

A Fourth Generation Reliability Predictor

Salvatore J. Bavuso; NASA Langley Research Center; Hampton

Anna L. Martensen; PRC Kentron Inc.; Hampton

Key Words: Fault Tolerance, HARP (Hybrid Automated Reliability Prediction), Reliability Estimation Fault Tree, Coverage, R&M Prediction.

Summary

A novel reliability/availability predictor computer program has been developed and is currently being beta tested by over 30 U.S. companies. The computer program is called the Hybrid Automated Reliability Predictor (HARP). The HARP capability was developed to fill an important gap in reliability assessment capabilities. This gap was manifested through the use of its third generation cousin, the Computer-Aided Reliability Estimation (CARE III) program, over a six-year development period and an additional three-year period during which CARE III has been in the public domain. With over 30 establishments now using CARE III, a rich body of experience has been accumulated and exploited in the development of the HARP program.

The CARE III program overcomes two major stumbling blocks in the reliability modeling of practical highly reliable systems: 1) the problem of describing the fault-handling mechanisms of a complex system, and 2) the problem of arriving at a numerical solution for the resulting reliability model. Typically, the Markov model for these systems is very large (1000+ states), and the fault-handling models are a major contributor to the state size explosion. The CARE III program is able to model many of these complex systems; however, due to a reduction in modeling flexibility, some system descriptions cannot be captured.

The HARP capability was developed to take advantage of CARE III's ability to model complex fault-handling mechanisms without incurring the penalty of introducing an enormous state space. At the same time, HARP maintains the complete modeling flexibility of the Markov modeling technique. After one year of beta testing HARP, it has become apparent that HARP has important limitations that at this time are being worked at a feverish pace.

After a brief introduction to the CARE III and HARP capabilities, this paper will address the strengths and weaknesses of each program. HARP limitations are presented and some solutions currently being implemented are discussed. In addition, some example problems are given to demonstrate HARP's capability.

Introduction

The design and implementation of reliability prediction computer programs for modeling highly reliable fault-tolerant digital computer-based systems is a slow and complex task. With the technology for developing digital fault-tolerant systems still in its infancy, the model designer faces a predicament. He must capture not only current modeling needs but he must also be able to anticipate future needs as well. Constraints are also imposed by available mathematical techniques to solve complex stochastic equations and by the large amount of computer code that is required to implement their solutions. As a result, reliability modeling programs are continually being evolved to meet the increasing complexity of emerging systems and to reflect our increasing understanding of the fault-free and faulted behavior of digital fault-tolerant systems.

Experience has shown that there is a practical limit to the degree of modification that can be effected

on a particular program; so the evolutionary process tends to generate new programs. Unfortunately, thus far it has not been possible to consolidate the models within these programs under one set of powerful mathematics, and therefore, a proliferation of programs has resulted. The positive aspect of this situation is that the independent development of reliability models that often use different mathematical techniques enhances our ability to validate them. Since many reliability evaluators have some overlapping capability, confidence in the validity of the programs can be derived by comparative analysis. This technique is by far the most practical validation technique at our disposal. The most powerful validation technique, however, is the tedious hand-derived exact solution.

CARE III - A Third Generation Reliability Predictor

In 1978, NASA at the Langley Research Center undertook the challenge of developing a powerful general purpose reliability evaluator (Ref. 1). The reliability evaluator was designed to be applicable to current and futuristic highly reliable fault-tolerant digital computer-based systems. Although few such systems existed in 1978, researchers had accumulated a vast amount of information on the predominant causes of system failure. Some of the more important characteristics of the evaluator were to include fault/error-handling parameters such as latent and near-coincident faults/errors and to enable the user to capture the effects of system redundancy management strategies associated with the handling of the faults (Fig. 1). The model was to cover transient and intermittent fault occurrences and to give the user the option of modeling exponential or non-exponential (Weibull) times to failure. One other feature that was recognized as paramount to the successful use of the evaluator was the ability to model very large systems, because it was predicted that systems would grow in size and complexity in the near future.

The use of the stochastic Markov process to model system reliability was well known at that time. It was also known that the inclusion of fault-handling models would cause an intractable explosion in the Markovian state space for all but the simplest models. Models of general interest that included fault-handling expanded the state space to typically thousands of states. A mathematical technique, now called behavioral decomposition, essentially solved the exploding state space problem (Ref. 2). In the behavioral decomposition technique, it is recognized that events associated with fault-handling occur on the order of fractions of a second to a few seconds, while events associated with component failures occur on the order of thousands of hours. The direct ordinary differential equation solution to systems with widely separated time constants is often intractable or, at best, subject to extreme round-off and/or truncation error.¹ However, the widely separated time difference of typically six orders of

1. A model with widely separated time constants is said to be "stiff," especially if at least one state has competing exit transitions with greatly different rates.

U.S. Government work not protected by U.S. Copyright.

magnitude can be exploited mathematically to treat the solution of the fault-handling model independently from the fault-occurrence model. CARE III was the first model to employ this technique in earnest.

CARE III first manipulates the fault-handling information and inserts that information into the fault occurrence model (Ref. 3). Because the user can specify non-exponential transitions within the fault-handling model, the fault-handling solution is semi-markov. Once this information is added to the fault occurrence model, the final model to be solved is nonhomogeneous. Since a nonhomogeneous model must inherently be solved, there is no additional cost in allowing Weibull times to failure for component devices. Therefore the user, instead of being limited to exponential times to failure, may accurately model component failure due to wear-out. Conversely, the user may model a system composed of components subject to mechanical hardening (decreasing rate of failure). Systems with no fault handling may be described; their solutions are easily calculated with the CARE III program.

In order to describe a potentially large state space, the compact fault tree notation was selected and implemented in CARE III. The power of this notation can be easily appreciated by recognizing that a fault tree that can be comfortably drawn on an eight and a half by eleven sheet of paper can generate over 25,000 Markovian states and over 300,000 transitions, and this model only accounts for fault occurrences. To model fault-handling, CARE III uses a notation similar to that of a typical Markov model and has a maximum of seven unique states. Many variations of this model are possible by judiciously choosing transition rate parameters.

The fault tree capability eliminated the tedious enumeration of a combinatorial model; however, it also introduced certain inherent limitations. Because fault trees are used, only combinatorial Markov states can be modeled and, therefore, models which include sequence dependent state transitions cannot be accurately described. A combinatorial state makes no distinction on how the process arrives at a state, i.e., all paths to a combinatorial state are possible. Sequence dependent state transitions specify less than all possible paths to a state. From an applications point of view, sequence dependency means that a given failure event can occur only if some other failure event occurs first; i.e., the order of failure is important. By ignoring sequence dependence, careful modeling can often produce a conservative result that is often acceptable. Also for highly reliable fault-tolerant systems, the lack of adequate fault handling is often the predominant contribution to system failure anyway.

The other major CARE III limitation is its inability to model cold or dormant spares. This limitation is a direct result of the nonhomogeneous model that CARE III solves for fault occurrences. A nonhomogeneous process allows time varying transition rates (Weibull) between Markov states. By definition, a cold spare has no probability of failing until it is powered up. A characteristic of time varying transition rates is that the component rate of failure is measured by a global clock that begins ticking at mission time zero. Therefore the component could not have a zero failure probability at some later mission time. This property also precludes the modeling of partially powered or dormant spares. Of course, this limitation does not apply to powered spares which is a CARE III capability.

Two additional deficiencies worth mentioning are its inability to compute instantaneous availability, and fault handling is not state dependent but is device dependent. The user can define up to five different fault-handling models. These model versions may be assigned to any or all devices, but once assigned, the parameters within the models don't change as a function of the number of failed devices remaining in the system (no state dependence is allowed). These major limitations prompted NASA Langley to seek the development of a

more general reliability evaluator that retained the best features of CARE III but without loss of the generality provided by the classical Markov method. That search led to the development of the Hybrid Automated Reliability Predictor (HARP). (See Fig. 2).

HARP - A Fourth Generation Reliability Predictor

The HARP model provides five innovations to CARE III (Ref. 4): The behavioral decomposition mathematical technique was simplified so that a nonhomogeneous Markov chain is not required for solution. Instead, all fault-handling information is collapsed into four probabilities (1) probability of a near-coincident fault, (2) probability of a single point failure, (3) probability of transient recovery, and (4) probability of successful recovery from the fault, and these probabilities are used to modify the exponential transitions from the fault-occurrence states. This simplification was proved to produce conservative results and allows the solution of a system model to be homogeneous (exponential times to failure). Weibull fault occurrences are still permitted, but they are optional and are invoked by the user.

HARP always solves a general Markov chain allowing sequence dependency, state dependent fault handling, cold and dormant spares, and instantaneous availability computations.

HARP allows the user to choose between entering a Markov model directly by specifying state identification and transition rates, or by specifying a Markov chain using fault tree notation. The former notation allows complete flexibility in model design but can become cumbersome or impossible to use for anything but small models. The latter notation makes HARP user-friendly for large systems because the user can initially describe his model in the fault tree notation. HARP in turn will proceed to convert the fault tree into an equivalent Markov chain but with the appropriate fault-handling model data included. At this point, the user may modify the Markov chain directly to include sequence dependency, cold spares, or to include renewal paths to model availability. This capability makes it entirely possible to model very large systems without losing the flexibility of the Markov chain. The use of this capability is depicted by Figs. 3-5 and for the Advanced Reconfigurable Computer System (ARCS) and Figs. 6 and 7 for a fault-tolerant jet engine controller. The ARCS model generated over 600 fault occurrence states while the jet engine controller model produced over 24,000 states. (The details of these models can be found in Ref. 5.) The value of the behavioral decomposition technique can be fully appreciated when models of this complexity are of interest. Without the technique, fault-handling states would have to be added to the already very large state size making the solution of these two model impractical if not impossible.

HARP has an expanded set of six fault-handling models to include simple histogram models for modeling experimental data and a detailed petri net model that is solved by Monte Carlo simulation.

A computer-aided reliability engineering version of HARP has been written and is currently undergoing beta testing. This capability is called the IBM PC version and is configured as a work station for reliability assessment. The PC version provides the user a graphical input capability. With a mouse or through keyboard entry, the user can draw fault trees or Markov chain graphs that are automatically converted into the same file structure as the mini-computer version. Thus, a user can interact with an inexpensive IBM PC or IBM compatible to create his reliability model files. Then, depending on the size of the Markov chain, the solution can be computed on the PC or a mini-computer. Aside from the fact that PC's or compatibles are ubiquitous, another important attribute of this version is that the user can input his fault-occurrence and fault-handling

models directly as images on the screen of the PC monitor. The compact fault tree notation that can easily be drawn on the screen can be used by HARP to generate Markov chains of enormous size. Fault/error-handling models are also manipulated graphically for user definition.

HARP IV+

With over 20 beta test sites exercising HARP for over a year, some weaknesses have been uncovered. Fortunately, the structure of the HARP code and the general Markov model lend themselves well to modifications which eliminate the weaknesses and allow further enhancement in capability. This process has been ongoing since the spring of 1987. The most serious weakness was long execution times for moderately large systems. A jet engine controller design, the largest system thus far presented to HARP, produced a Markov chain of 24,533 states and over 300,000 transitions (Ref. 5). This model was produced by a fault tree representation but also included fault-handling. Because of the behavioral decomposition technique used by HARP, only one state was added to account for single point failures. Careful analysis of HARP module execution times revealed the source of the computation intensive module.

The problem occurred because of a fundamental assumption that the HARP designers made early in the HARP design. It was assumed that users typically would enter a Markov chain to HARP as the preferred input. Based on experience with Markov chains, the designers knew that users would not typically order states in a numerical sequential manner. State ordering is essential however, in order to properly fill the transition matrix for numerical solution. HARP therefore automatically invoked a sorting subroutine to properly order the states.

After a year of HARP user experience, it was discovered that typically users initially invoke the fault tree input notation to generate the Markov chain. Since this process already orders the states, the automatic ordering algorithm was altered to become a user selected process. Small Markov chain models that would be entered directly as a Markov chain could be ordered automatically by user request. This simple change reduced the computation time of the jet engine controller model by a factor of seven.

Another HARP weakness became evident after several months of use. It was assumed that all system descriptions resulting in a stiff model would be decomposed into the fault-occurrence and fault-handling models. The aggregated model which results from behavioral decomposition, then, would be non-stiff. Under this assumption, the solution package chosen to solve the models was optimized for non-stiff, nonhomogeneous models. However, when entering a Markov model directly, it is possible to disregard behavioral decomposition and describe a system with widely separated transition rates. For stiff problems, the numeric solver is inefficient, and excessive run times may result. It should be noted that for some systems the fault-handling behavior cannot be described apart from the fault-occurrence behavior. For these systems, the resultant system description is necessarily stiff. In order to accommodate stiff models, the inclusion of at least one new ordinary differential equation (ODE) solver is required (Ref. 6). Recent research has shown that a combination of ODE solvers works better than any known solver by itself. In general, the model will be examined for stiffness and an appropriate ODE solver will then be selected from a "pool" of solvers. This process will of course be transparent to the user. A prototype version of HARP incorporating this technique should be implemented by the spring of 1988.

Concluding Remarks

This paper has introduced the reader to a fourth generation reliability predictor computer program. Beginning with a description of the motivation that produced the CARE III program, a third generation reliability predictor, the motivation for the development of the fourth generation reliability predictor program was subsequently described. Few designers of such complex programs can anticipate the outcome of the early decisions that shape the eventual implementation. Therefore it is not surprising that these complex programs are continually being upgraded as they are used by a greater number of people.

In order to give the reader a flavor of that upgrading process, two major HARP weaknesses and their solutions were described.

References

1. S. J. Bavuso, "A User's View of CARE III," Proc. Annual Reliability and Maintainability Symp., Jan. 1984, pp. 382-389.
2. J. J. Stiffler and L. A. Bryant, "CARE III Phase III Report - Mathematical Description," NASA Contractor Report 3566, Nov. 1982.
3. S. J. Bavuso and P. L. Petersen, "CARE III Model Overview and User's Guide," (first revision), NASA Technical Memorandum 86404, April 1985.
4. T. S. Kishor and J. B. Dugan, "Hybrid Reliability Modeling of Fault-Tolerant Computer Systems," Computers and Electrical Engineering, vol. 11, no. 2/3, pp. 87-108, 1984.
5. S. J. Bavuso, J. B. Dugan, K. S. Trivedi, B. Rothman, and M. Boyd, "Applications of the Hybrid Automated Reliability Predictor," NASA Technical Paper 2760, November 1987.
6. Andrew Reibman and Kishor Trivedi, "Numerical Transient Analysis of Markov Models," Dept. of Computer Sci., Duke Univ., Durham, NC, Feb. 1987.

Biographies

Salvatore J. Bavuso
NASA Langley Research Center
Hampton, Virginia 23665-5225 USA

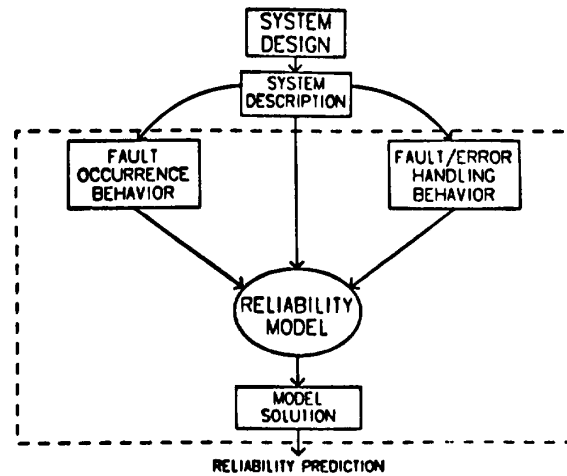
Mr. Bavuso is a Senior Researcher at NASA Langley Research Center in Hampton, Virginia. He received the B.S. degree in Mathematics from the Florida State University in 1964 and the M.S. degree in applied Mathematics from the North Carolina State University at Raleigh in 1971. He is the NASA project engineer for the development of CARE III and HARP.

Anna L. Martensen
PRC Kentron, Inc.
Hampton, Virginia 23665 USA

Anna L. Martensen received the B.S. degree in Industrial Engineering from Texas Tech University in 1984. She began work with PRC Kentron, Inc. in 1985 as a Reliability Engineer, and has concentrated on the development and enhancement of reliability analysis tools for ultra-reliable systems. Ms. Martensen is a member of Tau Beta Pi, Alpha Pi Mu, and the Institute of Industrial Engineers.

CARE III

(COMPUTER AIDED RELIABILITY ESTIMATION)



- 0 GENERAL-PURPOSE RELIABILITY ANALYSIS AND DESIGN TOOL FOR FAULT-TOLERANT SYSTEMS
- 0 LARGE REDUCTION OF STATE SIZE
- 0 FAULT-HANDLING MODEL BASED ON PROBABILISTIC DESCRIPTION OF DETECTION, ISOLATION, AND RECOVERY MECHANISMS
- 0 VARIETY OF FAULT AND ERROR MODELS

Figure 1.

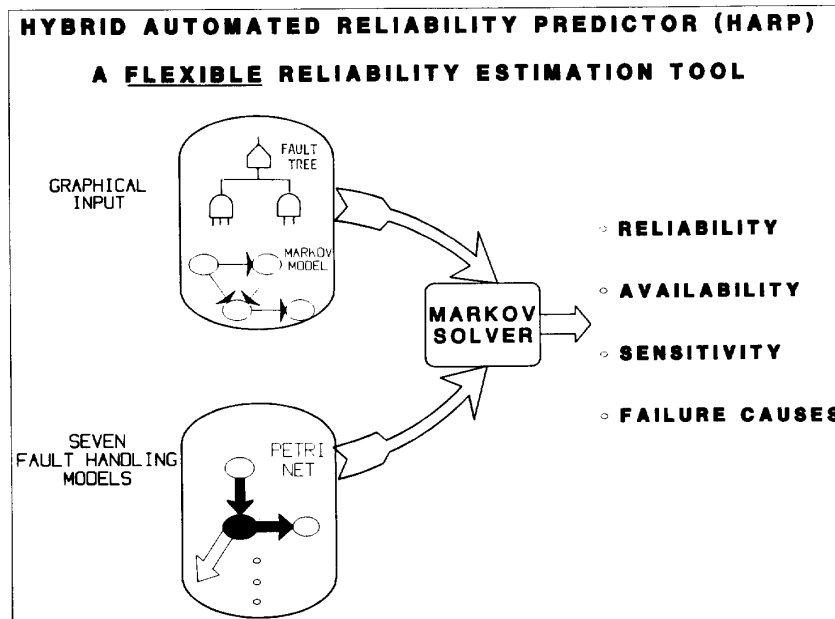


Figure 2.

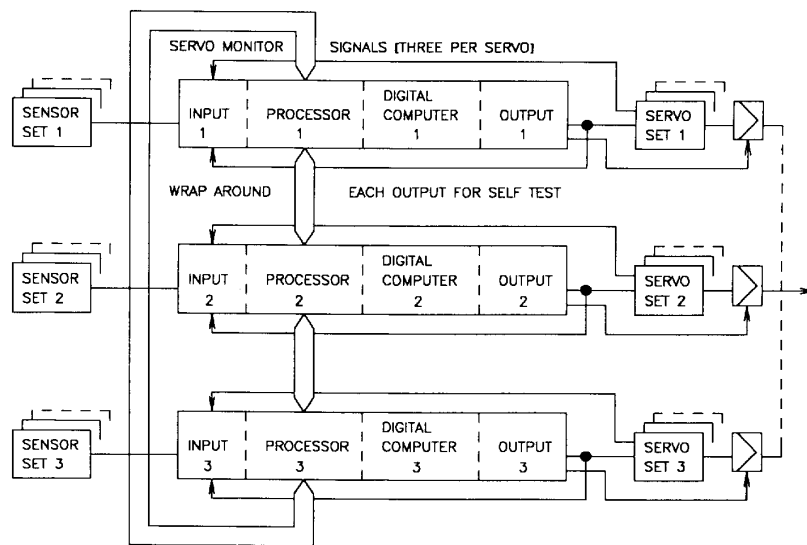


Figure 3. The ARCS (Advanced Reconfigurable Computer System).

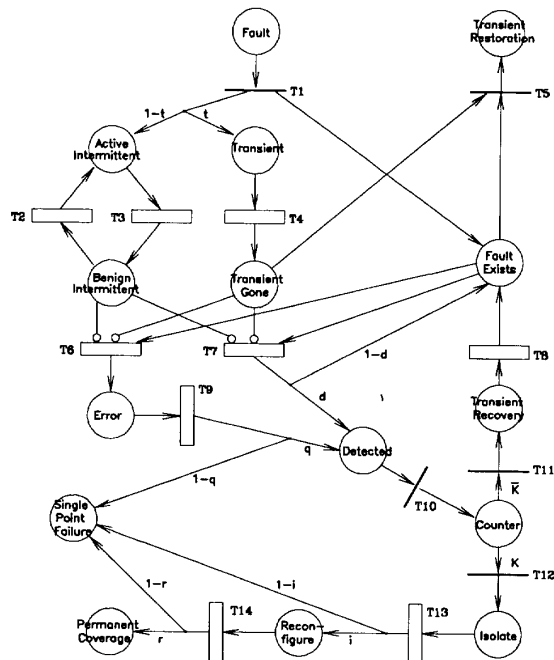


Figure 4. The HARP ESPN single fault model.

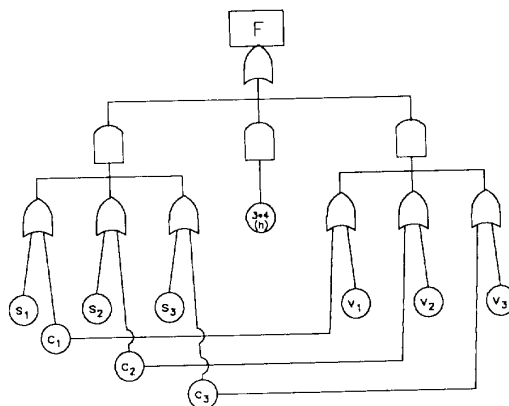


Figure 5. Fault tree representation of ARCS system.

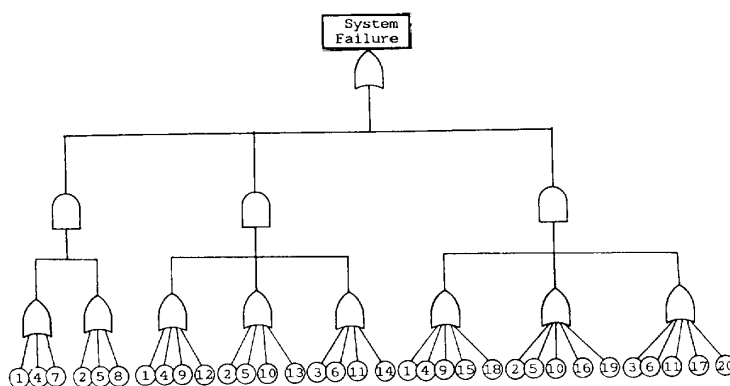
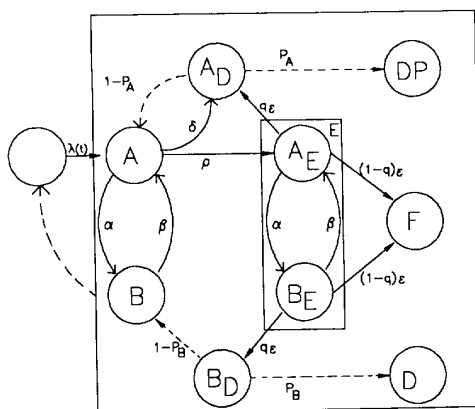


Figure 6. Fault tree representation of a fault-tolerant jet engine controller.
(If two basic event labels are the same, they represent the same component.)



PARAMETERS, ASSUMING ALL FAULTS ARE PERMANENT
(those not listed are 0.0)
Fault Detection Rate, $\delta = 360$ per hour
Error Propagation Rate, $\epsilon = 3600$ per hour
Error Production Rate, $\rho = 180$ per hour
Reconfiguration Probability, $P_A = 1.0$
Error Detectability, $q = .7$

Figure 7. Markov version of the CARE III single fault model.
(The error detectability is 0.97 for data collectors, and is 0.99 for the other stages.)