

# COMPUTERAIDED RELIABILITY ANALYSIS

## SINTEF's EXPERIENCE

Per-Johan Fallrø & Marvin Rausand

SINTEF, Division of Safety and Reliability

A presentation at the SINTOM seminar

"Datorhjälpmedel för tillförlitlighetsanalys"

Visby, April 27 - 29, 1987

fr  
fr

### 1. BACKGROUND AND POLICY

SINTEF, Division of Safety and Reliability has, for a number of years been developing computer programs for reliability analysis. The first programs, for mainframe computers, were designed for in-house use to solve specific problems during execution of projects for our clients. The programs were efficient, but not very user friendly. The users needed computer experience to utilize all the features of the programs.

During 1982 and 1983, personal computers started invading our clients' office environment. This invasion was welcomed by our division. For a long time, SINTEF has been involved in the education of engineers at the Norwegian Institute of Technology (NTH), within reliability and risk analysis. Many of our students got central positions in the oil industry working with reliability assessment. Most of them had a personal computer on their desk.

We got fascinated by the possibility to develop computer programs for reliability analysis which were easy to use and easy to transfer between SINTEF and numerous clients with a wide range of different computer systems. The industry standard, set by the IBM PC, made this task a lot easier.

The programs which existed on our mainframe computers, were transferred to the personal computer environment. These programs were well tested and only small changes were needed to make them run on a personal computer. We then started working with the user interface.

Several objectives were held in mind while implementing the mainframe programs in the personal computer environment. The personal computer programs should:

- be easy to install
- be easy to get to know
- be easy to use
- have addressed help functions assigned to a 'hot-key'
- have an informative program execution dialogue
- have a stringent interface to standard peripherals
- have a shell or a framework to uniform the line of programs
- offer the experienced users a set of shortcuts
- have an interface to standard packages when meaningful
- be based on modularization with structured input/output formats

Along with these objectives, we have strived to find program development tools and programming techniques which keep the development costs and the size of the program code at a minimum.

Our division has been engaged in projects within component and systems reliability, man-machine interaction and safety problems. Our clients may roughly be identified as large Norwegian industry companies and oil companies operating on the Norwegian Continental Shelf.

These clients have shown an increasing interest in the area of reliability assessment. The offshore industry, in which the subsea production technology have been developed, has initiated an enormous demand for quantitative reliability assessment. This demand for reliability verification has been directed toward all the subcontractors and suppliers to the oil companies.

The production and control systems used in the oil industry today, are complex and constructed with a very high reliability in mind. The application of new system designs, new materials and extensive use of redundant system channels represents a challenge to our activities for optimizing safety and production regularity.

## 2. ANALYZING TECHNIQUES

In the following, some well known techniques for reliability analysis which are used by SINTEF, are described and briefly discussed. Most of these techniques are supported by computer programs developed at SINTEF. We have tried to sum up the limitations of each method which have been experienced while carrying out reliability analyses.

### Failure Mode, Effect and Criticality Analysis (FMECA).

A Failure Mode, Effect and Criticality Analysis (FMECA) is often the first step in a reliability study. An FMECA is easy to conduct. It does not require any advanced analytical skills. The analysis involves reviewing components, assemblies and subsystems to identify possible failure modes and their causes and effects. For each component, the failure modes and the associated effects on the system are written onto a specific FMECA form.

FMECA is a mainly a qualitative method. A semi-quantitative approach may, however, be obtained by assessing the frequency of the failure modes, and a criticality ranking of the consequences of the failure modes. The results of the analysis may be presented in a cross-table.

An FMECA may be very effective when applied on a system where system failures most likely are the results of single component failures. During the analysis, each failure is considered individually, as an independent occurrence with no relation to other failures in the system. An FMECA is thus not a suitable approach for analysis of systems with a fair degree of redundancy. For such systems, a fault tree analysis would be a much better approach.

A second limitation of FMECA is the inadequate attention generally given to human errors. This is mainly due to the concentration on hardware failures.

Perhaps the worst drawback, is that all component failures are examined and documented, including those which do not have any significant consequences. For large systems with a high degree of redundancy, the amount of unnecessary documentation work is a major disadvantage.

Computerizing the FMECA has significantly improved our efficiency. We are now able to make corrections in the FMECA forms and adjust the format without major retyping. We have established a rather comprehensive FMECA database and may easily copy parts of an existing FMECA into a new FMECA form. The presentation of the results in cross-tables have been greatly appreciated by the clients who are to review the analysis.

### Fault Tree Analysis (FTA)

Fault Tree Analysis (FTA) is, perhaps, the most commonly used technique for analyzing system reliability. The method generates, when used in a structured way, a systematic and good documentation of reasons for, or event sequences leading to a critical event in a system.

FTA may be used both as a qualitative and quantitative technique. The procedures for constructing the fault tree and performing the qualitative analysis are rather straightforward. Caution should, however, be taken when selecting algorithms for quantitative analysis such that they are suited to the system and the numeric input data to be analyzed.

There are a number of limitations regarding the applicability of Fault Tree Analysis. A fault tree presents a static picture of the component and subsystem states leading to the undesired Top event. FTA is therefore not a suitable technique for analyzing systems with dynamical properties. Among such systems are:

- Systems with complicated testing and maintenance.
- Standby and switching systems.
- Systems with phased missions.

Most of the quantitative FTA programs on the market presuppose that all the basic events of the fault tree are statistically independent. Analysis of common mode failures is not easily attained. Most programs taking common mode failures into account, utilize techniques similar to the Beta-fraction technique.

Systems with a significant hazard potential will normally have subsystems which are subject to some sort of periodic testing. Such testing is normally carried out simultaneously on complex assemblies. Analysis of such systems by FTA is not straightforward. The common practice is to calculate the mean fractional downtime (MFDT) for each basic event and use these MFDTs as independent input data in the analysis. This will give a non-conservative estimate of the Top event probability, even if no common mode failure possibilities are present. Experience has shown that common mode failures are rather frequent, at least for periodically tested redundant safety systems, which often are present in systems analyzed by FTA.

Human errors are very difficult to model and analyze by a fault tree. In FTA the operator is often seen to be modelled as a single basic event with a constant, time independent demand probability. This is, at best, a very rough approximation. The operator as a safety potential, able to correct disturbances, is normally forgotten in the analysis.

The computer programs for FTA should first of all generate a fault tree which is easy to read and understand. Space must be allocated so that comments to the way the tree is constructed, is easily implemented. The algorithms to be used for quantitative analysis must be selected according to scope of work and the object of the analysis.

#### Event Tree Analysis (ETA)/Cause Consequence Diagrams (CCD)

An Event Tree Analysis (ETA) is an inductive technique which is carried out to identify all the possible chains of events or consequences following a potentially dangerous event or hazard. The initiating event may be identified from an FMECA and the quantitative reliability characteristics is often derived from a FTA. The analysis ends up describing the specter of consequences.

The Cause Consequence Diagram technique is very similar to the ETA technique. The graphical symbols are, however, rather different. At SINTEF we prefer to use CC-diagrams since these diagrams are easier to read than event trees. There is a logical connection between the CC-diagram and fault trees. Fault trees may be linked to the No-gates in the CC-diagram for modelling and analyzing barrier failures.

Quantitative analysis of both event trees and CC-diagrams may be very complicated especially with repeated and dependent events, and when the tree contains loops.

The computer programs should encourage the user to establish structured and descriptive event trees and/or CC-diagrams. If a quantitative analysis is carried out, one should be careful with respect to event trees where loops occur.

### Markov methods

Markov analysis is often used to analyze dynamic systems. Each component is said to be in one of several different states. The components may be defined to function perfectly, they may be degraded within certain limits, they may be subject to repair or maintenance or they may be in one of a number of failure modes. The state of the system is defined by specifying the state of each component.

The Markov analysis will assess the stationary probabilities for each state of the system, in particular giving the probabilities for the system to function, being failed, being maintained or being repaired. The probabilities may also be defined as a function of time.

Markov methods are very useful when analyzing all sorts of standby and redundant systems and systems having complex repair routines. Systems possessing non-independence of the components may also be successfully analyzed by Markov methods.

The limitations of Markov methods may be summed up as follows:

- Markov analysis is based on the assumption that all life times and repair times are exponentially distributed. Thus, the occurrence of any event is determined by the **present** state of the system only. Semi-Markov method may partly compensate for this limitation.
- Markov methods are not suitable for analyzing systems which are subjected to periodic testing and repair.
- The time to compute the results may be excessive even for moderately complex systems.

A program for Markov analysis must allow the user to identify each state and event by means of descriptive text. A Markov diagram where the states and events are identified by a code only, is very hard to review. A computer program is, however, needed even for small Markov diagrams due to the large amount of calculations to be carried out.

### Reliability simulation

Reliability simulation starts with defining the system's operational strategy, the failure modes, the repair action, the intervention strategy, the test routines and the maintenance to be carried out. Probability distributions are then assigned to each variable. All the information is used as input to a program which simulates the behavior of the system over a time period. By repeating the simulation over the same period, a number of times, a set of mean values for production regularity, the occurrence of failures, the number of interventions, etc., are established.

The method is very time consuming for large and complex systems, but reliability simulation is often used when the more traditional methods ends up being inadequate.

## Network methods/Flow diagrams

Certain network methods, such as the Petri net (/1/), may be used to model the occurrence of events and activities in a system. The technique is very useful when failures in a system are triggered by certain events. The method have been used to evaluate the production regularity of a gas and oil production system. In general, network models are very useful whenever connectivity is an important factor in the reliability analysis.

Flow diagrams have proved themselves to be useful when throughput is the essential parameter. The method starts with modelling the system as a directed graph with a source (the input node), a sink (the output node) and nodes. Failure rates and capacity measures are then assigned to each arc.

By evaluating the flow diagrams, bottlenecks may be identified as a function of failure rates and the throughput may be optimized with respect to failure rates and capacities.

In general, network models and flow diagrams are used when transport systems, such as pipelines, power supply systems and telegraph systems, are to be evaluated.

The existence of computer programs is strongly appreciated when numerous calculations are to be carried out on large networks and flow diagrams whereas simple and small systems may be analyzed by hand.

- fr -  
\* \*

## PRACTICAL APPLICATIONS

None of the techniques listed above is suitable for analyzing all types of systems. For practical applications one has to utilize a rather complete toolbox containing a number of the techniques listed above. One may, as an example, find it beneficial to analyze the top level of a system by fault trees. Particular subsystems may require a more detailed analysis e.g. by Markov methods. Other subsystems may require reliability simulation or flow diagrams. The output from all these methods must then be combined to produce final results on a system level.

One should not try to turn all sizes of nuts and bolts with a fixed wrench, and - at least according to our opinion, a toolbox with a suitable set of fixed wrenches is far better than a monkey wrench.

### 3. WHAT IS GAINED BY COMPUTERAIDED RELIABILITY ANALYSIS?

SINTEF has carried out a high number of reliability studies during the last 10 - 15 years. The majority of these have been studies of offshore production and safety systems. Most of the methods we use are very systematic and involve a large amount of information. We have been papering walls with fault trees and large structure equations to arrange the information, hours have been spent recalculating to check for errors, and valuable time has been spent twisting information to make it fit into predefined FMECA forms. Such activities reduced our efficiency and were not appreciated with respect to quality assurance. The objectives for developing computer codes have been:

- i) To improve our quality assurance
- ii) To execute our projects in a more efficient way, which is required due to the increasing amount of information to be handled in such projects
- iii) To take part in the development of reliability analysis techniques which require efficient computing due to the amount of information or the amount of calculations

In addition, computers were needed to keep track on the large amount of failure event reports and reliability data collected by the Division of Safety and Reliability.

With respect to quality assurance, we believe lots have been gained. Today, we are able to produce documentation describing the algorithms and the data used in the calculations in a rapid and efficient way. Clients may also get copies of the raw output data and are thus enabled to evaluate our interpretation of the results. In this way, misunderstandings are avoided, errors are easily detected and corrected, and the clients seem to get a better understanding of the methods used. All this is important for the client's confidence in the results derived by SINTEF as contractor.

Efficiency is vital considering the way man-hour rates for engineers have been escalating during the last years. It gets more and more important that the workers are provided with useful and efficient tools. The potential benefits from using efficient tools have, perhaps, never been greater than today.

SINTEF has experienced that the amount of tedious organization of information, the retrieval of the same, manual calculations and recalculations have been reduced enormously. This is, however, not the major contribution to improved efficiency as we see it. We are today able to establish analytical models which take more questions into account and look more closely at 'what if' situations. This expansion of our projects' scope of work would not be possible without expanding the budgets of the projects, if it hadn't been for our efficient computer programs.

SINTEF/NTH has recently installed a Cray computer for large scale computing. This super-computer will, along with the rest of our line of computers, enable us to advance into new and more complex applications for reliability analysis. We have yet not started developing programs for the Cray computer, but we have started looking at new methods for analyzing large data sets on the new computer.

#### 4. SINTEF's PROGRAM PORTFOLIO AND PLANS FOR THE FUTURE

SINTEF, Division of Safety and Reliability does not make a living out of selling computer codes. The programs have emerged from our needs while running projects for clients. We usually establish the program specifications on our own. We believe that a program specification should be based on thorough experience with the analytical techniques. In this way, we hope to develop programs which meet the needs in an efficient way and to reduce the amount of resources spent on implementing nice looking but not very needed features.

The following techniques of reliability analysis are now supported by computer programs developed at SINTEF:

- Reliability database structuring and analysis
- Life- and Repair-time Data Analysis (SAREPTA)
- Failure Mode, Effect and Criticality Analysis (FMECA)
- Fault Tree Analysis (CAFTAN)
- Markov Analysis
- Reliability simulation (SUBSIM)
- Analysis of periodically tested units subject to common mode failures
- A number of specialized programs e.g. safety barrier configurations

All the programs run on an IBM AT with 640 kB of internal memory (RAM), a harddisk, a co-processor and an enhanced graphics (EGA) monitor. The CAFTAN computer code is also running on mainframe computers (Norsk Data and VAX). A brief description of SAREPTA and CAFTAN is given at the end of the paper.

#### SINTEF's plans for the future

At SINTEF, we will continue our development of user-friendly and efficient computerized reliability techniques. We will, at least for the first few years, concentrate our development to programs for the IBM AT computer, but have a close eye on possible new industry standards. We will not try to develop large integrated program packages, because we doubt that such integrated packages can obtain the required user-friendliness and efficiency. Instead, we will develop separate modules with a common user interface and, as far as possible, a common structure.

We feel that the user dialogue in CAFTAN very good, and will therefore base the development of a new set programs for Cause-Consequence Diagrams and Event Tree Analysis on the same concept. The quantitative part of CAFTAN will, however, be improved mainly according to the ideas in chapter 2 of this paper. We will also make it possible to isolate modules of the fault tree which need to be analyzed by other techniques, like Markov methods or our program for periodic testing. These modules may then be "taken out" of the fault tree and subjected to special analyses before they are re-entered into the fault tree for overall calculations.

We will further consider the possibility of establishing a "communication" between our FMECA program and CAFTAN, e.g. creating a pop-up window in CAFTAN making it possible to search in the FMECA database to get ideas for the fault tree construction.

Fault tree analysis is an easy to learn and often efficient technique of reliability analysis. During the construction of the fault tree, the analyst is forced to work systematically to identify failure causes. System weaknesses are often revealed and corrected during the analysis.

At SINTEF we are therefore a bit negative to introduce automatic fault tree construction in our systems. We are afraid that we in this way will loose one of the prime effects of a fault tree analysis - the detailed asking "How can this failure occur?".

We are also a bit negative to linking the fault tree analysis program to a reliability database with automatic transfer of data to the fault tree. Reliability data from different sources normally show a significant scatter. The database should reflect this scatter and the user of the data should carefully try to evaluate the data and adapt the estimates before he incorporate them into a fault tree.

Our SAREPTA code is very good and we cannot see any reason for trying to improve it. We are, however, developing a new module for analysis of proportional hazards models and Cox regression. It has yet not been decided whether this module should be an integrated part of SAREPTA or not.

fr  
"

## 5. CAFTAN - Computeraided Fault Tree Analysis

CAFTAN is an integrated fault tree analysis program containing a number of separate modules. The fault tree construction part has a user interface as shown in figure 1. The construction of the fault tree is done page by page linked together by transfer symbols. The construction "mode" is selected from the rightmost menu by the function keys, F1,...,F10 on the keyboard. F9 changes menus. It is also possible to select "mode" from the menu by using the cursor. The fault tree symbols are selected either by using the cursor or by using the "TAB" key on the keyboard. The various fault tree symbols are drawn and organized on the left part of the screen. This is accomplished by moving the cursor, either by the cursor control keys or by a "mouse". The graphical output of CAFTAN is supported by subroutines from the IBM Graphical Kernel System (GKS).

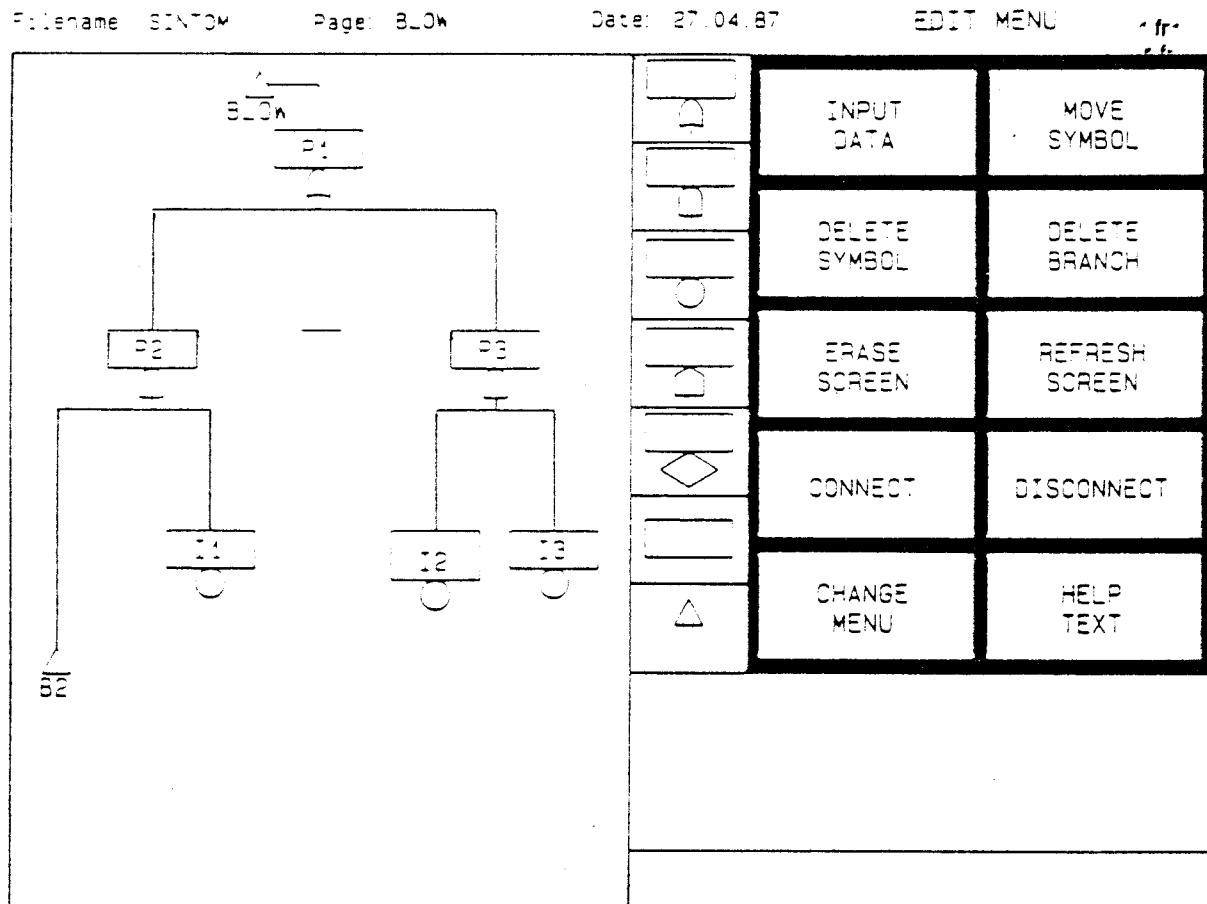


Figure 1 CAFTAN's user interface

The user may import fault tree pages from any previously constructed fault trees. He may establish "library" of fault trees for standard systems. Pages from the "library" may then be imported when needed by giving reference to the "library" name and the page name. An overview of the page names of a fault tree is implemented both as a listing and in graphical mode. When a graphical overview of the fault tree is presented, the user may zoom in on parts of the fault tree exactly as he wants to. This function is very flexible and useful and the resulting overview may be hardcopied on a plotter.

When a fault tree is established, the same basic events usually occur in several branches of the tree. Repeated events must be given the same identifier (4 letter code) during the construction of the tree. The event description and the input reliability data have, however, only to be entered once. The text and the reliability data are duplicated whenever a basic event is established with an existing identifier. This makes the input section more efficient and reduces the risk of erroneous input due to typing errors.

The tree can hold a maximum of 2000 different basic events, which should be sufficient for most practical applications. The fault tree may be plotted on e.g. a standard HP-plotter in A4 or A3 format.

A new algorithm for generating minimal cut sets, called PC-CUT, has recently been developed at SINTEF, Division of Safety and Reliability. The algorithm is programmed in IBM Professional FORTRAN and is based on the same ideas as the well known MOCUS algorithm, but is far more efficient. A thorough benchmark testing has not been carried out, but in all our testruns, the PC-CUT algorithm has been at least five times faster than MOCUS.

A simple program for quantitative analysis has been developed as an integrated part of the CAFTAN code. This code is comparable to SUPERPOCUS, but is more efficient. A considerable effort is now being put on improving the quantitative analysis code to cope with problems like: periodic testing, phased mission, complicated repair actions and common mode failures.

A module for uncertainty analysis based on Monte Carlo simulation comparable to the well known SAMPLE code will soon be implemented. This code is based on a more "intelligent" simulation. The simulation is carried out in a prioritized sequence based on a combination of the reliability importance (Vesely-Fussell) of the basic event and the uncertainty of the input data.

#### Mainframe version

The mainframe version of CAFTAN has approximately the same user interface as the PC version described above. This version uses MOCUS for generating minimal cut sets, IMPORTANCE for calculating reliability importance measures and approximative Top event probabilities. KITT 1 (and 2) is further be used for more detailed quantitative analyses. SAMPLE may be used for uncertainty analyses.

## 6. SAREPTA - Survival And Repair Time Analysis

SAREPTA is a program for analyzing life data and repair time data. The program accepts multiple censored data with a high number of failure modes. SAREPTA integrates a number of modules which may be selected from menus operated by the function keys F1,...,F10 on the keyboard. SAREPTA has a very good user interface and may be run without detailed knowledge on reliability technology. The output from the program is, however, rather advanced. The interpretation of the output should therefore be left to skilled reliability analysts.

SAREPTA is programmed in IBM Professional FORTRAN and is supported with graphical output subroutines from IBM Graphical Kernel System (GKS).

### Input section

Data may be entered from the keyboard or imported from an ordinary textfile. It is for example easy to use data from a dBASE III file. Before starting to enter data, one has to specify the actual failure modes and identify them with a three-letter code. The computer will "beep" if you later try to include data with unspecified failure mode.

When you have entered the data, it is easy to edit the data, rename failure modes etc.

### Parameter estimation

The parameters of the following distributions are rapidly estimated:

- Exponential distribution
- Weibull distribution
- Gamma distribution
- Lognormal distribution
- Normal distribution
- Birnbaum-Saunders distribution
- Extreme Value (Gumbel) distribution

In our selection or estimation procedures, efforts have been put on finding efficient procedures accepting multiple censored data.

### Plotting section

The analysis may be carried out with respect to each individual failure mode or an arbitrary specified subset of the failure modes. The program may thus be used for analyzing most types of competing risk models.

The following plots may be supplied:

- Kaplan-Meier plot (/2/)
- Hazard plot (/3/)
- Total Time on Test (TTT) plot (/4/)
- Failure rate plot (histogram)

Examples of the output from the Kaplan-Meier, the hazard and the TTT-plotting are given in figures 2 - 4 respectively.

Overlay curves for the Kaplan-Meier plot may be selected from a pop-up menu. The overlay curves may be used to visualize the goodness of fit to the various distributions. The TTT-plot is established on the basis of multiple censored data by combining the Kaplan-Meier estimate with the TTT-transform. TTT-transform overlay curves may be selected from a pop-up menu to visualize the goodness of fit. The TTT-plot may also be used to determine optimal strategies for replacement of components according to a technique proposed by Bo Bergman (/5/).

### Goodness of fit

In addition to the techniques with overlay curves described above, SAREPTA also includes a formal goodness of fit test. This test was proposed by Barlow and Proschan /6/ and is one of the very few relevant formal tests accepting censored data.

### Repair time plotting

Repair times are commonly assumed to be governed by a lognormal distribution. This section provides a plot of the cumulative distribution function together with a lognormal overlay curve to check the assumption.

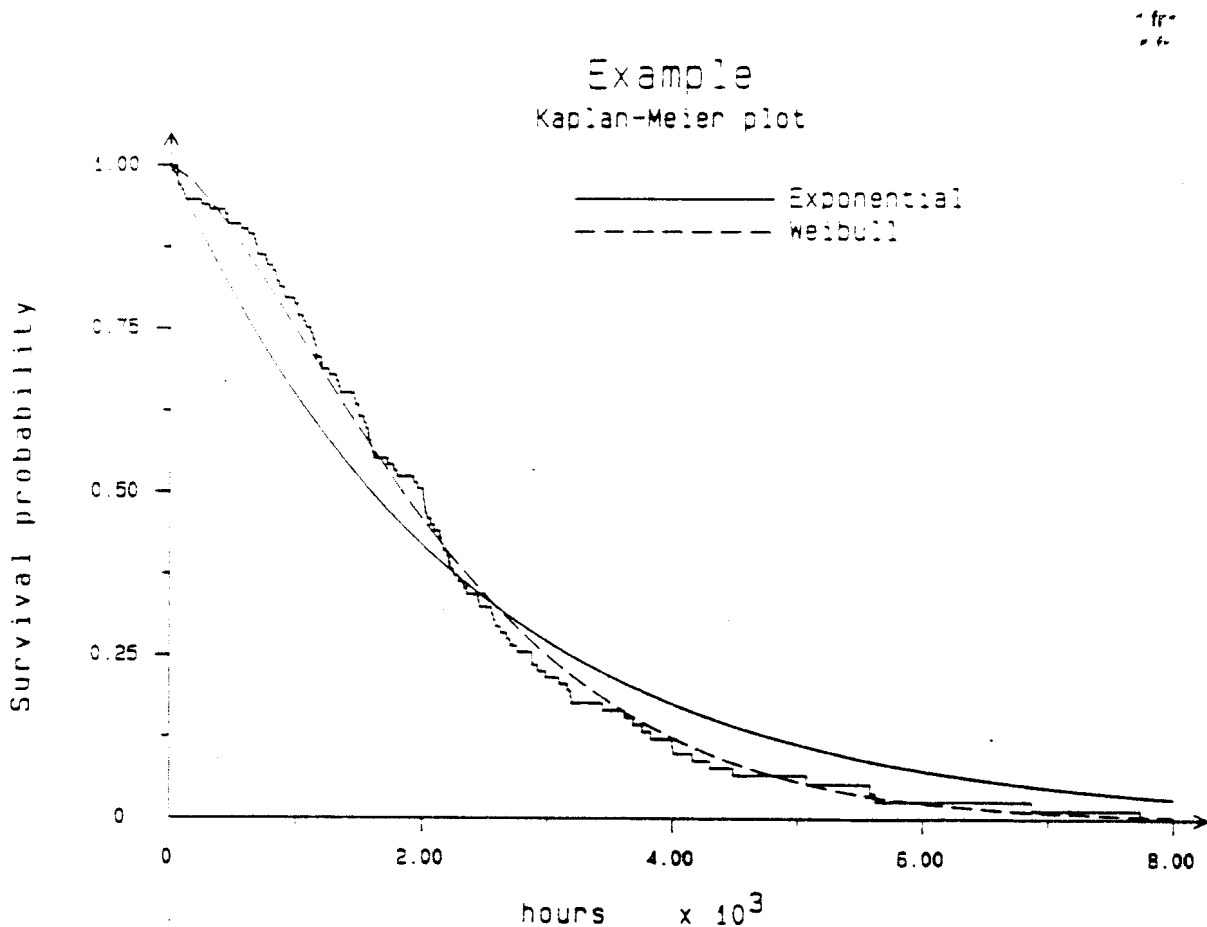
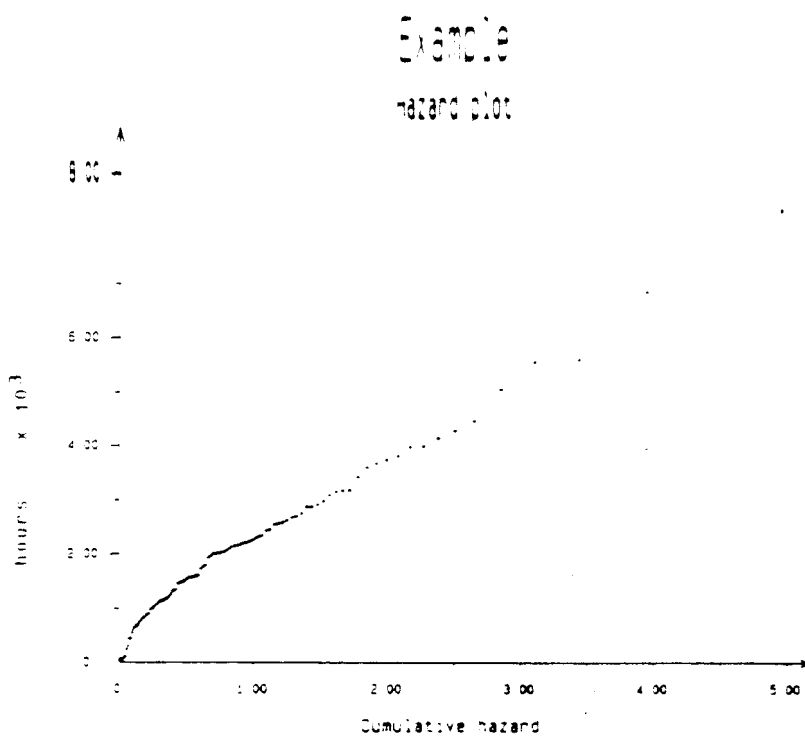


Figure 2 Example of Kaplan-Meier plot from SAREPTA



fr  
r

Figure 3 Example of hazard plot from SAREPTA

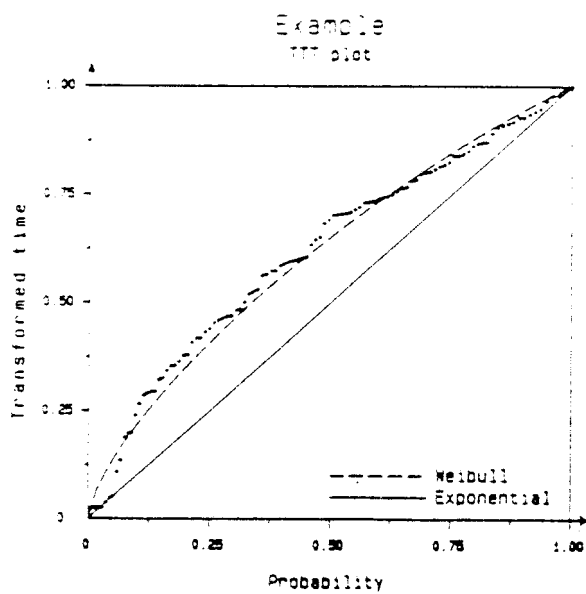


Figure 4 Example of TTT-plot from SAREPTA

## 7. REFERENCES

- /1/ **Peterson, James I.:** "Petri net theory and modeling of systems", Prentice-Hall, 1981.
- /2/ **Kaplan, E. L. & Meier, P.:** "Nonparametric estimation for incomplete observations", American Statistical Associated Journal, june 1958, pp 457-481.
- /3/ **Nelson, W.:** "Applied Life Data Analysis", John Wiley, 1982.
- /4/ **Barlow, R. & Campo, R.:** "Total Time on Test Processes and Applications to Failure Data Analysis", Reliability and Fault Tree Analysis, SIAM 1975, pp 451-481.
- /5/ **Bergman, B.:** "On age replacement and the total time on test concept", Lund Institute of Technology, Department of Matematical Statistics, 1978:4.
- /6/ **Barlow, R. & Proschan, F.:** "A Note on Tests for Monotone Failure Rate Based on Incomplete Data", Annals of Matematical Statistics, 1969, Vol 40, pp 595-600.

