EUR 4940 e

COMMISSION OF THE EUROPEAN COMMUNITIES

CADI

A COMPUTER CODE FOR SYSTEM AVAILABILITY AND RELIABILITY EVALUATION

by

A.G. COLOMBO

1973

Joint Nuclear Research Centre
Ispra Establishment - Italy

Scientific Data Processing Centre - CETIS

LEGAL NOTICE

This report is on sale at the addresses listed on cover page 4

at the price of B. Fr. 60.—

Commission of the European Communities
D.G. XIII - C.I.D.
29, rue Aldringen
Luxembourg

April 1973

EUR 4940 e

CADI — A COMPUTER CODE FOR SYSTEM AVAILABILITY AND RELIABILITY EVALUATION by A.G. COLOMBO

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Scientific Data Processing Centre - CETIS
Luxembourg, April 1973 - 44 Pages - 19 Figures - B. Fr. 60.—

This report describes a code for evaluating the system availability, and therefore as a particular case, for evaluating the system reliability, of systems having elements for which a binary state is assumed : functioning or not-functioning.

The code is oriented towards a fault-tree representation. First of all it finds deterministically the minimal cut sets, then calculates analytically the availability of each primary event. Finally, it evaluates the system availability as a function of the most significant minimal cut sets and of the availability of the primary events appearing in these critical sets. It also supplies a

EUR 4940 e

CADI — A COMPUTER CODE FOR SYSTEM AVAILABILITY AND RELIABILITY EVALUATION by A.G. COLOMBO

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Scientific Data Processing Centre - CETIS
Luxembourg, April 1973 - 44 Pages - 19 Figures - B. Fr. 60.—

This report describes a code for evaluating the system availability, and therefore as a particular case, for evaluating the system reliability, of systems having elements for which a binary state is assumed : functioning or not-functioning.

The code is oriented towards a fault-tree representation. First of all it finds deterministically the minimal cut sets, then calculates analytically the availability of each primary event. Finally, it evaluates the system availability as a function of the most significant minimal cut sets and of the availability of the primary events appearing in these critical sets. It also supplies a

EUR 4940 e

CADI — A COMPUTER CODE FOR SYSTEM AVAILABILITY AND RELIABILITY EVALUATION by A.G. COLOMBO

Commission of the European Communities
Joint Nuclear Research Centre - Ispra Establishment (Italy)
Scientific Data Processing Centre - CETIS
Luxembourg, April 1973 - 44 Pages - 19 Figures - B. Fr. 60.—

This report describes a code for evaluating the system availability, and therefore as a particular case, for evaluating the system reliability, of systems having elements for which a binary state is assumed : functioning or not-functioning.

The code is oriented towards a fault-tree representation. First of all it finds deterministically the minimal cut sets, then calculates analytically the availability of each primary event. Finally, it evaluates the system availability as a function of the most significant minimal cut sets and of the availability of the primary events appearing in these critical sets. It also supplies a

table, oriented towards sensitivity analysis, which is of great interest for the design and control of complex systems.

The code is programmed in FORTRAN IV language and has been implemented on an IBM 370/165 computer. To find the cut sets of order 1, 2 and 3 of a system composed of 200 primary events required roughly two minutes of computer time. The determination of the unavailability of the same system, at 10 different mission times, requires about half a minute. This last computation concerns 86 independent primary events with constant failure and restoration rates, and 957 minimal cut sets of order 2.

table, oriented towards sensitivity analysis, which is of great interest for the design and control of complex systems.

The code is programmed in FORTRAN IV language and has been implemented on an IBM 370/165 computer. To find the cut sets of order 1, 2 and 3 of a system composed of 200 primary events required roughly two minutes of computer time. The determination of the unavailability of the same system, at 10 different mission times, requires about half a minute. This last computation concerns 86 independent primary events with constant failure and restoration rates, and 957 minimal cut sets of order 2.

table, oriented towards sensitivity analysis, which is of great interest for the design and control of complex systems.

The code is programmed in FORTRAN IV language and has been implemented on an IBM 370/165 computer. To find the cut sets of order 1, 2 and 3 of a system composed of 200 primary events required roughly two minutes of computer time. The determination of the unavailability of the same system, at 10 different mission times, requires about half a minute. This last computation concerns 86 independent primary events with constant failure and restoration rates, and 957 minimal cut sets of order 2.

EUR 4940 e

COMMISSION OF THE EUROPEAN COMMUNITIES

CADI

A COMPUTER CODE FOR SYSTEM AVAILABILITY AND RELIABILITY EVALUATION

by

A.G. COLOMBO

1973

Joint Nuclear Research Centre
Ispra Establishment - Italy

Scientific Data Processing Centre - CETIS

ABSTRACT

This report describes a code for evaluating the system availability, and therefore as a particular case, for evaluating the system reliability, of systems having elements for which a binary state is assumed : functioning or not-functioning.

The code is oriented towards a fault-tree representation. First of all it finds deterministically the minimal cut sets, then calculates analytically the availability of each primary event. Finally, it evaluates the system availability as a function of the most significant minimal cut sets and of the availability of the primary events appearing in these critical sets. It also supplies a table, oriented towards sensitivity analysis, which is of great interest for the design and control of complex systems.

The code is programmed in FORTRAN IV language and has been implemented on an IBM 370/165 computer. To find the cut sets of order 1, 2 and 3 of a system composed of 200 primary events required roughly two minutes of computer time. The determination of the unavailability of the same system, at 10 different mission times, requires about half a minute. This last computation concerns 86 independent primary events with constant failure and restoration rates, and 957 minimal cut sets of order 2.

KEYWORDS

C. CODES
RELIABILITY
AVAILABILITY
FORTRAN

INDEX

LIST OF FIGURES

1. INTRODUCTION

Several codes have been developed for system availability and reliability evaluation. In [1] various open literature codes have been examined and the problem of interdependence between representations, calculation methods and codes has been discussed. Among the most significant codes we quote the following: ARMM [2], [3]; SAFTE [3]; NOTED [4]; PREP-KITT [5], [6]; BOUNDS (NBB), [7].

This examination has clearly shown the advantages of a fault-tree representation as compared with alternative representations. The symbology of a fault-tree representation is described, for example, in [3] and [6], and is illustrated in Fig. 1. The advantages of a fault-tree representation can be summed up as follows: the more immediate representation of a real system, the ease with which the desired level of detail can be obtained even in several subsequent analyses, the possibility of taking human factors into account, and the ease with which any type of modification can be introduced into the system. In addition, and this is perhaps the most important characteristic, the possibility of specifying directly and automatically, by means of a computer, the minimal cut sets of the system.

Another consideration: none of the codes examined in [1] develops systematically what is certainly one of the most important aspects of an availability analysis, and that is the qualitative aspect characterized by the critical sets of the system.

The characteristics which differentiate the CADI code from the codes quoted, and which qualify it, are essentially as follows:

- the accentuation of the qualitative aspect of the availability analysis;
- the calculation of the availability of each primary event as derived from the transition matrix: good state - bad state of the event itself;

- the evaluation of the system unavailability, up to the first three bounds, by means of a very efficient algorithm;
- input cards reduced to a minimum and easy to prepare;
- extensive output lists which can be understood immediately.

This code makes it possible to tackle in a satisfactory manner practically all the problems of evaluating the availability of systems that can be represented by a fault-tree, as for example safety systems of reactors or electrical supply systems. However, taking into account the rapid developments in this field, more sophisticated algorithms are being elaborated for the analysis of particular aspects of availability problems. Some of these algorithms, which will be included progressively into the code, are mentioned in Sec. 3.3.

2. CALCULATION METHODS FOR SYSTEM AVAILABILITY

2.1 Some definitions

By a system we mean a set of elements the state of which is assumed to be binary: functioning - not functioning. A fault-tree represents the primary events and the final event characterized by these elements. The state 0 is attributed to a "bad" or "fault" event, and to the "good" event is attributed the state 1. The calculation of a system availability means evaluating the probability that the final event will be in the desired state: 1, as a function of the states of the primary events and of the probability distributions corresponding to these states.

Generally, the real systems for which an availability analysis is of interest are coherent, and thus indicating by

n	the number of primary events in the system
x_i	the state of the i^{th} primary event ($i = 1, 2, \dots, n$) ($x_i = 1$ if the event is in state 1; $x_i = 0$ if the event is in state 0)

$X = (x_i, x_2, \ldots, x_n)$ the vector which indicates the state of the n primary events

$S(X)$ the "structure function" of the system (this function assumes the value 1 if the system is in state 1, and the value 0 if the system is in state 0)

they satisfy the conditions [8]:

$S(X) \geqslant S(Y)$ for each $X \geqslant Y$ (where by $X \geqslant Y$ we mean $x_i \geqslant y_i$, $i = 1, 2, \ldots, n$)

$S(1) = 1$ where $1 = (1, 1, \ldots, 1)$

$S(0) = 0$ where $0 = (0, 0, \ldots, 0)$

Let us introduce some other definitions.

<u>Critical set:</u> a set of primary events which particularly influences the system availability.

From what has been said above, the definitions of two particular types of critical sets are derived.

<u>Minimal cut set:</u> minimal set of primary events which, if they are all in the 0 state, cause the system also to be in the 0 state, i.e. "unavailable".

<u>Minimal tie set:</u> minimal set of primary events, which, if they are all in the 1 state, mean that the system is also in the 1 state, i.e. "available".

In the following table some definitions are summarized.

EVENT	STATE OF EVENT	PROBABILITY
failure (bad event)	0	unavailability
no failure (good event)	1	availability

2.2 Remarks on some codes examined and the methodology of the CADI code

Codes for evaluating system availability are essentially of two types: simulation codes and analytical codes. The simulation codes, see for example [3], simulate the operation of the system by causing changes of state in the primary events with probabilities corresponding to reality. By making use of a logical description of the system, they then specify the state of the final event corresponding to each situation created. Finally, on the basis of the results obtained by repeating many "histories" of this sort, they evaluate the availability of the system.

The analytical codes are characterized by three basic steps:

- specification of critical sets of the system
- computation of the availability of each primary event
- evaluation of the availability of the system

The first step requires logical elaborations, the second numerical elaborations, and the third both logical and numerical elaborations.

Critical sets. Code [3], oriented towards a fault-tree representation, employs as critical sets the minimal cut sets which it finds either by simulation or by deterministic testing. Code [7], oriented towards a block diagram representation, finds deterministically the minimal tie sets, from which it passes to the minimal cut sets. Code [4], oriented towards a particular block diagram representation with NOT logic gates, uses as criti-

cal sets the tie sets which should be explicit in the representation.
CADI finds the minimal cut sets on the basis of the structure function of the system, which can be derived directly from the fault-tree representation.

<u>Primary event availabilities.</u> The calculation of the availability of each primary event requires rather laborious numerical elaborations. Very few codes, see for example [5], tackle this problem.
In Sec. 2.3 we report a method derived directly from the transition matrix between the two states (0 and 1) of the event, [9]. This method, which does not appear to have been applied by other codes, is employed by CADI.

<u>System availability.</u> In order to determine the availability of a system, on the basis of the minimal cut sets, or on the basis of the minimal tie sets, the bounds method [10] can be employed. This method, used by code [7], is also used by CADI and is described in Sec. 2.4.

2.3 Availability of a primary event as a function of time

The probability that a primary event will be in state 1: availability, depends upon various factors. If the following hypotheses are assumed:

- the failure distribution is known;
- it is possible to summarize in a distribution, which we will call "restoration distribution", the characteristics of maintenance, of repair, and in general of restoration to the initial conditions;
- the failure rate $\lambda(t)$ and the restoration rate $\mu(t)$ are time dependent but not dependent on the previous "history" of the event;

one can consider a markovian process with transition matrix

$$\begin{array}{c} \\ 1 \\ 0 \end{array} \begin{array}{cc} 1 & 0 \\ \left| \begin{array}{cc} 1-\lambda(t)dt & \lambda(t)dt \\ \mu(t)dt & 1-\mu(t)dt \end{array} \right| \end{array} \qquad (1)$$

From matrix (1), indicating by $p_o(t)$ and $p_1(t)$ respectively the probability that the primary event will be in states 0 and 1 at time t, one obtains

$$\begin{cases} \dfrac{dp_1}{dt} = -\lambda(t)p_1(t) + \mu(t)p_o(t) \\ \dfrac{dp_o}{dt} = \lambda(t)p_1(t) - \mu(t)p_o(t) \end{cases} \tag{2}$$

To which must be added the initial condition

$$p_1(0) = 1 \tag{3}$$

In particular, in cases where the failure and restoration rates are not time-dependent, i.e., if

$$\begin{aligned} \lambda(t) &= \lambda \\ \mu(t) &= \mu \end{aligned} \tag{4}$$

one is reduced to a homogeneous process and obtains

$$p_o(t) = \frac{\lambda}{\lambda+\mu}\left[1 - e^{-(\lambda+\mu)t}\right] \tag{5}$$

At the beginning of the mission, see (3), the availability of each primary event is equal to 1. In the case of an unrestorable event, the availability is called reliability and tends towards 0 with time. In the case of a restorable event the availability initially follows the same trend as the reliability, and then remains at a higher value. In particular, if the failure and restoration rates are constant, both the reliability and the availability have an exponential trend as indicated in Fig. 2. The reliability tends asymptotically to zero, while the availability tends asymptotically to the value $\mu/(\lambda+\mu)$.

2.4 System availability and the bounds method

Let us consider a system having r minimal tie sets. Then, indicating by 1 the good state of the primary events and of the system, by A (availability) the probability that the system is in state 1, by T_i $(i = 1, 2, \ldots, r)$ the event "minimal tie set index i is good", we have

$$A = Pr(T_1 \cup T_2 \cup \ldots T_r) \tag{6}$$

(Probability that at least one minimal tie set is good, and therefore that there exists at least one tie between any primary event and the final event).

Similarly, indicating by C_j $(j = 1, 2 \ldots, s)$ the event "minimal cut set index j of the system is good", we have

$$A = Pr(C_1 \cap C_2 \cap \ldots C_s) \tag{7}$$

(Probability that all the minimal cut sets of the system are good, and therefore that there exists at least one good primary event for each cut set).

Taking into account the relations between unions of a set of events, we obtain from (6)

$$
\begin{aligned}
A = Pr(T_1 \cup T_2 \cup \ldots T_r) &\leq \sum_i Pr(T_i) \\
&\geq \sum_i Pr(T_i) - \sum_{i<j} Pr(T_i \cap T_j) \\
&\leq \sum_i Pr(T_i) - \sum_{i<j} Pr(T_i \cap T_j) + \\
&+ \sum_{i<j<k} Pr(T_i \cap T_j \cap T_k) \\
&\ldots\ldots\ldots\ldots\ldots\ldots
\end{aligned}
\tag{8}
$$

where

$\sum_{i} Pr(T_i)$ denotes the sum of probabilities that each tie set is good

$\sum_{i<j} Pr(T_i \cap T_j)$ denotes the sum of the probabilities that the tie sets taken two at a time jointly are good

the indexing $i < j$, $i < j < k$, ... guarantees that a given set is not counted more than once.

Furthermore, because

$$Pr(C_1 \cap C_2 \cap \ldots C_s) = 1 - Pr(\bar{C}_1 \cup \bar{C}_2 \cup \ldots \bar{C}_s) \tag{9}$$

we obtain from (7)

$$\begin{aligned} A = 1 - Pr(\bar{C}_1 \cup \bar{C}_2 \cup \ldots \bar{C}_s) &\geq 1 - \sum_{i} Pr(\bar{C}_i) \\ &\leq 1 - \sum_{i} Pr(\bar{C}_i) + \sum_{i<j} Pr(\bar{C}_i \cap \bar{C}_j) \\ &\geq 1 - \sum_{i} Pr(\bar{C}_i) + \sum_{i<j} Pr(\bar{C}_i \cap \bar{C}_j) + \\ &\quad - \sum_{i<j<k} Pr(\bar{C}_i \cap \bar{C}_j \cap \bar{C}_k) \\ &\quad \ldots\ldots\ldots\ldots\ldots\ldots \end{aligned} \tag{10}$$

Equations (8) and (10) make it possible for us to calculate the availability of a system, however complex it may be, even with connected primary events. But practically, for systems which include many critical sets (of the order of hundreds) and with critical sets consisting of several primary events (even if only 3 or 4), the calculation time becomes prohibitive. Among other things, even to find the critical sets, under these conditions, would take too long. Therefore, the introduction of approximations becomes unavoidable. These approximations are essentially of three kinds

- To assume that the primary events are unconnected. This condition is generally satisfied, but must be verified at the representation level.

- To consider only the most significant critical sets, for example the minimal cut sets of a lower order of the system, and those of an immediately higher order. It will be noted that the relative error made in such a case in evaluating the system unavailability is of the order of p_i^2, if p_i is the average value of the unavailability of each primary event, and if these unavailabilities do not differ very much from each other. A similar reasoning can be made for a case in which the minimal tie sets of a system are being considered.

- To limit oneself to considering the first terms of the development of expressions (6) and (7). For example, by considering only the first summation of expressions (8) and (10), which is equal to assuming the critical sets to be unconnected, approximations of the same type as those given in Fig. 3 are obtained. If the second summation is also taken into consideration, smaller errors and of opposite sign are obtained. By increasing the number of terms, the errors become always smaller and of alternate sign.

Fig. 3 shows that the tie set bounds (8) are most useful in the low availability region, while the cut set bounds (10) are most useful in the high availability region.

3. THE CADI CODE

CADI is a code for evaluating the unavailability of systems having high availability, is written in FORTRAN IV language and has been implemented on an IBM 370/165 computer. It consists of two programs: the CUTDET program and the AVANA program.

3.1 The CUTDET Program

Given the structure function of a system, the CUTDET program finds the minimal cut sets. This structure function must be described in FORTRAN language by means of a subroutine called TREE. This subroutine is written directly on the basis of a fault-tree representation. An example of how such a subroutine can be written is shown in Fig. 4.

The primary events are indicated by E(I), I = 1, 2,... NE (where NE is the highest index of the primary events). The derived events (gates) are indicated by G(I), I = 1, 2,... NG (where NG is the highest index of the gates). The final event is indicated conventionally by TOP.

Two important remarks:

- Not all the primary events E(I) and not all the gates G(I) need necessarily appear in the fault-tree. Some can be left out. This characteristic of the program is very useful during design because it allows the analysis of alternative versions of the same system without subsequent enumerations of the primary events, and of the gates, in cases where some of them are substituted or eliminated.

- If one gate employs other gates as input, these (last-mentioned) gates must be defined first.

The program subsequently finds the minimal cut sets of order 1, 2, ... 5. The maximum order of the minimal cut sets to be found must be chosen by the user.

The minimal cut sets, in addition to being printed, can also be punched on cards. These cards are then used as input for the AVANA program which calculates the system availability.

3.1.1 Input cards

In addition to the subroutine TREE, only two cards are necessary.

A. TITLE CARD (Format : 20A4)

This card is used to identify the system being studied.
From column 2 to column 80, any alphanumerical character can be punched.

B. PARAMETERS CARD (Format : 4I6)

The following parameters should be punched:

NE = Highest index of the primary events of the system; (NE $\leq$ NEMAX = 500).

According to what has been said in Sec. 3.1, the number of primary events considered in the fault-tree can be less than NE.

NG = Highest index of the system gates; (NG $\leq$ NGMAX = 500).

As with the primary events, the number of gates represented in the fault-tree can be less than NG.

MAX = Maximum order of the minimal cut sets of interest; ($1 \leqslant MAX \leqslant 5$).

IPP = Parameter which determines if the minimal cut sets should only be printed or punched as well.

If IPP = 1 the minimal cut sets are only printed.

If IPP = 2 the minimal cut sets are both printed and punched.

3.1.2 Output from CUTDET

First of all the program prints all the input information. This print allows easy control of the input parameters and shows up any possible error in the definitions of the parameters themselves.

The output information is collected in two tables. In the first table are reported all the minimal cut sets found listed according to their order. The second table, oriented towards a system sensitivity analysis, constitutes one of the characteristics peculiar to the CADI code. In it are indicated, for each order of minimal cut sets:

- the number of minimal cut sets;
- the total number of different primary events appearing in these cut sets;
- the complete list of indexes of these events.

In this way the primary events are automatically classified in order of importance with respect to the availability of the system. This information allows specification of alternative solutions to the same problem without carrying out numerical calculations but on the basis of qualitative considerations alone. These solutions can be oriented, for example, to contain the cost or to achieve a certain level of availability.

In the case where the number of minimal cut sets (NCT) of the system is more than NCTMAX = 4000, the program does not print the two tables described here, but prints only the NCTMAX minimal cut sets of the system which have been found first. This is due to the need to dimension the program arrays. The NCTMAX parameter, as also the NEMAX and NGMAX parameters, can easily be modified. This, of course, implies a variation in the memory requirements of the program; see the next section. For the printouts refer to the sample problem in Sec. 4 (a 22 primary event system is considered).

3.1.3 Memory requirements and running time

The maximum values of the NE, NG and NCTMAX parameters (NEMAX = 500, NGMAX = 500, NCTMAX = 4000) have been chosen so as to make complete use of the smallest memory partition of the present computing installation, which is of 132K bytes. These values can easily be changed by substituting five cards of the program. The memory requirements M_{CUTDET} can be estimated by means of the relation

$$M_{CUTDET}(\text{bytes}) \simeq 32,000 + 28 \cdot NEMAX + 4 \cdot NGMAX + 20 \cdot NCTMAX$$

The running time T_{CUTDET} depends essentially upon NG and on the number of combinations of NE objects taken MAX at a time, which is the number of states of the system that the program has to examine.

To find the cut sets of order 1, 2 and 3 of a system composed of 200 primary events and 50 gates required roughly two minutes.

3.2 The AVANA program

The AVANA program evaluates the unavailability of a system as a function of the most significant minimal cut sets and of the characteristics of the events appearing in these sets. It consists of two subroutines: AVACOM and CHASYS.

The subroutine AVACOM computes the unavailability of each primary event which appearsin the minimal cut sets as a function of: the type of failure distribution, the type of restoration distribution, the parameters of these distributions themselves.
At present the subroutine only handles the failure and restoration distributions at a constant rate and the constant availability distribution (failure distribution with a constant cumulative probability during the mission time considered, no restoration). Other distributions will be introduced shortly.
The method of computation is based upon the transition matrix (1). In general, it requires a numerical integration to determine the unavailability of a primary event. In particular, if the failure and restoration rates are constant, the computation is reduced to an analytical integration, see (5); this decreases the calculation time quite considerably.

The subroutine CHASYS computes the upper bound of the system unavailability, and the successive two bounds, as a function of the more significant minimal cut sets and of the unavailability of each primary event which appear in these critical sets.
The evaluation of the concept "most significant cut sets" is left to the user. It will be noted that, for high availability systems, if the unavailabilities of the primary events do not differ very much from each other, we can consider only the minimal cut sets of lowest order in the system, and eventually also those of the immediately higher order.

3.2.1 Input cards

The AVANA program requires the following input cards.

A. TITLE CARD (Format : 20A4)

As for the CUTDET program, this card serves to identify the system being studied. From column 2 to column 80 any alphanumerical character can be punched.

B. <u>PARAMETERS CARD</u> (Format : 6I6)

The following parameters should be punched:

NE - Highest index of the primary events of the system; (NE $\leqslant$ NEMAX = 500).

NECHR - Number of different primary events appearing in the cut sets under consideration, and for which the failure and restoration characteristics are given; NECHR $\leqslant$ NE.

NCT - Number of cut sets under consideration; (NCT $\leqslant$ NCTMAX = 1000).

IA - Parameter which determines whether it is the unavailability or the unreliability of the system which has to be computed.

If IA = 1 the program will compute the system unavailability.
If IA = 2 the input information on the restoration of the primary events will be ignored, and the program will compute the system unreliability.

IB - Parameter which determines whether only the first bound of the system unavailability must be computed or whether the successive two bounds should also be computed.

If IB = 1 only the first bound is computed.
If IB = 2 the successive two bounds are also computed.

IT - Parameter which selects the type of time points of interest.

If IT = 1 the time points are at a constant interval.
If IT = 2 the time points can be spaced out arbitrarily (see the following item).

C. <u>NUMBER OF DATA POINTS AND MISSION TIME CARDS</u>

These cards differ according to the value of the parameter IT.

<u>Case IT = 1</u> In this case only one card (Format : I6, E12. 5) is punched.

NP = Number of time points of interest,at constant interval; ($1 \leq NP \leq 10$).

TMAX = Maximum mission time; should be TMAX > 0. (Time interval = TMAX/NP)

Case IT = 2 In this second case two or three cards should be punched.

NP = Number of time points of interest; ($1 \leq NP \leq 10$). (Format : I6)

TM = Vector of length NP containing the mission times of interest. (Format : 6E12.5).

The times should be in ascending order, and should be $TM(1) > 0$.

One or two cards are necessary to punch this vector, according to the value of NP.

D. PRIMARY EVENTS CHARACTERISTICS CARDS (Format : 2I6, 2E12.5, I6, 2E12.5)

One card is required for each primary event which appears in the cut sets under consideration. In each of these cards should be punched:

I = Index of the primary event.

IFAIL = Type of failure distribution of the primary event.

PFAIL = Vector of length 2 containing the failure rate parameters.

IREP = Type of restoration distribution of the primary event.

PRER = Vector of length 2 containing the restoration distribution parameters.

The pairs of distribution types which the code can handle at present are as follows:

IFAIL	IREP	NOTE
1	0	constant cumulative probability of failure, no restoration; i. e. constant availability for the mission time of interest
2	0	constant failure rate, no restoration
2	2	constant failure rate, constant restoration rate

As far as the parameters are concerned, see the following table:

IFAIL or IREP	PFAIL(1) or PREP(1)	PFAIL(2) or PREP(2)	NOTE
0	-	-	the parameters have no significance
1	F	-	F = constant cumulative probability of failure of the primary event
2	λ or μ	-	λ = constant failure rate of the primary event μ = constant restoration rate of the primary event

The second parameter will be used for the two parameter distributions which will be introduced shortly into the code.

E. MINIMAL CUT SETS CARDS (Format : 5I6)

One card should be punched for each minimal cut set. These cards are punched by the CUTDET program.

3.2.2 Output from AVANA

As with the CUTDET program, the AVANA program also first prints all the input information.

The output information is collected in three tables.

For each mission time of interest:

- the first table shows the unavailability of each primary event,
- the second table shows the unavailability of the minimal cut sets,
- the third table shows the upper bound, and if required the two successive bounds, of the system unavailability.

For the printouts refer to the sample problem in Sec. 4.

3.2.3 Memory requirements and running time

The criterion adopted for the choice of maximum values of parameters NE and NCT (NEMAX = 500, NCTMAX = 1000), also for this program, is the complete utilization of the smallest memory partition of the computing installation. These values can easily be changed by substituting five cards of the program.

A remark: In the CUTDET program NCTMAX = 4000 has been assumed, while in the AVANA program we have assumed NCTMAX = 1000. This is due to the fact that for a qualitative analysis of a system it may be useful to know the cut sets of a higher order as well. But, for the analytical calculation of the system availability, it is sufficient to take only the most significant cut sets into consideration.

The memory requirements M_{AVANA} can be estimated by means of the relation

$$M_{AVANA}(\text{bytes}) \simeq 32,000 + 64 \cdot NEMAX + 60 \cdot NCTMAX$$

The running time depends essentially upon NECHR, on the failure and restoration distribution of the primary events, on the number and order of the cut sets under consideration, on NP and on the parameter IB.

For systems with some hundreds of primary events having a constant rate of failure and restoration, and with about a thousand cut sets of order 2, the calculation of the upper bound of the unavailability of the system at 10 different times needs only a few tens of seconds.

3.3 Developments planned

There are various improvements planned for progressive introduction into the code. The most important are:

- The possibility of handling failure and restoration distributions having rates variable in time; for example lognormal, normal, gamma, or Weibull distributions.
- The possibility of dealing with connected events, for example sequential events.
- Research into the most significant minimal cut sets of the system taking into account the availabilities of the primary events. This formulation will save a great amount of computer time in cases of large systems with primary events which have very different availabilities, and can be considered equivalent to a sensitivity analysis carried out in a classical manner.

4. SAMPLE PROBLEM

In order to describe the details of the code, we will consider the system presented in Fig. 5. The failure and restoration characteristics of the primary events are reported in Fig. 6. We will determine the unavailability and unreliability of the system for mission times of between 1 and 10,000 hours.

CUTDET program

The subroutine TREE, shown in Fig. 7, is derived directly from the fault-tree in Fig. 5; Fig. 8 shows the two input cards.

Figs. 9, 10 and 11 show the program prints. Fig. 9 gives the input information. Fig. 10 shows a summary of the logical analysis of the system. This last figure illustrates that the system does not have cut sets of order 1, or cut sets of order 4, and that, of the 22 primary events of which the system is composed, only 13 influence the system availability in a significant manner. These 13 primary events are those which appear in the cut sets of orders 2 and 3. For a numerical calculation of the system availability it is sufficient to consider these 13 primary events, the 7 minimal cut sets of order 2 (the more important), and the 3 minimal cut sets of order 3. In Fig. 11 a complete list of the minimal cut sets of the system is given.

AVANA program

The input cards for the calculation of the system unavailability are given in Fig. 12. In Figs. 13, 14 and 15 are shown the input information prints. Fig. 13 shows the parameters of the problem, in Fig. 14 are shown the characteristics of the primary events, and Fig. 15 groups the minimal cut sets.

In Figs. 16, 17 and 18 are shown the output information prints, Fig. 16 shows the unavailability of the primary events, Fig. 17 indicates the unavailability of the minimal cut sets. From Fig. 18 we can see that the system availability stabilizes after about 1,000 hours; it will be noticed, moreover, that the first and third bound of the system unavailability practically coincide, which is due to the high availability of the primary events.

The input cards of Fig. 11, with one modification only on the second card, column 24: IA = 2 instead of IA = 1, allow the system unreliability to be determined. Fig. 19 shows the system unreliability for the mission times under consideration. For very brief times the unavailability and unreliability practically coincide. Then, while the unavailability stabilizes at a value of $0.133 \cdot 10^{-4}$, the unreliability continues to increase quite noticeably, and tends to 1 with time. Finally, it will be noted that for these higher mission times, Fig. 19 reveals an appreciable difference between the first and third bound of the system unreliability. This is due to the fact that the system reliability is small.

ACKNOWLEDGEMENTS

Part of this work has been carried out in the frame of the contract EURATOM-CNEN 027-71-PIPGI (RVC-71-064-PEC). Thanks are due to Mr. R. Di Sapia of CNEN and to Mr. G. Volta of EURATOM, who are responsible for the contract, for support and discussions.

Acknowledgements are also due to Mr. L. Farese for critical reading of the manuscript and subsequent helpful suggestions and Messrs. G. Reina and L. Haemers for programming in connection with this work.

REFERENCES

[1] Colombo A.G., Ricchena R., Volta G., "Survey and Critical Analysis of Programs for System Reliability Computation", CREST Specialist Meeting on Reliability, Munich, May 1971, MRR 90 (1971)

[2] McKnight C.W., Modiest L.J., Schmidt N.E., "An Automatic Reliability Mathematical Model", Proc. 1965 Ann. Symp. on Reliability, Miami Beach, Florida, 1965, pp. 518-531

[3] Garrick B.J., Gekler W.C., Goldfisher L., Shimizn B., Wilson J.H., "Reliability Analysis of Nuclear Power Plant Protective Systems", HN-190, AEC - R & D - Report, Holmes & Narver, Inc., Nuclear Division, Los Angeles (1967)

[4] Woodcock E.R., "The Calculation of Reliability of Systems. The Program NOTED", UKAEA - AHSB(5), R117 (1968)

[5] Vesely W.E., "A Time-dependent Methodology for Fault-tree Evaluation", Nuclear Engineering and Design, 13 (1970), pp. 337-360

[6] Vesely W.E., Narum R.E., "Prep and Kitt: Computer Codes for the Automatic Evaluation of a Fault-tree", IN - 1349, Idaho Nuclear Corporation, Idaho (1970)

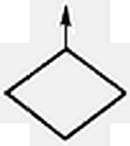
[7] Nelson A. C.Jr., Batts J.R., Beadles R.L., "A Computer Program for Approximating System Reliability", IEEE Transaction on Reliability, Vol.R-19, No.2, May 1970, pp. 61-65

[8] Essary J.D., Proschan F., "Coherent Structures of Non-Identical Components", Technometrics, Vol.5, No.2, 1963a, pp. 191-209

[9] Shooman M. L., "Probabilistic Reliability - An Engineering Approach", McGraw Hill Book Co., New York, 1967

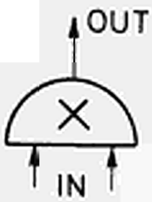
[10] Messinger M., Shooman M. L., "Reliability Approximations for Complex Structures", Proc. IEEE Ann. Symp. on Reliability, Washington, D. C. (1967), pp. 169-174

Primary event, characterized by a failure distribution and a restoration distribution



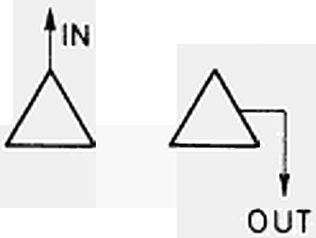
Event which is not developed further due to lack of information. This event must be characterized as a primary event

OR gate. The output event occurs only if at least one of the input events occurs



AND gate. The output event occurs if, and only if, all the input events occur

Event descriptor. This symbol is used to describe the event represented by a gate



Transfer symbols. These symbols are used to transfer an entire part of the tree to other locations on the tree

Fig. 1 - Fault-tree symbology

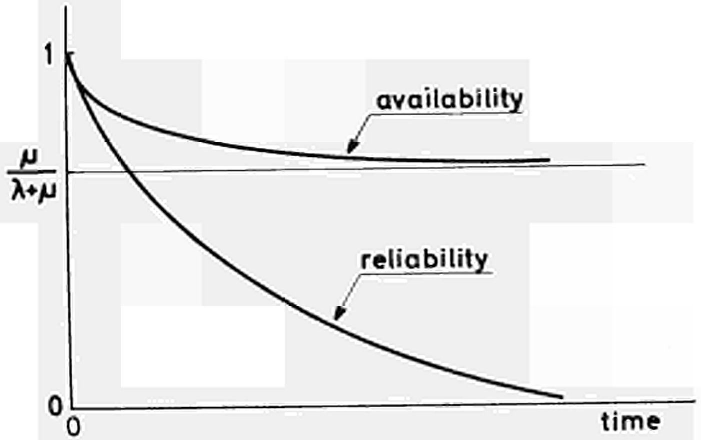


Fig. 2 – Availability and reliability of a primary event (λ and μ costants)

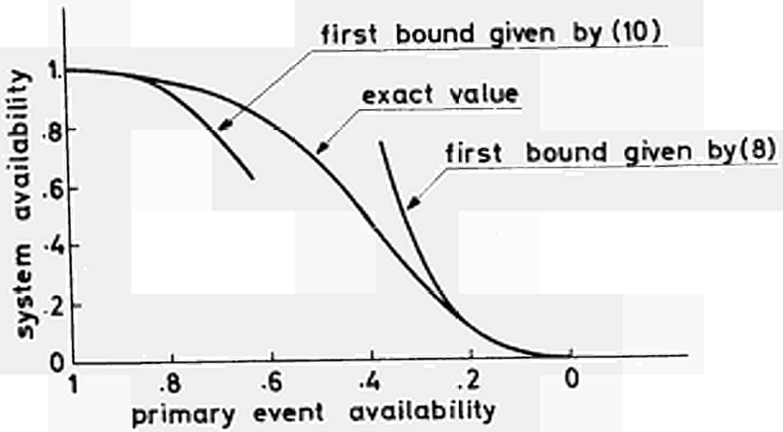
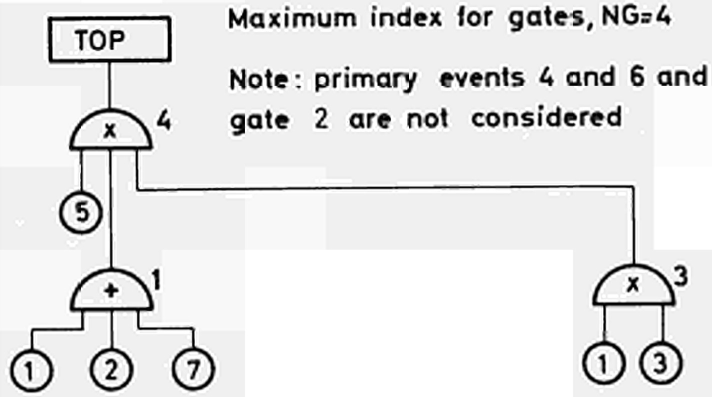


Fig. 3 – System availability approximations

Maximum index for primary events, NE=7

Maximum index for gates, NG=4

Note: primary events 4 and 6 and gate 2 are not considered




```
SUBROUTINE TREE (E, G, TOP)
LOGICAL E(1), G(1), TOP
G(1) = E(1).OR.E(2).OR.E(7)
G(3) = E(1). AND. E(3)
G(4) = G(1). AND. G(3). AND . E(5)
TOP = G(4)
RETURN
END
```

Fig. 4 – Fault-tree representation and subroutine TREE of a 7 primary event system

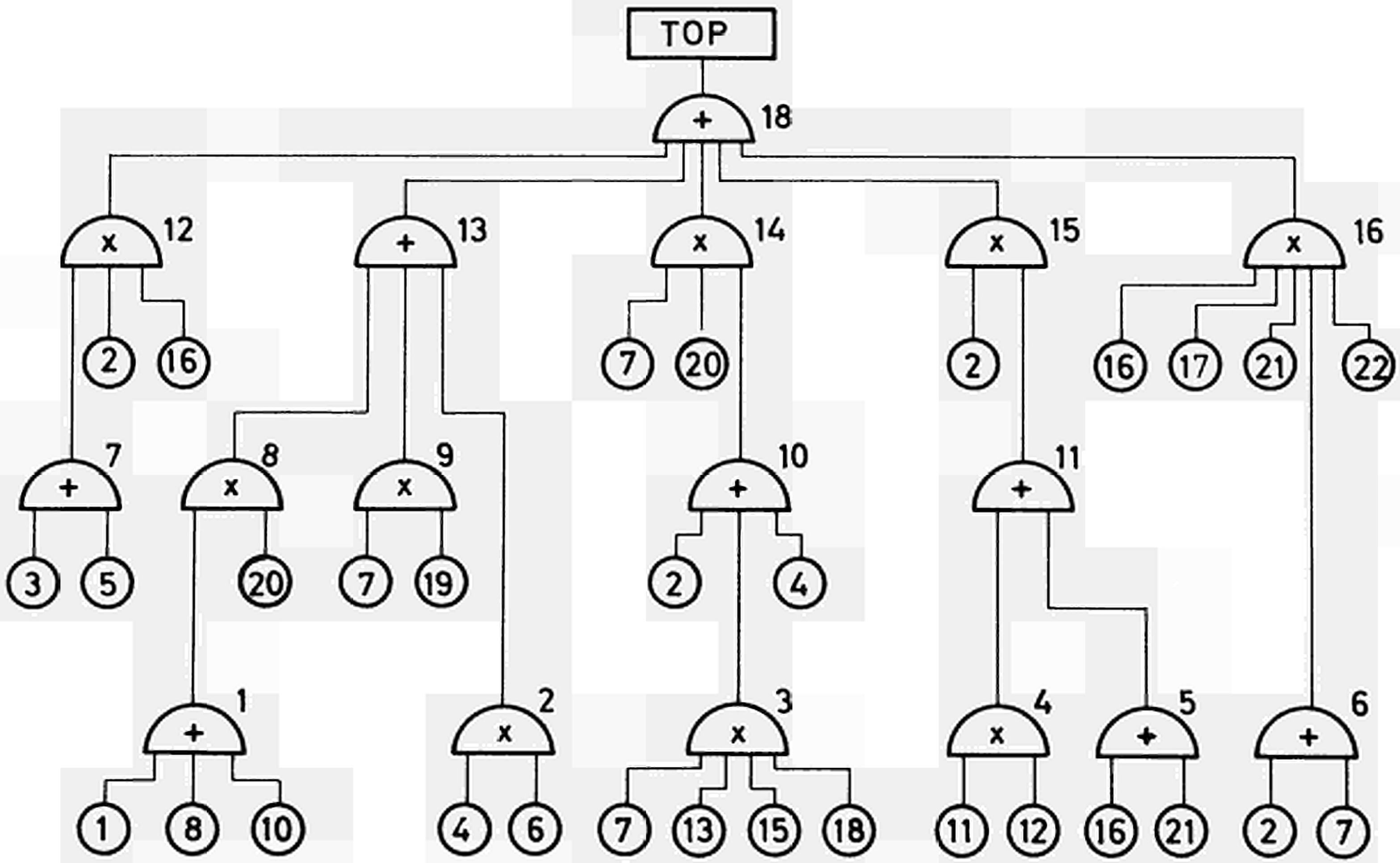


Fig. 5 - Fault-tree of the sample problem

PRIMARY EVENT INDEX	FAILURE CHARACTERISTICS*		RESTORATION CHARACTERISTICS*	
	Distribution	Parameter	Distribution	Parameter
1	Exp	$\lambda = .50.10^{-5}$	Exp	$\mu = .30.10^{-2}$
2	Exp	$\lambda = .30.10^{-4}$	Exp	$\mu = .40.10^{-1}$
3	Const	$F = .\ 5.10^{-4}$	-	-
4	Exp	$\lambda = .80.10^{-5}$	Exp	$\mu = .50.10^{-2}$
5	Exp	$\lambda = .35.10^{-5}$	Exp	$\mu = .45.10^{-2}$
6	Exp	$\lambda = .40.10^{-4}$	Exp	$\mu = .25.10^{-1}$
7	Exp	$\lambda = .40.10^{-5}$	Exp	$\mu = .25.10^{-2}$
8	Exp	$\lambda = .20.10^{-5}$	Exp	$\mu = .80.10^{-2}$
9	Const	$F = .\ 1.10^{-3}$	-	-
10	Exp	$\lambda = .15.10^{-4}$	Exp	$\mu = .75.10^{-1}$
11	Const	$F = .10.10^{-3}$	-	-
12	Exp	$F = .10.10^{-3}$	-	-
13	Exp	$\lambda = .30.10^{-5}$	Exp	$\mu = .20.20^{-2}$
14	Exp	$\lambda = .30.10^{-5}$	Exp	$\mu = .20.10^{-2}$
15	Exp	$\lambda = .30.10^{-5}$	Exp	$\mu = .20.10^{-2}$
16	Exp	$\lambda = .60.10^{-4}$	Exp	$\mu = .20.10^{-1}$
17	Const	$F = .\ 5.10^{-4}$	-	-
18	Exp	$\lambda = .15.10^{-6}$	-	-
19	Exp	$\lambda = .50.10^{-4}$	Exp	$\mu = .40.10^{-1}$
20	Exp	$\lambda = .75.10^{-4}$	Exp	$\mu = .30.10^{-1}$
21	Exp	$\lambda = .25.10^{-4}$	Exp	$\mu = .15.10^{-1}$
22	Exp	$\lambda = .50.10^{-6}$	-	-

* Exp means exponential distribution (constant rate); λ, μ = value of the rate. Const means constant cumulative probability distribution for the mission time of interest; F = value of the constant probability.

Fig. 6 - Primary event characteristics of the sample problem

```
SUBROUTINE TREE(E,G,TOP)
LOGICAL E(1),G(1),TOP
G(1)=E(1).OR.E(8).OR.E(10)
G(2)=E(4).AND.E(6)
G(3)=E(7).AND.E(13).AND.E(15).AND.E(18)
G(4)=E(11).AND.E(12)
G(5)=E(16).OR.E(21)
G(6)=E(2).OR.E(7)
G(7)=E(3).OR.E(5)
G(8)=E(20).AND.G(1)
G(9)=E(7).AND.E(19)
G(10)=E(2).OR.E(4).OR.G(3)
G(11)=G(4).OR.G(5)
G(12)=G(7).AND.E(2).AND.E(16)
G(13)=G(8).OR.G(9).OR.G(2)
G(14)=E(7).AND.E(20).AND.G(10)
G(15)=E(2).AND.G(11)
G(16)=E(16).AND.E(17).AND.E(21).AND.G(6).AND.E(22)
G(18)=G(12).OR.G(13).OR.G(14).OR.G(15).OR.G(16)
TOP=G(18)
RETURN
END
```

Fig. 7 - Program CUTDET. Subroutine TREE.

```
                                                      SAMPLE PROBLEM
 22    18     5    2
```

Fig. 8 - Program CUTDET. Input cards to find the minimal cut sets of the system.

CUTDET PROGRAM - INPUT INFORMATION

PROBLEM TITLE	SAMPLE PROBLEM
NUMBER OF PRIMARY EVENTS, NE	22
NUMBER OF DERIVED EVENTS (GATES), NG	18
UPPER ORDER OF THE MINIMAL CUT SETS TO BE CHECKED, MAX	5
PRINT-PUNCH PARAMETER, IPP	2
IF IPP=1 THE MINIMAL CUT SETS ARE PRINTED ONLY	
IF IPP=2 THE MINIMAL CUT SETS ARE BOTH PRINTED AND PUNCHED	

Fig. 9 - Program CUTDET. Input Information.

CUTDET PROGRAM - OUTPUT INFORMATION

ORDER OF THE MINIMAL CUT SETS	NUMBER OF MINIMAL CUT SETS	TOTAL NUMBER OF DIFFERENT PRIMARY EVENTS	INDEX OF EACH PRIMARY EVENT
1	0	0	0
2	7	11	1 2 4 6 7 8 10 16 19 20 21
3	3	6	2 4 7 11 12 20
4	0	0	0
5	2	9	7 13 15 16 17 18 20 21 22
FROM 1 TO 5	12	18	1 2 4 6 7 8 10 11 12 13 15 16 17 18 19 20 21 22

Fig. 10 - Program CUTDET. Output Information.
Summary of the logical analysis of the system.

MINIMAL CUT SETS

SET ORDER	SET INDEX	PRIMARY EVENTS INDEXES
2	1	1 20
2	2	2 16
2	3	2 21
2	4	4 6
2	5	7 19
2	6	8 20
2	7	10 20
3	1	2 7 20
3	2	2 11 12
3	3	4 7 20
5	1	7 13 15 18 20
5	2	7 16 17 21 22

Fig. 11 - Program CUTDET. Output Information (cont.) Minimal cut sets.

```
                                                                                SAMPLE PROBLEM
   22      13      10       1       2       2
    9
1.      E+00 3.     E+00 1.     E+01 3.     E+01 1.     E+02 3.     E+02
1.      E+03 3.     E+03 1.     E+04
    1       2  .5     E-05                  2  .3     E-02
    2       2  .3     E-04                  2  .4     E-01
    4       2  .8     E-05                  2  .5     E-02
    6       2  .4     E-04                  2  .25    E-01
    7       2  .4     E-05                  2  .25    E-02
    8       2  .2     E-05                  2  .8     E-02
   10       2  .15    E-04                  2  .75    E-01
   11       1  .1     E-03
   12       1  .1     E-03
   16       2  .6     E-04                  2  .2     E-01
   19       2  .5     E-04                  2  .4     E-01
   20       2  .75    E-04                  2  .3     E-01
   21       2  .25    E-04                  2  .15    E-01
    1      20
    2      16
    2      21
    4       6
    7      19
    8      20
   10      20
    2       7      20
    2      11      12
    4       7      20
```

Fig. 12 - Program AVANA. Input cards to compute system unavailability.

AVANA PROGRAM - INPUT INFORMATION

PROBLEM TITLE	SAMPLE PROBLEM
NUMBER OF PRIMARY EVENTS, NE	22
NUMBER OF PRIMARY EVENTS OF WHICH THE CHARACTERISTICS ARE GIVEN, NECHR	13
NUMBER OF MINIMAL CUT SETS, NCT	10
RESTORATION PARAMETER, IA IF IA=1 THE INPUT INFORMATION ON RESTORATION IS CONSIDERED IF IA=2 SUCH INFORMATION IS IGNORED, (UNAVAILABILITY=UNRELIABILITY)	1
BOUNDS PARAMETER, IB IF IB=1 ONLY THE UPPER BOUND OF THE SYSTEM UNAVAILABILITY IS COMPUTED IF IB=2 ALSO THE TWO SUCCESSIVE BOUNDS ARE COMPUTED	2
TIME POINTS PARAMETER, IT IF IT=1 THE TIME POINTS ARE EVENLY SPACED, AT INTERVAL TMAX/NP IF IT=2 THE TIME POINTS ARE ARBITRARILY SPACED, EXPLICITLY INPUT	2
TOTAL NUMBER OF TIME POINTS, NP	9

TIME POINTS (HOURS)

0.10000E 01	0.30000E 01	0.10000E 02	0.30000E 02	0.10000E 03
0.30000E 03	0.10000E 04	0.30000E 04	0.10000E 05	

Fig. 13 - Program AVANA. Input Information. Parameters.

PRIMARY EVENTS DATA

PRIMARY EVENT INDEX	TYPE OF FAILURE DISTRIB	FAILURE PARAMETERS		TYPE OF RESTOR DISTRIB	RESTORATION PARAMETERS	
1	2	0.50000E-05	0.0	2	0.30000E-02	0.0
2	2	0.30000E-04	0.0	2	0.40000E-01	0.0
4	2	0.80000E-05	0.0	2	0.50000E-02	0.0
6	2	0.40000E-04	0.0	2	0.25000E-01	0.0
7	2	0.40000E-05	0.0	2	0.25000E-02	0.0
8	2	0.20000E-05	0.0	2	0.80000E-02	0.0
10	2	0.15000E-04	0.0	2	0.75000E-01	0.0
11	1	0.10000E-03	0.0	0	0.0	0.0
12	1	0.10000E-03	0.0	0	0.0	0.0
16	2	0.60000E-04	0.0	2	0.20000E-01	0.0
19	2	0.50000E-04	0.0	2	0.40000E-01	0.0
20	2	0.75000E-04	0.0	2	0.30000E-01	0.0
21	2	0.25000E-04	0.0	2	0.15000E-01	0.0

Fig. 14 - Program AVANA. Input Information (cont.)
Characteristics of the primary events.

MINIMAL CUT SETS

CUT SET INDEX	PRIMARY EVENTS INDEXES		
1	1	20	
2	2	16	
3	2	21	
4	4	6	
5	7	19	
6	8	20	
7	10	20	
8	2	7	20
9	2	11	12
10	4	7	20

Fig. 15 - Program AVANA. Input Information (cont.)
Minimal cut sets.

AVANA PROGRAM - OUTPUT INFORMATION

PRIMARY EVENTS UNAVAILABILITY

PRIMARY EVENT INDEX	MISSION TIME (HOURS) 0.100E 01	0.300E 01	0.100E 02	0.300E 02	0.100E 03	0.300E 03	0.100E 04	0.300E 04	0.100E 05
1	0.499E-05	0.149E-04	0.493E-04	0.143E-03	0.432E-03	0.988E-03	0.158E-02	0.166E-02	0.166E-02
2	0.294E-04	0.848E-04	0.247E-03	0.524E-03	0.736E-03	0.749E-03	0.749E-03	0.749E-03	0.749E-03
4	0.798E-05	0.238E-04	0.780E-04	0.223E-03	0.629E-03	0.124E-02	0.159E-02	0.160E-02	0.160E-02
6	0.395E-04	0.116E-03	0.354E-03	0.844E-03	0.147E-02	0.160E-02	0.160E-02	0.160E-02	0.160E-02
7	0.399E-05	0.120E-04	0.395E-04	0.116E-03	0.354E-03	0.844E-03	0.147E-02	0.160E-02	0.160E-02
8	0.199E-05	0.593E-05	0.192E-04	0.533E-04	0.138E-03	0.227E-03	0.250E-03	0.250E-03	0.250E-03
10	0.145E-04	0.403E-04	0.106E-03	0.179E-03	0.200E-03	0.200E-03	0.200E-03	0.200E-03	0.200E-03
11	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03
12	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03	0.100E-03
16	0.594E-04	0.175E-03	0.544E-03	0.135E-02	0.259E-02	0.298E-02	0.299E-02	0.299E-02	0.299E-02
19	0.490E-04	0.141E-03	0.412E-03	0.873E-03	0.123E-02	0.125E-02	0.125E-02	0.125E-02	0.125E-02
20	0.739E-04	0.215E-03	0.648E-03	0.148E-02	0.237E-02	0.249E-02	0.249E-02	0.249E-02	0.249E-02
21	0.248E-04	0.733E-04	0.232E-03	0.604E-03	0.129E-02	0.165E-02	0.166E-02	0.166E-02	0.166E-02

Fig. 16 - Program AVANA. Output Information.
Primary events unavailability.

MINIMAL CUT SETS UNAVAILABILITY

CUT SET INDEX	MISSION TIME (HOURS) 0.100E 01	0.300E 01	0.100E 02	0.300E 02	0.100E 03	0.300E 03	0.100E 04	0.300E 04	0.100E 05
1	0.369E-09	0.321E-08	0.319E-07	0.213E-06	0.102E-05	0.246E-05	0.394E-05	0.415E-05	0.415E-05
2	0.175E-08	0.148E-07	0.134E-06	0.709E-06	0.190E-05	0.224E-05	0.224E-05	0.224E-05	0.224E-05
3	0.730E-09	0.622E-08	0.574E-07	0.316E-06	0.952E-06	0.123E-05	0.125E-05	0.125E-05	0.125E-05
4	0.315E-09	0.275E-08	0.276E-07	0.188E-06	0.923E-06	0.198E-05	0.253E-05	0.255E-05	0.255E-05
5	0.196E-09	0.169E-08	0.163E-07	0.101E-06	0.434E-06	0.105E-05	0.183E-05	0.199E-05	0.199E-05
6	0.147E-09	0.128E-08	0.124E-07	0.791E-07	0.326E-06	0.567E-06	0.623E-06	0.623E-06	0.623E-06
7	0.107E-08	0.867E-08	0.683E-07	0.265E-06	0.474E-06	0.499E-06	0.499E-06	0.499E-06	0.499E-06
8	0.868E-14	0.218E-12	0.633E-11	0.898E-10	0.617E-09	0.158E-08	0.274E-08	0.298E-08	0.299E-08
9	0.294E-12	0.848E-12	0.247E-11	0.524E-11	0.736E-11	0.749E-11	0.749E-11	0.749E-11	0.749E-11
10	0.236E-14	0.613E-13	0.200E-11	0.382E-10	0.528E-09	0.261E-08	0.580E-08	0.636E-08	0.636E-08

Fig. 17 - Program AVANA. Output Information (cont.)
Minimal cut sets unavailability.

SYSTEM UNAVAILABILITY

MISSION TIME (HOURS)	UNAV. (UPPER BOUND)	UNAV. (SECOND BOUND)	UNAV. (THIRD BOUND)
0.100E 01	0.457E-08	0.457E-08	0.457E-08
0.300E 01	0.386E-07	0.386E-07	0.386E-07
0.100E 02	0.348E-06	0.348E-06	0.348E-06
0.300E 02	0.187E-05	0.187E-05	0.187E-05
0.100E 03	0.604E-05	0.604E-05	0.604E-05
0.300E 03	0.100E-04	0.100E-04	0.100E-04
0.100E 04	0.129E-04	0.129E-04	0.129E-04
0.300E 04	0.133E-04	0.133E-04	0.133E-04
0.100E 05	0.133E-04	0.133E-04	0.133E-04

Fig. 18 - Program AVANA. Output Information (cont.)
System unavailability.

SYSTEM UNAVAILABILITY

MISSION TIME (HOURS)	UNAV. (UPPER BOUND)	UNAV. (SECOND BOUND)	UNAV. (THIRD BOUND)
0.100E 01	0.472E-08	0.472E-08	0.472E-08
0.300E 01	0.425E-07	0.425E-07	0.425E-07
0.100E 02	0.472E-06	0.472E-06	0.472E-06
0.300E 02	0.424E-05	0.424E-05	0.424E-05
0.100E 03	0.470E-04	0.470E-04	0.470E-04
0.300E 03	0.420E-03	0.419E-03	0.419E-03
0.100E 04	0.455E-02	0.449E-02	0.449E-02
0.300E 04	0.381E-01	0.363E-01	0.364E-01
0.100E 05	0.332E 00	0.260E 00	0.271E 00

Fig. 19 - System unavailability in the case of no restoration (system unreliability).

NOTICE TO THE READER

All scientific and technical reports published by the Commission of the European Communities are announced in the monthly periodical "euro-abstracts". For subscription (1 year : B.Fr. 1 025,—) or free specimen copies please write to :

Office for Official Publications
of the European Communities
Case postale 1003
Luxembourg 1
(Grand-Duchy of Luxembourg)

To disseminate knowledge is to disseminate prosperity — I mean general prosperity and not individual riches — and with prosperity disappears the greater part of the evil which is our heritage from darker times.

Alfred Nobel

SALES OFFICES

The Office for Official Publications sells all documents published by the Commission of the European Communities at the addresses listed below, at the price given on cover. When ordering, specify clearly the exact reference and the title of the document.

UNITED KINGDOM

H.M. Stationery Office
P.O. Box 569
London S.E. 1 — Tel. 01-928 69 77, ext. 365

BELGIUM

Moniteur belge — Belgisch Staatsblad
Rue de Louvain, 40-42 — Leuvenseweg 40-42
1000 Bruxelles — 1000 Brussel. — Tel. 12 00 26
CCP 50-80 — Postgiro 50-80

Agency :
Librairie européenne — Europese Boekhandel
Rue de la Loi 244 — Wetstraat 244
1040 Bruxelles — 1040 Brussel

DENMARK

J.H. Schultz — Boghandel
Møntergade 19
DK 1116 København K — Tel. 14 11 95

FRANCE

Service de vente en France des publications des Communautés européennes — Journal officiel
26, rue Desaix — 75 732 Paris - Cédex 15e
Tel. (1) 306 51 00 — CCP Paris 23-96

GERMANY (FR)

Verlag Bundesanzeiger
5 Köln 1 — Postfach 108 006
Tel. (0221) 21 03 48
Telex: Anzeiger Bonn 08 882 595
Postscheckkonto 834 00 Köln

GRAND DUCHY OF LUXEMBOURG

Office for Official Publications of the European Communities
Case postale 1003 — Luxembourg
Tel. 4 79 41 — CCP 191-90
Compte courant bancaire : BIL 8-109/6003/200

IRELAND

Stationery Office — The Controller
Beggar's Bush
Dublin 4 — Tel. 6 54 01

ITALY

Libreria dello Stato
Piazza G. Verdi 10
00198 Roma — Tel. (6) 85 08
CCP 1/2640

NETHERLANDS

Staatsdrukkerij- en uitgeverijbedrijf
Christoffel Plantijnstraat
's-Gravenhage — Tel. (070) 81 45 11
Postgiro 42 53 00

UNITED STATES OF AMERICA

European Community Information Service
2100 M Street, N.W.
Suite 707
Washington, D.C. 20 037 — Tel. 296 51 31

SWITZERLAND

Librairie Payot
6, rue Grenus
1211 Genève — Tel. 31 89 50
CCP 12-236 Genève

SWEDEN

Librairie C.E. Fritze
2, Fredsgatan
Stockholm 16
Post Giro 193, Bank Giro 73/4015

SPAIN

Libreria Mundi-Prensa
Castello 37
Madrid 1 — Tel. 275 51 31

OTHER COUNTRIES

Office for Official Publications of the European Communities
Case postale 1003 — Luxembourg
Tel. 4 79 41 — CCP 191-90
Compte courant bancaire : BIL 8-109/6003/200

OFFICE FOR OFFICIAL PUBLICATIONS OF THE EUROPEAN COMMUNITIES
Case postale 1003 — Luxembourg 6070

CDNA04940ENC