LANGLEY
GRANT
IN-33-CR
252026
16 P.

Sufficiency and Necessity in Reliability Modeling
NAG-1-755 Progress Report

Robert Geist

Department of Computer Science
Clemson University
Clemson, South Carolina 29634-1906

(NASA-CR-186119) SUFFICIENCY AND NECESSITY IN RELIABILITY MODELING Progress Report (Clemson Univ.) 16 p CSCL 09C

N90-13706

Unclas
G3/33 0252026

Abstract

Limitations of current analytic techniques in estimating the reliability of life-critical electronics systems are discussed. A new framework for specification of recovery and fault-handling submodels is suggested, and is shown through several examples to provide substantially improved modeling accuracy and flexibility. Implementation of the new technique in an X-windows based system, XHARP, is also described. The implementation allows for an automated behavioral decomposition of full system models, heretofore unavailable in such tools.

keywords: ultrahigh reliability, behavioral decomposition, semi-Markov models, X-windows, HARP, XHARP

1 Introduction.

Ultrareliable electronics systems are those designed to provide reliability in excess of $1 - 10^{-9}$ over an intended mission interval. Difficulties in the specification and evaluation of models of such systems can often be directly attributed to excessive model size and excessive model stiffness [6]. A representation sufficiently detailed to capture essential electronic system behavior often can require 10^5 or 10^6 model states [8], and rates of transition among states can differ by 7 or 8 orders of magnitude. Thus an enormous amount of time can be required of the modeler both to specify the details of a complete model and subsequently to extract from the model useful predictions of system performance and reliability.

Behavioral decomposition [9] is a technique designed to address both difficulties. We observe that the huge disparities in state transition rates within system reliability models do not occur randomly but rather between identifiable "groups" of states which are easily characterized by their intended behavior. Relatively low transition rates are typically attached to component failure and fault occurrence behavior, and relatively high transition rates are typically attached to fault handling and system recovery behavior.

Behavioral decomposition thus calls for extracting recovery submodels, which are identified as collections of adjacent states containing high rates of inter-transition, solving them in isolation for certain "coverage" parameters, and then attaching these parameters as modifiers to the (low rate) failure transitions in a reduced, *instantaneous coverage* model, a model that contains only the modified failure transitions.

High rate and low rate models are thus solved separately, thereby eliminating numerical stiffness. Further, to the extent that recovery submodels are replicated within the complete system model, the number of states to be considered is reduced from a product (fault-occurrence × fault-recovery) to a sum, thereby offering reduced effort in both model specification and model solution.

As an example, consider the triple modular redundant (TMR) system model of figure 1. Each of three components fails at constant rate λ. Upon component failure, a recovery module takes over and attempts reconfiguration to a reduced system, first to a dual system and then to a simplex. Recovery is successful unless the system experiences a second, near-coincident fault prior to its completion. The recovery submodel, represented by a box in figure 1, may itself contain numerous states and fast transitions among them, but since the only fast exit from this submodel is to the reconfigured, operational state, the crucial information here is simply the distribution of time to take this fast exit, that is, the distribution of recovery time. If recovery time R has

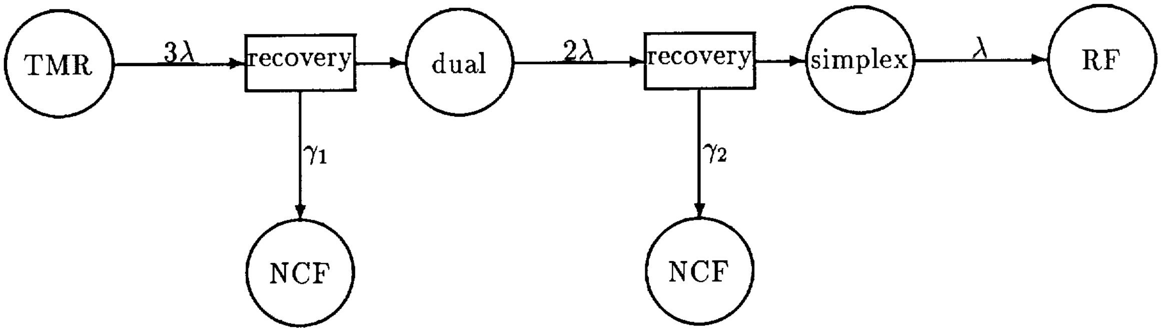

Figure 1: TMR to simplex model.

distribution $F_R(t)$, determined by considering the submodel in isolation, then the probability that we successfully recover from a failure (coverage) is given by

$$\begin{aligned} c &= P(\text{recovery time} \leq \text{time to next fault}) \\ &= \int_0^{+\infty} \int_0^{x} \frac{dF_R(r)}{dr} \gamma e^{-\gamma x} dr dx \\ &= \int_0^{+\infty} e^{-\gamma r} \frac{dF_R(r)}{dr} dr \\ &= L_R(\gamma), \end{aligned} \tag{1}$$

that is, the LaPlace transform of the recovery time distribution evaluated at the near-coincident fault rate, $\gamma = \gamma_1$ or γ_2.

If we now attach these coverage parameters, $c_1 = L_R(\gamma_1)$, $c_2 = L_R(\gamma_2)$, as rate modifiers to the associated failure transitions, we obtain a reduced *instantaneous coverage* model shown in figure 2, and a reliability estimate,

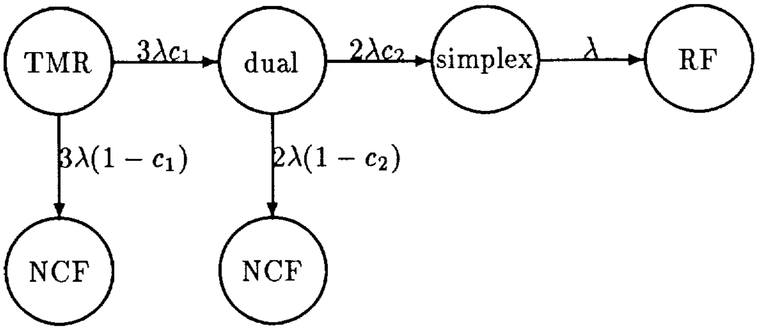

Figure 2: Instantaneous coverage model.

$$\begin{aligned} R(t) &= P_{TMR}(t) + P_{dual}(t) + P_{simplex}(t) \\ &= e^{-3\lambda t}(1 - 3c_1 + 3c_1c_2) + e^{-2\lambda t}(3c_1 - 6c_1c_2) + e^{-\lambda t}(3c_1c_2). \end{aligned} \tag{2}$$

Reliability estimates obtained in this way are provably conservative [4], i.e. the estimate (2) is guaranteed to be a lower bound on the reliability of the system as represented by the full model of figure 1, regardless of the actual distribution of time spent in the recovery submodel(s).

This behavioral decomposition technique has been implemented in the Hybrid Automated Reliability Predictor (HARP) [2, 3], a software package for evaluation of ultrareliable flight control electronics designs, which is distributed by NASA Langley Research Center. We note that the HARP implementation contains several restrictions on model specification that were chosen to facilitate the modeling process:

- Recovery (fault handling) submodels and instantaneous coverage (fault occurrence) submodels are specified separately with connection points (arcs) noted during specification.

- Recovery submodels contain a single entry and three exits:

  C, representing successful reconfiguration or coverage,
  R, representing transient restoration, and
  S, representing single point system failure.

  Although the internal structure of the recovery submodel may take numerous forms, the submodel must externally supply a *defective probability distribution function* associated with each of the three exits. That is, if $X = C, R, S$, the submodel must supply $P_X(t)$ = the conditional probability that holding time in the submodel is $\leq t$ and ends via exit X, given no interfering (near-coincident) faults. Note that the distribution of time to exit X, $F_X(t)$, is given by $F_X(t) = P_X(t)/P_X(+\infty)$.

- Near-coincident faults cause system failure.

In practice these restrictions usually facilitate model specification without substantial impact on modeling flexibility. HARP has been used in estimating the reliability of a wide variety of proposed architectures [1, 7].

Still, these restrictions can present problems in modeling certain classes of systems of interest. The assumption that near-coincident faults cause system failure may be overly conservative in modeling systems that are designed to provide fault-containment regions. In such systems a second, near-coincident fault within the same containment region may simply cause that module to become inactive rather than cause system failure. Further, the single-entry, three-exit recovery submodel framework may preclude direct representation of more sophisticated recovery modules that attempt multiple levels of system reconfiguration. Finally, although separate specification of fault handling and fault occurrence submodels can save modelers a great deal of time, it can be confusing when the modeler starts with a completely integrated total system model. In this case, the behavioral decomposition must be done by hand, a tedious and error-prone task.

In this paper we show how all three restrictions can be easily removed without additional computational complexity and without an increase in state space size. In section 2 we suggest a new recovery model specification framework that provides this additional modeling flexibility and accuracy of system representation. The HARP implementation of behavioral decomposition will be seen to be a special case. In section 3 we illustrate the use of this new framework in modeling a system of two nearly-independent processor triads and a TMR system with cold spare. In section 4 we discuss implementation of the new framework, an extended behavioral decomposition, in XHARP, a reliability estimation tool under development at Clemson University. Section 5 contains conclusions and current directions.

2 Recovery Model Specification.

The essence of the so-called *instantaneous jump theorem* from [4] can be described very simply: let state j in figure 3 be an intermediate state in a semi-Markov model with associated defective exit

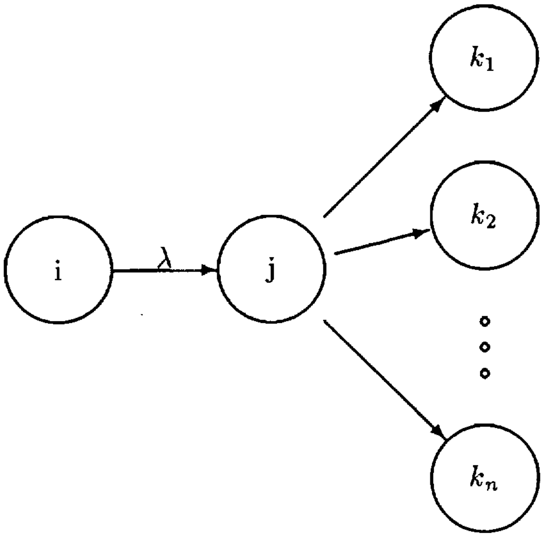

Figure 3: Instantaneous jump: before.

distributions, $P_{k_1}(t), ... P_{k_n}(t)$, that is, $P_{k_r}(t)$ = probability that the holding time in state j is $\leq t$ and ends with a jump to state k_r, r = 1,2,...n. Note that $\sum_{r=1}^{n} P_{k_r}(+\infty) = 1$. If we now remove state j and route incoming arcs, with rates modified by exit probabilities $P_{k_r}(+\infty)$, directly to the exit states as shown in figure 4 then in the resulting, reduced-state model the probability of reaching any absorbing state by any time t will be at least as great as in the original model.

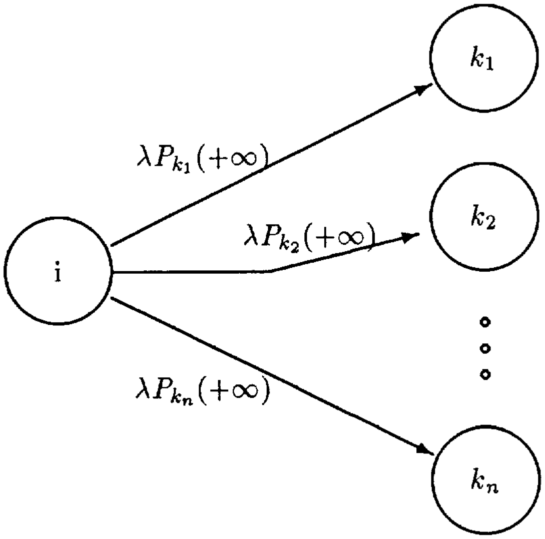
k_1
$\lambda P_{k_1}(+\infty)$
k_2
$\lambda P_{k_2}(+\infty)$
i
$\lambda P_{k_n}(+\infty)$
k_n

Figure 4: Instantaneous jump: after.

The direct use of this result in a reliability modeling package is precluded by the need to allow specification of recovery models independently of any consideration of interfering (low rate) failure transitions.

As noted in the previous section, the HARP package allows a multitude of choices for internal structures in recovery model specification, but the net effect of any choice is to reduce the recovery model to a single semi-Markov state j as in figure 3, but with three exits, C, R, and S, where $\sum_{X \in \{C,R,S\}} P_X(+\infty) = 1$. The effects of an interfering near-coincident fault at rate γ are then incorporated as in the TMR example of section 1, so that the net result is a direct transition from the incoming state i to exit state X at rate $\lambda P_X(+\infty) L_X(\gamma)$. The exit to state NCF, system failure due to near-coincident fault, is at rate $\lambda(1 - \sum_{X \in \{C,R,S\}} P_X(+\infty) L_X(\gamma))$.

Consider now a recovery model with arbitrary internal structure that is seen externally as a single semi-Markov state j with defective exit distributions $P_{k_r}(t), r = 1, 2, ...n$, where $\sum_{r=1}^{n} P_{k_r}(+\infty) =$

1. Suppose that interfering (low rate) transitions cause "premature" exits from this semi-Markov state (equivalently, from all internal submodel states) to states $E_s, s = 1, 2, ...m$, at constant rates $\gamma_s, s = 1, 2, ...m$ respectively. Note that states E_s need not be absorbing.

For exits $X = k_1, ..., k_n, E_1, ...E_m$, let $Q_X(t)$ denote the semi-Markov defective exit distributions from the single state j to exit X when the recovery processes and interfering processes are considered in competition. Then we have

$$Q_{k_r}(+\infty) = P_{k_r}(+\infty)P(\text{time to exit } k_r < min(\text{time to } E_1, ..., \text{time to } E_m))$$

But $min(\text{time to } E_1, ..., \text{time to } E_m)$ has exponential distribution with parameter $\gamma = \sum_{s=1}^{m} \gamma_s$, and so, as in section 1, we have

$$Q_{k_r}(+\infty) = P_{k_r}(+\infty)L_{k_r}(\gamma).$$

In the same spirit we have

$$Q_{E_s}(+\infty) = \frac{\gamma_s}{\gamma}\left[1 - \sum_{r=1}^{n} Q_{k_r}(+\infty)\right].$$

Clearly

$$\frac{Q_{k_r}(t)}{Q_{k_r}(+\infty)} = \frac{P_{k_r}(t)}{P_{k_r}(+\infty)}$$

and so

$$Q_{k_r}(t) = P_{k_r}(t)L_{k_r}(\gamma)$$

and

$$Q_{E_s}(t) = Q_{E_s}(+\infty)(1 - e^{-\gamma_r t})$$

together complete the new semi-Markov specification.

Now we can appeal to the instantaneous jump theorem and route each incoming arc at rate λ directly to exit state k_r at rate

$$\lambda Q_{k_r}(+\infty) = \lambda P_{k_r} L_{k_r}(\gamma)$$

and to exit state E_s at rate

$$\lambda Q_{E_s}(+\infty) \quad = \quad \lambda \frac{\gamma_s}{\gamma}(1 - \sum_r Q_r(+\infty))$$

$$= \lambda\frac{\gamma_s}{\gamma}(\sum_{r=1}^{n} P_{k_r}(+\infty)(1 - L_{k_r}(\gamma)))$$

Since the system failure states are precisely the absorbing states, reliability as estimated by the new model with state j removed will be no higher than that from the original model.

The HARP recovery model framework is then the special case given by $n = 3$, $m = 1$, and $E_1 = NCF$.

3 Examples.

To illustrate the effectiveness of this approach, we consider an example from each of the classes of systems of interest identified in section 1 as potentially difficult to model. The first example illustrates modeling fault-containment regions, and the second illustrates modeling multi-level recovery mechanisms.

3.1 Two Nearly-independent Triads.

Consider two processor triads with component failure rates λ_1 and λ_2 respectively. Nothing precludes $\lambda_1 = \lambda_2$. Recovery is at rate δ and is always successful, unless we have a near-coincident fault. A near-coincident fault in the same triad causes that triad to go off-line, but the system remains operational. A near-coincident fault in the other triad causes system failure. (Both triads cannot be executing recovery procedures simultaneously; somebody has to mind the store!) The system is operational as long as at least one triad is operational. A full model representation is shown in figure 5.

We choose parameter values to make this design ultra-reliable over a 100 hr. mission. If $\lambda_1 = \lambda_2 = 0.25 \times 10^{-4}$/hr. and $\delta = 0.72 \times 10^4$/hr. (1/2 sec. mean recovery), then unreliability at 100 hrs. is 0.504×10^{-9}. Now the instantaneous coverage (HARP) approach would require that near-coincident fault transitions (e.g. that from R_2 to $F3$) be modeled as system failures.

The HARP instantaneous coverage model is shown in figure 6. Here $\bar{c_i} = 1.0 - c_i$, and $c_i = \delta/(ncf_i+\delta)$, where the near-coincident fault rates, ncf_i, are given in table 1. When we solve this instantaneous coverage model, we obtain an unreliability at 100 hrs. of 0.608×10^{-9}. The absolute

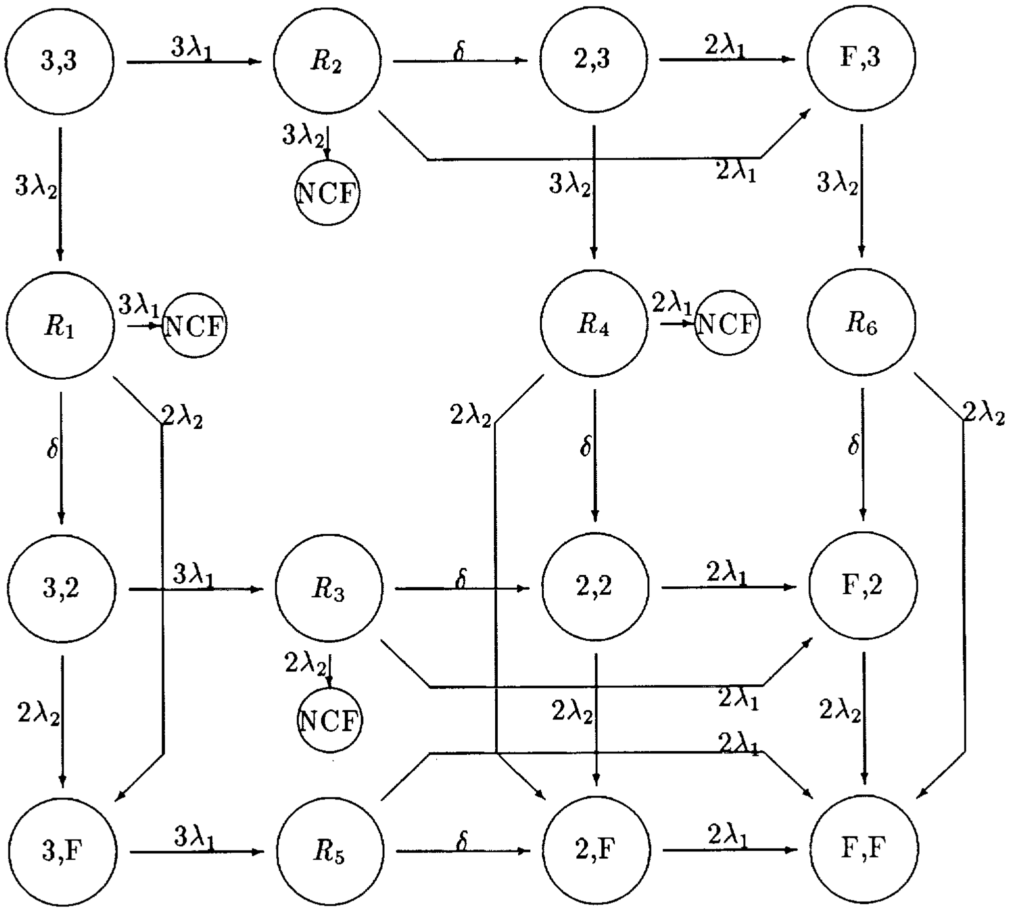

Figure 5: Two triad fault tolerant system.

error is not large, and, given that the estimate is conservative, this may be entirely acceptable. However, we might argue that a relative error of $(.608 - .504)/.504 = 20.6\%$ is unacceptable when it costs us no more to do better.

If we apply the proposed technique of section 2, we need not route near-coincident, same-triad component failures to a system failure state. This is the technique used in the XHARP system, described in more detail in the next section. In this case we obtain the (XHARP) instantaneous coverage model shown in figure 7. Here $\bar{c}_i = 1 - c_i$ and $\bar{k}_i = 1 - c_i - k_i$, where c_i and k_i are given in table 2. Note that the state space size is identical to that of the HARP instantaneous coverage model. In table 3 we compare exact unreliability with the HARP and XHARP instantaneous coverage estimates over an extended interval, [0,500 hrs]. Observe that the XHARP relative error at 100 hrs. is only 0.001%.

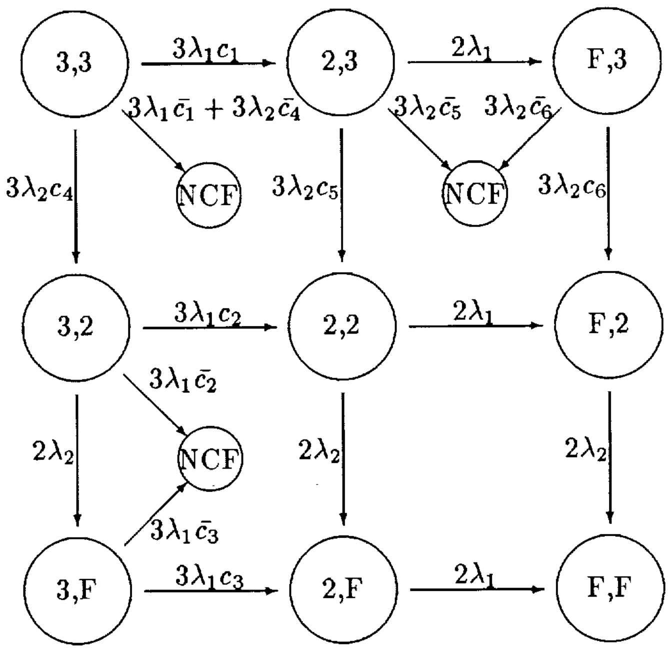

Figure 6: HARP Instantaneous Coverage Model.

ncf_1	$3\lambda_2 + 2\lambda_1$
ncf_2	$2\lambda_1 + 2\lambda_2$
ncf_3	$2\lambda_1$
ncf_4	$3\lambda_1 + 2\lambda_2$
ncf_5	$2\lambda_1 + 2\lambda_2$
ncf_6	$2\lambda_2$

Table 1: HARP near-coincident fault rates

3.2 TMR System with Cold Spare.

Consider a TMR system augmented with a spare that is maintained in a powered off (cold) state. Upon first component failure, a recovery procedure attempts a power-up and a reconfiguration switch to a standard TMR system. During power-up the spare fails at rate γ, and, if this occurs, a secondary recovery procedure takes over and attempts reconfiguration to simplex. During either reconfiguration period, failure of the second (hot) processor causes system failure. A model of this system is shown in figure 8. We take parameter values λ and δ to be the same as in the previous example and power-up failure rate γ to be 0.25×10^{-2}/hr.

The HARP instantaneous coverage model of this system is shown in figure 9. Here power-up failure must be regarded as a near-coincident fault causing system failure. Thus $c_1 = \delta/(2\lambda + \gamma + \delta)$,

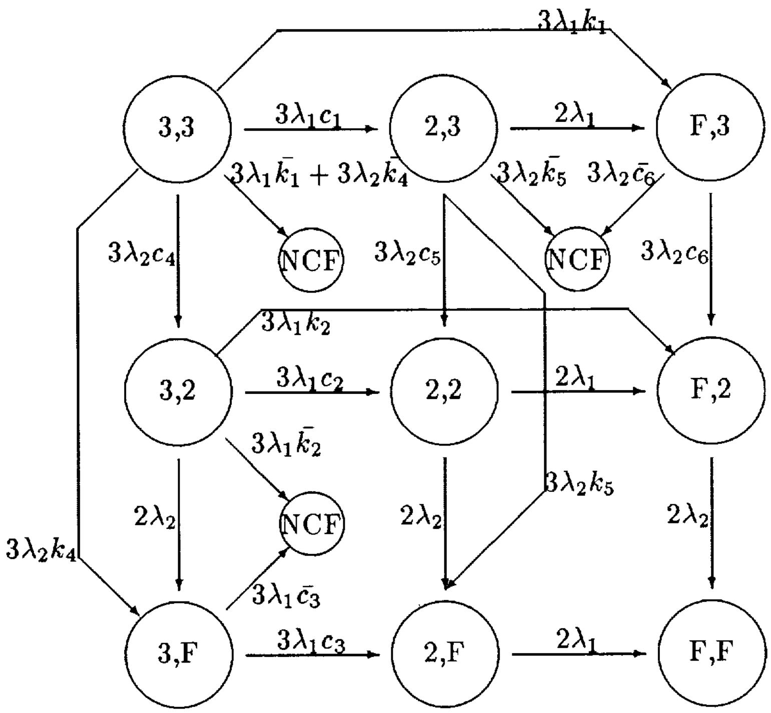

Figure 7: XHARP Instantaneous Coverage Model.

$c_2 = \delta/(2\lambda + delta)$, and $\bar{c}_i = 1 - c_i$.

The XHARP instantaneous coverage model is shown in figure 10. Recovery states R_1 and R_2 in the full model have been eliminated successively, as described in section 2, so here c_1 and c_2 are as in the HARP model, but $k_1 = (\gamma/(2\lambda + \gamma + \delta))(\delta/(2\lambda + \delta))$ and $\bar{k}_1 = 1 - c_1 - k_1$. Note that the state space is the same.

In table 4 we compare HARP and XHARP unreliability estimates with exact values for a 100 hr. mission. We see a relative error at 100 hrs. of 11% (HARP) has been reduced to 0.0008% (XHARP).

4 XHARP

XHARP is an X-windows [5] based reliability estimation tool, now under development at Clemson University, that incorporates the extended notion of behavioral decomposition illustrated in the previous section.

c_1	$\delta/(\delta + 3\lambda_2 + 2\lambda_1)$
k_1	$(2\lambda_1)/(\delta + 3\lambda_2 + 2\lambda_1)$
c_2	$\delta/(\delta + 2\lambda_2 + 2\lambda_1)$
k_2	$(2\lambda_1)/(\delta + 2\lambda_2 + 2\lambda_1)$
c_3	$\delta/(\delta + 2\lambda_1)$
c_4	$\delta/(\delta + 2\lambda_2 + 3\lambda_1)$
k_4	$(2\lambda_2)/(\delta + 2\lambda_2 + 3\lambda_1)$
c_5	$\delta/(\delta + 2\lambda_2 + 2\lambda_1)$
k_5	$(2\lambda_2)/(\delta + 2\lambda_2 + 2\lambda_1)$
c_6	$\delta/(\delta + 2\lambda_2)$

Table 2: XHARP near-coincident fault rates

time(hrs).	exact	xharp	harp
0	0.00000000 e+00	0.00000000 e+00	0.00000000 e+00
100	5.04113640 e-10	5.04118969 e-10	6.07893735 e-10
200	5.84145798 e-09	5.84148951 e-09	6.04825212 e-09
300	2.82358066 e-08	2.82359114 e-08	2.85448714 e-08
400	8.76653559 e-08	8.76655974 e-08	8.80759572 e-08
500	2.11533987 e-07	2.11534456 e-07	2.12045407 e-07

Table 3: Two-triad unreliability estimate comparison

The X-windows system provides a client-server model of interaction that allows both device independence in interface design and separation of interface engine from application engine. Applications that demand large address space and substantial processing power can execute in an appropriate environment without concern for the physical location of the user interface, usually an independent workstation or PC. Thus a graphical interface, such as that currently available on PC-HARP [1], need not restrict the user to the often extremely limited processing power directly available on display devices.

However, the XHARP interface goes beyond an extension of PC-HARP to the X-windows environment in that it addresses the previously mentioned model specification restriction. Recall that the requirement of independent submodel specification can be viewed as a restriction as well as a time-saving benefit. If the modeler already has a full system model specification at hand, then a manual behavioral decomposition to obtain the necessary independent submodels can be an arduous and error-prone task.

The graphical interface of XHARP resolves this issue by allowing complete model specification in either of two directions. Like PC-HARP, specification may be top-down, that is, the

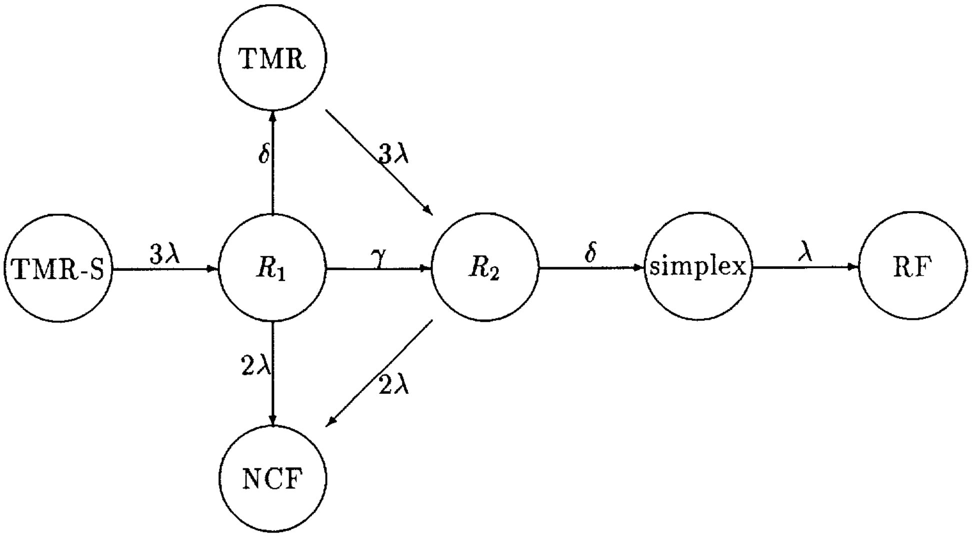

Figure 8: TMR System with Cold Spare.

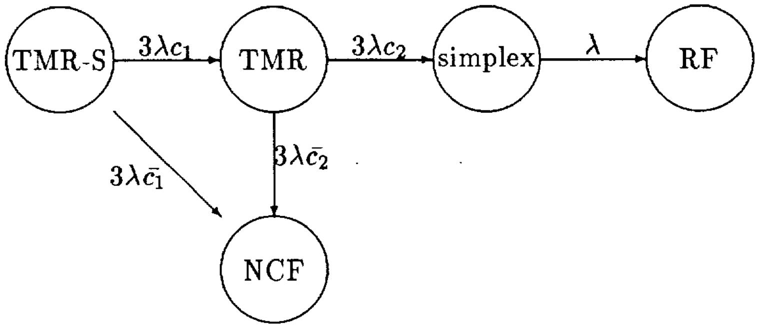

Figure 9: HARP Instantaneous Coverage Model.

modeler may specify the system in terms of states (circles), submodels (rectangles), and interconnecting transitions (arcs), where submodel specification is to be carried out in subsequent phases. An example display is shown in figure 11. Submodel specification is, of course, consistent with the extended behavioral decomposition illustrated in the previous section. Submodels may have an arbitrary number of entrance arcs, exit arcs, and submodel-wide, competing, "premature exit" transitions, again to arbitrary model states.

Unlike PC-HARP, the modeler may also provide a complete (semi-)Markov model specification, without submodel designations, and then call for automated behavioral decomposition. Internal model states having one or more fast exit transitions are then successively eliminated using the techniques illustrated in the previous section. Thus, automated decomposition applied to the

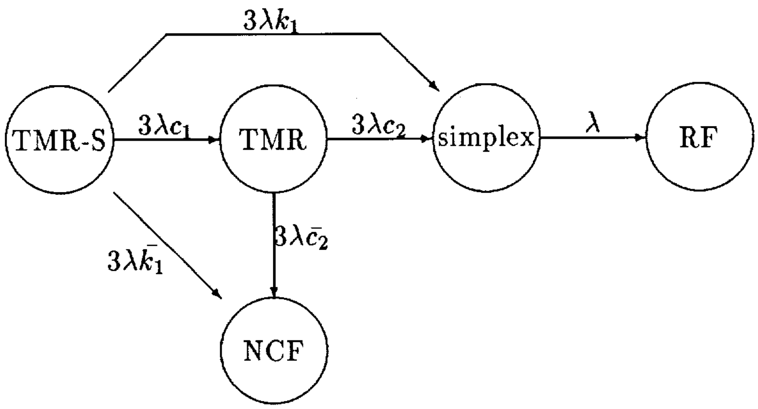

Figure 10: XHARP Instantaneous Coverage Model.

time(hrs).	exact	xharp	harp
0	0.00000000 e+00	0.00000000 e+00	0.00000000 e+00
10	2.86661805 e-11	2.86681789 e-11	2.88954194 e-10
20	1.97874828 e-10	1.97882821 e-10	7.18195059 e-10
30	6.47882414 e-10	6.47899956 e-10	1.42797818 e-09
40	1.51869872 e-09	1.51873047 e-09	2.55831534 e-09
50	2.95009062 e-09	2.95013924 e-09	4.24897140 e-09
60	5.08157738 e-09	5.08164755 e-09	6.63946720 e-09
70	8.05243605 e-09	8.05253109 e-09	9.86907889 e-09
80	1.20016974 e-08	1.20018209 e-08	1.40768384 e-08
90	1.70681478 e-08	1.70683059 e-08	1.94015328 e-08
100	2.33903332 e-08	2.33905286 e-08	2.59817075 e-08

Table 4: TMR with cold spare unreliability comparison

full model of figure 5 yields figure 7 and applied to the full model of figure 8 yields figure 10. We emphasize that although the accuracy of the XHARP estimate depends on the degree of disparity in transition rates, the conservativeness does not, and thus *any* internal state may be eliminated from the full model.

5 Conclusions.

We have suggested an extended framework for the use of behavioral decomposition in modeling ultrareliable electronics systems. The new framework makes full use of the so-called *instantaneous jump theorem* from [4] by viewing the collection of interfering, premature exits from any fault handling and recovery submodel as defining a new, competing process submodel. This approach allows

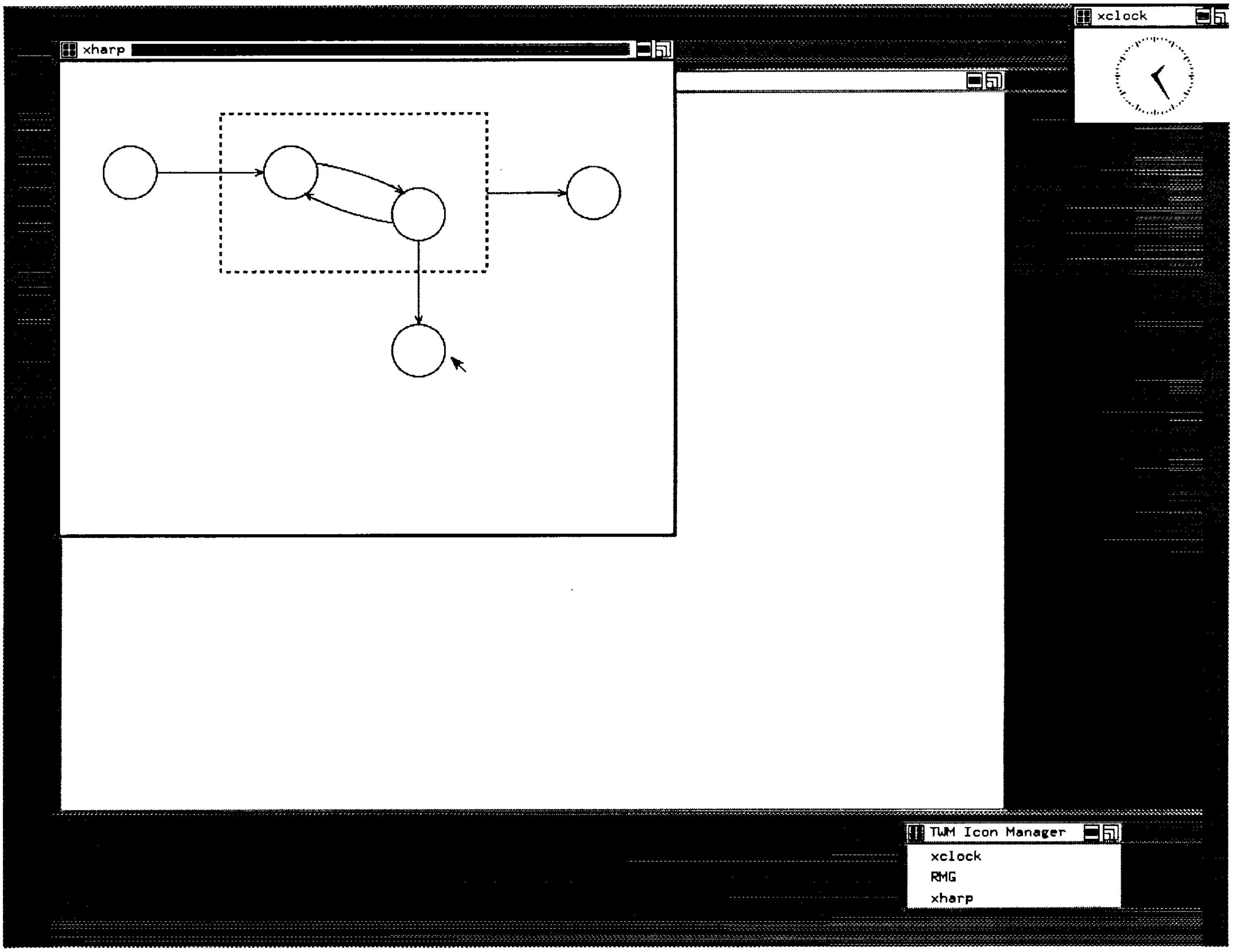

Figure 11: XHARP sample display.

a greater flexibility in submodel representation since submodels may contain arbitrary entrance arcs, exit arcs, and competing, interfering transitions with arbitrary destinations. Since the effects of near-coincident faults need not be represented as system failure events, reliability estimates produced by this approach need not be unduly conservative. Comparisons on small models, where exact results can be computed, show substantial improvement in accuracy.

We have also briefly discussed the implementation of this approach in the XHARP system, an X-windows based tool for reliability estimation of electronics systems. The dual top-down/bottom-up interface provides an added flexibility of allowing an automated behavioral decomposition that is based on the suggested new framework.

Extensions underway are numerous and include the development of a complementary, optimistic reliability estimate, a facility for handling global time dependence of both low-rate, failure transitions and entire recovery submodels, and a uniform, high-level model specification language that can offer the modeler both compactness of specification and wide modeling power.

References

[1] S. Bavuso. A fourth generation reliability predictor. *Proc. Annual Reliab. and Maintainability Symp.*, pages 11–16, 1988.

[2] J. Dugan, K. Trivedi, M. Smotherman, and R. Geist. The hybrid automated reliability predictor. *AIAA J. of Guidance, Control, and Dynamics*, **9**:319–331, 1986.

[3] R.M. Geist, M.K. Smotherman, K.S. Trivedi, and J.B. Dugan. The reliability of life-critical computer systems. *Acta Informatica*, **23**:621–642, 1986.

[4] J.G. McGough, M.K. Smotherman, and K.S. Trivedi. The conservativeness of reliability estimates based on instantaneous coverage. *IEEE Trans. on Comp.*, **C-34**:602–609, 1985.

[5] Adrian Nye. *Xlib Programming Manual.* O'Reilly & Associates, Sebastopol,CA, 1989.

[6] A. L. Reibman and K. S. Trivedi. Numerical transient analysis of Markov models. *Computers and Operations Research*, 15(1):19–36, 1988.

[7] A. Somani and T. Sarnaik. Reliability analysis and comparison of two fail-op/fail-op/fail-safe architectures. *Proc. FTCS-19*, pages 566–573, Chicago, IL, June, 1989.

[8] J.J. Stiffler and L.A. Bryant. Care iii phase iii report - mathematical description. *NASA Contractor Report 3566*, November, 1982.

[9] K.S. Trivedi and R. M. Geist. Decomposition in reliability analysis of fault-tolerant systems. *IEEE Trans. on Reliab.*, **R-32**:463–468, December, 1983.