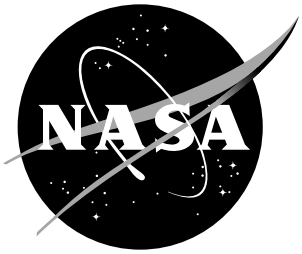
NASA/TM-1998-208427



A Note About HARP's State Trimming Method

Ricky W. Butler, Kelly J. Hayhurst, and Sally C. Johnson
Langley Research Center, Hampton, Virginia

May 1998

The NASA STI Program Office ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counter-part of peer reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that help round out the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results ... even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***

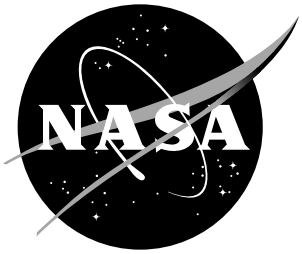
- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at (301) 621-0134

- Phone the NASA Access Help Desk at (301) 621-0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076-1320

NASA/TM-1998-208427



A Note About HARP's State Trimming Method

Ricky W. Butler, Kelly J. Hayhurst, and Sally C. Johnson
Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

May 1998

Available from the following:

NASA Center for AeroSpace Information (CASI)
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 487-4650

Contents

1 Introduction

This short note provides some additional insight into how the HARP program works. In some cases, it is possible for HARP to trim away too many states and obtain an optimistic result. The HARP Version 7.0 manual[1] warns the user that "Unlike the ALL model, the SAME model can automatically drop failure modes for certain system models. The user is cautioned to insure that no important failure modes are dropped; otherwise, a non-conservative result can be given". This note illustrates how this can occur and gives a pointer to further documentation that furnishes a means of bounding the error associated with trimming. This note provides a theoretical discussion of trimming, but does not provide testing results for HARP.

2 Overview of the Harp Program

2.1 Reliability Modeling: Background

Markov modeling provides a means for calculating the reliability of a fault-tolerant computer system when given values for its parameters. In the Markov modeling approach, a system is represented by a vector of attributes that change over time. A particular set of values of the attributes is called a "state" of the system. These attributes are typically system characteristics such as the number of working processors, the number of spare units, the number of faulty units that have not been removed, etc. The more attributes included in the model, the more complex the model will be. Thus, one typically tries to choose the smallest set of attributes that can *accurately* describe the fault-related behavior of the system. An important goal in reliability modeling is to ignore aspects of the system that are unimportant (i.e. do not affect the reliability) and to include aspects that are important. This is accomplished by letting each state in the reliability model represent many different states in the actual system.

Certain states in the system represent system failure, while others represent fault-free behavior or correct operation in the presence of faults. To adequately estimate reliability, the model chosen for the system must represent system failure properly. Defining exactly what constitutes system failure is difficult because system failure is often an extremely complex function of external events, software state, and hardware state. The next step in the modeling process is to characterize the transition time from one state to another. Since this transition time is rarely deterministic, the transition times are described using a probability distribution.

Typically, the transitions of a fault-tolerant system model fall into two categories: slow failure transitions and fast recovery transitions. If the states of the model are defined properly, then the slow transitions can be obtained from field data and/or MIL-STD 217C calculations. The faster transition rates correspond to system responses to fault arrivals and can be measured experimentally using fault injection.

The simplest architecture to model is a single computer. To model this, let T be a random variable representing the time to failure of the computer. Next, we must define a distribution for T, say $F(t)$. Typically, it is assumed that electronic components, and consequently computers, fail according to the exponential distribution:

$$F(t) = Prob[T < t] = 1 - e^{-\lambda t}$$

The parameter λ completely defines this distribution.

The Triple-Modular Redundant (TMR) is one of the simplest fault-tolerant computer architectures. The system consists of three computers all performing exactly the same computations on

exactly the same inputs. The computers are assumed to be physically isolated such that a failed computer cannot affect another working computer. Mathematically, therefore, the computers are assumed to fail independently. It is further assumed that the outputs are voted prior to being used by the external system (not included in this model), and thus a single failure does not propagate its erroneous value to the external world. Thus, system failure does not occur until two computers fail. The model of Figure 1 describes such a system. State 1 represents the initial condition of

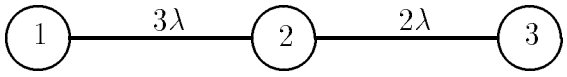


Figure 1: Model of a TMR System

three working computers. The transition from state 1 to state 2 is labeled 3λ to represent the rate at which any one of the three computers fail. Since all of the computers are identical, the failure rate is the same for each computer, λ. The rate at which any of the three computers fail is 3λ. The system is in state 2 when one processor has failed. The transition from state 2 to state 3 has rate 2λ since there are only two working computers that can fail. State 3 represents system failure because a majority of the computers in the system have failed.

Now, consider a reconfigurable quadraplex. Such a system starts with 4 processors then degrades to a triplex then to a duplex and then to a simplex in response to processor failures. The system fails if two near-coincident faults occur (i.e. a second fault arrives before the system can recover from the first fault) or if all of the processors fail before the end of a specified mission time. The probability of failure of this system can be computed using the Markov model in Figure 2. The

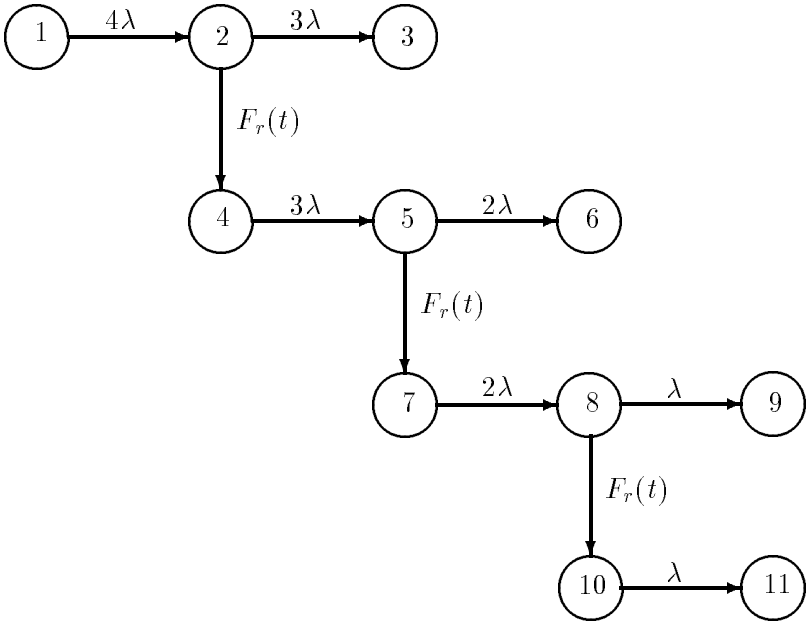


Figure 2: Model of a Degradable Quad

transition from state 1 to state 2 represents failure of one of the four processors in the quad. The

near-coincident faults that could lead to system failure from state 2 would be failures of either of the other three processors in the quad. Thus, the near-coincident fault rate is 3λ. The transition from state 2 to state 4 represents system recovery, which degrades the system to a triplex, and is labelled with F_r. The notation F_r represents the distribution of the recovery time. The transition from state 4 to state 5 represents failure of a processor in the newly formed triad. If either of the other two good processors fail before the system reconfigures then the system fails. Thus, the near-coincident failure rate at this state would be 2λ. The transition from state 5 to state 7 represents system recovery, which degrades the system to a duplex. The rest of the model is developed using similar logic.

2.2 Introduction to HARP User-Interface

The HARP FORM/FEHM approach to model specification is briefly described in this section.

In many reliability analysis programs, the model in Figure 2 would be entered as a single entity by delineating each transition in the model. The HARP program uses a different approach to describe a model predicated upon analyzing the slow fault-arrival behavior of the system and the fast fault-recovery behavior separately. To input a model, the HARP user creates three separate items: a Fault-Occurrence/Repair Model (FORM), a Fault/Error-Handling Model (FEHM) and an "Interfering Components Specification" (ICS). Conceptually, HARP calculates the probability of system failure, P_{sys}, from these three user inputs. The user-supplied FORM defines the sequence of events that leads to system failure by exhaustion of parts given that all recovery processes are instantaneous and perfect. This can be input directly by the user or generated by the HARP program from a fault-tree description of the fault-occurrence behavior. For the degradable quadraplex modeled in Figure 2, the HARP FORM is given in Figure 3. The FORM describes all of the sequences

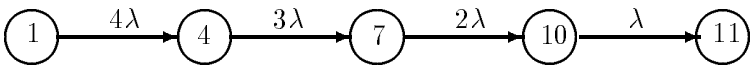
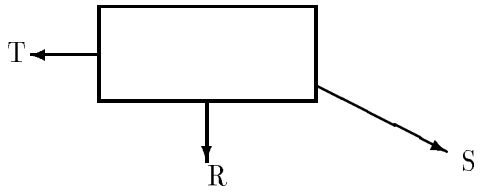


Figure 3: HARP FORM For a Degradable Quadraplex

of failures that can lead to system failure. In a FORM it is assumed that all reconfigurations take place perfectly and instantanously. Consequently there are no reconfiguration or near-coincident failure transitions in a FORM.

Next, the HARP user defines a FEHM that describes the recovery process for each fault type. In the case of the quadraplex in Figure 2, there is only one fault type so there is only one FEHM model. However, it is used in several places. The FEHM corresponds to the transitions labelled F_r in Figure 2. HARP allows the user to describe the recovery process in a variety of ways. One of the simplest methods is the "Moments" option. Here, the user supplies the first three moments of F_r. The following are alternative Fault/Error-Handling Models supported by HARP: Values, Probabilities and Distributions, Probabilities and Empirical Data, ARIES, CARE III, and ESPN. For every type of FEHM there are three exit transitions labelled T,R, and S corresponding to transient-fault recovery, permanent fault-recovery and single-point failure, respectively. This is illustrated in Figure 4. In many examples, the Moments option is used with the T and S transition probabilities set to zero. In this case the FEHM reduces to a single exponentially distributed transition with a mean transition time of μ (or rate δ equal to $1/\mu$.) There is one FEHM for each possible fault type in the FORM.

FEHM:



simple example (i.e. T and S = 0):

Figure 4: FEHM Exit Transitions And Simple Example

After defining system recovery, the user is asked to identify the interacting components. This is done by specifying one of the following options for ICS: ALL, SAME, or USER-DEFINED. From this information, the HARP program must determine the probability of entering the near-coincident-failure death states. That is, the HARP program must effectively deduce all of the information that is contained in Figure 2 from the three separate inputs: FORM, FEHM, and ICS. The HARP program merges the FORM and FEHMs into a single model internally to calculate the probability of system failure. The first step of the merging process is illustrated in Figure 5. The set of faults that coincidently fail the system must be inferred from the FORM and the ICS

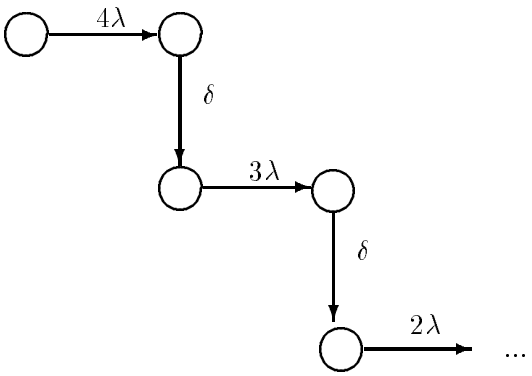


Figure 5: First Step Of FORM/FEHM Merging Process

specification since this information is not directly input by the user. HARP calculates the near-coincident failure rate for a given transition by an algorithm that examines the failure transitions of the previous and next states. The algorithm is different depending upon whether the user specifies the ICS input to be ALL, SAME, or USER. The ICS=ALL case is illustrated in Figure 6.

3 The HARP Trimming Method

Whenever there is only one component in the system, there is no difference between ICS=SAME and ICS=ALL. Thus, the simplest system one can use to illustrate FORM/FEHM merging for ICS=SAME is a system with two components. Consider a system that consists of two subsystems

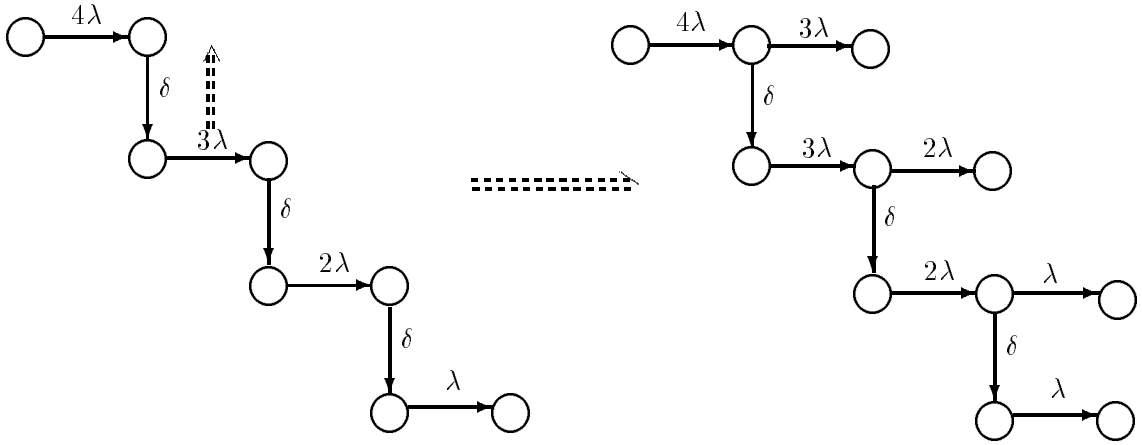


Figure 6: Second Step Of FORM/FEHM Merging Process

each of which consists of a triplex that degrades to a simplex. Each subsystem consists of a set of replicated processors of one component type. Although each processor within a subsystem is identical, the processors in one subsystem can be given a different failure rate than the processors in the other subsystem. Let λ_1 represent the failure rate of the processors in subsystem 1 and λ_2 the failure rate of the processors in subsystem 2. The FORM for this system is illustrated in Figure 7. For simplicity suppose that the FEHM for each subsystem does not have a single-point failure

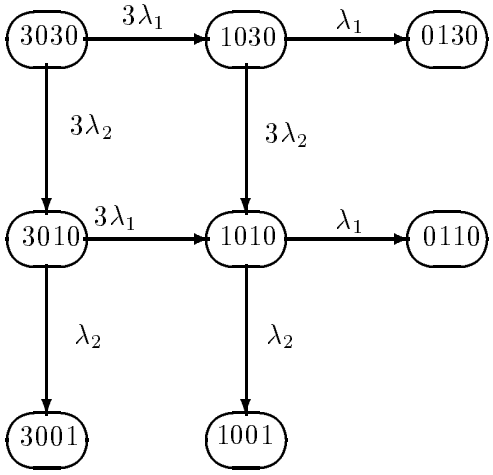


Figure 7: FORM for a Two Triplex-to-Simplex System

or transient-fault exit. Therefore, using the Moments FEHM option we need only specify the first three moments of the permanent fault recovery process. For simplicity, assume that each recovery is exponential with rate δ_1 for subsystem 1 and rate δ_2 for subsystem 2.

Before illustrating how the HARP FORM/FEHM merging technique works, consider what a complete model of this system would contain. In Figure 8 each of the states has been labelled with four numbers representing four attributes of the system (NW1,NF1,NW2,NF2):

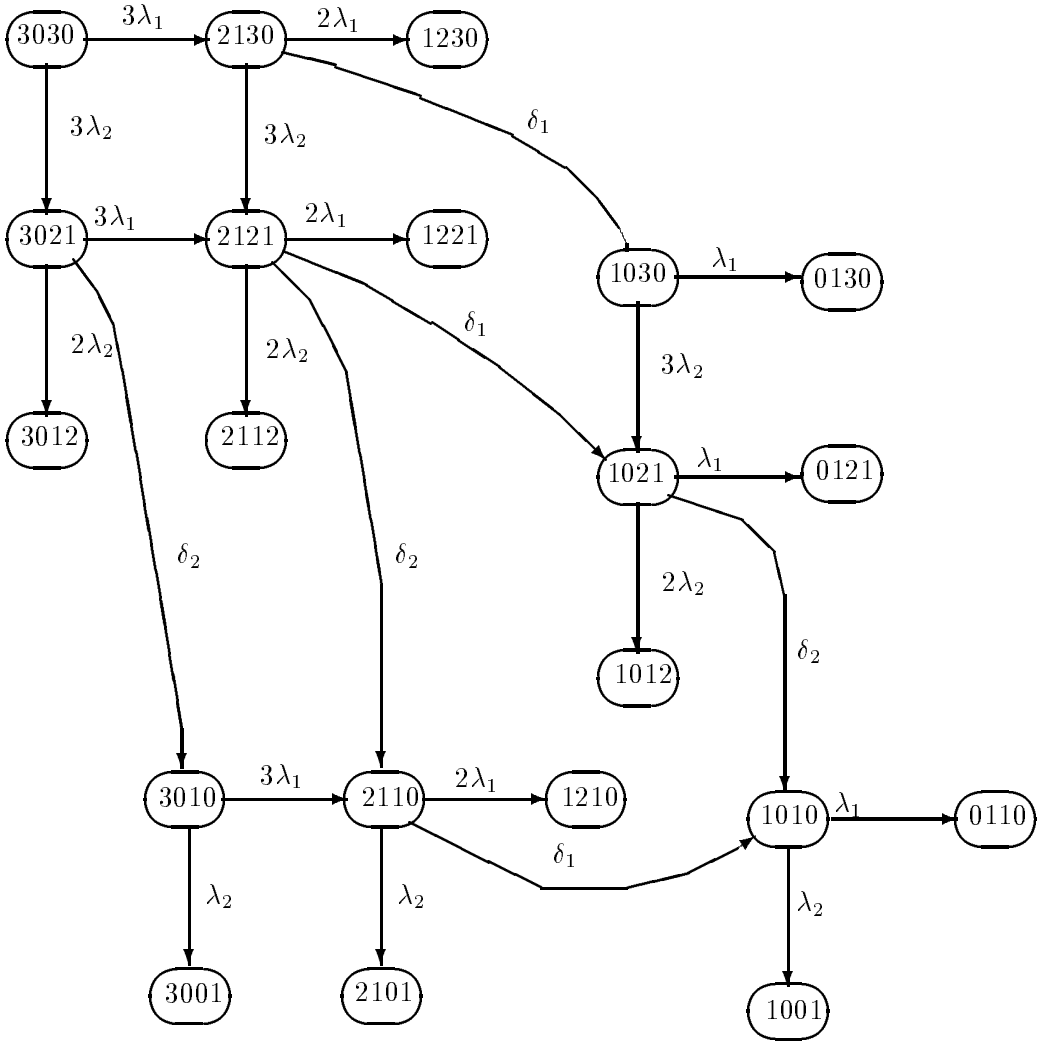


Figure 8: Complete Model of System Consisting of Two Triad-to-Simplex Subsystems

NW1: Number of working processors in subsystem 1
NF1: Number of faulty processors in subsystem 1
NW2: Number of working processors in subsystem 2
NF2: Number of faulty processors in subsystem 2

The system starts in state (3030). This means that each subsystem has 3 working processors and no faulty processors. If a processor in subsystem 1 fails, the system transitions to state (2130). If a processor in subsystem 2 fails, the system transitions to state (3021). While in state (2130), the system is trying to reconfigure. If it reconfigures before a second processor in subsystem 1 fails, the system transitions to state (1030), i.e. the first subsystem is a simplex and the second subsystem is still a triplex. If a second processor in subsystem 1 fails before it reconfigures, then the system fails in death state (1230). Note that there are two simultaneous faults in subsystem 1 in this situation. If a second processor in the other subsystem fails before reconfiguration is completed, then the system goes to state (2121). In state (2121) both triads have a single faulty processor. Since the triads are independent, this does not represent system failure. From this state, four possible events can happen next:

1) a second processor in subsystem 1 fails causing system failure,
2) a second processor in subsystem 2 fails causing system failure,
3) subsystem 1 reconfigures by degrading to a simplex, or
4) subsystem 2 reconfigures by degrading to a simplex.

The two reconfiguration transitions take you to states (1021) and (2110), respectively. Note that in state (2121) there are two competing recoveries.

The result of the FORM-FEHM merging process is illustrated in Figures 9 and 10. In the ICS=ALL case, the system state (2121) is made a death state (see figure 9). Note that this is a conservative model. In fact it can be a very conservative model because it ignores the fault isolation regions associated with each subsystem.

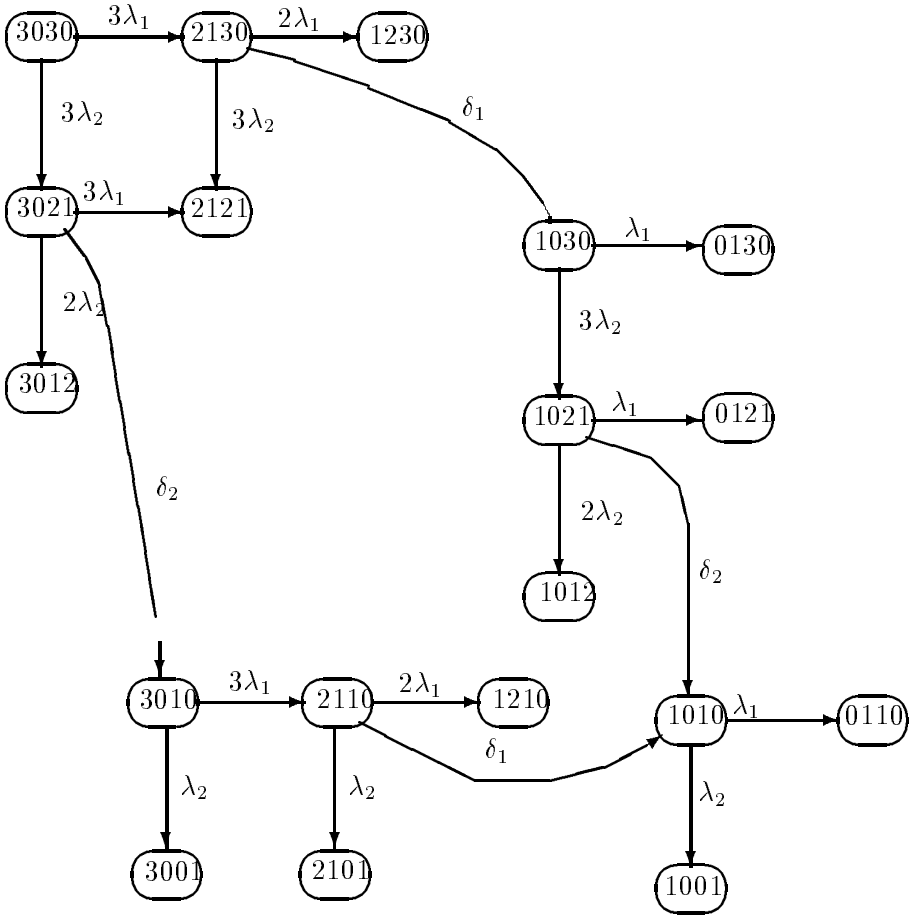


Figure 9: Merged FORM/FEHM for a Two Triplex-to-Simplex System (ICS=ALL)

In the ICS=SAME case (see figure 10), state (2121) and its descendents (1221) and (2112) are trimmed away. This is an optimistic model, since some failure modes are ignored, i.e. states (1221)

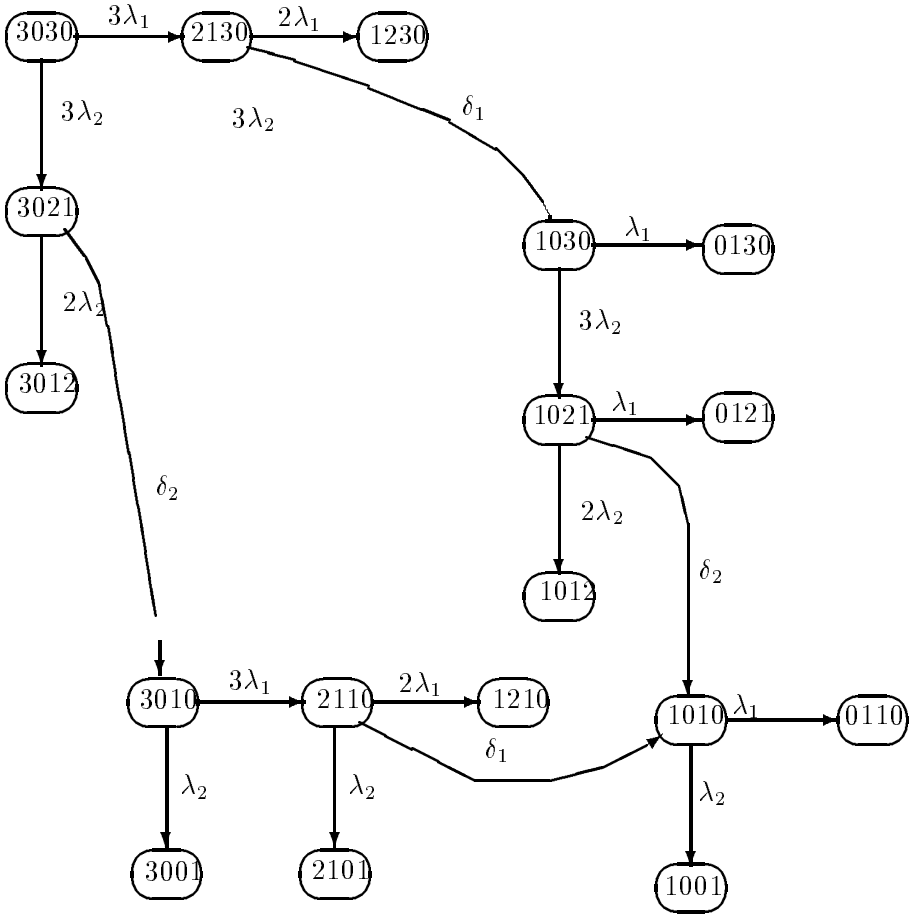


Figure 10: Merged FORM/FEHM for a Two Triplex-to-Simplex System (ICS=SAME)

and (2112) are not included in the computed probability of system failure. Usually these states contribute a very insignificant amount of probability in comparison to other states in the model, e.g., state (1230). This can be seen by noting that there are only two component failures leading to the death state (1230) whereas states (1221) and (2112) occur after 3 component failures. Thus, they are typically several orders of magnitude smaller than the dominant failure states in the model. However, the HARP program does not calculate any error bound on the amount of probability that is being ignored. Thus, the FORM/FEHM merging technique for ICS=SAME relies on a heuristic solution. This heuristic solution can be very good and can be used successfully if one can determine that the amount trimmed away is insignificant. A bound has been developed by Dr. Allan White[2, 3], but it is not implemented in HARP. However, the user of HARP can manually calculate this bound to determine if too much has been trimmed away by the FORM/FEHM merging process.

Since the ICS=SAME method ignores some failure modes, the question arises whether it can serve as a lower bound on the system probability of failure. However, since HARP uses both conservative approximations (e.g., instantaneous jump) and non-conservative approximations (e.g.,

trimming of certain failure modes) when computing its result, one cannot determine whether the answer is conservative or optimistic. To illustrate the problem, suppose in calculating x, one first uses a non-conservative approximation y: $x > y$. Then, one uses a conservative approximation z for y: $y < z$. The true relationship of z to x is not determinable. This is precisely the situation when one uses ICS=SAME.

4 Example of Non-Conservative Trimming

As shown in section 2, the HARP FORM/FEHM merging technique trims away some transitions from the model when the user specifies ICS=SAME. In this section some examples are constructed to demonstrate the effect of trimming states.

Consider a system of N triads that functions much like FTMP. If a processor in a triad fails and spares are available, the system repairs the triad with a spare. If no spares are available, the system removes the faulty triad from the configuration and adds the good processors to the spares pool. System failure occurs if a triad has two faulty processors (i.e. a second processor fails before it can be repaired or removed from the system) or if there are not enough triads remaining to run the workload (i.e. exhaustion of parts). The system has difficulty diagnosing which processors are faulty when more than one triad has a faulty processor. Therefore, in this situation the system does not reconfigure.[1] For simplicity, it is assumed that the system never misdiagnoses a faulty processor and knows when it has more than one triad with a faulty processor. Two parameters of this system are relevant:

NI	=	number of triads in the initial configuration
MNT	=	minimum number triads needed to execute the workload

Several (NI,MNT) system configurations are presented for which ICS=SAME yields non-conservative results. However, to facilitate the discussion, a configuration where the ICS=SAME result is acceptable is presented first.

4.0.1 Configuration Where ICS=SAME Produces Negligible Error

Consider a (2,1) configuration. The complete model is shown in Figure 11. Initially the system has two good triads and no spares. Thus, the first transition is from state (3) $\longrightarrow$ (4) with rate 6λ. After the system is in state (4) several things can happen. Another processor in the same triad could fail causing system failure (i.e. we enter state (1)). This is a near-coincident failure that occurs at rate 2λ. Alternatively, the system could recover from the first fault taking us to state (6). One option remains—a processor could fail in the other triad. This takes us to state (5) and occurs at rate 3λ. This is the transition that would be implicitly omitted by HARP if the user specifies ICS=SAME. The rest of the model is clear if one keeps in mind that the two good processors of the removed triad are made spares. Thus, the recovery transitions from (7) $\longrightarrow$ (8) and (9) $\longrightarrow$ (10) replace a faulty processor with a spare. While not being used, the failure rate of the spares are assumed to be zero to simplify the model. The ICS=SAME model is shown in Figure 12.

The ICS=SAME model differs from the full model in that there are no failure transitions from a recovery state that do not end in a death state. This is true in general for HARP. When one sets ICS=ALL, all failure transitions end in the death state, so there are no transitions omitted in this case. When one sets ICS=SAME they are implicitly omitted from the model. In the model of

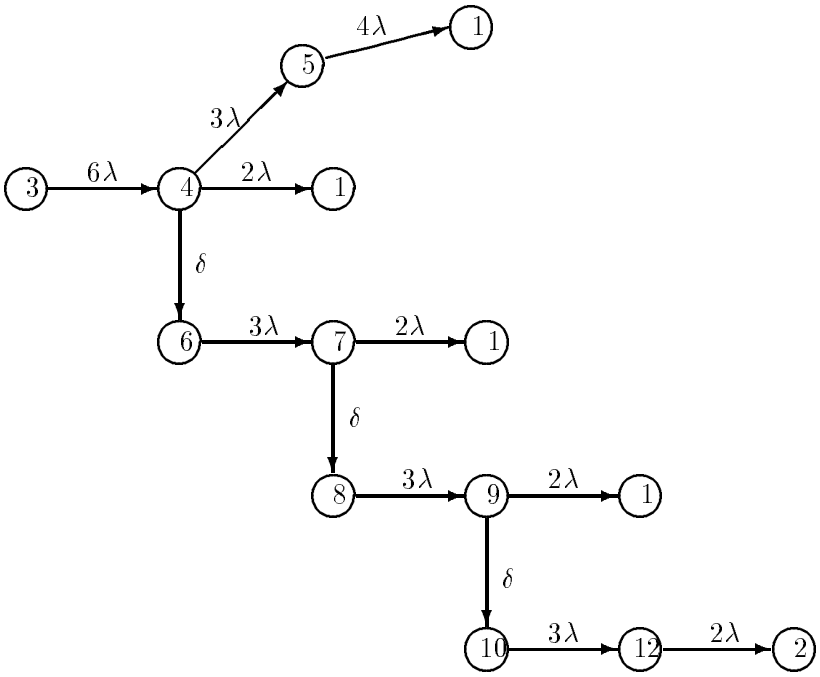


Figure 11: Two Triads—Full Model

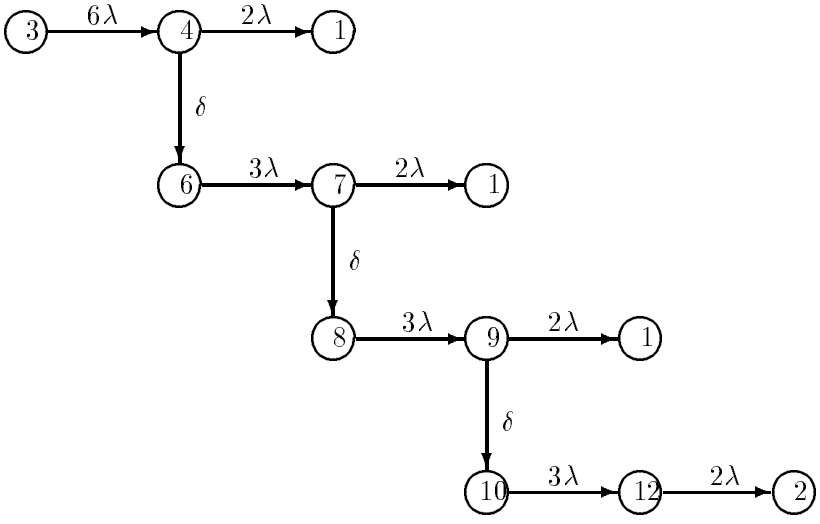


Figure 12: Two Triads—ICS=SAME Model

Model		P_f estimated by PAWS
Parameters:	$\lambda = 10^{-4}$ per hour	
	$\delta = 10^{4}$ per hour	
	Time = 10^{3} hours	
Full Model (Figure 1)		2.03968×10^{-5}
ICS=Same Model (Figure 2)		2.03942×10^{-5}

Table 1: P_f at 1000 Hours for (2,1) Configuration

Figure 12 the transition from (4) $\longrightarrow$ (5) has been omitted.

The probability of system failure after 1000 hours is given in Table 1. For this system, the error is clearly negligible, but in the non-conservative direction.

It should be noted that these computations were not performed using HARP, because HARP performs an additional approximation (based on behavioral decomposition) after trimming, which would greatly complicate the illustration here. See section 5 for an overview of this technique and a discussion of how this additional approximation may mask the error due to trimming in some cases.

4.0.2 Configuration Where ICS=SAME is Non-Conservative

The non-conservatism of the ICS=SAME model becomes increasingly more significant as NI is increased. Consider a system with 10 triads. A portion of the 175-state (10,1) model is shown in Figure 13.

As before, the ICS=SAME model does not have any failure transitions exiting from a recovery state that do not end in a failure state; e.g., transitions (4) $\longrightarrow$ (5) and (8) $\longrightarrow$ (10) are omitted. The results for a (10,1) configuration are shown in Table 2.

Model		P_f estimated by PAWS
Parameters:	$\lambda = 10^{-4}$ per hour	
	$\delta = 10^{4}$ per hour	
	Time = 10^{3} hours	
Full Model (Figure 3)		2.15×10^{-7}
ICS=Same Model		5.53×10^{-8}

Table 2: P_f at 1000 Hours for (10,1) Configuration

Thus, for this model, the ICS=SAME truncation method yields a result that is significantly non-conservative (i.e. the exact value is four times larger than the ICS=SAME value). The non-conservatism grows as the mission time is increased. The results for a 2000 hour mission are given in Table 3.

The error resulting from the ICS=SAME truncation method can be shown to be negligible for many systems. Thus, the HARP ICS=SAME technique can be used successfully if there is

[1] FTMP often had difficulty isolating the faults in such situations. However, FTMP, unlike this system, did reconfigure in this situation.

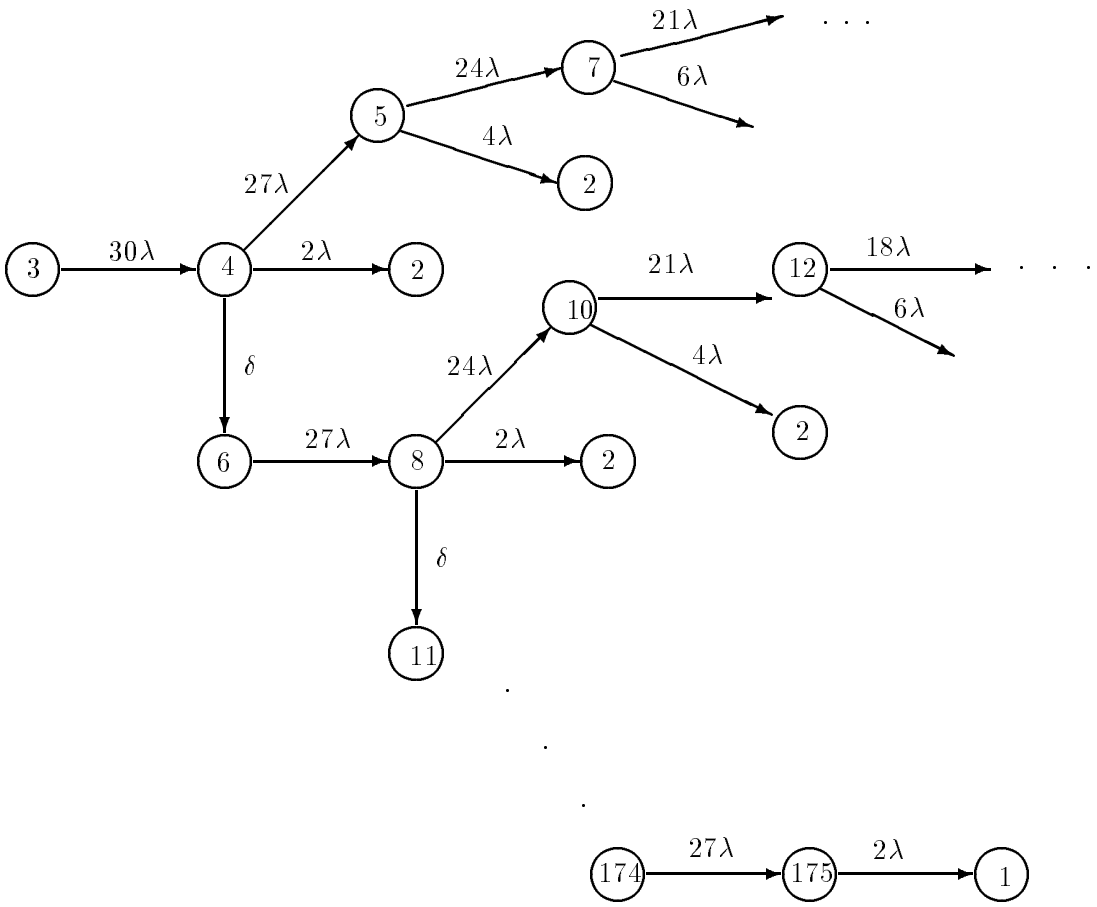


Figure 13: Ten Triads—Part of the Model

Model		P_f estimated by PAWS
Parameters:	$\lambda = 10^{-4}$ per hour $\delta = 10^{4}$ per hour Time $= 2 \times 10^{3}$ hours	
Full Model (Figure 3)		6.62×10^{-7}
ICS=Same Model		1.05×10^{-7}

Table 3: P_f at 2000 Hours for (10,1) Configuration

assurance that the probability contribution from all of the omitted transitions is insignificant. This can be accomplished by performing some additional hand calculations [3, 2]. [2] However, one should be aware that ICS=SAME can lead to non-conservative results. The HARP program does not warn the user when the omitted transitions are significant. *Thus, without some additional analysis, one does not know when the ICS=SAME technique is non-conservative.*

4.0.3 The HARP input

The model shown in Figure 13 can be converted into a HARP "FORM model" by removing all of the recovery states and the transitions emanating from them. This is illustrated in Figure 14.

Figure 14: HARP FORM for Ten Triads (ICS=ALL)

Although this FORM could be used in an ICS=ALL analysis, it cannot be used if one wishes to specify ICS=SAME. In order for HARP to determine the interfering failure rate, each triad must be given a unique name and symbolic failure rate. Thus, there would be 10 symbolic rates: $\lambda_1, \lambda_2, \lambda_3, \cdots \lambda_{10}$. The FORM would have 10 failure transitions from the start state, each with failure rate λ_i. Part of this FORM is shown in Figure 15.

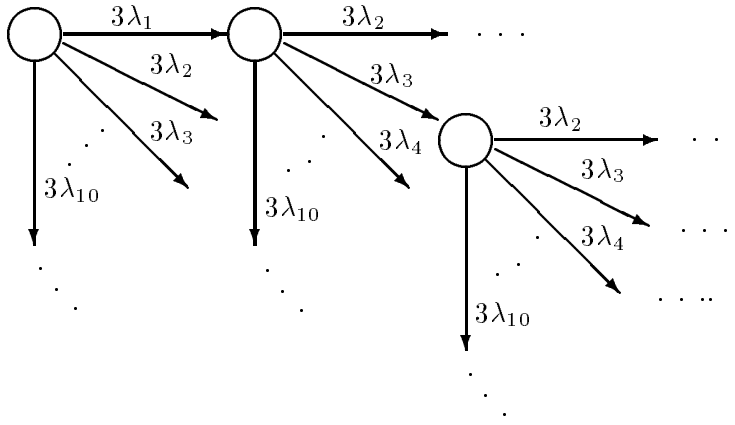


Figure 15: HARP FORM for Ten Triads (ICS=SAME)

5 HARP Solution Method—Behavioral DecompositionXFG

The HARP program solves the merged FORM/FEHM model using the technique of behavioral decomposition. A brief overview is given here. The basic idea of behavioral decomposition is to

[2] For small systems the number of transitions that are omitted is usually small and the required hand calculations are relatively simple. In large systems, there are often large numbers of transitions that are omitted by HARP and the cumulative effect can be significant. These systems are also more difficult to analyze by hand calculation than a small system.

solve the FEHM in isolation to determine the exit probabilities and then replace the distributions of the sojourn times of the FEHM with instantaneous jumps. The FORM model is then augmented with coverage parameters from the FEHM calculations. For example, the model of Figure 2 is converted into the form shown in Figure 16. This "instantaneous jump" model is a pure Markov

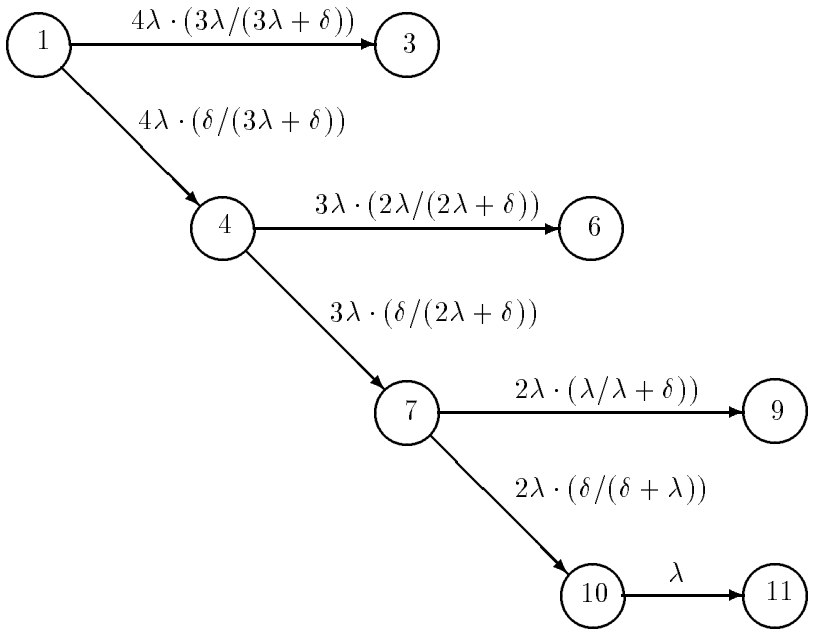


Figure 16: HARP Instantaneous Jump Model Of a Degradable Quad

model that is solved using a differential equations g package.

It should be noted that the instantaneous jump procedure uses a conservative approximation, which may *mask* the non-conservative trimming error. For example, if the exact answer to a Markov model is A, the trimming error is e_t, and the instantaneous-jump error is e_j, then the computed answer would be $A - e_t + e_j$. If the magnitude of e_j is greater than e_t, then the computed error would still be conservative. We do not know in practice under which conditions e_j will exceed e_t and hence when a user of HARP will actually see non-conservative answers. Although the potential for a non-conservative answer exists, the experimental studies have not been performed on HARP V7.0 to delineate the regions where this occurs. Therefore it is up to the user of HARP to make sure that the e_t trimming error is not too large. It should be noted that the computations in the previous section were not made on the instantaneous jump model that HARP uses to make its final calculations.

6 Conclusion

The HARP manual warns the user that the HARP program can drop failure modes for certain systems. This note provides an explanation of how this "state trimming" occurs. An error bound for this trimming has been developed [3, 2], but is not currently implemented in the HARP program. It is recommended that this bound be manually calculated to insure that the trimming is not excessive.

7 Acknowledgement

All of the results of this paper were derived from the pioneering work of Dr. Alan White, NASA Langley Research Center in developing a precise mathematical description of the HARP program.

References

[1] Salvatore J. Bavuso, et. al.: *HiRel: Hybrid Automated Reliability Predictor (HARP) Integrated Reliability Tool System (Version 7.0).* NASA Technical Paper 3452, Nov. 1994.

[2] White, Allan L.; and Palumbo, Daniel L.: State Reduction for Semi-Markov Reliability Models. In *The 36th Annual Reliability and Maintainability Symposium*, Los Angeles, CA, Jan. 1990.

[3] White, Allan L.; and Palumbo, Daniel L.: *Model Reduction by Trimming for a Class of Semi-Markov Reliability Models and the Corresponding Error Bound.* NASA Technical Memorandum 3089, May 1991.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1998	Technical Memorandum

4. TITLE AND SUBTITLE

A Note About HARP's State Trimming Methods

5. FUNDING NUMBERS

WU 522-33-31-01

6. AUTHOR(S)

Ricky W. Butler; Kelly J. Hayhurst; Sally C. Johnson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

NASA Langley Research Center
Hampton, VA 23681-2199

8. PERFORMING ORGANIZATION REPORT NUMBER

L-17684

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

NASA/TM-1998-208427

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Unclassified-Unlimited
Subject Category 65 Distribution: Standard
Availability: NASA CASI (301) 621-0390

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

This short note provides some additional insight into how the HARP program works. In some cases, it is possible for HARP to trim away too many states and obtain an optimistic result. The HARP Version 7.0 manual warns the user that "Unlike the ALL model, the SAME model can automatically drop failure modes for certain system models. The user is cautioned to insure that no important failure modes are dropped; otherwise, a non-conservative result can be given." This note provides an example of where this occurs and a pointer to further documentation that gives a means of bounding the error associated with trimming these states.

14. SUBJECT TERMS

Reliability Analysis, Fault Tolerance, Markov Models

15. NUMBER OF PAGES

21

16. PRICE CODE

A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	