Lower confidence interval bounds for coherent systems with cyclic components

Covington, Valerie A.
Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/34882

Downloaded from NPS Archive: Calhoun
THESIS

LOWER CONFIDENCE INTERVAL BOUNDS FOR COHERENT SYSTEMS WITH CYCLIC COMPONENTS

by

Valerie A. Covington

September, 1990

Thesis Advisor: W. Max Woods

Approved for public release; distribution is unlimited.
LOWER CONFIDENCE INTERVAL BOUNDS FOR COHERENT SYSTEMS WITH CYCLIC COMPONENTS

Valerie A. Covington

Master's Thesis

September 1990

Three lower confidence interval estimation procedures for system reliability of coherent systems with cyclic components are developed and their accuracy measured using Monte Carlo techniques. The procedures use either the Poisson approximation to the Binomial distribution, the lower Binomial confidence limit procedure, or a modified procedure using the Poisson approximation to the Binomial distribution to obtain an equation for the lower confidence limit. The accuracy of the interval estimators were evaluated using standard computer simulation methods for series, parallel, series-parallel, and Wheatstone Bridge systems. The method determined to be most accurate can be combined with similar procedures for components that have continuous failure times and applied to yield a lower confidence interval procedure for the reliability of coherent systems with cyclic and continuously operating components.
Approved for public release; distribution is unlimited.

Lower Confidence Interval
Bounds for Coherent Systems
With Cyclic Components

by

Valerie A. Covington
Lieutenant, United States Navy
B.A., University of South Florida, 1975
M.Ed., West Georgia College, 1979

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author:
Valerie A. Covington

Approved by:
W. Max Woods, Thesis Advisor
Lyn R. Whitaker, Second Reader
Peter Purdue, Chairman,
Department of Operations Research
ABSTRACT

Three lower confidence interval estimation procedures for system reliability of coherent systems with cyclic components are developed and their accuracy measured using Monte Carlo techniques. The procedures use either the Poisson approximation to the Binomial distribution, the lower Binomial confidence limit procedure, or a modified procedure using the Poisson approximation to the Binomial distribution to obtain an equation for the lower confidence limit. The accuracy of the interval estimators were evaluated using standard computer simulation methods for series, parallel, series-parallel, and Wheatstone Bridge systems. The method determined to be most accurate can be combined with similar procedures for components that have continuous failure times and applied to yield a lower confidence interval procedure for the reliability of coherent systems with cyclic and continuously operating components.
THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.
TABLE OF CONTENTS

I. INTRODUCTION .................................................................................. 1

II. PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY ......................................................... 5
   A. METHODOLOGY ............................................................................. 5
   B. RESULTS ....................................................................................... 8
      1. Series System ........................................................................... 8
      2. Series-Parallel Systems ........................................................... 10
      3. Series-Parallel Systems with a 2 of 3 Parallel Component .......... 10
      4. Parallel System ......................................................................... 12
      5. Wheatstone Bridge .................................................................. 13

III. ALTERNATE PROCEDURE A FOR THE LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY ...................... 15
    A. METHODOLOGY ........................................................................ 15
    B. RESULTS .................................................................................. 15

IV. ALTERNATE PROCEDURE B FOR THE LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY ...................... 18
    A. METHODOLOGY ........................................................................ 18
    B. RESULTS .................................................................................. 18

V. SIMULATION ...................................................................................... 21

VI. CONCLUSIONS AND RECOMMENDATIONS ................................................. 22

APPENDIX A. DISCRETE CONFIDENCE LIMIT PROPERTIES ..................................................... 23

APPENDIX B. INPUT PARAMETERS ............................................................. 25
APPENDIX C. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES SYSTEM AND WHEATSTONE BRIDGE SYSTEM) ................................................. 29

APPENDIX D. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES-PARALLEL SYSTEM) 43

APPENDIX E. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES-PARALLEL SYSTEM WITH A 2/3 COMPONENT) ........................................... 58

APPENDIX F. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (PARALLEL SYSTEM) ......... 73

APPENDIX G. FORTRAN CODE FOR ALTERNATE PROCEDURE A FOR ESTIMATING THE LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY ....................................................... 88

APPENDIX H. FORTRAN CODE FOR ALTERNATE PROCEDURE B FOR ESTIMATING THE LOWER CONFIDENCE LIMIT FOR SELECTED SYSTEMS ................................................................. 102

APPENDIX I. SUBROUTINES ........................................... 117

LIST OF REFERENCES .................................................. 121

BIBLIOGRAPHY .......................................................... 122

INITIAL DISTRIBUTION LIST ........................................... 123
LIST OF TABLES

Table 1. SERIES SYSTEM ......................................................... 9
Table 2. SERIES-PARALLEL SYSTEM ........................................... 10
Table 3. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM  11
Table 4. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM  (CONSTANT NUMBER OF MISSION TESTS, DECREASING RELIABILITY) .............................................. 12
Table 5. PARALLEL SYSTEM ..................................................... 13
Table 6. WHEATSTONE BRIDGE SYSTEM ...................................... 14
Table 7. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE  PROCEDURE A (SERIES SYSTEM) ..................................................... 16
Table 8. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE  PROCEDURE A (WHEATSTONE BRIDGE) ......................................... 17
Table 9. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE  PROCEDURE B (SERIES SYSTEM) ............................................... 19
Table 10. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE PROCEDURE B (WHEATSTONE BRIDGE) ................................. 20
Table 11. SERIES SYSTEM INPUT PARAMETERS ............................... 25
Table 12. SERIES-PARALLEL SYSTEM INPUT PARAMETERS .............................. 26
Table 13. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM INPUT PARAMETERS ............................................................ 27
Table 14. PARALLEL SYSTEM INPUT PARAMETERS ............................ 28
Table 15. WHEATSTONE BRIDGE INPUT PARAMETERS .......................... 28
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Series System</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Series-Parallel System</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Series Parallel With Two Out of Three Subcomponent System</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Parallel System</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Wheatstone Bridge</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Behavior of</td>
<td>24</td>
</tr>
</tbody>
</table>
I. INTRODUCTION

Coherent systems are those systems for which the system performs if all components function, the system fails if all the components fail, replacing a failed component with a working component does not cause the system to fail and similarly replacing a working component with a failed component does not cause the system to work [Ref. 1: p. 343]. The reliability of a coherent system is not reduced when the reliability of its components is increased. Cyclic components are those whose function is measured on a pass-fail basis and whose reliability is computed from a discrete probability distribution, usually the Geometric distribution.

The problem of obtaining confidence limits on the reliability of a coherent system based on data gathered on its individual components has attracted considerable interest. Confidence bounds for the reliability of series systems have been obtained asymptotically, based on methods such as Likelihood Ratio, Maximum Likelihood, or Modified Maximum Likelihood. Asymptotic methods are inaccurate at higher percentiles unless the component sample sizes are large. Bayesian methods have been developed for this problem, but they are extremely sensitive to the selection of the prior distribution. [Ref. 2: p. 21]

Exact confidence limits have been obtained for simple systems. This solution requires that the reliability of at least one of the components has to be greater than that of the system. However, identifying such a component becomes quite complex for all but simple systems composed of no more than two or three components [Ref. 3: p. 220]. Exact methods have also been developed for series systems using asymptotic approaches and the unbiased minimum variance estimators of the probability of success, $p$, based on Binomial data [Ref. 4: p. 782].

Unfortunately, none of the above referenced interval estimation procedures based on discrete data can be readily used in conjunction with the data of components that have continuous failure times. Therefore it is difficult to obtain interval estimates for the reliability of complex systems that have mixtures of cyclic components and components that operate continuously. The methods developed in this thesis can be combined with similar methods that use continuous data, namely those developed by Lee [Ref. 5]. The combination of these methods may provide interval estimation for the reliability of systems with cyclic and continuously operating components.
In this thesis, three procedures that provide lower confidence limits for the reliability of coherent systems with cyclic components were analyzed. These procedures use only discrete data. There is a problem with using component data to establish system reliability, especially for a system that has quite a bit of redundancy. Even though the tested components fail individually and their estimated reliabilities are moderate, had these components been assembled into a system, the system could very well have worked. In such a case, the system has a very high degree of reliability and methods that work well in estimating the more moderate component reliability will not work well in estimating the system reliability. Thus, each procedure that we study has modifications to accommodate component test data which when assembled into systems would exhibit zero system failures, one system failure, or more than one system failure, i.e.

- no component failures occur or only components fail that are redundant in the system, so that no system failure could occur if all the components were combined to form systems,
- exactly one component fails that would result in a system failure or redundant components in the system fail in a quantity, so that no more than one system failure could occur if the components were combined to form systems (this modification is explained in more detail in the following chapters),
- for any component, \( i \), more than one component fails out of \( n \), tested which would lead to more than one system failure.

The systems analyzed in this thesis are as follows:

- all components arranged in series (see Figure 1)

![Series System](image)

**Figure 1. Series System**

- all components arranged in series with at least one component consisting of two parallel subcomponents (see Figure 2)
Figure 2. Series-Parallel System

- all components arranged in series with at least one component consisting of two parallel subcomponents and at least one component in a two out of three subcomponent parallel arrangement (see Figure 3)

Figure 3. Series Parallel With Two Out of Three Subcomponent System

- a system with all components in a parallel arrangement (see Figure 4)

Figure 4. Parallel System
a five component redundant system commonly referred to as a Wheatstone Bridge (see Figure 5).

Figure 5. Wheatstone Bridge
II. PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY

A. METHODOLOGY

The procedure selected to determine the lower confidence limit of relatively simple complex systems uses a method employed by Lomnicki [Ref. 6: p. 109] and extended by Myhre and others [Ref. 2: p. 213]. It uses the Poisson approximation to the Binomial distribution when numerous component failures occur that would result in multiple system failures if the tested components were assembled into systems. In cases where zero system failures occur or only one system failure could result from the failed components, the Poisson approximation appears to be conservative and the lower confidence limit is computed using the Binomial distribution directly.

Suppose a coherent system has $k$ different types of components that are statistically independent. Test data is available on each type of component. The $n_i$ tests for component $i$ are assumed to be independent Bernoulli trials with probability of failure, $q_i$, for each test. Therefore the number of failures, $F_i$, in these $n_i$ tests has a Binomial distribution.

Any of the $k$ components, say component $m$, can be chosen as a base component and $q_i$ expressed as a fraction or multiple, $a_i$, of $q_m$. That is

$$q_i = a_i q_m. \quad (2.1)$$

Suppose it is appropriate to assume that the probability distribution of the number of failures of each component, $F_i$, can be approximated by the Poisson distribution with mean $n_i q_i$. If $F = \sum_{i=1}^{k} F_i$, then from Equation (2.1), the distribution of $F$ is approximately Poisson with parameter $q_m \sum_{i=1}^{k} n_i q_i$. Consequently the mean of $F$, $E(F)$, is given by

$$E(F) = q_m \sum_{i=1}^{k} n_i a_i \quad (2.2)$$

The system reliability, $R_s$, can be defined as a function of the unreliability, $q_m$ and the associated values $a_i = \frac{q_i}{q_m}$. That is,

$$R_s = h(q_m, a_1, a_2, ..., a_k) \quad (2.3)$$
By definition, \( h \) is non-increasing in \( q_m \), because the system is coherent. If \( a_1, a_2, \ldots, a_k \) are known, an approximate lower confidence limit, \( R_{u_{\alpha}(a)} \), may be obtained from an upper confidence limit, \( \hat{q}_{m,U(a)} \), for \( q_m \) by the equation

\[
\hat{R}_{s,L(a)} = h(\hat{q}_{m,U(a)}, a_1, a_2, \ldots, a_k)
\]  

(2.4)

When appropriate, the upper confidence limit, \( \hat{q}_{m,U(a)} \), may be obtained from the well-known upper confidence limit for the mean, \( \lambda \), of a Poisson distribution, namely, if \( F \) is distributed \( \text{POISSON}(\lambda) \) then

\[
\hat{\lambda}_{U(a)} = \frac{X^2_{2,2(1+F)}}{2}
\]  

(2.5)

where \( X^2_{2,2(1-F)} \) is the \((1 - \alpha)\) quantile of a Chi-square probability distribution with degrees of freedom equivalent to \( 2(1 + F) \), where \( F \) is the number of system failures. From Equation (2.2), substitution of \( q_m \sum_{i=1}^{k} n_i a_i \) for \( \lambda \) into Equation (2.5) gives

\[
q_{m,U(a)} = \frac{X^2_{2,2(1+F)}}{\sum_{i=1}^{k} n_i a_i}
\]  

(2.6)

If the \( a_i \) are unknown then an approximate upper confidence limit, \( \hat{q}_{m,U(a)} \), is given by

\[
\hat{q}_{m,U(a)} = \frac{X^2_{2,2(1+F)}}{\sum_{i=1}^{k} n_i \hat{a}_i}
\]  

(2.7)

where \( \hat{a}_i \) is an estimator for \( a_i, i = 1,2,\ldots,k \).

The Poisson approximation to the Binomial distribution is conservative when all \( F_i \) equal zero or redundant components of the system fail in such a way that results in zero system failures. In this case, let \( n^* \) represent the number of system tests equivalent to testing each component \( n_i \) times, \( i = 1,\ldots,k \). Then the distribution of \( s \), the number of system tests that would have been successful, is approximately \( \text{Binomial}(n^*, R_i) \). If \( s \) is distributed \( \text{BINOMIAL}(n^*, R_i) \) then we can use the binomial lower confidence limit

\[
\hat{R}_{s,L(a)} = n^* \sqrt{\alpha}
\]  

(2.8)
to compute the lower confidence limit for system reliability. The following two methods for the calculation of the number of equivalent system tests, \( n^* \), were selected and each applied separately using Equation (2.8).

\[
n_1^* = \min(n_1, n_2, \ldots, n_k)
\]

\[
n_2^* = \frac{\sum_{i=1}^{k} n_i}{k}
\]

(2.9)

(2.10)

where \( n_i \) is the number of tests of component \( i \).

Some instances of component failures could only result in one equivalent system failure if all of the tested components were combined into complete systems. In this case, we again define \( n^* \) and treat \( n^* \) system trials with one failure. The resulting lower confidence limit, \( \hat{R}_{L(a)} \), for system reliability is the solution for \( p = (1 - q) \) in the equation

\[
\alpha = \sum_{j=n^*-1}^{n^*} \binom{n^*}{j} (1 - q)^j q^{n^*-j}
\]

(2.11)

In the Wheatstone Bridge case, two or more component failures among the tested components could result in one system failure if the components are assembled into Wheatstone systems. In this case we set \( n^* = n_1^* \) for one interval procedure and \( n^* = n_2^* \) for the second interval procedure.

In a series system, \( n^* \) is equal to the number of tests performed on the failed single component, because the reliability of the system is determined largely by the reliability of the least reliable component. Since it is difficult to solve for \( p \) in Equation (2.11), an equivalent equation using the Snedecor F distribution is used. Thus,

\[
\hat{R}_{S, L(a)} = \frac{s}{s + (f + 1)F_{1-(a),2(f+1),2s}}
\]

(2.12)

where \( s \) is the number of system successes, \( f \) is the number of system failures, and \( F_{1-(a),2(f+1),2s} \) is the \( \alpha \) quantile of the Snedecor F distribution with \( 2(f+1) \) and \( 2s \) degrees of freedom [Ref. 7: p. 43].
B. RESULTS

The accuracy of this procedure was evaluated using computer simulations for each of the following systems described in Section A:

- series systems
- series systems where the second component is composed of two parallel subcomponents
- series systems where the second component is composed of two parallel subcomponents and the fourth component consists of a two of three subcomponent parallel arrangement
- parallel systems
- Wheatstone Bridge

Groups of test data were generated where the parameters, \( q \), and \( n \), were chosen to control the expected number of failures, \( E[F] = \sum n_q \). Confidence levels of 0.20 and 0.05 were used in each case. A total of 1000 replications were generated for each set of parameter values. Each replication produced one value of \( \hat{R}_{u,L} \). These 1000 values, \( \hat{R}_{u,L} \), were ordered and used to get the simulated probability distribution of \( \hat{R}_{u,L} \). The simulation procedures are described in Chapter IV. The 80\(^{th} \) and 95\(^{th} \) percentile point of the simulated probability distribution of \( \hat{R}_{u,L} \) was compared to \( R \), for determining the accuracy of the procedure. This comparison is made because \( \hat{R}_{u,L} \) is the lower 100(1 - \( \alpha \)) percentile confidence limit for \( R \), if \( 1 - \alpha = P(\hat{R}_{u,L} \leq R) \). This equation states that \( R \) is the 100(1 - \( \alpha \)) percentile point of the probability distribution of \( \hat{R}_{u,L} \). The "true confidence level" is the percentile point of the simulated distribution corresponding to the true value of \( R \).

The parameter values \( n_1 q_1, n_2 q_2, ..., n_k q_k \) determine a case number and are labeled as such in the tables that describe the simulation results. A summary table that provides the parameter values, \( q \), and \( n \), is given in Appendix B.

All tables report the 80\(^{th} \) and 95\(^{th} \) percentile points of the simulated distribution of \( \hat{R}_{u,L} \) and appear in the tables under the column labeled \( \hat{R}_{u,L} \).

1. Series System

By definition, the reliability, \( R_s \), for a series system of \( k \) independent components is

\[
R_s = \prod_{i=1}^{k} (1 - q_i) = \prod_{i=1}^{k} (1 - a_i q_i) \tag{2.13}
\]
The corresponding lower confidence limit is given by

\[ \hat{R}_{s,L(a)} = \prod_{i=1}^{n} \left( 1 - \hat{a}_i \hat{q}_{m,U(a)} \right) \]

(2.14)

These formulae are used to calculate the reliability when at least two components fail. If zero components fail Equation (2.8) is used and if one component fails Equation (2.12) is used. The results are presented in Table 1. In Table 1, column 1 of $\hat{R}_{s,L(a)}$ is calculated using Equation (2.10) and column 2 is calculated using Equation (2.9) for $n^a$ when the component failures equate to zero system failures.

<table>
<thead>
<tr>
<th>Case</th>
<th># Compts</th>
<th>$E[F]$</th>
<th>$R_s$</th>
<th>$\alpha$ Level</th>
<th>$\hat{R}_{s,L(a)}$</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>1.03</td>
<td>.95572</td>
<td>.20</td>
<td>.914477</td>
<td>.903457 75 75</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.851400</td>
<td>.851400 100 100</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>1.2</td>
<td>.93206</td>
<td>.20</td>
<td>.848910</td>
<td>.848910 100 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.851400</td>
<td>.851400 100 100</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>4.9</td>
<td>.95034</td>
<td>.20</td>
<td>.953610</td>
<td>.953610 76 76</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.959690</td>
<td>.959690 90 90</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>4.9</td>
<td>.85951</td>
<td>.20</td>
<td>.853690</td>
<td>.853690 85.2 85.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.867180</td>
<td>.860890 94.5 94.5</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5.74</td>
<td>.95084</td>
<td>.20</td>
<td>.950360</td>
<td>.950360 76.3 76.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.957799</td>
<td>.957799 91.1 91.1</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>5.75</td>
<td>.85951</td>
<td>.20</td>
<td>.859100</td>
<td>.859100 80.1 80.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.881140</td>
<td>.880720 92.5 92.6</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>10.5</td>
<td>.85828</td>
<td>.20</td>
<td>.871740</td>
<td>.871740 72.4 72.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.883660</td>
<td>.883660 86.6 86.6</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>10.5</td>
<td>.85951</td>
<td>.20</td>
<td>.871370</td>
<td>.871370 73.3 73.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.20</td>
<td>.890490</td>
<td>.890490 86.4 86.4</td>
</tr>
</tbody>
</table>

In some cases the true confidence level equals 100. This is not uncommon for confidence intervals based on discrete data and is further explained in Appendix A.
### 2. Series-Parallel Systems

A modification was made to the series systems to form the series-parallel system. The second component in the series was modified to consist of two parallel subcomponents of equal unreliability, $q_2$. For the second component to fail both subcomponents must fail. The unreliability of component two is $q_2^2$. Since component two is in series with the other components the corresponding lower confidence limit is calculated using Equation (2.8) when the equivalent number of system failures is zero, Equation (2.12) when the number of system failures is one, and Equation (2.14) is used in all other cases.

The results are presented in Table 2. In Table 2, column 1 of $\hat{R}_{L,t(0)}$ is calculated using Equation (2.10) and column 2 is calculated using Equation (2.9) for $n^*$ when the component failures equate to zero system failures.

### Table 2. SERIES-PARALLEL SYSTEM

<table>
<thead>
<tr>
<th>Case</th>
<th># Compts</th>
<th>$E[F]$</th>
<th>$R_1$</th>
<th>$a$ Level</th>
<th>$\hat{R}_{L,t(0)}$</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>5</td>
<td>1.03</td>
<td>.95572</td>
<td>.20</td>
<td>.9447</td>
<td>.90345</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>.85140</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>2.72</td>
<td>.93206</td>
<td>.20</td>
<td>.95770</td>
<td>.95770</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.93344</td>
<td>.93344</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>4.9</td>
<td>.95828</td>
<td>.20</td>
<td>.95369</td>
<td>.95369</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.86685</td>
<td>.86089</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>4.9</td>
<td>.93951</td>
<td>.20</td>
<td>.93546</td>
<td>.93546</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.86685</td>
<td>.86685</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
<td>9.79</td>
<td>.95084</td>
<td>.20</td>
<td>.95321</td>
<td>.95321</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.95779</td>
<td>.96029</td>
</tr>
<tr>
<td>14</td>
<td>10</td>
<td>10.5</td>
<td>.95951</td>
<td>.20</td>
<td>.95943</td>
<td>.86943</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.88606</td>
<td>.88606</td>
</tr>
</tbody>
</table>

Note the accuracy of the lower confidence interval is somewhat reasonable when the expected number of failures exceeds 4.9 in both the series and series-parallel systems.

### 3. Series-Parallel Systems with a 2 of 3 Parallel Component

Another modification was made to the five and ten component series systems. The third component in the series was modified and now consists of three parallel subcomponents of equal unreliability, $q_3$. For this component to fail, two or three parallel components must fail. The unreliability, $q_3$, of component three is $(\frac{2}{3})(1 - q_3)q_3^2 + q_3^3$. 
The fourth component of each system is composed of two parallel subcomponents as defined in the series-parallel system. Since these components are in series with the other components, the corresponding lower-confidence limit is calculated using the same series of equations as the series-parallel system. The results are presented in Table 3. The term $n^*$ is computed from Equation (2.10) when the equivalent number of system failures is zero.

Table 3. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM

<table>
<thead>
<tr>
<th>Case</th>
<th># Compts</th>
<th>$E[F]$</th>
<th>$R_s$</th>
<th>$\alpha$ Level</th>
<th>$\hat{R}_{s;10}$</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>5</td>
<td>.88</td>
<td>.96525</td>
<td>.20</td>
<td>.91447</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>100</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>2.6</td>
<td>.94136</td>
<td>.20</td>
<td>.95770</td>
<td>63.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.95344</td>
<td>85.7</td>
</tr>
<tr>
<td>17</td>
<td>5</td>
<td>3.21</td>
<td>.96035</td>
<td>.20</td>
<td>.95937</td>
<td>80.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.96877</td>
<td>88.5</td>
</tr>
<tr>
<td>18</td>
<td>10</td>
<td>4.6</td>
<td>.88945</td>
<td>.20</td>
<td>.88582</td>
<td>80.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.91735</td>
<td>89.5</td>
</tr>
<tr>
<td>19</td>
<td>5</td>
<td>6.17</td>
<td>.96035</td>
<td>.20</td>
<td>.96126</td>
<td>78.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.96886</td>
<td>90.5</td>
</tr>
<tr>
<td>20</td>
<td>10</td>
<td>8.37</td>
<td>.88521</td>
<td>.20</td>
<td>.89734</td>
<td>76.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.91817</td>
<td>87.8</td>
</tr>
</tbody>
</table>

Another series of simulations were conducted on these systems. For each case previously run, new cases were defined by decreasing the unreliability of selected components without changing the number of component tests. These changes result in a decrease in the number of expected failures and an increase in the reliability of the system. The results are presented in Table 4. The term $n^*$ is computed from Equation (2.10) when the equivalent number of system failures is zero.
Table 4. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM (CONSTANT NUMBER OF MISSION TESTS, DECREASING RELIABILITY)

<table>
<thead>
<tr>
<th>Case</th>
<th># Compts</th>
<th>$E[F]$</th>
<th>$R_j$</th>
<th>$\alpha$ Level</th>
<th>$R_{(loo)}$</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>10</td>
<td>.51</td>
<td>.99136</td>
<td>.20</td>
<td>.97661</td>
<td>.97661</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.95690</td>
<td>.95690</td>
</tr>
<tr>
<td>22</td>
<td>10</td>
<td>1.68</td>
<td>.96836</td>
<td>.20</td>
<td>.97661</td>
<td>.97661</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.95690</td>
<td>.95690</td>
</tr>
<tr>
<td>23</td>
<td>10</td>
<td>6.57</td>
<td>.95430</td>
<td>.20</td>
<td>.95497</td>
<td>.95497</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.96169</td>
<td>.96169</td>
</tr>
<tr>
<td>24</td>
<td>10</td>
<td>9.90</td>
<td>.90249</td>
<td>.20</td>
<td>.90515</td>
<td>.90515</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.91486</td>
<td>.91486</td>
</tr>
</tbody>
</table>

Tables 3 and 4 indicate that for $E(F) > 5$ the lower confidence limits are reasonably accurate.

4. Parallel System

The accuracy of the lower confidence procedure was evaluated when it was applied to a five component system with each component, $i$, in parallel with the others. The unreliabilities, $q$, of each component are equal and the unreliability of the system is defined as $q^5$. For the system to fail all five components must fail. The results of the computer simulations are presented in Table 5. The term $n^0$ is computed from Equation (2.10) when the equivalent number of system failures is zero. If $n$ of these components are tested and four or fewer failures occur, then no system failures would have occurred had these components been assembled into systems. The lower confidence limit is calculated using Equation (2.8) when the equivalent number of system failures is zero, Equation (2.12) when the number of system failures is one, and Equation (2.14), with $k = 1$, in all other cases.
5. Wheatstone Bridge

By definition, the reliability, $R_s$, for the Wheatstone Bridge, in Figure 5, with five independent components is

$$R_s = p_1p_2 + p_2p_3 + p_1p_3p_5 + p_2p_3p_4 - p_1p_2p_3p_5 - p_1p_3p_4p_5 - p_2p_3p_4p_5 + 2p_1p_2p_3p_4p_5$$

(2.16)

where $p_i = (1 - q_i)$. In terms of $q$, Equation (2.16) becomes

$$R_s = h(q_m, q_1, ..., q_5)$$

$$= 1 - q_m^2(a_1a_2 + a_4a_5) - q_m^3(a_1a_3a_5 + a_2a_3a_4)$$

$$+ q_m^2(a_1a_2a_3a_5 + a_1a_2a_5 + a_1a_2a_4a_5 + a_1a_3a_4a_5 + a_2a_3a_4a_5)$$

$$- 2q_m^3(a_1a_2a_3a_4a_5)$$

(2.17)

[Ref. 2: p. 215]. By substituting $\hat{q}_m, \hat{u}_(u), \hat{u}_v$, for $q$, the corresponding lower confidence limit is obtained.

$$\hat{R}_{s, u(a)} = h(\hat{q}_m, u(a), \hat{u}_1, ..., \hat{u}_5)$$

(2.18)
The reliability of the Wheatstone Bridge system is normally high due to the redundancy of the system. This system experiences zero system failures in the following five different failure patterns.

- \( F_1 = F_2 = F_3 = F_4 = F_5 = 0 \)
- \( F_1 = F_3 = F_5 = 0 \)
- \( F_1 = F_4 = 0 \)
- \( F_2 = F_5 = 0 \)
- \( F_2 = F_3 = F_4 = 0 \)

where \( F_i \) is the number of failures of component \( i \), among its \( n_i \) tests. Any other failure patterns will produce one or more system failures. The lower confidence limit is calculated using Equation (2.8) when the equivalent number of system failures is zero, Equation (2.12) when the number of system failures is one, and Equation (2.18) in all other cases. The results of the computer simulations are presented in Table 6. In Table 6, column 1 of \( \hat{R}_{\alpha,t(n)} \) is calculated using Equation (2.10) and column 2 is calculated using Equation (2.9) for \( n^* \) when the component failures equate to zero system failures.

### Table 6. WHEATSTONE BRIDGE SYSTEM

<table>
<thead>
<tr>
<th>Case</th>
<th># Compts</th>
<th>( E[F] )</th>
<th>( R_i )</th>
<th>( \alpha ) Level</th>
<th>( \hat{R}_{\alpha,t(n)} )</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>5</td>
<td>5.75</td>
<td>.99776</td>
<td>.20</td>
<td>.99658</td>
<td>88.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.99766</td>
<td>95.4</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>6</td>
<td>.99976</td>
<td>.20</td>
<td>.99960</td>
<td>91.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.99966</td>
<td>98.3</td>
</tr>
<tr>
<td>31</td>
<td>5</td>
<td>.5</td>
<td>.99977</td>
<td>.20</td>
<td>.91447</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.98678</td>
<td>100</td>
</tr>
<tr>
<td>32</td>
<td>5</td>
<td>.5</td>
<td>.99977</td>
<td>.20</td>
<td>.92622</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>.05</td>
<td>.99339</td>
<td>100</td>
</tr>
</tbody>
</table>

The system reliability values in cases 31 and 32 are too large for these interval estimation procedures to be accurate using the sample sizes given in Table 15 in Appendix B. Also, more than 1000 replications would be needed to assess the accuracy of any lower confidence limit procedure for system reliability when the true system reliability is as large as it is in these cases.
III. ALTERNATE PROCEDURE A FOR THE LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY

A. METHODOLOGY

This procedure determines the lower confidence limit for simple and complex systems using only the Poisson approximation to the Binomial distribution. It is used extensively by Bellini [Ref. 8: p.4-6].

This procedure uses the principles outlined in Chapter II of this thesis except in the cases where there are zero or one system failure. In the case where there are zero system failures (no components fail) the estimated unreliability, $\hat{q}_n$, becomes zero because $\hat{q}_n = \frac{F_i}{N_i}$ where $F_i$ is the number of failures in $n_i$ mission tests of component $i$. Therefore, the value of $\hat{q}_n$, the estimated value of $q_n$, becomes zero and Equation (2.7) becomes undefined. When this occurs the estimated lower confidence limit of the system reliability, $\hat{R}_s.L(n)$, is defined as

$$\hat{R}_s.L(n) = 1 - \frac{X^2_{x,2}}{2n^*}$$

(3.1)

where $n^*$ is defined as in Equation (2.10).

If only component, $m$, fails the value of $\hat{q}_n$ is equal to 1. The $\hat{q}_n$ for all other components are zero, and the lower confidence limit, $\hat{R}_s.L(n)$, is defined as

$$\hat{R}_s.L(n) = 1 - \frac{X^2_{x,2(1+n)}}{2n_m}$$

(3.2)

where $n_m$ is the number of mission tests of the failed component.

B. RESULTS

The accuracy of this procedure was evaluated only for series and Wheatstone Bridge systems. Testing was limited because a comparison of results of the three procedures, discussed in this thesis, indicated this procedure to be less accurate in determining the lower confidence limit for the reliability of simple systems. This observation is noted in those systems experiencing zero system failures or one system failure. Selected results illustrating the accuracy of this procedure compared to the accuracy of the "preferred
procedure" (Chapter II) are indicated in Tables 7 and 8. The term \( n^* \) is computed from Equation (2.10) when the equivalent number of system failures is zero.

Table 7. **COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE PROCEDURE A (SERIES SYSTEM)**

<table>
<thead>
<tr>
<th>Case</th>
<th>( R_s )</th>
<th>( \alpha ) Level</th>
<th>( \hat{R}_{s(t)}' )</th>
<th>True Confidence Level</th>
<th>( \hat{R}_{s(t)}' )</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.95572</td>
<td>.20</td>
<td>.91447</td>
<td>75</td>
<td>.83140</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>100</td>
<td>.92356</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>.93206</td>
<td>.20</td>
<td>.88491</td>
<td>100</td>
<td>.83140</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>100</td>
<td>.92356</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>.95084</td>
<td>.20</td>
<td>.95361</td>
<td>75</td>
<td>.95361</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.95969</td>
<td>90</td>
<td>.95969</td>
<td>90</td>
</tr>
<tr>
<td>4</td>
<td>.85951</td>
<td>.20</td>
<td>.85369</td>
<td>85.2</td>
<td>.85119</td>
<td>85.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.86718</td>
<td>94.5</td>
<td>.85083</td>
<td>95.2</td>
</tr>
<tr>
<td>5</td>
<td>.95084</td>
<td>.20</td>
<td>.95036</td>
<td>76.3</td>
<td>.95036</td>
<td>80.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.95779</td>
<td>91.1</td>
<td>.96359</td>
<td>89.6</td>
</tr>
<tr>
<td>6</td>
<td>.85951</td>
<td>.20</td>
<td>.85910</td>
<td>80.1</td>
<td>.86910</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.88114</td>
<td>92.5</td>
<td>.88072</td>
<td>92.9</td>
</tr>
<tr>
<td>7</td>
<td>.85828</td>
<td>.20</td>
<td>.87174</td>
<td>72.4</td>
<td>.87174</td>
<td>72.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.88366</td>
<td>86.6</td>
<td>.88366</td>
<td>86.6</td>
</tr>
<tr>
<td>8</td>
<td>.85951</td>
<td>.20</td>
<td>.87137</td>
<td>73.3</td>
<td>.87137</td>
<td>73.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.89049</td>
<td>86.4</td>
<td>.89049</td>
<td>86.4</td>
</tr>
</tbody>
</table>
This alternate procedure produced lower confidence limits that were more conservative than those produced by the “preferred procedure.”
IV. ALTERNATE PROCEDURE B FOR THE LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY

A. METHODOLOGY

The purpose of this alternate procedure was to construct a procedure that accommodates zero component failures in a different manner than that employed in other procedures. This procedure redefines the estimated value, $\hat{a}$, of $a$, and in so doing uses the Poisson approximation to the Binomial distribution.

Suppose a component, $i$, undergoes $n_i$ tests where there is a probability of success, $p_i$, on each component test. Let $F_i$ be the number of failures in these $n_i$ tests. We define $F = \sum F_i$.

The 50 percent lower binomial confidence limit for the reliability of component $i$, $\hat{P}_{iL}$, can be determined by using its number of failures, $F_i$, and number of tests, $n_i$. The upper Binomial confidence limit, $\hat{q}_{iU}$, is equal to $1 - \hat{P}_{iL}$. In this alternate procedure we define $\hat{a}_i$ by

$$\hat{a}_i = \frac{\hat{q}_{iU}}{\hat{q}_{mU}}$$

(4.1)

where $\hat{q}_{mU} = \max(\hat{q}_{1U}, \hat{q}_{2U}, ..., \hat{q}_{nU})$. Note that the index will be determined by the data and $\hat{a}_i$ is well defined even if no components fail. The probability distribution of $F_i$ is approximated by the Poisson distribution. The estimated upper confidence limit, $\hat{q}_{mU}$, can be calculated using Equation (2.7) and the system reliability lower confidence limit, $\hat{P}_{sL}$, can be obtained from Equation (2.4) where $\hat{a}_i$ is defined by Equation (4.1).

This procedure is used without exception, regardless of the number of system failures.

B. RESULTS

This procedure was evaluated only on series and Wheatstone Bridge systems. Evaluations were limited because a comparison of the results with other procedures discussed in this thesis, indicated this procedure to be less accurate. Selected results illustrating the accuracy of this procedure compared to the accuracy of the "preferred procedure" (Chapter II) are indicated in Tables 9 and 10. The term $n^*$ is computed for Equation (2.10) when the equivalent number of system failures is zero.
Table 9. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE PROCEDURE (SERIES SYSTEM)

<table>
<thead>
<tr>
<th>Case</th>
<th>$R_s$</th>
<th>$\alpha$ Level</th>
<th>$\hat{R}_{s,L(t)}$</th>
<th>True Confidence Level</th>
<th>$\hat{R}_{s,L(t)}$</th>
<th>True Confidence Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.95572</td>
<td>.20</td>
<td>.91447</td>
<td>75</td>
<td>.97788</td>
<td>37.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>100</td>
<td>.99489</td>
<td>7.4</td>
</tr>
<tr>
<td>2</td>
<td>.93206</td>
<td>.20</td>
<td>.88491</td>
<td>100</td>
<td>.97197</td>
<td>70.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.85140</td>
<td>100</td>
<td>.99350</td>
<td>33.2</td>
</tr>
<tr>
<td>3</td>
<td>.95084</td>
<td>.20</td>
<td>.95361</td>
<td>76</td>
<td>.97294</td>
<td>25.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.95969</td>
<td>90</td>
<td>.98732</td>
<td>9.7</td>
</tr>
<tr>
<td>4</td>
<td>.85951</td>
<td>.20</td>
<td>.85369</td>
<td>85.2</td>
<td>.93224</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.86718</td>
<td>94.5</td>
<td>.97896</td>
<td>8.3</td>
</tr>
<tr>
<td>5</td>
<td>.95084</td>
<td>.20</td>
<td>.95036</td>
<td>76.3</td>
<td>.97497</td>
<td>30.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.95779</td>
<td>91.1</td>
<td>.99155</td>
<td>9.2</td>
</tr>
<tr>
<td>6</td>
<td>.85951</td>
<td>.20</td>
<td>.85910</td>
<td>80.1</td>
<td>.91470</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.88114</td>
<td>92.5</td>
<td>.97405</td>
<td>7.1</td>
</tr>
<tr>
<td>7</td>
<td>.85828</td>
<td>.20</td>
<td>.87174</td>
<td>72.4</td>
<td>.89848</td>
<td>44.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.88366</td>
<td>86.6</td>
<td>.94607</td>
<td>19.4</td>
</tr>
<tr>
<td>8</td>
<td>.85951</td>
<td>.20</td>
<td>.87137</td>
<td>73.3</td>
<td>.88412</td>
<td>56.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.89049</td>
<td>86.4</td>
<td>.93886</td>
<td>24.9</td>
</tr>
</tbody>
</table>

Cases 4, 5, and 6 in Table 9 clearly illustrate a more accurate "preferred procedure" for a series system.
Table 10. COMPARISON OF PREFERRED PROCEDURE AND ALTERNATE PROCEDURE B (WHEATSTONE BRIDGE)

<table>
<thead>
<tr>
<th>Case</th>
<th>$R_s$</th>
<th>$\alpha$</th>
<th>Preferred Procedure</th>
<th>Alternate Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$\hat{R}_{0.01}$</td>
<td>$\hat{R}_{0.01}$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>True Confidence Level</td>
<td>True Confidence Level</td>
</tr>
<tr>
<td>29</td>
<td>.99776</td>
<td>.20</td>
<td>.99658</td>
<td>98123</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.99766</td>
<td>98235</td>
</tr>
<tr>
<td>30</td>
<td>.99976</td>
<td>.20</td>
<td>.99960</td>
<td>99810</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.99966</td>
<td>99856</td>
</tr>
<tr>
<td>31</td>
<td>.99977</td>
<td>.20</td>
<td>.91447</td>
<td>.90478</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.98678</td>
<td>.99900</td>
</tr>
<tr>
<td>32</td>
<td>.99977</td>
<td>.20</td>
<td>.92622</td>
<td>.90932</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.05</td>
<td>.99339</td>
<td>.99221</td>
</tr>
</tbody>
</table>

In the case of the Wheatstone Bridge, Table 10 illustrates that the "preferred procedure" is more accurate than the "alternate procedure".
V. SIMULATION

Standard simulation techniques are used to determine the accuracy of the lower confidence limit procedures. Some of the basic simulation programs were developed by Bellini [Ref. 8: Appendix A]. Each program was modified to incorporate the necessary mathematical formulae that are needed to define the lower confidence limit for a particular procedure.

Input parameters needed to run the computer programs are

- $k$ - number of components in the system
- $\bar{n}$ - vector of component tests ($n_1, n_2, ..., n_k$)
- $\bar{q}$ - vector of component unreliabilities ($q_1, q_2, ..., q_k$)
- $\alpha$ - level of confidence

Note that the value of system reliability, $R_s$, is determined by the vector $\bar{q}$.

The NON-IMSL random number generator, SRND, was used to simulate the success or failure of each test of the $k$ components. From this data, the values of $\hat{q}_n$, $\hat{a}_n$, $\hat{q}_{n,0}(a)$, and $\hat{R}_{s,L}(\alpha)$ are calculated. Each scenario was replicated 1000 times to generate 1000 ordered values of $\hat{R}_{s,L}(\alpha)$. The 1000$(1 - \alpha)^{th}$ ordered value, $\hat{R}_{s,L}(100(1 - \alpha))$, from smallest to largest denotes the $100(1 - \alpha)^{th}$ percentile point of the probability distribution of $\hat{R}_{s,L}(\alpha)$. If the lower confidence limit procedure is exact, $\hat{R}_{s,L}(100(1 - \alpha))$ should equal $R_s$. The percentiles in all of the tables are the values of $\hat{R}_{s,L}(100(1 - \alpha))$. A true confidence limit is then determined by finding the element of the vector of replications which is closest to $R_s$ and noting its index number, $j$. The true confidence level is then calculated to be $\frac{j}{1000} \times 100$.

A system report is generated and reports to the analyst the following information.

- $q_i$ - unreliability of each component, $i$
- $a_i$ - fraction of unreliability of base component, $m$
- $n_i$ - number of mission tests for each component, $i$
- $R_s$ - true system reliability
- $\hat{R}_{s,L}(\alpha)$ - estimated lower confidence limit for the $100(1 - \alpha)$ percentile
- difference between $R_s$ and $\hat{R}_{s,L}(\alpha)$
- true confidence level
VI. CONCLUSIONS AND RECOMMENDATIONS

The accuracies of three approximate interval estimation procedures, based on discrete component data, for the reliability of coherent systems were analyzed in this thesis. Computer simulations were used to perform this analysis. Each interval estimation method evaluated ratios of component failure rate estimates when two or more different component types have at least one failure. This specific ratio feature is needed to extend this work to more complex systems with mixtures of cyclic components and components whose failure times have a continuous probability distribution.

The simulations reveal that the method labeled the "preferred method" in this thesis appears to be reasonably accurate if four or more failures are expected to occur among all components tested. However, any general interpretation of this type is not warranted at this time. The potential for error can be significant if several components have relatively small samples (less than 15) and zero failures. Zero failures joined with small sample sizes will always be the bane of classical interval estimation procedures.

The ratio procedure does allow the possible use of information extraneous to the data. Previous test programs on similar hardware operating under similar environments, as that present for the current test data, might be used to modify the component failure rate estimates or perhaps only the ratios of the failure rate estimates. So called "off the shelf" hardware purchased in accordance with existing DOD specifications would be prime candidates for this type of failure rate modification. Supplementing current test data with other existing "similar" data has become more common as resources for reliability demonstration testing has been reduced.
APPENDIX A. DISCRETE CONFIDENCE LIMIT PROPERTIES

Equations for confidence limits on parameters of discrete probability distributions are not exact. If \( \hat{p}_{L(0)} \) is the lower 100(1 - \( \alpha \)) percent confidence limit for the parameter \( p \) in the Binomial distribution, then \( \hat{p}_{L(0)} \) is defined so that

\[
P(\hat{p}_{L(0)} < p) \geq 1 - \alpha \tag{A1}
\]

If the parameter \( p \) is the probability of success on each trial in a sample of size \( n \) and \( s \) is the observed number of successes then \( \hat{p}_{L(0)} \) is the solution for \( p \) in the equation

\[
\sum_{j=s}^{n} \binom{j}{s} p^j (1-p)^{n-j} = \alpha \tag{A2}
\]

if \( s > 0 \), and \( \hat{p}_{L(0)} = 0 \) if \( s = 0 \). Specifically, suppose \( s = n \) then in Equation (A2), \( \hat{p}_{L(0)} = \sqrt[n]{\alpha} \). This is the largest value of \( \hat{p}_{L(0)} \). Consequently if the true value of \( p \) is greater than \( \sqrt[n]{\alpha} \) then \( P(\hat{p}_{L(0)} \leq p) = 1 \). This has important implications when analyzing computer simulations of confidence limit procedures based on discrete data to assess their accuracy. If the value of \( p \) used to generate the data on the computer is greater than \( \sqrt[n]{\alpha} \), then all of the \( \hat{p}_{L(0)} \) values will be smaller than \( p \) and the analysis will show the procedure has confidence level 100. This is to be expected when evaluating these confidence interval procedures for some choices of sample sizes and parameter values.

The exact value of \( P(\hat{p}_{L(0)} \leq p) \) depends on the sample size, \( n \), and the true value of \( p \). For fixed \( n \), the possible values of \( s \) are 0,1,2,...,\( n \). Each value of \( s \) yields a specific value of \( \hat{p}_{L(0)} \), say \( p(s) \). Consequently,

\[
P(\hat{p}_{L(0)} = p(s) \mid p) = P(S = s \mid p) \tag{A3}
\]

and

\[
P(\hat{p}_{L(0)} < p(s) \mid p) = P(S < s \mid p) \tag{A4}
\]

If the true value of \( p \) equals \( \hat{p}_{L(0)}(s) \) for some \( s \) then, the probability in Equation (A4) has the value \( 1 - \alpha \) because this value of \( p \) satisfies Equation (A2). Consequently if the true value of \( p \) equals any of the values \( p(n), p(n-1), ..., p(1) \), then \( P(\hat{p}_{L(0)} < p) = 1 - \alpha \).
For all other values of $p$, $P(\hat{p}_{LB} < p) > 1 - \alpha$. Figure 6 is a sketch of the behavior of this phenomena for $\alpha = .10$ and $n = 5$.

Figure 6. Behavior of $P(\hat{p}_{LB} < p)$
## APPENDIX B. INPUT PARAMETERS

### Table 11. SERIES SYSTEM INPUT PARAMETERS

<table>
<thead>
<tr>
<th>Case</th>
<th>Components</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$q_i$</td>
<td>.02</td>
<td>.01</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>10</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$q_i$</td>
<td>.02</td>
<td>.01</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>15</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>$q_i$</td>
<td>.01</td>
<td>.005</td>
<td>.003</td>
<td>.008</td>
<td>.025</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>200</td>
<td>400</td>
<td>720</td>
<td>265</td>
<td>80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$q_i$</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>100</td>
<td>50</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>100</td>
<td>50</td>
<td>30</td>
<td>25</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>$q_i$</td>
<td>.01</td>
<td>.005</td>
<td>.003</td>
<td>.008</td>
<td>.025</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>150</td>
<td>80</td>
<td>240</td>
<td>265</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>$q_i$</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>150</td>
<td>20</td>
<td>20</td>
<td>50</td>
<td>25</td>
<td>150</td>
<td>20</td>
<td>20</td>
<td>50</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>$q_i$</td>
<td>.01</td>
<td>.02</td>
<td>.03</td>
<td>.04</td>
<td>.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>300</td>
<td>20</td>
<td>20</td>
<td>100</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>$q_i$</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
<td>300</td>
<td>20</td>
<td>20</td>
<td>100</td>
<td>50</td>
<td>300</td>
<td>20</td>
<td>20</td>
<td>100</td>
<td>50</td>
</tr>
<tr>
<td>Case</td>
<td>Components</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></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>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( q_i )</td>
<td>.02</td>
<td>.01</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>10</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>( q_i )</td>
<td>.02</td>
<td>.01</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td>.005</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>100</td>
<td>30</td>
<td>15</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>( q_i )</td>
<td>.01</td>
<td>.02</td>
<td>.03</td>
<td>.04</td>
<td>.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>100</td>
<td>50</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>( q_i )</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>100</td>
<td>50</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>100</td>
<td>50</td>
<td>30</td>
<td>25</td>
<td>20</td>
</tr>
<tr>
<td>13</td>
<td>( q_i )</td>
<td>.01</td>
<td>.005</td>
<td>.003</td>
<td>.008</td>
<td>.025</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>200</td>
<td>400</td>
<td>720</td>
<td>265</td>
<td>80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>( q_i )</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
<td>.005</td>
<td>.01</td>
<td>.015</td>
<td>.02</td>
<td>.025</td>
</tr>
<tr>
<td></td>
<td>( n_i )</td>
<td>300</td>
<td>20</td>
<td>20</td>
<td>100</td>
<td>50</td>
<td>300</td>
<td>20</td>
<td>20</td>
<td>100</td>
<td>50</td>
</tr>
</tbody>
</table>
Table 13. SERIES-PARALLEL WITH A 2 OUT OF 3 COMPONENT SYSTEM
INPUT PARAMETERS

<table>
<thead>
<tr>
<th>Case</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>16</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>17</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>18</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>19</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>20</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>21</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>22</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>23</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>24</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
</tbody>
</table>
Table 14. PARALLEL SYSTEM INPUT PARAMETERS

<table>
<thead>
<tr>
<th>Case</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>25</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>26</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>27</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>28</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
</tbody>
</table>

Table 15. WHEATSTONE BRIDGE INPUT PARAMETERS

<table>
<thead>
<tr>
<th>Case</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>30</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>31</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
<tr>
<td>32</td>
<td>$q_i$</td>
</tr>
<tr>
<td></td>
<td>$n_i$</td>
</tr>
</tbody>
</table>
APPENDIX C. FORTRAN CODE FOR THE PREFERRED LOWER
CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES SYSTEM
AND WHEATSTONE BRIDGE SYSTEM)

PROGRAM ZFYSCN
******************************************************************************
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE
* ZERO FAILURES ALLOWED; NO SCALING
* AUTHOR: E. F. BELLINI, LT, USN
* MODIFIED BY: LT VALERIE A. COVINGTON, USN (MAR 90)
* DATE: NOV 89
* *
* THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY
* OF THEIR COMPONENTS
* *
* IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'Bl EXEC'.
* THE REXX EXEC PROGRAM
* 'Bl' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT
* FILES RESULTING FROM THE 12 CONSECUTIVE RUNS. BY EDITING THE
* INDEX COUNTERS I, J, K OF THE 'Bl' EXEC ONE CAN RUN ANY USER-
* SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.
* *
* VARIABLES USED
* *
* AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT
* AI : INPUT WEIGHTS FOR EACH COMPONENT
* ALFA : LEVELS OF SIGNIFICANCE
* BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION
* CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE
* CIC15 : FORMAT LABEL
* DEGFR : DEGREES OF FREEDOM
* DELBRG : DIFFERENCE FOR BRIDGE SYSTEM
* DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM
* DELTAR : DIFFERENCE FOR SERIES SYSTEM
* DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.)
* EPS : SMALL QUANTITY(CONSTANT)
* ERROR : PARAMETER FOR IMSL ROUTINE
* FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE
* FI : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)
* FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS
* INC : INCREMENT STEP SIZE FOR ROUTINE USMNX
* KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS
* LOOP : COUNTS NO. OF REPLICATION PERFORMED
* MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)
* *
* 29
PARAMETER (KK=10, MAXALF=2, NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=.000001)
REAL*4 UNIRV(15,1000),TEMP(1000),QI(KK),AI(KK),AHATI(KK)
REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5),ALFA(5),SUMNAI,RSHAT(MAXALF,MAXREP),RS
REAL*4 KEY1(MAXREP),KEY2(MAXREP),KEY3(MAXREP),TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF,MAXREP),CHISQ(MAXALF,MAXREP)
REAL*4 QUPAI(MAXREP), QUPA2(MAXREP),RHTSTR(MAXALF,MAXREP)
REAL*4 DELTA(MAXALF), TRANQ(MAXREP),TRANSR(MAXREP),DIFF(MAXREP)
REAL*4 DELST(MAXALF),NIMIN,NMAX,NIREAL(KK)
REAL*4 RSHTR(MAXALF,MAXREP),DELBRG(MAXALF),KEY4(MAXREP)
REAL*4 TRANR(MAXREP), RSBRDG ,MSTRQ
REAL*4 ZFREP
REAL*4 AVGN, SUC, STUD

INTEGER SEED, MULT, SORT, TRAILS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTK, TOTREP
INTEGER CIC15, REPShD, SELCTB, ALF, FLAG, LOOP, PRNT
INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS, INC
INTEGER NTEST, FCT, BFLAG

CHARACTER*8 LOOPSO(MAXREP)

DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
DATA ALFA/.20,.050/
ASSIGN 8 TO CIC15
ASSIGN 9 TO REPShD

* CALL COMPRS
PRNT = NPRINT

DO 12 I=1,KK
   AI(I) = 9999.
   N(I) = 99999999
12 CONTINUE

READ(03,*!)K,MSTRQ

DO 11 I=1,K
   READ(03,*!) AI(I),N(I)
11 CONTINUE

IF(K.NE.5) THEN
   WRITE(1,'("WARNING: BRIDGE STRUCTURE \"ONLY USES THE FIRST 5 COMPONENTS\")')
ELSE
END IF

***INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS, ***
* RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS  *
* AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN  *
***ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER  ***

DO 172 ALF=1,MAXALF
   DO 173 REPS=1,MAXREP
      QHTUPR(ALF,REPS) = 0.
      RSHAT(ALF,REPS) = 0.
      RHTSTR(ALF,REPS) = 0.
      RHSTBR(ALF,REPS) = 0.
      LOOPSO(REPS)=" ***** "
   173 CONTINUE
172 CONTINUE

***SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****
FLAG=1
DO 50 I=1,K -1
   IF((N(I) - N(I+1)).NE.0) THEN
      FLAG=0
   ELSE
   END IF
50 CONTINUE
PRINT *, 'FLAG IS:', FLAG

***MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)***

ZFPREP = 0.
ZFFAILS = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
   LOOP = LOOP + 1
   IF(TOTREP.LT.MAXRUN) THEN
      TOTREP = TOTREP + 1
   END IF
   SELCTA = 1
   SELCTB = 2

***FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT***
***OF THE SUBROUTINE SHSORT***

DO 95 REPS=1, MAXREP
   KEY1(REPS) = REPS
   KEY2(REPS) = REPS
   KEY3(REPS) = REPS
   KEY4(REPS) = REPS
95 CONTINUE

***CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE***

***CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.***

CALL IMAX(N,K,NMAX,NINDX)

***CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S***

DO 115 I=1, K
   QI(I) = MSTRQ * AI(I)
115 CONTINUE

DO 120 I=1,15
   DO 125 J=1,500
      UNIRV(I,J) = 999.
      TRIALS(I,J) = 99999
125 CONTINUE
120 CONTINUE

*** DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS***

DO 130 I=1, K

32
CALL SRND(SEED, TEMP, N(I), MULT, SORT)
DO 135 J=1, N(I)
   UNIRV(I,J) = TEMP(J).
   IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
      TRIALS(I,J) = 0
   ELSE
      TRIALS(I,J) = 1
   END IF
135 CONTINUE
130 CONTINUE

***CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT***
DO 150 I=1, K
   FI(I) = 0
150 CONTINUE
IONECT = 0

***CALCULATE THE F SUB I'S AND THE GRAND TOTAL NO. OF FAILURES***
BIGF = 0
DO 155 I=1, K
   DO 160 J=1, N(I)
      FI(I) = FI(I) + TRIALS(I,J)
   160 CONTINUE
   IF(FI(I).EQ.0) THEN
      ZFPREP = ZFPREP + 1
   ELSE
   END IF
155 CONTINUE

***CALCULATE THE QHAT SUB I'S: F SUB I'S DIVIDED BY N SUB I'S***
   QHATI(I) = REAL(FI(I)) / N(I)
   BIGF = BIGF + FI(I)

***COUNTS NUMBER OF COMPONENTS THAT HAVE FAILED***
DO 156 I=1,K
   IF (FI(I) .NE. 0) IONECT=-IONECT+1
156 CONTINUE
CALL CPARE(FI,K,BFLAG)

***CASE WHERE NO COMPONENTS HAVE ANY FAILURES***
   AVGN=0.0
   DO 200 I=1,K
      AVGN=AVGN+REAL(N(I))
   200 CONTINUE
   AVGN=AVGN/REAL(K)
   IF(BIGF.EQ.0) THEN
      LOOPSO(LOOP)=' *ZERO* '
      ZFAILS = ZFAILS + 1
      DO 205 ALF=1, MAXALF
         RSHAT(ALF,LOOP) = ALFA(ALF)**(i./AVGN)
      205 CONTINUE
IF(FLAG.EQ.1) THEN
  RHTSTR(ALF,LOOP) = ALFA(ALF)**(1./N(1))
ELSE
  END IF
IF (BFLAG.EQ.0) THEN
  RSHTBR(ALF,LOOP) = ALFA(ALF)**(1./AVGN)
ENDIF

205 CONTINUE
DEGFR(LOOP) = 2.
GO TO 10
ELSE
  FAILS = FAILS + 1
END IF

***COUNTS NUMBER OF COMPONENTS THAT FAIL RECORDS NO. COMPT TESTS***
:FCT=0
DO 202 I=1,K
  IF (FI(I).NE.0) THEN
    FCT=FCT+1
    NTEST' = N(I)
  ENDIF
202 CONTINUE

***FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES***
CALL RMAX(QHATI, K, QHTMAX, QINDX)

***CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)***
***IF COMPONENT HAS NO FAILURES AHAT SUB I IS ZERO***
SUMNAI = 0.
DO 165 I=1, K
  AHATI(I) = QHATI(I) / QHTMAX
  SUMNAI = SUMNAI + N(I) * AHATI(I)
165 CONTINUE

***1 COMPONENT FAILURE SERIES SYSTEM***
IF (FCT.EQ.1) THEN
  LOOPSO(LOOP) = 'ONECF'
  DO 305 ALF=1, MAXALF
    SUC=REAL(NTEST-BIGF)
    STUD=FIN(1.-ALFA(ALF), 2.*(REAL(BIGF)+1.), 2.*SUC)
    RSHAT(ALF,LOOP) = SUC/(SUC+(REAL(BIGF)+1.)*STUD)
    IF (FLAG.EQ.1) THEN
      RHTSTR(ALF,LOOP) = RSHAT(ALF,LOOP)
    ELSE
      ENDIF
 305 CONTINUE
ENDIF

***CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY***
DEGFR(LOOP) = 2 * (1 + BIGF)
DO 170 ALF=1, MAXALF
CALL MDCHI(1 - ALFA(ALF), DEGFR(LOOP), CHISQ(ALF, LOOP), ERROR)
QHTUPR(ALF, LOOP) = CHISQ(ALF, LOOP) / (2 * SUMNAI)
IF(FLAG.EQ.1) THEN
   RHTSTR(ALF, LOOP) = 1 -(CHISQ(ALF, LOOP) / REAL(2*N(1)))
ELSE
   END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNITS. IN SERIES***
IF (PCT .NE. 1) THEN
   CALL RHTSRS(QHTUPR(ALF, LOOP), AHATI, K, RSHAT(ALF, LOOP))
END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***

***IF NO SYSTEM FAILURE AND BRIDGE SYSTEM***
IF (BFLAG .EQ. 0) THEN
   RSHTBRL(ALF, LOOP) = ALFA(ALF)**(1./AVGN)
ENDIF

***IF MORE THAN 1 SYSTEM FAILURE AND BRIDGE SYSTEM***
IF (BFLAG .EQ. 2) THEN
   CALL RHTBRG(QHTUPR(ALF, LOOP), AHATI, K, RSHTBRL(ALF, LOOP))
ENDIF

***EXACTLY 1 SYSTEM FAILURE AND BRIDGE SYSTEM***
IF (BFLAG .EQ. 1) THEN
   SUC=REAL(AVGN-1)
   STUD=FINT(1.-ALFA(ALF),2.*2.,2.*SUC)
   RSHTBRL(ALF, LOOP) = SUC/(SUC+2.*STUD)
ENDIF

170 CONTINUE

***THIS ELSE AND ENDIF ARE FOR THE TEST AGAINST MAXRUN***
ELSE
   WRITE(1,'(''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',
   +'' ALLOWED OF: '',I6)') TOTREP
   GOTO 9999
END IF

GOTO 10
END IF

WRITE(2,'(''UNSORTED RSHAT 1 IS:'',/10(F8.5))')
+(RSHAT1,LOOP), LOOP=1, MAXREP)
WRITE(2,'(''UNSORTED RSHAT 2 IS:'',/10(F8.5))')
+(RSHAT2,LOOP), LOOP=1, MAXREP)
IF(FLAG.EQ.1) THEN
   WRITE(2,'(''UNSORTED RHTSTR 1 IS:'',/10(F8.5))')
+(RHTSTR1,LOOP), LOOP=1, MAXREP)
WRITE(2,'(''UNSORTED RHTSTR 2 IS:'',/10(F8.5))')
+(RHTSTR2,LOOP), LOOP=1, MAXREP)
ELSE

35
**END IF**

**IF(K.EQ.5) THEN**

**WRITE(2,'("UNSORTED RSHTBR 1 IS:'',/10(F8.5))')**

**+(RSHTBR(1,LOOP), LOOP=1, MAXREP)**

**WRITE(2,'("UNSORTED RSHTBR 2 IS:'',/10(F8.5))')**

**ELSE**

**WRITE (2,'("ZERO AND ONE FAILURE REPS:'',/10(A8))')**

**+ (LOOPSO(LOOP),LOOP=1,MAXREP):**

***SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)***

DO 700 ALF=1, MAXALF

DO 800 REPS=1, MAXREP

TRANSQ(REPS) = OHTUPR(ALF,REPS)

TRANSR(REPS) = RSHAT(ALF,REPS)

TRNSTR(REPS) = RHTSTR(ALF,REPS)

TRANBR(REPS) = RSHTBR(ALF,REPS)

CONTINUE

CALL SHSORT(TRANSQ,KEY1 ,MAXREP)

CALL SHSORT(TRANSR,KEY2 ,MAXREP)

CALL SHSORT(TRNSTR,KEY3 ,MAXREP)

CALL SHSORT(TRANBR,KEY4 ,MAXREP)

DO 900 REPS=1, MAXREP

QHTUPR(ALF,REPS) = TRANSQ(REPS)

RSHAT(ALF,REPS) = TRANSR(REPS)

RHTSTR(ALF,REPS) = TRNSTR(REPS)

RSHTBR(ALF,REPS) = TRANBR(REPS)

CONTINUE

700 CONTINUE

***PRINT OUTPUT REPORT HEADINGS***

WRITE(1,6666)

WRITE(1,6667) MAXREP

WRITE(1,6668) K

WRITE(1,6669)

**IF(K.EQ.5) THEN**

**WRITE(1,6699)**

ELSE

**END IF**

WRITE(1,6670) MSTRQ

WRITE(1,6671)

WRITE(1,6672) MSTRR

WRITE(1,6673) N

WRITE(1,6674)

***COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN. (SERIES SYSTEM)***

*** AND FOR THE 5-COMPONENT BRIDGE STRUCTURE***
CALL RSRS(QI,K,RS)
WRITE(1,'(''?://''THE TRUE SERIES SYSTEM '','
+''RELIABILITY VALUE IS:''",TS1,F8.5)') RS
CALL RBRDG(QI,K,RSBRDG).
IF(K.EQ.5) THEN
WRITE(1,'(''?://''THE TRUE BRIDGE STRUCTURE '',
+''RELIABILITY VALUE IS:''",TS1,F8.5)') RSBRDG
ELSE
END IF
WRITE(1,6675)

***COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO***
***RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)***

IF(FLAG.EQ.1) THEN
WRITE(1,5755).
ELSE
END IF
DO 450 ALF=1, MAXALF
  QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
  DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
  DELBRG(ALF) = RSBRDG - RSHTR(ALF,ALPHA,QUANTL(ALF))
  IF(FLAG.EQ.1) THEN
    DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
    WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
    WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
    WRITE(1,5657) DELSTR(ALF)
  ELSE
  END IF
  IF(K.EQ.5) THEN
    DELBRG(ALF) = RSBRDG - RSHTR(ALF,ALPHA,QUANTL(ALF))
    WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
    WRITE(1,5666) RSHTR(ALF,QUANTL(ALF))
    WRITE(1,5667) DELBRG(ALF)
  ELSE
  END IF
  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
  WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
  WRITE(1,5557) DELTAR(ALF)
450 CONTINUE
PRINT *, 'QUANTL(1) IS: ', QUANTL(1)
PRINT *, 'QUANTL(2) IS: ', QUANTL(2)

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
***

WRITE(1,6676)
DO 400 ALF=1,MAXALF
TRUQNT(ALF) = 0
  DO 500 REPS=1, MAXREP
    DIFF(REPS) = RS - RSHAT(ALF,REPS)
  500 CONTINUE
  IF(ABS(DIFF(REPS)).LE.EPS) THEN
    TRUQNT(ALF) = REPS
  ELSE
    CONTINUE
  END IF
DO 400 ALF=1,MAXALF
WRITE(1,6676)
WRITE(1,'('' ''/''TRUE CONFIDENCE LIMIT IS:'',
+ F8.4))
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 620
ELSEIF(DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 610
ELSE
END IF
600 CONTINUE
610 IF(TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1,'('' ''/''THE SMALLEST''
+ '' DIFFERENCE BETWEEN RS AND RSHAT IS:'',F10.5)') DIFF(
+ MAXREP)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1,'('' ''/''ALL RSHAT''
+ '' ARE GREATER THAN RS'')')
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
+ THEN
WRITE(1,4444) ALFA(ALF),
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
620 END IF
400 CONTINUE

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
**** ********** RSHTBR (BRIDGE) ********** ****

IF(K.EQ.5) THEN
DO 401 ALF=1,MAXALF
TRUQNT(ALF) = 0
DO 501 REPS=1, MAXREP
DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
501 CONTINUE
DO 601 REPS=1, MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'('' ''/''TRUE CONFIDENCE LIMIT IS:'',
+ F8.4)')
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 621
ELSEIF(DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 611
ELSE
END IF
601 CONTINUE
IF(TRYQNT(ALF) .EQ. 0.) THEN
   WRITE(1,4443) ALFA(ALF)
   WRITE(1,'("" /""THE SMALLEST'
   '+ F10.5)') DIFF(MAXREP)
ELSEIF(TRYQNT(ALF) .EQ. 1.) THEN
   WRITE(1,4442) ALFA(ALF)
   WRITE(1,'("" /""ALL RSHTBR'
   '+ ' ARE GREATER THAN RSBRDG")')
ELSEIF(ABS(DIFF(TRYQNT(ALF))).LE.ABS(DIFF(TRYQNT(ALF) - 1))) THEN
   WRITE(1,4444) ALFA(ALF),
   (TRYQNT(ALF) / REAL(MAXREP)) * 100.
   WRITE(1,4449) RSHTBR(ALF,TRYQNT(ALF))
   WRITE(1,4446)
ELSE
   WRITE(1,4444) ALFA(ALF),
   ((TRYQNT(ALF) - 1) / REAL(MAXREP)) * 100.
   WRITE(1,4449) RSHTBR(ALF,TRYQNT(ALF) - 1)
   WRITE(1,4447)
621 END IF
401 CONTINUE
ELSE CONTINUE
END IF

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

IF(FLAG .EQ. 1) THEN
   DO 4400 ALF=1,MAXALF
      TRYQNT(ALF) = 0
   DO 5500 REPS=1,MAXREP
      DIFF(REPS) = RS - RHITSTR(ALF,REPS)
      CONTINUE
   DO 6600 REPS=1,MAXREP
      IF(ABS(DIFF(REPS)).LE.EPS) THEN
         TRYQNT(ALF) = REPS
         WRITE(1,'("" /""TRUE CONFIDENCE LIMIT IS:"
         '+ F8.4)')
         + (TRYQNT(ALF) / REAL(MAXREP)) * 100.
         GO TO 6620
      ELSEIF(DIFF(REPS).LT.0.) THEN
         TRYQNT(ALF) = REPS
         GO TO 6610
      ELSE
      END IF
   CONTINUE
6600 CONTINUE
6610 IF(TRYQNT(ALF) .EQ. 0.) THEN
   WRITE(1,4443) ALFA(ALF)
   WRITE(1,'("" /""THE SMALLEST'
   '+ F9.5)') DIFF(MAXREP)
ELSEIF(TRYQNT(ALF) .EQ. 1.) THEN
   WRITE(1,4442) ALFA(ALF)
   WRITE(1,'("" /""ALL RHITSTR'

39
+ ' ' ARE GREATER THAN RS'')')
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE. ABS(DIFF(TRUQNT(ALF) - 1)))
+ THEN
  WRITE(1,4444) ALFA(ALF),
  (TRUQNT(ALF) / REAL(MAXREP)) * 100.
  WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
  WRITE(1,4446)
ELSE
  WRITE(1,4444) ALFA(ALF),
  ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
  WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
  WRITE(1,4447)
END IF
4400 CONTINUE
ELSE
END IF

***PRINT THE ARRAYS PERTINENT TO THE OUTPUT OF EACH REPLOICATION***

IF(PRNT.EQ.1) THEN
  I = 1
185 WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
  ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
  ALFA(SELCTB)
175 IF(I.GE.(MAXREP + 1)) THEN
  GOTO 180
ELSE
    I = I + 70
    WRITE(1,'(''++'')')
  GOTO 185
ELSE
  WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
  CHISQ(2,I), QHTUPR(2,I)
END IF
IF(I + 70.LE.MAXREP) THEN
  WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),
  QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
ELSE
END IF
I = I + 1
GOTO 175
180 END IF
ELSE
ENDIF
9999 WRITE(1,'(''THE TOTAL NO OF REPS WAS:'',I8)') TOTREP
  WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS:'',I8)') LOOP
  WRITE(1,'(''AVERAGE NO. OF COMPONENTS PER REPLICATION WITH '',
  +''NO FAILURES:'',F5.2)') ZFPREP / MAXREP
  WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS:'',I8)') FAILS
0008 FORMAT (/ 3X,'C 1',5X,'C 2',
  +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
  +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X)
6676 FORMAT ('+', '***********', 'TRUE CONFIDENCE LIMITS ***********', '***********')
END
APPENDIX D. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES PARALLEL SYSTEM)

PROGRAM ZFYSCN

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

* TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE
* ZERO FAILURES ALLOWED; NO SCALING
* AUTHOR: E. F. BELLINI, LT, USN
* MODIFIED BY: LT VALERIE A. COVINGTON, USN (MAR 90)
* DATE: NOV 89
* THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY
* OF THEIR COMPONENTS
* IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.
* 'B1 EXEC PROGRAM
* 'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT
* FILES RESULTING FROM THE 12 CONSECUTIVE RUNS. BY EDITING THE
* INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-
* SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.

VARIABLES USED

AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT
AI : INPUT WEIGHTS FOR EACH COMPONENT
ALFA : LEVELS OF SIGNIFICANCE
BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION
CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE
CIC15 : FORMAT LABEL
DEGFR : DEGREES OF FREEDOM
DELBRG : DIFFERENCE FOR BRIDGE SYSTEM
DELSSTR : DIFFERENCE FOR SERIES SYSTEM - CLOSED FORM
DELTAR : DIFFERENCE FOR SERIES SYSTEM
DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.)
EPS : SMALL QUANTITY(CONSTANT)
ERROR : PARAMETER FOR IMSL ROUTINE
FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE
FI : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)
FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS
INC : INCREMENT STEP SIZE FOR ROUTINE USMNMX
KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT
KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT
KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT
KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT
KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS
LOOP : COUNTS NO. OF REPLICATION PERFORMED
MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED ARRAY SIZING

*
* MAXREP : MAX NO. OF REPLICATIONS
* MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED
* NSTRQ : MASTER UNRELIABILITY: USED WITH AI'S TO CALC. QI'S)
* MULT : MULTIPLIER FOR RANDOM NO. GENERATOR SRND
* N : NO. OF MISSION TEST FOR EACH COMPONENT
* NIMAX : MAX NO. OF MISSION TESTS
* NIMIN : MIN NO. OF MISSION TESTS
* NINDX : INDEX NO. OF MAX NO. OF MISSION TESTS
* NREAL : NO. OF MISSION TESTS TRANSFORMED TO REAL
* NMAX : MAX NO. OF MISSION TESTS
* NPRNT : FLAG FOR DETAILED REPORT OUTPUT
* PRNT : SAVE AS ABOVE PARAMETER
* QHATI : UNRELIABILITY ESTIMATES FOR EACH COMPONENT
* QHIMAX : LARGEST QHATI
* QHUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY
* QI : INPUT UNRELIABILITY FOR EACH COMPONENT
* QINDEX : INDEX
* QUANTL : QUANTILE
* REPSHD : REPLICATIONS HEADING FORMAT NUMBER
* RS : TRUE SERIES SYSTEM RELIABILITY
* RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY
* RSHAT : SERIES SYSTEM RELIABILITY ESTIMATE
* RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE
* SEED : PARAMETER
* SELECTA : SIGNIFICANCE LEVEL SELECTION
* SELECTB : SIGNIFICANCE LEVEL SELECTION
* SERIES : PARAMETER FOR ROUTINE SRND
* SUMNIAI : SUM OF THE PRODUCT OF NI'S AND AI'S
* TEMP : TEMPORARY ARRAY
* TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS
* TRANBR : TEMPORARY ARRAY
* TRANSQ : TEMPORARY ARRAY
* TRANSR : TEMPORARY ARRAY
* TRIALS : BERNOULLI TRIALS ARRAY (2-DIM)
* TRNSTR : TEMPORARY ARRAY
* TRUQNT : TRUE QUANTILE
* UNIRV : UNIFORM RANDOM DEVIATES (2-DIM)
* ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES
* ZFPREP : NO. OF COMPTNS. WITH ZERO FAILURES PER REPLICATION

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

PARAMETER (KK=10, MAXALF=2, NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=.000001)
REAL*4 UNIRV(15,1000), TEMP(1000), QI(KK), AI(KK), AHATI(KK)
REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5), AALFA(5), SUMNIAI, RSHAiTCMAXALF, MAXREP) , RS
REAL*4 KEY1(MAXREP), KEY2(MAXREP), KEY3(MAXREP), TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF, MAXREP), CHISQ(MAXALF, MAXREP)
REAL*4 QUPA1(MAXREP), QUPA2(MAXREP), RHTSTR(MAXALF, MAXREP)
REAL*4 DELTA(MAXALF), TRANSQ(MAXREP), TRANSR(MAXREP), DIFF(MAXREP)
REAL*4 DELSTR(MAXALF), NIMIN, NIMAX, NIREAL(KK)
REAL*4 RSHTBR(MAXALF, MAXREP), DELBRG(MAXALF), KEY4(MAXREP)
REAL*4 TRANBR(MAXREP), RSBRDG , MSTRQ
REAL*4 ZFPREP
REAL*4 AVGN,SUC,STUD,PTMP(10),FDEG1,FDEG2,P3,S3,S4

INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS, INC
INTEGER NTEST, FCT, HFI

INTEGER C1C15, REPISHD, SELCTA, SELCTK, TOTREP
INTEGER SORT, TRI-ALS(-15,1000), BIGF, FI(K), N(KK)
INTEGER NINDX, QINDX, ERROR, REPS, -SELCTA, SELCTK, TOTREP
INTEGER C1C15, REPISHD, SELCTA, SELCTK, TOTREP
INTEGER NINDX, QINDX, ERROR, REPS, -SELCTA, SELCTK, TOTREP

DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
DATA ALFA/.20,.050-/
DATA SORT/00/,
ASSIGN 8 TO C1C15
ASSIGN 9 TO REPISHD

* CALL COMPR
PRNT = NPRNT

DO 12 I=1,KK
  AI(I) = 9999.
  N(I) = 9999999
12 CONTINUE

READ(03,*) K,HSTRQ

DO 11 I=1,K
  READ(03,*) AI(I), N(I)
11 CONTINUE

IF(K.NE.5) THEN
  WRITE(1,'("WARNING: BRIDGE STRUCTURE ",'+'
"ONLY USES THE FIRST 5 COMPONENTS")')
ELSE
  END IF

***SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****
FLAG=1
DO 50 I=1,K-1
   IF((N(I) - N(I+1)).NE.0) THEN
      FLAG=0
   ELSE
   END IF
50 CONTINUE
PRINT *, 'FLAG IS:', FLAG

*** MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)***

ZFREP = 0.
ZFails = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
   LOOP = LOOP + 1
   IF(TOTREP.LT.MAXRUN) THEN
      TOTREP = TOTREP + 1
   END IF
   SELCTA = 1
   SELCTB = 2
   CALL SHSORT(REPS, MAXREP, KEY1(REPS), KEY2(REPS), KEY3(REPS), KEY4(REPS))
115 CONTINUE

*** CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE ***

*** CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO. ***

CALL IMAX(N,K,NMAX,NINDX)

*** CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S ***

DO 120 I=1, K
   QI(I) = MSTRQ * AI(I)
120 CONTINUE

C
S3=QI(3)
S4=QI(1)**.50
DO 125 J=1,15
   DO 125 J=1,15
      UNIRV(I,J) = 999.
      TRIALS(I,J) = 99999
125 CONTINUE

*** DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS ***

DO 130 I=1, K
130 CONTINUE
CALL SRND(SEED, TEMP, N(I), MULT, SORT)
DO 135 J=1, N(I)
   UNIRV(I,J) = TEMP(J)
   IF (UNIRV(I,J) .LE. 1 - QI(I)) THEN
      TRIALS(I,J) = 0
   ELSE
      TRIALS(I,J) = 1
   END IF
135 CONTINUE
130 CONTINUE

***CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT***

DO 150 I=1, K
   FI(I) = 0
150 CONTINUE
IONECT = 0

***CALCULATE THE F-SUB I'S AND THE GRAND TOTAL NO. OF FAILURES***

BIGF = 0
DO 155 I=1, K
   DO 160 J=1, N(I)
      FI(I) = FI(I) + TRIALS(I,J)
   CONTINUE
   IF (FI(I) .EQ. 0) THEN
      ZFPREP = ZFPREP + 1
   ELSE
      BIGF = BIGF + FI(I)
   END IF
   QHATI(I) = REAL(FI(I)) / N(I)
155 CONTINUE

***CHANGES FOR SERIES PARALLEL SYSTEM (COMPONENT 2)***

IF (FI(2) .EQ. 0) THEN
   ZFPREP = ZFPREP - 1
ENDIF
BIGF = BIGF - FI(2)
FI(1) = 0
DO 161 I=1, N(2)
   HFI = 0
   DO 162 J=1, 2
      CALL SRND(SEED, PTEMP(J), N(2), MULT, SORT)
      IF (PTEMP(J) .GT. 1-S4) THEN
         HFI = HFI + 1
      ENDIF
162 CONTINUE
   IF (HFI .EQ. 2) THEN
      FI(2) = FI(2) + 1
   ENDIF
161 CONTINUE
BIGF = BIGF + FI(2)
IF (FI(2) .EQ. 0) THEN
   ZFPREP = ZFPREP + 1
ENDIF
ELSE
ENDIF
C IF (FI(3) .EQ. 0) THEN
C ZFPREP = ZFPREP - 1
C ENDIF
C BIGF = BIGF + FI(3)
C FI(3) = 0
DO 163 I=1,N(3)
HFI = 0
DO 164 J=1,3
CALL SRND(SEED, PTEMP(J), N(3), MULT, SORT)
C IF (PTEMP(J) .GT. 1-S3) THEN
C HFI = HFI + 1
C ENDIF
C164 CONTINUE
C IF (HFI .GE. 2) THEN
C FI(3) = FI(3) + 1
C ENDIF
C163 CONTINUE
C BIGF = BIGF + FI(3)
C IF (FI(3) .EQ. 0) THEN
C ZFPREP = ZFPREP + 1
C ELSE
C ENDIF
DO 19 I=1,K
QI(I) = MSTRQ * AI(I)
QMATI(I) = REAL(FI(I)) / N(I)
19 CONTINUE
C QI(4) = QI(4)**2
C P3 = 1. - QI(3)
C QI(3) = 1. - ((3.*P3**2*QI(3))+(P3**3))
***COUNTS NUMBER OF COMPONENTS THAT HAVE FAILED***

DO 156 I=1,K
  IF (FI(I) .NE. 0) IONECT = IONECT + 1
156 CONTINUE

***CASE WHERE NO COMPONENTS HAVE ANY FAILURES***

IF (BIGF .EQ. 0) THEN
  LOOPSO(LOOP) = ' ' *ZERO*
  ZFAILS = ZFAILS + 1
  AVGN = 0.0
  DO 200 I=1,K
    AVGN = AVGN + REAL(N(I))
  AVGN = AVGN / REAL(K)
  CONTINUE
  AVGN = AVGN / REAL(K)
  DO 205 ALF = 1, MAXALF
    RSHAT(ALF, LOOP) = ALFA(ALF)**(1./AVGN)
    IF (FLAG.EQ.1) THEN
      RHTSTR(ALF, LOOP) = ALFA(ALF)**(1./N(I))
    ELSE
      END IF
  PRINT *, LOOP', LOOP', 'RSHAT', RSHAT(ALF, LOOP)
205 CONTINUE
DEGFR(LOOP) = 2.
GO TO 10
ELSE
FAILS = FAILS + 1
ENDIF

***COUNTS NUMBER OF COMPONENTS THAT FAIL RECORDS NO. COMPT TESTS***
FCT=0
DO 202 I=I,K
 IF (FI(I) .NE. 0) THEN
 FCT=FCT+1
 NTEST=N(I)
ENDIF
202 CONTINUE

***FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES***
CALL RMAX(QHATI, K, QHTMAX, QINDX)
IF (LOOP .EQ. 1) THEN
ENDIF

***CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)***
***IF COMPONENT HAS NO FAILURES AHAT SUB I IS ZERO***
SUMNAI = 0.
DO 165 I=I, K
  AHATI(I) = QHATI(I) / QHTMAX
  SUMNAI = SUMNAI + N(I) * AHATI(I)
165 CONTINUE

***1 COMPONENT FAILURE SERIES SYSTEM***
IF (FCT .EQ. 1) THEN
 LOOPSO(LOOP)=' *ONECF* '
 DO 305 ALF=1,MAXALF
   SUC=REAL(NTEST-BIGF)
   FDEG1=2.*REAL(BIGF)+1.
   FDEG2=2.*SUC
   STUD=FIN(1.-ALFA(ALF),FDEG1,FDEG2)
   RSHAT(ALF,LOOP)=SUC/(SUC+(REAL(BIGF)+1.)*STUD)
   PRINT *, 'SUC=',SUC,'FAIL=',BIGF,'NTEST=',NTEST
   PRINT *, 'ALFA=',ALFA(ALF),'RSHAT=',RSHAT(ALF,LOOP)
 IF (FLAG .EQ. 1) THEN
   RHTSTR(ALF, LOOP)=RSHAT(ALF, LOOP)
 ELSE
 ENDIF
305 CONTINUE
ENDIF

***CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY***
DEGFR(LOOP) = 2 * (1 + BIGF)
DO 170 ALF=1, MAXALF
 CALL MDCHI(1 - ALFA(ALF), DEGFR(LOOP), CHISQ(ALF, LOOP), ERROR)
49
**QHTUPR(ALF,LOOP)** = CHISQ(ALF,LOOP) / (2 * SUMNAI)

IF(FLAG.EQ.1) THEN
   RHTSTR(ALF,LOOP) = 1 - (CHISQ(ALF,LOOP) / REAL(2*N(1)))
ELSE
END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPONENTS IN SERIES***

IF (FCT .NE. 1) THEN
   CALL RHTSR(QHTUPR(ALF,LOOP), AHATI, K, RSHAT(ALF,LOOP))
END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***

IF (IONECT .NE. 1) THEN
   CALL RHTBRG(QHTUPR(ALF,LOOP), AHATI, K, RSHTBR(ALF,LOOP))
END IF

170 CONTINUE

***EXACTLY 1 COMPONENT FAILS AND REDUNDANT COMPONENT***

IF (((IONECT .EQ. 1) .AND. (K .EQ. 5)) THEN
   DO 207 I=1, K
      NIREAL(I) = REAL(N(I))
   CONTINUE
   CALL USMNMIX(NIREAL, K, INC, NMIN, NMAX)
   DO 206 ALF=1, MAXALF
      RSHTBR(ALF, LOOP) = ALFA(ALF)**(1./NMIN)
   CONTINUE
END IF

***THIS ELSE AND ENDIF ARE FOR THE TEST AGAINST MAXRUN***

ELSE
   WRITE(1,'(''PROGRAM EXCEEDED THE MAX. NO. OF RUNS'',
       +'' ALLOWED OF: '',16)') TOTREP
   GOTO 9999
END IF
GOTO 10
END IF

WRITE(2,'(''UNSORTED RSHAT 1 IS: '',/10(F8.5))')
+(RSHAT(1,LOOP), LOOP=1, MAXREP)
WRITE(2,'(''UNSORTED RSHAT 2 IS: '',/10(F8.5))')
+(RSHAT(2,LOOP), LOOP=1, MAXREP)
IF(FLAG.EQ.1) THEN
   WRITE(2,'(''UNSORTED RHTSTR 1 IS: '',/10(F8.5))')
+(RHTSTR(1,LOOP), LOOP=1, MAXREP)
   WRITE(2,'(''UNSORTED RHTSTR 2 IS: '',/10(F8.5))')
+(RHTSTR(2,LOOP), LOOP=1, MAXREP)
ELSE
END IF
IF(K.EQ.5) THEN
   WRITE(2,'(''UNSORTED RSHTBR 1 IS: '',/10(F8.5))')
+(RSHTBR(1,LOOP), LOOP=1, MAXREP)
C 50
**WRITE** (2, '(''UNSORTED RSHTBR 2 IS: '',/10(F8.5))')
**WRITE** (2, ' + (RSHTBR(2,LOOP), LOOP=1, MAXREP)
**ELSE**
**WRITE** (2, ' + (LOPSO(LOOP),LOOP=1,MAXREP)

***SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)***

DO 700 ALF=1, MAXALF
  DO 800 REPS=1, MAXREP
    **TRANSQ**(REPS) = QHTUPR(ALF,REPS)
    **TRANSR**(REPS) = RSHAT(ALF,REPS)
    **TRNSTR**(REPS) = RHTSTR(ALF,REPS)
    **TRANBR**(REPS) = RSHTBR(ALF,REPS)
  800  CONTINUE
  CALL S HSORT(TRANSQ,KEY1,MAXREP)
  CALL S HSORT(TRANSR,KEY2,MAXREP)
  CALL S HSORT(TRNSTR,KEY3,MAXREP)
  CALL S HSORT(TRANBR,KEY4,MAXREP)
  DO 900 REPS=1, MAXREP
    **QHTUPR**(ALF,REPS) = **TRANSQ**(REPS)
    **RSHAT**(ALF,REPS) = **TRANSR**(REPS)
    **RHTSTR**(ALF,REPS) = **TRNSTR**(REPS)
    **RSHTBR**(ALF,REPS) = **TRANBR**(REPS)
  900  CONTINUE
  700  CONTINUE

***PRINT OUTPUT REPORT HEADINGS***

**WRITE** (1,6666)
**WRITE** (1,6667) MAXREP
**WRITE** (1,6668) K
**WRITE** (1,6669) IF(K.EQ.5) THEN
  **WRITE** (1,6699) ELSE
  **END IF**
**WRITE** (1,6670) MSTRQ
**WRITE** (1,6671)
**WRITE** (1,C1C15)
**WRITE** (1,3334) AI
**WRITE** (1,0007)
**WRITE** (1,C1C15)
**WRITE** (1,3334) QI
**WRITE** (1,0005)
**WRITE** (1,C1C15)
**WRITE** (1,3335) N
**WRITE** (1,6674)

***COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN.(SERIES SYSTEM)***

*** AND FOR THE 5-COMPONENT BRIDGE STRUCTURE***

**CALL** RSRS(QI,K,RS)
WRITE(1,'('' '' ///''THE TRUE SERIES SYSTEM ''
'':''RELIABILITY VALUE IS:','T51,F8.5')') RS
CALL RBRIDG(QI,K,RSBRDG)
IF(K.EQ.5) THEN
WRITE(1,'('' '' ///''THE TRUE BRIDGE STRUCTURE ''
'':''RELIABILITY VALUE IS:','T51,F8.5')') RSBRDG
ELSE
END IF
WRITE(1,6675)

***COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEORETICAL QUANTILE GIVEN BY ALFA (MUST USE SORTED RSHAT ARRAY)***

IF(FLAG.EQ.1) THEN
WRITE(1,5755)
ELSE
END IF
DO 450 ALF=1, MAXALF
QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
IF(FLAG.EQ.1) THEN
DESTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
WRITE(1,5657) DESTR(ALF)
ELSE
END IF
IF(K.EQ.5) THEN
DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
WRITE(1,5667) DELBrg(ALF)
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)
450 CONTINUE
PRINT *, 'QUANTL(1) IS: ', QUANTL(1)
PRINT *, 'QUANTL(2) IS: ', QUANTL(2)

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

WRITE(1,6676)
DO 400 ALF=1, MAXALF
TRUQNT(ALF) = 0
DO 500 REPS=1, MAXREP
DIFF(REPS) = RS - RSHAT(ALF,REPS)
500 CONTINUE
DO 600 REPS=1, MAXREP
IF(ABS(DIFF(REPS)) .LE. EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'('' '' ///''TRUE CONFIDENCE LIMIT IS: ''
'':''F8.4)')
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 620
ELSEIF (DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 610
ELSE
END IF
600 CONTINUE
610 IF (TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1,'('''' ,/''THE SMALLEST'',
''DIFFERENCE BETWEEN RS AND RSHAT IS:'',F10.5)') DIFF(MAXREP)
ELSEIF (TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1,'('' '' ,/''ALL RSHAT'',
''ARE GREATER THAN RS'')')
ELSEIF (ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
THEN
WRITE(1,4444) ALFA(ALF),
(TRUQNT(ALF) / REAL(MAXREP)) * 100.
WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
620 END IF
400 CONTINUE

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
*** ********** RSHTBR (BRIDGE) ********** ***

IF(K.EQ.5) THEN
DO 401 ALF=1,MAXALF
TRUQNT(ALF) = 0
DO 501 REPS=1, MAXREP
DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
501 CONTINUE
DO 601 REPS=1, MAXREP
IF (ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'('' '' ,/''TRUE CONFIDENCE LIMIT IS:'',
F8.4)')
+(TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 621
ELSEIF (DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 611
ELSE
END IF
601 CONTINUE
611 IF (TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1, ('''''',''''THE SMALLEST''',''''DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'''', +''''F10.5''')) DIFF(MAXREP)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1, ('''''','' ''ALL RSHTBR'''', +''''ARE GREATER THAN RSBRDG'''))
ELSEIF(ABS(DIFF(TRUQNT(ALF)))LE.ABS(DIFF(TRUQNT(ALF) - 1)))
+ THEN
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF) - 1) / REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF) - 1)
WRITE(1,4447)
ELSE
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF) - 1) / REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF) - 1)
621   END IF
401  CONTINUE
ELSE
END IF

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
***  **********  RHTSTR **********  ***
IF(FLAG.EQ.1) THEN
DO 4400 ALF=1,MALF
TRUQNT(ALF) = 0
DO 5500 REPS=1, MAXREP
DIFF(REPS) = RS - RHTSTR(ALF,REPS)
5500 CONTINUE
DO 6600 REPS=1, MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,('''''','' ''TRUE CONFIDENCE LIMIT IS:'''', +''''F8.4'''))
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 6620
ELSEIF(DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 6610
ELSE
END IF
6600 CONTINUE
6610 IF(TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1, ('''''','' ''THE SMALLEST''','' ''DIFFERENCE BETWEEN RS AND RHTSTR IS:'''', +''''F9.5''')) DIFF(MAXREP)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1, ('''''','' ''ALL RHTSTR'''', +''''ARE GREATER THAN RS'''))
ELSEIF(ABS(DIFF(TRUQNT(ALF)))LE.ABS(DIFF(TRUQNT(ALF) - 1)))
...
'QHTUPR(',F4.3,')' IX
'CHISQR(?F4.3
l;X,'QHTUPR(',F4.3,')',lX,
+2X,'REP NO',2X,DF',lX,'CHISQR(',F4.3,')',lX,
+QHTUPR(',F4.3,')',lX,'CHISQR(',F4.3,')',lX,'QHTUPR(',F4.3,')'/
0001 FORMAT ('///'UNIFORM RANDOM DEVIATES ARE:')
0002 FORMAT ('///'BERNOULLI TRIALS ARE:')
0003 FORMAT ('///'TOTAL NO. OF FAILURES FOR EACH COMPONENT:')
0004 FORMAT ('///'ESTIMATED UNRELIABILITY FOR EACH COMPONENT:')
0005 FORMAT ('///'TOTAL NUMBER OF MISSION TESTS:')
0006 FORMAT ('///'ESTIMATED WEIGHTS FOR EACH COMPONENT:')
0007 FORMAT ('///'Q I FOR EACH COMPONENT:')
1111 FORMAT (15F8.5)
2222 FORMAT (/1X,15(I4,4X))
3333 FORMAT (/1X,15(I4,4X))
3334 FORMAT (15F8.5)
3335 FORMAT (/1X,15(I4,4X))
3336 FORMAT (T3,14,T9,13,T13,F11.5,T27,F8.5,T39,F11.5,T53,F8.5)
3337 FORMAT ('+',T67,14,T73,14,T77,F11.5,T91,F8.5,T103,F11.5,T117,F8.5)
4442 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
+LIMIT IS:',T50,' 00.000 ')
4443 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
+LIMIT IS:',T50,'100.0000')
4444 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
+LIMIT IS:',T50,F8.4)
4445 FORMAT (' ',///'THE RSHTAT VALUE CLOSEST TO RS IS: ',T51,F8.5)
4446 FORMAT (' ',///'THE FIRST NEGATIVE DIFFERENCE')
4447 FORMAT (' ',///'ELEMENT PRECEEDING FIRST NEGATIVE DIFFERENCE')
4448 FORMAT (' ',///'THE RSHTAT VALUE CLOSEST TO RS IS: ',T51,F8.5)
4449 FORMAT (' ',///'THE RSHTAT VALUE CLOSEST TO RSBRDG IS: ',T51,F8.5)
5555 FORMAT (' ',///'THE ',14,'(1- ',F4.3,') QUANTILE IS: ',T49,F8.3)
5556 FORMAT (' ',///'THE VALUE OF RSHTAT FOR THAT QUANTILE IS: ',T51,F8.5)
5557 FORMAT (' ',///'THE DIFFERENCE(RS - RSHTAT) IS: ',T51,F8.5)
5656 FORMAT (' ',///'THE VALUE OF RHSTR FOR THAT QUANTILE IS: ',T51,F8.5)
5666 FORMAT (' ',///'THE DIFFERENCE(RS - RHSTR) IS: ',T51,F8.5)
5667 FORMAT (' ',///'THE DIFFERENCE(RS - RSHTBR) IS: ',T51,F8.5)
5755 FORMAT (' ',///'SINCE THE NO. OF MISSION TESTS IS THE SAME FOR',
+ALL COMPONENTS THE CLOSED FORM SERIES SYSTEM RELIABILITY ',
+RHTSTR' IS COMPUTED')
6666 FORMAT ('+',//'RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********',
+RUN INPUT SETTINGS **********'.
APPENDIX E. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (SERIES-PARALLEL SYSTEM WITH A 2/3 COMPONENT)

PROGRAM ZFYSCN
******************************************************************************
* * TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE * *
* ZERO FAILURES ALLOWED; NO SCALING * *
* AUTHOR: E. F. BELLINI, LT, USN * *
* MODIFIED BY: LT VALERIE A. COVINGTON, USN (MAR 90) * *
* DATE: NOV 89 * *
* * THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE * *
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY * *
* OF THEIR COMPONENTS * *
* * IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12 * *
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT * *
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'. *
* THE REXX EXEC PROGRAM * *
* 'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT * *
* FILES RESULTING FROM THE 12 CONSECUTIVE RUNS. BY EDITING THE * *
* INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-* *
* SPECIFIC RUN FROM JUST ONE RUN TO ALL 12. * *
* * VARIABLES USED * *
* * AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT * *
* AI : INPUT WEIGHTS FOR EACH COMPONENT * *
* ALFA : LEVELS OF SIGNIFICANCE * *
* BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION * *
* CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE * *
* C1C15 : FORMAT LABEL * *
* DEGFR : DEGREES OF FREEDOM * *
* DELBRG : DIFFERENCE FOR BRIDGE SYSTEM * *
* DELSTR : DIFFERENCE FOR SERIES SYSTEM - CLOSED FORM * *
* DELTAR : DIFFERENCE FOR SERIES SYSTEM * *
* DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.) * *
* EPS : SMALL QUANTITY (CONSTANT) * *
* ERROR : PARAMETER FOR IMSL ROUTINE * *
* FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE * *
* FI : NO. OF FAILURES FOR EACH COMPONENT (ALL MISSION TEST) * *
* FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS * *
* INC : INCREMENT STEP SIZE FOR ROUTINE USMNMX * *
* KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT * *
* KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT * *
* KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT * *
* KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT * *
* KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS * *
* LOOP : COUNTS NO. OF REPPLICATION PERFORMED * *
* MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED (ARRAY SIZING) *
******************************************************************************

58
* MAXREP : MAX NO. OF REPLICATIONS
* MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED
* MSTRQ : MASTER UNRELIABILITY (USED WITH AI'S TO CALC. QI'S)
* MULT : MULTIPLIER FOR RANDOM NO. GENERATOR SRND
* N : NO. OF MISSION TEST FOR EACH COMPONENT
* NIMAX : MAX NO. OF MISSION TESTS
* MINIM : MIN NO. OF MISSION TESTS
* NINDX : INDEX NO. OF MAX NO. OF MISSION TESTS
* NIREAL : NO. OF MISSION TESTS TRANSFORMED TO REAL
* NMAX : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL
* NPRNT : FLAG FOR DETAILED REPORT OUTPUT
* PRNT : SAME AS ABOVE (PARAMETER)
* QHATI : UNRELIABILITY ESTIMATES FOR EACH COMPONENT
* QHTMAX : LARGEST QHATI
* QHTUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY
* QI : INPUT UNRELIABILITY FOR EACH COMPONENT
* R : DEPENDS (same as above parameter)
* RS : TRUE SERIES SYSTEM RELIABILITY
* RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY
* RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE
* RSHTSTR : SERIES SYSTEM RELIABILITY ESTIMATE (CLOSED FORM)
* SEED : PARAMETER
* SELECTA : SIGNIFICANCE LEVEL SELECTION
* SELECTB : SIGNIFICANCE LEVEL SELECTION
* SORT : PARAMETER FOR ROUTINE SRND
* SUMNAI : SUM OF THE PRODUCT OF N'I'S AND AI'S
* TEMP : TEMPORARY ARRAY
* TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS
* TRANBR : TEMPORARY ARRAY
* TRANSQ : TEMPORARY ARRAY
* TRANSR : TEMPORARY ARRAY
* TRIALS : BERNOLLI TRIALS ARRAY (2-DIM)
* TRNSTR : TEMPORARY ARRAY
* TRUQNT : TRUE QUANTILE
* UNIRV : UNIFORM RANDOM DEVIATES (2-DIM)
* ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES
* ZFPREP : NO. OF COMPONENTS WITH ZERO FAILURES PER REPLICATION

PARAMETER (KK=10, MAXALF=2, NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=.000001)
REAL*4 UNIRV(15,1000), TEMP(1000), QI(KK), AI(KK), QHATI(KK)
REAL*4 QHATI(KK), NMAX, NMIN, NIREAL, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5), AALFA(5), SUNNAI, RSHAT(MAXALF, MAXREP), RS
REAL*4 KEY1(MAXREP), KEY2(MAXREP), KEY3(MAXREP), TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXREP, MAXREP), CHISQR(MAXALF,MAXREP)
REAL*4 QUPA1(MAXREP), QUPA2(MAXREP), RSHSTR(MAXALF, MAXREP)
REAL*4 DELTAR(MAXALF), TRANSQ(MAXREP), TRANSR(MAXREP), DIFF(MAXREP)
REAL*4 DELSTR(MAXALF), NMIN, NMAX, NIREAL(KK)
REAL*4 RSHTBR(MAXALF, MAXREP), DELBRG(MAXALF), KEY4(MAXREP)
REAL*4 TRANSR(MAXREP), RSBRDG, MSTRQ
REAL*4 2FPREP

59
REAL*4 AVGN, SUC, STUD, PTEMP(10), FDEG1, FDEG2, P3, S3, S4
INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTK, TOTREP
INTEGER CI15, REPShD, SELCTB, ALF, FLAG, LOOP, PRNT
INTEGER QUANTL(MAXALF), TRUQTN(MAXALF), ZFAILS, FAILS, INC
INTEGER NTEST, FCT, HFI
CHARACTER*8 LOOPSO(MAXREP)
DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
DATA ALFA/.20,.050/
DATA SORT/0/
ASSIGN 8 TO CIC15
ASSIGN 9 TO REPShD
* CALL COMPR5
PRNT = NPRNT
DO 12 I=1,KK
   AI(I) = 9999.
   N(I) = 99999999
12 CONTINUE
READ(03,*)K,MSTRQ
DO 11 I=1,K
   READ(03,*) AI(I),N(I)
11 CONTINUE

IF(K.NE.5) THEN
   WRITE(1,'(13 WARNIN(: BRIDGE STRUCTURE ',
   +'ONLY USES THE FIRST 5 COMPONENTS')')
ELSE
   END IF
***INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS, 
* RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS 
* AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN 
***ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER 
***
DO 172 ALF=1,MAXALF
   DO 173 REPS=1,MAXREP
      QHTUPR(ALF,REPS) = 0.
      RSHAT(ALF,REPS) = 0.
      RHTSTR(ALF,REPS) = 0.
      RSHTBR(ALF,REPS) = 0.
      LOOPS0(REPS)=' '****
173 CONTINUE
172 CONTINUE

***SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****

60
FLAG=1
DO 50 I=1,K-1
   IF((N(I) - N(I+1)).NE.0) THEN
      FLAG=0
   ELSE
      END IF
50 CONTINUE
PRINT *, 'FLAG IS: ', FLAG

***MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)***

ZFPREP = 0.
ZFAILS = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
   LOOP = LOOP + 1
   IF(TOTREP.LT.MAXRUN) THEN
      TOTREP = TOTREP + 1
   END IF
END IF
SELCTA = 1
SELCTB = 2

***FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT***
***OF THE SUBROUTINE SHSORT***

DO 95 REPS=1, MAXREP
   KEY1(REPS) = REPS
   KEY2(REPS) = REPS
   KEY3(REPS) = REPS
   KEY4(REPS) = REPS
95 CONTINUE

***CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE***

***CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.***

CALL IMAX(N,K,NMAX,NINDX)

***CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S***

DO 115 I=1, K
   QI(I) = MSTRQ * AI(I)
115 CONTINUE
S3=QI(3)
S4=QI(4)

DO 120 I=1,15
   DO 125 J=1,500
      UNIRV(I,J) = 999.
      TRIALS(I,J) = 99999
   125 CONTINUE
120 CONTINUE

*** DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS***

DO 130 I=1, K
CALL SRND(SEED, TEMP, N(I), MULT, SORT)
DO 135 J=1, N(I)
   UNIRV(I,J) = TEMP(J)
   IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
      TRIALS(I,J) = 0
   ELSE
      TRIALS(I,J) = 1
   END IF
135 CONTINUE
130 CONTINUE

***CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT***
    DO 150 I=1, K
       FIMI = 0
150 CONTINUE
    IONECT = 0

***CALCULATE THE FSUB I'S AND THE GRAND TOTAL NO. OF FAILURES***
    BIGF = 0
    DO 155 I=1, K
       DO 160 J=1, N(I)
          FI(I) = FI(I) + TRIALS(I,J)
160 CONTINUE
    IF (FI(I) .EQ. 0) THEN
       ZFPREP=ZFPREP+1
    ELSE
       ZFPREP=ZFPREP+1
    ENDIF
    BIGF = BIGF + FI(I)
    QHATI(I) = REAL(FI(I)) / N(I)
155 CONTINUE

***Change for Series-Parallel System with 2 out of 3 component***
    IF (FI(4) .EQ. 0) THEN
       ZFPREP=ZFPREP - 1
    ENDIF
    BIGF = BIGF - FI(4)
    FI(4) = 0
    DO 161 I=1,N(4)
       HFI=0
       DO 162 J=1,2
          CALL SRND(SEED, PTEMP(J), N(4), MULT, SORT)
          IF (PTEMP(J) .GT. 1-S4) THEN
             HFI = HFI + 1
          ENDIF
162 CONTINUE
    IF (HFI .EQ. 2) THEN
       FI(4) = FI(4) + 1
    ENDIF
161 CONTINUE
    BIGF = BIGF + FI(4)
    IF (FI(4) .EQ. 0) THEN
ZFPREP = ZFPREP + 1
ELSE
ENDIF
IF (FI(3) .EQ. 0) THEN
ZFPREP = ZFPREP + 1
ENDIF
BIGF = BIGF - FI(3)
FI(3) = 0
DO 163 I=1,N(3)
HFI=0
DO 164 J=1,3
CALL SRND(SEED, PTEM(J), N(3), MULT, SORT)
IF (PTEMP(J) .GT. 1-S3) THEN
HFI = HFI + 1
ENDIF
CONTINUE
164 CONTINUE
IF (HFI .GE. 2) THEN
FI(3) = FI(3) + 1
ENDIF
163 CONTINUE
BIGF = BIGF + FI(3)
IF (FI(3) .EQ. 0) THEN
ZFPREP = ZFPREP + 1
ELSE
ENDIF
DO 19 I=1,K
QI(I) = MSTRQ * AI(I)
QHAT(I) = REAL(FI(I)) / N(I)
19 CONTINUE
QI(4) = QI(4)**2
P3 = 1. - QI(3)
QI(3) = 1. - ((3.*P3**2*QI(3))+(P3**3))
*** COUNTS NUMBER OF COMPONENTS THAT HAVE FAILED ***
DO 156 I=1,K
IF (FI(I) .NE. 0) IONCT=IONCT+1
156 CONTINUE
*** CASE WHERE NO COMPONENTS HAVE ANY FAILURES ***
IF (BIGF .EQ. 0) THEN
LOOPSO(LOOP)=' *ZERO* '
ZFAILS = ZFAILS + 1
AVGN=0.0
DO 200 I=1,K
AVGN=AVGN+REAL(N(I))
200 CONTINUE
AVGN=AVGN/REAL(K)
DO 205 ALF=1, MAXALF
RSHAT(ALF,LOOP)= ALFA(ALF)**(1./AVGN)
IF (FLAG .EQ. 1) THEN
RHTSTR(ALF,LOOP)=ALFA(ALF)**(1./N(1))
ELSE
ENDIF
* PRINT *,LOOP', LOOP, ' RSHAT', RSHAT(ALF,LOOP)
205 CONTINUE
    DEGFR(LOOP) = 2.
    GO TO 10
ELSE
    FAILS = FAILS + 1
END IF

***COUNTS NUMBER OF COMPONENTS THAT FAIL RECORDS NO. COMPT TESTS***
    FCT=0
    DO 202 I=1,K
        IF (FI(I).NE.0) THEN
            FCT=FCT+1
            NTEST=N(I)
        ENDIF
    202 CONTINUE

***FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES***
    CALL RMAX(QHATI, K, QHTMAX, QINDX)

***CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)***
***IF COMPONENT HAS NO FAILURES AHAT SUB I IS ZERO***
    SUMNAI = 0.
    DO 165 I=1, K
        AHATI(I) = QHATI(I) / QHTMAX
        SUMNAI = SUMNAI + N(I) * AHATI(I)
    165 CONTINUE

***1 COMPONENT FAILURE SERIES SYSTEM***
    IF (FCT.EQ.1) THEN
        LOOPSO(LOOP) = 'ONECF'
        DO 305 ALF=1,MAXALF
            SUC=REAL(NTEST-BIGF)
            FDEG1=2.* (REAL(BIGF)+1.)
            FDEG2=2.*SUC
            STUD=FIN(I.-ALFA(ALF),FDEG1,FDEG2)
            RSHAT(ALF,LOOP)=SUC/(SUC+(REAL(BIGF)+1.)*STUD)
            PRINT *, SUC=SUC, FAIL'=BIGF,'NTEST='NTEST
            PRINT *, ALFA=ALFA(ALF), RSHAT='RSHAT(ALF,LOOP)
            IF (FLAG.EQ.1) THEN
                RHTSTR(ALF,LOOP)=RSHAT(ALF,LOOP)
            ELSE
                ENDIF
        305 CONTINUE
    ELSE
    ENDIF

***CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY***
    DEGFR(LOOP) = 2 * (1 + BIGF)
    DO 170 ALF=1, MAXALF
        CALL MDCHI(1 - ALFA(ALF), DEGFR(LOOP), CHISQ(ALF, LOOP), ERROR)
        QHTUPR(ALF, LOOP) = CHISQ(ALF, LOOP) / (2 * SUMNAI)
IF(FLAG.EQ.1) THEN
  RHTSTR(ALF,LOOP) = 1 -(CHISQ(ALF,LOOP) / REAL(2*N(1)))
ELSE
END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNTS. IN SERIES***

IF (FCT .NE. 1) THEN
  CALL RHSRSR(QHTUPR(ALF,LOOP), AHATI,K, RSHAT(ALF,LOOP))
ENDIF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***

IF (IONECT .NE. 1) THEN
  CALL RHTBRG(QHTUPR(ALF,LOOP),AHATI,K,RSHTBR(ALF,LOOP))
ENDIF

170 CONTINUE

***EXACTLY 1 COMPONENT FAILS AND REDUNDANT COMPONENT***

IF ((IONECT .EQ. 1) .AND. (K .EQ. 5)) THEN
  DO 207 I=1, K
       NIREAL(I) = REAL(N(I))
  CONTINUE
  CALL USHNMX(NIREAL,K,INC,NIMIN,NIMAX)
  DO 206 ALF=1,MAXALF
       RSHTBR(ALF,LOOP)=ALFA(ALF)**(1./NIMIN)
  CONTINUE
ENDIF

***THIS ELSE AND ENDF IF ARE FOR THE TEST AGAINST MAXRUN***

ELSE
  WRITE(1,'(''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',
+'' ALLOWED OF:''',''I6)') TOTREP
  GOTO 9999
END IF
GOTO 10
END IF

C WRITE(2,'(''UNSORTED RSHAT 1 IS:'',/10(F8.5))')
C +(RSHAT(1,LOOP), LOOP=1, MAXREP)
C WRITE(2,'(''UNSORTED RSHAT 2 IS:'',/10(F8.5))')
C +(RSHAT(2,LOOP), LOOP=1, MAXREP)
C IF(FLAG.EQ.1) THEN
C WRITE(2,'(''UNSORTED RHTSTR 1 IS:'',/10(F8.5))')
C +(RHTSTR(1,LOOP), Loop=1, MAXREP)
C WRITE(2,'(''UNSORTED RHTSTR 2 IS:'',/10(F8.5))')
C +(RHTSTR(2,LOOP), Loop=1, MAXREP)
C ELSE
C END IF
C IF(K.EQ.5) THEN
C WRITE(2,'(''UNSORTED RSHTBR 1 IS:'',/10(F8.5))')
C +(RSHTBR(1,LOOP), Loop=1, MAXREP)
C WRITE(2,'(''UNSORTED RSHTBR 2 IS:'',/10(F8.5))')
C
+(RSHTBR(2,LOOP), LOOP=1, MAXREP)
ELSE
ENDIF
WRITE (2,(''ZERO AND ONE FAILURE REPS: '',/10(A8)''))
+(LOOPSO(LOOP),LOOP=1,MAXREP)

***SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)***

DO 700 ALF=1, MAXALF
   DO 800 REPS=1, MAXREP
      TRANSQ(REPS) = QHTUPR(ALF,REPS)
      TRANR(REPS) = RSHAT(ALF,REPS)
      TRNSTR(REPS) = RHTSTR(ALF,REPS)
      TRANBR(REPS) = RSHTBR(ALF,REPS)
   800 CONTINUE
   CALL SHSORT(TRANSQ,KEY1,MAXREP)
   CALL SHSORT(TRANR,KEY2,MAXREP)
   CALL SHSORT(TRNSTR,KEY3,MAXREP)
   CALL SHSORT(TRANBR,KEY4,MAXREP)
   DO 900 REPS=1, MAXREP
      QHTUPR(ALF,REPS) = TRANSQ(REPS)
      RSHAT(ALF,REPS) = TRANR(REPS)
      RHTSTR(ALF,REPS) = TRNSTR(REPS)
      RSHTBR(ALF,REPS) = TRANBR(REPS)
   900 CONTINUE
700 CONTINUE

***PRINT OUTPUT REPORT HEADINGS***

WRITE(1,6666)
WRITE(1,6667) MAXREP
WRITE(1,6668) K
WRITE(1,6669)
IF(K.EQ.5) THEN
   WRITE(1,6699)
ELSE
ENDIF
WRITE(1,6670) MSTRQ
WRITE(1,6671)
WRITE(1,C1C15)
WRITE(1,3334) AI
WRITE(1,0007)
WRITE(1,C1C15)
WRITE(1,3334) QI
WRITE(1,0005)
WRITE(1,C1C15)
WRITE(1,3335) N
WRITE(1,6674)

***COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN. (SERIES SYSTEM)***

CALL RSRS(QI,K,RS)
WRITE(1,'(''THE TRUE SERIES SYSTEM '')',
RELIABILITY VALUE IS '',T51,F8.5)') RS
CALL RBRIDG(QI,K,RSBRDG)
IF(K.EQ.5) THEN
  WRITE(1,'(''THE TRUE BRIDGE STRUCTURE '',
+''RELIABILITY VALUE IS: '',T51,F8.5)') RSBRDG
ELSE
END IF
WRITE(1,6675)

***COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO***
***RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)***

IF(FLAG.EQ.1) THEN
  WRITE(1,5755)
ELSE
END IF
DO 450 ALF=1, MAXALF
  QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
  DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
  DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
IF(FLAG.EQ.1) THEN
  DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
  WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
  WRITE(1,5657) DELSTR(ALF)
ELSE
END IF
END IF
IF(K.EQ.5) THEN
  DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
  WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
  WRITE(1,5667) DELBRG(ALF)
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)
450 CONTINUE

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
***
WRITE(1,6676)
DO 400 ALF=1,MAXALF
  TRUQNT(ALF) = 0
  DO 500 REPS=1, MAXREP
    DIFF(REPS) = RS - RSHAT(ALF,REPS)
  500 CONTINUE
  DO 600 REPS=1, MAXREP
    IF(ABS(DIFF(REPS)).LE.EPS) THEN
      TRUQNT(ALF) = REPS
      WRITE(1,'(''TRUE CONFIDENCE LIMIT IS: '',F8.4)')
    + (TRUQNT(ALF) / REAL(MAXREP)) * 100.
  600 CONTINUE

47
ELSEIF(DIFF(REPS).LT.0.) THEN
  TRUQNT(ALF) = REPS
  GO TO 610
ELSE
END IF

600 CONTINUE

610 IF(TRYQNT(ALF).EQ.0.) THEN
  WRITE(1,4443) ALFA(ALF)
  WRITE(1,'(1,1,1/1"THE SMALLEST",
  +"DIFFERENCE BETWEEN RS AND RSHAT IS:"1,F10.5)') DIFF(MAXREP)
ELSEIF(TRYQNT(ALF).EQ.1.) THEN
  WRITE(1,4442) ALFA(ALF)
  WRITE(1,'(1,"ALL RSHAT",
  +"ARE GREATER THAN RS")')
ELSEIF(ABS(DIFF(TRYQNT(ALF))).-LE.ABS(DIFF(TRYQNT(ALF)-1))) THEN
  WRITE(1,4444) ALFA(ALF),
  ((TRYQNT(ALF)-1)/REAL(MAXREP)) * 100.
  WRITE(1,4445) RSHAT(ALF,TRYQNT(ALF))
  WRITE(1,4447)
ELSE
  WRITE(1,4444) ALFA(ALF),
  ((TRYQNT(ALF)-1)/REAL(MAXREP)) * 100.
  WRITE(1,4445) RSHAT(ALF,TRYQNT(ALF)-1)
  WRITE(1,4447)
END IF

620 END IF

400 CONTINUE

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

*** ************ RSHTBR (BRIDGE) ************ ***

IF(K.EQ.5) THEN
  DO 501 ALF=1,MAXALF
  TRUQNT(ALF) = 0
  DO 511 REPS=1, MAXREP
    DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
  511 CONTINUE
  DO 601 REPS=1, MAXREP
    IF(ABS(DIFF(REPS)).LE.EPS) THEN
      TRUQNT(ALF) = REPS
      WRITE(1,'(1,"'1","/"TRUE CONFIDENCE LIMIT IS:",
      +"F8.4")')
      + (TRUQNT(ALF)/REAL(MAXREP)) * 100.
    ELSEIF(DIFF(REPS).LT.0.) THEN
      TRUQNT(ALF) = REPS
    GO TO 621
    ELSEIF(DIFF(REPS).LT.0.) THEN
      TRUQNT(ALF) = REPS
    GO TO 611
    ELSE
      END IF
  601 CONTINUE

611 IF(TRYQNT(ALF).EQ.0.) THEN
  WRITE(1,4443) ALFA(ALF)
  WRITE(1,'(1,"'1","/"THE SMALLEST",
  +"DIFFERENCE BETWEEN RS AND RSHAT IS:"1,F10.5)') DIFF(MAXREP)
'DIFFERENCE BETWEEN RSRBDG AND RSHTBR IS: 

`   F10.5`) DIff(MAXREP)`

ELSEIF(TRUQNT(ALF).EQ. 1.) THEN
   WRITE(1,4442) ALFA(ALF)
   WRITE(1,('(1'',''/''ALL RSHTBR'',
   '   ARE GREATER THAN RSRBDG''))
   ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1))) THEN
   WRITE(1,4444) ALFA(ALF),
   (TRUQNT(ALF) / REAL(MAXREP)) * 100.
   WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
   WRITE(1,4447) END IF

END IF

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

IF(FLAG.EQ.1) THEN
   DO 4400 ALF=1,MAXALF
      TRUQNT(ALF) = 0
      DO 5500 REPS=1, MAXREP
         DIFF(REPS) = RS - RHTSTR(ALF,REPS)
         CONTINUE
      DO 6600 REPS=1, MAXREP
         IF(ABS(DIFF(REPS)).LE.EPS) THEN
            TRUQNT(ALF) = REPS
            WRITE(1,('(1'',''/''TRUE CONFIDENCE LIMIT IS: '',
            F8.4)')
            (TRUQNT(ALF) / REAL(MAXREP)) * 100.
            GO TO 6620
         ELSEIF(DIFF(REPS).LT.0.) THEN
            TRUQNT(ALF) = REPS
            GO TO 6610
         ELSE
            END IF
         CONTINUE
      END IF
   END IF
WRITE(1,4444) ALFA(ALF),
( (TRUQNT(ALF) / REAL(MAXREP)) * 100.
WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
+ ( (TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
END IF
CONTINUE
ELSE
END IF

***PRINT THE ARRAYS PERTINENT TO THE OUTPUT OF EACH REPLICATION***

IF(PRNT.EQ.1) THEN
I = 1
185 WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
+ ALFA(SELCTB), ALFA(SELCTB), ALFA(SELCTA), ALFA(SELCTA),
+ ALFA(SELCTB)
175 IF(I.GE.(MAXREP + 1)) THEN
GOTO 180
ELSE
I = I + 70
WRITE(1,'(''++'')'
GOTO 185
ELSE
WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
+ CHISQ(2,I), QHTUPR(2,I)
END IF
IF(I+70.LE.MAXREP) THEN
WRITE(1,3337) I+70, INT(DEGFR(I+70)), CHISQ(1,I+70),
+ QHTUPR(1,I+70), CHISQ(2,I+70), QHTUPR(2,I+70)
ELSE
I = I + 1
GOTO 175
180 END IF
ELSE
END IF
ENDIF
9999 WRITE(1,'(''THE TOTAL NO. OF REPS WAS: ',I8')') TOTREP
WRITE(1,'(''THE TOTAL NO. OF EFFECTIVE REPS WAS: ',I8')') LOOP
WRITE(1,'(''THE TOTAL NO. OF NO FAILURE RUNS WAS: ',I8')') ZFAILS
WRITE(1,'(''AVERAGE NO. OF COMPONENTS PER REPLICATION WITH ''
+ 'NO FAILURES: F5.2)') ZFPREP / MAXREP
WRITE(1,'(''THE TOTAL NO. OF RUNS WITH FAILURES WAS: ',I8')') FAILS
0008 FORMAT (/ 3X,'C 1',5X,'C 2',
+ 5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
+ 'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,
+ 'C 12',4X,'C 13',4X,'C 14',4X,'C 15'
)
0009 FORMAT(/1X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
+ 'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')',
)
APPENDIX F. FORTRAN CODE FOR THE PREFERRED LOWER CONFIDENCE LIMIT FOR SYSTEM RELIABILITY (PARALLEL SYSTEM)

PROGRAM ZFYSCN

*******************************************************************************
* * TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE
* * ZERO FAILURES ALLOWED; NO SCALING
* * AUTHOR: E. F. BELLINI, LT, USN
* * MODIFIED BY: LT VALERIE A. COVINGTON, USN (MAR 90)
* * DATE: NOV 89
* * *
* THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY
* OF THEIR COMPONENTS
* *
* IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.
* THE REXX EXEC PROGRAM
* 'B1' CALLS THE I Files TO BE READ AND NAMES THE 12 OUTPUT
* FILES RESULTING THE 12 CONSECUTIVE RUNS. BY EDITING THE
* INDEX COUNTERS 1, 1 OF THE 'B1' EXEC ONE CAN RUN ANY USER-
* SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.
* *
* VARIABLES USED
* *
* AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT
* A1 : INPUT WEIGHTS FOR EACH COMPONENT
* ALFA : LEVELS OF SIGNIFICANCE
* BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION
* CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE
* CIC15 : FORMAT LABEL
* DEGFR : DEGREES OF FREEDOM
* DELBRG : DIFFERENCE FOR BRIDGE SYSTEM
* DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM
* DELTAR : DIFFERENCE FOR SERIES SYSTEM
* DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.)
* EPS : SMALL QUANTITY(CONSTANT)
* ERROR : PARAMETER FOR IMSL ROUTINE
* FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE
* FI : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)
* FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS
* INC : INCREMENT STEP SIZE FOR ROUTINE USMNMX
* KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS
* LOOP : COUNTS NO. OF REPLICATION PERFORMED
* MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)
* MAXREP : MAX NO. OF REPLICATIONS
* MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED
* MSTRQ : MASTER UNRELIABILITY (USED WITH AI'S TO CALC. QI'S)
* MULT : MULTIPLIER FOR RANDOM NO. GENERATOR SRND
* N : NO. OF MISSION TEST FOR EACH COMPONENT
* NIMAX : MAX NO. OF MISSION TESTS
* NIMIN : MIN NO. OF MISSION TESTS
* NIINDEX : INDEX NO. OF MAX NO. OF MISSION TESTS
* NIREAL : NO. OF MISSION TESTS TRANSFORMED TO REAL
* NHAK : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL
* NPRNT : FLAG FOR DETAILED REPORT OUTPUT
* PRNT : SAME AS ABOVE (PARAMETER)
* QHTI : UNRELIABILITY ESTIMATES FOR EACH COMPONENT
* QHTMAX : LARGEST QHTI
* QHTUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY
* QI : INPUT UNRELIABILITY FOR EACH COMPONENT
* QINDEX : INDEX
* QUANTL : QUANTILE
* REPSHD : REPLICATIONS HEADING FORMAT NUMBER
* RHTSTR : SERIES SYSTEM RELIABILITY ESTIMATE (CLOSED FORM)
* RS : TRUE SERIES SYSTEM RELIABILITY
* RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY
* RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE
* QHATI : QUANTILE
* SEED : PARAMETER
* SELECTA : SIGNIFICANCE LEVEL SELECTION
* SELECTB : SIGNIFICANCE LEVEL SELECTION
* SORT : PARAMETER FOR ROUTINE SRND
* SUMNAI : SUM OF THE PRODUCT OF NI'S AND AI'S
* TEMP : TEMPORARY ARRAY
* TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS
* TRANBR : TEMPORARY ARRAY
* TRANSQ : TEMPORARY ARRAY
* TRANSR : TEMPORARY ARRAY
* TRIALS : BERNOULLI TRIALS ARRAY (2-DIM)
* TRNSTR : TEMPORARY ARRAY
* TRUQNT : TRUE QUANTILE
* UNIRV : UNIFORM RANDOM DEVIATES (2-DIM)
* ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES
* ZFPREP : NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION

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

PARAMETER (KK=10,MAXALF=2,NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=1.000001)
REAL*4 UNIRV(15,1000), TEMP(1000), QI(KK), AI(KK), AHTI(KK)
REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5), ALFA(5), SUMNAI, RSHAT(MAXALF, MAXREP), RS
REAL*4 KEY1(MAXREP), KEY2(MAXREP), KEY3(MAXREP), TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF, MAXREP), CHISQ(MAXALF, MAXREP)
REAL*4 QUPA1(MAXREP), QUPA2(MAXREP), RHTSTR(MAXALF, MAXREP)
REAL*4 DELTAR(MAXALF), TRANSQ(MAXREP), TRANSR(MAXREP), DIFF(MAXREP)
REAL*4 DELSTR(MAXALF), NMIN, NMAX, NIREAL(KK)
REAL*4 RSHTBR(MAXALF, MAXREP), DELBRG(MAXALF), KEY4(MAXREP)
REAL*4 TRANBR(MAXREP), RSBRDG, MSTRQ
REAL*4 ZFPREP

74
REAL*4 AVGN,SUC,STUD,PTEMP(10), FDEG1,FDEG2,P3,S3,S4

INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTK, TOTREP
INTEGER CIC15, REPSHD, SELCTB, ALF, FLAG, LOOP, PRNT
INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS, INC
INTEGER NTEST, FCT, HFI

CHARACTER*8 LOOPSO(MAXREP)

DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
DATA ALFA/.20,.050/

ASSIGN 8 TO CIC15
ASSIGN 9 TO REPSHD

* CALL COMPRS
PRNT = NPRNT

DO 12 I=1,KK
   AI(I) = 9999.
   N(I) = 99999999
12 CONTINUE

READ(03,*)K,MSTRQ

DO 11 I=1,K
   READ(03,*) AI(I), N(I)
11 CONTINUE

IF(K. NE. 5) THEN
   WRITE(1,'("WARNING: BRIDGE STRUCTURE ",
       "ONLY USES THE FIRST 5 COMPONENTS")')
ELSE
   END IF

***INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS, ***
* RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS *
* AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN *
***ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER ***

DO 172 ALF=1,MAXALF
   DO 173 REPS=1,MAXREP
      QHTUPR(ALF,REPS) = 0.
      RSHAT(ALF,REPS) = 0.
      RHTSTR(ALF,REPS) = 0.
      RSHTBR(ALF,REPS) = 0.
      LOOPSO(REPS) = '****** '
173 CONTINUE
172 CONTINUE

***SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS***

75
FLAG=1
DO 50 I=1,K-1
   IF((N(I) - N(I+1)).NE.0) THEN
      FLAG=0
   ELSE
      END IF
50 CONTINUE
PRINT *, 'FLAG IS:', FLAG

***MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)***

ZFPREP = 0.
ZFAILS = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
   LOOP = LOOP + 1
   IF(TOTREP.LT.MAXRUN) THEN
      TOTREP = TOTREP + 1
   END IF
   SELCTA = 1
   SELCTB = 2
   ***FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT***
   ***OF THE SUBROUTINE SHSORT
   DO 95 REPS=1, MAXREP
      KEY1(REPS) = REPS
      KEY2(REPS) = REPS
      KEY3(REPS) = REPS
      KEY4(REPS) = REPS
   95 CONTINUE
   ***CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE***
   ***CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.***
   CALL IMAX(N,K,NMAX,NINDX)
   ***CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S***
   DO 115 I=1, K
      QI(I) = MSTRQ * AI(I)
   115 CONTINUE
   C
      S3=QI(3)
      S4=QI(1)**.20
      DO 120 I=1,15
         DO 125 J=1,500
            UNIRV(I,J) = 999.
            TRIALS(I,J) = 99999
         125 CONTINUE
      120 CONTINUE
   *** DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS***
   DO 130 I=1, K
        
76
CALL SRND(SEED, TEMP, N(I), MULT, SORT)
DO 135 J=1, N(I)
   UNIRV(I,J) = TEMP(J)
   IF (UNIRV(I,J) .LE. 1 - QI(I)) THEN
      TRIALS(I,J) = 0
   ELSE
      TRIALS(I,J) = 1
   END IF
135  CONTINUE
130  CONTINUE

***CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT***

DO 150 I=1, K
   FI(I) = 0
150  CONTINUE
IONECT = 0

***CALCULATE THE F'S AND THE GRAND TOTAL NO. OF FAILURES***

BIGF = 0
DO 155 I=1, K
   DO 160 J=1, N(I)
      FI(I) = FI(I) + TRIALS(I,J)
   160 CONTINUE
   IF (FI(I) .EQ. 0) THEN
      ZFPREP = ZFPREP + 1
   ELSE
      ENDIF
      BIGF = BIGF + FI(I)
   QHATI(I) = REAL(FI(I)) / N(I)
155  CONTINUE

***Changes for parallel system***

IF (FI(1) .EQ. 0) THEN
   ZFPREP = ZFPREP - 1
ENDIF
BIGF = BIGF - FI(1)
FI(1) = 0
DO 161 I=1, N(1)
   HFI = 0
   DO 162 J=1, 5
      CALL SRND(SEED, PTEMP(J), N(1), MULT, SORT)
      IF (PTEMP(J) .GT. I - S4) THEN
         HFI = HFI + 1
      ENDIF
   162 CONTINUE
   IF (HFI .EQ. 5) THEN
      FI(1) = FI(1) + 1
   ENDIF
161  CONTINUE
BIGF = BIGF + FI(1)
IF (FI(1) .EQ. 0) THEN
77
ZFPREP = ZFPREP + 1
ELSE
ENDIF
C IF (FI(3) .EQ. 0) THEN
C ZFPREP = ZFPREP - 1
C ENDIF
C BIGF = BIGF - FI(3)
C FI(3) = 0
C DO 163 I=1,N(3)
C HFI=0
C DO 164 J=1,3
C CALL SRND(SEED, PTEMP(J), N(3), MULT, SORT)
C IF (PTEMP(J) .GT. 1-S3) THEN
C HFI = HFI + 1
C ENDIF
C164 CONTINUE
C IF (HFI .GE. 2) THEN
C FI(3) = FI(3) + 1
C ENDIF
C163 CONTINUE
C BIGF = BIGF + FI(3)
C ELSE
C ENDIF
C DO 19 I=1,K
C QI(I) = MSTRQ * AI(I)
C QHATI(I) = REAL(FI(I)) / N(I)
C19 CONTINUE
C QI(4) = QI(4)**2
C P3 = 1. - QI(3)
C QI(3) = 1. - ((3.*P3**2* QI(3))+(P3**3))
***COUnTS number Of components That have failed***
C DO 156 I=1,K
C IF (FI(I) .NE. 0) IONECT = IONECT + 1
C156 CONTINUE
***case where no components have any failures***
C IF(BIGF.EQ.0) THEN
C LOOPSO(LOOP)= ' *ZERO* '
C ZFAILS = ZFAILS + 1
C AVGN=0.0
C DO 200 I=1,K
C AVGN=AVGN+REAL(N(I))
C200 CONTINUE
C AVGN=AVGN/REAL(K)
C DO 205 ALF=1, MAXALF
C RSHAT(ALF,LOOP)= ALFA(ALF)**(1./AVGN)
C IF(FLAG.EQ.1) THEN
C RHTSTR(ALF,LOOP)= ALFA(ALF)**(1./N(1))
C ELSE
C END IF
C END IF
* PRINT *,LOOP',LOOP,'RSHAT',RSHAT(ALF,LOOP)
205    CONTINUE
        DEGFR(LOOP) = 2.
        GO TO 10
    ELSE
        FAILS = FAILS + 1
    END IF

***COUNTS NUMBER OF COMPONENTS THAT FAIL RECORDS NO. COMPT TESTS***
    FCT=0
    DO 202 I=1,K
       IF (FI(I) .NE. 0) THEN
          FCT=FCT+1
          NTEST=N(I)
       ENDIF
    202 CONTINUE

***FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES***
    CALL RMAX(QHATI, K, QHTMAX, QINDX)
    IF (LOOP .EQ. 1) THEN
    ENDIF

***CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)***
***IF COMPONENT HAS NO FAILURES AHAT SUB I IS ZERO***
    SUMNAI = 0.
    DO 165 I=i, K
       AHATI(I) = QHATI(I) / QHTMAX
       SUMNAI = SUMNAI + N(I) * AHATI(I)
    165 CONTINUE

***1 COMPONENT FAILURE SERIES SYSTEM***
    IF (FCT .EQ. 1) THEN
        LOOPSO(LOOP)=' *ONECF* '
        DO 305 ALF=1,MAXALF
           SUC=REAL(NTEST-BIGF)
           FDEG1=2.*(REAL(BIGF)+I.)
           FDEG2=2.*SUC
           STUD=FIN(I.-ALFA(ALF),FDEG1,FDEG2)
           RSHAT(ALF,LOOP)=SUC/(SUC+(REAL(BIGF)+I.)*STUD)
           * PRINT *, 'SUC=',SUC,'FAIL=',BIGF,'NTEST=',NTEST
           * PRINT *, 'FIN=',STUD
           * PRINT *, 'ALFA=',ALFA(ALF),'RSHAT=',RSHAT(ALF,LOOP)
           IF (FLAG .EQ. 1) THEN
              RHTSTR(ALF,LOOP)=RSHAT(ALF,LOOP)
           ELSE
           ENDIF
        305 CONTINUE
    END IF

***CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY***
    DEGFR(LOOP) = 2 * (1 + BIGF)
    DO 170 ALF=1, MAXALF

79
CALL MDCHI(1 - ALFA(ALF), DEGFR(LOOP), CHISQ(ALF, LOOP), ERROR)
• QHTUPR(ALF, LOOP) = CHISQ(ALF, LOOP) / (2 * SUMNAIL)
IF(FLAG.EQ.1) THEN
   RHTSTR(ALF, LOOP) = 1 - (CHISQ(ALF, LOOP) / REAL(2*N(1)))
ELSE
END IF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPONENTS IN SERIES***
IF (FCT .NE. 1) THEN
   CALL RHSRS(QHTUPR(ALF, LOOP), AHATI, K, RSHAT(ALF, LOOP))
ENDIF

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***
IF (IONECT .NE. 1) THEN
   CALL RHITBG(QHTUPR(ALF, LOOP), AHATI, K, RSHTBR(ALF, LOOP))
ENDIF

170 CONTINUE

***EXACTLY 1 COMPONENT FAILS AND REDUNDANT COMPONENT***
IF (((IONECT .EQ. 1) .AND. (K .EQ. 5)) THEN
   DO 207 I=1, K
      NREAL(I) = REAL(N(I))
   CONTINUE
   CALL USHNNX(NREAL, K, INC, NMIN, NMAX)
   DO 206 ALF=1, MAXALF
      RSHTBR(ALF, LOOP) = ALFA(ALF)**(1./NMIN)
   CONTINUE
ENDIF

***THIS ELSE AND ENDFIRST ARE FOR THE TEST AGAINST MAXRUN***
ELSE
   WRITE(1,'(''PROGRAM EXCEEDED THE MAX NO. OF RUNS'', +'' ALLOWED OF:'','I6)' ) TOTREP
   GOTO 9999
END IF
GOTO 10
ENDIF

C WRITE(2,'(''UNSORTED RSHAT 1 IS:'','/10(F8.5))')
C +(RSHAT(1, LOOP), LOOP=1, MAXREP)
C WRITE(2,'(''UNSORTED RSHAT 2 IS:'','/10(F8.5))')
C +(RSHAT(2, LOOP), LOOP=1, MAXREP)
C IF(FLAG.EQ.1) THEN
C WRITE(2,'(''UNSORTED RHTSTR 1 IS:'','/10(F8.5))')
C +(RHTSTR(1, LOOP), LOOP=1, MAXREP)
C WRITE(2,'(''UNSORTED RHTSTR 2 IS:'','/10(F8.5))')
C +(RHTSTR(2, LOOP), LOOP=1, MAXREP)
C ELSE
C END IF
C IF(K.EQ.5) THEN
C WRITE(2,'(''UNSORTED RSHTBR 1 IS:'','/10(F8.5))')
C
C +(RSHTBR(1,LOOP), LOOP=1, MAXREP)
C WRITE(2, (''UNSORTED RSHTBR 2 IS: '',/10(F8.5))')
C +(RSHTBR(2,LOOP), LOOP=1, MAXREP)
C ELSE
C END IF
C WRITE(2,'(''ZERO AND ONE FAILURE REPS: '',/10(A8))')
C + (LOOPS0(LOOP),LOOP=1,MAXREP)

***SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)***

DO 700 ALF=1, MAXALF
    DO 800 REPS=1, MAXREP
        TRANSQ(REPS) = QHTUPR(ALF,REPS)
        TRANSR(REPS) = RSHAT(ALF,REPS)
        TRNSTR(REPS) = RHTSTR(ALF,REPS)
        TRANBR(REPS) = RSHTBR(ALF,REPS)
    800 CONTINUE
    CALL SHSORT(TRANSQ,KEY1 ,MAXREP)
    CALL SHSORT( TRANSR,KEY2 ,MAXREP)
    CALL SHSORT( TRNSTR,KEY3 ,MAXREP)
    CALL SHSORT(TRANBR,KEY4,MAXREP)
    DO 900 REPS=1, MAXREP
        QHTUPR(ALF,REPS) = TRANSQ(REPS)
        RSHAT(ALF,REPS) = TRANSR(REPS)
        RHTSTR(ALF,REPS) = TRNSTR(REPS)
        RSHTBR(ALF,REPS) = TRANBR(REPS)
    900 CONTINUE
700 CONTINUE

***PRINT OUTPUT REPORT HEADINGS***

WRITE(1,6666)
WRITE(1,6667) MAXREP
WRITE(1,6668) K
WRITE(1,6669)
IF(K.EQ.5) THEN
    WRITE(1,6699)
ELSE
    END IF
WRITE(1,6670) MSTRQ
WRITE(1,6671)
WRITE(1,C15)
WRITE(1,3334) AI
WRITE(1,0007)
WRITE(1,C15)
WRITE(1,3334) QI
WRITE(1,0005)
WRITE(1,C15)
WRITE(1,3335) N
WRITE(1,6674)

***COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN. (SERIES SYSTEM)***
*** AND FOR THE 5-COMPONENT BRIDGE STRUCTURE***
CALL RSRS(QI,K,RS)
WRITE(1,'(C'--',////''THE TRUE SERIES SYSTEM '',T51,F8.5)') RS
CALL RBRIDG(QI,K,RSBRDG)
IF(K.EQ.5) THEN
WRITE(1,'(C'--',////''THE TRUE BRIDGE STRUCTURE '',
+''RELIABILITY VALUE IS:'',T51,F8.5)') RSBRDG
ELSE
END IF
WRITE(1,1,6675)

***COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEORETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)***

IF(FLAG.EQ.1) THEN
WRITE(1,5755)
ELSE
END IF
DO 450 ALF=1, MAXALF
QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
IF(FLAG.EQ.1) THEN
DELSR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
WRITE(1,5657) DELSTR(ALF)
ELSE
END IF
IF(K.EQ.5) THEN
DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
WRITE(1,5667) DELBRG(ALF)
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)
450 CONTINUE
PRINT *, 'QUANTL(1) IS:', QUANTL(1)
PRINT *, 'QUANTL(2) IS:', QUANTL(2)

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

WRITE(1,1,6676)
DO 400 ALF=1, MAXALF
TRUQNT(ALF) = 0
DO 500 REPS=1, MAXREP
DIFF(REPS) = RS - RSHAT(ALF,REPS)
500 CONTINUE
DO 600 REPS=1, MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'(C'--',////''TRUE CONFIDENCE LIMIT IS:'',
+''********* RSHAT *********''',72,E11.3)') MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
WRITE(1,5667) DELBRG(ALF)
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)
400 CONTINUE
PRINT *, 'QUANTL(1) IS:', QUANTL(1)
PRINT *, 'QUANTL(2) IS:', QUANTL(2)

WRITE(1,1,6676)
***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***

**RSHTBR (BRIDGE)**

**IF(K.EQ.5) THEN**

**DO 401 ALF=1,MAXALF**

TRUQNT(ALF) = 0

**DO 501 REPS=1, MAXREP**

DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)

**501 CONTINUE**

**DO 601 REPS=1, MAXREP**

**IF(ABS(DIFF(REPS)).LE.EPS) THEN**

TRUQNT(ALF) = REPS

WRITE(1,'(F8.4)')

( (TRUQNT(ALF) / REAL(MAXREP)) * 100."

GO TO 621

ELSEIF(DIFF(REPS).LT.0.) THEN

TRUQNT(ALF) = REPS

GO TO 611

ELSE

END IF

**601 CONTINUE**

**611 IF(TRUQNT(ALF).EQ.0.) THEN**

WRITE(1,442)ALF(ALF)

WRITE(1,'(C',/''ALL RSHAT'',

+'' ARE GREATER THAN RS'')')

ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))

** THEN**

WRITE(1,4444)ALF(ALF),

((TRUQNT(ALF) - 1) / REAL(MAXREP)) * 100.

WRITE(1,4446)RSHAT(ALF,TRUQNT(ALF))

ELSE

WRITE(1,4447)

ENDIF

**END IF**

**END IF**

**620 END IF**

**400 CONTINUE**
WRITEx1,4443) ALFA(ALF)
WRITE(1,'(', F10.5)', 'DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:', +
F10.5)', 'DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1,'(', 'ALL RSHTBR', '+
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.0.1) THEN
WRITE(1,4444) ALFA(ALF),
(DIFTRUQNT(ALF)/REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
((TRUQNT(ALF)-1)/REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
WRITE(1,4447)
621 END IF
401 CONTINUE
ELSE
END IF

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
***          ********** RHTSTR **********          ***

IF(FLAG.EQ.1) THEN
DO 4400 ALF=1,MAXALF
TRUQNT(ALF) = 0
DO 5500 REPS=1,MAXREP
DIFF(REPS) = RS - RHTSTR(ALF,REPS)
5500 CONTINUE
DO 6600 REPS=1,MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'((' ', './TRUE CONFIDENCE LIMIT IS:', +
F8.4)')
+ (TRUQNT(ALF)/REAL(MAXREP)) * 100.
GO TO 6620
ELSEIF(DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 6610
ELSE
END IF
6600 CONTINUE
6610 IF(TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1,'(', 'THE SMALLEST', +
F9.5)') DIFF(MAXREP)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1,'(', 'ALL RSHTBR', +
' ARE GREATER THAN RS'')')
84
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
    THEN
        WRITE(1,4444) ALFA(ALF),
        (TRUQNT(ALF) / REAL(MAXREP)) * 100.
        WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
        WRITE(1,4446)
    ELSE
        WRITE(1,4444) ALFA(ALF),
        ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
        WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
        WRITE(1,4447)
    END IF

6620  END IF
4400  CONTINUE
ELSE
END IF

***PRINT THE ARRAYS PERTINENT TO THE OUTPUT OF EACH REPLICATION***

IF(PRNT.EQ.1) THEN
    I = 1
    185  WRITE(1,REPSHD) ALFA(SELECTA), ALFA(SELECTA),
         +ALFA(SELECTB),ALFA(SELECTB),ALFA(SELECTA),ALFA(SELECTB),
         +ALFA(SELECTB)
    175  IF(I.GE.(MAXREP + 1)) THEN
         GOTO 180
    ELSE
             I = I + 70
             WRITE(1,'(''+'')')
             GOTO 185
         ELSE
             WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
             +CHISQ(2,I), QHTUPR(2,I)
         END IF
         IF(I + 70.LE.MAXREP) THEN
             WRITE(1,'(THE TOTAL NO OF REPS WAS:':'' ,I8)') TOTREP
             WRITE(1,'(THE TOTAL NO OF EFFECTIVE REPS WAS:''' ,I8)') LOOP
             WRITE(1,'(THE TOTAL NO OF NO FAILURE RUNS WAS:''' ,I8)') 2FAILS
             WRITE(1,'(AVERAGE NO. OF COMPONENTS PER REPLICATION WITH ''',
             +''NO FAILURES:''' ,F5.2)') ZFPREP / MAXREP
             WRITE(1,'(THE TOTAL NO OF RUNS WITH FAILURES WAS:''' ,I8)') FAILS
             9999 FORMAT (/ 3X,'C 1',5X,'C 2',
             +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
             +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,
             +'C 12',4X,'C 13',4X,'C 14',4X,'C 15')
         END IF
    180  END IF
ELSE
END IF

9999
TRUE CONFIDENCE LIMITS

END
APPENDIX G. FORTRAN CODE FOR ALTERNATE PROCEDURE A
FOR ESTIMATING THE LOWER CONFIDENCE LIMIT FOR SYSTEM
RELIABILITY

PROGRAM ZFYSCN

******************************************************************************
*                                                                         *
* TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE                             *
* ZERO FAILURES ALLOWED; NO SCALING                                       *
* AUTHOR: E. F. BELLINI, LT, USN                                          *
* DATE: NOV 89                                                            *
*                                                                         *
* THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE         *
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY         *
* OF THEIR COMPONENTS                                                     *
*                                                                         *
* IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12            *
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT            *
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.         *
* THE REXX EXEC PROGRAM                                                   *
* 'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT           *
* FILES RESULTING FROM THE 12 CONSECUTIVE RUNS. BY EDITING THE            *
* INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-SPECIFIC     *
* RUN FROM JUST ONE RUN TO ALL 12.                                        *
*                                                                         *
* VARIABLES USED                                                          *
*                                                                         *
* AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT                              *
* AI : INPUT WEIGHTS FOR EACH COMPONENT                                    *
* ALFA : LEVELS OF SIGNIFICANCE                                           *
* BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION                        *
* CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE                                *
* C1C15 : FORMAT LABEL                                                     *
* DEGFR : DEGREES OF FREEDOM                                               *
* DELBRG : DIFFERENCE FOR BRIDGE SYSTEM                                   *
* DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM                      *
* DELTAR : DIFFERENCE FOR SERIES SYSTEM                                   *
* DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.)                          *
* EPS : SMALL QUANTITY(CONSTANT)                                         *
* ERROR : PARAMETER FOR IMSL ROUTINE                                      *
* FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. ' FAILURE               *
* FI : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)*               *
* FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS                    *
* INC : INCREMENT STEP SIZE FOR ROUTINE USMNMX                            *
* KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT                              *
* KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT                              *
* KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT                              *
* KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT                              *
* KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS*               *
* LOOP : COUNTS NO. OF REPLICATION PERFORMED                               *
* MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)*           *
* MAXREP : MAX NO. OF REPLICATIONS                                         *

88
MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED
MSTRQ : MASTER UNRELIABILITY (USED WITH AI'S TO CALC. QI'S)
MULT : MULTIPLIER FOR RANDOM NO. GENERATOR SRND
N : NO. OF MISSION TEST FOR EACH COMPONENT
NIMAX : MAX NO. OF MISSION TESTS
NIMIN : MIN NO. OF MISSION TESTS
NINDX : INDEX NO. OF MAX NO. OF MISSION TESTS
NRE : NO. OF MISSION TEST TRANSFORMED TO REAL
NMAX : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL
NPRNT : FLAG FOR DETAILED REPORT OUTPUT
PRNT : SAME AS ABOVE (PARAMETER)
QHATI : UNRELIABILITY ESTIMATES FOR EACH COMPONENT
QHTMAX : LARGEST QHATI
QHUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY
QI : INPUT UNRELIABILITY FOR EACH COMPONENT
QINDX : INDEX
QUANTL : QUANTILE
REPSHD : REPLICATIONS HEADING FORMAT NUMBER
RHTSTR : SERIES SYSTEM RELIABILITY ESTIMATE (CLOSED FORM)
RS : TRUE SERIES SYSTEM RELIABILITY
RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY
RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE
SEED : PARAMETER
SELETA : SIGNIFICANCE LEVEL SELECTION
SELECB : SIGNIFICANCE LEVEL SELECTION
SORT : PARAMETER FOR ROUTINE SRND
SUMNAI : SUM OF THE PRODUCT OF NI'S AND AI'S
TEMP : TEMPORARY ARRAY
TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS
TRANBR : TEMPORARY ARRAY
TRANQ : TEMPORARY ARRAY
TRANSR : TEMPORARY ARRAY
TRIALS : BERNOULLI TRIALS ARRAY (2-DIM)
TRNSTR : TEMPORARY ARRAY
TRUQNT : TRUE QUANTILE
UNIRV : UNIFORM RANDOM DEVIATES (2-DIM)
ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES
ZFPREP : NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION

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

PARAMETER (KK=10, MAXALF=2, NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=0.000001)
REAL*4 UNIRV(15,1000), TEMP(1000), QI(KK), AI(KK), AHATI(KK)
REAL*4 QHATI(KK), NMAX, NMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5), AALFA(5), SUMNAI, RSHAT(MAXALF, MAXREP), RS
REAL*4 KEY1(MAXREP), KEY2(MAXREP), KEY3(MAXREP), TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF, MAXREP), CHISQ(MAXALF, MAXREP)
REAL*4 QUPA1(MAXREP), QUPA2(MAXREP), RHTSTR(MAXALF, MAXREP)
REAL*4 DELTA(MAXALF), TRANSQ(MAXREP), TRANSR(MAXREP), DIFF(MAXREP)
REAL*4 DELSTR(MAXALF), NIMIN, NIMAX, NIREAL(KK)
REAL*4 RSHTBR(MAXALF, MAXREP), DELBRG(MAXALF), KEY4(MAXREP)
REAL*4 TRANBR(MAXREP), RSBR, MSTRQ
REAL*4 ZFPREP

89
INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTK, TOTREP
INTEGER C1C15, REPSHD, SELCTB, ALF, FLAG, LOOP, PRNT
INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS, INC

DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/,
DATA DF/1,5,10,30,40/
DATA ALFA/.20,.050/>
DATA SORT/0/>

ASSIGN 8 TO ClC15
ASSIGN 9 TO REPSHD

CALL COMPRS
PRNT = NPRINT

DO 12 I=1,KK
     AI(I) = 9999.
     N(I) = 99999999
12 CONTINUE

READ(03,*)K,MSTRQ

DO 11 I=1,K
     READ(03,*) AI(I), N(I)
11 CONTINUE

IF(K.NE.5) THEN
     WRITE(1,'("WARNING: BRIDGE STRUCTURE ",
            +"ONLY USES THE FIRST 5 COMPONENT")')
ELSE
     END IF

***// INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS,  
***// RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS  
***// AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN  
***// ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER  

DO 172 ALF=1,MAXALF
   DO 173 REPS=1,MAXREP
      QHTUPR(ALF,REPS) = 0.
      RSHAT(ALF,REPS) = 0.
      RHTSTR(ALF,REPS) = 0.
      RSHTBR(ALF,REPS) = 0.
173 CONTINUE
172 CONTINUE

***// SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****

   FLAG=1
   DO 50 I=1,K-1
      IF((N(I) - N(I+1)).NE.0) THEN
         FLAG=0
      ELSE
         END IF
50 CONTINUE
CONTINUE
PRINT *, 'FLAG IS:', FLAG

***// MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)/***  

ZFREP = 0.
ZFAILS = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
    LOOP = LOOP + 1
    IF(TOTREP.LT.MAXUN) THEN
        TOTREP = TOTREP + 1
        SELCTA = 1
        SELCTB = 2

***// FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT  
***// OF THE SUBROUTINE SHSORT
DO 95 REPS=1, MAXREP
    KEY1(REPS) = REPS
    KEY2(REPS) = REPS
    KEY3(REPS) = REPS
    KEY4(REPS) = REPS
95 CONTINUE

***// CALCULATE: NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE
***// CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO. //***
   CALL IMAX(N,K,NMAX,NINDX)

***// CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S
   DO 115 I=1, K
       QI(I) = MSTRQ * AI(I)
115 CONTINUE

   DO 120 I=1,15
       DO 125 J=1,500
           UNIRV(I,J) = 999.
           TRIALS(I,J) = 99999
125 CONTINUE
120 CONTINUE

***// DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS //***
   DO 130 I=1, K
       CALL SRND(SEED, TEMP, N(I), MULT, SORT)
       DO 135 J=1, N(I)
           UNIRV(I,J) = TEMP(J)
           IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
               TRIALS(I,J) = 0
           ELSE
               TRIALS(I,J) = 1
           END IF
135 CONTINUE
130 CONTINUE

***// CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT  //***
    DO 150 I=1, K
       FI(I) = 0
150 CONTINUE

***// CALCULATE THE F_SUB I'S AND THE GRAND TOTAL NO. OF FAILURES
    BIGF = 0
    DO 155 I=1, K
       DO 160 J=1, N(I)
          FI(I) = FI(I) + TRIALS(I,J)
160 CONTINUE
    IF(FI(I).EQ.0) THEN
       ZFPREP = ZFPREP + 1
    ELSE
       END IF

***// CALCULATE THE QHAT SUB I'S: F SUB I'S DIVIDED BY N SUB I'S
    QHATI(I) = REAL(FI(I))/N(I)
    BIGF = BIGF + FI(I)
155 CONTINUE

***// CASE WHERE NO COMPONENTS HAVE ANY FAILURES  //***
    IF(BIGF.EQ.0) THEN
       ZFAILS = ZFAILS + 1
       DO 200 I=1, K
          NIREAL(I) = REAL(N(I))
200 CONTINUE
       CALL USMNMX(NIREAL,K,INC,NMIN,NIMAX)
       CALL HDFCHI(1 - ALFA(ALF),2.,CHISQ(ALF,LOOP),ERROR)
       RSHAT(ALF,LOOP) = 1 - (CHISQ(ALF,LOOP) / REAL(2*NMIN))
       IF(FLAG.EQ.1) THEN
          RHTSTR(ALF,LOOP) = 1 - (CHISQ(ALF,LOOP) / REAL(2*N(1)))
       ELSE
          END IF
205 CONTINUE
    IF(PRNT.EQ.1) THEN
       WRITE(1,0007)
       WRITE(1,1111) (UNIRV(I,J), I=1, K)
141 CONTINUE
       WRITE(1,0002)
       WRITE(1,12222) (TRIALS(I,J), I=1, K)
146 CONTINUE
       WRITE(1,0003)
       WRITE(1,C1C15)
WRITE(1,3333) FI
WRITE(1,0005)
WRITE(1,C1C15)
WRITE(1,3335) N
WRITE(1,0004)
WRITE(1,C1C15)
WRITE(1,3334) QHATI
WRITE(1,'(''THE MAXIMUM Q HAT SUB I IS: 'T40, F8.5')') QHTMAX
WRITE(1,'(''THE MAXI Q HAT SUB I IS ELMNT NO.: 'T40,I5')') QINDX
WRITE(1,'(''THE GRAND TOTAL NO. OF FAILURES IS: 'T40, I5')') BIGF
ENDIF
ELSE
DEGFR(LOOP) = 2.
GO TO 10
ELSE
FAILS = FAILS + 1
ENDIF
***// FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES
CALL RMAX(QHATI, K, QHTMAX, QINDX)
***// PRINT THE RESULT OF THE MAIN OPERATING ELEMENTS OF THE PROGRAM
IF(PRNT.EQ.1) THEN
WRITE(1,0007)
WRITE(1,C1C15)
WRITE(1,3334) QI
WRITE(1,0001)
WRITE(1,C1C15)
DO 140 J=1,NM
WRITE(1,1111) (UNIRV(I,J), I=1, K)
CONTINUE
140 WRITE(1,0002)
WRITE(1,C1C15)
DO 145 J=1,NMA
WRITE(1,2222) (TRIALS(I,J), I=1, K)
CONTINUE
145 WRITE(1,0003)
WRITE(1,C1C15)
WRITE(1,3333) FI
WRITE(1,0005)
WRITE(1,C1C15)
WRITE(1,3335) N
WRITE(1,0004)
WRITE(1,C1C15)
WRITE(1,3334) QHATI
WRITE(1,'(''THE MAXIMUM Q HAT SUB I IS: 'T40, F8.5')') QHTMAX
WRITE(1,'(''THE MAXI Q HAT SUB I IS ELMNT NO.: 'T40,I5')') QINDX
WRITE(1,'(''THE GRAND TOTAL NO. OF FAILURES IS: 'T40, I5')') BIGF
ELSE
ENDIF
***// CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)
SUMNAI = 0.
DO 165 I=1, K
\[ \text{AHATI}(i) = \frac{QHATI(i)}{QHTMAX} \]
\[ \text{SNMNAI} = \text{SNMNAI} + N(i) \times \text{AHATI}(i) \]

165 CONTINUE
IF(PRNT.EQ.1) THEN
  WRITE(1,006)
  WRITE(1,1515)
  WRITE(1,3334) AHATI
ELSE
END IF

***// CALCULATE 1 REPPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY

\[ \text{DEGFR}(\text{LOOP}) = 2 \times (1 + \text{BIGF}) \]

DO 170 ALF=1, MAXALF
  CALL MDCHI(1 - ALFA(ALF), DEGFR(LOOP), CHISQ(ALF,LOOP), ERROR)
  QHUPR(ALF,LOOP) = CHISQ(ALF,LOOP) / (2 \times \text{SNMNAI})
  IF(FLAG.EQ.1) THEN
    RHTSTR(ALF,LOOP) = 1 - (CHISQ(ALF,LOOP) / REALCZ*N(1))
  ELSE
  END IF
  * + (ALF,LOOP), ALFA(ALF)

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNTS. IN SERIES

CALL RHSRS(QHUPR(ALF,LOOP), AHATI, K, RSHAT(ALF,LOOP))
* +T40,F8.5) RSHAT(ALF,LOOP)

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE ***

CALL RHTBRG(QHUPR(ALF,LOOP), AHATI, K, RSHTBR(ALF,LOOP))

170 CONTINUE

***// THIS ELSE AND ENDFI ARE FOR THE TEST AGAINST MAXRUN ***************
ELSE
  WRITE(1,'("\text{\textquote双Program exceeded the max no. of runs",}
+A" ALLOWED OF: \text{\textquote双I6") TOTREP
  GOTO 9999
END IF
GOTO 10
END IF
WRITE(2,'("\text{\textquote双Unsorted RSHAT 1 IS: */10(F8.5")}
+(RSHAT(1,LOOP), LOOP=1, MAXREP)
WRITE(2,'("\text{\textquote双Unsorted RSHAT 2 IS: */10(F8.5")}
+(RSHAT(2,LOOP), LOOP=1, MAXREP)
IF(FLAG.EQ.1) THEN
  WRITE(2,'("\text{\textquote双Unsorted RHTSTR 1 IS: */10(F8.5")}
+(RHTSTR(1,LOOP), LOOP=1, MAXREP)
WRITE(2,'("\text{\textquote双Unsorted RHTSTR 2 IS: */10(F8.5")}
+(RHTSTR(2,LOOP), LOOP=1, MAXREP)
ELSE
END IF
IF(K.EQ.5) THEN
  WRITE(2,'("\text{\textquote双Unsorted RSHTBR 1 IS: */10(F8.5")}
+(RSHTBR(1,LOOP), LOOP=1, MAXREP)
WRITE(2,'("\text{\textquote双Unsorted RSHTBR 2 IS: */10(F8.5")}
+(RSHTBR(2,LOOP), LOOP=1, MAXREP)
+ (RSHTBR(2, LOOP), LOOP=1, MAXREP)
ELSE
END IF

***// SORT THE ARRAYS OF SYSTEM UNRELIABILITIES (1 FOR EACH CONF. LEVEL)

DO 700 ALF=1, MAXALF
DO 800 REPS=1, MAXREP
   TRANSQ(REPS) = QHTUPR(ALF, REPS)
   TRANSR(REPS) = RSHAT(ALF, REPS)
   TRNSTR(REPS) = RHTSTR(ALF, REPS)
   TRANBR(REPS) = RSHTBR(ALF, REPS)
800 CONTINUE
   CALL SHSORT(TRANSQ, KEY1, MAXREP)
   CALL SHSORT(TRANSR, KEY2, MAXREP)
   CALL SHSORT(TRNSTR, KEY3, MAXREP)
   CALL SHSORT(TRANBR, KEY4, MAXREP)
DO 900 REPS=1, MAXREP
   QHTUPR(ALF, REPS) = TRANSQ(REPS)
   RSHAT(ALF, REPS) = TRANSR(REPS)
   RHTSTR(ALF, REPS) = TRNSTR(REPS)
   RSHTBR(ALF, REPS) = TRANBR(REPS)
900 CONTINUE
700 CONTINUE

***// PRINT OUTPUT REPORT HEADINGS **************

WRITE(1,6666)
WRITE(1,6667) MAXREP
WRITE(1,6668) K
WRITE(1,6669)
IF(K.EQ.5) THEN
   WRITE(1,6699)
ELSE
   END IF
WRITE(1,6670) MSTRQ
WRITE(1,6671)
WRITE(1,6672)
WRITE(1,6673)
WRITE(1,6674)
WRITE(1,6675)
WRITE(1,6676)
WRITE(1,6677) MAXREP
WRITE(1,6678) K
WRITE(1,6679)
IF(K.EQ.5) THEN
   WRITE(1,6699)
ELSE
   END IF
WRITE(1,6670) MSTRQ

95
C WRITE(2,6671)
C WRITE(2,C1C15)
C WRITE(2,3334) AI
C WRITE(2,0007)
C WRITE(2,C1C15)
C WRITE(2,3334) QI
C WRITE(2,0005)
C WRITE(2,C1C15)
C WRITE(2,3335) N
C WRITE(2,6674)
C
C WRITE(2,(''SORTED RSHAT 1 IS: '',/10(F8.5))')
C +(RSHAT(1,REPS), REPS=1, MAXREP)
C WRITE(2,(''SORTED RSHAT 2 IS: '',/10(F8.5))')
C +(RSHAT(2,REPS), REPS=1, MAXREP)
C IF(FLAG.EQ.1) THEN
C WRITE(2,(''SORTED RHTSTR 1 IS: '',/10(F8.5))')
C +(RHTSTR(1,REPS), REPS=1, MAXREP)
C WRITE(2,(''SORTED RHTSTR 2 IS: '',/10(F8.5))')
C +(RHTSTR(2,REPS), REPS=1, MAXREP)
C ELSE
C END IF
C IF(K.EQ.5) THEN
C WRITE(2,(''SORTED RSHTBR 1 IS: '',/10(F8.5))')
C +(RSHTBR(1,REPS), REPS=1, MAXREP)
C WRITE(2,(''SORTED RSHTBR 2 IS: '',/10(F8.5))')
C +(RSHTBR(2,REPS), REPS=1, MAXREP)
C ELSE
C END IF
C
C ***// AND FOR THE 5-COMPONENT BRIDGE STRUCTURE
C CALL RSRS(QI,K,RS)
C WRITE(1,('' ''THE TRUE SERIES SYSTEM ''
C +''RELIABILITY VALUE IS: '',T51,F8.5)') RS
C CALL RBRIDGE(QI,K,RSBRDG)
C IF(K.EQ.5) THEN
C WRITE(1,('' ''THE TRUE BRIDGE STRUCTURE ''
C +''RELIABILITY VALUE IS: '',T51,F8.5)') RSBRDG
C ELSE
C END IF
C WRITE(1,6675)
C
C ***// RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)
C IF(FLAG.EQ.1) THEN
C WRITE(1,5755)
C ELSE
C END IF
C DO 450 ALF=1, MAXALF
C QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
C DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
C DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
C IF(FLAG.EQ.1) THEN
C DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
C END IF
C
C 96
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
WRITE(1,5657) DELSTR(ALF)
ELSE
END IF
IF(K.EQ.5) THEN
  DELBRG(ALF) = RSRDG - RSHTBR(ALF,QUANTL(ALF))
  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
  WRITE(1,5666) RSHAT(ALF,QUANTL(ALF))
  WRITE(1,5667) DELBRG(ALF)
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)
CONTINUE
PRINT*, 'QUANTL(1) IS: ', QUANTL(1)
PRINT*, 'QUANTL(2) IS: ', QUANTL(2)

***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***// RSHAT ************** //***
WRITE(1,6676)
DO 400 ALF=1,MAXALF
  TRUQNT(ALF) = 0
  DO 500 REPS=1, MAXREP
    DIFF(REPS) = RS - RSHAT(ALF,REPS)
  500 CONTINUE
  DO 600 REPS=1, MAXREP
    IF(ABS(DIFF(REPS)).LE.EPS) THEN
      TRUQNT(ALF) = REPS
      WRITE1,(':
       ',/''TRUE CONFIDENCE LIMIT IS:'t,
       + (TRUQNT(ALF) / REAL(MAXREP)) * 100.
       GO TO 620
    ELSEIF(DIFF(REPS).LT.0.) THEN
      TRUQNT(ALF) = REPS
      GO TO 610
    ELSE
      END IF
  600 CONTINUE
  IF(TRUQNT(ALF).EQ.0.) THEN
    WRITE(1,4443) ALFA(ALF)
    WRITE1,(':
       '+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
       WRITE(1,4446) RSHAT(ALF,TRUQNT(ALF))
    ELSE
      END IF
  97
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
620 END IF
400 CONTINUE

***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//  ********** RSHTBR (BRIDGE) ********** //***

IF(K.EQ.5) THEN
DO 401 ALF=1,MALF
TRUQNT(ALF) = 0
DO 501 REPS=1, MAXREP
DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
501 CONTINUE
DO 601 REPS=1, MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
TRUQNT(ALF) = REPS
WRITE(1,'(1, '' ''/''TRUE CONFIDENCE LIMIT IS:'',
+ F8.4)')
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
GO TO 621
ELSEIF(DIFF(REPS).LT.0.) THEN
TRUQNT(ALF) = REPS
GO TO 611
ELSE
END
\n601 CONTINUE
611 IF(TRUQNT(ALF).EQ.0.) THEN
WRITE(1,4443) ALFA(ALF)
WRITE(1,'(1, '' ''/''THE SMALLEST'',
+ '' DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'',
+ F10.5)') DIFF(MAXREP)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
WRITE(1,4442) ALFA(ALF)
WRITE(1,'(1, '' ''/''ALL RSHTBR'',
+ '' ARE GREATER THAN RSBRDG''))
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
+ THEN
WRITE(1,4444) ALFA(ALF),
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
621 END IF
401 CONTINUE
ELSE
END IF
**/** FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM-REL. ESTIMATE /**/**

**/** ********** RHTSTR ********** /**/**

IF(FLAG.EQ.1) THEN
DO 4400 ALF=1,MAXALF
TRUQNT(ALF) = 0
   DO 5500 REPS=1, MAXREP
      DIFF(REPS) = RS - RHTSTR(ALF,REPS)
   CONTINUE
   DO 6600 REPS=1, MAXREP
      IF(ABS(DIFF(REPS)).LE.EPS) THEN
         TRUQNT(ALF) = REPS
         WRITE(1, '(*',1, F8.4, ')') (TRUQNT(ALF)/REAL(MAXREP)) * 100.
         GO TO 6620
      ELSEIF(DIFF(REPS).LT.0.) THEN
         TRUQNT(ALF) = REPS
         GO TO 6610
      ELSE
         END IF
   CONTINUE
6600 IF(TRUQNT(ALF).EQ.0.) THEN
   WRITE(1,4443) ALFA(ALF)
   WRITE(1, '(*',1, F9.5, ')') DIFF(MAXREP)
   ELSEIF(TRUQNT(ALF).EQ.1.) THEN
   WRITE(1,4442) ALFA(ALF)
   ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF)-1))) THEN
   ELSE
   END IF
6610 IF(I.GE.(MAXREP+1)) THEN
GOTO 180
6620 END IF
4400 CONTINUE
END IF

**/** PRINT THE ARRAYS PERTINENT TO THE OUTPUT OF EACH REPLICATION *****/

IF(PRNT.EQ.1) THEN
   I = 1
   WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
   +ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
   +ALFA(SELCTB)
   IF(I.GE.(MAXREP+1)) THEN
   GOTO 180
99
ELSE
    I = I + 70
    WRITE(1,'(''I''+''+''')
    GOTO 185
  ELSE
    WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),  
    CHISQ(2,I), QHTUPR(2,I)
    END IF
    IF(I+70.LE.MAXREP) THEN
      WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),  
      + QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
    ELSE
      END IF
      I = I + 1
      GOTO 175
  END IF
  END IF
  N9999 WRITE(1,'(''THE TOTAL NO OF REPS WAS: '',I8)') TOTREP
  WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS: '',I8)') LOOP
  WRITE(1,'(''THE TOTAL NO OF NO FAILURE RUNS WAS: '',I8)') ZFAILS
  WRITE(1,'(''AVERAGE NO. OF COMPONENTS PER REPLICATION WITH '',  
  + ''NO FAILURES: '',F5.2)') ZFPREP / MAXREP
  WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS: '',I8)') FAILS
  0008 FORMAT (/ 3X,'C 1',5X,'C 2',  
  +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,  
  +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,  
  +'C 12',4X,'C 13',4X,'C 14',4X,'C 15')
  0009 FORMAT (/1X,'REP NO',2X,'DF ',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,  
  +'QHTUPR(',F4.3,')',1X,'CHISQR('}
THE RSHAT VALUE CLOSEST TO RS IS: T51,F8.5
(FIRST NEGATIVE DIFFERENCE)
(ELEMENT PRECEEDING FIRST NEGATIVE DIFFERENCE)
THE RHTSTR VALUE CLOSEST TO RS IS: T51,F8.5
THE VALUE OF RSHAT FOR THAT QUANTILE IS: T51,F8.5
THE DIFFERENCE (RS - RSHAT) IS: T51,F8.5
THE VALUE OF RHTSTR FOR THAT QUANTILE IS: T51,F8.5
THE DIFFERENCE (RS - RHTSTR) IS: T51,F8.5
THE DIFFERENCE (RS - RSHTBR) IS: T51,F8.5
SINCE THE NO. OF MISSION TESTS IS THE SAME FOR
ALL COMPONENTS THE CLOSED FORM SERIES SYSTEM RELIABILITY
"RHTSTR" IS COMPUTED
RUN INPUT SETTINGS
NUMBER OF REPLICATIONS: T50,I4
NUMBER OF COMPONENTS: T50,I4
SYSTEM RELIABILITY FUNCTION: T50,'SERIES'
SYSTEM RELIABILITY FUNCTION: T50,'BRIDGE'
MASTER UNRELIABILITY USED: T50,F8.5
INPUT WEIGHTS (A SUB I'S):'
RUN RESULTS
************** RUN INPUT SETTINGS **************
************** MASTER UNRESULT **************
ESTIMATE ERRORS *************
TRUE CONFIDENCE LIMITS *******
END
APPENDIX H. FORTRAN CODE FOR ALTERNATE PROCEDURE B
FOR ESTIMATING THE LOWER CONFIDENCE LIMIT FOR SELECTED
SYSTEMS

PROGRAM ZFYSCN

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

* TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE
* ZERO FAILURES ALLOWED; NO SCALING
* AUTHOR: E. F. BELLINI, LT, USN
* MODIFIED BY: LT. VALERIE A. COVINGTON, USN (MAR 90)
* DATE: NOV 89
* THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE
* RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY
* OF THEIR COMPONENTS

* IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12
* TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT
* SET OF INPUT DATA. RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.
* THE REXX EXEC PROGRAM
* 'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT
* FILES RESULTING FROM THE 12 CONSECUTIVE RUNS. BY EDITING THE
* INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-
* SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.

* VARIABLES USED

* AHATI : WEIGHT ESTIMATES FOR EACH COMPONENT
* AI : INPUT WEIGHTS FOR EACH COMPONENT
* ALFA : LEVELS OF SIGNIFICANCE
* BIGF : TOTAL NO. OF FAILURES FOR EACH REPLICATION
* CHISQ : CHI-SQUARE RANDOM VARIABLE VALUE
* C1C15 : FORMAT LABEL
* DEGFR : DEGREES OF FREEDOM
* DELBRG : DIFFERENCE FOR BRIDGE SYSTEM
* DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM
* DELTAR : DIFFERENCE FOR SERIES SYSTEM
* DIFF : DIFFERENCE (TRUE REL. - ESTIMATED REL.)
* EPS : SMALL QUANTITY(CONSTANT)
* ERROR : PARAMETER FOR IMSL ROUTINE
* FAILS : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE
* FI : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)
* FLAG : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS
* INC : INCREMENT STEP SIZE FOR ROUTINE USMNMX
* KEY1 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY2 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY3 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KEY4 : ARRAY OF INDECES FOR ROUTINE SHSORT
* KK : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS
* LOOP : COUNTS NO. OF REPLICATION PERFORMED
* MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXREP</td>
<td>MAX NO. OF REPLICATIONS</td>
</tr>
<tr>
<td>MAXRUN</td>
<td>MAX NO. OF PROGRAM ITERATIONS ALLOWED</td>
</tr>
<tr>
<td>MSTRQ</td>
<td>MASTER UNRELIABILITY (USED WITH AI'S TO CALC. QI'S)</td>
</tr>
<tr>
<td>MULT</td>
<td>MULTIPLIER FOR RANDOM NO. GENERATOR SRND</td>
</tr>
<tr>
<td>N</td>
<td>NO. OF MISSION TEST FOR EACH COMPONENT</td>
</tr>
<tr>
<td>NIMAX</td>
<td>MAX NO. OF MISSION TESTS</td>
</tr>
<tr>
<td>NIMIN</td>
<td>MIN NO. OF MISSION TESTS</td>
</tr>
<tr>
<td>NINDX</td>
<td>INDEX NO. OF MAX NO. OF MISSION TESTS</td>
</tr>
<tr>
<td>NIREAL</td>
<td>NO. OF MISSION TESTS TRANSFORMED TO REAL</td>
</tr>
<tr>
<td>NMAX</td>
<td>MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL</td>
</tr>
<tr>
<td>NPRNT</td>
<td>FLAG FOR DETAILED REPORT OUTPUT</td>
</tr>
<tr>
<td>PRNT</td>
<td>SAME AS ABOVE (PARAMETER)</td>
</tr>
<tr>
<td>QHATI</td>
<td>UNRELIABILITY ESTIMATES FOR EACH COMPONENT</td>
</tr>
<tr>
<td>QHTMAX</td>
<td>LARGEST QHATI</td>
</tr>
<tr>
<td>QHTUPR</td>
<td>UPPER LIMIT ON SYSTEM UNRELIABILITY</td>
</tr>
<tr>
<td>QI</td>
<td>INPUT UNRELIABILITY FOR EACH COMPONENT</td>
</tr>
<tr>
<td>QINDX</td>
<td>INDEX</td>
</tr>
<tr>
<td>QUANTL</td>
<td>QUANTILE</td>
</tr>
<tr>
<td>REPSHD</td>
<td>REPLICATIONS HEADING FORMAT NUMBER</td>
</tr>
<tr>
<td>RHTSTR</td>
<td>SERIES SYSTEM RELIABILITY ESTIMATE (CLOSED FORM)</td>
</tr>
<tr>
<td>RS</td>
<td>TRUE SERIES SYSTEM RELIABILITY</td>
</tr>
<tr>
<td>RSRDG</td>
<td>TRUE BRIDGE SYSTEM RELIABILITY</td>
</tr>
<tr>
<td>RSHORT</td>
<td>SERIES SYSTEM RELIABILITY ESTIMATE</td>
</tr>
<tr>
<td>RSHTBR</td>
<td>BRIDGE SYSTEM RELIABILITY ESTIMATE</td>
</tr>
<tr>
<td>SEED</td>
<td>PARAMETER</td>
</tr>
<tr>
<td>SELCTA</td>
<td>SIGNIFICANCE LEVEL SELECTION</td>
</tr>
<tr>
<td>SELCTB</td>
<td>SIGNIFICANCE LEVEL SELECTION</td>
</tr>
<tr>
<td>SORT</td>
<td>PARAMETER FOR ROUTINE SRND</td>
</tr>
<tr>
<td>SUNNAI</td>
<td>SUM OF THE PRODUCT OF NI'S AND AI'S</td>
</tr>
<tr>
<td>TEMP</td>
<td>TEMPORARY ARRAY</td>
</tr>
<tr>
<td>TOTREP</td>
<td>TOTAL NUMBER OF PROGRAM ITERATIONS</td>
</tr>
<tr>
<td>TRANBR</td>
<td>TEMPORARY ARRAY</td>
</tr>
<tr>
<td>TRANSQ</td>
<td>TEMPORARY ARRAY</td>
</tr>
<tr>
<td>TRANSR</td>
<td>TEMPORARY ARRAY</td>
</tr>
<tr>
<td>TRIALS</td>
<td>BERNOULLI TRIALS ARRAY (2-DIM)</td>
</tr>
<tr>
<td>TRNSTR</td>
<td>TEMPORARY ARRAY</td>
</tr>
<tr>
<td>TRUQNT</td>
<td>TRUE QUANTILE</td>
</tr>
<tr>
<td>UNIRV</td>
<td>UNIFORM RANDOM DEVIATES (2-DIM)</td>
</tr>
<tr>
<td>ZFAILS</td>
<td>TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES</td>
</tr>
<tr>
<td>ZFPREP</td>
<td>NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION</td>
</tr>
</tbody>
</table>

PARAMETER (KK=10, MAXALF=2, NPRNT=0)
PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=.000001)
REAL*4 UNIRV(15,1000), TEMP(1000), QI(KK), AI(KK), AHATI(KK)
REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
REAL*4 DF(5), AALFA(5), SUNNAI, RSHAT(MAXALF, MAXREP), RS
REAL*4 KEY1(MAXREP), KEY2(MAXREP), KEY3(MAXREP), TRNSTR(MAXREP)
REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF, MAXREP), CHISQ(MAXALF, MAXREP)
REAL*4 QUPA1(MAXREP), QUPA2(MAXREP), RHTSTR(MAXALF, MAXREP)
REAL*4 DELTAR(MAXALF), TRANQ(MAXREP), TRANR(MAXREP), Diff(MAXREP)
REAL*4 DELSTR(MAXALF), NMIN, NMAX, NIREAL(KK)
REAL*4 RSHORT(MAXALF, MAXREP), DELBREG(MAXALF, KEY4(MAXREP))
REAL*4 TRANBR(MAXREP), RSRDG , MSTRQ
REAL*4 2ZFPREP
REAL*4 RSHA(TI(KK),SI(KK),QHATU(KK)
REAL*4 MXQHAT,RI(KK),SUMRN
REAL*4 QHATMU(MAXALF),CHIVAL(MAXALF),P(KK),PTEMP(10)

INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
INTEGER NINDX, QINDEX, ERROR, REPS, SELCTA, SELCTK, TOTREP
INTEGER CIC15, REPSHD, SELCTB, ALF, FLAG, LOOP, PRNT, HFI
INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS, INC

CHARACTER*8 LOOPSO(MAXREP)

DATA SEED/123457/, MULT/1/, INC/1/
DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
DATA ALFA/.-20,.050/

ASSIGN 8 TO CIC15
ASSIGN 9 TO REPSHD

CALL COMPRS
PRNT = NPRNT

DO 12 I=1,KK
   AI(I) = 9999.
   N(I) = 99999999
12 CONTINUE

READ(03,*)K,MSTRQ

DO 11 I=1,KK
   READ(03,*) AI(I),N(I)
11 CONTINUE

IF(K.NE.5) THEN
   WRITE(1,'(''WARNING: BRIDGE STRUCTURE ''
   +''ONLY USES THE FIRST 5 COMPONENTS'')')
ELSE
END IF

***INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS,
***RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS
*** AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN
***ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER

DO 172 ALF=1,MAXALF
   DO 173 REPS=1,MAXREP
      QHTUPR(ALF,REPS) = 0.
      RSHAT(ALF,REPS) = 0.
      RHTSTR(ALF,REPS) = 0.
      RSHTBR(ALF,REPS) = 0.
      LOOPSO(REPS) = '******'
173 CONTINUE
172 CONTINUE

***SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS***
FLAG=1
DO 50 I=1,K-1
   IF((N(I)-N(I+1)).NE.0) THEN
      FLAG=0
   ELSE
   END IF
50 CONTINUE
PRINT *, 'FLAG IS:', FLAG

***MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)***

ZFPREP = 0.
ZFFAILS = 0
FAILS = 0
TOTREP = 0
LOOP = 0
10 IF(LOOP.LT.MAXREP) THEN
   LOOP = LOOP + 1
   IF(TOTREP.LT.MAXRUN) THEN
      TOTREP = TOTREP + 1
   END IF
   SELCTA = 1
   SELCTB = 2

***FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT***
***OF THE SUBROUTINE SHSORT***

DO 95 REPS=1, MAXREP
   KEY1(REPS) = REPS
   KEY2(REPS) = REPS
   KEY3(REPS) = REPS
   KEY4(REPS) = REPS
95 CONTINUE

***CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE***

***CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.***

   CALL IMAX(N,K,NMAX,NINDX)

***CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S***

   DO 115 I=1, K
      QI(I) = MSTRQ * AI(I)
115 CONTINUE

   DO 120 I=1,15
      DO 125 J=1,500
         UNIRV(I,J) = 999.
         TRIALS(I,J) = 99999
125 CONTINUE
120 CONTINUE

*** DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOUlli TRIALS***
DO 130 \texttt{I}=1, K
    CALL \texttt{SRND(SEED, TEMP, N(I), MULT, SORT)}
    DO 135 \texttt{J}=1, N(I)
        \texttt{UNIRV(I,J)} = \texttt{TEMP(J)}
        IF (\texttt{UNIRV(I,J)} \leq 1 - \texttt{QI(I)}) THEN
            \texttt{TRIALS(I,J)} = 0
        ELSE
            \texttt{TRIALS(I,J)} = 1
        END IF
    \end{verbatim}
135 \texttt{CONTINUE}
130 \texttt{CONTINUE}

\*\*\*CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT\*\*\*

\begin{verbatim}
DO 150 \texttt{I}=1, K
    \texttt{FI(I)} = 0
150 \texttt{CONTINUE}

IONECT = 0

\*\*\*CALCULATE THE F'S AND THE GRAND TOTAL NO. OF FAILURES**

BIGF = 0

\begin{verbatim}
DO 155 \texttt{I}=1, K
    \texttt{DO 160 J}=1, N(I)
        \texttt{FI(I)} = \texttt{FI(I)} + \texttt{TRIALS(I,J)}
    \end{verbatim}
160 \texttt{CONTINUE}

IF (\texttt{FI(I)} \texttt{.EQ. 0.}) THEN
    \texttt{ZFPPREP} = \texttt{ZFPPREP} + 1
ELSE
    END IF
\end{verbatim}

\*\*\*CALCULATE THE QHAT'S: F'S DIVIDED BY N'S**

\begin{verbatim}
QHATI(I) = REAL(\texttt{FI(I)}) / N(I)
BIGF = BIGF + \texttt{FI(I)}
155 \texttt{CONTINUE}

IF (\texttt{FI(2)} \texttt{.EQ. 0.}) THEN
    \texttt{ZFPPREP} = \texttt{ZFPPREP} - 1
ELSE
    ENDIF
    \texttt{BIGF} = \texttt{BIGF} - \texttt{FI(2)}
    \texttt{FI(2)} = 0
    \texttt{DO 161 I}=2, N(2)
        \texttt{HFI}=0
    \end{verbatim}
163 \texttt{CONTINUE}

CALL \texttt{SRND(SEED, PTEMP(J), N(2), MULT, SORT)}
    IF (PTEMP(J) \texttt{.GT. 1-QI(2)**.5}) THEN
        \texttt{HFI} = HFI + 1
    END IF
161 \texttt{CONTINUE}

\begin{verbatim}
IF (\texttt{HFI} \texttt{.EQ. 2}) THEN
    \texttt{FI(2)} = \texttt{FI(2)} + 1
ENDIF
163 \texttt{CONTINUE}

QHATI(2) = REAL(\texttt{FI(2)}) / N(2)
BIGF = BIGF + \texttt{FI(2)}
IF (\texttt{FI(2)} \texttt{.EQ. 0.}) THEN
ZFPREP = ZFPREP + 1
ENDIF

***COUNTS NUMBER OF COMPONENTS THAT HAVE FAILED***
DO 136 I=1,K
   IF (FI(I) .NE. 0) IONECT=IONEC+1.
136 CONTINUE

***CASE WHERE NO COMPONENTS HAVE ANY FAILURES***
IF(BIGF.EQ.0) THEN
   LOOPSO(LOOP)=' *ZERO* '
   ***SERIES ESTIMATE MODIFICATION (NO. OF FAILURES IRRELEVANT)***
   DO 152 I=1,K
      SI(I) = REAL(N(I))-REAL(FI(I))
      RSHATI(I)=SI(I)/(SI(I)+(REAL(FI(I))+1.)
      + * FIN(.50,2.*(REAL(FI(I))+1.),2*SI(I)))
      QHATI(U(I))=1.-RSHATI(I)
152 CONTINUE
   MXQHAT=0.
   DO 154 I=1,K
      IF (QHATIU(I).GT. MXQHAT) THEN
         MXQHAT= QHATIU(I)
      ENDIF
154 CONTINUE
   DO 156 ALF=1,MAXALF
      CALL MDCXI(ALFA(ALF),2*(1.+REAL(BIGF)),CHIVAL(ALF),ERROR)
      SUMRN=0.
      DO 157 I=1,K
         RI(I)=QHATIU(I)/MXQHAT
         SUMRN=SUMRN+RI(I)*REAL(N(I))
157 CONTINUE
      QHATMU(ALF)=CHIVAL(ALF)/(2*SUMRN)
156 CONTINUE
   DO 158 ALF=1,MAXALF
      RSHAT(ALF,LOOP)=1.
      DO 159 I=1,K
         RSHAT(ALF,LOOP)=RSHAT(ALF,LOOP)*(1.-(RI(I)*QHATMU(ALF)))
         IF(FLAG.EQ.1) THEN
            RHTSTR(ALF,LOOP) = RSHAT(ALF,LOOP)
         ELSE
            END IF
159 CONTINUE
158 CONTINUE

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***
   DO 141 ALF=1, MAXALF
   DO 142 I=1,K

P(I) = 1 - RI(I) * QHATMU(ALF)

CONTINUE

RSHTBR(ALF, LOOP) = P(1) * P(4) + P(2) * P(5) + P(1) * P(3) *
C
P(5) * P(2) * P(3) * P(4) - P(1) * P(2) * P(3) * P(4) - P(1) *
C
C
C
P(3) * P(4) * P(5)

CONTINUE

ZFAILS = ZFAILS + 1
<br>DEGFR(LOOP) = 2.
<br>GO TO 10.
<br>
ELS:
<br>FAILS = FAILS + 1
<br>END IF

*** FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES ***
<br>CALL RMAX(QHATI, K, QHTMAX, QINDX)

*** CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES) ***
<br>SUMNAI = 0.
<br>DO 165 I = 1, K
<br>AHATI(I) = QHATI(I) / QHTMAX
<br>SUMNAI = SUMNAI + N(I) * AHATI(I)
<br>165 CONTINUE

*** 1 FAILURE ONLY SERIES SYST. ***
<br>IF (IONECF .EQ. 1) THEN
<br>LOOPSO(LOOP) = 1 * ONE *
<br>ENDIF

*** CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY ***
<br>DEGFR(LOOP) = 2 * (1 + BIGF)

*** SERIES ESTIMATE MODIFICATION (NO. OF FAILURES IRRELEVANT) ***
<br>DO 162 I = 1, K
<br>S(I) = REAL(N(I)) - REAL(QHATI(I))
<br>RSHATI(I) = S(I) / (S(I) + REAL(FI(I)) + 1.)
<br>MXQHAT = 0.
<br>DO 164 I = 1, K
<br>IF (QHATIU(I) .GT. MXQHAT) THEN
<br>MXQHAT = QHATIU(I)
<br>ENDIF
<br>164 CONTINUE

108
DO 166 ALF=1, MAXALF
   CALL MDCHI(ALFA(ALF), 2*(1.+REAL(BIGF)), CHIVAL(ALF), ERROR)
   SUMRN=0.
   DO 167 I=1, K
      RI(I)=QHATU(I)/NXQHAT
      SUMRN=SUMRN+RI(I)*REAL(N(I))
   CONTINUE
167 CONTINUE
   QHATMU(ALF)=CHIVAL(ALF)/(2.*SUMRN)
166 CONTINUE

DO 168 ALF=1, MAXALF
   RSHAT(ALF, LOOP)=1.
   DO 169 I=1, K
      RSHAT(ALF, LOOP)=RSHAT(ALF, LOOP)*(1.-RI(I)*QHATMU(ALF))
      IF(FLAG.EQ.1) THEN
         RHTSTR(ALF, LOOP) = RSHAT(ALF, LOOP)
      ELSE
      END IF
   CONTINUE
169 CONTINUE
168 CONTINUE

***CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE***

***THIS ELSE AND ENDFi are for the Test AGAINST MAXRUN***

ELSE
   WRITE(1,'(''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',+''ALLOWED OF: '',I6)'') TOTREP
   GOTO 9999
END IF
GOTO 10
END IF

WRITE(2,'(''UNSORTED RSHAT 1 IS: '',/10(F8.5))')
+(RSHAT(1, LOOP), LOOP=1, MAXREP)
WRITE(2,'(''UNSORTED RSHAT 2 IS: '',/10(F8.5))')
+(RSHAT(2, LOOP), LOOP=1, MAXREP)
1F(FLAG.EQ.1) THEN
   WRITE(2,'(''UNSORTED RHTSTR 1 IS: '',/10(F8.5))')
+(RHTSTR(1, LOOP), LOOP=1, MAXREP)
   WRITE(2,'(''UNSORTED RHTSTR 2 IS: '',/10(F8.5))')
+(RHTSTR(2, LOOP), LOOP=1, MAXREP)
109
ELSE
END IF
IF(K.EQ.5) THEN
WRITE(2, ('''UNSORTED RSHTBR 1 IS: ''/10(F8.5))''
+(RSHTBR(1,LOOP), LOOP=1, MAXREP)
WRITE(2, ('''UNSORTED RSHTBR 2 IS: ''/10(F8.5))''
+(RSHTBR(2,LOOP), LOOP=1, MAXREP)
ELSE
END IF
WRITE (2, ('''ZERO AND ONE FAILURE REPS: '',/10(A8))''
+ (LOOPSO(LOOP), LOOP=1, MAXREP)

***SORT THE ARRAYS OF SYSTEM UNRELIABILITIES (1 FOR EACH CONF. LEVEL)***

DO 700 ALF=1, MAXALF
DO 800 REPS=1, MAXREP
    TRANSQ(REPS) = QHTUPR(ALF,REPS)
    TRANSR(REPS) = RSHAT(ALF,REPS)
    TRNSTR(REPS) = RHTSTR(ALF,REPS)
    TRANBR(REPS) = RSHTBR(ALF,REPS)
800 CONTINUE
    CALL SHSORT(TRANSQ,KEY1,MAXREP)
    CALL SHSORT(TRANSR,KEY2,MAXREP)
    CALL SHSORT(TRNSTR,KEY3,MAXREP)
    CALL SHSORT(TRANBR,KEY4,MAXREP)
DO 900 REPS=1, MAXREP
    QHTUPR(ALF,REPS) = TRANSQ(REPS)
    RSHAT(ALF,REPS) = TRANSR(REPS)
    RHTSTR(ALF,REPS) = TRNSTR(REPS)
    RSHTBR(ALF,REPS) = TRANBR(REPS)
900 CONTINUE
700 CONTINUE

***PRINT OUTPUT REPORT HEADINGS***

WRITE(1,6666)
WRITE(1,6667) MAXREP
WRITE(1,6668) K
WRITE(1,6669)
IF(K.EQ.5) THEN
    WRITE(1,6699)
ELSE
END IF
WRITE(1,6670) MSTRQ
WRITE(1,6671)
WRITE(1,3334) AI
WRITE(1,0007) QI
WRITE(1,0005) N
WRITE(1,3335)
110
***COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN. (SERIES SYSTEM)***
*** AND FOR THE 5-COMPONENT BRIDGE STRUCTURE***

CALL RSRS(Q1,K,RS)
WRITE(1,'(''THE TRUE SERIES SYSTEM '', ''RELIABILITY VALUE IS: '',T51,F8.5)') RS
CALL RBRDG(Q1,K,RSBRDG)

IF(K.EQ.5) THEN
WRITE(1,'(''THE TRUE BRIDGE STRUCTURE '', ''RELIABILITY VALUE IS: '',T51,F8.5)') RSBRDG
ELSE
END IF
WRITE(1,6675)

***COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO***
***RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)***

IF(FLAG.EQ.1) THEN
WRITE(1,5755)
ELSE
END IF
DO 450 ALF=1, MAXALF
QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))

IF(FLAG.EQ.1) THEN
DELSR(ALF) = RS - RHTSR(ALF,QUANTL(ALF))
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5656) RHTSR(ALF,QUANTL(ALF))
WRITE(1,5657) DELSR(ALF)
ELSE
END IF
ELSE
END IF
WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
WRITE(1,5557) DELTAR(ALF)

.450 CONTINUE
PRINT *, 'QUANTL(1) IS: ', QUANTL(1)
PRINT *, 'QUANTL(2) IS: ', QUANTL(2)

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
***
WRITE(1,6676)
DO 400 ALF=1,MAXALF
TRUQNT(ALF) = 0
DO 500 REPS=1, MAXREP
DIFF(REPS) = RS - RSHAT(ALF,REPS)

500 CONTINUE
DO 600 REPS=1, MAXREP
IF(ABS(DIFF(REPS)).LE.EPS) THEN
  TRUQNT(ALF) = REPS
  WRITE(1,'(''''''''TRUE CONFIDENCE LIMIT IS:'''','
   $F8.4')
  + (TRUQNT(ALF) / REAL(MAXREP)) * 100.
  GO TO 620
ELSEIF(DIFF(REPS).LT.0.) THEN
  TRUQNT(ALF) = REPS
  GO TO 610
ELSE
  CONTINUE
ENDIF
600
610 IF(TRUQNT(ALF).EQ.0.) THEN
  WRITE(1,4443) ALFACALF)
  WRITE(1,'(''THE SMALLEST''
   'DIFFERENCE BETWEEN RS AND RSHAT IS:'''',F10.5)' ) DIFF(MAXB)
ELSEIF(TRUQNT(ALF).EQ.1.) THEN
  WRITE(1,4442) ALFACALF)
  WRITE(1,'(''''''''ALL RSHAT''
   'ARE GREATER THAN RS''')')
ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1))) THEN
  WRITE(1,4444)
  + ((TRUQNT(ALF) - 1) / REAL(MAXREP)) * 100.
  WRITE(1,4445)
  RSHAT(ALF,TRUQNT(ALF) - 1)
ELSE
  WRITE(1,4444)
  + ((TRUQNT(ALF) - 1) / REAL(MAXREP)) * 100.
  WRITE(1,4445)
  RSHAT(ALF,TRUQNT(ALF))
ENDIF
620 END IF
400 CONTINUE

***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
*** ********** RSHTBR (BRIDGE) ********** ***

IF(K.EQ.5) THEN
  DO 401 ALF=1,MAXALF
    TRUQNT(ALF) = 0
  DO 501 REPS=1, MAXREP
    DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
  CONTINUE
  DO 601 REPS=1, MAXREP
    IF(ABS(DIFF(REPS)).LE.EPS) THEN
      TRUQNT(ALF) = REPS
      WRITE(1,'(''TRUE CONFIDENCE LIMIT IS:'''','
       $F8.4')
      + (TRUQNT(ALF) / REAL(MAXREP)) * 100.
      GO TO 621
    ELSEIF(DIFF(REPS).LT.0.) THEN
      TRUQNT(ALF) = REPS
      GO TO 611
    ELSE

112
IF(T;RuQNTr(ALF).-EQ.0.) THEN
  WRITE( 1,4443)
  ALFA(ALF)
  WRITE(1,'("","","THE SMALLEST",
  " DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:","",
  F10.5)') DIFF(MAXREP)
ELSEIF(TUQNT(ALF). EQ. 1.) THEN
  WRITE(1,4442) ALFA(ALF)
  WRITE(1,'("","","ALL RSHTBR")
  ELSEIF(ABS(DIFF(TUQNT(ALF))). LE. ABS(DIFF(TUQNT(ALF) - 1))))
  THEN
  WRITE(1,4444) ALFA(ALF),
  (TUQNT(ALF) / REAL(MAXREP)) * 100.
  WRITE(1,4449) RSHTBR(ALF,TUQNT(ALF))
ELSE
  WRITE(1,4444) ALFA(ALF),
  ((TUQNT(ALF)-1) / REAL(MAXREP)) * 100.
  WRITE(1,4447)
END IF
CONTINUE
ELSE
" ***FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE***
" ********** RHTSTR **********
" IF(FLAG.EQ.1) THEN
DO 4400 ALF=1,MAXALF
  TUQNT(ALF) = 0
  DO 5500 REPS=1, MAXREP
    DIFF(REPS) = RS - RHTSTR(ALF,REPS)
  5500 CONTINUE
  DO 6600 REPS=1, MAXREP
    IF(ABS(DIFF(REPS)). LE. EPS) THEN
      TUQNT(ALF) = REPS
      WRITE(1,'("","","TRUE CONFIDENCE LIMIT IS:","",
      F8.4)')
      (TUQNT(ALF) / REAL(MAXREP)) * 100.
    END IF
  6600 CONTINUE
ELSE
  IF(TUQNT(ALF). EQ. 0.) THEN
    WRITE(1,4443) ALFA(ALF)
    WRITE(1,'("","","THE SMALLEST",
    " DIFFERENCE BETWEEN RS AND RHTSTR IS:","",
    F9.5)') DIFF(MAXREP)
  END IF
END IF
WRITE(1,4442) ALFA(ALF),
WRITE(1,'(''''.,''''ALL RHTSTR''
, 
'' ARE GREATER THAN RS''')
ELSEIF(ABS(DIFF(TRUQNT(ALF))) LE. ABS(DIFF(TRUQNT(ALF) - 1)))
+ THEN
WRITE(1,4444) ALFA(ALF),
+ (TRUQNT(ALF) / REAL(MAXREP)) * 100.
WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
WRITE(1,4446)
ELSE
WRITE(1,4444) ALFA(ALF),
+ ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
WRITE(1,4447)
6620 END IF
4400 CONTINUE
ELSE
ENDIF

***PRINT THE ARRAYS PERTINENT TO THE OUTPUT OF EACH REPLICATION***

IF(PRNT.EQ.1) THEN
I = 1
185 WRITE(1,REPShD) ALFA(SELCTA), ALFA(SELCTA),
+ ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
+ALFA(SELCTB)
175 IF(I.GE.(MAXREP + 1)) THEN
GOTO 180
ELSE
I = I + 70
WRITE(1,'(''++''
GOTO 185
ELSE
WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
+ CHISQ(2,I), QHTUPR(2,I)
END IF
IF(I + 70.LE.MAXREP) THEN
WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),
+ QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
ELSE
180 END IF
ELSE
185 ENDIF
9999 WRITE(1,'(''THE TOTAL NO OF REPS WAS:'',I8)') TOTREP
WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS:'',I8)') LOOP
WRITE(1,'(''THE TOTAL NO OF NO FAILURE RUNS WAS:'',I8)') ZFAILS
WRITE(1,'(''AVERAGE NO. OF COMPONENTS PER REPLICATION WITH ''
+''NO FAILURES:'',F5.2)') ZFPREP / MAXREP
WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS:'',I8)') FAILS
0008 FORMAT (/ 3X,'C 1',5X,'C 2',

114
ESTIMATE ERRORS

ESTIMATE ERRORS

6676 FORMAT ('+1', 'TRUE CONFIDENCE LIMITS')

END
APPENDIX I. SUBROUTINES

IMSL ROUTINE NAME - USMNMX

COMPUTER - IBM/SINGLE

LATEST REVISION - JANUARY 1, 1978

PURPOSE - DETERMINATION OF THE MINIMUM AND MAXIMUM VALUES OF A VECTOR

USAGE - CALL USMNMX (X,N,INC,XMIN,XMAX)

ARGUMENTS X - INPUT VECTOR OF LENGTH N FROM WHICH MINIMUM, MAXIMUM VALUES ARE TO BE TAKEN.

N - LENGTH OF THE INPUT VECTOR X. (INPUT)

INC - DISPLACEMENT BETWEEN CONSECUTIVE VALUES OF X TO BE CONSIDERED.

XMIN - OUTPUT SCALAR CONTAINING MINIMUM VALUE OF X.

XMAX - OUTPUT SCALAR CONTAINING MAXIMUM VALUE OF X.

PRECISION/HARDWARE - SINGLE AND DOUBLE/H32

- SINGLE/H36,H48,H60

REQD. IMSL ROUTINES - NONE REQUIRED

NOTATION - INFORMATION ON SPECIAL NOTATION AND CONVENTIONS IS AVAILABLE IN THE MANUAL INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

COPYRIGHT - 1978 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 USMNMX (X,N,INC,XMIN,XMAX)

DIMENSION X(N)

FIRST EXECUTABLE STATEMENT

XMIN = X(1)
XMAX = X(1)
DO 10 I=1,N,INC
  IF (X(I) .GE. XMIN) GO TO 5
  XMIN = X(I)
GO TO 10
5  IF (X(I) .GT. XMAX) XMAX = X(I)
10  CONTINUE
RETURN
A. IDENTIFICATION:
TITLE: NUMERICAL SORT
ID: M1-NPG-SHSORT (F-IV)
PROGRAMMER: R. BRUNELL
DATE: MARCH 1968
MODIFIED: DEC. 1973 BY L. NOLAN

B. PURPOSE:
TO SORT, IN ASCENDING ORDER, AN ARRAY OF SINGLE PRECISION REAL NUMBERS BY THE METHOD OF SHELL, AND TO PRODUCE AN ARRAY OF INDEXES SO USER CAN RE-ORDER OTHER CORRESPONDING INFORMATION ACCORDING TO ASCENDING VALUES OF "A".

C. USAGE:
1. CALLING STATEMENT:
   CALL SHSORT(A,KEY,N)
2. ARGUMENTS:
   A - ARRAY OF NUMBERS TO BE SORTED. THIS ARRAY IS SORTED (RE-ORDERED) BY "SHSORT".
   KEY - ARRAY, DIMENSIONED AT LEAST N IN CALLING PROGRAM, TO BE FILLED BY USER WITH INTEGERS FROM 1 TO N. AFTER EXIT FROM SHSORT, KEY(1) WILL CONTAIN THE ORIGINAL INDEX OF THE SMALLEST ELEMENT OF "A"; KEY(2) WILL CONTAIN THE ORIGIN INDEX OF THE NEXT-TO-SMALLEST ELEMENT OF "A"; ETC. KEY(N) WILL CONTAIN THE ORIGINAL INDEX OF THE LARGEST ELEMENT OF "A".
   N - NUMBER OF MEMBERS IN ARRAYS "A" AND "KEY".

D. REFERENCES:

SUBROUTINE SHSORT(A,KEY,N)
DIMENSION A(N),KEY(N)
M1=1
6 M1=M1*2
IF (M1 .LE. N) GO TO 6
M1=M1/2-1
MM=MAXO(M1/2, 1)
GO TO 21
20 MM=MM/2
IF (MM .LE. 0) GO TO 100
21 K=N-MM
22 DO 1 J=1,K
   II=J
11 IM=II+MM
   IF (A(IM) .GE. A(II)) GO TO 1
   TEMP=A(II)
   IT=KEY(II)
   KEY(II)=KEY(IM)
   KEY(IM)=IT
   TEMP=A(IM)
   GO TO 21
A(I) = A(IM)

KEY(I) = KEY(IM)

A(IM) = TEMP

KEY(IM) = IT

II = II - MM

IF (II .GT. 0) GO TO 11

1 CONTINUE

GO TO 20

100 RETURN

END

SUBROUTINE RHTSRS(QHTUP, AAHTI, N, RRSHAT)

***/// THIS ROUTINE CALCULATES THE VALUE OF THE SYSTEM RELIABILITY OF A SERIES SYSTEM OF 'N' NO. OF COMPONENTS WHICH HAVE UNRELIABILITY 'QHTUP'. THE FINAL SYSTEM RELIABILITY VALUE PASSED IS 'RRSHAT'

REAL*4 QHTUP, RRSHAT, AAHTI(N)

INTEGER I, N

RRSHAT = 1.

DO 100 I = 1, N

RRSHAT = RRSHAT * (1 - (AAHTI(I) * QHTUP))

100 CONTINUE

END

SUBROUTINE RSRS(QIS, N, RRS)

***/// THIS ROUTINE CALCULATES THE VALUE OF THE SYSTEM RELIABILITY OF A SERIES SYSTEM OF 'COMP' NO. OF COMPONENTS WHICH HAVE UNRELIABILITY 'QIS'. THE FINAL SYSTEM RELIABILITY VALUE PASSED IS 'RRS'

REAL*4 QIS(N), RRS

INTEGER I, N

RRS = 1.

DO 100 I = 1, N

RRS = RRS * (1 - QIS(I))

100 CONTINUE

END

***/// THIS SUBROUTINE CALCULATES THE ESTIMATED RELIABILITY OF A 5-COMPONENT BRIDGE STRUCTURE. (ONLY CARRIED OUT TO THE Q-CUBED TERM

SUBROUTINE RHTBRG(QHTUP, AHT, N, RRBRDG)

REAL*4 QHTUP, RRBRDG, AHT(N)

INTEGER N

* PRINT *, 'THE VALUES FOR AHT PASSED ARE:', AHT

RRBRDG = 1 - ((QHTUP**2)*(AHT(1)*AHT(2) + AHT(3)*AHT(4)*AHT(5))) +

C((QHTUP**3)*(AHT(1)*AHT(3)*AHT(5) + AHT(2)*AHT(3)*AHT(4)))+

C((QHTUP**4)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))+

C((QHTUP**5)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))+

C((QHTUP**6)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))+

C((QHTUP**7)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))+

C((QHTUP**8)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))+

C((QHTUP**9)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5))))

* PRINT *, 'COMPUTED RRBRDG IS:', RRBRDG

END

SUBROUTINE CPARE(FI, K, BFLAG)
INTEGER BFLAG, FI(K)
BFLAG=0
 IF ((FI(1) .EQ. 0) .AND. (FI(3) .EQ. 0) .AND. (FI(5) .EQ. 0)) THEN
   BFLAG=0
 ELSE IF ((FI(1) .EQ. 0) .AND. (FI(4) .EQ. 0)) THEN
   BFLAG = 0
 ELSE IF ((FI(2) .EQ. 0) .AND. (FI(5) .EQ. 0)) THEN
   BFLAG = 0
 ELSE IF ((FI(2) .EQ. 0) .AND. (FI(3) .EQ. 0) .AND. (FI(4) .EQ. 0)) THEN
   BFLAG = 0
 ELSE IF (((P1(1) .EQ. 1) .AND. (P1(2) .EQ. 1)) .AND. (FI(2) .EQ. 0) .AND. (FI(4) .EQ. 0) .AND. (FI(5) .EQ. 0)) THEN
   BFLAG = 0
 ELSE IF (((FI(1) .EQ. 1) .AND. (FI(2) .EQ. 1)) .AND. (FI(3) .EQ. 1) .AND. (FI(4) .EQ. 1) .AND. (FI(5) .EQ. 1)) THEN
   BFLAG = 1
 ELSE IF (((FI(2) .EQ. 0) .AND. (FI(4) .EQ. 0)) .AND. (FI(5) .EQ. 0)) THEN
   BFLAG = 1
 ELSE IF (((FI(1) .EQ. 0) .AND. (FI(5) .EQ. 0)) .AND. (FI(3) .EQ. 1) .AND. (FI(4) .EQ. 1)) THEN
   BFLAG = 1
 ELSE IF (((FI(1) .EQ. 1) .AND. (FI(2) .EQ. 1)) .AND. (FI(3) .EQ. 1) .AND. (FI(4) .EQ. 1)) THEN
   BFLAG = 1
 ELSE IF (((FI(1) .EQ. 0) .AND. (FI(5) .EQ. 0))) THEN
   BFLAG = 2
ENDIF
RETURN
END

***// THIS SUBROUTINE CALCULATES THE "TRUE" RELIABILITY OF A 5-COMPONENT BRIDGE STRUCTURE
***// SUBROUTINE RBRIDG(QI,N,RRSS)
REAL*4 QI(N), RRSS
INTEGER N
IF(N.NE.5) THEN
   WRITE(1,'("WARNING: BRIDGE STRUCTURE ONLY USES", +"THE FIRST 5 COMPONENTS")')
ELSE
END IF
RRSS=(1-QI(1))*(1-QI(2))*(1-QI(3))*(1-QI(4))*(1-QI(5))=1-QI(1)*QI(2)*QI(3)*QI(4)*QI(5)
C(1-QI(5))+QI(2)+(1-QI(2))*QI(3)*(1-QI(4))*(-QI(1))*(1-QI(2))*QI(1)*QI(4)*(1-QI(5))
C(1-QI(1))*(1-QI(2))*QI(4)*(1-QI(5))=1-QI(1)*QI(2)*QI(4)*QI(5)
C(1-QI(4))*(1-QI(5))=1-QI(4)*QI(5)
C2*(1-QI(1))*(1-QI(2))*QI(3)*QI(4)*QI(5)
END

120
LIST OF REFERENCES


7. *Confidence Limits For Attributes Data*, LMSC-803324, Lockheed Missiles and Space Company.

BIBLIOGRAPHY


## INITIAL DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>No.</th>
<th>Copies</th>
<th>Recipient Details</th>
</tr>
</thead>
</table>
| 1.  | 2      | Defense Technical Information Center  
Cameron Station  
Alexandria, VA 22304-6145 |
| 2.  | 2      | Library, Code 52  
Naval Postgraduate School  
Monterey, CA 93943-5002 |
| 3.  | 4      | Professor W. Max Woods  
Naval Postgraduate School, Code OR-Wo  
Monterey, CA 93943-5000 |
| 4.  | 1      | Professor Lyn R. Whitaker  
Naval Postgraduate School, Code OR-Wh  
Monterey, CA 93943-5000 |
| 5.  | 1      | Chief of Naval Operations (OP-81)  
Department of the Navy  
Washington, DC 20350 |
| 6.  | 2      | Base Library  
FL 4887  
Luke Air Force Base, AZ 85309 |
| 7.  | 3      | LT Valerie A. Covington, USN  
P.O. Box 62  
Litchfield Park, AZ 85340 |