PROBABILISTIC ANALYSIS
FOR FATIGUE STRENGTH DEGRADATION OF MATERIALS

Prepared by:
Lola Boyce, Ph. D., P. E.

Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135

(NASA-CR-182844) PROBABILISTIC ANALYSIS FOR
FATIGUE STRENGTH DEGRADATION OF MATERIALS
Annual Report (Texas Univ.) 133 p CSCL 20K

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989
PROBABILISTIC ANALYSIS

FOR FATIGUE STRENGTH DEGRADATION OF MATERIALS

Prepared by:

Lola Boyce, Ph. D., P. E.

Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989
TABLE OF CONTENTS

SECTION ................................................... PAGE

PREFACE. .................................................. i
ABSTRACT. ................................................ ii
LIST OF FIGURES. .......................................... iii
LIST OF TABLES. ............................................ iv
1.0 INTRODUCTION. ......................................... 1
2.0 FATIGUE CRACK GROWTH MODEL. ....................... 2
  2.1 Background. .......................................... 2
  2.2 RANDOM2 Computer Program. ........................ 3
3.0 FATIGUE STRENGTH REDUCTION MODEL. ............... 4
  3.1 Background. .......................................... 4
  3.2 RANDOM3 and RANDOM4 Computer Programs. ....... 5
  3.3 Sensitivity Study. .................................. 5
4.0 REFERENCES. .......................................... 9
5.0 APPENDIX 1: FATIGUE CRACK GROWTH MODEL: RANDOM2
                  USER MANUAL. ................................ 10
6.0 APPENDIX 2: FATIGUE STRENGTH DEGRADATION MODEL:
                  RANDOM3 AND RANDOM4 USER MANUAL. .... 47
The University of Texas at San Antonio (UTSA) is a relatively new university. It was established in 1969 and opened for classes in 1973. As the only comprehensive public university serving the nation’s ninth largest city, it was and is vital to San Antonio and the entire South Texas Region. In 1983, just seven years ago, an undergraduate engineering program was established at UTSA with the support of the community and its leaders. Today, all three undergraduate engineering programs are ABET accredited and serve about 800 students, a significant percentage of whom are Hispanic. The future includes a new engineering building, providing new laboratory facilities and equipment, together with offices and laboratories, planned to open in January, 1991. Furthermore, a graduate program is planned at both M.S. and Ph.D. levels, and it is hoped that the first Master’s Degree students will be able to enroll in Fall, 1989.

Naturally, the engineering research environment is just developing at UTSA. Now, thanks in great measure to the UT System support and this ongoing NASA grant, good progress is being made. Specifically, the purchase of a UT System CRAY-XM/P in March, 1986 and a second one in December, 1988 has provided a world-class analytical and numerical research environment not ordinarily available to a new university. As a result the UTSA Supercomputer Network Research Facility (SNRF) was developed by the principal investigator, Dr. Lola Boyce. This has allowed the successful completion of this research project, the first of its kind at UTSA.

This NASA research grant has allowed two Mechanical Engineering students, Thomas Lovelace and Callie Scheidt, to work directly with the principal investigator, Dr. Boyce, providing them with a quality research experience they would otherwise probably not have had. Both students have expressed an interest in continuing their educations at the graduate level.

In conclusion, and in view of the significant accomplishments in fundamental research, enhancement of the engineering research environment at UTSA, and direct support of Mechanical Engineering students, it is hoped that the proposed extension of this grant will receive favorable consideration at NASA. The principal investigator sincerely thanks NASA for funding this first year grant.
ABSTRACT

This report presents the results of the first year of effort of a program of research conducted for NASA-LeRC by The University of Texas at San Antonio (UTSA). The research included development of methodology that provides a probabilistic treatment of lifetime prediction of structural components of aerospace propulsion systems subjected to fatigue. Material strength degradation models, based on primitive variables, include both a fatigue strength reduction model and a fatigue crack growth model. Linear elastic fracture mechanics is utilized in the latter model. Probabilistic analysis is based on simulation, and both maximum entropy and maximum penalized likelihood methods are used for the generation of probability density functions. The resulting constitutive relationships are included in several computer programs, RANDOM2, RANDOM3 AND RANDOM4. These programs determine the random lifetime, of an engine component, in mechanical load cycles, to reach a critical fatigue strength or crack size. The material considered was a cast nickel base-superalloy, one typical of those used in the Space Shuttle Main Engine (SSME).
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>p.d.f. of base line room temperature (RT) problem.</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>c.d.f. of base line room temperature (RT) problem.</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>c.d.f. of base line room temperature (RT) problem compared with c.d.f. of base line high temperature (HT) problem.</td>
<td>8</td>
</tr>
<tr>
<td>A1.1</td>
<td>p.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.</td>
<td>19</td>
</tr>
<tr>
<td>A1.2</td>
<td>c.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.</td>
<td>20</td>
</tr>
<tr>
<td>A2.1</td>
<td>p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.</td>
<td>58</td>
</tr>
<tr>
<td>A2.2</td>
<td>c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.</td>
<td>58</td>
</tr>
<tr>
<td>A2.3</td>
<td>p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.</td>
<td>59</td>
</tr>
<tr>
<td>A2.4</td>
<td>c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.</td>
<td>59</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>Description</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Base line room temperature (RT) problem input using the fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4).</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Sensitivity study cases for fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4).</td>
<td>7</td>
</tr>
<tr>
<td>A1.1</td>
<td>RANDOM2 Sample problem input (SI units).</td>
<td>18</td>
</tr>
<tr>
<td>A1.2</td>
<td>Physical quantities, symbols, and units for fatigue crack growth model for RANDOM3 and RANDOM4.</td>
<td>22</td>
</tr>
<tr>
<td>A2.1</td>
<td>RANDOM3 and RANDOM4 input (SI units).</td>
<td>56</td>
</tr>
<tr>
<td>A2.2</td>
<td>Physical quantities, symbols, and units for fatigue crack growth model for RANDOM 3 and RANDOM4.</td>
<td>61</td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

This report presents the results of the first year effort of a research program entitled "Development of Advanced Methodologies for Probabilistic Constitutive Relationships of Material Strength Models." This research is sponsored by the National Aeronautics and Space Administration-Lewis Research Center (NASA-LeRC). The principal investigator is Dr. Lola Boyce, Associate Professor of Mechanical Engineering, The University of Texas at San Antonio (UTSA). The objective of the research program is the development of methodology that provides a probabilistic treatment of lifetime prediction of structural components of aerospace propulsion systems subjected to fatigue.

Two material strength degradation models, based on primitive variables were developed as part of this first year effort: a fatigue crack growth model and a fatigue strength reduction model. The former model utilizes principles of linear elastic fracture mechanics while the latter is, recently developed at NASA-LeRC, quantifies the reduction of strength under cyclic loading, including elevated temperature treatment. Probabilistic analysis is based on simulation, and both maximum entropy and maximum penalized likelihood methods are used for the generation of probability density functions that predict the random lifetime of a material typical of those used in the Space Shuttle Main Engine (SSME), namely a cast nickel base-superalloy.

The resulting constitutive relationships are included in several computer programs, RANDOM2, RANDOM3, and RANDOM4. The programs were developed using both the NASA-LeRC and UTSA Supercomputer Network Research Facility (SNRF) Cray X-MP. New versions of the program accompany this report (see enclosed floppy disk), utilizing the new IMSL Ver. 10 subroutines. Thus, these new versions of the programs will execute on the current NASA-LeRC supercomputer facilities. Also the floppy disk contains sample problems to verify program performance at NASA-LeRC.

Finally, a sensitivity study was carried out for the fatigue strength reduction model for the case of a relatively high mean stress and a relatively low constant amplitude alternating stress at failure. In addition to varying the stresses, the effect of temperature was also considered. A paper was produced documenting much of the effort of this first year research program. This paper is entitled "Probabilistic Constitutive Relationships for Cyclic Material Strength Models", by L. Boyce and C.C. Chamis. It was presented at the 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, April, 1988 and is published in the Proceedings. It has also been submitted to the AIAA Journal of Propulsion and Power.
2.0 FATIGUE CRACK GROWTH MODEL

2.1 Background

Fatigue crack growth data are usually presented as cycles, N, to reach a particular crack length, a. The initial crack size is a_i. It is generally accepted that under constant amplitude alternating stress, fatigue crack growth can be related to stress intensity through a first order differential equation

\[ \frac{da}{dN} = C(\Delta K)^m \]  

where C is a material parameter, m is a material property (often a constant) and \( \Delta K \) is the stress intensity range. Stress intensity range is given by

\[ \Delta K = Y \Delta \sigma / \pi a \]

where Y is a constant dependent upon component and crack geometry and \( \Delta \sigma \) is the constant amplitude alternating stress. Therefore, equation (1) can be written as

\[ \frac{da}{dN} = C(Y\Delta \sigma / \pi a )^m \]

or,

\[ \frac{da}{dN} = C Y^m \Delta \sigma^m \pi^{m/2} a^{m/2}. \]

Equation (2) can be integrated, from the initial crack length, a_i, to the final crack length, a_f, to yield N, the number of cycles. The result is

\[ N = \frac{1}{CY^m \pi^{m/2} \Delta \sigma^m} \left[ \frac{a_f^{-m/2+1} - a_i^{-m/2+1}}{-m/2 + 1} \right] \]

Thus, equation (3) gives the "cycles to reach a given crack length."

Metallurgical evidence indicates that casting pores play a significant role in the high-cycle fatigue life of cast nickel base-superalloys, especially at high temperatures. The location and size of these fatigue crack-initiating pores vary greatly from one aerospace propulsion system component to another. This accounts for the large variability in fatigue life and leads to consideration of fatigue crack growth as a random phenomenon.

Fatigue life directly relates to casting pore size, and pore size can be used to determine initial crack size, a_i. Thus, utilizing principles of both probabilistic analysis and fatigue crack growth, a quantitative probabilistic constitutive relationship between fatigue life and fracture mechanics parameters can be developed. Using the "randomized equation" approach, the fatigue crack growth model, given by equation (3) has the following form:

\[ N = f(C,m,\Delta \sigma,a_i,a_f,Y) \]
or, in general,
\[ N = f(X_i), \quad i = 1, ..., 6, \]  
(5)

where the \( X_i \) are the six independent variables in equations (3) and (4). Equation (3) is "randomized" by assuming the first four variables in equation (4) to be random. Assuming a small crack in a relatively large component leads to assuming \( Y = 1.0 \), a deterministic value. A deterministic final crack size was chosen since experimental evidence indicated that it was relatively unimportant.\(^1\)

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, \( N \). A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method. Maximum penalized likelihood generates the p.d.f. estimate using the method of maximum likelihood together with a penalty function to smooth it.\(^3\)

2.2 RANDOM2 Computer Program

A FORTRAN computer program for the fatigue crack growth model, called RANDOM2, was written using the above-described probabilistic methodology and the constitutive relationship expressed in equation (3). Although the four independent random variables could have any distribution, this initial program provided for normal or lognormal only.

A complete Users Manual for RANDOM2 is contained in Appendix 1. Also, a disk containing a new version of RANDOM2 and a sample problem accompanies this report. The new version of RANDOM2, documented in the Users Manual, uses the new ISML, Ver. 10 subroutines and provides for parameter input from an input file.
3.0 FATIGUE STRENGTH REDUCTION MODEL

3.1 Background

Fatigue strength data are usually presented as cycles to failure for each of several stress amplitudes, the familiar S-N diagram. Results indicate that for lower stress amplitudes the cycles (or time) to failure increases. Thus, a power curve fit through the data yields a monotonically decreasing curve. In general, this curve is represented as

\[ S = \left(\frac{N}{C'}\right)^{-1/m'} \]  

where the primitive variables in this equation are as follows: S is the applied constant amplitude alternating stress at failure or fatigue strength, N is number of cycles, C' is a material parameter that varies from specimen to specimen and m' is a material constant.\(^4\) Equation (6) can be written in terms of "cycles to reach a given fatigue strength" as

\[ N = C' S^{-m'} \]  

Recently another fatigue strength reduction model has been proposed that takes into account the effect of temperature as well as other parameters that affect strength.\(^5\) The general form of the constitutive relationships for this model is applied to the constituents of high temperature composite materials. Specifically, it is applied herein for the case of a single material constituent. The mechanical property of interest is fatigue strength which is expressed in terms of primitive variables, including the general categories of temperature, mechanical cycles and mean stress. For these categories, the relationship becomes

\[ \frac{S}{S_0} = \left[ \left( \frac{T_F - T}{T_F - T_0} \right)^n \left[ \frac{S_F - \sigma}{S_F - \sigma_0} \right]^m \left[ \frac{\log N_{MF} - \log N_M}{\log N_{MF} - \log N_{MO}} \right]^q \right] \]  

where S is the applied constant amplitude alternating stress at failure (fatigue strength) at current (or operating) temperature, T, mean stress, \( \sigma \), and mechanical cycle, \( N_M \). \( S_0 \) is fatigue strength at reference temperature, \( T_0 \) (usually room temperature), reference mean stress (or residual stress), \( \sigma_0 \), and reference mechanical cycle, \( N_{MO} \). Also, \( T_F \) is the final or melting temperature of the material, \( S_F \) is the final or tensile strength of the material, and \( N_{MF} \) is the final mechanical cycle or lifetime. Empirical parameters, \( n \), \( m \), and \( q \), are determined from available experimental data or estimated from anticipated behavior of the particular product term.\(^6\) Note that the term containing mechanical cycles is expressed in terms of the log of cycles rather than cycles. This formulation is attractive when \( N_M \) and \( N_{MO} \) are small compared to \( N_{MF} \). The equation may be solved for \( N_M \), or the "cycles to reach a given fatigue strength." The expression is

\[ N = 10 \exp \left[ \log N_{MF} - \left( \log N_{MF} - \log N_{MO} \right) \left[ \frac{S}{S_0} \left( \frac{T_F - T}{T_F - T_0} \right) \left[ \frac{S_F - \sigma}{S_F - \sigma_0} \right]^m \right]^{1/q} \right] \]  

(9)
For values typical of a cast nickel base-superalloy subjected to typical loads and temperatures, equation (9) indicates increasing life for decreasing temperature, decreasing tensile mean stress, and decreasing applied alternating stress. It indicates decreasing life for increasing temperature, decreasing compressive mean stress, and increasing applied alternating stress. Therefore, equation (9) predicts observed trends in general.

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method for RANDOM3. For RANDOM4, a p.d.f. of cycles is generated using the maximum entropy method. Maximum entropy uses Jaynes' principle which says that "the minimally prejudiced distribution is that which maximizes the entropy subjected to the constraints supplied by the given information."7

3.2 RANDOM3 and RANDOM4 Computer Programs

FORTRAN computer programs for the fatigue strength reduction model called RANDOM3 and RANDOM4 were written using the above-described probabilistic methodology and the constitutive relationship expressed in equation (9). Although the thirteen independent random variables could have any distribution, these programs provided for normal or lognormal only.

A complete Users Manual for RANDOM3 and RANDOM4 is contained in Appendix 2. Also, a disk containing new versions of RANDOM3 and RANDOM4 uses the new IMSL, Ver. 10 subroutines and provides for parameter input from an input file.

3.3 Sensitivity Study

The fatigue strength degradation model using the maximum entropy method of p.d.f. generation (RANDOM4) was selected for use in a sensitivity study. A base line problem utilizing a high mean stress (σ = 90 ksi) and a low constant amplitude alternating stress at failure (S = 22.5 ksi) was established. A room temperature (T = 68°F) problem was executed. The input for this problem is given in Table 1 and the output, in the form of a p.d.f. and a c.d.f. is given in Figures 1 and 2. A high temperature (T = 1562°F) base line problem was also selected. Then, for a fixed base line alternating stress at failure, the mean stress was varied above and below the base line value. Both room and high temperatures were selected. Finally, for a fixed base line mean stress, the alternating stress at failure was varied above and below the base line value. Again, room and high temperatures were selected. A summary of the cases studied is given in Table 2.

Conclusions drawn from this sensitivity study are summarized below. Increasing temperature for the same stress conditions reduces lifetime for all cases (see, for example, Figure 3). At room temperature, when mean stress is increased by 10%, lifetime decreases only very slightly. At high temperature, however, when mean stress is increased by 10%, lifetime decreases substantially. Also, at room temperature, when alternating stress is increased by 30%, lifetime decreases only slightly. At high temperatures, however, when alternating stress is increased by 30%, lifetime decreases very substantially. Considering the above points, lifetime is more sensitive to increasing alternating stress, rather than mean stress. This is probably because alternating stress was increased by 30%, whereas mean stress was increased by only 10%.
Table 1 Baseline room temperature (RT) problem input, using the fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Distribution Type</th>
<th>Mean</th>
<th>Std. Dev. Value % of Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_F$ (Melting Temp.)</td>
<td>Normal</td>
<td>2732.0 °F</td>
<td>82.0</td>
</tr>
<tr>
<td>$S_F$ (Ult. Tensile Str.)</td>
<td>Lognormal</td>
<td>130.0 ksi</td>
<td>6.5</td>
</tr>
<tr>
<td>$N_{MF}$ (Log of Final Cycle)</td>
<td>Lognormal</td>
<td>8.0</td>
<td>0.8</td>
</tr>
<tr>
<td>$T_0$ (Ref. Temp.)</td>
<td>Normal</td>
<td>68.0 °F</td>
<td>2.0</td>
</tr>
<tr>
<td>$\sigma_0$ (Residual Comp. Stress)</td>
<td>Lognormal</td>
<td>-2.9 ksi</td>
<td>0.145</td>
</tr>
<tr>
<td>$N_M$ (Log of Ref. Cycle)</td>
<td>Lognormal</td>
<td>7.0</td>
<td>0.7</td>
</tr>
<tr>
<td>$S_0$ (Ref. Fatigue Str.)</td>
<td>Lognormal</td>
<td>72.6 ksi</td>
<td>3.6</td>
</tr>
<tr>
<td>$T$ (Current Temp.)</td>
<td>Normal</td>
<td>68.0 °F</td>
<td>2.0</td>
</tr>
<tr>
<td>$\sigma$ (Current Mean Stress)</td>
<td>Lognormal</td>
<td>90.0 ksi</td>
<td>4.5</td>
</tr>
<tr>
<td>$S$ (Current Fatigue Str.)</td>
<td>Lognormal</td>
<td>22.5 ksi</td>
<td>1.125</td>
</tr>
<tr>
<td>$n$ (Temp. Exponent)</td>
<td>Normal</td>
<td>0.5</td>
<td>0.015</td>
</tr>
<tr>
<td>$m$ (Stress Exponent)</td>
<td>Normal</td>
<td>0.5</td>
<td>0.015</td>
</tr>
<tr>
<td>$q$ (Cycle Exponent)</td>
<td>Normal</td>
<td>0.5</td>
<td>0.015</td>
</tr>
</tbody>
</table>

![Fig. 1 p.d.f. of baseline room temperature (RT) problem](image-url)
Fig. 2  c.d.f. of base line room temperature (RT) problem

Table 2  Sensitivity study cases for fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4)

<table>
<thead>
<tr>
<th>σ (ksi)</th>
<th>S (ksi)</th>
<th>T (°F)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base Line (RT)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>22.5</td>
<td>68</td>
</tr>
<tr>
<td>90</td>
<td>22.5</td>
<td>68</td>
</tr>
<tr>
<td>100</td>
<td>22.5</td>
<td>68</td>
</tr>
<tr>
<td><strong>Base Line (HT)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>22.5</td>
<td>1562</td>
</tr>
<tr>
<td>90</td>
<td>22.5</td>
<td>1562</td>
</tr>
<tr>
<td>100</td>
<td>22.5</td>
<td>1562</td>
</tr>
<tr>
<td><strong>Base Line (RT)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>15.0</td>
<td>68</td>
</tr>
<tr>
<td>90</td>
<td>22.5</td>
<td>68</td>
</tr>
<tr>
<td>90</td>
<td>30.0</td>
<td>68</td>
</tr>
<tr>
<td><strong>Base Line (HT)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>15.0</td>
<td>1562</td>
</tr>
<tr>
<td>90</td>
<td>22.5</td>
<td>1562</td>
</tr>
<tr>
<td>90</td>
<td>30.0</td>
<td>1562</td>
</tr>
</tbody>
</table>
Fig. 3 c.d.f. of base line room temperature (RT) problem compared with c.d.f. of base line high temperature (HT) problem.
4.0 REFERENCES


5.0 APPENDIX 1

FATIGUE CRACK GROWTH MODEL:
RANDOM2 USER MANUAL
FATIGUE CRACK GROWTH MODEL
RANDOM2 USER MANUAL

Prepared by:
Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 1
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>2.0 Theoretical Background</td>
<td>2</td>
</tr>
<tr>
<td>3.0 Input Data</td>
<td>4</td>
</tr>
<tr>
<td>4.0 Sample Problem</td>
<td>6</td>
</tr>
<tr>
<td>5.0 References</td>
<td>9</td>
</tr>
<tr>
<td>6.0 Appendix A: Physical Quantities, Symbols, and Units</td>
<td>10</td>
</tr>
<tr>
<td>7.0 Appendix B: RANDOM2 Sample Problem: Source, Input and Output Files</td>
<td>11</td>
</tr>
<tr>
<td>8.0 Appendix C: IMSL Subroutine Calls from RANDOM2</td>
<td>33</td>
</tr>
<tr>
<td>9.0 Appendix D: SAS/GRAPH Program for RANDOM2</td>
<td>34</td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

This User Manual documents the FORTRAN program RANDOM2. RANDOM2 is based on fracture mechanics using a probabilistic fatigue crack growth model. It predicts the random lifetime of an engine component to reach a given crack size (see Section 2.0, Theoretical Background).

Included in this Manual are details regarding the theoretical background of RANDOM2, input data instructions and a sample problem illustrating the use of RANDOM2. Appendix A gives information on the physical quantities, their symbols, FORTRAN names, and both SI and U.S. Customary units. Appendix B includes photocopies of the actual computer printout corresponding to the sample problem. Appendices C and D detail the IMSL, Ver. 10 $^1$, subroutines and functions called by RANDOM2 and a SAS/GRAPH $^2$ program that can be used to plot both the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.).
2.0 THEORETICAL BACKGROUND

Fatigue crack growth data are usually presented as cycles, \( N \), to reach a particular crack length, \( a \). The initial crack size is \( a_i \). It is generally accepted that under constant amplitude alternating stress, fatigue crack growth can be related to stress intensity through a first order differential equation.\(^3\)

\[
\frac{da}{dN} = C(\Delta K)^m \tag{1}
\]

where \( C \) is a material parameter, \( m \) is a material property (often a constant) and \( \Delta K \) is the stress intensity range. Stress intensity range is given by

\[
\Delta K = Y \Delta \sigma / \pi a
\]

where \( Y \) is a constant dependent upon component and crack geometry and \( \Delta \sigma \) is the constant amplitude alternating stress. Therefore, equation (1) can be written as

\[
\frac{da}{dN} = C(\Delta \sigma / \pi a)^m
\]

or,

\[
\frac{da}{dN} = C Y^m \Delta \sigma^m \pi^{m/2} a^{m/2}. \tag{2}
\]

Equation (2) can be integrated, from the initial crack length, \( a_i \), to the final crack length, \( a_f \), to yield \( N \), the number of cycles. The result is

\[
N = \frac{1}{C Y^m \pi^{m/2} \Delta \sigma^m} \left[ \frac{a_f^{m/2+1} - a_i^{m/2+1}}{-m/2 + 1} \right] \tag{3}
\]

Thus, equation (3) gives the "cycles to reach a given crack length."

Metallurgical evidence indicates that casting pores play a significant role in the high-cycle fatigue life of cast nickel base-superalloys, especially at high temperatures.\(^4\) The location and size of these fatigue crack-initiating pores vary greatly from one aerospace propulsion system component to another. This accounts for the large variability in fatigue life and leads to consideration of fatigue crack growth as a random phenomenon.

Fatigue life directly relates to casting pore size, and pore size can be used to determine initial crack size, \( a_i \). Thus, utilizing principles of both probabilistic analysis and fatigue crack growth, a quantitative probabilistic constitutive relationship between fatigue life and fracture mechanics parameters can be developed. Using the "randomized equation" approach, the fatigue crack growth model, given by equation (3) has the following form:

\[
N = f(C, m, \Delta \sigma, a_i, a_f, Y) \tag{4}
\]
or, in general,

\[ N = f(X_i), \ i = 1, \ldots, 6, \]  

where the \( X_i \) are the six independent variables in equations (3) and (4). Equation (3) is "randomized" by assuming the first four variables in equation (4) to be random. Assuming a small crack in a relatively large component leads to assuming \( Y = 1.0 \), a deterministic value. A deterministic final crack size was chosen since experimental evidence indicated that it was relatively unimportant.\(^3\)

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, \( N \). A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method. Maximum penalized likelihood generates the p.d.f. estimate using the method of maximum likelihood together with a penalty function to smooth it.\(^5\)
3.0 INPUT DATA

Data input for RANDOM2 is user friendly and easy to manipulate (see, for example, the file entitled NORMAL.INP, in Section 4.0). The first five lines of input have the same format, namely 2E12.4, and the last two lines differ. The last two lines of input have the formats I3,2X,I3,2X,2E12.4,2X,I3 and I3, respectively. A brief line by line description is given along with an example for each line (Note: the ruler is to aid the user in formatting and is not a part of the input). A table listing the physical quantities, their units and symbols is given in Appendix A.

1. Random Number Generator Seed, ISEED, and Sample Size, NTOT

EXAMPLE:

123456789012345678901234567890
1
40

2. Material Property, RMM

EXAMPLE:

123456789012345678901234567890
28.0E-01 1.4E-01

3. Initial Crack Size (Pore Diameter), RAI

EXAMPLE:

123456789012345678901234567890
300.0E-06 45.0E-06

4. Material Property, RCC

EXAMPLE:

123456789012345678901234567890
2.20E-11 0.22E-11

5. Stress Range, DELSIG

EXAMPLE:

123456789012345678901234567890
6.2E+02 6.2E+01
6. The DESPL \(^1\) parameters are NODE, INIT, ALPHA, EPS, MAXIT and are entered in that order as follows:

**EXAMPLE:**

```
1234567890123456789012345678901234567890
21 0 50.0E-01 10.0E-05 30
```

7. The DESPL parameter, IOPT, is entered as follows:

**EXAMPLE:**

```
1234567890
2
```
4.0 SAMPLE PROBLEM FOR RANDOM2

The objective of this program is to predict the random lifetime, to reach a given crack size for an engine component. The theory is based on fracture mechanics, using a probabilistic fatigue crack growth model (see Section 2.0, Theoretical Background). RANDOM2 input parameters are given in Table A1.1. Note that the first four parameters are random. Their means and standard deviations are input by the user. The last two parameters, $A_f$ and $\gamma$, are deterministic and are fixed internally by the program. They are equal to the values shown in Table A1.1.

Table A1.1 RANDOM2 sample problem input (SI units)

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Distribution Type</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(Value)</td>
<td>(%) of Mean</td>
</tr>
<tr>
<td>RMM</td>
<td>normal</td>
<td>28.0E-01</td>
<td>1.4E-01</td>
</tr>
<tr>
<td>AI</td>
<td>lognormal</td>
<td>300.0E-06</td>
<td>45.0E-06</td>
</tr>
<tr>
<td>RCC</td>
<td>lognormal</td>
<td>2.20E-11</td>
<td>0.22E-11</td>
</tr>
<tr>
<td>DELSIG</td>
<td>lognormal</td>
<td>6.2E+02</td>
<td>6.2E+01</td>
</tr>
<tr>
<td>AF</td>
<td>N/A</td>
<td>2.0E-03</td>
<td>N/A</td>
</tr>
<tr>
<td>YY</td>
<td>N/A</td>
<td>1.0</td>
<td>N/A</td>
</tr>
</tbody>
</table>

The input is entered in the following format in a file entitled NORMAL.INP.

```
1234567890123456789012345678901234567890
1        40
28.0E-01  1.4E-01
300.0E-06 45.0E-06
2.20E-11  0.22E-11
6.2E+02  6.2E+01
21       0      50.0E-01  10.0E-05  30
2
```
Execution of RANDOM2 (source code entitled NR2.FOR) produces an output file entitled RANDM22 giving intermediate results (see Appendix B). Execution also produces the plotfiles OUT1 and OUT2 (see Appendix B). These files are used to plot the X and Y axes of the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.), respectively, generated by RANDOM2. The plots are drawn from the plotfiles by the SAS/GRAPH graphing program (see Appendix C). These plots for the sample problem are shown in Figures A1.1 and A1.2.

This same sample problem has been reported in Boyce and Chamis. There, however, it utilized U.S. Customary units and an older version of RANDOM2 (IMSL Version 9.2 subroutines).

Fig. A1.1  p.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.
Fig. A1.2 c.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.
5.0 REFERENCES


6.0 APPENDIX A

PHYSICAL QUANTITIES, SYMBOLS, AND UNITS

The physical quantities, their symbols, and units for the fatigue crack growth model are given in the following table.

Table A1.2 Physical quantities, symbols, and units for fatigue crack growth model for RANDOM2

<table>
<thead>
<tr>
<th>Physical Quantity</th>
<th>Theory Symbol</th>
<th>FORTRAN Name</th>
<th>SI Units</th>
<th>U.S. Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Material Property</td>
<td>m</td>
<td>RMM</td>
<td>m/cycle/M Pa</td>
<td>m/in/cycle/ksi/in</td>
</tr>
<tr>
<td>Initial Crack Size</td>
<td>A&lt;sub&gt;i&lt;/sub&gt;</td>
<td>RAI</td>
<td>m</td>
<td>in</td>
</tr>
<tr>
<td>Material Property</td>
<td>C</td>
<td>RCC</td>
<td>m/cycle</td>
<td>in/cycle</td>
</tr>
<tr>
<td>Alternating Stress</td>
<td>Δσ</td>
<td>DELSIG</td>
<td>M Pa</td>
<td>ksi</td>
</tr>
<tr>
<td>Final Crack Size</td>
<td>A&lt;sub&gt;f&lt;/sub&gt;</td>
<td>AF</td>
<td>m</td>
<td>in</td>
</tr>
<tr>
<td>Geometry Dependent Constant</td>
<td>Y</td>
<td>YY</td>
<td>(dimensionless)</td>
<td></td>
</tr>
</tbody>
</table>
7.0 APPENDIX B

SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUT FILES
CALL RNSET(ISEED)
CALL RN4(NTOT,YM+RCC)
WRITE(*,202)

2021 FORMAT(*,1840,1841,1842,1843)
WRITE(*,2021)

C LOGNORMAL STRESS RANGE—DELSIG
WRITE(*,2022)
READ(S+101),XN
WRITE(S+101),XN

C XN=0.62E+02
XN=0.62E+02

C YM = LOG(XM) - 0.5*YS**2
CALL RNSET(ISEED)
CALL RN4(NTOT,YM+RCC)
WRITE(6+2023)

2022 FORMAT(*,1840,1841,1842)
WRITE(*,2022)

C DEFINE DETERMINISTIC PARAMETERS
C PI = 3.1415926535897932384626433
C COMPONENT AND CRACK SHAPE PARAMETER, Y*

C FINAL CRACK SIZE, "AF"
AF=2.0E-03
C CALCULATE CYCLES TO REACH_CRACK_SIZE 2.0E-03M
GO TO 100
101 CONINUE

2033 FORMAT(*,1840,1841,1842,1843)
WRITE(*,2033)

C GIVEN STRESS AMPLITUDE=6.25E+02MPA"
WRITE(6+2034)

C UNIFORM LOAD CYCLES
C WRITE(*,2035)

2043 FORMAT(*,1840,1841,1842,1843)
WRITE(*,2043)

C XN=0.62E+02
XN=0.62E+02

C CALCULATE LOG OF CYCLES TO REACH CRACK SIZE 2.0E-03M

102 CONTINUE

2024 FORMAT(*,1840,1841,1842,1843)
WRITE(*,2024)

C CALCULATE PDF OF LOG OF CURRENT CYCLES,LOG XNF

105 CONTINUE

2054 FORMAT(*,1840,1841,1842,1843)
WRITE(*,2054)

C CALL DESPL PARAMETERS"

985 FORMAT(*,1840,1841,1842,1843)
WRITE(*,985)

C INMISS"

979 FORMAT(*,1840,1841,1842,1843)
WRITE(*,979)
WRITE(6,1010)MMISS

CALCULATE WINDOW WIDTH, HH

HH = (BNDS(2) - BNDS(1)) / (NODE - 1)

CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED

ALSO CALLED "NODE" VALUES

DO 6001 I = 1, NODE - 2
    BNDS(I + 1) = BNDS(I) + (I * HH)
  6001 CONTINUE

WRITE(6,983)

983 FORMAT(' LOG OF CURRENT CYCLES, LOG_XNF')

WRITE(6,1001)(BNDS(I), I = 1, NODE)

REORDER BNDS FOR PLOTTING

SAVE1 = BNDS(1)
SAVE2 = BNDS(NODE)
BNDS(NODE) = BNDS(2)
BNDS(2) = BNDS(1)
BNDS(1) = SAVE1
BNDS(NODE - 1) = SAVE2

6002 CONTINUE

984 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG_XNF,' &
          ' IX AXIS PDF, CDF PLOT')

WRITE(6,1003)(BNDS(I), I = 1, NODE)

WRITE(6,1002) (BNDS(I), I = 1, NODE)

WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
LOG XNF TO PLOT FILES

990 FORMAT('E12.4,1X,E12.4/')

WRITE(34,990)

991 FORMAT('E12.4,1X,E12.4')

CALCULATE CDF OF LOG OF CURRENT CYCLES

READ(5,1010)IOPT
WRITE(6,992)

992 FORMAT(' CDF PARAMETERS')
WRITE(6,1010)IOPT
X0 = BNDS(1)
DO 6003 I = 1, NODE
   P = GCDF(X0, IOPT, NODE, BNDS, DENS)
   BNDSX(I) = X0
   X0 = X0 + HH
   DISTX(I) = P
  6003 CONTINUE

WRITE(6,994)

994 FORMAT(' CDF OF LOG OF CURRENT CYCLES, LOG_XNF,' &
          ' IY AXIS PDF, CDF PLOT')
WRITE(6,1001)(DISTX(I), I = 1, NODE)

WRITE(6,993)

993 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG_XNF,' &
          ' IY AXIS PDF, CDF PLOT')
WRITE(6,1001)(BNDS(I), I = 1, NODE)
WRITE(6,1001)(BNDSX(I), I = 1, NODE)

WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT
TO THE PLOT FILES
DENEST - NODE by Jmatrix containing the gradient vector, among
other quantities, (Output)
B - Vector of length NODE containing the NODE values, 
   (Output)
IPUT - Pivot vector of length NODE-2, (Output).
WK2 - Work vector of length NODE-2, (Output).

Chapter: STAT/LIBRARY Density and Hazard Estimation

Copyright: 1985 by IMSL, Inc. All Rights Reserved.
Warranty: IMSL warrants only that IMSL testing has been applied
to this code. No other warranty, expressed or implied,
is applicable.

SUBROUTINE D3PSL (NORS, X, NODE, BNOS, INIT, ALPHA, MAXIT, EPS,
DEN, STAT, HESS, LDHESS, ILIOI, DENEST, B,
IPUT, WK2)
!
INTEGER NORS, NODE, INIT, MAXIT, LDHESS, ILIOI(NODE)!
REAL IPUT(*), ALPHA, EPS, X(*), BNOS(2), DEN(*), STAT(*),
HESS(LDHESS), DENEST(NODE), B(*), WK2(*)
!
INTEGER I, IMPR, IT, K, KM1, KM2, KPI, KP2, M, M1
!
REAL NER, BN1, BSN2, BSML, CK, CKM1, CKM2, CKMCH1, CKP1, CKP2,
CONS, EPS1, FACTOR, FK, FRM1, FRM2, FKPI, H, H2, H3,
SUM, TEMP, WK14,
!
DOUBLE PRECISION SUM1, SUM2, SUM3
!
INTEGER MINSK(8), SAVE MINSK

SPECIFICATIONS FOR LOCAL VARIABLES

SPECIFICATIONS FOR INTRINSICS

SPECIFICATIONS FOR SUBROUTINES

SPECIFICATIONS FOR FUNCTIONS

DATA MINSK/5, 9, 17, 33, 65, 129, 253, 100001/

CALL EIPSH ('D3PSL ')

Error checks
IF (NORS .LT. 1) THEN
   CALL EIMES (5, 1, 'After removing all missing (NaN) not a
   'number values from X there are no valid 
   'observations. At least one valid observation is 
   'necessary. ')
END IF

IF (NODE .LE. 4) THEN
   CALL EISTI (1, NODE)
   CALL EIMES (5, 2, 'NODE = %d, The number of mesh
   'nodes, NODE, must be an odd integer greater ')

'S21  '
M = 3
ELSE
M = NODE
END IF
C 20 IF (INIT .EQ. 0) THEN
    H = (BNDS(2) - BNDS(1)) / (M - 1)
    END IF
C
H2 = H2 * H
C
H3 = H3 * H
C
IF (INIT .NE. 0) THEN
    CALL SSCAL (NODE, 1.0 / (H * SUM(NODE, DENS, 1)), DENS, 1)
    END IF
C
B(1) = BNDS(1)
DO 30 I = 2, M
    B(I) = B(I-1) + H
    30 CONTINUE
C
B(1) = BNDS(1)
DO 40 I = 2, M
    B(I) = B(I-1) + H
    40 CONTINUE
C
IPT = 0
IF (X(IPT) .LT. BNDS(1)) GO TO 40
DO 50 K = 1, M - 1
    LM = ILMO(K) = IPT
    IF (X(IPT) .LT. B(K+1)) THEN
        LM = ILMO(K+1) = ILMO(K) + 1
    END IF
    IF (IPT + LE. NOBS) GO TO 50
    50 CONTINUE
C
IPT = 0
IF (X(IPT) .LT. BNDS(1)) GO TO 40
DO 60 K = 1, M - 1
    LM = ILMO(K) = IPT
    IF (X(IPT) .LT. B(K+1)) THEN
        LM = ILMO(K+1) = ILMO(K) + 1
    END IF
    IF (IPT + LE. NOBS) GO TO 60
    60 CONTINUE
C
FACT = 2.0 * ALPHA / H3
C
IF (INIT .EQ. 0) THEN
    CALL D2SPT (H-2, B(2), 1, MLOD, BNDS, DENS, DENSEST, WK, WK)
    TEMP = 1.0 / (M * M * M)
    DO 80 K = 1, M - 1
        DENS(K) = MAX1(TEMP, SUM(DENSEST(K+1), 1))
        80 CONTINUE
    ELSE
        DO 90 K = 1, M - 1
            DENS(K) = SUM(DENS(K))
            90 CONTINUE
    END IF
C
DENS(M) = 0.0
C
DO 100 ITER = 1, MAXIT
    CALL 1
    MAXIMIZE
C
HESS(1,1) = 0.0
HESS(1,2) = 0.0
HESS(2,2) = 0.0
CALL 1
SUM = 0.0
C
CK** are true estimates = FK**2
DO 120 K=2, M-1
   KM1 = K - 1
   KM2 = MAX0(1, K - 2)
   KP1 = K + 1
   KP2 = MIN0(K + 1, M - K + 2)
   FKM1 = DENS(KM1)
   FKM2 = DENS(KM2)
   CKM1 = FKM1**2
   CKM2 = FKM2**2
   CK = FKM1**2
   BK = B(K)
   BM1 = B(KM1)
   SUM = SUM + CK
   IF (K .GE. 4) HESS(1, KM1) = 4.0*FK*FKM2*FACTOR
   SUM1 = 0.0
   SUM2 = 0.0
   SUM3 = 0.0
   DO 100 I=1:ILOH1(KM1), ILOH1(KM1+2)
      TEMP = (X(I) - BK)/H
      CONS = (1.0 - TEMP)/(CK + (CKP1 - CK)*TEMP)
      SUM1 = SUM1 + CONS
      SUM2 = SUM2 + CONS*CONS
500 CONTINUE
   100 SUM = SUM1 + CONS*CONS
   SUMM = SUM2 + CONS*CONS/TMP
   TEMP = TEMP*TEMP
   BSMALL = BSMALL + 2.0*CONS/CONS
   HESS(I, KM1) = TEMP + 4.0*CONS*CONS*CONS*CONS + SUM2
   IF (K .GE. 2) HESS(2, KM1) = 4.0*FK*FKM1*(-4.0*CONS*CONS + SUM3)
   DE'NESt(KM1) = FK*CONS
   DE'NESt(KM1+2) = -2.0*CONS
   DO 120 I=1:ILOH1(KM1+1), ILOH1(KM1+2)
      TEMP = TEMP*CONS
      BSMALL = BSMALL + 2.0*CONS/CONS
      HESS(I, KM1) = TEMP + 4.0*CONS*CONS*CONS + SUM2
      IF (K .GE. 2) HESS(2, KM2) = 4.0*FK*FKM1*(-4.0*CONS*CONS + SUM3)
500 CONTINUE
   120 CONTINUE
    BSMALL = 1.0/H - SUM + BSMALL
    CALL SCOPY (M-2, DE'NESt(1, K-1), DE'NESt(1, K-1))
    CALL SCADD (M-2, -BSMALL/(2.0*SUM), HESS(3, 1), LDOMESs)
    CALL SCOPY (M-4, HESS(3, 1), LDOMESs, HESS(3, 1), LDOMESs)
    HESS(3, M-3) = 0.0
    HESS(3, M-2) = 0.0
    CALL SCOPY (M-3, HESS(2, 2), LDOMESs, HESS(4, 1), LDOMESs)
    HESS(4, M-5) = 0.0
    CALL L2TRB (M-2, HESS, LDOMESs, 2, 2, HESS, LDOMESs, IPVT, WK2)
    CALL LFSRB (M-2, HESS, LDOMESs, 2, 2, IPVT, DE'NEst(1, K-1))
    CALL LFSRB (M-2, HESS, LDOMESs, 2, 2, IPVT, DE'NESt(1, K-1))
    IF (NIRCD(1) .NE. 0) GO TO 9000
    CONS = SBOT(M-2, DE'NEst(1, K-1), DE'NEst(1, K-1))
    CONS = (1.0/H - SUM - SBOT(M-2, DE'NEst(1, K-1), 1) + DE'NEst(1, K-1)) / CONS
    END
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>User</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>11/29/88</td>
<td>17:04</td>
<td>US</td>
<td>CRAY X-MP/24 S/N-130/89/26 Austin, Texas</td>
</tr>
<tr>
<td></td>
<td>17:05</td>
<td>US</td>
<td>CRAY Operating System COS 1.1.6 BF2 Assembled</td>
</tr>
<tr>
<td></td>
<td>04/30</td>
<td>US</td>
<td>10/24/88</td>
</tr>
</tbody>
</table>

** JOB: JH-RA0822 US-USA0530 RT=60 T=30 MFL=3000000.**

**Job Name: RANDM22.**

**libraries:**

- **lib:*mslib**
- **lib:**mslib

**Id:** STATIC

**Ed:** 4

**Own:** LIBRARY

**Files:**

- **VAX:** CRAY: FILE=**USA0530.RANDOM2.NORMAL.INP:** 12

**DATASET RECEIVED FROM FRONT END:**

- **ASSIGN:** DN=AA.AA

**USER:**

- **USER:** NAME=RANDM20

**User Number:**

- **User Number:** USA0530

**Job Sequence Number:**

- **Job Sequence Number:** 4788

**Time Executing in CPU:**

- **Time Executing in CPU:** 00:00:10:07 6742

**Time Waiting to Execute:**

- **Time Waiting to Execute:** 00:00:41 9159

**Time Waiting Semaphore:**

- **Time Waiting Semaphore:** 00:00:00 0000

**Input Queue:**

- **Input Queue:** 00:00 0000

**Memory CPU Time (MIDSEC):**

- **Memory CPU Time (MIDSEC):** 1.79942

**Memory Semaphore Time (MIDSEC):**

- **Memory Semaphore Time (MIDSEC):** 1.81683

**Minimum Job Size (Words):**

- **Minimum Job Size (Words):** 52736

**Minimum FL (Words):**

- **Minimum FL (Words):** 47520

**Maximum FL (Words):**

- **Maximum FL (Words):** 466432

**Minimum JTA (Words):**

- **Minimum JTA (Words):** 4096

**Maximum JTA (Words):**

- **Maximum JTA (Words):** 5632

**PSS SECTORS MOVED:**

- **PSS SECTORS MOVED:** 619

**User I/O Requests:**

- **User I/O Requests:** 467

**User Suspensions:**

- **User Suspensions:** 12 1632

**Open Calls:**

- **Open Calls:** 258

**Memory Resident Datasets:**

- **Memory Resident Datasets:** 0
8.0 APPENDIX C
IMSL SUBROUTINE CALLS FROM RANDOM2

1. RNSET  - Initiates a random seed for use in the IMSL random number generators.
2. RNNOR  - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.
3. RNLN L - Generates pseudorandom numbers from a lognormal distribution.
4. DESPL  - Performs nonparametric probability density function estimation by the penalized likelihood method.
5. GCDF  - Evaluates a general continuous cumulative distribution function given ordinates of the density.
9.0 APPENDIX D

SAMPLE SAS/GRAPH (VER. 5.16) PROGRAM FOR RANDOM2

data a;
INFILE 'OUT1.CPR' FIRSTOBS=2; input x y;
GOPTIONS DEVICE=HP7470;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
          value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'PROBABILITY DENSITY FUNCTION';
   symbol i=spline v=square;
data B;
INFILE 'OUT2.CPR' FIRSTOBS=2; input x y;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
          value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'CUMULATIVE DISTRIBUTION FUNCTION';
   symbol i=spline v=square;
6.0 APPENDIX 2

FATIGUE STRENGTH DEGRADATION MODEL:
RANDOM3 AND RANDOM4 USER MANUAL
FATIGUE STRENGTH REDUCTION MODEL:
RANDOM3 and RANDOM4 USER MANUAL

Prepared by:

Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 2
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989
<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>2.0 Theoretical Background</td>
<td>2</td>
</tr>
<tr>
<td>3.0 Input Data</td>
<td>4</td>
</tr>
<tr>
<td>4.0 Sample Problem</td>
<td>7</td>
</tr>
<tr>
<td>5.0 References</td>
<td>11</td>
</tr>
<tr>
<td>6.0 Appendix A: Physical Quantities, Symbols, and Units</td>
<td>12</td>
</tr>
<tr>
<td>7.0 Appendix B: RANDOM3 Sample Problem: Source, Input and Output Files</td>
<td>13</td>
</tr>
<tr>
<td>8.0 Appendix C: RANDOM4 Sample Problem Source, Input and Output Files</td>
<td>37</td>
</tr>
<tr>
<td>9.0 Appendix D: IMSL Subroutine Calls from RANDOM3 and RANDOM4</td>
<td>76</td>
</tr>
<tr>
<td>10.0 Appendix E: SAS/GRAPH Program for RANDOM3 and RANDOM4</td>
<td>77</td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

This User Manual documents the FORTRAN programs RANDOM3 and RANDOM4. They are based on fatigue strength reduction, using a probabilistic constitutive model. They predict the random lifetime of an engine component to reach a given fatigue strength (see Section 2.0, Theoretical Background).

Included in this Manual are details regarding the theoretical backgrounds of RANDOM3 and RANDOM4, input data instructions and sample problems illustrating the use of RANDOM3 and RANDOM4. Appendix A gives information on the physical quantities, their symbols, FORTRAN names and both SI and U.S. Customary units. Appendix B and C include photocopies of the actual computer printout corresponding to the sample problems. Appendices D and E detail the IMSL, Version 10 ¹, subroutines and functions called by RANDOM3 and RANDOM4 and SAS/GRAPH ² programs that can be used to plot both the probability density functions (p.d.f.) and the cumulative distribution functions (c.d.f.).
2.0 THEORETICAL BACKGROUND

Fatigue strength data are usually presented as cycles to failure for each of several stress amplitudes, the familiar S-N diagram. Results indicate that for lower stress amplitudes the cycles (or time) to failure increases. Thus, a power curve fit through the data yields a monotonically decreasing curve. In general, this curve is represented as

\[ S = \left(\frac{N}{C}\right)^{-1/m'} \]  

(6)

where the primitive variables in this equation are as follows: \( S \) is the applied constant amplitude alternating stress at failure or fatigue strength, \( N \) is number of cycles, \( C \) is a material parameter that varies from specimen to specimen and \( m' \) is a material constant. Equation (6) can be written in terms of "cycles to reach a given fatigue strength" as

\[ N = CS^{-m'} \]  

(7)

Recently another fatigue strength reduction model has been proposed that takes into account the effect of temperature as well as other parameters that affect strength. The general form of the constitutive relationships for this model is applied to the constituents of high temperature composite materials. Specifically, it is applied herein for the case of a single material constituent. The mechanical property of interest is fatigue strength which is expressed in terms of primitive variables, including the general categories of temperature, mechanical cycles and mean stress. For these categories, the relationship becomes

\[
\frac{S}{S_0} = \left[ \frac{T_F - T}{T_F - T_0} \right]^n \left[ \frac{S_F - \sigma}{S_F - \sigma_0} \right]^{m} \left[ \frac{\log N_{MF} - \log N_{MO}}{\log N_{MF} - \log N_{MO}} \right]^q
\]

(8)

where \( S \) is the applied constant amplitude alternating stress at failure (fatigue strength) at current (or operating) temperature, \( T \), mean stress, \( \sigma \), and mechanical cycle, \( N_M \). \( S_0 \) is fatigue strength at reference temperature, \( T_0 \) (usually room temperature), reference mean stress (or residual stress), \( \sigma_0 \), and reference mechanical cycle, \( N_{MO} \). Also, \( T_F \) is the final or melting temperature of the material, \( S_F \) is the final or tensile strength of the material, and \( N_{MF} \) is the final mechanical cycle or lifetime. Empirical parameters, \( n, m, \) and \( q \), are determined from available experimental data or estimated from anticipated behavior of the particular product term. Note that the term containing mechanical cycles is expressed in terms of the log of cycles rather than cycles. This formulation is attractive when \( N_M \) and \( N_{MO} \) are small compared to \( N_{MF} \). The equation may be solved for \( N_M \), or the "cycles to reach a given fatigue strength." The expression is

\[
N = 10 \exp \left[ \log N_{MF} \left( \log N_{MF} - \log N_{MO} \right) \right] \left[ \frac{S}{S_0} \left[ \frac{T_F - T}{T_F - T_0} \right]^n \left[ \frac{S_F - \sigma}{S_F - \sigma_0} \right]^{m} \right]^{1/q}\]

(9)
For values typical of a cast nickel base-superalloy subjected to typical loads and temperatures, equation (9) indicates increasing life for decreasing temperature, decreasing tensile mean stress, and decreasing applied alternating stress. It indicates decreasing life for increasing temperature, decreasing compressive mean stress, and increasing applied alternating stress. Therefore, equation (9) predicts observed trends in general.

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method for RANDOM3. For RANDOM4, a p.d.f. of cycles is generated using the maximum entropy method. Maximum entropy uses Jaynes' principle which says that "the minimally prejudiced distribution is that which maximizes the entropy subjected to the constraints supplied by the given information."

52
3.0 INPUT DATA

Data input for RANDOM3 and RANDOM4 is user friendly and easy to manipulate (see, for example, the file entitled NORMAL.INP, in Section 4.0). The first twelve lines of input have the same format, 2E12.4 and the last two lines differ. The last two lines of input have the formats I3,2X,I3,2X,2E12.4,2X,I3 and I3, respectively. A brief, line by line description is given along with an example for each line (NOTE: the ruler is to aid the user in formatting and is not a part of the input). A table listing the physical quantities, their units and symbols is given in Appendix A.

1. Random Number Generator Seed, ISEED, and Sample Size, NTOT

EXAMPLE:

123456789012345678901234567890

1 40

2. Ultimate Tensile Strength, SF

EXAMPLE:

123456789012345678901234567890

900.0000 45.0000

3. Log of Final Cycle, NMF

EXAMPLE:

123456789012345678901234567890

8.0000 0.8000

4. Reference Fatigue Strength, SO

EXAMPLE:

123456789012345678901234567890

500.0000 25.0000

5. Log of Reference Cycle, NMO

EXAMPLE:

123456789012345678901234567890

7.0000 0.7000

53
6. Current Fatigue Strength, S

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
250.0000 & 12.0000
\end{array}
\]

7. Residual Compressive Stress, SIGO

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
20.0000 & 1.0000
\end{array}
\]

8. Current Mean Stress, SIG

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
150.0000 & 7.5000
\end{array}
\]

9. Temperature Exponent, XXN, Stress Exponent, XXM, and Cycle Exponent, XXQ

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
0.5000 & 0.0150
\end{array}
\]

10. Melting Temperature, TF

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
1500.0000 & 75.0000
\end{array}
\]

11. Reference Temperature, TO

EXAMPLE:

\[
\begin{array}{cc}
123456789012345678901234567890 \\
20.0000 & 0.6000
\end{array}
\]
12. Current Temperature, T

EXAMPLE:

\[ \begin{array}{cc}
123456789012345678901234567890 \\
850.0000 & 25.0000 \\
\end{array} \]

13. The DESPL \(^1\) parameters are NODE, INIT, ALPHA, EPS, and MAXIT and are entered in that order as follows:

EXAMPLE:

\[ \begin{array}{cccccc}
1234567890123456789012345678901234567890 \\
21 & 0 & 20.0000 & 1.0E-05 & 30 \\
\end{array} \]

14. The DESPL parameter, IOPT, is entered as follows:

EXAMPLE:

\[ \begin{array}{c}
1234567890 \\
2 \\
\end{array} \]
4.0 SAMPLE PROBLEMS FOR RANDOM3 AND RANDOM4

The objective of these programs is to predict the random lifetime to reach a given fatigue strength for an engine component. The theory is based on fatigue strength reduction, using a probabilistic constitutive model. The only difference between RANDOM3 and RANDOM4 is the method used to generate p.d.f. estimates. RANDOM3 uses maximum penalized likelihood, while RANDOM4 uses maximum entropy (see Section 2.0, Theoretical Background). RANDOM3 and RANDOM4 input parameters are given in Table A2.1.

### TABLE A2.1 RANDOM3 and RANDOM4 input (SI units)

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Distribution Type</th>
<th>Mean</th>
<th>Standard Deviation (Value)</th>
<th>Standard Deviation (% of Mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SF</td>
<td>normal</td>
<td>900.0</td>
<td>45.0</td>
<td>(3%)</td>
</tr>
<tr>
<td>NMF</td>
<td>lognormal</td>
<td>8.0</td>
<td>0.8</td>
<td>(10%)</td>
</tr>
<tr>
<td>SO</td>
<td>lognormal</td>
<td>500.0</td>
<td>25.0</td>
<td>(5%)</td>
</tr>
<tr>
<td>NMO</td>
<td>lognormal</td>
<td>7.0</td>
<td>0.7</td>
<td>(10%)</td>
</tr>
<tr>
<td>S</td>
<td>lognormal</td>
<td>250.0</td>
<td>12.5</td>
<td>(5%)</td>
</tr>
<tr>
<td>SIGO</td>
<td>lognormal</td>
<td>-20.0</td>
<td>-1.0</td>
<td>(1%)</td>
</tr>
<tr>
<td>SIG</td>
<td>lognormal</td>
<td>150.0</td>
<td>7.5</td>
<td>(5%)</td>
</tr>
<tr>
<td>XXN</td>
<td>normal</td>
<td>0.5</td>
<td>0.015</td>
<td>(0.3%)</td>
</tr>
<tr>
<td>XXM</td>
<td>normal</td>
<td>0.5</td>
<td>0.015</td>
<td>(0.3%)</td>
</tr>
<tr>
<td>XXQ</td>
<td>normal</td>
<td>0.5</td>
<td>0.015</td>
<td>(0.3%)</td>
</tr>
<tr>
<td>TF</td>
<td>normal</td>
<td>1500.0</td>
<td>45.0</td>
<td>(3%)</td>
</tr>
<tr>
<td>TO</td>
<td>normal</td>
<td>20.0</td>
<td>0.6</td>
<td>(3%)</td>
</tr>
<tr>
<td>T</td>
<td>normal</td>
<td>850.0</td>
<td>25.5</td>
<td>(3%)</td>
</tr>
</tbody>
</table>
The input is entered in the following format in a file entitled NORMAL.INP.

```
1234567890123456789012345678901234567890
1  40
  900.0000 45.0000
   8.0000   0.8000
 500.0000 25.0000
   7.0000   0.7000
 250.0000 12.5000
  20.0000   1.0000
 150.0000   7.5000
  0.5000   0.0150
1500.0000  75.0000
 20.0000   0.6000
 850.0000 25.5000
  21   0   20.00  1.0E-05  30
  2
```

Execution of RANDOM3 and RANDOM4 (source code entitled NR3.FOR and NR4.FOR, respectively) produces files entitled RANDM33 and RANDM44. These give intermediate results (see Appendices B and C). Execution also produces plotfiles entitled PLOT1 and PLOT2 (see Appendices B and C). These files are used to plot the X and Y axes of the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.), respectively, generated by RANDOM3 and RANDOM4. The plots are drawn from the plotfiles by the SAS/GRAPH graphing program (see Appendix D). These plots for the sample problem are shown Figures 1, 2, 3, and 4. This same sample problem has been reported in Boyce and Chamis.7 There, however, it utilized U.S. Customary units and older versions of RANDOM3 and RANDOM4 (using IMSL Version 9.2 subroutines).
Fig. A2.1  p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.

Fig. A2.2  c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.
Fig. A2.3  p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.

Fig. A2.4  c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.
5.0 REFERENCES

1 IMSL, "STAT/LIBRARY, FORTRAN Subroutines for Statistical Analysis", Houston, Texas


### 6.0 APPENDIX A

**PHYSICAL QUANTITIES, SYMBOLS, AND UNITS**

The physical quantities, their symbols and units for the fatigue crack growth model are given in the following table.

Table A2.2 Physical quantities, symbols, and units for fatigue crack growth model for RANDOM3 and RANDOM4.

<table>
<thead>
<tr>
<th>Physical Quantity</th>
<th>Theory Symbol</th>
<th>FORTRAN Symbol</th>
<th>Units SI</th>
<th>Units U.S.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ultimate Tensile Strength</td>
<td>SF</td>
<td>SF</td>
<td>MPa</td>
<td>ksi</td>
</tr>
<tr>
<td>Final Cycle (lifetime)</td>
<td>N_MF</td>
<td>NMF</td>
<td>dimensionless</td>
<td></td>
</tr>
<tr>
<td>Reference Fatigue Strength</td>
<td>S0</td>
<td>SO</td>
<td>MPa</td>
<td>ksi</td>
</tr>
<tr>
<td>Reference Cycles</td>
<td>N_MO</td>
<td>NMO</td>
<td>dimensionless</td>
<td></td>
</tr>
<tr>
<td>Current Fatigue Strengths</td>
<td>S</td>
<td>S</td>
<td>MPa</td>
<td>ksi</td>
</tr>
<tr>
<td>Residual Compressive Stress</td>
<td>σ₀</td>
<td>SIGO</td>
<td>MPa</td>
<td>ksi</td>
</tr>
<tr>
<td>Current Mean Stress</td>
<td>σ</td>
<td>SIG</td>
<td>MPa</td>
<td>ksi</td>
</tr>
<tr>
<td>Empirical Material Parameters</td>
<td>n</td>
<td>XXN</td>
<td>dimensionless</td>
<td></td>
</tr>
<tr>
<td></td>
<td>m</td>
<td>XXM</td>
<td>dimensionless</td>
<td></td>
</tr>
<tr>
<td></td>
<td>q</td>
<td>XXQ</td>
<td>dimensionless</td>
<td></td>
</tr>
<tr>
<td>Melting Temperature</td>
<td>TF</td>
<td>TF</td>
<td>°C</td>
<td>°F</td>
</tr>
<tr>
<td>Reference Temperature</td>
<td>TO</td>
<td>TO</td>
<td>°C</td>
<td>°F</td>
</tr>
<tr>
<td>Current Temperature</td>
<td>T</td>
<td>T</td>
<td>°C</td>
<td>°F</td>
</tr>
</tbody>
</table>
7.0 APPENDIX B

RANDOM3 SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUT FILES
C CHAMIS MICROMECHANICS CONSTITUTIVE EQUATIONS
C RANDOMIZED AND APPLIED TO FATIGUE STRENGTH
    INTEGER NTOT, ISEED, M, INT, NMISS, MAXIT, NODE
    REAL XM, XS, YM, YS, EPS, P, REN3F(7*12), ALPHAN
    COMMON /URASP/ RWKSP
    DIMENSION SF(10000), XLMF(10000), SO(10000)
    DIMENSION XLMO(10000), S(10000)
    DIMENSION SIGD(10000), SIG(10000)
    DIMENSION XM(10000), YXM(10000), XQ(10000)
    DIMENSION X(10000), BDNSX(10000)
    DIMENSION TF(10000), TO(10000), T(10000)
    DIMENSION STAT(10000), BENS(10000), BISTX(10000)
    DIMENSION BDNS(10000), BB(999), FF(999)
    DIMENSION XX(999), PP(999)
    DIMENSION XXXX(999), PPPPP(999)
    DIMENSION BBP(999), FFFF(999)
    DIMENSION C(10000)
    EXTERNAL RNLNL, RNSET, RNNOR, DESPL, IWKIN

1001 FORMAT(5E12.4)
1002 FORMAT(I12)
1003 FORMAT(I4)
1004 FORMAT(I4)
1009 FORMAT(I3, 2X, I3, 2X, I2, 12, 2, 2, 2, 2)
1010 FORMAT(I3)
1011 FORMAT(I3, 2, 2, 2, 2)

C LOGNORMAL ULTIMATE TENSILE STRENGTH-SF
READ (3, 1002) ISEED, NTOT
WRITE (4, 1002) ISEED, NTOT
READ (3, 1001) XM, XS
WRITE (6, 1001) XM, XS
YS = SQRT(LOG(1.0 + (XS/XM)**2.))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLNL(NTOT, YM, YS, SF)
WRITE (6, 2001)
2001 FORMAT(5E12.4)

C LOGNORMAL LOG OF FINAL CYCLE, XLMF
WRITE (6, 1002) ISEED, NTOT
READ (3, 1001) XM, XS
WRITE (6, 1001) XM, XS
YS = SQRT(LOG(1.0 + (XS/XM)**2.))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLNL(NTOT, YM, YS, XLMF)
WRITE (4, 2001)
2001 FORMAT(5E12.4)

C LOGNORMAL XLMF AT REFERENCE CONDITIONS, SO
WRITE (4, 1002) ISEED, NTOT
READ (3, 1001) XM, XS
WRITE (6, 1001) XM, XS
YS = SQRT(LOG(1.0 + (XS/XM)**2.))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLMN(NTOT+1,M,YS+1)
WRITE(6,2022)
2022 FORMAT(' LOGNORMAL SO')
WRITE(6,1001) (SDI(I),I=1,NTOT)
C LOGNORMAL LOG OF REFERENCE CYCLES, XLMO
WRITE(6,1002) (ISEED,NTOT)
READ(5,1011) XM, XS
WRITE(6,1011) XM, XS
YM = LOG(XM) - 0.5*YS**2
CALL RNSET(ISEED)
CALL RNLMN(NTOT,YM,YS,XLMO)
WRITE(6,2023)
2023 FORMAT(' LOGNORMAL XLMO')
WRITE(6,1001)(XLMO(I),I=1,NTOT)
C LOGNORMAL FATIGUE STRENGTH AT CURRENT CONDITIONS, S
WRITE(6,1002) ISEED,NTOT
READ(3,1011) XM, XS
WRITE(6,1011) XM, XS
YS = SORT(LOG(1.0+(XS/XM)**2))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLMN(NTOT,YM,YS,S)
WRITE(6,2024)
2024 FORMAT(' LOGNORMAL S')
WRITE(6,1001)(S(I),I=1,NTOT)
C DEFINE RANDOM STRESSES
C LOGNORMAL REFERENCE STRESSES, SIGO
WRITE(6,1002) ISEED,NTOT
READ(3,1011) XM, XS
WRITE(6,1011) XM, XS
YS = SORT(LOG(1.0+(XS/XM)**2))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLMN(NTOT,YM,YS,SIGO)
C CHANGE SIGO TO NEGATIVE VALUES FOR COMPRESSIVE
C RESIDUAL STRESSES
DO 201 I = 1,NTOT
SIGO(I) = -SIGO(I)
201 CONTINUE
2036 FORMAT(' LOGNORMAL SIGO')
WRITE(6,1001) (SIGO(I),I=1,NTOT)
C LOGNORMAL CURRENT STRESSES, SIG
WRITE(6,1002) ISEED,NTOT
READ(3,1011) XM, XS
WRITE(6,1011) XM, XS
YS = SORT(LOG(1.0+(XS/XM)**2))
YM = LOG(XM) - 0.5*YS**2.
CALL RNSET(ISEED)
CALL RNLMN(NTOT,YM,YS,SIG)
WRITE(6,2037)
2037 FORMAT(' LOGNORMAL SIG')
WRITE(6,1001)(SIG(I),I=1,NTOT)
C NORMAL EXPONENTS, XXN, XNM, XXU
WRITE(6,1002) ISEED,NTOT
READ(3,1011) YM, YS
WRITE(6,1011) YM, YS
CALL RNSET(ISEED)
CALL RNLMN(NTOT,XXN)
DO 202 I=1,NTOT
XXN(I) = XXN(I)+YM
202 CONTINUE
WRITE(6,2025)
2025 FORMAT(' NORMAL XXM'), WRITE(*,1001)(XXM(I),I=1,NTOT) WRITE(*,1002)ISEED,NTOT CALL RNSET(ISEED) CALL RNNO(NTOT,XXM) DO 203 I=1,NTOT XXM(I)=YS**XXM(I)+YM 203 CONTINUE WRITE(*,2026) 2026 FORMAT(' NORMAL XXM'), WRITE(*,1001)(XXM(I),I=1,NTOT) WRITE(*,1002)ISEED,NTOT CALL RNSET(ISEED) CALL RNNO(NTOT,XXM) DO 204 I=1,NTOT XXM(I)=YS**XXM(I)+YM 204 CONTINUE WRITE(*,2027) 2027 FORMAT(' NORMAL XXD'), WRITE(*,1001)(XXD(I),I=1,NTOT) C NORMAL TEMPERATURES, TF,TD,I C NORMAL FINAL (MELTING) TEMPERATURE, TF WRITE(*,1002)ISEED,NTOT READ(3,1011) YM,YS WRITE(*,1011) YM,YS CALL RNSET(ISEED) CALL RNNO(NTOT,TF) DO 205 I=1,NTOT TF(I)=YS**TF(I)+YM 205 CONTINUE WRITE(*,2046) 2046 FORMAT(' NORMAL TF'), WRITE(*,1001)(TF(I),I=1,NTOT) C NORMAL REFERENCE TEMPERATURE, TO WRITE(*,1002)ISEED,NTOT READ(3,1011) YM,YS WRITE(*,1011) YM,YS CALL RNSET(ISEED) CALL RNNO(NTOT,TO) DO 206 I=1,NTOT TO(I)=YS**TO(I)+YM 206 CONTINUE WRITE(*,2047) 2047 FORMAT(' NORMAL TO'), WRITE(*,1001)(TO(I),I=1,NTOT) C NORMAL CURRENT TEMPERATURE,T C CALL RNNO(NTOT,T) READ(3,1011) YM,YS WRITE(*,1011) YM,YS CALL RNSET(ISEED) CALL RNNO(NTOT,T) DO 207 I=1,NTOT T(I)=YS**T(I)+YM 207 CONTINUE WRITE(*,2048) 2048 FORMAT(' NORMAL T'), WRITE(*,1001)(T(I),I=1,NTOT) C CALCULATE LOG OF CURRENT CYCLES, LOG XNM DO 208 I=1,NTOT LOG(XNM(I))=(T(I)-SIG(I))/(TF(I)-SIG(I))**XXM(I) XNM(I)=(T(I)/SIG(I)**LOG(I))**(1./XXM(I)) XNM(2)=(XNM(I)-(XLMN+XNM0(I))**XXM(I)) IF(XNM(I).LT.0.0)XNM2=0.0


102 CONTINUE
WRITE(6,2028)
2028 FORMAT('LOG OF CYCLES TO REACH MEAN FATIGUE STR = ')
WRITE(6,1001)(XNM(I),I=1,NTOT)
C SORT LOG OF CYCLES
CALL SORT(XNM,NTOT)
WRITE(6,2029)
2029 FORMAT('SORTED LOG OF CYCLES')
WRITE(6,1001)(XNM(I),I=1,NTOT)
C CALCULATE PDF OF LOG OF CURRENT CYCLES; LOG XNM
READ(3,1009)NODE,INIT,ALPHA, EPS,MAIT
WRITE(6,985)
985 FORMAT('DEPL PARAMETERS')
WRITE(6,1009)NODE,INIT,ALPHA, EPS, MAIT
BND(1)=XNM(1)-0.05*XNM(1)
BND(2)=XNM(NTOT)+0.05*XNM(NTOT)
WRITE(6,979)BND(1),BND(2)
979 FORMAT('BND(1),BND(2)=E12.4*1X,E12.4')
CALL DEPL(NTOT,XNM,NODE,BND,INIT,ALPHA,MAIT,EPS, DENS,STAT,
1HMSS)
WRITE(6,980)
980 FORMAT('PDF OF LOG OF CURRENT CYCLES; LOG XNM; Y AXIS OF PDF PLOT')
WRITE(6,1001)(DENS(I),I=1,NODE)
WRITE(6,981)
981 FORMAT('OUTPUT STATISTICS')
WRITE(6,1001)(STAT(I),I=1,NODE)
WRITE(6,982)
982 FORMAT('NUMBER OF MISSING VALUES')
WRITE(6,1010)1HMSS
CALCULATE WINDOW WIDTH, NH
NH=(BND(2)-BND(1))/(NODE-1)
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED;
ALSO CALLED 'NODE' VALUES
DO 6001 I=1,NODE-2
BND(I+2)=BND(I)+NH
6001 CONTINUE
983 FORMAT('LOG OF CURRENT CYCLES, LOG XNM')
WRITE(6,1001)(BND(I),I=1,NODE)
C REORDER BND FOR PLOTTING
SAVE1 = BND(2)
SAVE2 = BND(NODE)
DO 6002 I=1,NODE-1
BND(I+1)=BND(I+2)
6002 CONTINUE
BND(NODE)=SAVE2
BND(NODE)=SAVE1
WRITE(6,984)
984 FORMAT('ORDERED LOG OF CURRENT CYCLES, LOG XNM')
WRITE(6,1001)(BND(I),I=1,NODE)
C WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES
C LOG XNM TO PLOT FILES
WRITE(34,990)
990 FORMAT('E12.4*1X,E12.4')
CALCULATE CDF OF LOG OF CURRENT CYCLES
READ(3,1010)IOPT
WRITE(6,992)
992 FORMAT('CDF OF LOG OF CURRENT CYCLES')
WRITE(6,1010)IOPT
X0=BND SL
DO 5003=1,NODE
P=GCDF(X0,IOPT,NODE,BNDS,DENS)
BNDX(I)=X0
X0=X0+HM
DISTX(I)=P
5003 CONTINUE
WRITE(6,994)
994 FORMAT('CDF OF LOG OF CURRENT CYCLES, LOG XMIN, 
LY AXIS OF PDF, CDF PLOT')
WRITE(6,1001)(DISTX(I),I=1,NODE)
C
WRITE(6,993)
993 FORMAT('ORDERED LOG OF CURRENT CYCLES, LOG XMIN, 
IX AXIS OF PDF, CDF PLOT')
WRITE(6,1001)(BNDS(I),I=1,NODE)
WRITE(6,1001)(BNDSX(I),I=1,NODE)
WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT 
TO THE PLOT FILES
WRITE(35,990)
WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
STOP
END
SUBROUTINE SORT(Y,N)
DIMENSION Y(10000)
N=N-1
DO 1 I=1, N
J=I+1
DO 2 K=J, N
IF(Y(I),LT,Y(K))GO TO 2
TEMP=Y(I)
Y(I)=Y(K)
Y(K)=TEMP
2 CONTINUE
1 CONTINUE
RETURN
END

IMSL Name: D3SPL/DD3SPL (Single/Double precision version)
Computer: IBM/SINGLE
Revised: November 1, 1985
Purpose: Nonparametric probability density function estimation
estimation by the normalized likelihood method.
Usage: CALL D3SPL(NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS, 
DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B, 
IPUT, WK2)
Arguments:
NOBS - Number of observations. (Input)
SUBROUTINE DISPL (N, X, NODE, BANDS, INIT, ALPHA, MAXIT, EPS,
&    DENS, STAT, HESS, LDHESS, ILOH, DENSEST, B,
&    IPUT, WK2)
!
C INTEGER N, NODE, INIT, MAXIT, LDHESS, ILOH(NODE),
&    IPUT(8)
C REAL 
&    ALPHA, EPS, X(*), BANDS(2), BANDS(2), STAT(4),
&    HESS(LDHESS,*), DENSEST(NODE), B(*), WK2(*)
!
C INTEGER I, IMIT, IPT, ITER, K, KM1, KM2, KPI, KP2, H, HOLD,
&    IMR, NOBI
C REAL BK, BKMI, BSALL, CK, CKM1, CKM2, CKMCHI, CKPI, CKP2,
&    CONS, EPS1, FACTOR, FK, FKM1, FKM2, FKPI, H, H2, H3,
&    SUM, TEMP, SUM1, SUM2, SUM3
!
C INTEGER MINCR(8)
C SAVE MINCR
C intrinsic alg, max, max0, min, mod, sort
INTRINSIC:  ALGB, AMAX1, MAX0, MIN0, MOD, SORT
INTEGER  MAX0, MIN0, MOD
REAL  ALGB, AMAX1, SORT

EXTERNAL  EIMES, EIPOP, EIPOP, EIPOP, EIPOP, EIPOP, EIPOP, EIPOP, EIPOP

! SPECIFICATIONS FOR SUBROUTINES
! SPECIFICATIONS FOR SUBROUTINES

EXTERNAL  ISMNM, ISMNM, ISMNM, ISMNM, ISMNM, ISMNM
 INTEGER  ISMNM, ISMNM
 REAL  ISMNM, ISMNM


CALL EIPOP ("DISPL")  Error checks

NER = 1
IF (MINS .LT. 1) THEN
 CALL EIMES (5., 1., "After removing all missing (NaN, not a"
 &  "'number') values from X there are no valid"
 &  "observations. At least one valid observation"
 &  "is necessary.")
END IF

IF (NODE .LE. 4.) THEN
 CALL EISTI (1., NODE)
 CALL EIMES (5., 2., "NODE = %I(I). The number of mesh"
 &  "nodes, NODE, must be an odd integer greater"
 &  "than 4.")
 ELSE IF (MOD(NODE,2) .EQ. 0) THEN
 CALL EISTI (1., NODE)
 CALL EIMES (5., 3., "NODE = %I(I) must be an odd integer"
 &  "greater than 4.")
END IF

IF (ALPHA .LE. 0.0) THEN
 CALL EISTR (1., ALPHA)
 CALL EIMES (5., 4., "ALPHA = %R(I). The penalty weighting"
 &  "factor which controls smoothness, ALPHA, must"
 &  "be greater than 0.")
END IF

IF (MAXIT .LE. 0.0) THEN
 CALL EISTR (1., MAXIT)
 CALL EIMES (5., 5., "MAXIT = %I(I). The maximum number"
 &  "of iterations, MAXIT, must be greater than 0.")
END IF

IF (BNDS(1), GT, BNDS(2)) THEN
 CALL EISTR (1., BNDS(1))
 CALL EISTR (2., BNDS(2))
 CALL EIMES (5., 6., "BNDS(1) = %R(I) and BNDS(2) = %R(I)"
 &  "The minimum value for X, BNDS(1), must"
 &  "be less than or equal to the maximum value for"
 &  "X, BNDS(2)."
END IF

IF (INIT .NE. 0) THEN
 IF (DENS(1), .NE. 0.0) THEN
 CALL EISTR (1., DENS(1))
 CALL EISTR (2., DENS(2))
 CALL EISTR (1., NODE)
 CALL EIMES (5., 7., "DENS(1) = %R(I) and DENS(NODE = %I(I))"
 &  "= %R(I). The beginning and ending initial"
 &  "estimates of the density must be zero.")
END IF
 IF (DENS(ISMIN(NODE,DENS(I)), .LT. 0) THEN
 CALL EIMES (5., 8., "The initial estimates of the"
 &  "density, DENS, must be greater than or"
 &  "equal to 0.")
END IF
END IF
M01 = 0
GO TO 10
IF (X(I).LT.BNDS(1) .OR. X(I).GT.BNDS(2)) THEN
M01 = M01 + 1
END IF
10 CONTINUE
IF (M01.EQ. NOBS) THEN
CALL EINES (5, 9, 'All elements in X lie outside the ' //
'interval BNDS(1) to BNDS(2). At least one ' //
'element of X must lie in this interval.' )
END IF
IF (EPS .LE. 0.0) THEN
EPS1 = 1.0E-4
ELSE
EPS1 = EPS
END IF
GO TO 9000
C IMPTR = 0 Initialization
C IF (INIT .EQ. 0) THEN Set initial densities
DENS1 = 0.0
DENS2 = 2.0/(BNDS(2)-BNDS(1))
M = 3
ELSE M = NODE
END IF
C 20 IF (INIT .EQ. 0) THEN Refine mesh
MOLD = M
IMPTR = IMPTR + 1
M = MINO(NODE+MINCR(IMPTR))
END IF
C H = (BNDS(2)-BNDS(1))/(M-1) Get mesh interval width
H2 = H/2
H3 = H/3
C IF (INIT .NE. 0) THEN Make initial DENS integrate to 1.
CALL SSCAL (NODE, 1.0/(H#SUM(NODE+DENS+1)), DENS, 1)
END IF
C B(1) = BNDS(1) Set mesh nodes
DO 30 I=2, M
B(I) = B(I-1) + H
30 CONTINUE
C IMPTR = 0 Set B indices for interpolating X
C 40 IMPTR = IMPTR + 1
IF (X(IPTR).LT. BNDS(1)) GO TO 40
DO 50 K=1, M - IMPTR
ILBMI(K+2) = IMPTR
IF (K.GT. K1) THEN
50 IF (X(IPTR).LT. B(K+1)) THEN
ILBMI(K+2) = ILBMI(K+2) + 1
IF (IPTR .LE. NOBS) GO TO 50
END IF
END IF
FACTOR = 2.0*ALPHA/M3

IF (INIT.EQ. 0) THEN
    CALL D2SPT (M, B2, I, MOLD, BDDS, DENS, DENSEST, WK, WK,
    TEMP = 1.0/(M*M*M)
    DO 80 I=2, M-1
        DENS(I) = AMAX1(TEMP, SQT(DENS(I)))
    80 CONTINUE
ELSE
    DO 80 I=2, M-1
        DENS(I) = SQT(DENS(I))
    80 CONTINUE
END IF

DO 140 ITER=1, MAXIT
    MAXIMIZE
    HESS(1,1) = 0.0
    HESS(2,1) = 0.0
    BSMALL = 0.0
    SUM = 0.0
    CK** are true estimates = FK**2
    KM1 = K - 1
    KM2 = MAX(K1+1, K-2)
    KP1 = K + 1
    KP2 = MIN(M+K+2)
    FK = DENS(K)
    FKM1 = DENS(KM1)
    FKM2 = DENS(KM2)
    CKM2 = FKM2**2
    CKN1 = FKM1**2
    CK = FK**2
    CKP1 = DENS(KP1)**2
    CKP2 = DENS(KP2)**2
    BK = B(K)
    BKM1 = B(KM1)
    SUM = SUM + CK
    IF (K1.GE.4) HESS(1,KM1) = 4.0*FK*FKM2*FACTOR
    SUM1 = 0.0DO
    SUM2 = 0.0DO
    SUM3 = 0.0DO
    DO 100 I=ILOHI(K1), ILOHI(K-2)
        TEMP = (X(I) - BK)/H
        CONS = (1.0 - TEMP)/ (C(K*CKP1 - CK)**TEMP)
        SUM1 = SUM1 + CONS
        SUM2 = SUM2 + CONS*CONS
        SUM3 = SUM3 + CONS**(1.0 - CONS)/SUM
    100 CONTINUE
    CKNM1 = CK - CKN1
    DO 110 I=ILOHI(KM1), ILOHI(KM2)
        CONS = (X(I) - BK)*H
        TEMP = CKN1 + CKNM1*CONS
        SUM1 = SUM1 - CONS/TEMP
        SUM2 = SUM2 + CONS*CONS/TEMP
        SUM3 = SUM3 + CONS**CONS/(TEMP - CONS)^2
    110 CONTINUE
    TEMP = FACTOR*(CKM2 + CKF2 - 4.0*(CKM1 + CKP1) + 6.0*CK) + SUM1
    TEMP = 2.0*TEMP
    BSMALL = BSMALL + 2.0*CK*TEMP
HESS(3,KM1) = TEMP + 4.0*CK*KM1*SUM2
IF (K, NE, 2) HESS(2,KM1) = 4.0*FK*KM1*SUM4
IF (K, NE, 1) = FK*TEMP
DRENEST(KM1,2) = -2.0*FK

CONTINUE
BSMALL = 1.0/H - SUM + BSMALL
CALL SCOPY (M-2, DRENEST(1,2), 1, DRENEST(1,3), 1)
CALL SADD (M-2, BSMALL/(2.0*SUM), HESS(3,1), LDXHESS)
CALL SCOPY (M-3, HESS(1,3), LDXHESS, HESS(5,1), LDXHESS)
HESS(5,M-3) = 0.0
CALL SCOPY (M-3, HESS(2,2), LDXHESS, HESS(4,1), LDXHESS)
HESS(4,M-2) = 0.0
CALL ILRFB (M-2, HESS, LDXHESS, 2, 2, HESS, LDXHESS, IPVT, WK)
CALL ILFB (M-3, HESS, LDXHESS, 2, 3, IPVT, DRENEST, 3, DRENEST)

IF (NIRCD(1) .NE. 0) GO TO 900
C
CONS = SDOT(M-2, DRENEST(1,3), 1, DRENEST(1,3), 1)/CONS
CONS = (0.0/H-SUM-SDOT(M-2, DRENEST(1,3), 1, DRENEST(1,3), 1))/CONS
CALL SAXPY (M-2, CONS, DRENEST(1,2), 1, DRENEST(1,1), 1)
CALL SAXPY (M-2, -1.0, DRENEST(1,1), 1, DRENEST(2,1), 1)
C
TEMP = SNRM2(M-2, DRENEST1, 1)
IF (SNRM2(M-2, DRENEST1, 1) .LT. EPS1*TEMP) GO TO 150
C
C
DO 130 I=2, M-1
DENS(I) = AMAX1(TEMP, DENS(I))
C
CONTINUE
130 DO 140 CONTINUE
CALL EISTI (1, MAXI)
CALL EIMES (3, 1, 'The maximum number of iterations '/
' (MAXI=K(I)) was exceeded.',)
C
150 IF (M .NE. NODE) GO TO 20
C
SUM1 = 0.0
C
DO 160 K=1, M
KM1 = MAXI(K-1, 1)
KP1 = MINI(K+1, M)
SUM1 = SUM1 + (DENS(KM1)-2.0*DENS(K)+DENS(KP1))**2
C
STAT(2) = -0.5*FACT#SUM1
SUM2 = 0.0
C
DO 170 I=1, NOBS
IF (X(K), LE, BND5K(2)) THEN
CALL D2SF (1, X(K), 1, NOBS, BNDK, DENS, DRENEST, WK, WK)
SUM2 = SUM2 + ALOG(DRENEST(1,1))
END IF
C
170 CONTINUE
STAT(1) = SUM2
C
Evaluate the log-likelihood and penalty
C
Evaluate the log-likelihood
C
Evaluate the log-likelihood
C
Evaluate the M.L.P.E. mean and variance
SUM1 = 0.0
SUM2 = 0.0
DO 120 K = 1, M - 1
   FK = DENS(K)
   FKP1 = DENS(K + 1)
   BK = B(K)
   CONS = FK + FKP1
   TEMP = CONS + FKP1
   SUM1 = SUM1 + H2*TEMP/6.0 + 0.5*H*BK*CONS
   SUM2 = SUM2 + H3*(TEMP*FKP1)/12.0 + H2*BK*TEMP/3.0 +
   0.5*H*BK*BK*CONS
120 CONTINUE
STAT(3) = SUM1
STAT(4) = SUM2 - SUM1*SUM1
C 9000 CALL E1POP ('D3SPL ')
RETURN
END
8.0 APPENDIX C

RANDOM4 SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUTFILES
C HAMIS MICROMECHANICS CONSTITUTIVE EQUATIONS
C RANDOMIZED AND APPLIED TO FAILURE STRENGTH

INTEGER NDOT,ISEED,M,TIM,MISS,MAXIT,NODE
REAL XM,YS,XM,YS,EPSP,EPSSP,ALPHA
DIMENSION SF(10000),XLMF(10000),SIG(10000)
DIMENSION XLMF(10000),SF(10000)
DIMENSION SIG(10000),SIG(10000)
DIMENSION XXN(10000),XMM(10000),XXQ(10000)
DIMENSION XXM(10000),XMMX(10000),XXNT(10000)
DIMENSION TF(10000),TD(10000),T(10000)
DIMENSION STAT(999),ENS(999)
DIMENSION C(999),PP(999)
DIMENSION SM(10)
DIMENSION XP(I),Y(1)
DIMENSION AL(I)
COMMON/NEPI/XP,KPRINT,TOL,MASFN

KPRINT=1
TOL=0.01
MASFN=30

C LOGNORMAL ULTIMATE TENSILE STRENGTH, SF
READ (5,1005) ISEED,NDOT
WRITE (6,1005) ISEED,NDOT
XM=300.
XS=45.
READ (5,1006) XM, XS
WRITE (6,1006) XM, XS
YS(1)=SRT(LOG(1.0+(XS/XM)**2.) )
YM(1)=LOG(XM) - 0.5*YS(1)**2.
CALL RNSET(ISEED)
CALL RLMLNT(DT,MP,YS,SP)
WRITE (18,1001) (SF(I),I=1,NDOT)
WRITE (6,2001)

C LOGNORMAL LOG OF FINAL CYCLE, XLMF
READ (5,1006) XM, XS
WRITE (6,1006) XM, XS
XM=98.
XS=0.8
YS=SRT(LOG(1.0+(XS/XM)**2.) )
YM=LOG(XM) - 0.5*YS**2
CALL RNSET(ISEED)
CALL RLMLNT(DT,MP,YS,XLMF)
WRITE (18,1001) (XLMF(I),I=1,NDOT)
WRITE (6,2001)

C LOGNORMAL XLMF
WRITE (6,1001) (XLMF(I),I=1,NDOT)
2046 FORMAT( 'NORMAL TFR')
WRITE (*,1001)(TF(I),I=1,NTOT)
C NORMAL REFERENCE TEMPERATURE TO
WRITE (*,1005) ISED,NTOT
READ(*,1006) YM*YS
YM=20.
YS=0.6
CALL RSET(ISEED)
CALL RNOR(NTOT,T)
DO 406 I=1,NTOT
TO(I)=YM*TS(I)+YM
406 CONTINUE
WRITE(*,2047)
2047 FORMAT( 'NORMAL TFR')
WRITE(*,1001)(TOT(I),I=1,NTOT)
C NORMAL CURRENT TEMPERATURE T
WRITE(*,1005) ISED,NTOT
READ(*,1006) YM*YS
WRITE(*,1006) YM*YS
C YM=850.
YS=42.9
CALL RSET(ISEED)
CALL RNOR(NTOT,T)
TIM=TS(I)+YM
WRITE(*,2048)
407 CONTINUE
WRITE(*,2048)
2048 FORMAT( 'NORMAL TFR')
WRITE(*,1001)(TIM(I),I=1,NTOT)
C CALCULATE CURRENT LOG OF CYCLES: LOG XNM
DO 102 I=1,NTOT
XNM=(SF(I1)-SIGD(I))/((SF(I1)-SIGD(I)))***XNM(I)
WRITE(*,1007) R

6876 FORMAT( 'XNM=**E12.4')
C TEMP=((TF(I1)/TF(I0))*XNM(I))
TEMP=((TF(I1)-TF(I0))/TF(I1)-TD(I1))*XNM(I)
WRITE(*,1010) TEMP
C786 FORMAT( 'TEMP=**E12.4')
SS=SF(I)

6886 FORMAT( 'XNM=**E12.4)

69876 FORMAT( 'XNM=**E12.4)

69875 FORMAT( 'XNM=**E12.4)

69875 FORMAT( 'XNM=**E12.4)

69875 FORMAT( 'XNM=**E12.4)

102 CONTINUE
WRITE(20,1001)(XNM(I),I=1,NTOT)
WRITE(6,2028)
2028 FORMAT('LOG OF CYCLES TO REACH MEAN FATIGUE STR = ',250. MPA')
WRITE(6,1001)(XNM(I),I=1,NTOT)
C SORT LOG OF CYCLES
CALL SORT(XNM,NTOT)
WRITE(*,2010)(XNM(I),I=1,NTOT)
WRITE(*,6,2029)
3029 FORMAT ("SORT ED LOG OF CYCLES")
WRITE(6,1001)(XNM(I)+1=1,NTO)
C CALCULATE PDF OF LOG OF CURRENT CYCLES, LOG XNM
C USING THE MAXIMUM ENTROPY METHOD
C CALCULATE SAMPLE MOMENTS, SM
C NUMBER OF MOMENTS, MM
XNM=4
CALL SMHM(XNM,MM,NHT,SM)
WRITE(30,1001)(SM(I),I=1,MM)
WRITE(6,2019)
3038 FORMAT ("SAMPLE MOMENTS")
WRITE(6,1001)(SM(I),I=1,MM)
C OBTAIN MAXIMUM ENTROPY DISTRIBUTION
4 START=1
DATA=1
C CALCULATE MAX AND MIN ORDINATES FOR PDF AND CDF
BNDS(1)=XNM(1) - 0.05*XNM(1)
BNDS(2)=XNM(NOT)+0.05*XNM(NOT)
WRITE(6,8877)BNDS(1),BNDS(2)
WRITE(6,9877)BNDS(1),BNDS(2)
8877 FORMAT ("BNDS(1),BNDS(2)='E12.4*IX,E12.4")
CALL MP01(MM,SM,SM(1),BNDS(2),0,XP,START,KDATA,AL,CUM)
WRITE(31,1001)(AL(I),I=1,MM+1)
WRITE(6,2039)
2039 FORMAT ("LAGRANGIAN MULTIPLIERS")
WRITE(6,1001)(AL(I),I=1,MM+1)
C CALCULATE VALUES OF ORDINATES FOR PDF AND CDF
C NUMBER OF ORDINATES USED
C CALCULATE WINDOW WIDTH, MM
NODE=2
MM=(BNDS(2)-BNDS(1))/(NODE-1)
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED
ALSO CALLED "NODE" VALUES
DO 6001(I=1,1,MM-2)
BNDS(I)=BNDS(1)+(I*MM)
6001 CONTINUE
WRITE(6,983)
983 FORMAT ("LOG OF CURRENT CYCLES; LOG XNM")
WRITE(6,1001)(BNDS(I),I=1,NODE)
C REORDER BNDS FOR PLOTTING
C SAVE = BNDS(NODE)
SAVE1 = BNDS(NODE)
BNDS(NODE)=BNDS(2)
DO 6002(I=1,NODE-2)
BNDS(I+1)=BNDS(I+2)
6002 CONTINUE
BNDS(NODE-1)=SAVE2
BNDS(NODE)=SAVE1
WRITE(6,984)
984 FORMAT ("ORDERED LOG OF CURRENT CYCLES; LOG XNM,
1X AXIS PDF, CDF PLOT")
WRITE(6,1001)(BNDS(I),I=1,NODE)
C CALCULATE VALUES OF THE PDF AT EACH ORDINATE
DO 1000(I=1,NODE)
C FOR 4 MOMENTS THERE ARE 5 LAGRANGIAN MULTIPLIERS
DENS(I)=EXP(AL(1)+AL(2)*BNDS(I)+AL(3)*BNDS(I)**2
+AL(4)*BNDS(I)**3+AL(5)*BNDS(I)**4)
1000 CONTINUE
C WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
LOG XNM TO PLOT FILES
WRITE(34,990)
990 FORMAT(1X,E12.4,1X,E12.4 *)
WRITE(34,991)(BNDS(J),DENS(J),J=1,NODE)
991 FORMAT(1X,E12.4,1X,E12.4 *)
CALCULATE CDF OF LOG OF CURRENT CYCLES
IOPT=2
REM(3,1004)IOPT
WRITE(6,992)
992 FORMAT('CDF PARAMETERS')
WRITE(6,1004)IOPT
X=BNDS(1)
DO 6003 I=1,NODE
F=Gcdf (XQ(IOPT),NODE,BNDS,DENS)
BNDSX(I)=XQ
XQ=XQ+HH
DISTX(I)=P
6003 CONTINUE
WRITE(6,994)
994 FORMAT('CDF OF LOG OF CURRENT CYCLES, LOG XNM, 
LX AXIS OF PDF, CDF PLOT')
WRITE(6,1001)(DISTX(I),I=1,NODE)
WRITE(6,993)
993 FORMAT('ORDERED LOG OF CURRENT CYCLES, LOG XNM, 
SX AXIS OF PDF, CDF PLOT')
WRITE(6,1001)(BNDS(I),I=1,NODE)
WRITE(6,1001)(BNDSX(I),I=1,NODE)
WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT 
TO THE PLOT FILES
WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
STOP
END
SUBROUTINE SORT(Y,N)
DIMENSION Y(100000)
Y IS THE ARRAY TO BE SORTED
C AT COMPLETION Y(N) IS THE SMALLEST VALUE
C AT COMPLETION Y(N) IS THE LARGEST VALUE
N1 = N
DO 1 I=1,N1
J= I+1
DO 2 K=J,N
IF (Y(J),LT,Y(K))GO TO 2
Y(J)=Y(J)
Y(J)=TEMP
2 CONTINUE
1 CONTINUE
RETURN
END
SUBROUTINE SNDM(X,M,NSAMP,SN)
C CALCULATES SAMPLE CENTRAL MOMENTS
C X(I) = SAMPLE VALUES; DIMENSION NSAMP
C M = NUMBER OF MOMENTS DESIRED
C NSAMP = SAMPLE SIZE
C SN = VALUE OF MOMENTS; DIMENSION M
DIMENSION X(100000),SN(10)
C. CALCULATE MEAN
    SUM=0.0
    DO 1 I=1,NSAMP
        SUM=SUM+X(I)
    1 CONTINUE
    SM1=SUM/FLOAT(NSAMP)
    IF (M.LT.1) RETURN

C. CALCULATE VARIANCE
    SUM=0.0
    DO 2 I=1,NSAMP
        SM2=SUM+X(I)-SM1
    2 CONTINUE
    SM2=SUM/FLOAT(NSAMP-1)
    IF (M.LT.2) RETURN

C. CALCULATE HIGHER MOMENTS
    DO 4 I=3, M
        SUM=0.0
    3 CONTINUE
    DO 4 J=1,NSAMP
        SM3=SUM+X(J)-SM1
    4 CONTINUE
    RETURN
END

SUBROUTINE MEPI(N,CM,XMIN,XMAX,NXP,XP,KSTART,KDATA,AL,CUM)
  IMPLICIT REAL*8 (A-H,O-Z)
  DIMENSION AL(*), CM(*), ETA(4), XP(*), CUM(*), C(8), ALE(10)
  COMMON /FAIL, NFAIL
  COMMON/HELP(5,101),XX(101,101),C(8),M

C. EXECUTIVE PROGRAM FOR USING MAXIMUM ENTROPY METHOD CONSTRAINED BY
   MOMENTS TO GENERATE A DENSITY FUNCTION
   DIMENSION AL(*), CM(*), ETA(4), XP(*)
   COMMON/FAIL, NFAIL
   COMMON/HELP(5,101),XX(101,101),C(8),M
   IF (N.PT.1) KSTART=2
   DATA KPRINT, TOL, MAXFN/1,1,1.E-6,70/

WRITE THE INPUT DATA
   IF (KDATA.EQ.0) GO TO 1
   WRITE (6,24) KDATA
   WRITE (6,25) KPRINT
   WRITE (6,28) N
   WRITE (6,29) XP
   WRITE (6,25) XMAX
   WRITE (6,30) XMIN
   WRITE (6,31) (CM(I), I=1,4)
   IF (N.GT.4) WRITE (6,32) (CM(I), I=5,N)
   IF (ABS(CM(I)), I.LT.1.E-4) GO TO 40
   WRITE (6,33) TOL
   WRITE (6,33) NXP
   1 CONTINUE
   NFAIL=0
   M=1
   X2MIN=0.0
   X2MAX=1.0
   DO 100 I=1,N
       100 CCI=CM(I)

C. CALCULATE THE MOMENTS AT THE MODIFIED LIMITS
   CALL TRNI (XMAX,XMIN,CC,X2MAX,X2MIN,N)
C. CALCULATE THE MOMENTS ABOUT THE ORIGIN FOR THE MODIFIED LIMITS
   STORE THEM IN COMMON IN C
CALL CONVER(CC,N)

GENERATE THE SIMPSON MULTIPLIERS AND STORE THEM IN HELP COMMON

CALL SIMSON

GENERATE THE X*S POWER FOR SUBROUTINE FUNCT, STORE THEM IN HELP COMMON ARRAY

CALL MULTI(XMAX,XMIN,N)

DEFINE THE INPUT DATA FOR SUBROUTINE MPOPT

ETA(1)=1.0-12
ETA(2)=TOL
ETA(3)=1.0-24
ETA(4)=1.0-24
MODE=1
UMIN=0.0

WRITE THE INTERMEDIATE RESULTS YOU HAVE OBTAINED SO FAR

IF (KPRINT.EQ.0) GO TO 2
WRITE (6,34)
WRITE (6,35) M
WRITE (6,36) XMAX,XMIN
WRITE (6,37) (CC(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (CC(I),I=5,N)
WRITE (6,38) (C(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (C(I),I=5,N)
WRITE (6,39) (ETA(I),I=1,4)
CONTINUE

FIND A STARTING POINT FOR SUBROUTINE MPOPT TO START THE OPTIMIZATION ALGORITHM

IF (KSTART.EQ.0) GO TO 16
IF (KSTART.EQ.4) WRITE (6,44)
CALL START(XMAX,XMIN,AL,KSTART,CC,N,KPRINT,UMIN,MODE,MAXFN,ETA)
IF (NFAIL.EQ.1) GO TO 9

PRINT THE STARTING VALUES

IF (KPRINT.EQ.0) GO TO 7
GO TO (3,4,5,6,7)
KSTART
WRITE (6,40)
WRITE (6,41) (AL(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
GO TO 7
WRITE (6,42)
WRITE (6,41) (AL(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
GO TO 7
WRITE (6,43)
WRITE (6,41) (AL(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
GO TO 7
WRITE (6,44)
WRITE (6,41) (AL(I),I=1,4)
IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
GO TO 7
CONTINUE

RANGE=XMAX-XMIN

CHANGE STARTING VALUES TO 0-1 DOMAIN FOR KSTART=0
C.... THIS ALGORITHM IS SIMILAR TO TRN2, BUT APPEARS TO GIVE BETTER RESULTS.

MPL=N+1
IF(ABS(XMIN),LT.1.E-10) GO TO 19
DO 17 I=2,NPL
ALS(I)=0.0
I=I+1
DO 18 I=IAL+1,N
ALS(I)=ALS(I)+FACTOR(J)*XMIN*(J-I)*RANGE*II*AL(I)/FACTOR(II)
1/FAC(I)*J-I)
18 CONTINUE
7 CONTINUE
I=I+1
GO TO 50
19 DO 20 I=2,NPL
20 ALS(I)=RANGE*(I-1)*AL(I)
21 CONTINUE
C.... PUT AL(I) IN PROPER LOCATIONS
51 AL(I)=ALS(I+1)

7 CONTINUE
IFAIL=0
IF(KPRINT,EQ.0) GO TO 8
WRITE (6,45)
8 CONTINUE
AL(N+1)=2.0
AL(H+1)=0.0
CALL MPOPT(AL,N,ETA,UMIN+MAXFN+MODE,KPRINT)
IF(FAIL,EQ.0) GO TO 10
IF(KSTART,EQ.4) GO TO 9

THE PROGRAM HAS FAILED SO FAR, TRY ANOTHER STARTING POINT AND TRY AGAIN.
KSTART=KSTART+1
IF(KSTART,EQ.4,AND.M.LE.2) GO TO 9
GO TO 2
9 CONTINUE
WRITE (6,46)
CALL EXIT
CONTINUE
C.... CALCULATE THE ZEROTH LAGRANGIAN MULTIPLIER
SUM=0.0
DO 12 I=1,NPL
22 DO 11 K=1,N
SZ=SZ+AL(K)*X(K,I)
11 CONTINUE
SUM=SUM=S(K)*EXP(SZ)
12 CONTINUE
NPL=N+1
DO 13 I=1,N
13 K=NPL+1
AL(K)=AL(K-1)
CONTINUE
DELTA=(X2MAX-X2MIN)/FLOAT(N-1)
AL(I)=ALG(SUM*DELTA/3)
WRITE (6,101)SUM
101 FORMAT(2CH SUM OF RESIDUALS SQUARED=,E12.5)
IF(KPRINT,EQ.0) GO TO 14
WRITE (6,47) (AL(I),I=1,NPL)
CONTINUE
C.... RESET KSTART TO ZERO
CALCULATE THE LAGRANGIAN MULTIPLIERS FOR THE ORIGINAL LIMITS
CALL TRNZ(XMAX+XMIN+AL+XMAX+XMIN+H)

CALCULATE THE CUMULATIVE DISTRIBUTION FUNCTION VALUE AT THE GIVEN POINT
IF(NXP.EQ.0)RETURN
DO 15 I=1,NXP
CUM(I)=CDF(XMIN+XMAX+X(I)+AL+HPL)
15 CONTINUE
RETURN

FORMAT (5X,4E18.9/)  
FORMAT (5X,4E18.9/)  
FORMAT (5X,4E18.9/)  
FORMAT (5X,4E18.9/)  
FORMAT (5X,4E18.9/)  

I+

INPUT DATA IS PRINTED OUT FOR KDATA =1 ONLY KDATA =

INTERMEDIATE OUTPUT EVERY KPRINT CYCLE KPRINT =

NUMBER OF KNOWN FIRST MOMENTS N=

HIGHER LIMIT XMAX =

LOWER LIMIT XMIN =

FIRST MOMENTS CC(I) =

THE ALLOWED TOLERANCE IN LAGRANGIAN EQUATIONS TOL =

THE CUMULATIVE DISTRIBUTION REQUIRED AT NX POINTS.NXP =

INTERMEDIATE RESULTS FOR SUBROUTINE MEP1/2

NUMBER OF INTEGRATION STATION M =

MODIFIED MAXIMUM AND MINIMUM LIMITS XMAX XMIN =

MODIFIED MOMENTS ABOUT THE EXPECTED VALUE CC(I) =

MODIFIED MOMENTS ABOUT THE ORIGIN C(I) =

SUBROUTINE MPOSIT TOLERANCES ETA(I) =

NORMAL ASSUMPTION STARTING METHOD/34(--)/

STARTING VALUES AL(I) =

UNIFORM ASSUMPTION STARTING METHOD/35(--)/

M POINTS STARTING METHOD/29(--)/

STEP BY STEP STARTING METHOD/29(--)/

CYC NUNF NORMNRC TOTAL=24X VARIABLES=40

X, RESIDUALS/ NO.'S, 10X RESIDUALS X(1) X(2)

R(1) R(2) R(3) R

THE PROGRAM HAS FAILED
THE MODIFIED LAGRANGIAN MULTIPLIERS ARE
WRITE(6,49)
FORMAT(3H,34)

MEAN IS NEARLY ZERO AND MEP1 WILL NOT WORK/12
SUBROUTINE MPOPT (X,NDIM,ETA,EST,MAX,MODE,IPRINT)

IMPLICIT REAL*8 (A-H,O-Z)

REAL*8 KT1,IPRINT

COMMON /FAIL,/ NFAL

DIMENSION X(*) , ETA(*) , X(10), B1(10), B2(10), ALEA(10), H(10), P

EXTERNAL FUNCT

REST=0

IFLAG=0

M=0

N2=NDIM+1

N1=NDIM+2

NUMF=0

IER=0

DO 1 I=1,N1

X1(I)=X(I)

1 CONTINUE

CALL FUNCT (NDIM,X1,F1,G1,RR)

NUMF=NUMF+1

DO 2 I=1,NDIM

X2(I)=X1(I)

G2(I)=G1(I)

2 CONTINUE

CONTINUE

F2=F1

X2(N2)=X1(N2)

X2(N1)=X1(N1)

CONTINUE

KOUNT=0

EPS=ETA(4)

CALL LINES (FUNCTION,X2,H,RO,NDIM,F2,B2,NUMF,IER,EPS,EST,RR)

IF (IFAIL.EQ.1) RETURN

IF (IER.NE.0) GO TO 10.30

DO 4 I=1,N1

A1OU(I)=X2(I)

ALFA(I)=A2(I)

4 CONTINUE

RO=RO

GO=GO

GO=GO+G2(1)*G2(1)

CONTINUE

DO 5 I=1,NDIM

GO=GO+G2(1)*G2(1)

5 CONTINUE

IF (IPRINT.EQ.0) GO TO 7

IF (MOD(KT1,IPRINT).NE.0) GO TO 6

CALL OUTP (X2,F2,NDIM,RO,NUMF,RR)

6 KT1=KT1+1

7 DO 9 J=1,N1

P1(I,J)=0.

9 CONTINUE

P1(1,1)=1.

CONTINUE

PRINT*,KOUNT

KOUNT=KOUTN+1

11 DO 12 J=1,NDIM

Y(J)=G2(J)

12 CONTINUE

Y(N2)=F2
I

ORIGIN PAGE IS OF POOR QUALITY

32 CONTINUE
XI(N2)=X(N2)
X(N1)=X(N1)
Y(N2)=Y(N2)
22(N1)=2(N1)
GO TO 3

33 FORMAT ('... SOLUTION FOUND')
34 FORMAT ('///, IX.' THE PROGRAM HAS FAILED--IER = 'I2')
END

SUBROUTINE OUTP (XNEW,FO,KOUNT,NI,GO,NUMF,R)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION XNEW(*), R(*)
WRITE (6,6) KOUNT,GO,FO,(XNEW(I),I=1,4),(R(I),I=1,1)
IF (NI.LT.4) RETURN
NH=NI-3
GO TO (1,2,3,5), NN
RETURN
WRITE (6,7) XNEW(5),R(5)
RETURN
WRITE (6,8) (XNEW(I),I=5,6),(R(I),I=5,6)
RETURN
WRITE (6,9) (XNEW(I),I=7,7),(R(I),I=7,7)
RETURN
WRITE (6,10) (XNEW(I),I=8,8),(R(I),I=8,8)
RETURN

61 FORMAT (1X,13,14,6E14,5,4E10,3)
62 FORMAT (13X,14,5,42X,10,3)
63 FORMAT (36X,3E14,5,28X,10,3)
64 FORMAT (36X,3E14,5,14X,10,3)
65 FORMAT (36X,4E14,5,4E10,3)
END

SUBROUTINE LINES (FUNCT,X,H,AMBD,N,F,G,NUMF,IER,EPS,EST,RR)
IMPLICIT REAL*8 (A-H,O-Z)
REAL*8 Z,DA, DY
COMMON /FAIL/ NFAIL
DIMENSION H(*), X(*), G(*), RR(*)
IER=0
DY=0.
HHRM=0.
GHRM=0.
D0=0.1,N.
HHRM=HHRM+ABS(H(J))
GHRM=GHRM+ABS(G(J))
DY=DY+H(J)*X(J)
PRINT*,'GOT BY B1'
CONTINUE
IF (DY) 2,31,31
PRINT*,'GOT BY B2'
IF (X(N1)/NRM/DRM<=EPS) 31,31+3
PRINT*,'GOT BY B3'
F=F+F
FA=FA+2.*(EST-F)/DY
IF (X(N1)<0.0) ALFA=X(N1)*ALFA/2.
PRINT*, 'GOT BY B4'
IF (F-EY) 21, 21, 22
IF (F-EY) 30, 30, 22
DALFA=0.
DO 23 L=1, N
DALFA=DALFA+G(I)*H(I)
23 CONTINUE
IF (DALFA) 24, 27, 27
IF (F-EY) 26, 26, 27
IF (DX-DALFA) 26, 30, 26
FX=F
DX=DALFA
T=ALFA
AMBDA=ALFA
GO TO 11
IF (FX-F) 29, 28, 29
IF (DY-DALFA) 29, 30, 29
FY=F
DY=DALFA
AMBDA=AMBDA-ALFA
GO TO 13
AMBDA=AMBDA-ALFA
RETURN
31 CONTINUE
IF (DY. GE. 0.) IER=-2
IF (GNNM. LE. 1.E-10) GO TO 32
IF (NRRM/GRMN. LE. EPS) IER=-3
32 CONTINUE
IF (DALFA. LT. 0.) IER=-1
NFail=1
WRITE(5,33)
FORMAT(1X, 'THE PROGRAM HAS FAILED')
RETURN
END

SUBROUTINE FUNCT(M,AL,U,GRAD,RR)
IMPLICIT REAL*(8) (A-H,O-Z)

THIS SUBROUTINE IS USED TO CALCULATE THE OPTIMIZATION AND THE
GRADIENT AT ANY GIVEN POINT FOR SUBROUTINE PDPT

DIMENSION AL(*), GRAD(*), SUM(17), RR(*)
COMMON/HELP/S(101),XX(16,101),C(8),H
C..... ABOVE LINE CHANGED FROM TEXT

N21=20+N21
ZERO=0.0
DO I=1,N21
PRINT*, 'GOT BY CI'
CONTINUE
CONTINUE
DO I=1, M
S=ZERO
DO 3 K=1, N
S=S+AL(K)*XX(K, I)
3 CONTINUE
IF (S=0.74) GO TO 9
F=EXP(S)*GOT BY C3'
SS=EXP(S)*S(I)
SUM(I)=SUM(I)+SS
DO 4 J=2,N2
  SUM(J)=SUM(J)+XX(J-1,I)*SS
  PRINT*,'GOT BY C4'
  CONTINUE
  DO 5 I=2,N2
    SUM(I)=SUM(I)/SUM(1)
    PRINT*,'GOT BY C5'
    CONTINUE
    U=0.0
    DO 6 I=1,N
      RR(I)=(SUM(I+1)-C(I))/C(I)
      U=U+RR(I)**2
    PRINT*,'GOT BY C6'
    CONTINUE
    DO 7 K=1,N
      RR(K)=0.0
      DO 8 J=1,N
        RR(K)=RR(K)+(SUM(J+K+1)-SUM(J+1)*SUM(K+1))/RR(J)/C(J)
      PRINT*,'GOT BY C7'
      CONTINUE
      PRINT*,'GOT BY C8'
      RETURN
      PRINT*,'GOT BY C10'
      CONTINUE
      AA=S2-S2
      ZERO=ZERO-AA
      DO 2 TO 2
        PRINT*,'GOT BY C11'
        END

SUBROUTINE START (XMAX,XMIN,ALAMDA,KSTOP,CC,NL,IPRINT,UMIN,MODE,M)
   IMPLICIT REAL*8 (A-H,O-Z)
   THIS SUBROUTINE IS USED TO FIND A REASONABLE STARTING POINT FOR
   SUBROUTINE MPOPT
   DIMENSION R(N), ETA(N), ALAMDA(*)
   DIMENSION ALAMDA(*)
   COMMON/HELP/S(101),XX(16,101),C(8),M
   COMMON /FAIL/ NFAIL
   GO TO (3,1,5,26), KSTOP
   1 CONTINUE
   NFAL=0
   DO 2 I=1,NL
     ALAMDA(I)=0.0
   2 CONTINUE
   RETURN
   3 CONTINUE
   NFAL=0
   DO 4 I=1,2
     ALAMDA(I)=CC(I)/CC(2)
   4 CONTINUE
   DO 4 A=1,3,NL
     ALAMDA(A)=0.0
   5 CONTINUE
CONTINUE
DO 23 I=1,NP1
W(I,I)=R(I)
23 CONTINUE
DO 24 J=1,NP1
DO 24 T=1,NP1
W(I,J)=W(I,J)+X(I)*Y(J)
Y(J)=Y(J)+DELTA
24 CONTINUE
CALL SOLVE (W,Y,NID,NP1,10)
GO TO 12
CONTINUE
N=2
ALAMDA(2)=-.5/CC(2)
ALAMDA(1)=CC(1)/CC(2)
NFAIL=0
CONTINUE
ALAMDA(N+1)=2.0
ALAMDA(N+2)=0.0
C PRINTA*,GOT BY A,
CALL MP0FT (ALAMDA,N,ETA,UMIN,MAXFN,MODE,IPRINT)
C PRINTA*,GOT BY B
IF (NFAIL.EQ.1) RETURN
IF (N.EQ.NL).RETURN.
ALAMDA(N+1)=0.0
N=N+1
GO TO 27
END

C SUBROUTINE SOLVE (A*X+ID-N,N,NA)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(NA,N), X(N)
D=0.0
DATA DIV/69347181/
DO 6 I=1,N
AA=0.0
DO 1 J=1,N
AB=AB+AB(A(J,I))
IF (AB.LE.AA) GO TO 1
K=J
AA=AB
CONTINUE
B=D+ALGB(AA)
IF (I.EQ.N) GO TO 7
IF (K.EQ.I) GO TO 3
DO 2 J=1,N
AB=AB+AB(A(J,J))
A(J,J)=AB
2 CONTINUE
AB*X(I)
X(I)=X(K)
X(K)=AB
II=I+1
DO 5 J=I+1,N
AA=AA+B(A(I,J))/A(I,I)
A(J,I)=AA
5 CONTINUE
DO 4 K=1,N
A(J,K)=A(J,K)+AA*A(I,K)
4 CONTINUE
SUBROUTINE SIMSON
  IMPLICIT REAL*8 (A-H,O-Z)
  THIS SUBROUTINE IS TO CALCULATE THE SIMPSON MULTIPLIERS
  COMMON/HELP/S(101),XX(16,101),C(8)*M
  ABOVE LINE CHANGED FROM TEXT
  S(1)=&1.
  S(M)=1.
  N=M-1
  DO 1 I=2,N+2
  S(I)=4.
  CONTINUE
  N=N-1
  DO 2 I=3,N+2
  S(I)=2.
  CONTINUE
  RETURN
  END

SUBROUTINE MULTI (XMAX,XMIN,N)
  IMPLICIT REAL*8 (A-H,O-Z)
  THIS SUBROUTINE IS USED TO GENERATE THE X,S POWER FOR SUBROUTINE
  FUNCT
  COMMON/HELP/S(101),XX(16,101),C(8)*M
  ABOVE LINE CHANGED FROM TEXT
  DELTA=(XMAX-XMIN)/FLOAT(M-1)
  DO 1 I=1,M
  XX(I,I)=XMIN+FLOAT(I-1)*DELTA
  M=M+2
  DO 1 J=2,NN
  XX(J,I)=XX(J-1,I)*XX(1,I)
  CONTINUE
  RETURN
  END

SUBROUTINE CONVER (CM,NL)
  IMPLICIT REAL*8 (A-H,O-Z)
  THIS SUBROUTINE IS TO CALCULATE THE MOMENTS ABOUT THE ORIGIN
  DIMENSION CM(*)
COMMON/HELP/S(101)*X(16+101)-C(1)*H

C.... ABOVE LINE CHANGED FROM TEXT
C(I)=CM(I)
IF (NL.EQ.1) RETURN
DO 1 J=1,N
C(J)=C(J)-C(I)**J*(-1)**J
N=J-1
DO 1 K=1,N
C(J)=C(J)*(-1.)**K*FACT0(J)/(FACT0(K)*FACT0(J-K))*C(I)**(K)*C(J-K)
CONTINUE
CONTINUE
RETURN
END

SUBROUTINE TRN1 (X1MAX,X1MIN,C,X2MAX,X2MIN,NL)
IMPLICIT REAL*8 (A-H,O-Z)

DIMENSION C(I)
SCL=(XMAX-XMIN)/(X2MAX-X2MIN)
C(I)=C(I)/SCL-XMIN/SCL*X2MIN
IF (NL.EQ.1) RETURN
DO 1 I=2,N
1 CONTINUE
RETURN
END

SUBROUTINE TRN2 (X1MAX,X1MIN,X,X2MAX,X2MIN,N)
IMPLICIT REAL*8 (A-H,O-Z)

THIS SUBROUTINE IS AN ALTERNATIVE TO TRN2 (BELOW)

CALCULATES THE LAGRANGIAN MULTIPLIERS FOR A DIFFERENT INTERVAL

DOUBLE PRECISION S*A,DX(I),FAC,DX1MAX,DX1MIN,DX2MAX,DX2MIN

DIMENSION X(I)
DX1MAX=X1MAX
DX1MIN=X1MIN
DX2MAX=X2MAX
DX2MIN=X2MIN
NPI=N+1
DO 10 I=1,NPI
10 DX(I)=X(I)
S=(DX1MAX-DX1MIN)/(DX2MAX-DX2MIN)
A=DX2MIN-DX1MIN/S
DX(I)=DX(I)-ALOG(S)
DO 1 L=1,N
1 CONTINUE
IF (NL.EQ.1) GO TO 6
DO 5 J=2,N
DO 3 I=J,N
FAC=1.
K=I-J+2
B=DX(I)/S
3 CONTINUE
10 FAC=FAC**B(FLOAT(K))
6 CONTINUE
DX(J)=DX(J)+FAC**B(FLOAT(J-I))**B*(I-J+1)**B*(I+1)
CONTINUE
DX(J)=DX(J)/S***(J-I)
SUBROUTINE TRN2 (XIMAX, XIMIN, X, X2MAX, X2MIN, N)
IMPLICIT REAL*8 (A-H, O-Z)

THIS SUBROUTINE IS USED TO CALCULATE THE LAGRANGIAN MULTIPLIERS
AT THE ORIGINAL LIMITS

DIMENSION X(1)
S=(XIMAX-XIMIN)/(X2MAX-X2MIN)
A=X2MIN-XIMIN/S
X(1)=X(1)-ALOG(S)
DO 1 I=1, N
X(I)=X(I)*X(I+1)**A
1 CONTINUE
IF (N.EQ.1) GO TO 5
DO 5 J=2, N
DO 2 I=1, N
FAC=1.
K=I+J-2
DO 2 K=K, I+J-2
FAC=FAC*FLOAT(K)
2 CONTINUE
X(J)=X(J)+FAC/FACFACT(J-1)**A**((I-J+1)**X(I+1))
CONTINUE
X(J)=X(J)/S**(J-1)
CONTINUE
CONTINUE
X(N1)=X(N1)/S**N
RETURN
END

FUNCTION CDF (XMIN, XMAX, XP, AL, N)
IMPLICIT REAL*8 (A-H, O-Z)

THIS FUNCTION IS TO CALCULATE THE CUMULATIVE DISTRIBUTION FUNCTION AT A GIVEN POINT

INPUT
XMIN = LOWER BOUND
XMAX = UPPER BOUND
XP = SPECIFIED POINT
AL(I) = ARRAY OF PARAMETERS, DIMENSION N
N = NUMBER OF PARAMETERS

DO 1 I=2, N
IF (XP, LE, XMIN) GO TO 3
IF (XP, GE, XMAX) GO TO 4
RANGE=XMAX-XMIN
SS=Range/FLOAT(JSM1)
JSS=SS
JSS=(JSS/2)**2+5
JSM1=JSS-1
DELTA=Range/FLOAT(JSM1)
DO 1 I=2, JSM1+2
X=XMIN+FLOAT(I-1)*DELTA
AREA=AREA+FRRF(AL, N, X)
1 CONTINUE
RETURN
END
CONTINUE
JSM1=JSM1-1
DO 3 I=3,JSM1+2
X=XMIN*FLOAT(I-1)*DELTA
AREA=AREA+2.*ENTRPF(AL,N,X)
CONTINUE
3 AREA=AREA+ENTRPF(AL,N,XMIN)+ENTRPF(AL,N,X)
AREA=AREA*DELTA/3.
CDF=AREA
DO TO 5
CDF=0.
DO TO 5
CDF=1.
CONTINUE
RETURN
END

FUNCTION ENTRPF (AL,NPL,X)
IMPLICIT REAL*8 (A-H,O-Z)
FUNCTION TO EVALUATE THE ENTROPY DENSITY FUNCTION AT A GIVEN POINT
INPUT
AL(I) = ARRAY CONTAINING PARAMETERS, DIMENSION NPL
NPL = NUMBER OF PARAMETERS
X = GIVEN VALUE
DIMENSION AL(N)
S=AL(I)
DO 1 I=2,NPL
S=S*AL(I)***(I-1)
CONTINUE
1 ENTRPF=EXP(S)
END

FUNCTION FACTO (M)
C IMPLICIT REAL*8 (A-H,O-Z)
C. . . . CALCULATES FACTORIAL OF M
FACTO=1.
IF (M.EQ.0) RETURN
DO 1 I=1,M
FACTO=FACTO*FLOAT(I)
1 CONTINUE
RETURN
END
<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input data is printed out for KDATA = 1 only</td>
<td>KDATA = 1</td>
</tr>
<tr>
<td>Intermediate output every KPRINT(TH) cycle</td>
<td>KPRINT = 1</td>
</tr>
<tr>
<td>Number of known first moments</td>
<td>N = 4</td>
</tr>
<tr>
<td>Higher limit</td>
<td>XMAX = 9.63779301E+01</td>
</tr>
<tr>
<td>Lower limit</td>
<td>XMIN = 0.572349819E+01</td>
</tr>
<tr>
<td>First moments</td>
<td>CC(1) = 0.735481628E+01</td>
</tr>
<tr>
<td>The allowed tolerance in Lagrangian equations</td>
<td>TOL = 0.17816965E+00</td>
</tr>
<tr>
<td>The cumulative distribution required at NXP points, NXP =</td>
<td>NXP = 0</td>
</tr>
</tbody>
</table>
### INTERMEDIATE RESULTS FOR SUBROUTINE MEP

<table>
<thead>
<tr>
<th>CYC NUM</th>
<th>NORMGRAD RESIDUALS</th>
<th>TOTAL X(1)</th>
<th>X(2)</th>
<th>VARIABLES X(3)</th>
<th>X(4)</th>
<th>R(1)</th>
<th>R(2)</th>
<th>RESIDUALS R(3)</th>
<th>R(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.17606E-01</td>
<td>0.23719E-02</td>
<td>0.11222E+02</td>
<td>-0.13405E+02</td>
<td>0.23393E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>1</td>
<td>0.3297E-02</td>
<td>0.22026E-03</td>
<td>0.11265E+03</td>
<td>-0.13460E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>2</td>
<td>0.1941E-02</td>
<td>0.7907E-03</td>
<td>0.11279E+03</td>
<td>-0.13460E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>3</td>
<td>0.8846E-02</td>
<td>0.3427E-03</td>
<td>0.11279E+03</td>
<td>-0.13460E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>4</td>
<td>0.2778E-02</td>
<td>0.2301E-03</td>
<td>0.11279E+03</td>
<td>-0.13460E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>5</td>
<td>0.4249E-02</td>
<td>0.1884E-03</td>
<td>0.11279E+03</td>
<td>-0.13460E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>6</td>
<td>0.1050E-02</td>
<td>0.2289E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>7</td>
<td>0.1307E-02</td>
<td>0.2338E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>8</td>
<td>0.1214E-02</td>
<td>0.2701E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>9</td>
<td>0.5431E-02</td>
<td>0.2611E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>10</td>
<td>0.1404E-02</td>
<td>0.2748E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>11</td>
<td>0.1485E-02</td>
<td>0.3692E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>12</td>
<td>0.3403E-02</td>
<td>0.4070E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>13</td>
<td>0.6337E-02</td>
<td>0.3892E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>14</td>
<td>0.1450E-03</td>
<td>0.3004E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>15</td>
<td>0.5985E-04</td>
<td>0.3613E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>16</td>
<td>0.1214E-04</td>
<td>0.3004E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>17</td>
<td>0.2407E-04</td>
<td>0.2007E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>18</td>
<td>0.1502E-03</td>
<td>0.1979E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>19</td>
<td>0.1962E-03</td>
<td>0.1384E-03</td>
<td>0.11238E+03</td>
<td>-0.13414E+03</td>
<td>0.23395E-01</td>
<td>0.19799E-01</td>
<td>0.224E-01</td>
<td>0.210E-01</td>
<td>0.417E-02</td>
</tr>
<tr>
<td>#</td>
<td>Residuals</td>
<td>Lagrangian Multipliers</td>
<td>Ordered LOD of Current Cycles, Log. XNM.</td>
<td>CDF of LOD of Current Cycles, Log. XNM.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>-----------</td>
<td>------------------------</td>
<td>------------------------------------------</td>
<td>-----------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>43</td>
<td>0.2023E+03</td>
<td>0.1372E-06</td>
<td>0.2882E-03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>43</td>
<td>0.2056E+03</td>
<td>0.7092E-06</td>
<td>0.1276E-03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>50</td>
<td>0.1662E+03</td>
<td>0.9174E-07</td>
<td>0.1245E-03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>51</td>
<td>0.1086E+03</td>
<td>0.6297E-07</td>
<td>0.1268E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>52</td>
<td>0.1552E+03</td>
<td>0.8974E-07</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>53</td>
<td>0.9119E+03</td>
<td>0.8217E-04</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>54</td>
<td>0.1045E+03</td>
<td>0.3594E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>55</td>
<td>0.9119E+03</td>
<td>0.2052E-04</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>56</td>
<td>0.1045E+03</td>
<td>0.1629E-04</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>57</td>
<td>0.9119E+03</td>
<td>0.1069E-04</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>58</td>
<td>0.1045E+03</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>59</td>
<td>0.9119E+03</td>
<td>0.6474E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>60</td>
<td>0.7436E-04</td>
<td>0.1897E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>61</td>
<td>0.7436E-04</td>
<td>0.1629E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>62</td>
<td>0.7436E-04</td>
<td>0.1069E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>63</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>64</td>
<td>0.7436E-04</td>
<td>0.6474E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>65</td>
<td>0.7436E-04</td>
<td>0.5345E-09</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>66</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>67</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>68</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>69</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>70</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>71</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>72</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>73</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>74</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>75</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>76</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>77</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>78</td>
<td>0.7436E-04</td>
<td>0.8119E-05</td>
<td>0.1265E+03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SOLUTION FOUND

SUM OF RESIDUALS SQUARED = 0.87947E-12

THE MODIFIED LAGRANGIAN MULTIPLIERS ARE:

-0.39044043E+01
-0.44473114E+02
-0.14786617E+03
-0.20140271E+03

LAGRANGIAN MULTIPLIERS

<table>
<thead>
<tr>
<th>LOD OF CURRENT CYCLES, LOG XNM.</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1472E+04</td>
</tr>
<tr>
<td>0.9723E+01</td>
</tr>
<tr>
<td>0.9295E+01</td>
</tr>
<tr>
<td>0.8464E+01</td>
</tr>
<tr>
<td>0.9442E+01</td>
</tr>
</tbody>
</table>

ORDERED LOD OF CURRENT CYCLES, LOG. XNM.

<table>
<thead>
<tr>
<th>X AXIS PDF, CDF PLOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7743E+01</td>
</tr>
<tr>
<td>0.7681E+01</td>
</tr>
<tr>
<td>0.8459E+01</td>
</tr>
</tbody>
</table>

CDF PARAMETERS

<table>
<thead>
<tr>
<th>Y AXIS OF PDF, CDF PLOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.39044043E+01</td>
</tr>
</tbody>
</table>

ORIGINAL PAGE 12

OF POOR QUALITY
<table>
<thead>
<tr>
<th>ORDERED LOG OF CURRENT CYCLES</th>
<th>LOG AXM.</th>
<th>Y AXIS OF PDF, CDF PLOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9723E+01</td>
<td>0.9819E+01</td>
<td>0.9112E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9999E+01</td>
<td>0.7094E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9874E+01</td>
<td>0.9072E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9051E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9123E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9123E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9123E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9123E+01</td>
</tr>
<tr>
<td>0.9723E+01</td>
<td>0.9600E+01</td>
<td>0.9123E+01</td>
</tr>
</tbody>
</table>
File _DBAO: C:\PLOTZ.CPR 1 (363, 197, 0), last revised on 23-NOV-1988 11:21, is a 2 block sequential file owned by UIC [11,111]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

9.0 APPENDIX D
IMSL SUBROUTINE CALLS FROM RANDOM3 AND RANDOM4

RANDOM3

1. RNSET - Initializes a random seed for use in the IMSL random number generators.

2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.

3. RNRLNL - Generates pseudorandom numbers from a lognormal distribution.

4. DESPL - Performs nonparametric probability density function estimation by the penalized likelihood method.

5. GCDF - Evaluates a general continuous cumulative distribution function given the ordinates of the density.

RANDOM4

1. RNSET - Initializes a random seed for use in the IMSL random number generators.

2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.

3. RNRLNL - Generates pseudorandom numbers from a lognormal distribution.

125
APPENDIX E

SAMPLE SAS/GRAPH PROGRAM FOR RANDOM3 AND RANDOM4

data a;
INFILE 'PLOT1.CPR' FIRSTOBS=2; input x y;
GOPTIONS DEVICE=HP7470;
proc gplot;
axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
value=(h=1 f=simplex);
axis2 value=(h=1 f=simplex) label=none;
plot y*x / haxis=axis1 vaxis=axis2;
TITLE H=1 A=90 F=SIMPLEX 'PROBABILITY DENSITY FUNCTION';
symbol i=spline v=square;
data B;
INFILE 'PLOT2.CPR' FIRSTOBS=2; input x y;
proc gplot;
axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
value=(h=1 f=simplex);
axis2 value=(h=1 f=simplex) label=none;
plot y*x / haxis=axis1 vaxis=axis2;
TITLE H=1 A=90 F=SIMPLEX 'CUMULATIVE DISTRIBUTION FUNCTION';
symbol i=spline v=square;