROUTINE RDMAST for the baseline case. If JTYPE is a positive integer, the probability distribution is a user-constructed or empirical histogram with the positive value of JTYPE indicating the number of histogram boxes of equal width. In this case the probability/Y-value relationship is directly supplied in the data rather than being calculated from distribution parameters. Up to twenty histogram boxes can be utilized with additional cards needed to input the probability associated with each box. JTYPE occupies columns 18-20 (right justified).

X1, X2, X3, and X4: These values are the parameters which mathematically define the distribution to be sampled for variable Y(INDEX). Table 5.6 shows each parameter depending on the value of JYTPE. (Figure 5.7 shows the geometric forms for each distribution and JTYPE corresponding to the data in Table 5.5.) These parameters can be entered in floating point or exponential format using columns 12 characters wide (Cols. 21-32; 33-44; 45-56; 57-68). Use of floating point entries avoids the problem of right justification. Card 5 has "v" pointers which assist in keeping these entries within their designated columns. Note that for some distributions, X1, X2, etc., may actually designate locations on the abscissa or Y(I) axis. With pdfs, do not think of Y(I) and X as a set of rectangular co-ordinates.

Output Variable Designator Cards: After all input distribution parameter cards have been read, RDMVSS requires definition of the output variables or figures of merit for which probability histograms are to be plotted. A card similar to an input card accomplishes this task. The following variables are read:

INDEX: The Y-vector index of the output variable (Columns 3-5).

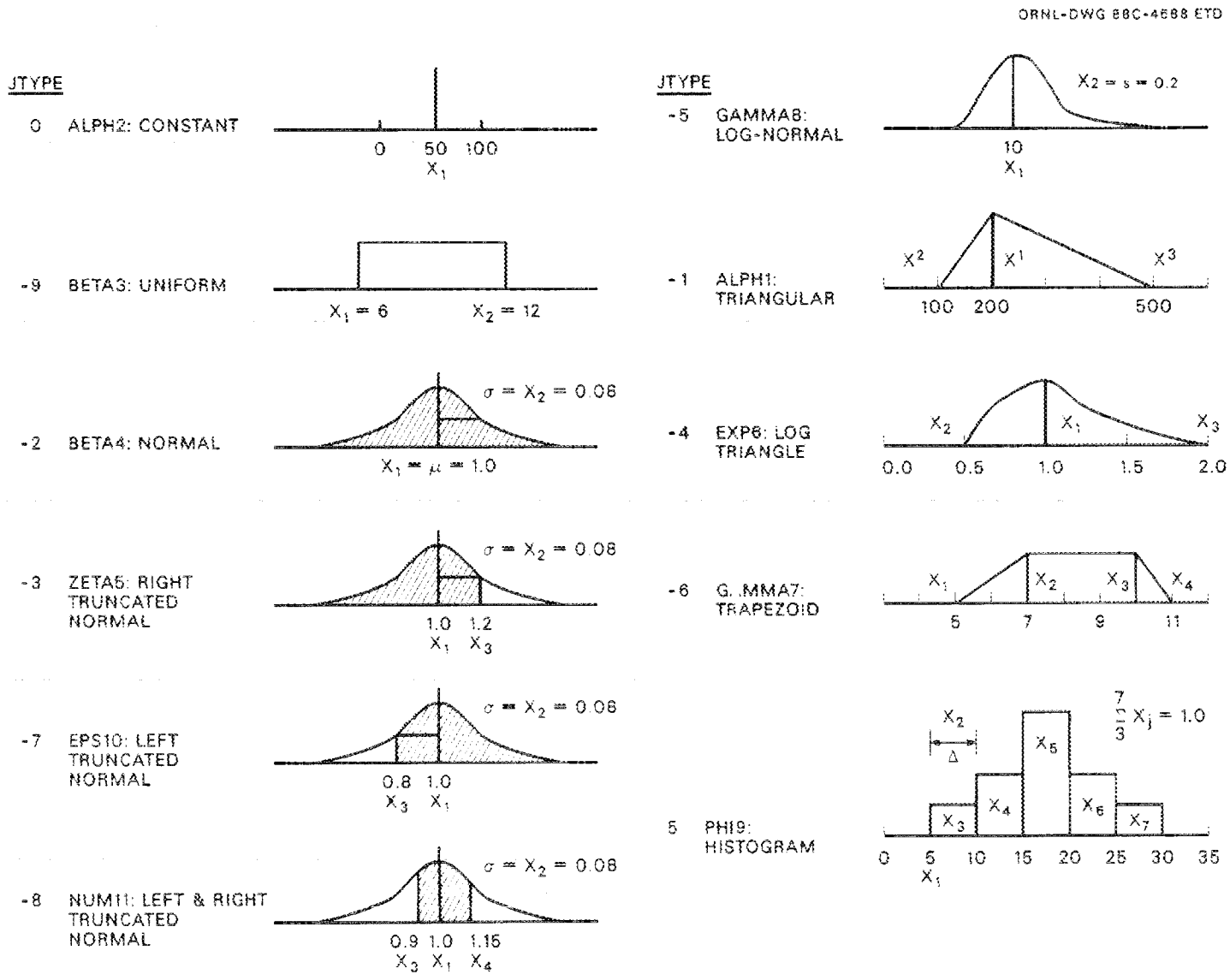


Fig. 5.7. Available input uncertainty distributions constructed from statistical parameters used for the example model.

Table 5.6. Defining parameters for non-histogram input distributions

Type of Distribution	JTYPE	X1	X2	X3	X4
Constant	0	Value	-	-	-
Triangle	-1	Mode	Min	Max	-
Normal	-2	Mean		-	-
Truncated normal with upper limit	-3	Mean		Max	-
Log-transformed triangle	-4	Mode (=Median)	Min	Max	-
Log-transformed normal	-5	Mean	S-parameter	-	-
Trapezoid	-6	Min	Vertex-1	Vertex-2	Max
Truncated normal with lower limit	-7	Mean		Min	
Truncated normal with upper and lower limits	-8	Mean		Min	Max
Uniform	-9	Min	Max		

XNAME: The user-designated output variable name (Columns 7-12).

JTYPE: Set JTYPE=-99 to designate an output variable whose probability histogram is to be printed.

JTYPE: Set JTYPE=-88 to designate an output variable whose probability histogram is not to be printed.
(Columns 18-20)

No distribution parameters are needed on these cards, since for these variables distributions will be built up by storing and sorting the output of the numerous iterations of MODEL. The number of output cards should equal the number of output variables, and the cards should be in sequential order with the Y-indices of the outputs. The program knows the number of inputs and number of outputs by counting the number of cards for both.

Histogram Input Data Cards: If JTYPE is positive on an input variable definition card a box type histogram with JTYPE boxes is designated. The card(s) following this card should contain the following data:

XDATA(ICNT): The minimum possible value of input Y(INDEX), i.e., the value designating left side of the leftmost histogram box.

XDATA(ICNT+1): The width of each of the JTYPE histogram boxes in the same units as Y(INDEX).

XDATA(ICNT+2) through XDATA(IEND): The relative probability fractiles associated with each of the JTYPE boxes.

The format for XDATA(I) input is 7E10.3 (column width of 10). The user can use F-format and reduce the chance of a right justification error. More than one data card of this type may be required for histograms with more than five boxes. It is suggested that no more than 20 boxes be specified.

ICNT: A counter used to index all distribution parameters for the XDATA array for both canned and histogram distributions. The XDATA array is used to transfer all distribution parameters to SUBROUTINE XVALUE. The LDTYPE array is a pointer array used to correctly associate XDATA elements with the correct distribution forms as identified by JTYPE. LDTYPE is also used to transfer data to XVALUE. NAME is used to transfer the user-defined variable names. All of these transfer arrays appear in COMMON/PARAM/.

As the MVSS.DAT dataset is read, RDMVSS also prints out the card images. At the end a message confirming the completed read of all data is printed.

5.7.2 Input Distribution Descriptions and Parameters

Unless noted, negative numbers are permissible within the range of a distribution. Exceptions are described below for each distribution type.

The triangular distribution (JTYPE=-1)

The three parameters defining the triangular distribution are the mode value (X1), the leftmost value or minimum (X2), and the rightmost value or maximum (X3). No value of Y(INDEX) less than X2 or greater than X3 will ever be used in a sample. Values of Y(INDEX) in the vicinity of the X1 mode will be selected more frequently than those near the endpoints. The triangular distribution has the advantage that any degree of skewness (see Appendix A for definition of skewness) can be easily represented by the choice of mode value X1 relative to the two endpoint parameters X2 and X3. For an isosceles triangle the mode is equal to the median, since 50% of the area under the triangle lies on either side of X1. The triangular distribution is popular because the parameters X1, X2, and X3 can often be thought

of in terms of most likely, most pessimistic (or optimistic), or most optimistic (or pessimistic), respectively.

The normal distribution (JTYPE=-2)

The two parameters defining the normal distribution are the mean (X1 or μ) and the standard deviation (X2 or σ). Since the domain of the normal distribution extends to infinity in both directions, some truncation of this distribution must be used for this computer application. The return values are limited to a range of plus or minus six sigmas (σs or standard deviations) from the mean. To prevent the occurrence of negative return values within this range, the lowest value a positive return value can assume is the mean/50 (X1/50 or μ/50). Other truncations of the normal distribution's tails are possible if the options JTYPE=-3, -7, or -8 are used. The 6 sigma criterion used for this option (JTYPE=-2) includes 99.99% of the points which would be represented by a distribution where no truncation of tails occurs. Negative values for X1 or X2 are not permitted. The mathematical methodology for generating the normally distributed random numbers is described in the section on SUBROUTINE XVALUE.

The right-truncated normal distribution (JTYPE=-3)

This variation of the normal distribution requires three parameters for its definition: X1=mean; X2=standard deviation; and X3=input variable value for which right tail is truncated. Note that X1 and X2 have the same definitions as for the JTYPE=-2 option above. X3 must lie within +6 sigmas of the mean (X1) as above. Negative values for X1 through X3 are not permitted. If a normally distributed input value greater than X3 is returned from subroutine XVALUE, it is discarded and a new random input value is chosen.

The log-triangle distribution (JTYPE=-4)

This variation of the triangle distribution results when the logarithm of the input variable on the abscissa is assumed to be triangularly distributed, i.e., the logs of the triangle base lengths to the left and right of the mode are used in the calculation of a return value. As with the triangular distribution (JTYPE=-1), the three parameters X1, X2, and X3 are the mode, minimum, and maximum input variable values respectively. This distribution has the property that the median is forced closer to the mode, i.e., the location of the most probable value tends to split the area beneath the curve into two quasi-triangles of nearly equal area. The sample mode obtained from this distribution differs somewhat from the parameter X1 depending on the skewness, i.e., the relative distance of X2 from X1 compared to the distance of X3 from X1. If the value of X2 = 1/X3 the mode of the resultant log triangle sample will approximate 1.0. This is a useful distribution for costs in which the high end value of the range is further from the mode than the low end value of the range, but where the user wishes that the mode value split the sample such that 50% of the sampled points (costs) lie above the mode and 50% of the points (costs) lie below the

mode. Figure 5.7 shows the shape of such a typical rightward skewed log triangle distribution.

The log-normal distribution (JTYPE=-5)

The log-normal distribution occurs whenever one encounters a variable which is such that its logarithm has a standard normal distribution. This rightward skewed bell-shaped distribution is often defined by μ_Y and σ_Y^2 parameters in statistics textbooks and has a complicated functional form. For purposes of this code, two alternate parameters are used to define the distribution. X1 is the median value for the input variable, and X2 is the "S-parameter" analogous to the standard deviation but on a logarithmic scale. The S-parameter can be calculated as follows:

$$S = \ln (X1/X_b) ,$$

where X_b is an input value one standard deviation (or 1σ) below the median X1. Note that S is not the same as $\ln \sigma$. Figure 5.7 shows the shape of the log-normal distribution. In the example problem the normal and log-normal distributions each have parameters that have been chosen to generate samples with nearly similar bell-shaped histograms. X1 and X2 may not have negative values.

The trapezoidal distribution (JTYPE=-6)

This distribution combines features of the uniform and triangle distributions. X1 designates the minimum attainable input variable. X4 designates the maximum attainable value. Between values X2 and X3 a value for the variable Y(I) has an equal chance of being drawn, i.e., the distribution is uniform between these values. The following parameter value ordering must be used: X4 > X3 > X2 > X1.

The left-truncated normal distribution (JTYPE=-7)

This distribution is similar to JTYPE=-3 above except that the parameter X3 represents the minimum value attainable by the input variable. X3 must lie within 6 standard deviations (X2s) to the left of the mean, X1. X1 through X3 must be positive.

The left and tight truncated normal distribution (JTYPE=-8)

This distribution combines the features of JTYPE=-3 and JTYPE = -7 above. X3 is the minimum value for the input variable, and X4 is the maximum value. X1 through X4 must be positive.

The uniform distribution (JTYPE=-9)

If an input value has the same probability of occurrence over its entire range, the uniform distribution should be utilized. Two parameters, X1 and X2, define it;

these are merely the endpoints of the variable's allowable range. X1 is the minimum and X2 is the maximum.

The empirical or histogram distribution (JTYPE= +N)

The user must determine the relative probabilities for each of the N boxes constituting the histogram. These boxes must be of equal width, and the relative probabilities or fractiles must sum to 1.0. The input card description above explains the input parameters and their formatting. Note that this distribution can be used to approximate other distribution forms not included in the "canned" distribution list above. As an example a Weibull distribution could be represented by a series of narrow boxes with heights drawn to approximately fit the shape of a Weibull curve. Dividers and a ruler can be used to determine the relative probability values. The user should also note that SUBROUTINE RDMVSS scales the relative probability values that are read in so that they sum to 1.0. Figure 5.7 shows a sample histogram with five boxes.

5.7.3 SUBROUTINE PRINTX

SUBROUTINE PRINTX merely prints out a table of the input variable distribution parameters read by RDMVSS. When the probabilities associated with a user-defined histogram (positive JTYPE) are printed, they are printed in cumulative rather than relative probability form. The two large arrays LDTYPE and XDATA are used to order, identify, and transfer the parameter data.

5.7.4 SUBROUTINE XVALUE: Calculation of return values from the input distribution parameters and the random numbers

XVALUE is essentially the heart of the ORMONTE code in the sense that for each iteration it selects the random values for the sensitivity inputs submitted to MODEL. With the assistance of a random number generator, XVALUE makes sure that over the course of a Monte Carlo simulation the input samples actually drawn approximate the distribution forms specified by the parameters read by RDMVSS. The random values for the sensitivity inputs are usually called "return values," and for each Monte Carlo iteration, the input portion of the Y-vector or ensemble of return values is submitted to MODEL for calculation of figures of merit (the output portion of the Y-vector). Figure 4.1 shows this process in schematic form. This mathematical process (using a uniformly distributed random number between 0. and 1.0 in conjunction with distribution-defining parameters to generate return values) is known as inversion. This section will explain how inversion is accomplished for each distribution type. (The generation of uniformly distributed random numbers is discussed in Appendix E.)

A first step in most inversion techniques is to convert the specified relative probability distribution into a cumulative distribution. This task is accomplished by

integration of the function defining the relative probability distribution in terms of Y (the input). The total area under the relative probability curve or pdf (probability density function) must be equal to 1.00. The area from the left endpoint of the pdf to any location Y_i on the variable range beneath the pdf curve envelope (on the abscissa) must be number greater than zero and less than or equal to 1.0. This value is known as the cumulative probability and represents the probability that the variable will have a value of Y_i or less. If the cumulative probabilities are plotted against the variable range, the familiar S-shaped cumulative probability curve results as in Fig. 4.2. The ordinate of this curve has a maximum value of 1.0 and a minimum of 0.0.

The uniformly distributed random number (UDRN) also has a range of 0.0 to 1.0, and a particular UDRN value within this range (e.g., 0.6781) has the same chance of occurrence as any other value (say 0.0231). If we now say that our UDRI value represents a cumulative probability value or cumulative density function value [$cdf(Y_i)$], an association between $cdf(Y_i)$ and an actual value of the input Y_i can be made via the S-curve as shown on Fig. 4.2. If this association process is repeated many times, however, the analyst will soon note that ultimately the values of Y_i chosen, i.e., the return values, will not be uniformly distributed along the abscissa (unless the uniform distribution for Y_i was requested). The distribution of the return (Y_i) values will depend on the shape of the S-curve or $cdf(Y_i)$. Y-values will tend to group more often around the region of the S-curve which is steep and less often around the flatter regions (see Fig. 4.2.) The JTYPE chosen for the input pdf (and its parameters) determine the shape of $pdf(Y_i)$ and therefore also for $cdf(Y_i)$. The UDRNs and $cdf(Y_i)$ assure that over the course of the Monte Carlo simulation, the return value sample histogram approximates nicely the shape of the parameter-defined $pdf(Y_i)$.

Random Number Generation

ORMONTE requires a floating point uniformly distributed random number between 0.0 and 1.0. For mainframe computers, a built-in library function call such as RAN(I) or UNI(I) will supply a new UDRN every time it is called. For Personal Computers such as the IBM XT/AT series, the user must supply his own UDRN generator. The type of microchip used in the machine will affect the UDRN generator algorithms, since these algorithms are based on the overflow characteristics of the PCs internal arithmetic. The UDRN generator supplied in the actual coding (files RANSET and TCTRAN) is written in both FORTRAN and machine language for purposes of CPU time enhancement. It is applicable for an IBM-PC/AT or compatible with an 80287 processor and was written by T. C. Tucker of Martin Marietta Energy Systems. A short description of this UDRN generator and others is given in Appendix E.

Most UDRN generators require use of a "seed" argument to start generating a sequence of numbers. This integer argument (IY in this program) is set once in XVALUE and is regenerated by the UDRN generator algorithms for the second and all subsequent random numbers.

Since the distributions based on variations of the normal distribution have complex functional relationships between PDF(Y), Y, μ, and σ, the algebraic calculation of Y from CDF(X), μ, and σ (i.e. inversion) would be very difficult and probably require an iterative numerical routine. If a normally distributed random number (NDRN) were available, the return value Y could be calculated in terms of the two parameters μ and σ as follows:

$$Y = \mu + NDRN * \sigma$$

where

μ is the mean (X1 in the RDMVSS parameter list)

σ the standard deviation (X2 in the RDMVSS parameter list)

NDRN is essentially the number of standard deviations away from the mean, μ, occupied by the input variable (return value) Y_i. FUNCTION RNOR calculates a NDRN from a series of UDRNs by use of the Central Limit Theorem of Statistics. This theorem states that under rather general conditions, sums and means of samples of random measurements (or values) drawn from a population tend to possess, approximately, a bell-shaped or normal distribution upon repeated sampling. RNOR takes the number -6.0 and adds to it twelve different UDRNs (all, of course, between 0. and 1.0). If this call to RNOR is repeated many times and a large number of UDRNs are available, the NDRNs calculated will have a normal distribution with a sample mean of 0. and all lie between -6.0 and 6.0. It can thus be stated that a given NDRN tells how many standard deviations Y_i lies from a given mean and that Y_i will never be more than six standard deviations away from the mean. This limitation is not a problem since 99.99% of all possible occurrences will be covered. It should be noted that some mainframe computers have a library function capable of supplying NDRNs.

XVALUE logic

For each iteration of a Monte Carlo simulation, MVSSOP calls XVALUE once. Each call to XVALUE generates one input ensemble or partial vector, Y_i (i = 1, NI), of return values which are then transferred to MODEL via the Y-array and COMMON/ MASTER/. A loop using INDEX from 1 to NTOT or NVART (the total number of sensitivity variables) controls the logic of XVALUE. For each input variable INDEX (the Y-vector index) is associated a JTYPE which selects the correct inversion algorithms for the specified distribution. If JTYPE=-99 or -88 is encountered, the variable is an output variable [Y_i ($i \geq NI+1$)] and no inversion is necessary. In the FORTRAN coding for subroutine XVALUE the generic name for the return value being calculated is X. After inversion the particular Y(INDEX) or Y_i being calculated is assigned the value of X and inversion of the next input distribution proceeds. This procedure is repeated until all of the input Y-array positions have been filled with return values. The "X" nomenclature will be used for the

following descriptions, even though we may be describing locations on the Y(I) axis (abscissa) for a particular pdf.

Inversion algorithms

For distributions based on the normal distribution (RDMVSS JTYPE=-2, -3, -7, or -8) the inversion process is simple and is based on use of the normally distributed random number RNOR(IY) or QN. For the untruncated normal distribution (JTYPE=-2) the user-specified parameters XMEAN and SIGMA are used along with QN to calculate the return value X as follows:

X = QN*SIGMA + XMEAN

where

SIGMA = XDATA (ISTART +1) = X2 and
XMEAN = X1

In order to keep X positive, a minimum value of XMEAN/50 is assigned to X if the above result is negative. For the truncated normal distributions for which the user defines the truncation values (JTYPE=-3, -7, or -8) simple tests are applied to the X as calculated above. If X falls outside the truncation values, a new QN is drawn, and the X calculation and test are repeated. This procedure ensures that all Xs fall within the truncation boundaries.

For the log-normal distribution (JTYPE=5), the user defined parameters are SIGMA (the S-Value) and XMEAN (XMEAN in this case is really the desired median). The return value X is calculated as follows:

X= XMEAN*EXP(QN*SIGMA) .

No truncation options are available with this distribution.

The triangular distribution (JTYPE=-1) is defined by three parameters; a minimum value [XDATA(ISTART+1)] or X2, a maximum value [XDATA (ISTART+2)] or X3; and the mode value [XDATA(ISTART) or XMEAN] or X1. Calculation of the return value is performed by integration of the pdf function defining the triangle's sides and obtaining the cdf function equation (keeping in mind that the area under the entire triangle is always 1.0). The algebraic steps required for the integration and subsequent inversion (solving for X as a function of the uniformly distributed random number Q and the three triangle-defining parameters below) are described in detail in the source code as COMMENT statements and in Appendix C. Since the base lengths of the two right triangles (formed by the altitude at the mode value X1) comprising the overall triangle are an important part of this calculation, transformation of coordinates is utilized to move from

an origin-based frame of reference to a mode-based frame-of-reference [base length A = (mode-min value); base length B = (max-mode value)].

Calculation of a return value for the log-triangle distribution (JTYPE=-4) is very similar to that for the triangular distribution except for the fact that the logarithms of the right triangle base lengths are used and are defined as follows:

A = -ln(min/mode)
B = ln(max/mode).

Calculation of a return value from a uniform distribution (JTYPE=-9) defined by maximum and minimum endpoints is straightforward and involves use of the uniformly distributed random number Q and a coordinate transformation. If XMEAN is the leftmost endpoint and XDATA(ISTART+1) is the rightmost endpoint, the following expression can be used to calculate X:

X = XMEAN + Q*(XDATA(ISTART+1) – XMEAN) or X = X1 + Q*(X2 –X1) .

For the trapezoidal distribution (JTYPE=-6), calculation of the return value is accomplished by breaking the trapezoid into two triangles and a uniform distribution. The geometric and algebraic details of the inversion are included in the COMMENT statements within the source code listing.

Calculation of a return value from a user-defined histogram [JTYPE=+ (number of boxes)] is also very straightforward. Calculation of the cdf was accomplished in SUBROUTINE PRINTX by cumulative addition of the read-in pdf fractiles. The value of the uniformly distributed random number, Q, is compared to the cdf values for each box of the histogram. Once the box whose cdf range envelops Q is identified, a correspondence is made between Q and the incremental X-value range associated with the particular box. Interpolation is used to locate a distinct X within that range. The associated algebra is described in the source code COMMENT statements in XVALUE.

If the user wishes to have discrete values rather than boxes on the horizontal axis of his histogram, he needs to insert a discretization subroutine in his model such that rounded or integer return values are available to SUBROUTINE XVALUE.

Storage of return values

Now that a return value X is calculated for variable Y(I), it is stored by the statement Y(INDEX)=X and calculation of a return value for Y(INDEX+1) begins, again using X as the generic name for the return value. After all input Y values have return values, the input part of the Y-vector and the input part of /MASTER/ COMMON block is full and control is returned to MVSSOP, which then uses the return values for one iteration of SUBROUTINE MODEL.

5.7.5 Use of MVSSOP for Statistical Analysis and Display of MVSS Results

After each call to MODEL by MVSSOP, the output variable positions of the /MASTER/ Y-vector are filled. These values constitute the figures of merit from the user's model. The input and output values now stored in the Y-vector constitute one complete case or iteration of the MVSS. Output unit IBUG can be used to store and write the Y-vectors resulting from every iteration if desired. Normally the temporary storage unit, LTEMP, (preferably on a hard disk) is used to store this data for statistical analysis and graphical presentation. MVSSOP calculates a cumulative sum and cumulative sum-of-the-squares for each Y(I) such that the mean and variance can be calculated for all variables after the last iteration. The computation of the variance will yield meaningless results if the variance is very small compared with the mean. This is due to taking the difference of two nearly equal numbers whose real difference is smaller than the accumulated truncation error. Tests are also performed to make sure no variable has any value below 10^{-30} or above 10^{30}.

After all iterations are complete, NITER or M values for each variable Y(I) are stored in the Y-array. The analyst is now interested in the statistics for the Y(I) values resulting from all NITER submitted cases (NITER = 5000 iterations for the sample problem). For input Y-variables the analyst is interested in verifying that the resulting sampling distribution approximates the input distribution specified. For output Y-variables the analyst is interested in the uncertainty range and calculated statistical parameters for the important figures of merit. Such data is much easier to visualize if it is presented in histogram form in addition to presentation of the usual calculated statistical parameters (mean, variance, percentiles, max, min, etc.). The balance of subroutine MVSSOP is concerned with the "binning" of the data for each variable and the preparation of histograms capable of display from a line printer.

In order to construct a histogram, every value of Y(I) must be located within a bin on the range from the lowest occurring value of Y(I) to the highest. Selection of the proper bin size depends on factors such as the size of the page available for printout, the ratio of the maximum Y(I) to the minimum Y(I), the desirability of having a rational or round number for the bin width, and the character width and line width for the printer or screen. The algorithms and tests used to determine the histogram bin width are documented in detail by COMMENT statements in the source code listing.

Once the histogram boundaries and bin widths have been determined for all input and output variables, the code must put the input sample and output results into the bins. This process is accomplished one iteration at a time. The bin number, counting from the left, is calculated as follows for a given Y(I) or Y(K) value for a given iteration:

$$J = (Y(K)-XLOW(K))/XDIFF(K) + 1 \ ,$$

where

Y(K) is the variables value

K (or I) is the variable identification index for the Y-array

XLOW(K) is the lowest value encountered for Y(K) over all iterations

XDIFF is the bin width for variable Y(K).

Of the NITER values of a particular Y(I) which are partitioned, count must be kept of the number of times a Y-value falls within a given bin. The array NHIST is used to keep the count every time bin J is encountered:

NHIST(K,J)=NHIST(K,J) + 1 ,

where K is the variable index and J is the bin number.

Once all NITER values for a particular Y(I) have been "binned," the code counts from the left and calculates both the relative and cumulative percentages of occurrences from the leftmost bin to the bin of interest. These percentiles are then printed next to each bin. A separate table is also printed which displays the minimum, 5%, 50%, 95%, and maximum values for each variable. In order to print histograms, a character must be associated with each occurrence or nonoccurrence in a particular bin. A "*" is used to indicate one or more occurrences and a blank (" ") a nonoccurrence. If there are many iterations a * may indicate more than one occurrence and the number of occurrences per * (NSCALE) is noted on the output. The variable NSCALE is calculated by noting the number of occurrences in the tallest bin and making sure that the associated "*******" bar will not run off the page. For a 132-character line length printer, 75 "*" characters are allowed. Figure 5.8 shows the variable nomenclature on an example histogram.

Normally MVSSOP will print histograms for both the input and output Y(I) variables in the /MASTER/COMMON block. If a negative sign is placed in front of the number of iterations in the RDMVSS input table, only the designated output variables (with JTYPE = -99) will have histograms printed. If a constant or delta-function distribution (JTYPE=0) is utilized for input or JTYPE = -88 for output, no histogram is printed.

Once all histograms are printed, the large amount of data (the Y-array) stored in output unit LTEMP is scratched. Unit IPLOT can be used to store the histogram building information in the event that the user wishes to use DISSPLA or other plotting software to plot the data. The PDA option described in Sect. 7.2 can be used to convert pdf histograms into cdf data amenable to plotting and data characterization. The IPLOT format can be directly read by INTOHIST.EXE within the directory of the PDA option.

Control of the program is now returned to the MAIN and an "end of run" message is printed.

ORNL-DWG 89-4905 ETD

Y(1) K=1

STATISTICS FOR INPUT VARIABLE ALPH1 (1) JTYPE= -1 SAMPL.DIST TYPE: TRIANGLE

MINIMUM 104.09 MAXIMUM 493.95

MEAN 266.98 VARIANCE 7256.7 STD.DEVIATION 85.186

PERCENTILES: 2.5% 132. 5.0% 145. 50.0% 255. 95.0% 423. 97.5% 446.

NITER → 5000 POINT HISTOGRAM * = 4POINTS

XDIFF=10

NHIST (K,J)=# of occurrences or selections per bin

PROB	CUME	Y VALUE		
0.0016	0.0000	100.0	8	**
0.0068	0.0016	110.0	34	********* J=1
0.0124	0.0084	120.0	62	********** J=2
0.0188	0.0208	130.0	94	********** J=3, etc.
0.0210	0.0396	140.0	105	**********
0.0274	0.0606	150.0	137	**********
0.0314	0.0880	160.0	157	**********
0.0388	0.1194	170.0	194	**********
0.0426	0.1582	180.0	213	**********
0.0460	0.2008	190.0	230	**********
0.0496	0.2468	200.0	248	**********
0.0562	0.2964	210.0	281	**********
0.0436	0.3526	220.0	218	**********
0.0430	0.3962	230.0	215	**********
0.0416	0.4392	240.0	208	**********
0.0422	0.4808	250.0	211	**********
0.0386	0.5230	260.0	193	**********
0.0398	0.5616	270.0	199	**********
0.0352	0.6014	280.0	176	**********
0.0290	0.6366	290.0	145	**********
0.0322	0.6656	300.0	161	**********
0.0334	0.6978	310.0	167	**********
0.0294	0.7312	320.0	147	**********
0.0216	0.7606	330.0	108	**********
0.0220	0.7822	340.0	110	**********
0.0242	0.8042	350.0	121	**********
0.0256	0.8284	360.0	128	**********
0.0260	0.8540	370.0	130	**********
0.0184	0.8800	380.0	92	**********
0.0168	0.8984	390.0	84	**********
0.0136	0.9152	400.0	68	**********
0.0170	0.9288	410.0	85	**********
0.0116	0.9458	420.0	58	**********
0.0112	0.9574	430.0	56	**********
0.0108	0.9686	440.0	54	**********
0.0066	0.9794	450.0	33	********
0.0060	0.9860	460.0	30	********
0.0048	0.9920	470.0	24	******
0.0028	0.9968	480.0	14	****
0.0004	0.9996	490.0	2	*
	1.0000	500.0		

interpretation: 84 values of Y(1) or ALPH1 were selected (input dist) or occurred (output dist) between 390 and 400

Fig. 5.8. Nomenclature and printed example for histogram construction portion of ORMONTE MVSS option.

6. SPECIAL CONSIDERATIONS WHEN CONSTRUCTING A MODEL TO BE DRIVEN BY ORMONTE

The following guidelines, if observed when constructing a model for your application, will greatly enhance its compatibility with ORMONTE and minimize the chance for execution errors.

6.1 Variable Naming

Variable names in the MASTER common block may be mnemonic and up to eight characters long. They should be of type REAL*4. If ORMONTE is to be connected with an existing model where non-REAL*4 variables are important, the user should write insert the necessary preprocessing and conversion routines for renaming variables prior to and after accessing by ORMONTE. These routines should be part of the user's model. The labeling names (alphameric variables XNAME and NAMEN(I,J) can be up to six characters long. (See comments in last paragraph of Section 5.4.)

6.2 Internal Loops

If the modeling application requires many loops or iterations within each iteration of ORMONTE, the MVSS run may take a very long time to execute. If the time to run a single deterministic case is long, the time to run an MVSS will be approximately NITER times as long.

6.3 Independence of Input Variables

All of the uncertain input variables should be statistically independent, i.e., if the value of one input changes, it should not affect the value of another. Consider a process model in which some of the operating conditions are uncertain, such as the temperature within a vessel. Another uncertain variable might be a rate coefficient for a chemical reaction taking place within the vessel. In reality these two possible input variables are not independent, since rate coefficients usually vary with temperature. The modeler would be prudent to make the temperature an uncertain independent sensitivity variable and create an algorithm or look-up table within his model which keys the most likely value of the rate coefficient to the temperature. The uncertainty in the rate coefficient could then be handled by use of an uncertain multiplier which affects this temperature-keyed most likely rate coefficient value. This procedure is often called "keying" variables and is a very useful modeling technique.

6.4 Reoptimization Within a Model

Many models are designed such that the program calculates a maximum or minimum figure of merit for a given set of input variables. This procedure usually involves nested loops or the repeated use of numerical algorithms to search for the

combination of input values that produce the optimum case. The user of ORMONTE should take care not to mix uncertain ORMONTE input variables with optimization variables. For each MVSS iteration in ORMONTE, the input uncertainty values should remain unchanged within that iteration. If an optimization variable were used for an ORMONTE input, the final value optimized value of that input variable is very unlikely to be the same value as the initial sample input value. A model can be designed which will optimize every Monte Carlo iteration with respect to the optimization variables for a given set of uncertain ORMONTE input variables that remain unchanged during that iteration. Reference 1.2 discusses this model design problem in greater detail.

6.5 Reduction of Output

Models that print extensive output tables for deterministic cases usually require some modification before use in conjunction with ORMONTE. Print suppression logic must be added to the WRITE statements such that every iteration does not produce a complete set of output tables. For some mainframe computers such as the CRAY, it may be necessary to COMMENT out many of the WRITE statements and their FORMATs in order to realize the full computing efficiency and speed of these machines.

7. ACQUISITION AND CHARACTERIZATION OF INPUT DATA

7.1 Some DOE Experience-Based Guidelines to Assist in the Selection of Numerical Input MVSS Values

Since the results obtained from any Monte Carlo simulation are only as good as the input data, the quantitative determination of MVSS input variable ranges and distributions is of paramount importance to a probabilistic analysis. Unfortunately, there is no clear-cut or robust procedure which guarantees consistency and accuracy in the selection of input parameter values. This is an especially serious problem where comparison of competing systems or technologies is in progress and where consistency of evaluation must be maintained in order to protect the evaluator's objectivity and fairness.

Since many inputs require projection now of what their values will be at the time of future large-scale system implementation, the evaluator finds himself in the realm of technological forecasting; and he can benefit from exploring what this heuristic science can offer as an aid to input selection. One useful forecasting principle gleaned from the literature is that of collective judgment utilizing expert opinion. Policy makers often use formalized procedures, such as the Delphi method, to obtain a qualitative view of future development trends. Anonymity is maintained among participants, and written viewpoints are exchanged until consensus evolves. In this manner no well-known or dominant individual can persuade others on the basis of his standing in his field alone.

For the quantitative task of determining MVSS input variable ranges and distributions, consistency of evaluation and assignation is more important than consensus and the associated participant anonymity. Since ranges rather than single values are to be determined, consensus need not focus on one deterministic value, but rather on whether the range of input values is indeed possible. Less formal face-to-face discussion and debate among peers, i.e., peer review, with a consistent set of ground rules has been found to be a useful procedure and was successfully used as part of the selection procedure for the Department of Energy's choices of advanced isotope separation processes for uranium enrichment (Ref. 3.9) in both 1982 and 1985. The "peers" in this case consisted of technical personnel from DOE and the participating development and process evaluation contractor organizations.

The primary task of a peer review is to sort out the important technology, modeling, and cost issues from the insignificant issues that might be erected by competitors as diversions to delay considerations of technical areas where real problems exist in their own applications. Participants in such a venture must have access to technical documentation and relevant experimental data for the alternative technologies. If alternative technologies are not being considered, peer

review among the various project teams working on a single project can still be useful as a preliminary to uncertainty analysis. Such a procedure can be instituted among R&D, engineering design, cost estimating, and operations analysis teams. Each group represents expertise in different disciplines; hence opinion may vary widely.

Once the peer review has identified the important issues, the analyst must submit the values going into the system model and relationships within the model itself to peer review. Although the latter task may be time-consuming, it is important that all participants have confidence that the models driven by ORMONTE adequately represent the systems and cause/effect relationships considered. Since first-of-a-kind technologies vary so widely in their descriptions and relevant phenomena, no generalized quantitative rules can be set forth for assigning input ranges. Along with the qualitative properties of the various previously discussed probability distributions, the following guidelines are given for assignment of numerical values.

1. The mode input values do not necessarily have to correspond to the base or reference case. The analyst must determine the reasonableness of the baseline inputs and keep in mind that a baseline supplied by a development organization is likely to include optimistic values; however, there are cases in which a confident developer might submit an overly conservative baseline. In reports by Tversky and Kakneman (Ref. 7.1) and Capen (Ref. 7.2), tendency of estimators, regardless of their sophistication, to state overly narrow confidence intervals is discussed. Tversky and Kakneman believe this tendency is due to a heuristic of thinking common to all people that an estimator anchors the expected range to the original best estimate chosen by the estimator. A report by Tonn and Goeltz (Ref. 7.3) explores how subjects mentally combine estimates of uncertainty in probabilistic and evidential contexts.

2. Even if all the input distributions have as their mean the baseline value for each input, the Monte Carlo driver or sampling program will never reproduce the base case exactly, i.e., no single design is produced in which all the input distributions have their baseline values chosen by the random number generator.

3. It should be qualitatively ascertained whether any ensemble of random input variables selected by the ORMONTE Monte Carlo driver might represent an inconsistent or physically nonrealizable set of input data. If so, there are dependencies between variables or logic problems within the user's model itself which need correction. By using WRMAST to print out the input distribution return values and output figures of merit every couple of iterations, one can get ensembles of random data to inspect for possible problems of this nature. L. Kryzanowski, et al., (Ref. 7.4) suggest a means of handling dependencies between variables which impose conditional probabilities via "base" and "keyed" variables.

7.2 PDA (Probability Data Analaysis) Software for Analysis and Characterization of Raw Data

The effectiveness of using a computerized tool has long been known to be dependent upon the quality of the input data (i.e., garbage in-garbage out). The ORMONTE code and the model(s) it drives are no different in this respect. This section will address the sources and methods of representing data for ORMONTE variables whose values are uncertain. While not exhaustive, techniques which have been successfully used in conjunction with the ORMONTE code will be described. PC software which has been developed at ORNL for analyzing and characterizing raw data will also be discussed.

7.2.1 Sources of Data

Data related to an uncertain variable can come from several different sources. If the variable of interest is a process parameter capable of quantitative measurement, then data on the variability of the parameter can be collected as measurement observations and stored in a computer file. Many times, automated data logging equipment can be used to measure and record literally hundreds or thousands of individual observations (data points) for a given parameter. In the next section, a software tool which can analyze this raw data will be described.

The preceding discussion is, in a general sense, related to the measurement of the variability of a physical quantity. That is, the parameter is associated with something that can be physically measured. There are, however, many times when the variable of interest cannot be physically measured because the variable describes something that may exist only in the future. For example, output parameters for a new or scaled-up process, costs/revenues for a future equipment item, future escalation/inflation rates name only a few of the possible variables that cannot be immediately measured. The uncertainty in variables of this type has been characterized through a number of techniques which develop a collective judgment based on expert opinion. The earlier-mentioned Delphi method is one of the more well known in this class.

Another technique that has been used at ORNL (Ref. 3.13) to quantify uncertain parameters based on expert opinion is called the SRI Probability Encoding Method. Developed in the 1970s at SRI International, this method seeks to minimize biases in the expert's response while extracting the expert's knowledge on the topic. It is a structured interview technique which is conducted by a trained interviewer. It is beyond the scope of this report to fully describe the SRI Encoding Method. A summary of the method has been written in the technical literature (Ref. 7.5). Certain aspects of the method will be discussed here, however, to relate them to a supporting software tool that has been developed at ORNL.

As mentioned above, the method involves a structured interview process between a trained interviewer and a subject who is an expert on the topic at hand.

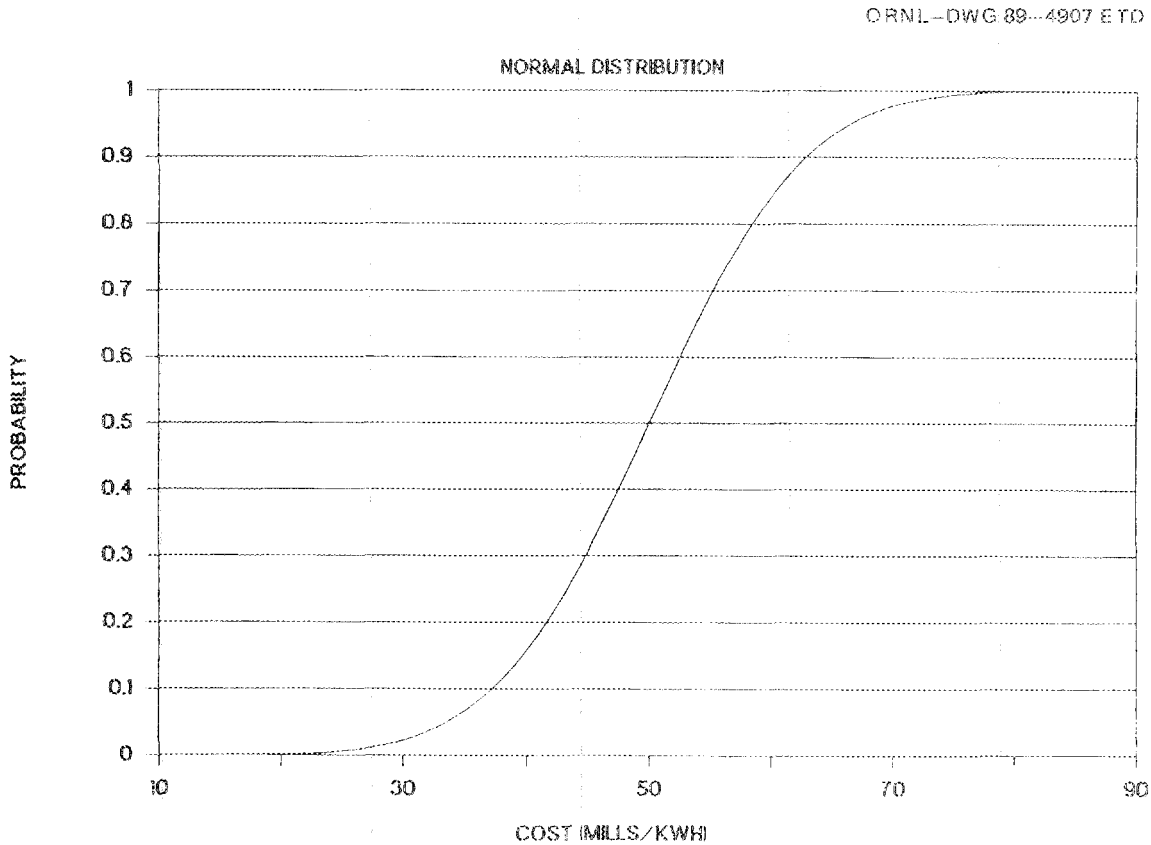


Fig. 7.1. Typical cumulative probability S-curve.

It is important to stress that the interviewer needs to be knowledgeable of the SRI Encoding Method in order to minimize biases that can erroneously influence the expert's responses. In general, the outcome of an encoding session is a cumulative probability function curve describing the expert's opinion of the particular parameter. An example of such a curve is shown in Fig. 7.1. The curve can be interpreted such that for a given probability on the Y-axis the associated quantity on the X-axis (a unit electric power cost in Fig. 7.1) will be equal to or less than the value indicated by the curve.

In developing the curve, probabilities will be associated with specific quantitative values. In arriving at an appropriate probability/quantity combination, several types of questions (as discussed in Ref. 7.5) may be posed to the subject, some of which use a visual aid known as a probability wheel to assist in the assessment. In such an assessment at ORNL, an electronic version of the probability wheel was developed and is contained within a PC-based software tool known as the Probability Data Analysis or PDA software package. The following paragraphs will discuss this portion of the PDA package. The data analysis features of the package will be discussed in Sect. 7.2.2.

The heart of the PDA package is a Lotus 1-2-3 spreadsheet file, PDA.WK1. When retrieving the file in Lotus 1-2-3, the file appears as a typical spreadsheet. There is, however, a set of Macro instructions which allow the user to automate the functioning of the spreadsheet and to collect and record cumulative probability and quantity data for uncertain parameters. This Macro is invoked by an Alt-W keystroke. The operation of the Alt-W Macro supports an indirect response method used in the SRI encoding process which uses a probability wheel (also called in less scientific circles a wheel of fortune) as a visual aid in eliciting responses from the interviewee. [Batteries and Vanna White are not included.] In an indirect response mode, the subject is asked to choose between two alternatives; one relating to the parameter of interest and the other based on a random chance device such as a wheel of fortune or probability wheel. Figure 7.2 shows the face of the wheel with a shaded area being the "winning" region. In general, the shaded area is increased or decreased until the subject is indifferent about choosing between an outcome related to the wheel and one related to the uncertain parameter. For example, a question could be asked of the subject, "Would you rather wager on a spin of the wheel with this much "winning" area or that the operating availability fraction of System X will be at least 0.78?" When the expert is indifferent, the relative area on the wheel corresponds to the expert's opinion of the probability of the parameter conditions posed in the question. This type of questioning can be repeated to develop a set of probability/quantity outcomes. It is very important that the reader note that this line of questioning is only one of several methods used in the encoding process, and that reliable results are only obtained when the interviewer is knowledgeable of the complete encoding process as well as the underlying statistical rules.

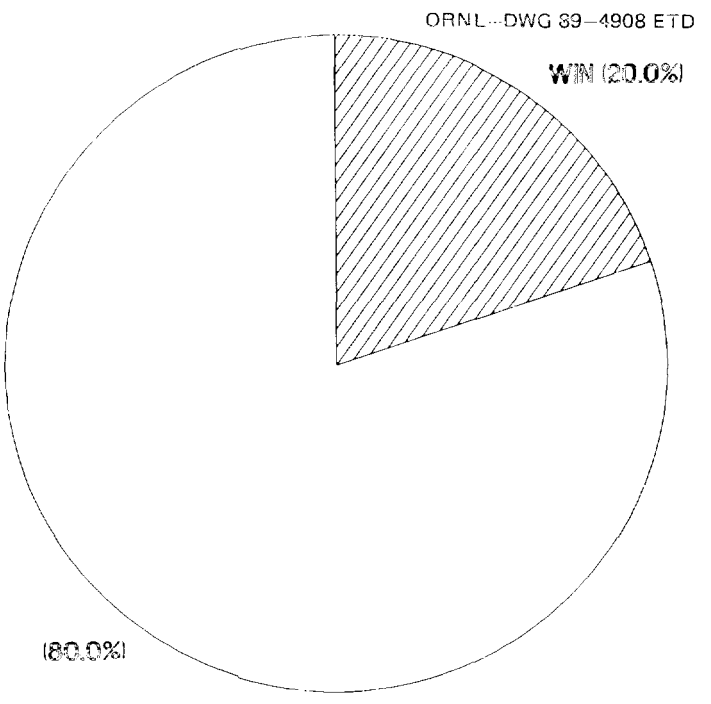


Fig. 7.2. Example probability wheel display.

When the Alt-W Macro is invoked in PDA.WK1, the user will see a menu at the top of their screen similar to that shown in Fig. 7.3. Selecting BEGIN and placing the cursor in the appropriate row of the Topic column will allow the user to select a new or existing parameter topic for study. As shown in Fig. 7.4, a topic or variable name is provided, in this case TEST DISTRIBUTION, and the first quantitative value desired to be analyzed is entered. Next, the program will present an example probability wheel image on the screen. Any keystroke will clear the image, and the user will be expected to provide his own probability value corresponding to the quantitative value under consideration. As mentioned, this Macro is designed to support the wheel type of questioning, but a final probability value may be entered at this time if another interview technique has been used. As this point, using the menu, the user may CONTINUE manipulating the probability wheel or KEEP the current value. The user may continue to invoke the Alt-W Macro to enter additional data points for the variable or topic of interest. When the Alt-W menu is present, the user may QUIT the automated macro and return to a manual spreadsheet mode. In the manual mode, data may be entered in the probability data storage area directly.

Two other features available in the Alt-W mode are the PLOT and SPIN choices. In conducting the probability wheel questioning, it may be useful to demonstrate the effects of "spinning" the wheel for a given "winning" fraction on the wheel. The SPIN option will use a random number generator to electronically "spin the wheel" and determine a win or loss on a given spin. This option may be repeated as many times as desired to demonstrate the randomness of outcomes. When several quantity/probability data points have been obtained, such as in Fig. 7.5, it is useful to look at the general shape of the cumulative distribution the expert is providing. The PLOT option will plot the cumulative probability curve for the current topic as shown in Fig. 7.6. This feature is helpful in resolving certain discrepancies that may appear in the responses such as the negative slope in part of the cumulative curve shown in Fig. 7.6. When PLOT is selected, the raw data in the probability data storage area of the spreadsheet are automatically sorted in ascending quantity sequence as shown in Fig. 7.7.

It should be noted that in using PDA.WK1 when the cursor is on a spreadsheet cell that already has a value that is satisfactory to the user, a simple return keystroke will continue operation of the Macro.

7.2.2 Characterization of the Input Data

As discussed in the previous section, uncertainty data that are based upon variability of a physical quantity can be developed by repeated measurements. For use in a code such as ORMONTE, such raw data, which may number into the thousands of points, must be reduced to a form which can serve as input to the program. A histogram is a useful mechanism to describe this type of data, and a FORTRAN utility program called INTOHIST.EXE is provided with the PDA software package to characterize such data into a histogram form. The program is self-prompting and expects the input raw data file to consist of single records for

```
E1: [W11]                                                          MENU
BEGIN  CONTINUE  KEEP  PLOT  SPIN  QUIT
Select/enter topic
        E              F                G              H
1                PROBABILITY DATA STORAGE AREA
2
3              Topic          Data values  Probability
4          ==========================================
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
                                  CMD
```

Fig. 7.3. LOTUS 1-2-3 macro designed for probability questioning (screen display menu).

ORNL-DWG 89-4910 ETD

```
G7: [W14] 200                                                    VALUE
200
        E          F          G          H
1                PROBABILITY DATA STORAGE AREA
2
3          Topic              Data values   Probability
4          ==========================================
5
6
7          TEST DISTRIBUTION                 200
8
9
10
11
12
13
14
15
16
17
18
19
20
```

Fig. 7.4. Screen display showing topic name and variable value for which probability is to be evaluated.

E1: [W11]

READY

PROBABILITY DATA STORAGE AREA

Topic	Data values	Probability
TEST DISTRIBUTION	200	0.00
	3000	1.00
	1600	0.70
	1000	0.45
	500	0.07
	2000	0.88
	2500	0.94
	700	0.15
	400	0.05
	800	0.20
	1200	0.50
	1000	0.30
	1500	0.75

Fig. 7.5. Screen display showing complete set of value/probability data obtained by questioning.

ORNL-DWG 89-4912 ETD

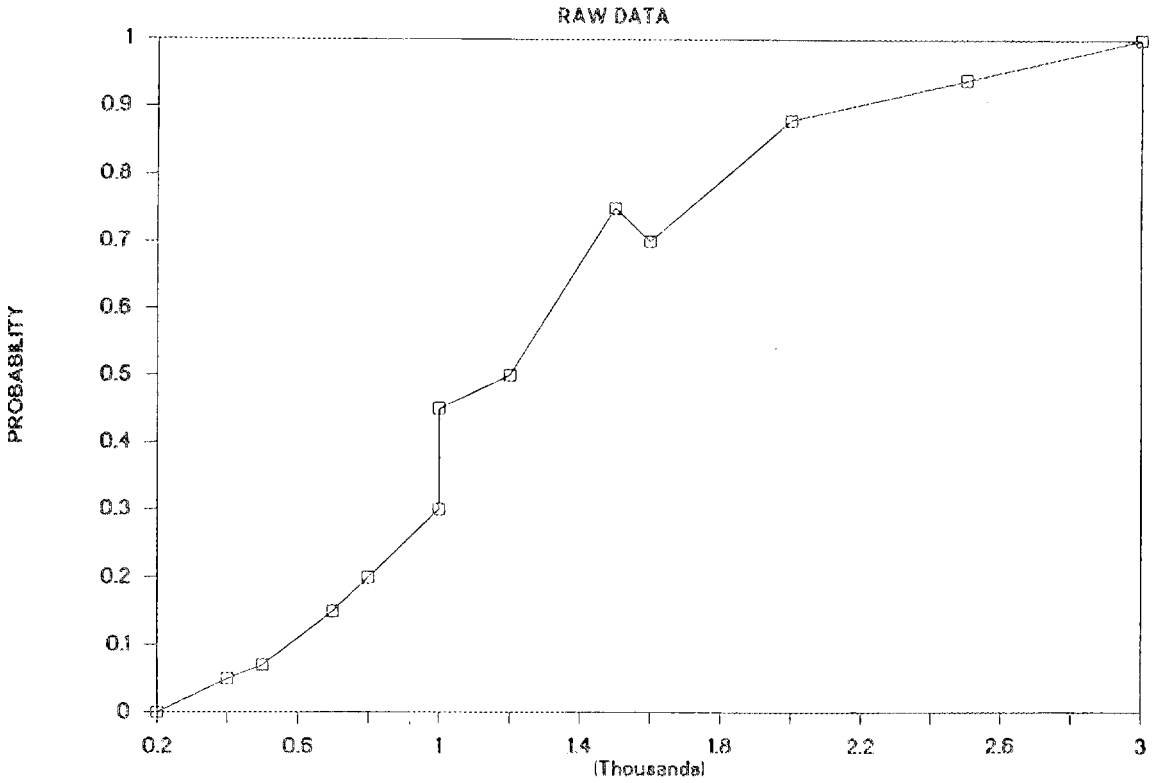


Fig. 7.6. Data from Fig. 7.5 as drawn by PLOT option.

each data point (i.e., a line feed/carriage return after each point). The output of the program is cumulative probability/quantity data suitable for use in PDA.WK1 (see Fig. 7.7). INTOHIST.EXE also optionally provides the relative probability histogram parameters (number of bins, bin width, etc.) in the same format as produced for output unit IPLOT by ORMONTE. A second option of INTOHIST picks up relative probability histogram parameter data (in IPLOT format) and converts it to cumulative probability/quantity data suitable for use in PDA.WK1.

The remainder of this section will discuss the analysis and data characterization capabilities of the PDA.WK1 spreadsheet program. As discussed in Chap. 5, several types of statistical distributions are available to describe the parameter uncertainties. The PDA software package allows the user to determine if the cumulative probability data for a given variable can be effectively described by either a normal, log-normal, triangular, or log-triangular distribution.

Similar to the data entry portion of the PDA.WK1 spreadsheet program described in Sect. 7.1, the data analysis routines are Macro controlled. The analysis is started by entering an Alt-R keystroke. Initially, a menu such as shown in Fig. 7.8 will be presented at the top of the screen. As a first step, the user should select

G20: [W14]

READY

PROBABILITY DATA STORAGE AREA

Topic	Data values	Probability
TEST DISTRIBUTION	200	0.00
	400	0.05
	500	0.07
	700	0.15
	800	0.20
	1000	0.45
	1000	0.30
	1200	0.50
	1500	0.75
	1600	0.70
	2000	0.88
	2500	0.94
	3000	1.00

CMD

Fig. 7.7. Screen display showing Fig. 7.5 data after automatic sorting.

```
E1: [W11]                                                                  MENU
NORMAL  LOG_NORMAL  PLOT  SUMMARIZE  TRIANGLE  CLEAR  QUIT
Normal dist.
        E                F                G                H
1                PROBABILITY DATA STORAGE AREA
2
3                Topic            Data values  Probability
4             ==========================================
5
6
7             NORMAL DISTRIBUTION              10        0.00
8             mean=50 sigma=10                 20        0.00
9                                              30        0.02
10                                             40        0.16
11                                             45        0.31
12                                             50        0.50
13                                             55        0.69
14                                             60        0.84
15                                             70        0.98
16                                             80        1.00
17                                             90        1.00
18
19
20
                                   CMD
```

Fig. 7.8. LOTUS 1-2-3 macro designed for data analysis (screen display, menu and example data to be fit to normal distribution.

CLEAR to reset the temporary working storage areas and prepare the spreadsheet for a new analysis. Failure to CLEAR the program before starting a new analysis can cause old data values in certain working areas to be included in the current analysis, thus corrupting the evaluation. CLEAR the program before you begin a new analysis! Should the work areas be empty already when CLEAR is selected, an error message will be issued, and the user should press ESC followed by another Alt-R and proceed as though the CLEAR had been successful.

After CLEARing the work area, the user must select a base characterization of the cumulative probability data as being either normally or lognormally distributed. Actually both distribution types can be tried by selecting NORMAL or LOG_NORMAL from the menu and moving the cursor to the desired row of the Topic column. The important point is that one or both Gaussian distribution types must be checked first before the triangular distributions can be evaluated. This procedure is necessary because the program presents the triangular distributions in comparison to the Gaussian distributions. As an example, the data shown in Fig. 7.8 will be evaluated. These cumulative probability data were developed from normal probability tables for a mean of 50 and a standard deviation of 10, so in this example we expect to see an ideal fit in the NORMAL selection.

When either the NORMAL or LOG_NORMAL option is selected, a linear regression is performed on the data to explicitly solve for the mean and deviation. This is possible due to the property of the standard normal distribution that the standardized random variable z, defined as,

$$z = \frac{(x - \mu)}{\sigma} \quad ,$$

can be matched to a given value of x by the associated probability provided by the user for that value of x. Thus, regression pairs of z and x are formed such that the above equation, when rearranged as

$$x = \sigma \cdot z + \mu \quad ,$$

is of the linear form $y = mx + b$. The mean and standard deviation are then directly determined by regression with the resulting output shown in Fig. 7.9. A printer summary of the regression analyses can be obtained by selecting the SUMMARIZE option on the menu.

A plot of the resulting parameter-defined (μ, σ) distribution and the raw input data is very useful in judging the appropriateness of the distribution as a descriptor of the raw data. Once the normal and/or log-normal regression(s) are performed, the PLOT option may be selected to view the distributions. An added feature of the PDA spreadsheet program is that the user may alter or "fine-tune" the distribution parameters (μ, σ for the Gaussian options), if desired, during the PLOT option to adjust the distribution shape (via the standard deviation parameter, σ) and

```
J58: [W9] 'NORMAL DISTRIBUTION                                          MENU
NORMAL  LOG_NORMAL  PLOT  SUMMARIZE  TRIANGLE  CLEAR  QUIT
Normal dist.
        J            K            L            M
58  NORMAL DISTRIBUTION
59
60    Normal Regression Output:
61  Constant                                  -5.0000
62  Std Err of Y Est                           0.0000
63  R Squared                                  1.0000
64  No. of Observations                            11
65  Degrees of Freedom                              9
66
67  X Coefficient(s)            0.1000
68  Std Err of Coef.            0.0000
69
70                           Current     Calculated
71                          ==========================
72  Mean                      50.0000       50.0000
73  Sigma                     10.0000       10.0000
74
75  Avg. Error on Probability                  0.0000
76  (Valid ONLY after running PLOT)
77
                                            CMD
```

Fig. 7.9. Screen display showing normal fit regression analysis results for data on Fig. 7.8.

location (via the mean value, μ). A plot of the resulting normal distribution in the above example is shown in Fig. 7.10.

As mentioned earlier, the normal or log-normal analysis (and associated PLOT) must be performed before the triangular distributions can be considered. When the normal and/or log normal distributions have been checked, the TRIANGLE option may be selected to evaluate either a triangular or log-triangular distribution against the input data. The TRIANGLE option is different from the NORMAL or LOG_NORMAL choices in that no explicit calculation of the distribution parameters (i.e., minimum, mode, and maximum) can be made. Rather, the user must select the parameters for best or nearest fit in an iterative process from the graphics image developed earlier by the program for the normal or log- normal evaluation. With an 80286-based processor (AT-type PC), the response in this iterative portion of the program is quite fast. However, this process may be rather time consuming for those operating with the earlier 8086-based PCs. Figure 7.11 shows a nearfit triangular distribution to the normal distribution shown in Fig. 7.10.

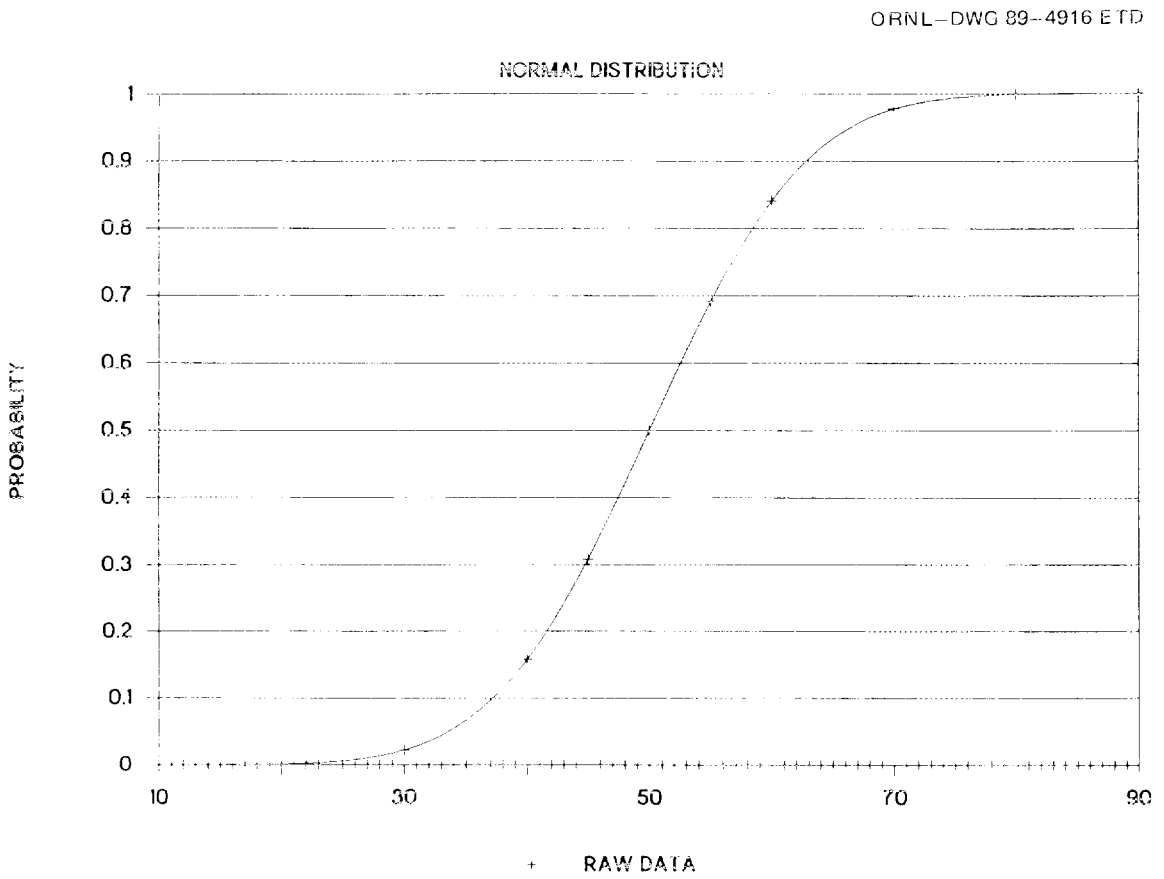


Fig. 7.10. Plot of Fig. 7.8 raw data points and normal distribution obtained by using parameters calculated from regression analysis in Fig. 7.9.

ORNL-DWG 89-4917 ETD

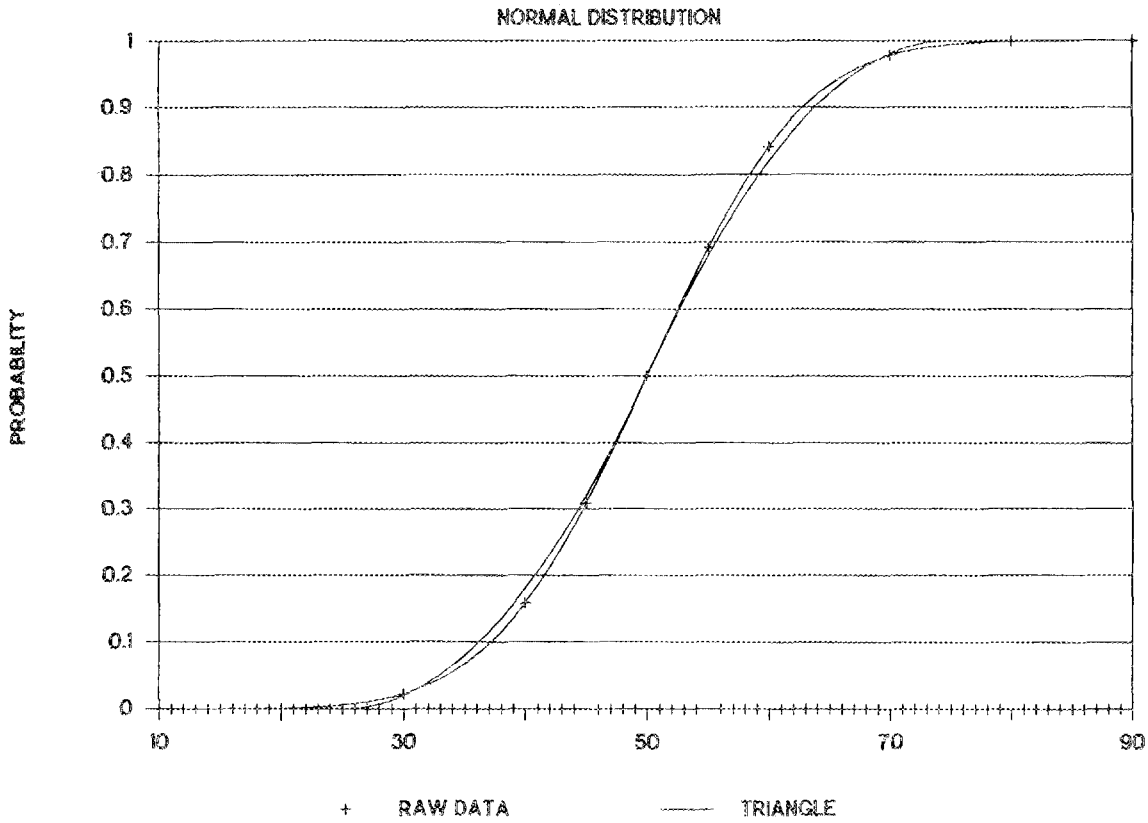


Fig. 7.11. Plot of attempt to fit triangular distribution S-curve to normally distributed raw data in Figs 7.8 and 7.10.

Once it is determined with the PDA software that the raw data for a given ORMONTE input variable can be accurately characterized with one of the statistical distributions discussed above, its parameters can be used as input into ORMONTE as described in Sect. 5.7. Of course, should the raw data not be amenable to a description in this manner, the raw data can always be represented as a histogram for input into ORMONTE. The necessary parameters for a histogram input are given in Section 5.7 as well.

7.3 REFERENCES

7.1 Tversky, A., et al., *Sci.*, 185, pg. 1124 (1974).

7.2 Capen, E. C., "The Difficulty of Assessing Uncertainty," *J. of Petrol. Technol.*, (Aug. 1976).

7.3 Tonn, B. and Goeltz, R.; *People Versus Probability*, ORNL/TM-10898, Martin Marietta Energy Systems, Oak Ridge National Laboratory, August 1988.

7.4 Kryzanowski, L., et al., "Monte Carlo Simulation and Capital Expenditure Decisions –A Case Study," *The Engineering Economist*, 18(1), (1972)

7.5 Miley W. Merkhofer, "Quantifying Judgmental Uncertainty: Methodology, Experiences, and Insights", *IEEE Transactions on Systems, Man, and Cybernetics*, Volume SMC-17, Number 5, September/October 1987, pp. 741–752.

8. INTERPRETATION OF THE OUTPUT STATISTICS AND HISTOGRAMS

8.1 Understanding the Output of MVSSOP

Figure 8.1 presents a typical histogram output for a figure of merit in an MVSS run from ORMONTE. Subroutine MODEL was accessed 5000 times in order to produce this data. The first item of interest is the table at the top of the figure which presents the summary statistics for all 5000 values of the figure of merit CUNIT calculated by MODEL. The minimum is the lowest value and the maximum the highest value encountered over all 5000 cases. The mean, standard deviation, and variance have the same meaning as for any large sample of 5000 items. The percentiles pass on very useful information to the analyst; for example, the data on Fig. 8.1 say that CUNIT has a 2.5% chance or less of being less than 0.913 and a 95% chance of being less than 4.73. The median or 50% value for CUNIT is 2.62.

The histograms themselves produce useful data. The highest bin (between CUNIT values of 2.2 and 2.4) or the one with the most occurrences (380 out of 5000) represents the narrow value range for which the occurrence of CUNIT is most likely. The PROB and CUME values to the left of the "****" stack in each bin give the relative and cumulative probabilities bounding the bin's left CUNIT boundary. The number of iterations or occurrences corresponding to each bin is also printed immediately to the left of the histogram "*"s.

It is of interest to consider the location of the deterministic or baseline case on the probability spectrum. In this case the baseline value of CUNIT (2.498) lies just to the right of the mode; hence, the analyst chose a deterministic baseline with a 45% probability of the cost goal actually occurring, i.e., there is a 45% chance of the unit cost being $2.50 or less. The next subsection (8.2) discusses how this comparison of deterministic and probabilistic results can be useful in economic comparisons of emerging technologies. ORMONTE has been frequently used by DOE and Martin Marietta Energy Systems to make this type of comparison.

8.2 Location of the Best-Guess Deterministic Figure of Merit Estimate Relative to the Probabilistic Figure of Merit Spectrum

For systems in the early R&D stages, performance or economic risk may be defined as the likelihood that the system will not work as well as predicted or not reach the economic goal established by its proponent. [Note that this definition of risk is not exactly the same as that for uncertainty, although the words are frequently synonomous in common usage.] The proponent's deterministic, model-generated figures of merit for the baseline design, along with associated best-guess single-point input values which support it, are sometimes referred to as the "base case," "baseline," or "point design."

ORNL–DWG 89–4906 ETD

Y-index

STATISTICS FOR OUTPUT VARIABLE CUNIT (14) JTYPE=-99 SAMPL.DIST TYPE: NOT-APPL

MINIMUM 0.49148 MAXIMUM 7.8217

MEAN 2.7276 VARIANCE 1.2089 STD.DEVIATION 1.0995

PERCENTILES: 2.5% 0.913 5.0% 1.10 50.0% 2.62 95.0% 4.73 97.5% 5.20

5000 POINT HISTOGRAM * = 8POINTS

Y(I)

PROB	CUME	VALUE		
0.0012	0.0000	0.4000	6	*
0.0102	0.0012	0.6000	51	*********
0.0240	0.0114	0.8000	120	**********
0.0292	0.0356	1.000	146	**********
0.0372	0.0646	1.200	186	**********
0.0462	0.1018	1.400	231	**********
0.0530	0.1480	1.600	265	**********
0.0682	0.2010	1.800	341	**********
0.0758	0.2692	2.000	379	**********
0.0760	0.3450	2.200	380	**********
0.0714	0.4210	2.400	357	**********
0.0734	0.4924	2.600	367	**********
0.0714	0.5658	2.800	357	**********
0.0654	0.6372	3.000	327	**********
0.0544	0.7026	3.200	272	**********
0.0482	0.7570	3.400	241	**********
0.0378	0.8052	3.600	189	**********
0.0350	0.8430	3.800	175	**********
0.0260	0.8780	4.000	130	**********
0.0206	0.9040	4.200	103	**********
0.0166	0.9246	4.400	83	**********
0.0132	0.9412	4.600	66	**********
0.0116	0.9544	4.800	58	**********
0.0090	0.9660	5.000	45	*********
0.0058	0.9750	5.200	29	*****
0.0042	0.9808	5.400	21	****
0.0026	0.9850	5.600	13	**
0.0034	0.9876	5.800	17	***
0.0032	0.9910	6.000	16	***
0.0018	0.9942	6.200	9	**
0.0018	0.9960	6.400	9	**
0.0002	0.9978	6.600	1	
0.0002	0.9980	6.800	1	
0.0004	0.9982	7.000	2	
0.0004	0.9986	7.200	2	
0.0004	0.9990	7.400	2	
0.0004	0.9994	7.600	2	
0.0002	0.9998	7.800	1	
	1.0000	8.000		

Mode Bin

Base Case (CUNIT=2.498) would fall within this bin

Relative Prob | Cum Prob | CUNIT Spectrum | No. of Occurrences | Histograms

Figure 8.1. Interpreting the output relative probability histograms from an ORMONTE MVSS run.

If ensembles of data (Y_i) from sampled input distributions (rather than one set of single-point values) are fed to the process model, the array representing the cost figures of merit calculated is displayed in the form of a probability histogram which approximates the shape of a normal distribution. The baseline figure of merit can be represented as a spike or single point value on the same abscissa as the histogram. Figure 8.2 illustrates such an example, in this case a unit cost. Since the area under a probability density function has a value of 1.0, the fraction of the area that lies above the base case value can be said to be a representation of the risk that a system will not meet its economic goals. Conversely, the probability of success may be defined as the area lying to the left (i.e., lower unit cost) of the base case. This left-lying area can be used if one defines future project success as the attainment of a given base line unit cost, $C_{U,base}$, or less (most often representing the result of baseline deterministic inputs submitted to the model by a proponent). One can integrate the curve pdf(C_U) from its lower limit to $C_{U,base}$ and assign a probability to attainment of that point. If $C_{U,base}$ lies above (i.e., to the right of) the mode of pdf(C_U), the probability of success is more likely than not. This result means that

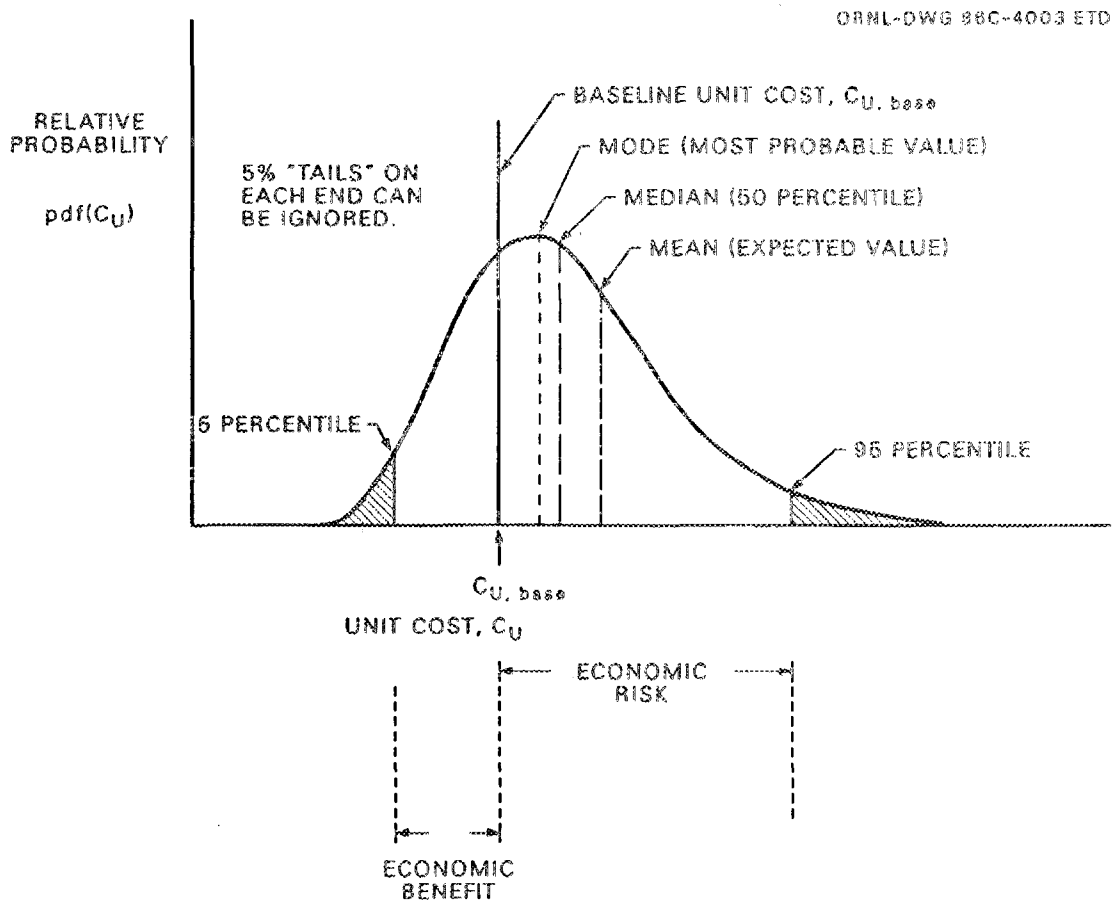


Figure 8.2. Locating a baseline unit cost on a hypothetical unit cost output histogram.

the developer has been very conservative in defining his baseline system. If $C_{U,base}$ lies to the left or below the mode, one states that the economic risk is considerable, i.e., the probability of success is not high given that "successful" is defined as meeting or bettering the base case. In this situation the developer has been overly optimistic in defining his baseline, and the future scaled-up technology runs a considerable risk of experiencing unit production costs well over the baseline prediction. When used in evaluating alternative or competing systems this type of information is especially meaningful, particularly if the pdf curves (histograms) and single-point estimates for each process are compared on the same axes.

8.3 A Caution Regarding Mode Values in Output Histograms

Although there is an implied mode value in the output distribution of a figure-of-merit, there is no associated "maximum likelihood" set of inputs. Thus, one cannot easily "dissect" the mode case; it may not even exist (or perhaps several may). Note also that if several figures-of-merit are tabulated, there is unlikely to be a single case (particular iteration) that maximizes all of them. That task, after all, must be handled by trade-offs within the users' model.

9. INSTALLATION OF ORMONTE

9.1 Installation of Sensitivity Analysis Program

In this section, the installation and execution of ORMONTE on a personal computer and a mainframe computer is discussed. The personal computer system assumed to be used is a IBM/PC-AT or AT-compatible with the following items.

One hard disk drive
One or more floppy drives
One laser-jet or dot matrix printer
One video display terminal
DOS 2.1 or higher
LOTUS 1-2-3 version 2.01 or higher (for PDA option only)
RM (Ryan McFarland) FORTRAN software (version 2 or higher)
Type 80287 memory chips in the AT
Software for editing and creating DOS (ASCII) files (Word Perfect is acceptable)

Installation on a mainframe, such as the STC/VAX 8600 discussed here, presumes that the user has a means of copying the necessary files from floppy to his mainframe user area and has a means of editing these files.

9.1.1 Installation and Execution of ORMONTE on a PC

1. Create a subdirectory entitled \ORMONTE\ on your hard disk.

2. Using DOS copy all files from the ORMONTE floppy to the ORMONTE subdirectory.

The following files are provided on the ORMONTE floppy:

Filename	Type	Resident FORTRAN subroutines
MAIN.FOR	Fortran 77 source code	MAIN
READOP.FOR	Fortran 77 source code	READOP
RDMAST.FOR	Fortran 77 source code	RDMAST
WRMAST.FOR	Fortran 77 source code	WRMAST
RDMODL.FOR	Fortran 77 source code	RDMODL(example problem)
MODEL.FOR	Fortran 77 source code	MODEL(example problem)
MASTER.FOR	Fortran 77 source code	INCLUDE statement for SUBROUTINE MODEL
SVSS.FOR	Fortran 77 source code	SVSSOP
ELAST.FOR	Fortran 77 source code	ELASTI
MVSS.FOR	Fortran 77 source code	MVSSOP,RDMVSS,PRINTX, XVALUE,FUNCTION INT1

RANSET.FOR	Fortran 77 source code	FUNCTIONS UNI, RNOR, URAND
TCTRAN.ASM	Machine language	RANINT
TCTRAN.OBJ	Compiled mach. lang.	RANINT
OPTIO.DAT	Data (example problem)	
MASINI.DAT	Data (example problem)	
MODL.DAT	Data (example problem)	
SVSS.DAT	Data (example problem)	
MVSS.DAT	Data (example problem)	
ELAST.DAT	Data (example problem)	

The PDA files are listed in Sect. 9.2.

3. Using the RMFortran software (Version 2), each XXX.FOR file should be compiled such that XXX.OBJ modules are created (the command is RMFORT XXX). When MODEL.FOR is compiled, the INCLUDE module MASTER.FOR will automatically be picked up and compiled within MODEL.FOR to form MODEL.OBJ. If changes are made to MODEL or the ORMONTE source code after a run, these source files should be recompiled.

4. The user should create the following file named ORMONTE.LNK using an editor:

```
FI MAIN,READOP,RDMODL,RDMAST,WRMAST,MODEL,SVSS,ELAST,MVSS,
   RANSET,TCTRAN OUTPUT ORMONTE
```

The RMFortran command PLINK86 @ORMONTE will link-edit the above string of XXX.OBJ modules and create an executable output load module entitled ORMONTE.EXE. Other output load module names may be used by changing the file name after the word OUTPUT in the LNK file above. If changes to source code are made after a run, the LNK file should be relinked after recompilation.

5. At the "C:\ORMONTE\>" prompt, typing ORMONTE will cause execution to commence using the XXX.DAT files resident in the \ORMONTE\ subdirectory. Output will be to the screen unless other DOS commands such as > PRT or > PRN are utilized.

6. Output on units other than 6 (which is normally unit IPRNT) will be written to output files FORTXX where XX is the output unit number (e.g., SVSS summary tables are written on file FORT15).

7. If during a run the program "hangs up" and produces no output, the user should check data set MVSS.DAT and be sure that the X-parameters are in the correct order.

9.1.2 Installation of ORMONTE on a mainframe computer (VAX 8600)

Installation on the VAX 8600 was very simple and involved only one simple FORTRAN change. The procedure is as follows:

1. Transport the XXX.FOR and the XXX.DAT files from the floppy to your user area on the VAX. KERMIT or other communication software can accomplish this task.

2. Using a VAX editor delete the last part of module RANSET.FOR, leaving only FUNCTION UNI(X) and FUNCTION RNOR(X). Add the statement INTEGER X to FUNCTION(X) and change URAND(X) to RAN(X). You are now invoking the pseudo-random number generator in the VAX system library instead of the one supplied as TCTRAN.OBJ for the PC application.

3. Compile all XXX.FOR files using the VAX command: FORTRAN XXX. When compiling MODEL.FOR, the VAX will pick up the INCLUDE module MASTER.FOR and include it in the module MODEL.FOR.

4. Link edit all of the XXX.OBJ files into a load module:
LINK MAIN,READOP,RDMAST,WRMAST,RDMODL,MODEL,SVSS,MVSS,
ELAST,RANSET
The load module will have the name MAIN.EXE

5. To run with the XXX.DAT files use the following command
RUN MAIN

Somewhat similar steps would be needed for the CRAY, PDP-10, and other mainframes. Use on an IBM mainframe requires a JCL (Job Control Language) program tailored for this application.

9.2 Installation of PDA (Probability Data Analysis) Options

Using DOS copy all files from the PDA diskette to your LOTUS-1-2-3 directory (usually \123\).

The following files are provided on the PDA diskette:

Filename	Type	Purpose
PDA.WK1	Lotus 1-2-3	Spreadsheet file for performing probability data analysis (PDA). Holds user-supplied input data and presents PDA results. Contains macro-instructions for automated operation.

REGIN.FOR	FORTRAN	Source code for REGIN.EXE
REGIN.EXE	Executable	Used in conjunction with PDA.WK1. Performs a look-up function in a standard normal data table to provide a z-value for a given probability.
REGIN.DAT	Data	Output from REGIN.EXE
REGOUT.FOR	FORTRAN	Source code for REGOUT.EXE
REGOUT.EXE	Executable	Used in conjunction with PDA.WK1. Performs a look-up function in a standard normal data table to provide the corresponding probability for a given z-value.
REGVAL.DAT	Data	Probability output from REGOUT.EXE
REGOUT.DAT	Data	Passes mean and standard deviation results from regression analysis in PDA.WK1 to REGOUT-.EXE.
NPROB.DAT	Data	Standard normal data file.
SWI.DAT	Data	Indicator used by REGIN.EXE and REGOUT.EXE to identify logarithmic analysis.
RAW.DAT	Data	Passes user-supplied raw input data from PDA.WK1 to REGIN.EXE and REGOUT.EXE.
XVAL.DAT	Data	Contains x-axis values for a given mean and sigma.
XMAP.DAT	Data	Contains y-axis values for a given mean and sigma.
AVGERR.DAT	Data	Passes sum-of-the-squares average error of probability from REGOUT.EXE to PDA.WK1.
INTOHIST.FOR	FORTRAN	Source code for INTOHIST.EXE
INTOHIST.EXE	Executable	Converts raw data or relative probability histogram parameters into cumulative distribution data.

TESTRAW.DAT	Data	Raw test data for input into INTOHIST.EXE. Contains 200 random selections along a normal curve with mean - 50 and sigma - 10.
RAWOUT-DAT	Data	Cumulative probability data output from INTOHIST.EXE using TESTRAW.DAT as input.
HISTPAR.DAT	Data	Histogram parameters (bin widths, number of occurrences, etc.) for data analyzed by INTOHIST.EXE.
LMRPLOT.DAT	Data	Sample data set of histogram parameters created on UNIT IPLOT of ORMONTE during an MVSS run.
PLOTOUT.DAT	Data	Cumulative probability data output from INTOHIST.EXE using LMRPLOT.DAT as input.

Appendix A

MATHEMATICAL FORMS FOR INPUT VARIABLE PROBABILITY DISTRIBUTIONS

The principal difficulty in applying a simulation approach to economic evaluation lies in developing the probabilistic input representations, i.e., distributions (probability density functions or pdfs), for pertinent system performance and/or cost-driving variables. In the usual investment risk analysis applications such as those described by Hertz (Ref. 3.3), many of the input probability distributions used in simulation studies can be developed as frequency histograms from historical or marketing data. However, since a technology development project is often a unique, nonstate-of-the-art project, the accumulation of historical economic or productivity data is often not possible. Thus, some alternative means for developing input probability distributions must be used. It should be remembered that input probability distributions are in essence mathematical models from which the Monte Carlo driver code can draw samples. At the end of the simulation, each input will have its own frequency histogram which should approximate the functional form of the specified input pdf.

In the area of investment analysis, several contributors have suggested solutions to the problem of developing probability distributions. Hertz and others present the argument that probability distributions can be based on subjective judgments made by the manager or decision-maker. In such procedures, three subjective estimates for each investment outcome are obtained: a most likely, an optimistic, and a pessimistic.

For evaluations involving cost projections, no robust selection procedure exists; however, the PDA methodology described in Sec. 7.2 should be very useful in obtaining data from experts. Selection of the input numeric values defining the desired range of the distribution is the most difficult step, followed by determination of the correct type or shape of probability distribution to encompass, or in essence model, the distribution of these values. It is the purpose of this appendix to discuss a few types of important distributions and their appropriate applications. Reference A.1 also discusses the problem of distribution selection.

Distribution Parameters

The input variable probability distributions one encounters in Monte Carlo simulation represent the probability density functions (pdfs) of the random variables from which single point input values, Y_i ($i = 1, NI$, where NI = the number of input variables) are selected for each input vector, Y_i, submitted to the model. The pdf shows only the relative probability for each Y_i.

Among the parameters of interest for a given distribution are the mode, median, and mean. The mode represents the most frequently occurring value, i.e., the incremental range or "bin" in which the most samples fall or the maximum of the defining pdf curve. The median or 50 percentile represents the value for which any random variable on the distribution has an equal chance of falling above or below. Half of the area under a pdf curve will lie to each side of this point. The mean represents the expected value which would be calculated from the random variables. It is calculated during the Monte Carlo procedure by taking the arithmetic average of all the samples taken. The skewness or departure from symmetry of a distribution can be qualitatively assessed by looking at the position of the pdf mean relative to the pdf mode. The value of of the mean relative to the mode is the indicator of the distribution skewness. The following definitions apply:

positive skewness: mean > mode
negative skewness: mean < mode

A dimensionless measure of skewness can be calculated as follows:

$$skewness = \frac{mean - mode}{standard\ deviation}$$

The above two symmetry attributes are sometimes referred to as Pearson's first and second coefficients of skewness.

Another characterization parameter is the kurtosis or "peakedness" of a distribution, expressed as the fourth moment about $\bar{x}$

$$Moment\ coefficient\ of\ kurtosis,\ a_4 = \frac{\overline{(x-\bar{x})^4}}{\sigma^4}$$

where σ is the standard deviation and $\overline{(x-\bar{x})}$ is the average deviation from the distribution mean. For the normal distribution, $a_4 = 3$. A leptokurtic, or highly peaked distribution, has an a_4 value greater than 3; a platykurtic, or lightly peaked distribution, has an a_4 value less than 3. Kurtosis is useful in comparing the shapes of distributions. The properties and uses of a few selected distributions will now be discussed, with continuous distributions to be considered first.

Normal distribution. The normal, bell-shaped, or Gaussian distribution is descriptive of the distribution obtained for most measured natural variables. The mathematical form is expressed as:

$$N(x;\mu,\sigma) = \left[1/\left(\sqrt{2\pi}\,\sigma\right)\right] e^{-(1/2)[(x-\mu)/\sigma]^2} \quad ,$$

where:

μ = the mean of the distribution,
σ^2 = the variance (σ is the standard deviation), and
$\infty < x < \infty$, where x is a continuous random variable.

The mean and standard deviation completely determine the distribution and are the only parameters which need be supplied in the input data. Since the domain of x extends to infinity in both directions, some truncation must be used in computer applications. The return values are limited to a range of $\pm 6\sigma$ from the mean for the ORMONTE driver code utilized in this document.

The use of the normal distribution, by definition, usually implies that some type of measurement, sample, or historical data are available to the analyst. Even though only two parameters are needed to define the normal distribution, the user also assumes that the information one plans to input inherently fits this type of curve.

Log-normal distributions. The parameters for the log-normal distribution are similar to those for the normal distribution. If one considers a random variable x where $Y = ln\ x$ is normally distributed with mean μ_Y, and variance σ_Y^2, a logarithmic transformation to obtain μ_x and σ_x^2 in terms of x is necessary:

$$N(x;\mu_x,\sigma_x) = \frac{1}{x\,\sigma_Y\sqrt{2\pi}}\, e^{-(1/2)[nx - \mu_Y]^2} .$$

This distribution is positively skewed and is often used in economics and for the representation of time distributions in queueing problems. For application in process uncertainty analysis, it can be useful for representing natural parameters where inaccuracy in measurement causes their values to span more than one order of magnitude, such as difficult-to-measure minute spectroscopic cross sections.

Triangular distribution. The triangular distribution has the advantage that any degree of skewness can be easily represented. This is the situation represented when the mode differs from the median or mean. Triangular distributions are defined by a minimum, mode, and maximum value. The minimum and maximum are absolute, i.e., they do not have the long, extremely low probability tails that characterize the normal and exponential distributions.

The triangular distribution has always been popular with decision makers and analysts. Because of the ease with which the triangular distribution can be comprehended by decision makers and implemented by the analyst, it is used frequently in making economic projections.

Three subjective estimates (min, mode, and max) for each input variable are used to form the triangular distributions. Generally speaking, the optimistic estimate is the range endpoint value which would produce the lowest unit cost if a single variable sensitivity analysis were performed over the entire range of the distribution. This is the result which might be experienced if good luck is experienced and everything in the R&D, design, and construction program goes well. The pessimistic estimate represents the opposite outcome, a result which can occur if

development, schedule, or cost disappointments are experienced. The most likely or modal estimate is the "best guess" value for the parameter of interest. This estimate should be expected to appear most frequently over the population of all ensembles during the course of the simulation. This value can be, but is not necessarily, the baseline input value $(Y_i)_{base}$.

Control of skewness can designate the evaluator's degree of optimism or pessimism concerning the range of a given variable; for example, experience indicates that hardware cost variables are more likely to increase than decrease; hence an unit cost distribution is generally skewed toward values above the mode.

Log-triangle distribution. The relationship between the log-triangle and triangle distribution is analogous to that between the normal and log-normal distribution. By use of a log-triangle distribution, which appears as a triangle with a slightly curved side as in Fig. 5.7, one can arbitrarily pick a mode and two endpoints and force the mode and median to be fairly close together, especially when input data span orders of magnitude. This latter type of distribution is frequently employed for theoretical process science related variables used in performance models and for cost variables. This distribution is also useful for variables used as multipliers which lie on both sides of a mode of 1.0. It has the property that the mode and median are the same if distribution range endpoints 1/Z and +Z are used (Z is a multiplier times a fixed value). The use of the log-triangle forces the median closer to the mode for positively skewed triangles.

Other continuous distributions. The following distributions, like the triangular, are mentioned because of their mathematical simplicity and intuitive appeal to the analyst.

The uniform distribution. If a variable has an equal probability of occurring over the range of interest, its distribution can be represented as a horizontal line or flat distribution. This type of distribution is used where a most probable value is not known but for which the variable range can be determined. Use of uniform distributions in cost/risk analysis is often called "range estimating" (Curran: Ref. A.2.)

Trapezoidal distribution. The trapezoidal distribution requires four values for its determination and is a hybrid of the triangular and uniform distributions.

Discrete distributions. If one cannot find a continuous pdf that represents his data, the use of a spike-histogram or a discrete distribution is warranted. Discrete distributions are also often used where integer quantities are considered or "yes-no" decisions are encountered such as in decision tree analysis. Some of the more mathematically complex ones require that the analyst have some intuitive reason for their application and for fitting them to a particular class of data. Among

such distributions are the binomial, geometric, hypergeometric, and Poisson distributions. Others, such as the histogram distributions described below, can be used in a more subjective manner and can be easily constructed by the user of this methodology.

Histogram distributions. Two types of histogram distributions are considered. In the "box" type histogram such as shown in Fig. 5.7, only discrete values or ranges of the random variable exist, i.e., n equals the number of boxes. If the intervals of each box are equal, the cumulative distribution is easily calculated by summation. The return value is found by checking to see which cumulative histogram box the random number intersects and then supplying the midpoint of that range on the integer Y as a return value. When adjacent box-type histograms are used, one is really dealing with several adjacent uniform distributions.

REFERENCES

A.1 McGrath, E. J., et al, *"Techniques for Efficient Monte Carlo Simulation, Vol 1: Selecting Probability Distributions;"* ORNL-RSIC-38 (Vol. 1); Union Carbide Corp., Nuclear Division, Oak Ridge National Laboratory; April 1975.

A.2 Curran, M. W.; "Range Estimating Operating and Manufacturing Costs", *Transactions: American Association of Cost Engineers* (1986); pg. F.1–F.1.8.

Appendix B

DETERMINATION OF NUMBER OF MULTIVARIABLE SENSITIVITY STUDY ITERATIONS REQUIRED

D. L. Hanline
Martin Marietta Energy Systems, Inc.

The running of each Multivariable Sensitivity Study (MVSS) iteration of a process model can require significant computer time. This appendix proposes a methodology where the analyst making the MVSS run calculates the maximum number of iterations (NITER) required, based upon the characteristics of the results to be studied. The user of the MVSS driver can then reduce the number of iterations and still be provided results within his designed limits. Implementation of this capability should be relatively simple.

The advantages of this procedure are reduced turnaround time and reduced mainframe computer costs. The disadvantage is the analyst must designate which results are important figures of merit and give some thought to the levels of accuracy needed for them before an MVSS run is submitted. This procedure assumes that such figures of merit will have nearly normal MVSS histogram output.

With this statistical procedure, the true MVSS results are a set of statistics calculated from a sample number of observations taken from an infinitely large population. Inherent in any sample's estimate is an error or deviation from the true value in the population. The objective is to select a sample size that minimizes the number of MVSS iterations while still meeting maximum error tolerances.

Step 1: Determination of the Maximum Number of MVSS Iterations Required

For each important figure of merit in the results, the analyst must determine within what fraction of a standard deviation the MVSS result must be estimated. Secondly, the analyst must specify the desired confidence interval. The maximum number of iterations required for a figure is calculated by this formula:

$$n_{max} = \left(\frac{Z}{f}\right)^2$$

Z = Number of standard deviations corresponding to a percent confidence interval for a normal distribution.

f = Fraction of a standard deviation accuracy required.

For example: If a 95% confidence interval is desired (Z = 1.96) and required accuracy is one-tenth of a standard deviation,

$$n_{max} = \left(\frac{1.96}{0.1}\right)^2$$

$$= 384$$

The above calculation must be done for each important figure of merit to be reported. The largest value for n_{max} calculated is the maximum number of MVSS iterations to be run. This number is input as NITER into the MVSS driver.

Step 2: Determination of Actual Number of MVSS Iterations Needed

For each important figure of merit in the results, the analyst should determine, wherever possible, the absolute value of any acceptable error in the estimate. For example, it may be decided that for a unit cost, the maximum acceptable error is $2/unit.

The MVSS driver could be programmed to automatically complete 30 iterations, stop, and calculate the standard deviation for each important figure. The number of iterations required can be calculated using this formula:

$$n_{req} = \left(\frac{Zs}{e_{max}}\right)^2$$

s = Estimated standard deviation
e_{max} = Maximum acceptable error

For example: If a 95% confidence interval is required ($Z = 1.96$), s is calculated to be $12/unit, and the maximum acceptable error is $2/unit.

$$n_{req} = \left[\frac{(1.96)\,(12)}{2}\right]^2$$

$$= 138$$

Similar calculations are done for each important figure of merit. The number of required iterations is the largest of the values calculated.

In the above example, an additional 108 iterations are to be run. The estimated standard deviation is recalculated, and if the newly calculated standard deviation is larger than the initial estimate, additional iterations are required.

Appendix C

ANALYTICAL INVERSION OF THE TRIANGULAR DISTRIBUTION

Consider the triangular distribution for variable X in Fig. C.1 with minimum X_L, mode X_M, and maximum X_H. We wish to relate X_o, a point on the distribution, with a random number, Q, representing the cumulative distribution for X. To get the cdf we need to integrate the analytical expression for the triangle's sides from X_L to a generic X_o. This area, Q, represents the area under this triangle from X_L (a known value) to X_o and also represents the cumulative probability of variable X being less than or equal to X_O. Q has a known value between 0 and 1 and also represents the uniformly distributed random number drawn for variable X during an iteration as shown in Fig. C.1. We wish to find the analytical expression for X_O in terms of Q.

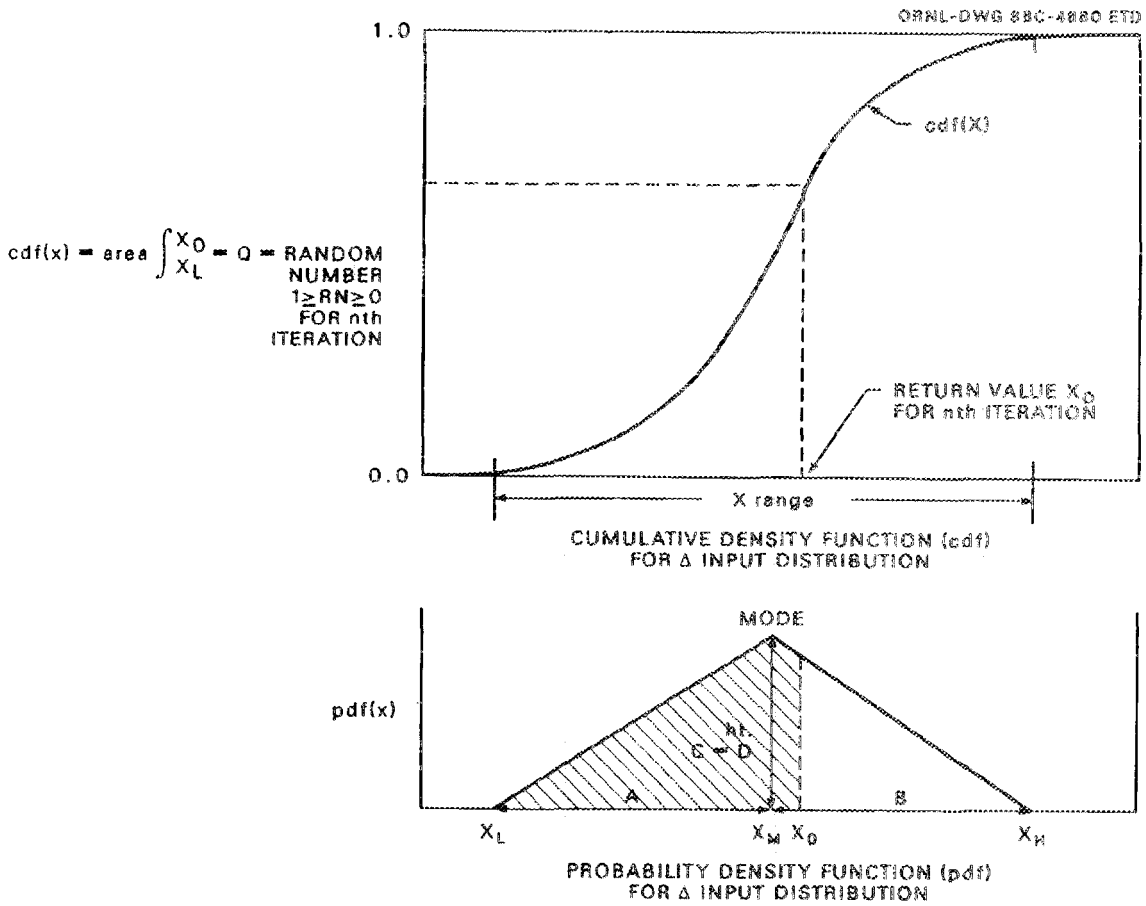


Fig. C.1. Inversion of the triangular distribution.

The inversion routine will consider the triangular distribution broken into two right triangles, Δ_{left} and Δ_{right} separated by the altitude at the mode, X_M. We want to calculate the total area under the two triangles and obtain an expression for the area, Q, under one or both triangles from X_L to X_O as a function of X. The area under the entire triangle must sum to 1.0 by definition of the cumulative probability for X over the entire range.

First solve for the triangle altitude C in terms of the base lengths for the two right triangles:

$$(A + B)\, C/2 = \text{area under both right triangles} = 1.0$$

Therefore,

$$C = 2/(A + B)$$

The use of the magnitude of the two right triangle bases, A + B, allows us to avert the algebraic coordinate transformation required to do all calculations in terms of X_L, X_M, and X_H.

We must next determine whether the value X_O associated with a given random number Q lies within the left (A-base) or right (B-base) triangle. The areas of these two triangles are:

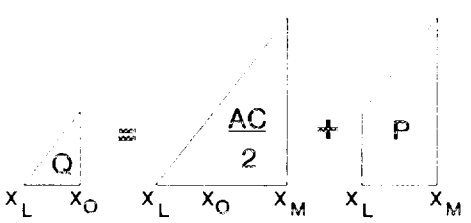
Left: AC/2 Right: BD/2 .

The variable P is an indicator of which triangle will contain X_O and has the geometric representation shown below.

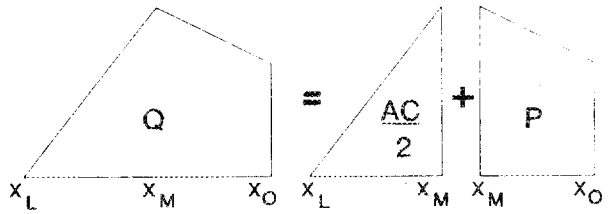
Solving for P:

$$P = Q - \frac{AC}{2} \ .$$

If P is negative, X_O lies beneath the leftmost triangle



If P is positive, X_O lies beneath the right most triangle.

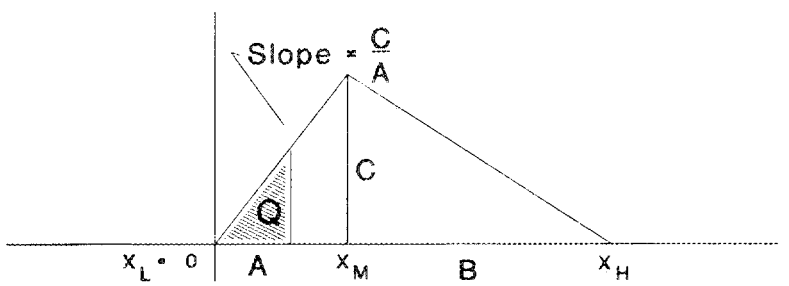


For the left most triangle we now solve for X_O in term of Q:

The hypotenuse of the left triangle has an equation of the form

$$Y = \left(\frac{C}{A}\right) X + Z$$

where C/A is its slope and Z an intercept. For now the origin is assumed to be at $X_L = 0$, hence $Z = 0$



The area Q (area under Δ from O *to* X_O) is found by integration

$$Q = \int_O^{X_O} \left(\frac{C}{A}\right) X dX = \left(\frac{C}{2A}\right) X^2 \Big|_O^{X_O}$$

$$Q = \frac{C X_O^2}{2A} \ .$$

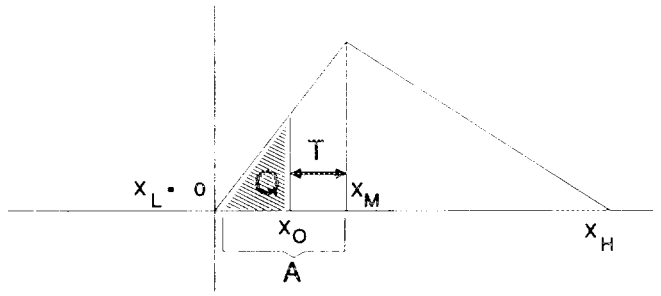
Solve for the square of the return value X_O, substituting $P + \frac{AC}{2}$ for Q

$$DISC = X_O^2 = \frac{2AO}{C} = A^2 + \frac{2PA}{C} \ .$$

For ORMONTE applications, we are interested in using the triangle mode as the frame of reference. T tells how far from the mode the true return value will lie

(for a return value in the leftmost triangle, T will be negative as shown below).

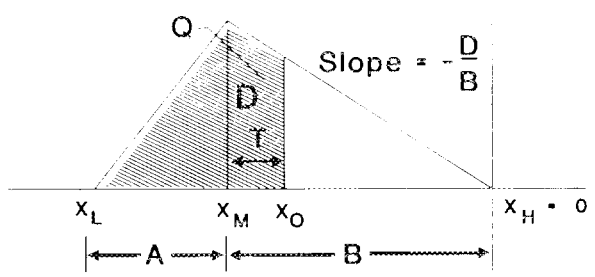
$$T = -A + X_O$$



The return value X for subroutine XVALUE is then calculated as follows:

$$X = X_M + T \ .$$

For an X_O in the rightmost triangle (P > O), using X_H as the origin, Q is calculated as follows. Note that the origin is at X_H to avoid dealing with an intercept.



$$Q = \underbrace{\left(\frac{AD}{2}\right)}_{\text{leftmost } \Delta} + \underbrace{\left(\frac{BD}{2}\right)}_{\text{rightmost } \Delta} - \int_{X_O}^{O} \underbrace{\left(-\frac{D}{B}\right)}_{\text{slope}} X dX$$

$$= \frac{(A + B)\,D}{2} - \frac{DX_O^2}{2B} \ .$$

Substituting $Q = P + \frac{AD}{2}$ and solving for x_0^2

$$DISC = X^2 = B^2 - \frac{B}{D} P \quad .$$

T tells how for to the right of the mode the actual return value lies

$$T = B - X_o \quad ,$$

and again for subroutine XVALUE

$$X = X_M + T \quad .$$

Appendix D

SAMPLE PROBLEM

Data for and Output of ORMONTE
for
Single Variable Sensitivity Analysis (SVSS) Option

Input Data for SVSS Option

Unit 4: Option Data

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [SVSS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]
```

Unit 8: Base Case Data (Sensitivity Variables)

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
ALPH1     200.
ALPH2     50.
BETA3     10.
BETA4     1.
ZETA5     1.
EXP6      1.
GAMMA7    8.
GAMMA8    10.
PHI9      18.
EPS10     1.
NUM11     1.
PERF      -1.
PCOST     -1.
CUNIT     -1.
```

Unit 5: Non-sensitivity Data for User's Model

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT
C1      1.5
C2      2.0
C3      50.
```

Unit 11: SVSS Data

```
TITLE:  SVSS DRIVER TEST CASE  DSN=SVSS.DAT
00002  003   12   13   14
001  ALPH1   06      100.        150.        200.        300.
                     400.        500.
000
001
002        100.
00001  003   12   13   14
002  ALPH2   04       25.        50.         75.         100.

000
00001  003   12   13   14
003  BETA3   04       6.          8.          10.         12.

000
00001  003   12   13   14
004  BETA4   05      .75          .9          1.00        1.1
                     1.3
000
00001  003   12   13   14
005  ZETA5   05      .75          0.9         1.          1.1
                     1.2
000
00001  003   12   13   14
006  EXP6    05      0.5          .75         1.0         1.5
                     2.0
000
00001  003   12   13   14
007  GAMMA7  05      5.           6.5         8.          9.5
                     11.
000
00001  003   12   13   14
008  GAMMA8  06      5.           7.5         10.         14.
                     18.          22.
000
00001  003   12   13   14
009  PHI9    05      5.           12.         18.         24.
                     30.
000
00001  003   12   13   14
010  EPS10   05      0.8          0.9         1.0         1.1
                     1.3
000
00003  003   12   13   14
011  NUM11   05      0.9          0.95        1.          1.08
                     1.15
000
001
002        100.
002
001        300.
002        100.
00000
```

Output of Unit IPRNT or (6)

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]    INOPT [ 4]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [SVSS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]


READIN OF SENS. OPTION DATA ON UNIT  4 COMPLETED
```

Output of REAPOP

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000

READIN OF /MASTER/ INITIALIZATION DATA
ON UNIT  8 COMPLETED
```

Output of RDMAST

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT

 C1                    1.500
 C2                    2.000
 C3                   50.000
MODEL NON-SENS. DATA READIN COMPLETED ON UNIT  5
```

Output of RDMODL

```
VALUES IN /MASTER/ FOR ITERATION NO     0


TITLE: 14 VARIABLE TEST CASE: DSN=MASIN1.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000    0
```

Output of WRMAST (before call to MODEL)

```
PRINTOUT OF USERS MODEL FOR ITERATION #    0

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000

INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   200.000        ALPH2=    50.000        BETA3=    10.000
        BETA4=     1.000        ZETA5=     1.000        EXP6 =     1.000
        GAMMA7=    8.000        GAMMA8=   10.000        PHI9 =    18.000
        EPS10=     1.000        NUM11=     1.000

OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    10.007        PCOST=    25.000        CUNIT=     2.498
```

Output of MODEL (Base Case)

```
VALUES IN /MASTER/ FOR ITERATION NO     0


TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          10.007
 13   PCOST         25.000
 14   CUNIT          2.498
```

Base Outputs Now Appear

Output of WRMAST (after call to MODEL)
Base Y-array values are stored

```
OPTION = SVSS

TITLE:  SVSS DRIVER TEST CASE  DSN=SVSS.DAT

NO. OF /MASTER/ Y(150) SENSITIVITY VARIABLE LOCATIONS
USED (INPUT + OUTPUT)
  14
BASELINE VALUES OF Y(1) THROUGH Y(NTOT)
    0.20000E+03    0.50000E+02    0.10000E+02    0.10000E+01    0.10000E+01
    0.10000E+01    0.80000E+01    0.10000E+02    0.18000E+02    0.10000E+01
    0.10000E+01    0.10007E+02    0.25000E+02    0.24983E+01
```

Output of SVSSOP

```
 SVSS CASE OR PLOT NUMBER    1
NO. OF CURVES ON THIS PLOT =  2
NO. OF FIGURES OF MERIT FOR TABLES:   3
THESE OUTPUTS & THEIR INDICES ARE:

PERF       OR   Y( 12)

PCOST      OR   Y( 13)

CUNIT      OR   Y( 14)

THE INPUT VARIABLE IS ALPH1  OR Y(  1)
 6 DIFFERENT VALUES ARE CONSIDERED FOR EACH CURVE


INDEX,NPTS,NUMCRV
   1    6    2
```

Output of SVSSOP

```
CASE OR PLOT NO:    1
INPUT VARIABLE:  ALPH1  OR Y(  1)
CURVE NO.  1 OF  2
POINT NO.  1 OF  6
VALUE FOR THE INPUT VARIABLE IS:   100.0000
NO. OF NON-BASE PARAMETER VALUES FOR THIS CURVE: 0
PARAMETER VALUES ARE:

CHG.#.    NAME & Y-ARRAY POS.                    VALUE

NO PARAMETERS ARE ALTERED FROM BASE VALUES

    0
```

Output of SVSSOP

```
PRINTOUT OF USERS MODEL FOR ITERATION #    0

CONSTANTS (NON-SENSITIVITY):

          C1=     1.500
          C2=     2.000
          C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

          ALPH1=   100.000        ALPH2=    50.000        BETA3=    10.000
          BETA4=     1.000        ZETA5=     1.000        EXP6 =     1.000
          GAMMA7=    8.000        GAMMA8=   10.000        PHI9 =    18.000
          EPS10=     1.000        NUM11=     1.000


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

          PERF=     9.174        PCOST=    17.500        CUNIT=     1.908
```

Output of MODEL (first non-baseline point)

```
CASE OR PLOT NO:    1
INPUT VARIABLE:  ALPH1  OR Y(  1)
CURVE NO.  1 OF  2
POINT NO.  2 OF  6
VALUE FOR THE INPUT VARIABLE IS:   150.0000
NO. OF NON-BASE PARAMETER VALUES FOR THIS CURVE: 0
PARAMETER VALUES ARE:

CHG.#.    NAME & Y-ARRAY POS.                    VALUE

NO PARAMETERS ARE ALTERED FROM BASE VALUES

    0
```

Output of SVSSOP (second non-baseline point)

```
PRINTOUT OF USERS MODEL FOR ITERATION #    0

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   150.000        ALPH2=    50.000        BETA3=    10.000
        BETA4=     1.000        ZETA5=     1.000        EXP6 =     1.000
        GAMMA7=    8.000        GAMMA8=   10.000        PHI9 =    18.000
        EPS10=     1.000        NUM11=     1.000


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=     9.590        PCOST=    21.250        CUNIT=     2.216
```

Output of MODEL (second non-baseline point)

PLEASE NOTE: End of partial output of SVSS run.

DEBUG output from SVSSOP on Unit 14 or IBUG (one point)

```
TITLE:  SVSS DRIVER TEST CASE  DSN=SVSS.DAT

 1 ALPH1    6  0.10000E+03 0.15000E+03 0.20000E+03 0.30000E+03
               0.40000E+03 0.50000E+03 0.00000E+00 0.00000E+00
FOR ALPH1  EQUAL TO  0.10000E+03 THE Y ARRAY IS GIVEN BY:

    1   0.10000E+03     2   0.50000E+02     3   0.10000E+02     4   0.10000E+01
    5   0.10000E+01     6   0.10000E+01     7   0.80000E+01     8   0.10000E+02
    9   0.18000E+02    10   0.10000E+01    11   0.10000E+01    12   0.91736E+01
   13   0.17500E+02    14   0.19076E+01    15   0.00000E+00    16   0.00000E+00
   17   0.00000E+00    18   0.00000E+00    19   0.00000E+00    20   0.00000E+00
   21   0.00000E+00    22   0.00000E+00    23   0.00000E+00    24   0.00000E+00
   25   0.00000E+00    26   0.00000E+00    27   0.00000E+00    28   0.00000E+00
   29   0.00000E+00    30   0.00000E+00    31   0.00000E+00    32   0.00000E+00
   33   0.00000E+00    34   0.00000E+00    35   0.00000E+00    36   0.00000E+00
   37   0.00000E+00    38   0.00000E+00    39   0.00000E+00    40   0.00000E+00
   41   0.00000E+00    42   0.00000E+00    43   0.00000E+00    44   0.00000E+00
   45   0.00000E+00    46   0.00000E+00    47   0.00000E+00    48   0.00000E+00
   49   0.00000E+00    50   0.00000E+00    51   0.00000E+00    52   0.00000E+00
   53   0.00000E+00    54   0.00000E+00    55   0.00000E+00    56   0.00000E+00
   57   0.00000E+00    58   0.00000E+00    59   0.00000E+00    60   0.00000E+00
   61   0.00000E+00    62   0.00000E+00    63   0.00000E+00    64   0.00000E+00
   65   0.00000E+00    66   0.00000E+00    67   0.00000E+00    68   0.00000E+00
   69   0.00000E+00    70   0.00000E+00    71   0.00000E+00    72   0.00000E+00
   73   0.00000E+00    74   0.00000E+00    75   0.00000E+00    76   0.00000E+00
   77   0.00000E+00    78   0.00000E+00    79   0.00000E+00    80   0.00000E+00
   81   0.00000E+00    82   0.00000E+00    83   0.00000E+00    84   0.00000E+00
   85   0.00000E+00    86   0.00000E+00    87   0.00000E+00    88   0.00000E+00
   89   0.00000E+00    90   0.00000E+00    91   0.00000E+00    92   0.00000E+00
   93   0.00000E+00    94   0.00000E+00    95   0.00000E+00    96   0.00000E+00
   97   0.00000E+00    98   0.00000E+00    99   0.00000E+00   100   0.00000E+00
  101   0.00000E+00   102   0.00000E+00   103   0.00000E+00   104   0.00000E+00
  105   0.00000E+00   106   0.00000E+00   107   0.00000E+00   108   0.00000E+00
  109   0.00000E+00   110   0.00000E+00   111   0.00000E+00   112   0.00000E+00
  113   0.00000E+00   114   0.00000E+00   115   0.00000E+00   116   0.00000E+00
```

117	0.00000E+00	118	0.00000E+00	119	0.00000E+00	120	0.00000E+00
121	0.00000E+00	122	0.00000E+00	123	0.00000E+00	124	0.00000E+00
125	0.00000E+00	126	0.00000E+00	127	0.00000E+00	128	0.00000E+00
129	0.00000E+00	130	0.00000E+00	131	0.00000E+00	132	0.00000E+00
133	0.00000E+00	134	0.00000E+00	135	0.00000E+00	136	0.00000E+00
137	0.00000E+00	138	0.00000E+00	139	0.00000E+00	140	0.00000E+00
141	0.00000E+00	142	0.00000E+00	143	0.00000E+00	144	0.00000E+00
145	0.00000E+00	146	0.00000E+00	147	0.00000E+00	148	0.00000E+00
149	0.00000E+00	150	0.00000E+00				

OUTPUT of SVSSOP on Unit 15 or IPLOT
(summary tables for plot preparation)

```
TITLE:  SVSS DRIVER TEST CASE  DSN=SVSS.DAT

NO. OF /MASTER/ Y(150) SENSITIVITY VARIABLE LOCATIONS
USED (INPUT + OUTPUT)
  14
BASELINE VALUES OF Y(1) THROUGH Y(NTOT)
    0.20000E+03     0.50000E+02     0.10000E+02     0.10000E+01     0.10000E+01
    0.10000E+01     0.80000E+01     0.10000E+02     0.18000E+02     0.10000E+01
    0.10000E+01     0.10007E+02     0.25000E+02     0.24983E+01
-----------------------------------------------------------------------------
INDEX,NPTS,NUMCRV
   1    6    2
# OF PARAMETER CHANGES=    0
PT#      ALPH1          PERF          PCOST         CUNIT         ******        ******
 1    1.00000E+02  9.17361E+00  1.75000E+01  1.90765E+00  0.00000E-01  0.00000E-01
 2    1.50000E+02  9.59028E+00  2.12500E+01  2.21579E+00  0.00000E-01  0.00000E-01
 3    2.00000E+02  1.00069E+01  2.50000E+01  2.49827E+00  0.00000E-01  0.00000E-01
 4    3.00000E+02  1.08403E+01  3.25000E+01  2.99808E+00  0.00000E-01  0.00000E-01
 5    4.00000E+02  1.16736E+01  4.00000E+01  3.42653E+00  0.00000E-01  0.00000E-01
 6    5.00000E+02  1.25069E+01  4.75000E+01  3.79789E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------------
# OF PARAMETER CHANGES=    1
  2     ALPH2    100.000
PT#      ALPH1          PERF          PCOST         CUNIT         ******        ******
 1    1.00000E+02  1.47292E+01  1.75000E+01  1.18812E+00  0.00000E-01  0.00000E-01
 2    1.50000E+02  1.51458E+01  2.12500E+01  1.40303E+00  0.00000E-01  0.00000E-01
 3    2.00000E+02  1.55625E+01  2.50000E+01  1.60643E+00  0.00000E-01  0.00000E-01
 4    3.00000E+02  1.63958E+01  3.25000E+01  1.98221E+00  0.00000E-01  0.00000E-01
 5    4.00000E+02  1.72292E+01  4.00000E+01  2.32164E+00  0.00000E-01  0.00000E-01
 6    5.00000E+02  1.80625E+01  4.75000E+01  2.62976E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------------
```

```
INDEX,NPTS,NUMCRV
   2    4    1
# OF PARAMETER CHANGES=    0
PT#      ALPH2          PERF          PCOST         CUNIT         ******        ******
 1    2.50000E+01  7.22917E+00  2.50000E+01  3.45821E+00  0.00000E-01  0.00000E-01
 2    5.00000E+01  1.00069E+01  2.50000E+01  2.49827E+00  0.00000E-01  0.00000E-01
 3    7.50000E+01  1.27847E+01  2.50000E+01  1.95546E+00  0.00000E-01  0.00000E-01
 4    1.00000E+02  1.55625E+01  2.50000E+01  1.60643E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------
INDEX,NPTS,NUMCRV
   3    4    1
# OF PARAMETER CHANGES=    0
PT#      BETA3          PERF          PCOST         CUNIT         ******        ******
```

```
 1    6.00000E+00  1.11181E+01  2.50000E+01  2.24859E+00  0.00000E-01  0.00000E-01
 2    8.00000E+00  1.04236E+01  2.50000E+01  2.39840E+00  0.00000E-01  0.00000E-01
 3    1.00000E+01  1.00069E+01  2.50000E+01  2.49827E+00  0.00000E-01  0.00000E-01
 4    1.20000E+01  9.72917E+00  2.50000E+01  2.56959E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------
INDEX,NPTS,NUMCRV
   4    5    1
# OF PARAMETER CHANGES=    0
PT#      BETA4          PERF          PCOST         CUNIT         ******        ******
 1    7.50000E-01  9.31250E+00  2.50000E+01  2.68456E+00  0.00000E-01  0.00000E-01
 2    9.00000E-01  9.72917E+00  2.50000E+01  2.56959E+00  0.00000E-01  0.00000E-01
 3    1.00000E+00  1.00069E+01  2.50000E+01  2.49827E+00  0.00000E-01  0.00000E-01
 4    1.10000E+00  1.02847E+01  2.50000E+01  2.43079E+00  0.00000E-01  0.00000E-01
 5    1.30000E+00  1.08403E+01  2.50000E+01  2.30621E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------
```

INDEX,NPTS,NUMCRV

5 5 1

OF PARAMETER CHANGES= 0

PT#	ZETA5	PERF	PCOST	CUNIT	******	******
1	7.50000E-01	1.33426E+01	2.50000E+01	1.87370E+00	0.00000E-01	0.00000E-01
2	9.00000E-01	1.11188E+01	2.50000E+01	2.24844E+00	0.00000E-01	0.00000E-01
3	1.00000E+00	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	1.10000E+00	9.09722E+00	2.50000E+01	2.74809E+00	0.00000E-01	0.00000E-01
5	1.20000E+00	8.33912E+00	2.50000E+01	2.99792E+00	0.00000E-01	0.00000E-01

INDEX,NPTS,NUMCRV

6 5 1

OF PARAMETER CHANGES= 0

PT#	EXP6	PERF	PCOST	CUNIT	******	******
1	5.00000E-01	1.00196E+01	2.50000E+01	2.49510E+00	0.00000E-01	0.00000E-01
2	7.50000E-01	1.00117E+01	2.50000E+01	2.49708E+00	0.00000E-01	0.00000E-01
3	1.00000E+00	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	1.50000E+00	1.00025E+01	2.50000E+01	2.49939E+00	0.00000E-01	0.00000E-01
5	2.00000E+00	1.00009E+01	2.50000E+01	2.49978E+00	0.00000E-01	0.00000E-01

INDEX,NPTS,NUMCRV

7 5 1

OF PARAMETER CHANGES= 0

PT#	GAMMA7	PERF	PCOST	CUNIT	******	******
1	5.00000E+00	1.00111E+01	2.50000E+01	2.49723E+00	0.00000E-01	0.00000E-01
2	6.50000E+00	1.00085E+01	2.50000E+01	2.49786E+00	0.00000E-01	0.00000E-01
3	8.00000E+00	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	9.50000E+00	1.00058E+01	2.50000E+01	2.49854E+00	0.00000E-01	0.00000E-01
5	1.10000E+01	1.00051E+01	2.50000E+01	2.49874E+00	0.00000E-01	0.00000E-01

INDEX,NPTS,NUMCRV
8 6 1
OF PARAMETER CHANGES= 0

PT#	GAMMA8	PERF	PCOST	CUNIT	******	******
1	5.00000E+00	1.00069E+01	1.75000E+01	1.74879E+00	0.00000E-01	0.00000E-01
2	7.50000E+00	1.00069E+01	2.12500E+01	2.12353E+00	0.00000E-01	0.00000E-01
3	1.00000E+01	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	1.40000E+01	1.00069E+01	3.10000E+01	3.09785E+00	0.00000E-01	0.00000E-01
5	1.80000E+01	1.00069E+01	3.70000E+01	3.69743E+00	0.00000E-01	0.00000E-01
6	2.20000E+01	1.00069E+01	4.30000E+01	4.29702E+00	0.00000E-01	0.00000E-01

INDEX,NPTS,NUMCRV
9 5 1
OF PARAMETER CHANGES= 0

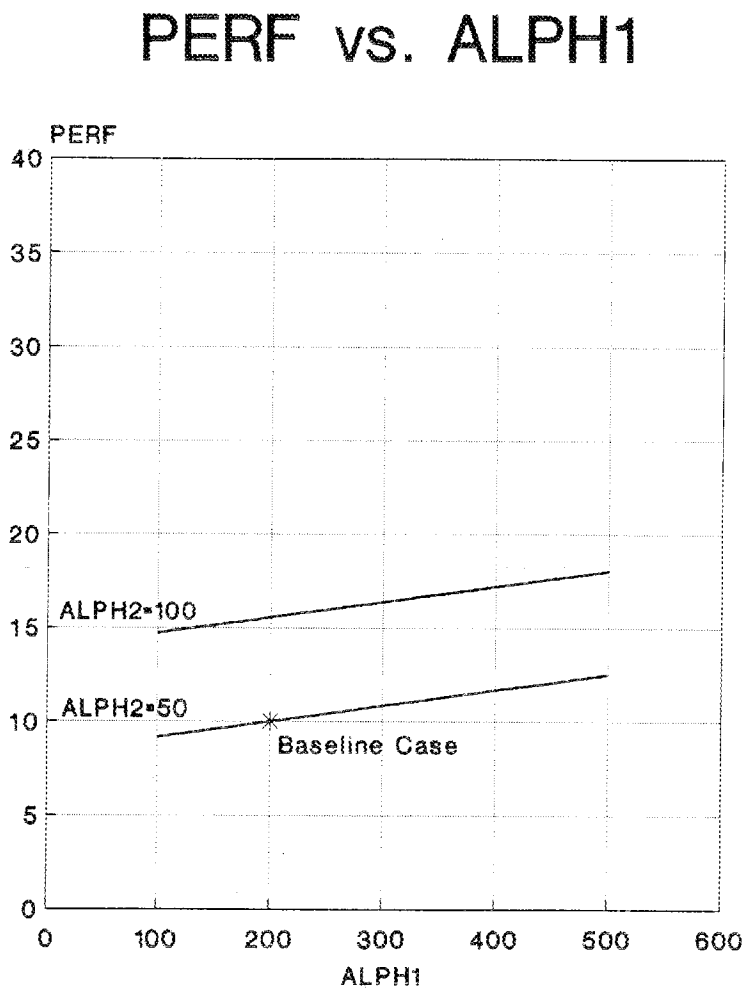
PT#	PHI9	PERF	PCOST	CUNIT	******	******
1	5.00000E+00	3.60250E+01	2.50000E+01	6.93963E-01	0.00000E-01	0.00000E-01
2	1.20000E+01	1.50104E+01	2.50000E+01	1.66551E+00	0.00000E-01	0.00000E-01
3	1.80000E+01	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	2.40000E+01	7.50521E+00	2.50000E+01	3.33102E+00	0.00000E-01	0.00000E-01
5	3.00000E+01	6.00417E+00	2.50000E+01	4.16377E+00	0.00000E-01	0.00000E-01

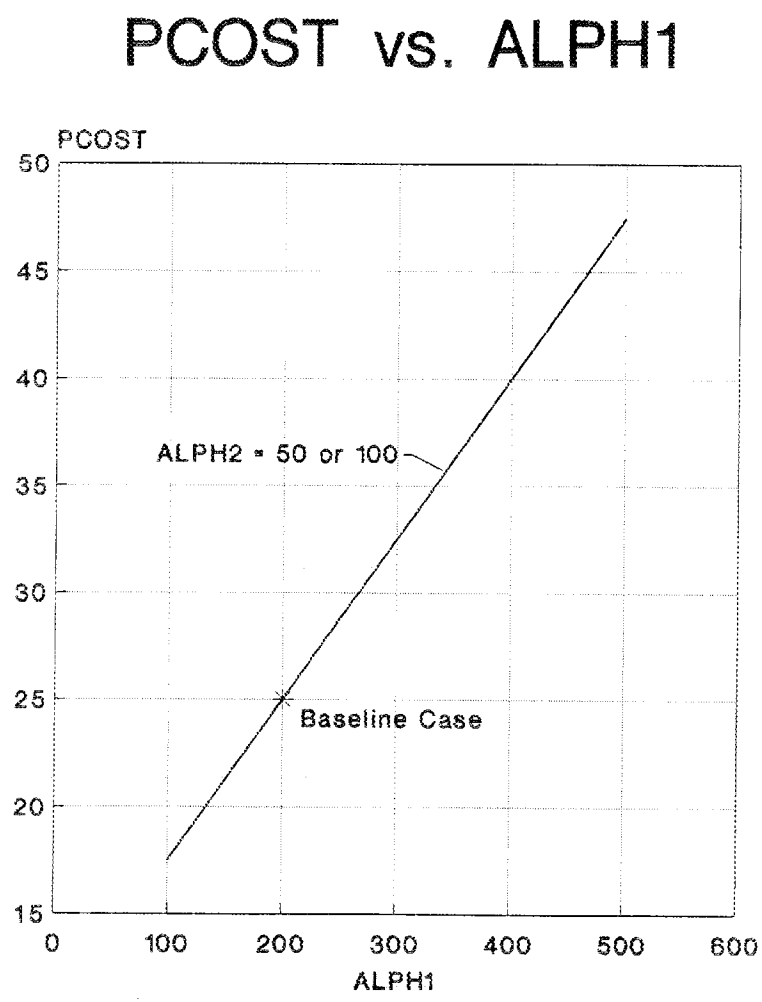
INDEX,NPTS,NUMCRV
10 5 1
OF PARAMETER CHANGES= 0

PT#	EPS10	PERF	PCOST	CUNIT	******	******
1	8.00000E-01	1.00087E+01	2.50000E+01	2.49783E+00	0.00000E-01	0.00000E-01
2	9.00000E-01	1.00077E+01	2.50000E+01	2.49807E+00	0.00000E-01	0.00000E-01
3	1.00000E+00	1.00069E+01	2.50000E+01	2.49827E+00	0.00000E-01	0.00000E-01
4	1.10000E+00	1.00063E+01	2.50000E+01	2.49842E+00	0.00000E-01	0.00000E-01
5	1.30000E+00	1.00053E+01	2.50000E+01	2.49867E+00	0.00000E-01	0.00000E-01

```
INDEX,NPTS,NUMCRV
  11    5    3
# OF PARAMETER CHANGES=    0
PT#      NUM11          PERF          PCOST         CUNIT         ******        ******
 1    9.00000E-01  1.00063E+01  2.35000E+01  2.34853E+00  0.00000E-01  0.00000E-01
 2    9.50000E-01  1.00066E+01  2.42500E+01  2.42340E+00  0.00000E-01  0.00000E-01
 3    1.00000E+00  1.00069E+01  2.50000E+01  2.49827E+00  0.00000E-01  0.00000E-01
 4    1.08000E+00  1.00075E+01  2.62000E+01  2.61804E+00  0.00000E-01  0.00000E-01
 5    1.15000E+00  1.00080E+01  2.72500E+01  2.72283E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------------
# OF PARAMETER CHANGES=    1
  2     ALPH2    100.000
PT#      NUM11          PERF          PCOST         CUNIT         ******        ******
 1    9.00000E-01  1.55618E+01  2.35000E+01  1.51011E+00  0.00000E-01  0.00000E-01
```

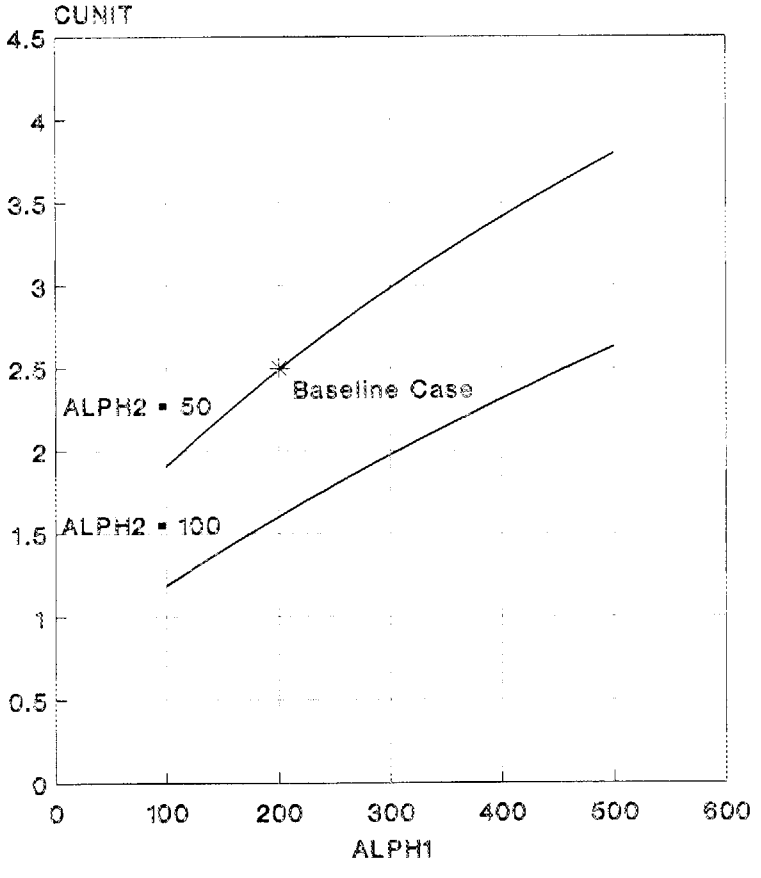
```
 2    9.50000E-01  1.55622E+01  2.42500E+01  1.55827E+00  0.00000E-01  0.00000E-01
 3    1.00000E+00  1.55625E+01  2.50000E+01  1.60643E+00  0.00000E-01  0.00000E-01
 4    1.08000E+00  1.55631E+01  2.62000E+01  1.68347E+00  0.00000E-01  0.00000E-01
 5    1.15000E+00  1.55635E+01  2.72500E+01  1.75089E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------------
# OF PARAMETER CHANGES=    2
  1     ALPH1    300.000
  2     ALPH2    100.000
PT#      NUM11          PERF          PCOST         CUNIT         ******        ******
 1    9.00000E-01  1.63951E+01  3.02500E+01  1.84506E+00  0.00000E-01  0.00000E-01
 2    9.50000E-01  1.63955E+01  3.13750E+01  1.91364E+00  0.00000E-01  0.00000E-01
 3    1.00000E+00  1.63958E+01  3.25000E+01  1.98221E+00  0.00000E-01  0.00000E-01
 4    1.08000E+00  1.63964E+01  3.43000E+01  2.09192E+00  0.00000E-01  0.00000E-01
 5    1.15000E+00  1.63969E+01  3.58750E+01  2.18792E+00  0.00000E-01  0.00000E-01
-----------------------------------------------------------------------------
```



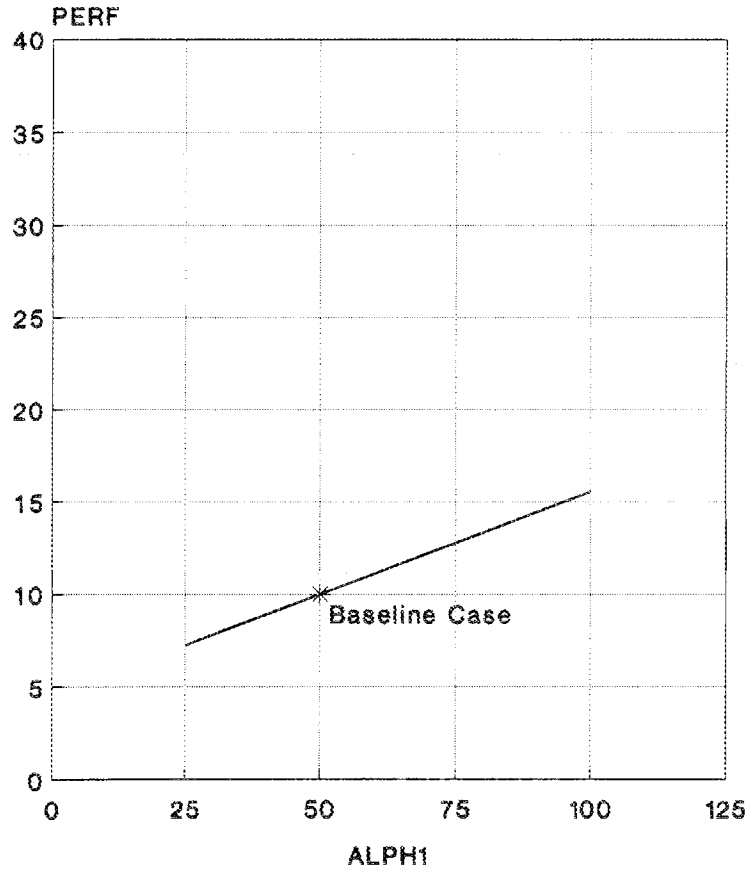


Plotted Results of
Single Value Sensitivity Analyses

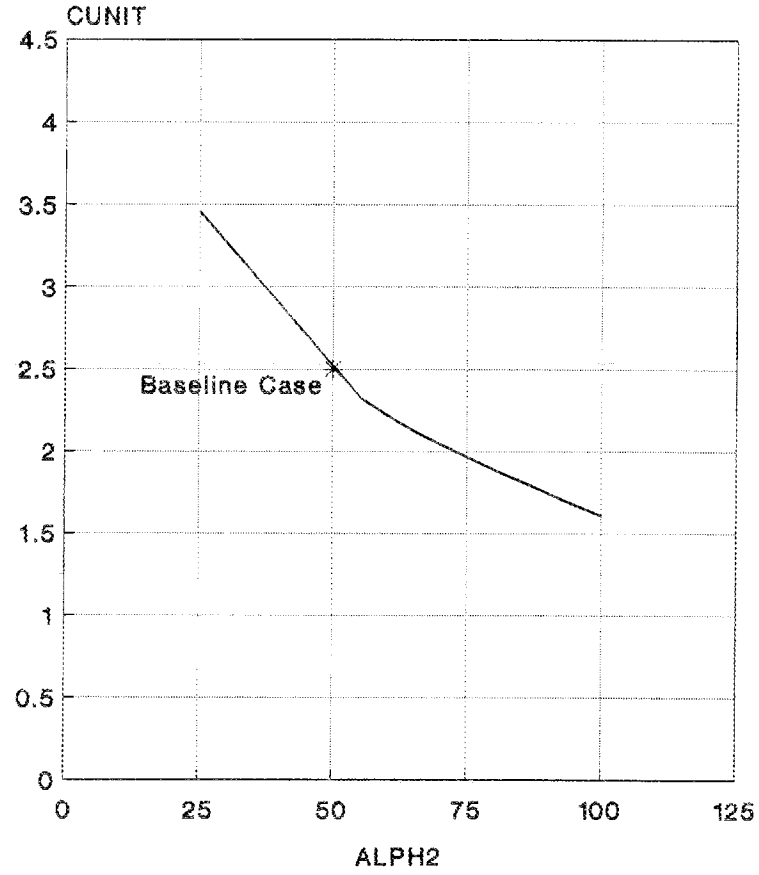
CUNIT vs. ALPH1



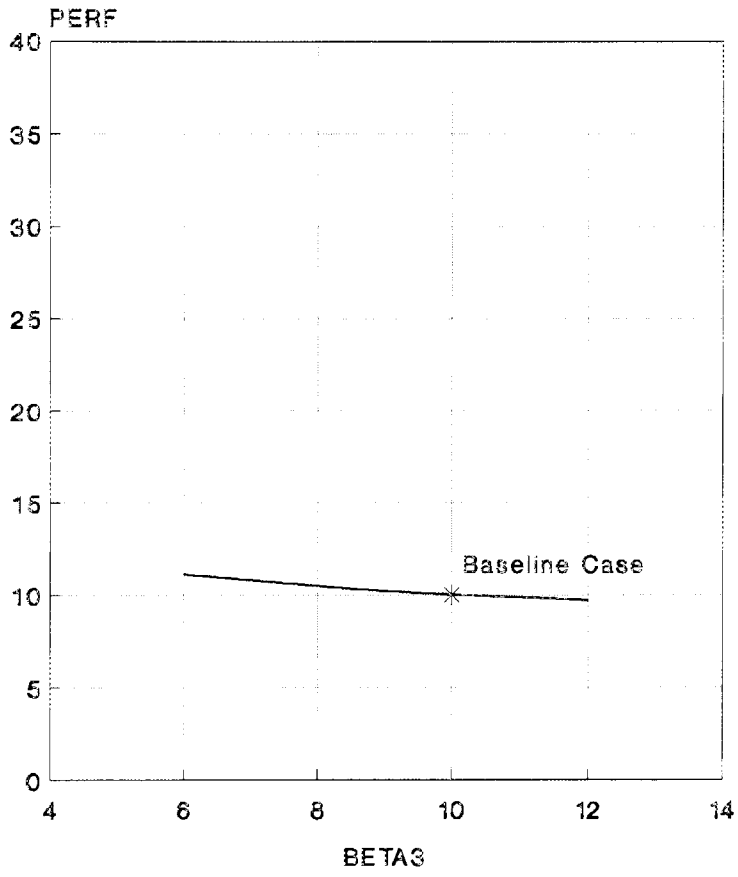
PERF vs. ALPH2



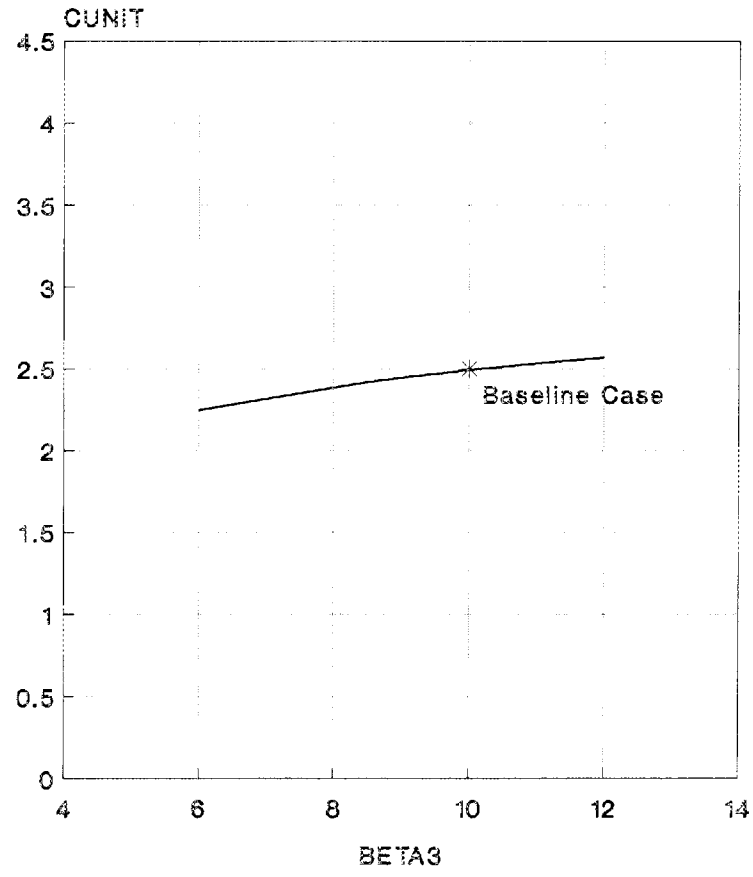
CUNIT vs. ALPH2



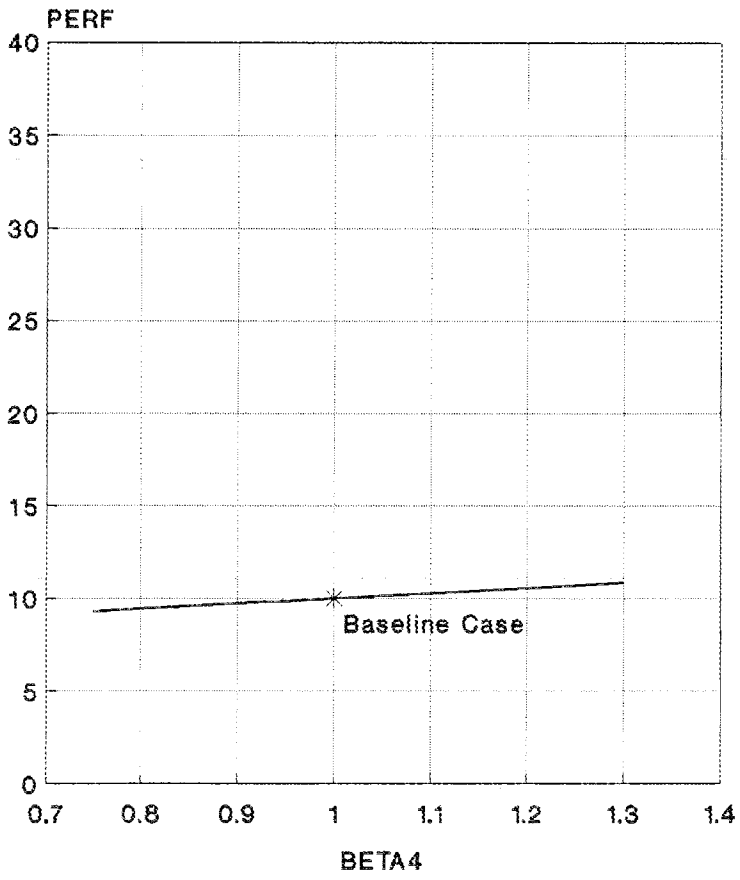
PERF vs. BETA3



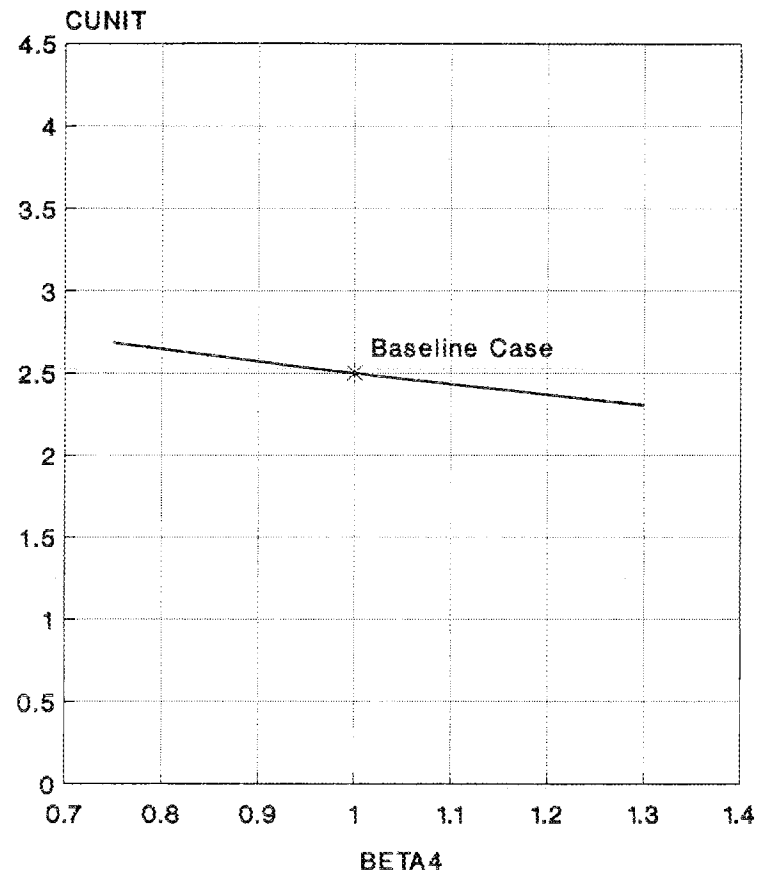
CUNIT vs. BETA3



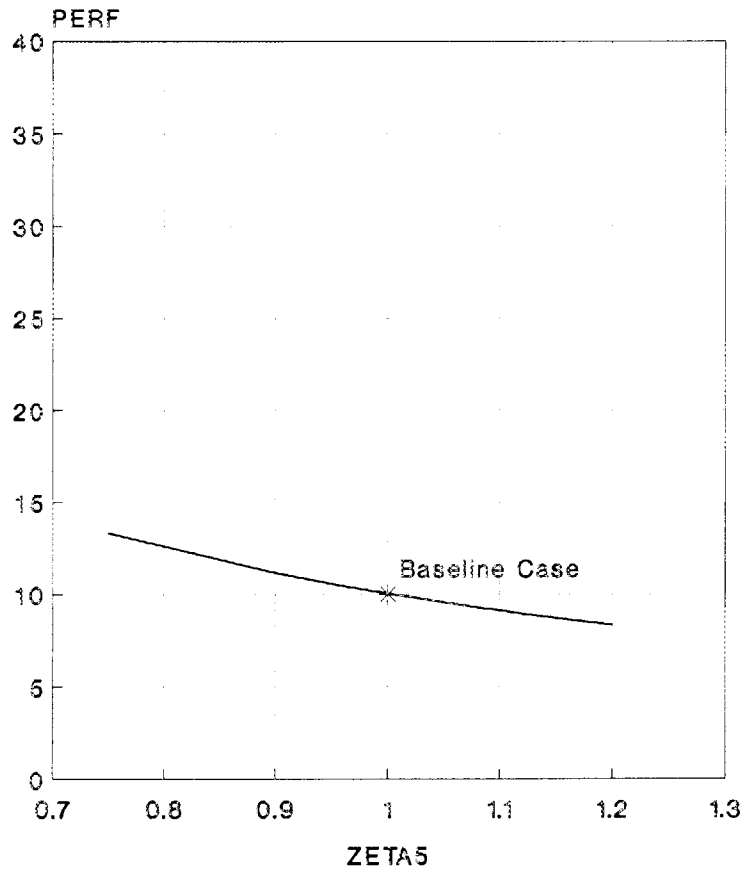
PERF vs. BETA4



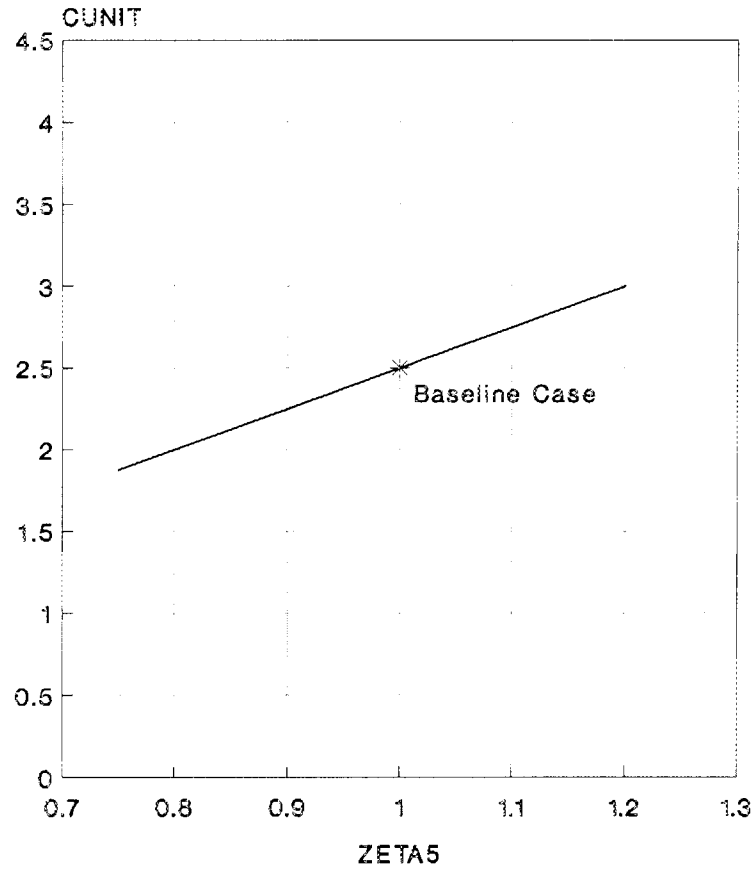
CUNIT vs. BETA4



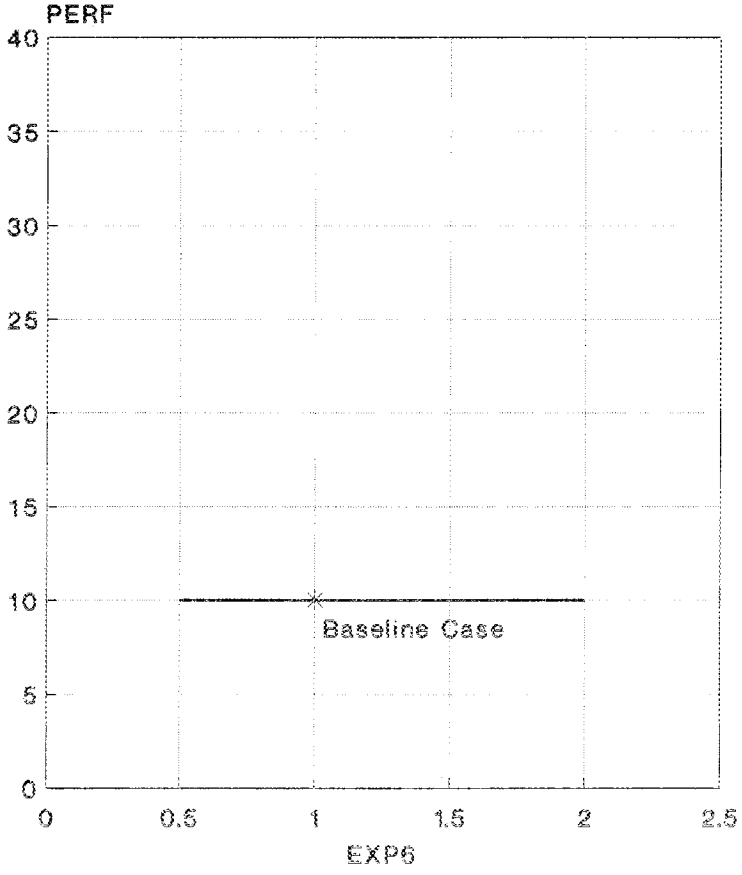
PERF vs. ZETA5



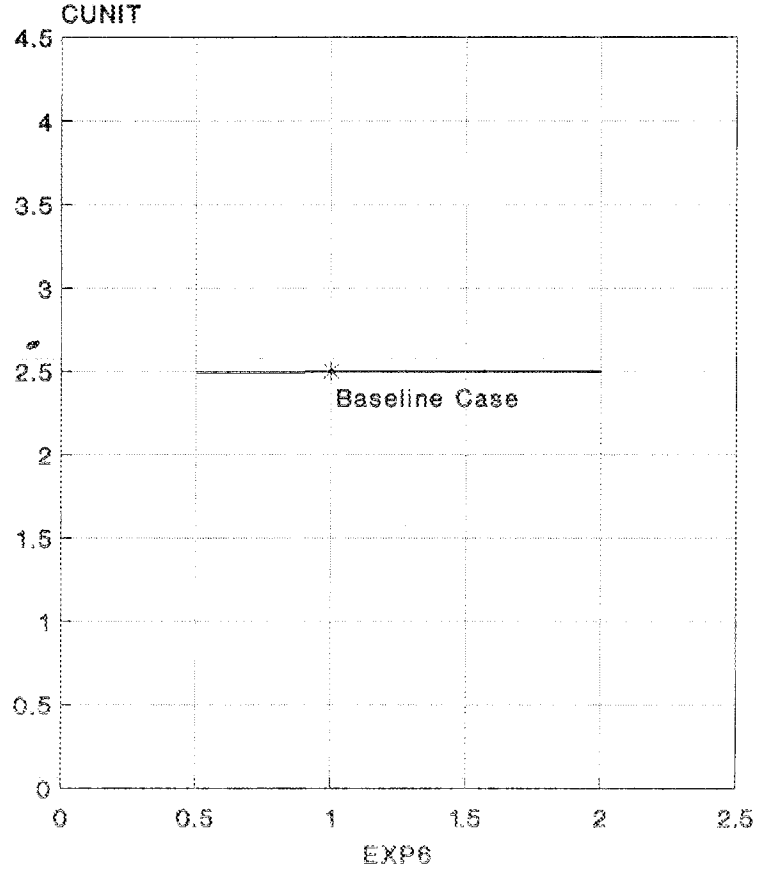
CUNIT vs. ZETA5



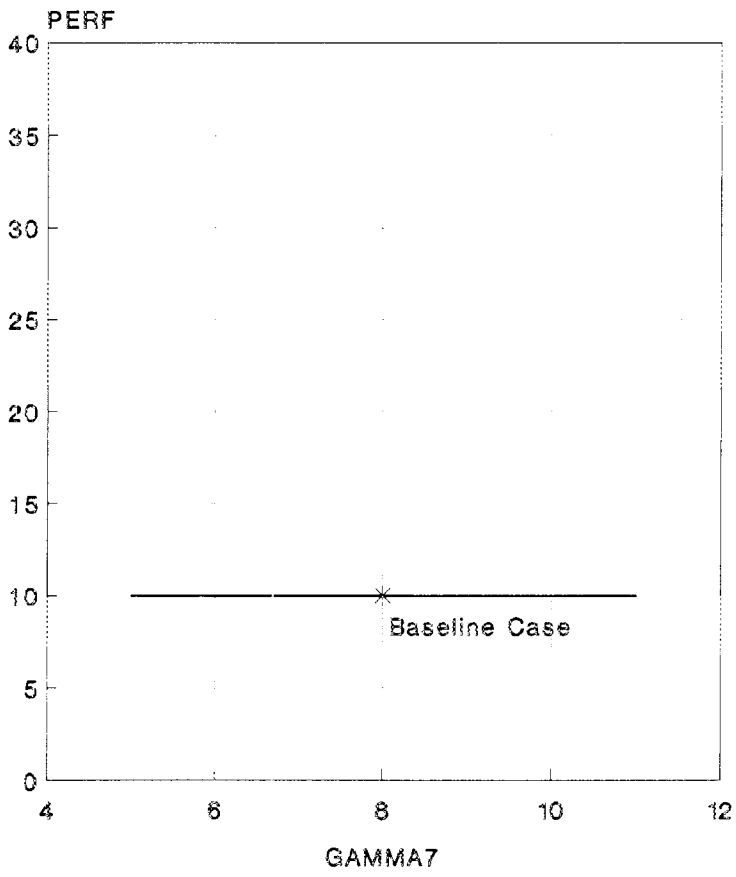
PERF vs. EXP6



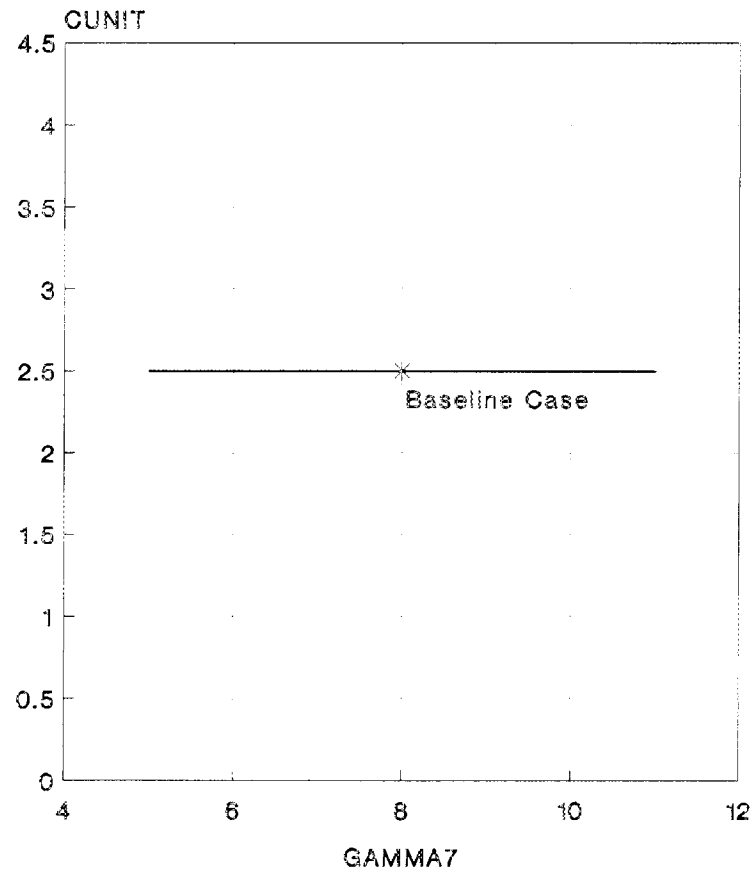
CUNIT vs. EXP6



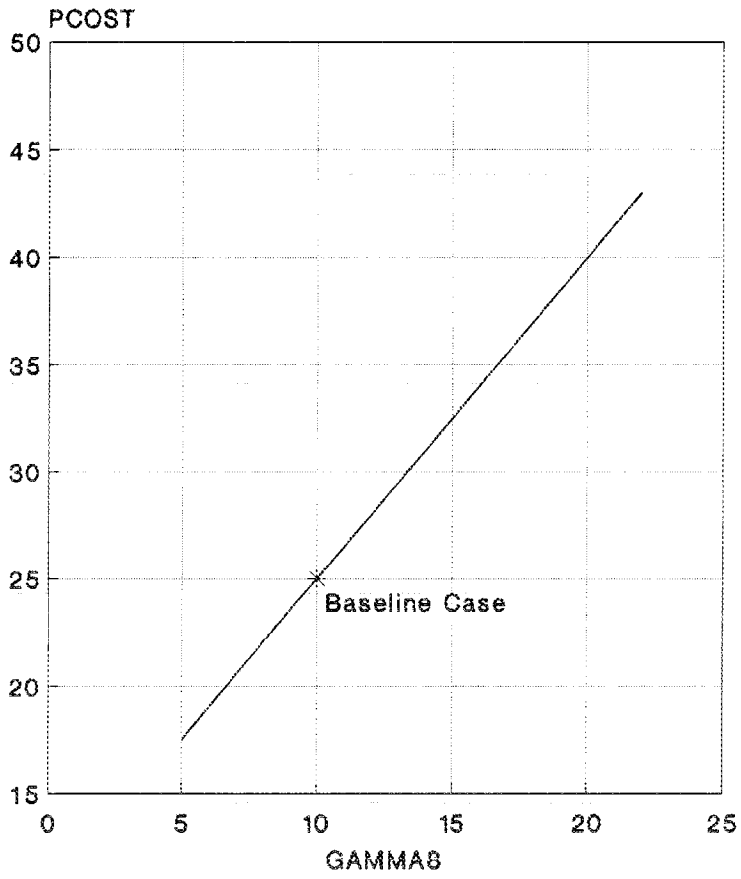
PERF vs. GAMMA7



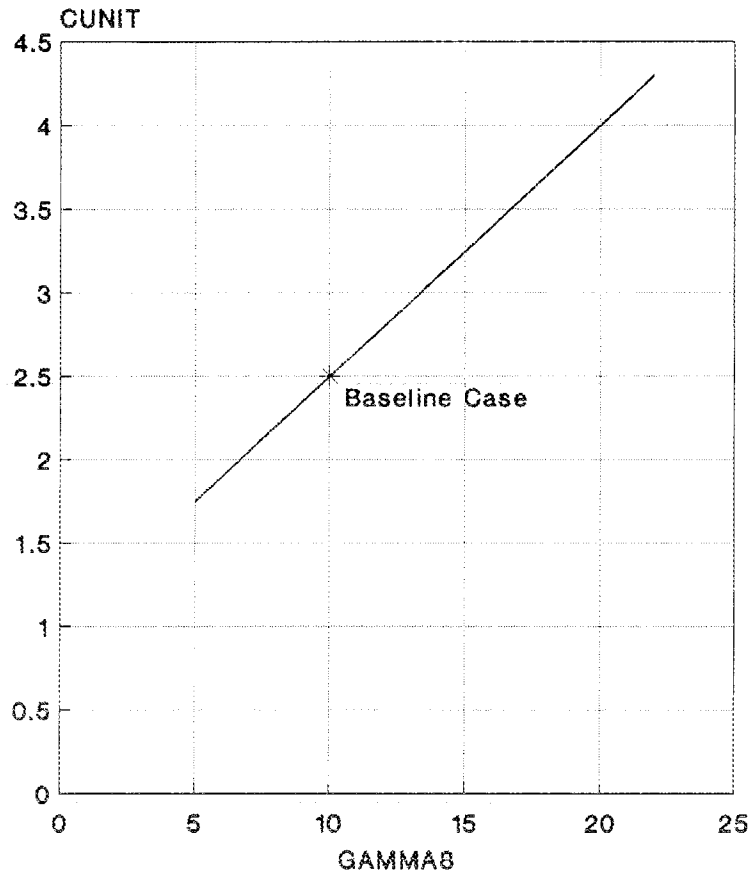
CUNIT vs. GAMMA7



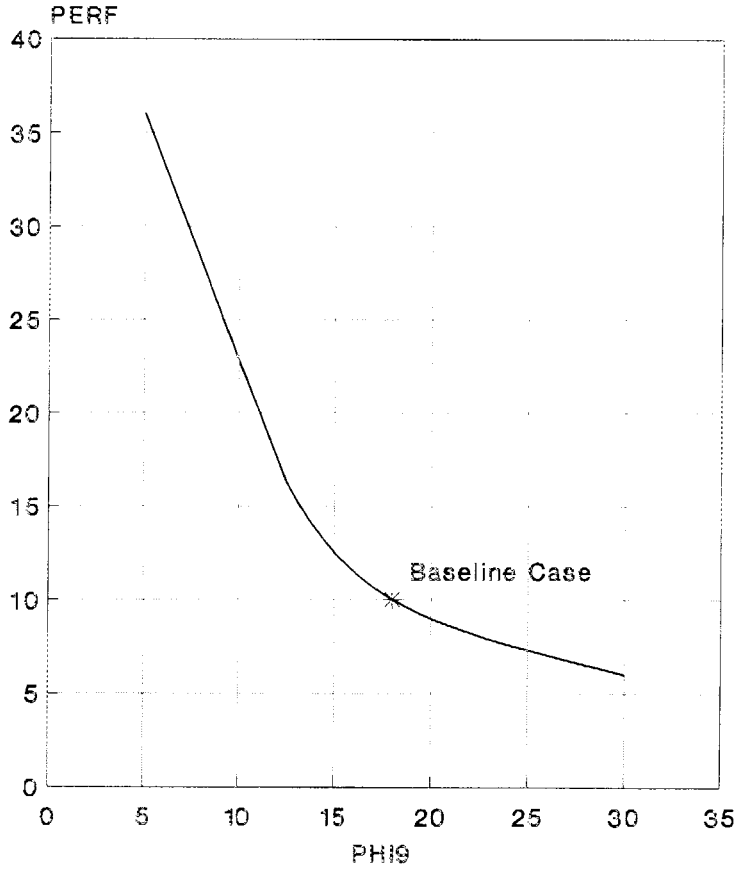
PCOST vs. GAMMA8



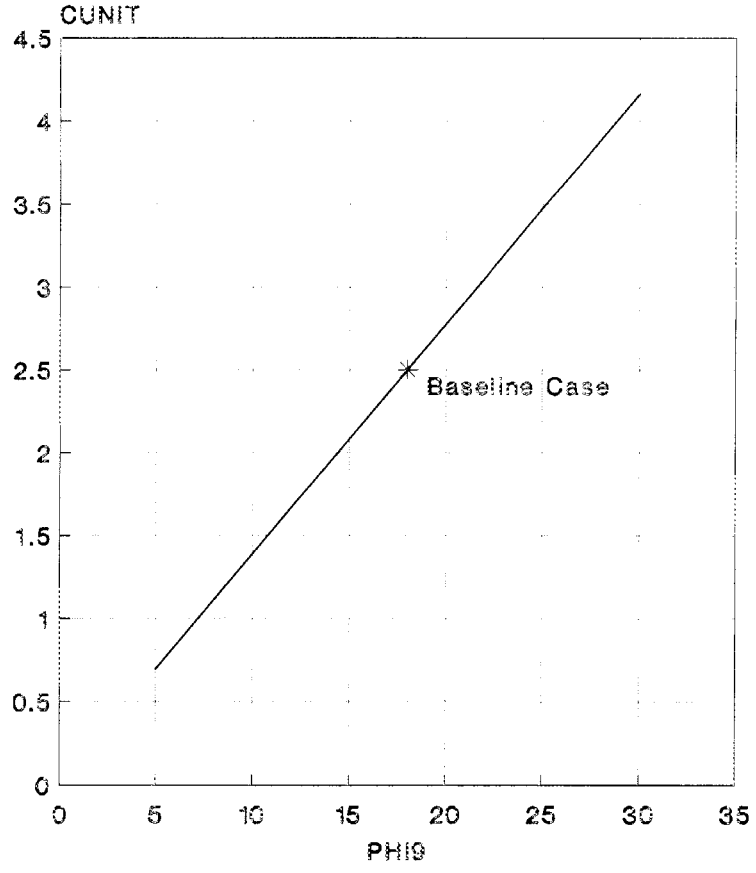
CUNIT vs. GAMMA8



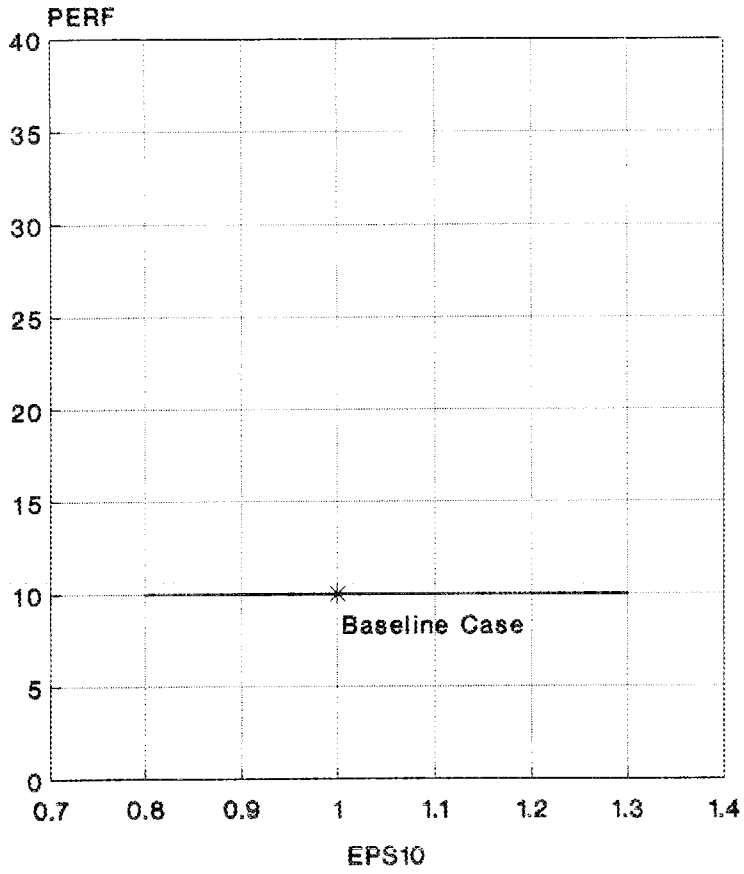
PERF vs. PHI9



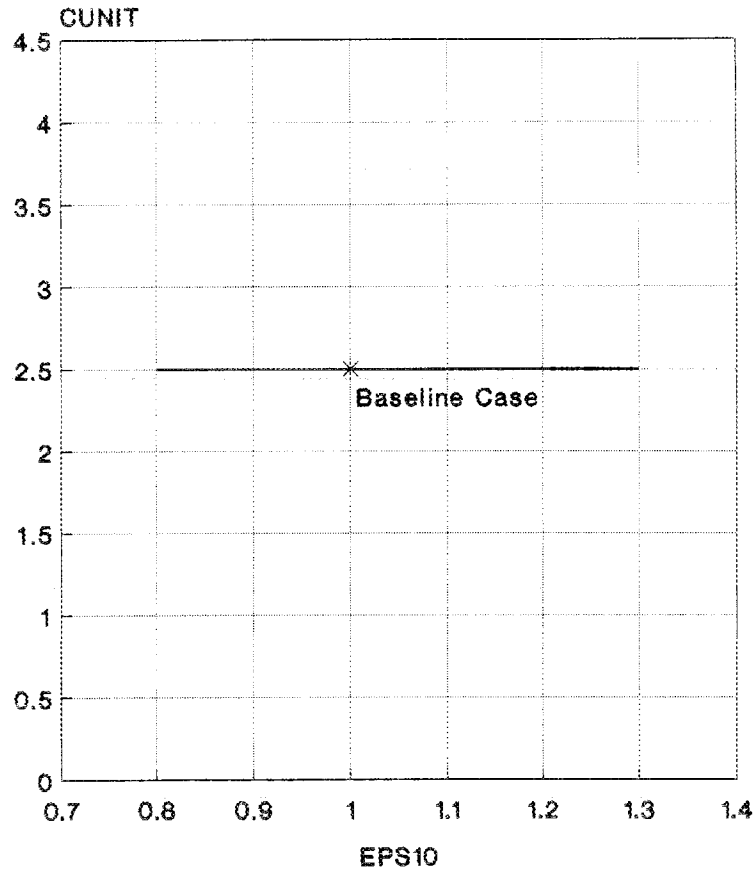
CUNIT vs. PHI9



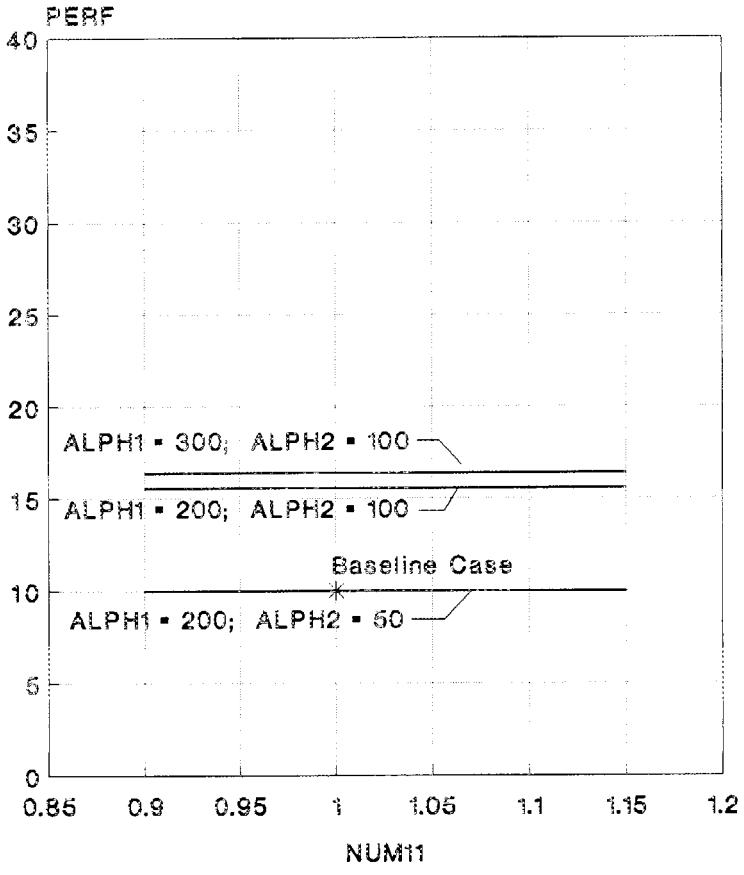
PERF vs. EPS10



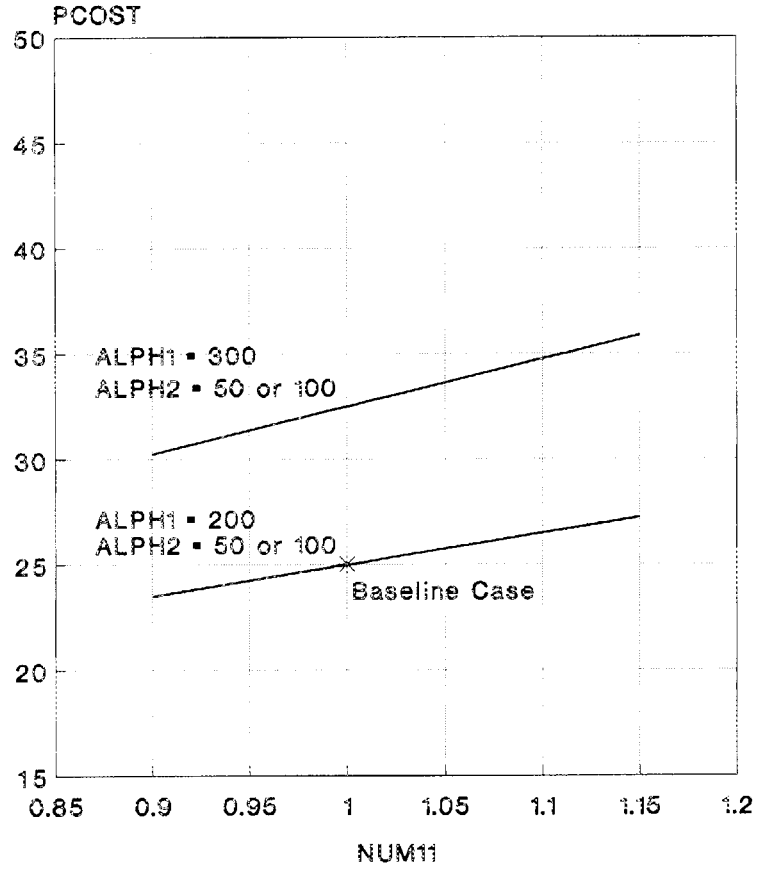
CUNIT vs. EPS10



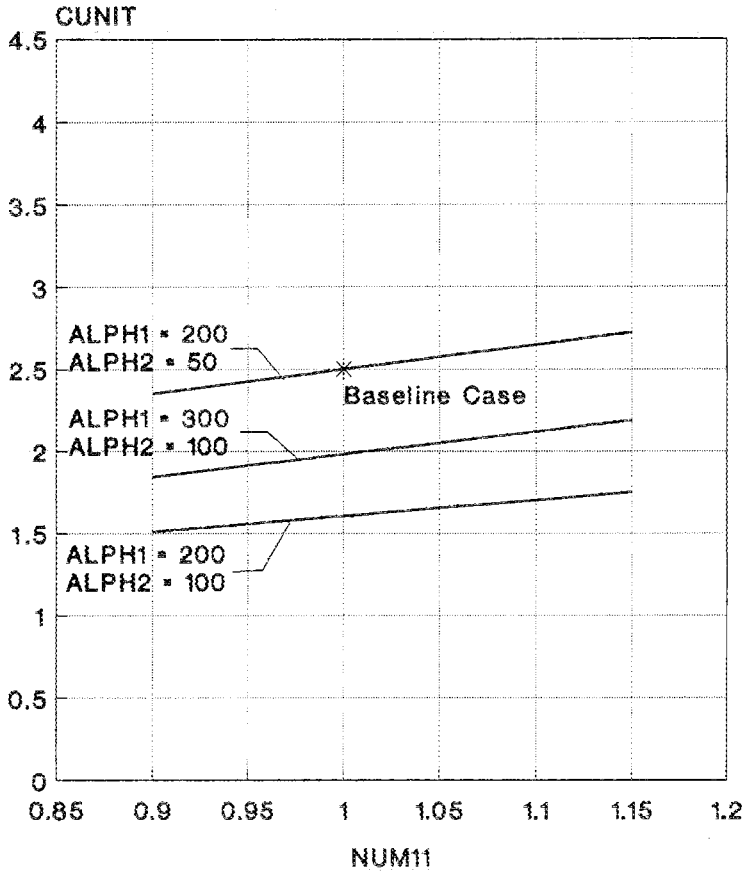
PERF vs. NUM11



PCOST vs. NUM11



CUNIT vs. NUM11



Data for and Output of ORMONTE
for
Elasticity Analysis Option

Input Data for Elasticity Option

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [ELAS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]
```

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
ALPH1     200.
ALPH2     50.
BETA3     10.
BETA4     1.
ZETA5     1.
EXP6      1.
GAMMA7    8.
GAMMA8    10.
PHI9      18.
EPS10     1.
NUM11     1.
PERF      -1.
PCOST     -1.
CUNIT     -1.
```

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT
C1      1.5
C2      2.0
C3      50.
```

All ELAST Output Is on Unit IPRNT

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]    INOPT [ 4]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [ELAS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]


READIN OF SENS. OPTION DATA ON UNIT  4 COMPLETED
```

Output of RDOPT

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000

READIN OF /MASTER/ INITIALIZATION DATA
ON UNIT  8 COMPLETED
```

Output of RDMAST

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT

 C1                    1.500
 C2                    2.000
 C3                   50.000
MODEL NON-SENS. DATA READIN COMPLETED ON UNIT  5
```

Output of RDMODL

```
VALUES IN /MASTER/ FOR ITERATION NO     0


TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000    0
```

Output of WRMAST
prior to call to MODEL

```
PRINTOUT OF USERS MODEL FOR ITERATION #    0

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   200.000        ALPH2=    50.000        BETA3=    10.000
        BETA4=     1.000        ZETA5=     1.000        EXP6 =     1.000
        GAMMA7=    8.000        GAMMA8=   10.000        PHI9 =    18.000
        EPS10=     1.000        NUM11=     1.000


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    10.007        PCOST=    25.000        CUNIT=     2.498
```

Output of MODEL (Base Case)

```
VALUES IN /MASTER/ FOR ITERATION NO     0


TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          10.007
 13   PCOST         25.000
 14   CUNIT          2.498

OPTION = ELAS
```

Output of WRMAST
after call to MODEL
(baseline values now stored in Y-array)

```
TITLE: SAMPLE ELASTICITY ANALYSIS DATA DSN=ELAST.DAT
# OF INPUTS: [ 11]    # OF FIGS OF MERIT: [ 3]

/MASTER/ Y( ) INDICES FOR FIGS OF MERIT:12  13  14   0   0
ELASTICITY ANALYSIS DATA READIN COMPLETED ON UNIT 12

   0    0
```

RESULTS:
Output of ELAST for three of the eleven input variables

PERF vs. ALPH1

```
***************   ALPH1    ************************************

SENSITIVITY VARIABLE NAME: ALPH1     INDEX #:   1                BASE VALUE:     200.000000

FIGURE OF MERIT #1    F.O.M. NAME: PERF     F.O.M. INDEX # 12     BASE VALUE:      10.006945

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:    201.000000    CORRESPONDING F.O.M.:     10.015278

LOW END SENS VARIABLE:  199.000000       CORRESPONDING F.O.M.:      9.998611

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     1.000000    CORRESP. DELTA F.O.M.:     0.008333

LOW END DELTA SENS.VAR.:     1.000000    CORRESP. DELTA F.O.M. :    0.008333

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00083274

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00083274

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00083274

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.166548

HIGH END SLOPE:     0.008333    LOW END SLOPE:     0.008333

AVERAGE SLOPE:      0.008333

*************************************************************
```

PCOST vs. ALPH1

```
**************   ALPH1   ************************************

SENSITIVITY VARIABLE NAME: ALPH1     INDEX #:   1                BASE VALUE:     200.000000

FIGURE OF MERIT #2    F.O.M. NAME: PCOST    F.O.M. INDEX # 13     BASE VALUE:      25.000000

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:    201.000000    CORRESPONDING F.O.M.:     25.075001

LOW END SENS VARIABLE:  199.000000       CORRESPONDING F.O.M.:     24.924999

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     1.000000    CORRESP. DELTA F.O.M.:     0.075001

LOW END DELTA SENS.VAR.:     1.000000    CORRESP. DELTA F.O.M. :    0.075001

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00300003

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00300003

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00300003

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.600006

HIGH END SLOPE:     0.075001    LOW END SLOPE:     0.075001

AVERAGE SLOPE:     0.075001

*****************************************************************
```

CUNIT vs. ALPH1

```
**************   ALPH1   ****************************************

SENSITIVITY VARIABLE NAME: ALPH1     INDEX #:   1                BASE VALUE:     200.000000

FIGURE OF MERIT #3    F.O.M. NAME: CUNIT    F.O.M. INDEX # 14     BASE VALUE:       2.498265

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:    201.000000    CORRESPONDING F.O.M.:      2.503675

LOW END SENS VARIABLE:  199.000000       CORRESPONDING F.O.M.:      2.492846

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     1.000000    CORRESP. DELTA F.O.M.:     0.005410

LOW END DELTA SENS.VAR.:     1.000000    CORRESP. DELTA F.O.M. :    0.005419

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00216549

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00216902

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00216725

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.433450

HIGH END SLOPE:     0.005410    LOW END SLOPE:     0.005419

AVERAGE SLOPE:      0.005414

*************************************************************

   0    0
```

PERF vs. ALPH2

```
**************   ALPH2   ************************************

SENSITIVITY VARIABLE NAME: ALPH2     INDEX #:   2                BASE VALUE:      50.000000

FIGURE OF MERIT #1    F.O.M. NAME: PERF     F.O.M. INDEX # 12     BASE VALUE:      10.006945

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     50.250000    CORRESPONDING F.O.M.:     10.034722

LOW END SENS VARIABLE:   49.750000       CORRESPONDING F.O.M.:      9.979167

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.250000    CORRESP. DELTA F.O.M.:     0.027778

LOW END DELTA SENS.VAR.:     0.250000    CORRESP. DELTA F.O.M. :    0.027778

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00277584

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00277584

AVERAGE FRACT DELTA SENS.VAR.      0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00277584

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.555168

HIGH END SLOPE:     0.111111    LOW END SLOPE:     0.111111

AVERAGE SLOPE:     0.111111

*************************************************************
```

PCOST vs. ALPH2

```
***************   ALPH2    ************************************

SENSITIVITY VARIABLE NAME: ALPH2     INDEX #:   2                BASE VALUE:      50.000000

FIGURE OF MERIT #2    F.O.M. NAME: PCOST    F.O.M. INDEX # 13     BASE VALUE:      25.000000

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     50.250000    CORRESPONDING F.O.M.:     25.000000

LOW END SENS VARIABLE:   49.750000       CORRESPONDING F.O.M.:     25.000000

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.250000    CORRESP. DELTA F.O.M.:     0.000000

LOW END DELTA SENS.VAR.:     0.250000    CORRESP. DELTA F.O.M. :    0.000000

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00000000

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00000000

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00000000

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.000000

HIGH END SLOPE:     0.000000    LOW END SLOPE:     0.000000

AVERAGE SLOPE:     0.000000

*****************************************************************
```

CUNIT vs. ALPH2

```
***************   ALPH2    ****************************************

SENSITIVITY VARIABLE NAME: ALPH2     INDEX #:   2                  BASE VALUE:      50.000000

FIGURE OF MERIT #3    F.O.M. NAME: CUNIT    F.O.M. INDEX # 14      BASE VALUE:       2.498265

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     50.250000    CORRESPONDING F.O.M.:      2.491349

LOW END SENS VARIABLE:   49.750000       CORRESPONDING F.O.M.:      2.505219

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.250000    CORRESP. DELTA F.O.M.:    -0.006916

LOW END DELTA SENS.VAR.:     0.250000    CORRESP. DELTA F.O.M. :   -0.006954

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.: -0.00276815

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.: -0.00278361

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  -0.00277588

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:    -0.555176

HIGH END SLOPE:    -0.027662    LOW END SLOPE:    -0.027817

AVERAGE SLOPE:     -0.027740

*****************************************************************

   0    0
```

PERF vs. BETA3

```
***************   BETA3    ************************************

SENSITIVITY VARIABLE NAME: BETA3     INDEX #:   3                BASE VALUE:      10.000000

FIGURE OF MERIT #1    F.O.M. NAME: PERF     F.O.M. INDEX # 12     BASE VALUE:      10.006945

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     10.050000    CORRESPONDING F.O.M.:      9.998652

LOW END SENS VARIABLE:    9.950000       CORRESPONDING F.O.M.:     10.015320

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.050000    CORRESP. DELTA F.O.M.:    -0.008292

LOW END DELTA SENS.VAR.:     0.050000    CORRESP. DELTA F.O.M. :   -0.008375

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.: -0.00082864

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.: -0.00083694

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  -0.00083279

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:    -0.166557

HIGH END SLOPE:    -0.165843    LOW END SLOPE:    -0.167503

AVERAGE SLOPE:     -0.166673

*************************************************************
```

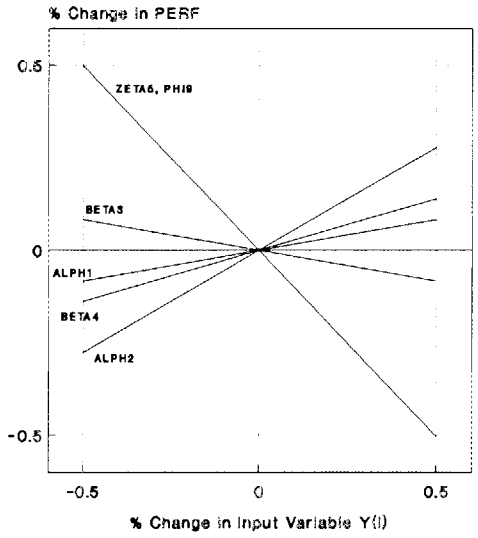
PCOST vs. BETA3

```
***************   BETA3   ****************************************

SENSITIVITY VARIABLE NAME: BETA3     INDEX #:  3                BASE VALUE:      10.000000

FIGURE OF MERIT #2    F.O.M. NAME: PCOST    F.O.M. INDEX # 13     BASE VALUE:      25.000000

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     10.050000    CORRESPONDING F.O.M.:     25.000000

LOW END SENS VARIABLE:    9.950000       CORRESPONDING F.O.M.:     25.000000

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.050000    CORRESP. DELTA F.O.M.:     0.000000

LOW END DELTA SENS.VAR.:     0.050000    CORRESP. DELTA F.O.M. :    0.000000

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00000000

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00000000

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00000000

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.000000

HIGH END SLOPE:     0.000000    LOW END SLOPE:     0.000000

AVERAGE SLOPE:     0.000000

*****************************************************************
```

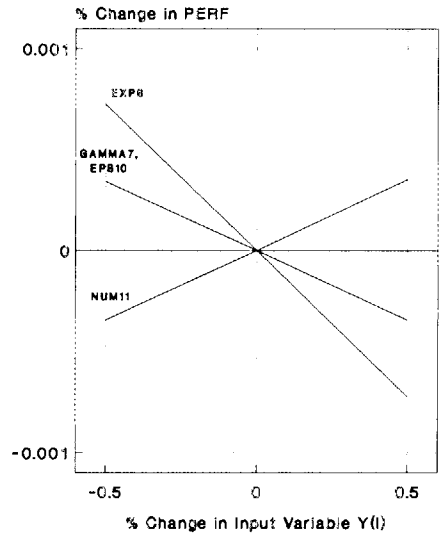
CUNIT vs. BETA3

```
***************   BETA3    ************************************

SENSITIVITY VARIABLE NAME: BETA3     INDEX #:   3                BASE VALUE:      10.000000

FIGURE OF MERIT #3    F.O.M. NAME: CUNIT    F.O.M. INDEX # 14     BASE VALUE:      2.498265

*** ALTERED INPUT AND ALTERED F-O-M VALUES: ***

HIGH END SENS VARIABLE:     10.050000    CORRESPONDING F.O.M.:      2.500337

LOW END SENS VARIABLE:    9.950000       CORRESPONDING F.O.M.:      2.496176

*** DEVIATIONS OF INPUT VARIABLE AND F-O-M FROM BASELINE VALUES ***

HIGH END DELTA SENS.VAR:     0.050000    CORRESP. DELTA F.O.M.:     0.002072

LOW END DELTA SENS.VAR.:     0.050000    CORRESP. DELTA F.O.M. :    0.002089

*** FRACTIONAL DEVIATIONS OF INPUT VAR. AND F-O-M FROM BASELINE VALUES ***

HIGH END FRACT DELTA SENS VAR:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00082932

LOW END FRACT DELTA SENS VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:  0.00083619

AVERAGE FRACT DELTA SENS.VAR.:     0.005000    CORRESPONDING FRACT DELTA F.O.M.:   0.00083275

*** ELASTICITY & SLOPE ***

AVERAGE ELASTICITY:     0.166550

HIGH END SLOPE:     0.041437    LOW END SLOPE:     0.041780

AVERAGE SLOPE:     0.041609

*************************************************************

    0    0
```

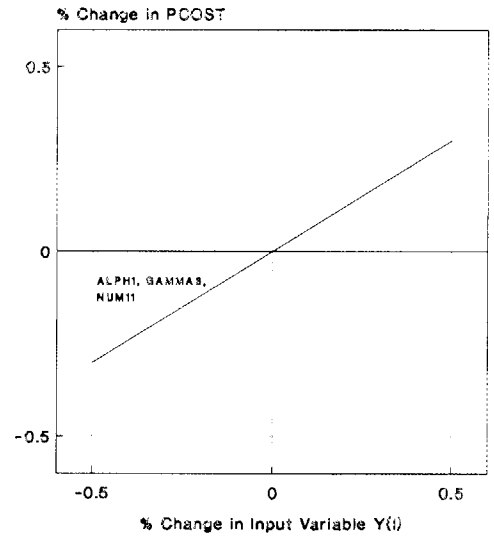
High Leverage Variables for Performance



Low Leverage Variables for Performance

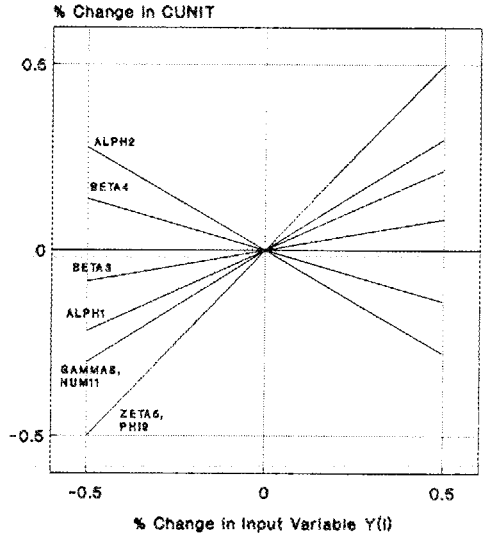


Variables Affecting Production Cost

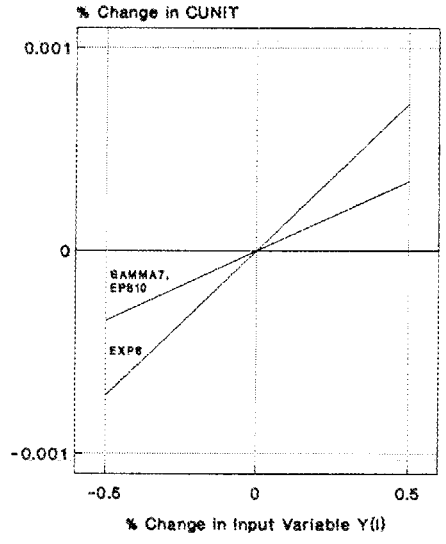


Plotted Results of Elasticity Analysis

High Leverage Variables Affecting Unit Cost



Low Leverage Variables Affecting Unit Cost



Data for and Output of ORMONTE
for
Multivariable Sensitivity Analysis

(MVSS or Probabilistic Monte Carlo Option)

Input Data for MVSS Option

Unit 4: Options

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [MVSS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]
```

Unit 8: Base Case Inputs (Sensitivity Variables)

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
ALPH1     200.
ALPH2     50.
BETA3     10.
BETA4     1.
ZETA5     1.
EXP6      1.
GAMMA7    8.
GAMMA8    10.
PHI9      18.
EPS10     1.
NUM11     1.
PERF      -1.
PCOST     -1.
CUNIT     -1.
```

Unit 5: Inputs to User's Model (Non-sensitivity)

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT
C1      1.5
C2      2.0
C3      50.
```

Unit 9: Distribution Data

```
TITLE: 14 VARIABLE MVSS TEST CASE
NITER [ 5000]   IPR [   5]  INTER [   1]
DISTRIBUTION DATA:
INDEX  XNAME  JTYPE     X1          X2          X3          X4
    V      V      V           V           V           V
   1 ALPH1      -1      200.        100.        500.
   2 ALPH2       0      50.
   3 BETA3      -9      6.          12.
   4 BETA4      -2      1.          .08
   5 ZETA5      -3      1.          .08         1.2
   6 EXP6       -4      1.          0.5         2.
   7 GAMMA7     -6      5.          7.          10.     11.
   8 GAMMA8     -5      10.         0.20
   9 PHI9        5
  5.     5.        .1        .2        .4        .2        .1
  10 EPS10      -7      1.          .08         .8
  11 NUM11      -8      1.          .08         .9      1.15
  12 PERF      -99
  13 PCOST     -99
  14 CUNIT     -99
```

Unit 6 (IPRNT) Output

```
TEST CASE: DSN=OPTIO.DAT  14 VAR. TEST CASE
INPUT UNITS :INMAST [ 8]    INMOD [ 5]    INOPT [ 4]
             ISVSS  [11]    IMVSS [ 9]    IELAS [12]
OUTPUT UNITS:IPRNT  [ 6]    IPLOT [15]    IBUG  [14]
OPTION: [MVSS]  ( DETO,SVSS,MVSS,OR ELAS)
TOTAL NO. OF INPUT + OUTPUT VARIABLES: [ 14]


READIN OF SENS. OPTION DATA ON UNIT  4 COMPLETED
```

Output of RDOPT

```
TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000

READIN OF /MASTER/ INITIALIZATION DATA
ON UNIT  8 COMPLETED
```

Output of RDMAST

```
TITLE: CONSTANTS C1,C2 & C3 FOR 14 VAR. EXAMPLE  DSN=MODL.DAT

 C1                    1.500
 C2                    2.000
 C3                   50.000
MODEL NON-SENS. DATA READIN COMPLETED ON UNIT  5
```

Output of RDMODL

```
VALUES IN /MASTER/ FOR ITERATION NO     0


TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          -1.000
 13   PCOST         -1.000
 14   CUNIT         -1.000    0
```

Output of WRMAST
(before call to MODEL)

```
PRINTOUT OF USERS MODEL FOR ITERATION #    0

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000

INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   200.000        ALPH2=    50.000        BETA3=    10.000
        BETA4=     1.000        ZETA5=     1.000        EXP6 =     1.000
        GAMMA7=    8.000        GAMMA8=   10.000        PHI9 =    18.000
        EPS10=     1.000        NUM11=     1.000

OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    10.007        PCOST=    25.000        CUNIT=     2.498
```

Output of MODEL
(Base Case)

```
VALUES IN /MASTER/ FOR ITERATION NO      0


TITLE: 14 VARIABLE TEST CASE: DSN=MASINI.DAT
  1   ALPH1        200.000
  2   ALPH2         50.000
  3   BETA3         10.000
  4   BETA4          1.000
  5   ZETA5          1.000
  6   EXP6           1.000
  7   GAMMA7         8.000
  8   GAMMA8        10.000
  9   PHI9          18.000
 10   EPS10          1.000
 11   NUM11          1.000
 12   PERF          10.007
 13   PCOST         25.000
 14   CUNIT          2.498

OPTION = MVSS
```

Output of WRMAST
(Base Case inputs and outputs stored in Y-array)

```
TITLE: 14 VARIABLE MVSS TEST CASE
NITER [  5000]   IPR [   5]  INTER [   1]
DISTRIBUTION DATA:
INDEX  XNAME  JTYPE     X1           X2           X3           X4
     V      V       V            V            V            V
    1 ALPH1       -1   2.0000E+02   1.0000E+02   5.0000E+02   0.0000E-01
    2 ALPH2        0   5.0000E+01   0.0000E-01   0.0000E-01   0.0000E-01
    3 BETA3       -9   6.0000E+00   1.2000E+01   0.0000E-01   0.0000E-01
    4 BETA4       -2   1.0000E+00   8.0000E-02   0.0000E-01   0.0000E-01
    5 ZETA5       -3   1.0000E+00   8.0000E-02   1.2000E+00   0.0000E-01
    6 EXP6        -4   1.0000E+00   5.0000E-01   2.0000E+00   0.0000E-01
    7 GAMMA7      -6   5.0000E+00   7.0000E+00   1.0000E+01   1.1000E+01
    8 GAMMA8      -5   1.0000E+01   2.0000E-01   0.0000E-01   0.0000E-01
    9 PHI9         5 BOX HISTOGRAM
LEFT VAL**BOX WIDTH**REST OF ENTRIES ARE REL FREQ
FOR EACH BOX (FRACTIONAL) AND MUST SUM TO 1.0
 5.000E+00 5.000E+00 1.000E-01 2.000E-01 4.000E-01 2.000E-01 1.000E-01
   10 EPS10       -7   1.0000E+00   8.0000E-02   8.0000E-01   0.0000E-01
   11 NUM11       -8   1.0000E+00   8.0000E-02   9.0000E-01   1.1500E+00
   12 PERF       -99   0.0000E-01   0.0000E-01   0.0000E-01   0.0000E-01
   13 PCOST      -99   0.0000E-01   0.0000E-01   0.0000E-01   0.0000E-01
   14 CUNIT      -99   0.0000E-01   0.0000E-01   0.0000E-01   0.0000E-01

PARAMETERS FOR /MASTER/ VARIABLE PROBABILITY
DISTRIBUTIONS SUCCESSFULLY READ IN ON UNIT  9
```

Output of RDMVSS

MONTE CARLO UNCERTAINTY ANALYSIS:
TITLE: 14 VARIABLE MVSS TEST CASE

11 INPUT VARIABLES 3 OUTPUT VARIABLES 5000 ITERATIONS 3 OUTPUT HISTS. TO BE PRINTED
34 STORAGE LOCATIONS USED FOR INPUT DATA

SUMMARY OF INPUT DATA:

INDEX	NAME	DIST	DATA						
1	ALPH1	TRIANGLE	200.0	100.0	500.0				
2	ALPH2	CONSTANT	50.00						
3	BETA3	UNIFORM	6.000	12.00					
4	BETA4	NORMAL	1.000	0.8000E-01					
5	ZETA5	NORM-MAX	1.000	0.8000E-01	1.200				
6	EXP6	LOG-TRI	1.000	0.5000	2.000				
7	GAMMA7	TRAPZOID	5.000	7.000	10.00	11.00			
8	GAMMA8	LOG-NORM	10.00	0.2000					
9	PHI9	HISTOGRM	5.000	5.000	0.1000	0.3000	0.7000	0.9000	1.000
10	EPS10	NORM-MIN	1.000	0.8000E-01	0.8000				
11	NUM11	NORM-LIM	1.000	0.8000E-01	0.9000	1.150			

Output of PRINTX

```
PRINTOUT OF USERS MODEL FOR ITERATION #    1

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   191.940        ALPH2=    50.000        BETA3=     6.339
        BETA4=     0.947        ZETA5=     1.026        EXP6 =     1.328
        GAMMA7=    6.156        GAMMA8=   12.054        PHI9 =    18.747
        EPS10=     0.962        NUM11=     1.018


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    10.025        PCOST=    27.668        CUNIT=     2.760
```

Output of MODEL
(first non-baseline case or
first iteration)
first of five to be printed

```
PRINTOUT OF USERS MODEL FOR ITERATION #    2

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   348.095        ALPH2=    50.000        BETA3=     8.176
        BETA4=     0.988        ZETA5=     0.819        EXP6 =     1.178
        GAMMA7=    8.466        GAMMA8=    8.483        PHI9 =    18.845
        EPS10=     1.096        NUM11=     0.936


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    13.829        PCOST=    30.719        CUMIT=     2.221
```

Output of MODEL
second of five to be printed

```
PRINTOUT OF USERS MODEL FOR ITERATION #    3

CONSTANTS (NON-SENSITIVITY):

          C1=     1.500
          C2=     2.000
          C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

          ALPH1=   232.978          ALPH2=    50.000          BETA3=     8.923
          BETA4=     1.125          ZETA5=     1.032          EXP6 =     1.090
          GAMMA7=     7.350         GAMMA8=    10.174          PHI9 =    19.902
          EPS10=     1.085          NUM11=     0.961


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

          PERF=     9.519          PCOST=    27.078          CUNIT=     2.844
```

Output of MODEL
third of five to be printed

```
PRINTOUT OF USERS MODEL FOR ITERATION #    4

CONSTANTS (NON-SENSITIVITY):

        C1=     1.500
        C2=     2.000
        C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

        ALPH1=   175.079        ALPH2=    50.000        BETA3=     8.039
        BETA4=     1.053        ZETA5=     1.065        EXP6 =     0.712
        GAMMA7=    9.660        GAMMA8=    7.563        PHI9 =    17.038
        EPS10=     0.950        NUM11=     0.990


OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

        PERF=    10.220        PCOST=    19.831        CUNIT=     1.940
```

Output of MODEL
fourth of five to be printed

```
PRINTOUT OF USERS MODEL FOR ITERATION #    5

CONSTANTS (NON-SENSITIVITY):

          C1=     1.500
          C2=     2.000
          C3=    50.000


INPUT VARIABLES (UNCERTAIN SENS. VARS.):

          ALPH1=   271.256          ALPH2=    50.000          BETA3=     9.886
          BETA4=     0.905          ZETA5=     0.993          EXP6 =     0.704
          GAMMA7=    6.190          GAMMA8=   10.331          PHI9 =    25.575
          EPS10=     0.995          NUM11=     0.921


 OUTPUT VARIABLES (UNCERTAIN FIGURES-OF-MERIT):

          PERF=     7.351          PCOST=    29.350          CUNIT=     3.992

  100  200  300  400  500  600  700  800  900 1000 1100 1200 1300 1400 1500 1600 1700 1800
 1900 2000 2100 2200 2300 2400 2500 2600 2700 2800 2900 3000 3100 3200 3300 3400 3500 *
```

Output of MODEL
fifth of five to be printed

Values of ITER as Monte Carlo proceeds
(Output of MODEL)
Iterations 6 - 5000:
this will appear on screen

*Continues through iteration 5000 (3600 - 5000 not shown here).

Final Output of MVSSOP

(Histograms)

(prints one histogram per page)

STATISTICS FOR INPUT VARIABLE ALPH1 (1) JTYPE= -1 SAMPL.DIST TYPE: TRIANGLE

MINIMUM 104.09 MAXIMUM 493.95

MEAN 266.98 VARIANCE 7256.7 STD.DEVIATION 85.186

PERCENTILES: 2.5% 132. 5.0% 145. 50.0% 255. 95.0% 423. 97.5% 446.

5000 POINT HISTOGRAM * = 4POINTS

PROB	CUME	VALUE		
0.0016	0.0000	100.0	8	**
0.0068	0.0016	110.0	34	*********
0.0124	0.0084	120.0	62	**********
0.0188	0.0208	130.0	94	**********
0.0210	0.0396	140.0	105	**********
0.0274	0.0606	150.0	137	**********
0.0314	0.0880	160.0	157	**********
0.0388	0.1194	170.0	194	**********
0.0426	0.1582	180.0	213	**********
0.0460	0.2008	190.0	230	**********
0.0496	0.2468	200.0	248	**********
0.0562	0.2964	210.0	281	**********
0.0436	0.3526	220.0	218	**********
0.0430	0.3962	230.0	215	**********
0.0416	0.4392	240.0	208	**********
0.0422	0.4808	250.0	211	**********
0.0386	0.5230	260.0	193	**********
0.0398	0.5616	270.0	199	**********
0.0352	0.6014	280.0	176	**********
0.0290	0.6366	290.0	145	**********
0.0322	0.6656	300.0	161	**********
0.0334	0.6978	310.0	167	**********
0.0294	0.7312	320.0	147	**********
0.0216	0.7606	330.0	108	**********
0.0220	0.7822	340.0	110	**********
0.0242	0.8042	350.0	121	**********
0.0256	0.8284	360.0	128	**********
0.0260	0.8540	370.0	130	**********
0.0184	0.8800	380.0	92	**********
0.0168	0.8984	390.0	84	**********
0.0136	0.9152	400.0	68	**********
0.0170	0.9288	410.0	85	**********
0.0116	0.9458	420.0	58	**********
0.0112	0.9574	430.0	56	**********
0.0108	0.9686	440.0	54	**********
0.0066	0.9794	450.0	33	********
0.0060	0.9860	460.0	30	********
0.0048	0.9920	470.0	24	******
0.0028	0.9968	480.0	14	****
0.0004	0.9996	490.0	2	*
	1.0000	500.0		

STATISTICS FOR INPUT VARIABLE BETA3 (3) JTYPE= -9 SAMPL.DIST TYPE: UNIFORM

MINIMUM 6.0002 MAXIMUM 11.996

MEAN 9.0309 VARIANCE 3.0376 STD.DEVIATION 1.7429

PERCENTILES: 2.5% 6.16 5.0% 6.30 50.0% 9.00 95.0% 11.7 97.5% 11.9

5000 POINT HISTOGRAM • = 3POINTS

PROB	CUME	VALUE		
0.0310	0.0000	6.000	155	**********
0.0380	0.0310	6.200	190	**********
0.0340	0.0690	6.400	170	**********
0.0320	0.1030	6.600	160	**********
0.0290	0.1350	6.800	145	**********
0.0300	0.1640	7.000	150	**********
0.0298	0.1940	7.200	149	**********
0.0366	0.2238	7.400	183	**********
0.0338	0.2604	7.600	169	**********
0.0328	0.2942	7.800	164	**********
0.0336	0.3270	8.000	168	**********
0.0336	0.3606	8.200	168	**********
0.0328	0.3942	8.400	164	**********
0.0372	0.4270	8.600	186	**********
0.0356	0.4642	8.800	178	**********
0.0310	0.4998	9.000	155	**********
0.0330	0.5308	9.200	165	**********
0.0302	0.5638	9.400	151	**********
0.0320	0.5940	9.600	160	**********
0.0324	0.6260	9.800	162	**********
0.0324	0.6584	10.00	162	**********
0.0306	0.6908	10.20	153	**********
0.0340	0.7214	10.40	170	**********
0.0304	0.7554	10.60	152	**********
0.0380	0.7858	10.80	190	**********
0.0336	0.8238	11.00	168	**********
0.0398	0.8574	11.20	199	**********
0.0332	0.8972	11.40	166	**********
0.0324	0.9304	11.60	162	**********
0.0372	0.9628	11.80	186	**********
	1.0000	12.00		

STATISTICS FOR INPUT VARIABLE BETA4 (4) JTYPE= -2 SAMPL.DIST TYPE: NORMAL

MINIMUM 0.75197 MAXIMUM 1.3048

MEAN 1.0003 VARIANCE 6.32429E-03 STD.DEVIATION 7.95254E-02

PERCENTILES: 2.5%0.845 5.0%0.868 50.0%1.000 95.0% 1.13 97.5% 1.16

5000 POINT HISTOGRAM * = 7POINTS

PROB	CUME	VALUE		
0.0004	0.0000	0.7400	2	
0.0012	0.0004	0.7600	6	*
0.0038	0.0016	0.7800	19	***
0.0050	0.0054	0.8000	25	****
0.0106	0.0104	0.8200	53	********
0.0166	0.0210	0.8400	83	**********
0.0302	0.0376	0.8600	151	**********
0.0366	0.0678	0.8800	183	**********
0.0602	0.1044	0.9000	301	**********
0.0696	0.1646	0.9200	348	**********
0.0756	0.2342	0.9400	378	**********
0.0916	0.3098	0.9600	458	**********
0.1000	0.4014	0.9800	500	**********
0.0904	0.5014	1.000	452	**********
0.0872	0.5918	1.020	436	**********
0.0914	0.6790	1.040	457	**********
0.0722	0.7704	1.060	361	**********
0.0548	0.8426	1.080	274	**********
0.0358	0.8974	1.100	179	**********
0.0264	0.9332	1.120	132	**********
0.0196	0.9596	1.140	98	**********
0.0086	0.9792	1.160	43	******
0.0060	0.9878	1.180	30	*****
0.0032	0.9938	1.200	16	**
0.0020	0.9970	1.220	10	*
0.0004	0.9990	1.240	2	
0.0004	0.9994	1.260	2	
0.0000	0.9998	1.280	0	
0.0002	0.9998	1.300	1	
	1.0000	1.320		

STATISTICS FOR INPUT VARIABLE ZETA5 (5) JTYPE= -3 SAMPL.DIST TYPE: NORM-MAX

MINIMUM 0.75758 MAXIMUM 1.1996

MEAN 0.99851 VARIANCE 6.19945E-03 STD.DEVIATION 7.87366E-02

PERCENTILES: 2.5%0.842 5.0%0.866 50.0%1.000 95.0% 1.13 97.5% 1.15

5000 POINT HISTOGRAM • = 7POINTS

PROB	CUME	VALUE		
0.0002	0.0000	0.7400	1	
0.0016	0.0002	0.7600	8	*
0.0026	0.0018	0.7800	13	**
0.0070	0.0044	0.8000	35	*****
0.0118	0.0114	0.8200	59	********
0.0168	0.0232	0.8400	84	**********
0.0314	0.0400	0.8600	157	**********
0.0390	0.0714	0.8800	195	**********
0.0546	0.1104	0.9000	273	**********
0.0694	0.1650	0.9200	347	**********
0.0814	0.2344	0.9400	407	**********
0.0912	0.3158	0.9600	456	**********
0.0944	0.4070	0.9800	472	**********
0.1010	0.5014	1.000	505	**********
0.0938	0.6024	1.020	469	**********
0.0812	0.6962	1.040	406	**********
0.0626	0.7774	1.060	313	**********
0.0550	0.8400	1.080	275	**********
0.0412	0.8950	1.100	206	**********
0.0282	0.9362	1.120	141	**********
0.0182	0.9644	1.140	91	**********
0.0110	0.9826	1.160	55	********
0.0064	0.9936	1.180	32	*****
	1.0000	1.200		

STATISTICS FOR INPUT VARIABLE EXP6 (6) JTYPE= -4 SAMPL.DIST TYPE: LOG-TRI

MINIMUM 0.50536 MAXIMUM 1.9857

MEAN 1.0410 VARIANCE 8.60484E-02 STD.DEVIATION 0.29334

PERCENTILES: 2.5%0.575 5.0%0.617 50.0% 1.00 95.0% 1.60 97.5% 1.69

5000 POINT HISTOGRAM * = 5POINTS

PROB	CUME	VALUE		
0.0106	0.0000	0.5000	53	**********
0.0288	0.0106	0.5500	144	**********
0.0320	0.0394	0.6000	160	**********
0.0468	0.0714	0.6500	234	**********
0.0486	0.1182	0.7000	243	**********
0.0604	0.1668	0.7500	302	**********
0.0582	0.2272	0.8000	291	**********
0.0680	0.2854	0.8500	340	**********
0.0694	0.3534	0.9000	347	**********
0.0736	0.4228	0.9500	368	**********
0.0688	0.4964	1.000	344	**********
0.0570	0.5652	1.050	285	**********
0.0554	0.6222	1.100	277	**********
0.0502	0.6776	1.150	251	**********
0.0438	0.7278	1.200	219	**********
0.0372	0.7716	1.250	186	**********
0.0298	0.8088	1.300	149	**********
0.0306	0.8386	1.350	153	**********
0.0214	0.8692	1.400	107	**********
0.0246	0.8906	1.450	123	**********
0.0160	0.9152	1.500	80	**********
0.0178	0.9312	1.550	89	**********
0.0168	0.9490	1.600	84	**********
0.0120	0.9658	1.650	60	**********
0.0080	0.9778	1.700	40	********
0.0072	0.9858	1.750	36	*******
0.0030	0.9930	1.800	15	***
0.0018	0.9960	1.850	9	**
0.0016	0.9978	1.900	8	**
0.0006	0.9994	1.950	3	*
	1.0000	2.000		

STATISTICS FOR INPUT VARIABLE GAMMA7 (7) JTYPE= -6 SAMPL.DIST TYPE: TRAPZOID

MINIMUM 5.0423 MAXIMUM 10.962

MEAN 8.1984 VARIANCE 1.8270 STD.DEVIATION 1.3516

PERCENTILES: 2.5% 5.69 5.0% 6.00 50.0% 8.21 95.0% 10.3 97.5% 10.5

5000 POINT HISTOGRAM * = 4POINTS

PROB	CUME	VALUE		
0.0018	0.0000	5.000	9	**
0.0060	0.0018	5.200	30	********
0.0102	0.0078	5.400	51	**********
0.0164	0.0180	5.600	82	**********
0.0154	0.0344	5.800	77	**********
0.0244	0.0498	6.000	122	**********
0.0282	0.0742	6.200	141	**********
0.0340	0.1024	6.400	170	**********
0.0372	0.1364	6.600	186	**********
0.0448	0.1736	6.800	224	**********
0.0474	0.2184	7.000	237	**********
0.0518	0.2658	7.200	259	**********
0.0444	0.3176	7.400	222	**********
0.0482	0.3620	7.600	241	**********
0.0426	0.4102	7.800	213	**********
0.0444	0.4528	8.000	222	**********
0.0484	0.4972	8.200	242	**********
0.0456	0.5456	8.400	228	**********
0.0456	0.5912	8.600	228	**********
0.0440	0.6368	8.800	220	**********
0.0422	0.6808	9.000	211	**********
0.0422	0.7230	9.200	211	**********
0.0460	0.7652	9.400	230	**********
0.0414	0.8112	9.600	207	**********
0.0454	0.8526	9.800	227	**********
0.0410	0.8980	10.00	205	**********
0.0222	0.9390	10.20	111	**********
0.0210	0.9612	10.40	105	**********
0.0122	0.9822	10.60	61	**********
0.0056	0.9944	10.80	28	*******
	1.0000	11.00		

STATISTICS FOR INPUT VARIABLE GAMMAB (8) JTYPE= -5 SAMPL.DIST TYPE: LOG-NORM

MINIMUM 4.8805 MAXIMUM 22.354

MEAN 10.199 VARIANCE 4.1446 STD.DEVIATION 2.0358

PERCENTILES: 2.5% 6.76 5.0% 7.21 50.0% 10.0 95.0% 13.8 97.5% 14.8

5000 POINT HISTOGRAM * = 7POINTS

PROB	CUME	VALUE		
0.0004	0.0000	4.500	2	
0.0018	0.0004	5.000	9	*
0.0028	0.0022	5.500	14	**
0.0090	0.0050	6.000	45	******
0.0212	0.0140	6.500	106	**********
0.0348	0.0352	7.000	174	**********
0.0578	0.0700	7.500	289	**********
0.0788	0.1278	8.000	394	**********
0.0956	0.2066	8.500	478	**********
0.0950	0.3022	9.000	475	**********
0.0952	0.3972	9.500	476	**********
0.1032	0.4924	10.00	516	**********
0.0886	0.5956	10.50	443	**********
0.0778	0.6842	11.00	389	**********
0.0634	0.7620	11.50	317	**********
0.0486	0.8254	12.00	243	**********
0.0338	0.8740	12.50	169	**********
0.0290	0.9078	13.00	145	**********
0.0188	0.9368	13.50	94	**********
0.0120	0.9556	14.00	60	*********
0.0116	0.9676	14.50	58	********
0.0078	0.9792	15.00	39	******
0.0052	0.9870	15.50	26	****
0.0018	0.9922	16.00	9	*
0.0012	0.9940	16.50	6	*
0.0016	0.9952	17.00	8	*
0.0014	0.9968	17.50	7	*
0.0010	0.9982	18.00	5	*
0.0004	0.9992	18.50	2	
0.0000	0.9996	19.00	0	
0.0002	0.9996	19.50	1	
0.0000	0.9998	20.00	0	
0.0000	0.9998	20.50	0	
0.0000	0.9998	21.00	0	
0.0000	0.9998	21.50	0	
0.0002	0.9998	22.00	1	
	1.0000	22.50		

STATISTICS FOR INPUT VARIABLE PHI? (9) JTYPE= 5 SAMPL.DIST TYPE: HISTOGRM

MINIMUM 5.0130 MAXIMUM 29.995

MEAN 17.589 VARIANCE 32.142 STD.DEVIATION 5.6694

PERCENTILES: 2.5% 6.31 5.0% 7.62 50.0% 17.6 95.0% 27.4 97.5% 28.7

5000 POINT HISTOGRAM • = 14POINTS

PROB	CUME	VALUE		
0.0478	0.0000	5.000	239	**********
0.0474	0.0478	7.500	237	**********
0.1052	0.0952	10.00	526	**********
0.0946	0.2004	12.50	473	**********
0.1978	0.2950	15.00	989	**********
0.2010	0.4928	17.50	1005	**********
0.0970	0.6938	20.00	485	**********
0.1044	0.7908	22.50	522	**********
0.0560	0.8952	25.00	280	**********
0.0488	0.9512	27.50	244	**********
	1.0000	30.00		

STATISTICS FOR INPUT VARIABLE EPS10 (10) JTYPE= -7 SAMPL.DIST TYPE: NORM-MIN

MINIMUM 0.80128 MAXIMUM 1.2817

MEAN 1.0022 VARIANCE 6.26762E-03 STD.DEVIATION 7.91683E-02

PERCENTILES: 2.5%0.849 5.0%0.871 50.0% 1.00 95.0% 1.14 97.5% 1.16

5000 POINT HISTOGRAM * = 7POINTS

PROB	CUME	VALUE		
0.0056	0.0000	0.8000	28	****
0.0114	0.0056	0.8200	57	********
0.0170	0.0170	0.8400	85	**********
0.0286	0.0340	0.8600	143	**********
0.0434	0.0626	0.8800	217	**********
0.0516	0.1060	0.9000	258	**********
0.0684	0.1576	0.9200	342	**********
0.0724	0.2260	0.9400	362	**********
0.0994	0.2984	0.9600	497	**********
0.0908	0.3978	0.9800	454	**********
0.0990	0.4886	1.0000	495	**********
0.0940	0.5876	1.020	470	**********
0.0814	0.6816	1.040	407	**********
0.0702	0.7630	1.060	351	**********
0.0550	0.8332	1.080	275	**********
0.0384	0.8882	1.100	192	**********
0.0304	0.9266	1.120	152	**********
0.0204	0.9570	1.140	102	**********
0.0102	0.9774	1.160	51	*******
0.0056	0.9876	1.180	28	****
0.0044	0.9932	1.200	22	***
0.0016	0.9976	1.220	8	*
0.0004	0.9992	1.240	2	
0.0002	0.9996	1.260	1	
0.0002	0.9998	1.280	1	
	1.0000	1.300		

STATISTICS FOR INPUT VARIABLE NUM11 (11) JTYPE= -8 SAMPL.DIST TYPE: NOR-2LIM

MINIMUM 0.90001 MAXIMUM 1.1500

MEAN 1.0111 VARIANCE 3.65252E-03 STD.DEVIATION 6.04361E-02

PERCENTILES: 2.5%0.908 5.0%0.917 50.0% 1.01 95.0% 1.12 97.5% 1.13

5000 POINT HISTOGRAM * = 5POINTS

PROB	CUMP	VALUE		
0.0298	0.0000	0.9000	149	**********
0.0310	0.0298	0.9100	155	**********
0.0384	0.0608	0.9200	192	**********
0.0402	0.0992	0.9300	201	**********
0.0426	0.1394	0.9400	213	**********
0.0476	0.1820	0.9500	238	**********
0.0520	0.2296	0.9600	260	**********
0.0558	0.2816	0.9700	279	**********
0.0602	0.3374	0.9800	301	**********
0.0612	0.3976	0.9900	306	**********
0.0548	0.4588	1.000	274	**********
0.0554	0.5136	1.010	277	**********
0.0526	0.5690	1.020	263	**********
0.0486	0.6216	1.030	243	**********
0.0512	0.6702	1.040	256	**********
0.0448	0.7214	1.050	224	**********
0.0462	0.7662	1.060	231	**********
0.0334	0.8124	1.070	167	**********
0.0360	0.8458	1.080	180	**********
0.0280	0.8818	1.090	140	**********
0.0286	0.9098	1.100	143	**********
0.0208	0.9384	1.110	104	**********
0.0174	0.9592	1.120	87	**********
0.0126	0.9766	1.130	63	**********
0.0108	0.9892	1.140	54	**********
	1.0000	1.150		

STATISTICS FOR OUTPUT VARIABLE PERF (12) JTYPE=-99 SAMPL.DIST TYPE: NOT-APPL

MINIMUM 5.1038 MAXIMUM 50.673

MEAN 12.911 VARIANCE 36.956 STD.DEVIATION 6.0791

PERCENTILES: 2.5% 6.38 5.0% 6.93 50.0% 11.2 95.0% 26.0 97.5% 31.4

5000 POINT HISTOGRAM * = 8POINTS

PROB	CUME	VALUE		
0.0078	0.0000	5.000	39	*****
0.0452	0.0078	6.000	226	**********
0.0898	0.0530	7.000	449	**********
0.1052	0.1428	8.000	526	**********
0.1154	0.2480	9.000	577	**********
0.1176	0.3634	10.00	588	**********
0.1034	0.4810	11.00	517	**********
0.0726	0.5844	12.00	363	**********
0.0610	0.6570	13.00	305	**********
0.0426	0.7180	14.00	213	**********
0.0372	0.7606	15.00	186	**********
0.0374	0.7978	16.00	187	**********
0.0248	0.8352	17.00	124	**********
0.0218	0.8600	18.00	109	**********
0.0160	0.8818	19.00	80	**********
0.0136	0.8978	20.00	68	*********
0.0092	0.9114	21.00	46	******
0.0084	0.9206	22.00	42	*****
0.0064	0.9290	23.00	32	****
0.0076	0.9354	24.00	38	*****
0.0066	0.9430	25.00	33	****
0.0040	0.9496	26.00	20	***
0.0064	0.9536	27.00	32	****
0.0044	0.9600	28.00	22	***
0.0054	0.9644	29.00	27	***
0.0032	0.9698	30.00	16	**
0.0048	0.9730	31.00	24	***
0.0038	0.9778	32.00	19	**
0.0028	0.9816	33.00	14	**
0.0036	0.9844	34.00	18	**
0.0022	0.9880	35.00	11	*
0.0024	0.9902	36.00	12	**
0.0008	0.9926	37.00	4	*
0.0026	0.9934	38.00	13	**
0.0008	0.9960	39.00	4	*
0.0002	0.9968	40.00	1	
0.0010	0.9970	41.00	5	*
0.0006	0.9980	42.00	3	
0.0004	0.9986	43.00	2	
	0.9990	44.00		

STATISTICS FOR OUTPUT VARIABLE PCOST (13) JTYPE=-99 SAMPL.DIST TYPE: MOT-APPL

MINIMUM 15.301 MAXIMUM 81.777

MEAN 30.676 VARIANCE 65.110 STD.DEVIATION 8.0691

PERCENTILES: 2.5% 18.8 5.0% 20.0 50.0% 29.3 95.0% 45.9 97.5% 49.6

5000 POINT HISTOGRAM * = 8POINTS

PROB	CUME	VALUE		
0.0008	0.0000	14.00	4	*
0.0080	0.0008	16.00	40	*****
0.0410	0.0088	18.00	205	**********
0.0722	0.0498	20.00	361	**********
0.0968	0.1220	22.00	484	**********
0.1072	0.2188	24.00	536	**********
0.1088	0.3260	26.00	544	**********
0.1012	0.4348	28.00	506	**********
0.0874	0.5360	30.00	437	**********
0.0754	0.6234	32.00	377	**********
0.0672	0.6988	34.00	336	**********
0.0526	0.7660	36.00	263	**********
0.0466	0.8186	38.00	233	**********
0.0364	0.8652	40.00	182	**********
0.0282	0.9016	42.00	141	**********
0.0214	0.9298	44.00	107	**********
0.0158	0.9512	46.00	79	**********
0.0098	0.9670	48.00	49	******
0.0092	0.9768	50.00	46	******
0.0060	0.9860	52.00	30	****
0.0032	0.9920	54.00	16	**
0.0016	0.9952	56.00	8	*
0.0008	0.9968	58.00	4	*
0.0010	0.9976	60.00	5	*
0.0006	0.9986	62.00	3	
0.0000	0.9992	64.00	0	
0.0004	0.9992	66.00	2	
0.0000	0.9996	68.00	0	
0.0002	0.9996	70.00	1	
0.0000	0.9998	72.00	0	
0.0000	0.9998	74.00	0	
0.0000	0.9998	76.00	0	
0.0000	0.9998	78.00	0	
0.0002	0.9998	80.00	1	
	1.0000	82.00		

STATISTICS FOR OUTPUT VARIABLE CUNIT (14) JTYPE=-99 SAMPL.DIST TYPE: NOT-APPL

MINIMUM 0.49148 MAXIMUM 7.8217

MEAN 2.7276 VARIANCE 1.2089 STD.DEVIATION 1.0995

PERCENTILES: 2.5%0.913 5.0% 1.10 50.0% 2.62 95.0% 4.73 97.5% 5.20

5000 POINT HISTOGRAM * = 6POINTS

PROB	CUME	VALUE		
0.0012	0.0000	0.4000	6	*
0.0102	0.0012	0.6000	51	*********
0.0240	0.0114	0.8000	120	**********
0.0292	0.0354	1.000	146	**********
0.0372	0.0646	1.200	186	**********
0.0462	0.1018	1.400	231	**********
0.0530	0.1480	1.600	265	**********
0.0682	0.2010	1.800	341	**********
0.0758	0.2692	2.000	379	**********
0.0760	0.3450	2.200	380	**********
0.0714	0.4210	2.400	357	**********
0.0734	0.4924	2.600	367	**********
0.0714	0.5658	2.800	357	**********
0.0654	0.6372	3.000	327	**********
0.0544	0.7026	3.200	272	**********
0.0482	0.7570	3.400	241	**********
0.0378	0.8052	3.600	189	**********
0.0350	0.8430	3.800	175	**********
0.0260	0.8780	4.000	130	**********
0.0206	0.9040	4.200	103	**********
0.0166	0.9246	4.400	83	**********
0.0132	0.9412	4.600	66	**********
0.0116	0.9544	4.800	58	**********
0.0090	0.9660	5.000	45	*********
0.0058	0.9750	5.200	29	*****
0.0042	0.9808	5.400	21	****
0.0026	0.9850	5.600	13	**
0.0034	0.9876	5.800	17	***
0.0032	0.9910	6.000	16	***
0.0018	0.9942	6.200	9	**
0.0018	0.9960	6.400	9	**
0.0002	0.9978	6.600	1	
0.0002	0.9980	6.800	1	
0.0004	0.9982	7.000	2	
0.0004	0.9986	7.200	2	
0.0004	0.9990	7.400	2	
0.0004	0.9994	7.600	2	
0.0002	0.9998	7.800	1	
	1.0000	8.000		

Appendix E

A RANDOM NUMBER GENERATOR FOR COMPUTERS USING THE INTEL 80X86 MICROPROCESSOR

T. C. Tucker
Martin Marietta Energy Systems, Inc.

The family of computers based upon the Intel 8086, 8088, 80286, and 80386 microprocessors use an IEEE standard for numeric processing that is incompatible with common pseudo-random number generators. These generators calculate a pseudo-random integer by a formula of the type

i(n) = modulus (i(n-1) * ia ,im) + ic

where ia, ic, and im are suitably chosen constants. The ratio of i(n)/im is then used as a uniformly distributed random number in the range [0,1]. One popular implementation is the FORTRAN function URAND, based upon a presentation in Knuth, Vol. 2, (Ref. E.1.).

Commonly, im is the largest integer which can be expressed in the computer. The modulus operation is then performed automatically through truncation of a multiplication overflow. The IEEE standard prescribes that integer overflow (and other arithmetic errors) return a fixed bit pattern which is an error result. This pattern, if used in a succeeding calculation, again causes the error pattern to be produced. After an initial set of random numbers, the first overflow occurs, and all succeeding numbers are identical.

I have rewritten the multiplication and modulus operations by expressing each integer as a linear polynomial and operating on them term by term so that the number range of the machine is never exceeded. The revised algorithm was first written and tested in FORTRAN. To achieve greater speed, a second version (TCTRAN.ASM) was written in assembly language, currently linkable (to TCTRAN-.OBJ) only with codes compiled with Ryan-McFarland or IBM Professional Fortran.

The next three pages present the source code for three random number generators based on a PASCAL generator described in Ref. E.2. The first is for the Microsoft Fortran compiler, the second for the RM Fortran compiler, and the third for a PASCAL compiler. One of these can be compiled to form OBJ modules to replace TCTRAN.OBJ if necessary.

REFERENCES

E.1 Knuth, D. E., "Seminumerical Algorithms," *The Art of Computer Programming*, Vol. 2; Addison-Wesley; Reading, MA (1969)

E.2 Park, S. K. and Miller, K. W., "Random Number Generators: Good Ones are Hard to Find"; *Communications of the ACM*, Vol. 31, No. 10, October 1988.

```
         type ranrm.for
c     Random number generator for Ryan-McFarland (RM) Fortran

      function random(dummy)
c****
c     translated from Ref E.2 by
c     T.C.Tucker
c     Computing and Telecommunications Division
c     Martin Marietta Energy Systems, Inc.
c****
      real*4 random
      real*8 a,m,seed,temp
      common /ranumb/ seed
      data a/16807.0/, m/2147483647d0/
c
      temp = a * seed
      seed = temp - m * dint(temp/m)
      random = seed / m
c
      return
      end
      subroutine ranseed(iy)
      real*8 seed
      integer*4 iy
      common /ranumb/ seed
      seed=iy
      return
      end
      block data randseed
      real*8 seed
      common /ranumb/ seed
      data seed / 1.0 /
      end
```

Fig. E.1. Random number generator for Ryan McFarland Fortran.

```
    type ranms.for
c     Random number generator for Microsoft Fortran
c
      function random(dummy)
c****
c     translated from Reference E.2 by
c     T.C.Tucker
c     Computing and Telecommunications Division
c     Martin Marietta Energy Systems, Inc.
c****
      real*4 random
      real*8 a,m,seed,temp
      common /ranumb/ seed
      data a/16807.0/, m/2147483647d0/
c
      temp = a * seed
      seed = temp - m * dint(temp/m)
      random = seed / m
c
      return
      end
      subroutine ranseed(iy)
      real*8 seed
      integer*4 iy
      common /ranumb/ seed
      seed=iy
      return
      end
```

Fig. E.2. Random number generator for Microsoft Fortran.

```
    type ran.pas
Pascal random number generator from Ref E.2

program testran;
var seed,u : real ; n : integer;
function random: real;
const
  a = 16807.0;
  m = 2147483647.0;
var
  temp : real;
begin
  temp := a*seed;
  seed := temp - m*trunc(temp/m);
  random := seed/m;
end;
begin
  seed := 1.0;
  for n := 1 to 10000 do
    u := random;
  writeln('The current value of seed is : ',seed);
end.
```

Fig. E.3. Random number generator using Pascal.

APPENDIX F

LIST OF ACRONYMS

AGC	Advanced Gas Centrifuge
AIS	Advanced Isotope Separation
AVLIS	Atomic Vapor Laser Isotope Separation
CDF or cdf	Cumulative Density Function
COS	Coefficient of Sensitivity
C&TD	Computing and Telecommunications Division of Martin Marietta Energy Systems, Inc.
DOE	U.S. Department of Energy
DOS	Disk Operating System
DP	Office of Defense Programs; U.S. DOE
EOF	End-of-File
ETD	Engineering Technology Division (ORNL)
FAST-E	Freiman Analysis of Systems Technique - Equipment (a parametric cost model)
FOM	Figure of Merit
I-O	Input-output
LMR	Liquid Metal Reactor
MLIS	Molecular Laser Isotope Separation
MVSS	Multivariable Sensitivity Study
NDRN	Normally Distributed Random Number
NPR	New Production Reactor
OA&P	Operations Analysis & Planning Division of ORGDP
ORNL	Oak Ridge National Laboratory
ORGDP	Oak Ridge Gaseous Diffusion Plant
ORMONTE	Oak Ridge Monte Carlo (Code Name)
PDA	Probability Data Analysis
PDF or pdf	Probability Density Function
PC	Personal Computer
PRA	Probabilistic Risk Assessment
PSP	Plasma Separation Process
RM	Ryan McFarland (FORTRAN software)
SDI	Strategic Defense Initiative
SVSS	Single Value Sensitivity Study
UCC-ND	Union Carbide Corporation - Nuclear Division
UDRN	Uniformly Distributed Random Number

ORNL/TM-10714
Dist. Category UC-05 & UC-32

Internal Distribution

1. M. A. Akerman
2. T. D. Anderson
3. D. E. Bartine
4. J. B. Berry
5. M. M. Brown
6. T. J. Burns
7. R. L. Campbell
8. J. J. Coen
9. K. E. Cross
10. J. G. Delene
11. D. M. Eissenberg
12. A. K. Farrar
13. C. A. Flanagan
14. J. D. Galambos
15. E. C. Fox
16. L. C. Fuller
17. J. A. Getsi
18. J. M. Googin
19. D. L. Hanline
20. R. A. Hoffman
21. R. F. Holdaway
22-31. C. R. Hudson, II
32. J. E. Jones Jr.
33. D. L. Mason
34. B. L. McGill
35. J. R. Merriman
36. G. E. Michaels
37. J. W. Miller
38. W. R. Rish
39. L. M. Rosenbaum
40. J. E. Rushton
41. J. P. Sanders
42. D. B. Simpson
43. A. Sozer
44. D. A. Steinert
45. R. Taleyarkhan
46. M. J. Taylor
47. H. E. Trammell
48. T. C. Tucker
49. C. L. Wagner
50-59. K. A. Williams
60. V. K. Wilkinson
61. ORNL Patent Office
62. Central Research Library
63. Document Reference Section
64-65. Laboratory Records Department
66. Laboratory Records (RC)

External Distribution

67. Office of Assistant Manager for Defense Programs, Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37831
68. Office of Assistant Manager for Energy Research and Development, Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37831
69. R. E. Allen, EEDB Program Project Manager, United Engineers & Constructors, Inc., P.O. Box 8223, Philadelphia, PA 19101
70. W. B. Arthur, 15 Newcastle Lane, Oak Ridge, TN 37830
71. R. Asamoto, General Electric Company, P.O. Box 53094, San Jose, CA 95153
72. L. L. Bennett, Head, Planning and Economic Studies Section, International Atomic Energy Agency, Wagremerstrasse 5, P.O. Box 100, A-1400, Vienna, Austria
73. H. I. Bowers, 104 Adelphi Road, Oak Ridge, TN 37830

74. W. W. Brandfon, Sargent & Lundy, 55 East Monroe, Chicago, IL 60603
75. Chaim Braun, United Engineers & Constructors, Inc., P.O. Box 8223, Philadelphia, PA 19101
76. Y. I. Chang, Bldg. 208, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439
77. J. Colsh, NE-40:GTN, Department of Energy, Washington, DC 20545
78. R. E. Dierlam, NE-35:GTN, Department of Energy, Washington, DC 20545
79. C. W. Ebbecke, NP-20:FORS, 1000 Independence Avenue SW, Washington, DC 20585
80. R. J. Fete, Nuclear Equipment Division, Babcock & Wilcox, 91 Stirling Avenue, Barberton, OH 44203
81. A. H. Fuldner, EI-542, 2G-060, Energy Information Administration, Forrestal Bldg., 1000 Independence Avenue SW, Washington, DC 20585
82. J. C. Geidl, EI-50,BG-041, Energy Information Administration, Forrestal Bldg., 1000 Independence Avenue SW, Washington, DC 20585
83. M. Gielecki, EI0532, BF-115, Energy Information Administration, Forrestal Bldg., 1000 Independence Avenue SW, Washington, DC 20585
84. L. Grantham, Rockwell International Corporation, Rocketdyne Division, 6633 Canoga Avenue, Canoga Park, CA 91304
85. J. Griffith, NE-40:GTN, Department of Energy, Washington, DC 20545
86. M. Grunspan, NE-44:GTN, Department of Energy, Washington, DC 20545
87. N. Haberman, NE-35:GTN, Department of Energy, Washington, DC 20545
88. J. Hewlett, EI-532, BF-115, Energy Information Administration, Forrestal Bldg., 1000 Independence Avenue SW, Washington, DC 20585
89. R. L. Hoglund, Martin Marietta International, C/0 Rose Wood, 55 Jefferson Circle, P.0. Box 2003, MS 7603, Oak Ridge, TN 37831-7603
90. K. K. Hudson, DP-533:GTN, Department of Energy, Washington, DC 20545
91. B. A. Hutchins, General Electric Company, P.0. Box 530954, San Jose, CA 92121
92. J. J. Jicha, NP-40:FORS, 1000 Independence Avenue SW, Washington, DC 20585
93. E. Kokkelenberg, Department of Economics, SUNY at Binghamton, NY 13901
94. M. P. LaBar, Gas-Cooled Reactor Associates, 10240 Sorrento Valley Road, San Diego, CA 92121
95. S. J. Lanes, NE-50:GTN, Department of Energy, Washington, DC 20545
96. E. G. Lazur, DP-533:GTN, Department of Energy, Washington, DC 20545
97. M. L. Marx, TASC, 1700 N. Moore St., Suite 1800, Arlington, VA 22209
98. D. J. McGoff, NE-40:GTN, Department of Energy, Washington, DC 20545

99. E. F. Mastal, NE-53:GTN, Department of Energy, Washington, DC 20545
100. E. W. Merrow, Independent Project Analysis, 9409 Lagovista Court, Great Falls, VA 22066
101. T. A. Meterko, Manager of New Business Development, Occidental Chemical Corporation, 360 Rainbow Blvd. S., Niagara Falls, NY 14302
102. M. M. Mickelsen, RCED-Energy, USGAO, Room E-178, Germantown, MD 20874
103. A. C. Millunzi, NE-451:GTN, Department of Energy, Washington, DC 20545
104. M. L. Myers, 5404 Palmetto Road, Knoxville, TN 37921
105. National Energy Software Center, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439
106. J. D. Nulton, NP-40 FORS, Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585
107. R. G. Oehl, NE-44:GTN, Department of Energy, Washington, DC 20545
108. G. Pavlenco, United Engineers & Constructors, Inc., P.O. Box 8223, Philadelphia, PA 19101
109. J. Powers, NP-20:FORS, 1000 Independence Avenue SW, Washington, DC 20585
110. B. J. Rock, NE-46:GTN, Department of Energy, Washington, DC 20545
111. E. Rodwell, Electric Power Research Institute, P.O. Box 10412, Palo Alto, CA 94303
112. H. H. Rohm, NE-40; B-107, GTN, Department of Energy, Washington, DC 20545
113. G. Santos-Leon, NP-40: FORS, Department of Energy, 1000 Independence Avenue SW, Washington, DC 20585
114. W. F. Savage, NE-44:GTN, Department of Energy, Washington, DC 20545
115. B. R. Scarlett, MA-221:FORS, 1000 Independence Avenue SW, Washington, DC 20585
116. R. G. Schwarz, Applied Decision Analysis Corporation, Bldg. 4, Suite 255; 300 Sand Hill Rd., Menlo Park, CA 94025
117. B. H. Shapiro, NE-46:GTN, Department of Energy, Washington, DC 20545
118. D. Sonn, General Atomics, P.O. Box 85608, San Diego, CA 92138
119. B. St. John, NUS Corporation, 910 Clopper Rd., Gaithersburg, MD 20878
120. Stan Vejtasa, Electric Power Research Institute, 3412 Hillview Avenue, P.O. Box 10412, Palo Alto, CA 94305
121. Rhonda Walker, Lawrence Livermore National Laboratory, c/o Jeannette Morris, AVLIS Program, L-467, P.O. Box 808, Livermore, CA 94330
122. H. L. Walton, EI-532, BC-041, Energy Information Administration, Forrestal Bldg., 1000 Independence Avenue SW, Washington, DC 20585
123. T. A. Werner, NE-44:GTN, Department of Energy, Washington, DC 20454
124. R. Williams, NE-50:GTN, Department of Energy, Washington, DC 20545

125. R. D. Williamson, Idaho National Engineering Laboratory, EG&G Idaho Corporation, Idaho Falls, ID 83145

126-268. Given distribution as shown in DOE/OSTI-4500-INTERIM3 under categories UC-32 (Mathematics and Computer Science) and UC-05 (General Energy)

☆U.S. GOVERNMENT PRINTING OFFICE:1989 -648-119/ 00012 REGION NO. 4