Proceedings of the 1985 Pressure Vessels and Piping Conference

Computer-Aided Engineering

presented at
THE 1985 PRESSURE VESSELS AND PIPING CONFERENCE AND EXHIBITION
NEW ORLEANS, LOUISIANA
JUNE 23-26, 1985

sponsored by
PRESSURE VESSELS AND PIPING DIVISION
COMPUTER TECHNOLOGY COMMITTEE

Contributing Editors
H.H. FONG
F.M. SKALAK
W.K. LIU
J.Y. LEUNG

Volume Coordinating Editor
K.H. HSU

General Proceedings Editor
S.J. BROWN

THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
United Engineering Center 345 East 47th Street New York, N.Y. 10017
The procedures for analog Monte Carlo simulation of Markov processes are examined. Two variance reduction techniques are then included in a nonanalog formulation to increase the sampling efficiency for highly reliable systems, and a method for incorporating uncertainty in failure and repair rates data is outlined. Models for three classes of component dependencies appearing in reliability and availability problems are incorporated into the Markov formulation. They are (1) shared repair crews between components, (2) load sharing between components, and (3) standby mode. Results are given for a series of model problems to demonstrate the efficiency of the methods as well as the effects of the dependencies on system unreliability and unavailability.
ABSTRACT

The procedures for analog Monte Carlo simulation of Markov processes are examined. Two variance reduction techniques are then included in a nonanalog formulation to increase the sampling efficiency for highly reliable systems, and a method for incorporating uncertainty in failure and repair rate data is outlined. Models for three classes of component dependencies appearing in reliability and availability problems are incorporated into the Markov formulation. They are (1) shared repair crews between components, (2) load sharing between components, and (3) standby nodes. Results are given for a series of model problems to demonstrate the efficiency of the methods as well as the effects of the dependencies on system unreliability and unavailability.

I INTRODUCTION

The estimate of reliability and/or availability of systems with large numbers of components in highly redundant configurations frequently is required in the analysis of a variety of safety and protection systems. This analysis most often proceeds by first constructing a fault tree [1,2] to determine the possible combinations of primary component failures that will result in system failure. Then the reliability or availability of the system is estimated quantitatively in terms of the failure and repair rates of the components.

The quantitative evaluation may be carried out by standard deterministic methods, provided the probabilities of component failure and repair are mutually independent. However, the task becomes prodigious in the presence of component dependencies such as appear with backup systems, repair crews shared between components, or components with shared loading. In such situations the system may be modeled as a Markov process [3-5], for then such dependencies can be properly represented, provided the failure and repair rates are time-independent. It then follows, however, that a system of \(2^N\) coupled first-order differential equations must be solved numerically if a system with \(N\) components is to be modeled. Since this represents over a million differential equations for a system of only twenty components, numerical integration of other deterministic approaches rapidly becomes intractable.

Markov Monte Carlo sampling [6,7] provides means of circumventing the detailed numerical solution of the coupled differential equations that can be very effective, provided only a few integral system parameters, such as reliability and availability, are to be estimated. Moreover, for highly reliable systems for which analog Monte Carlo sampling is very inefficient, powerful variance reduction schemes [8] may be incorporated into Markov Monte Carlo to increase computational efficiency by orders of magnitude. Finally, the large uncertainties that are inherent to most failure and repair rate data may be taken into account systematically such that the variance of the result can be divided into that associated with data uncertainty and that due to the finite number of Monte Carlo trials.

In what follows we first set forth the basis for analog Monte Carlo simulation of Markov processes. Two variance reduction techniques, forced failure and failure biasing, are then introduced to increase computational efficiency. The treatment of data uncertainty is examined, and the paper is concluded with numerical results for a series of example problems that differ by the types of component dependencies that are taken into account.

2. ANALOG MONTE CARLO

Each of the \(2^N\) Markov states for an \(N\) component system is defined by a unique combination of functional and failed components. The probability density function that a system is in state \(k'\) at time \(t\) will make a state transition at time \(t'\) is

\[f(t'|t,k') = \exp[-\lambda_k(t-t')]
\]

where \(\lambda_k\) is the transition rate out of state \(k'\).

In the case where there are no component dependencies, \(\lambda_k\) is the sum of the failure rates of the functional components and the repair rates of the failed components. In the event that there are component dependencies, certain of the failure or repair rates will depend on the states of other
components. To sample the time interval \( \Delta t = t - t' \) between state transitions we obtain the cumulative probability distribution corresponding to (1), set it equal to a random number \( \xi \), that is uniformly distributed between zero and one, and invert to obtain [8]

\[
\Delta t = -\frac{1}{\lambda} \ln(1 - \xi).
\]

Having determined the time of transition, we must determine the new state of the system. The transition probability \( q(k' | k) \) from state \( k' \) to state \( k \) is just \( 1/|k| \), multiplied by the failure or repair rate of the component that must change states in order for the \( k' + k \) transition to take place. To sample the transition, a uniformly distributed random number \( \xi' \) is generated and the new state is determined by the value of \( k \) that satisfies the inequality

\[
k = k + 1 \quad \frac{1}{q(k' | k')} < \xi' < \frac{1}{q(k' | k')}.
\]

The most general version of the Markov code estimates either unreliability and unavailability from \( N \) independent trials. Each trial consists of alternate samplings of the times of the successive transitions and the state of the system after the transitions. For unreliability calculations the trial ends when the design life \( T \) is exceeded. For unavailability calculations the trial ends at the first system failure or when \( T \) is exceeded. After each transition, the system status is checked to determine whether the system is in an operational or a failed state. Tallys are required only after transitions for which there is a change in system status between a failed and an operational state.

For purposes of determining its status, the system is represented by an equivalent fault tree. The status is then determined either by comparing the combination of failed components to the tree's minimal cut sets, which have previously been determined, or by direct bottom-up logical evaluation of the tree [1].

In analog Monte Carlo the unreliability tally is binomial, consisting of a one for each trial resulting in system failure before \( t = T \), and zero otherwise. For the unreliability, the tally is just the fraction of the time \( T \) for which the system is in a failed state. Thus if \( M \) trials are carried out, and \( m \) of them are found to result in system failure, the binomial estimator for the unreliability is

\[
U = m/N,
\]

and the corresponding sample variance is

\[
S^2(U) = \frac{1}{M-1} \sum \left( 1 - U \right).
\]

Likewise, if \( t/T \) is the fraction of the time \( T \) that the system is in a failed state during the \( m \)th trial, the unavailability is estimated by

\[
U = \frac{1}{M} \sum \frac{t_m}{T}
\]

with a sample variance of

\[
S^2(U) = \frac{1}{M-1} \sum \left( \frac{t_m}{T} - U \right)^2.
\]

Then, according to the central limit theorem the 68% confidence interval for the results is \( U \pm S \).

**VARIANCE REDUCTION**

Even with the improved sampling that results from the Lagrangian approach of the Markov simulation, analog Monte Carlo analysis of highly reliable systems is likely to be very expensive, since only very rarely will a trial contribute a nonzero tally. Fortunately, powerful variance reduction techniques are easily adapted to the Markov Monte Carlo formalism. These techniques, which are analogous to the highly refined importance sampling methods employed in neutron transport calculations [8-10], greatly increase the computational efficiency for highly reliable systems, without biasing the results.

We employ two such techniques, which we refer to as forced transitions and failure biasing. In these the sampling distributions are modified, first to produce an artificially large number of component transitions, and second to increase the ratio of failures to repairs. To each trial is attached a weight, initialized to one, that is modified appropriately each time a biased sampling distribution is used. Then by defining weighted tallies, unbiased estimators may be shown to result.

In forced transitions, we replace \( f(t, t', k') \) by

\[
f(t, t', k') = \begin{cases} e^{-\gamma(t-t')} & t' < t < T \\ 1 - e^{-\gamma(T-t')} & t > T \\ 0 & \text{otherwise} \end{cases}
\]

With this, the uniformly distributed random number \( \xi \) can be used to sample the interval to the next transition:

\[
\Delta t = -\frac{1}{\lambda} \ln \left( 1 - \xi(1 - e^{-\gamma(T-t')}) \right), \quad 0 < \Delta t < T - t',
\]

causing the next transition to be forced before the end of design life. To compensate for the modified sampling the trial weight \( w_m \) is modified by

\[
w = \nu(1 - e^{-\gamma(T-t')}).
\]

This technique is only applied when \( \gamma(T-t') \) is small, and therefore there is only a small change of analog transition before the end of design life. When there are large transition rates, due for example to the presence of large repair rates in \( \gamma \), then analog sampling is used. Conversely, when \( \gamma(T-t') \) is very small, a rare event approximation may be used to simplify the modified probability density function to

\[
f(t, t', k') = \frac{1}{(T-t')} \quad t' < t < T.
\]

Therefore \( \Delta t \) can be sampled from a uniform distribution, and the weight is modified by

\[
w = \gamma(T-t')w.
\]

In failure biasing the transition probabilities \( q(k' | k) \) are modified to increase the ratio of failures to repairs. This is necessary for efficient computation, since normally the repair rates are an order of magnitude or more larger than the failure rates. The biased transition probabilities are taken as

\[
q(k' | k) = q(k' | k) \frac{1}{\sum q(k' | k)}
\]

for \( k \epsilon A \).
and

\[ q(k|k') = \frac{1}{\lambda} q(k'|k) (1 - \lambda), \quad k \in A \]

Here \( A \) is used to indicate the set of transitions out of state \( k' \) that result from component failures, and \( \lambda \) indicates those that result from repairs. These biased probabilities are defined such that a fraction \( \lambda \) of the transitions are forced to be failures. Typically, we take \( \lambda \) to be at least one half, which is much larger than normally would be the case with unbiased sampling. To maintain unbiased results the trial weight is modified by

\[ w + \frac{1}{1-\lambda} q(k'|k') \]

for component failure and

\[ w + \frac{1}{\lambda} q(k'|k') \]

for repairs.

In using these variance reduction techniques, the trial weight is appropriately modified at each biased sampling and the first system failure occurs. The weight \( w_k \) for the \( m \)th history is then tabulated, the variance reduction technique \( \lambda \) is turned off for the remainder of trial, and the trial is completed using analog Markov Monte Carlo. This combination of biased and analog sampling is found to be ideal, for it results in a substantial fraction of the trials contributing nonzero tallies to the results. At the same time the tally procedure for the unavailability is simplified, and one does not encounter the problem of very long trails, with insignificantly small weights resulting from many system failures.

With the foregoing importance sampling procedures, the estimate for the unreliability is just

\[ U_T = \frac{1}{M} \sum_{m=1}^{M} w_m. \]

The sampling variance is then

\[ S^2(U_T) = \frac{1}{M-1} \sum_{m=1}^{M} (w_m - U_T)^2. \]

The estimate for the unavailability is

\[ U_a = \frac{1}{M} \sum_{m=1}^{M} \frac{w_m}{T}, \]

where \( T \) is just the fraction of the \( m \)th trial for which the system is in a failed state. For the unavailability, the sample variance is

\[ S^2(U_a) = \frac{1}{M-1} \sum_{m=1}^{M} \left( \frac{w_m}{T} - U_a \right)^2. \]

The positive effects of the variance reduction techniques on computational efficiency are most pronounced for highly reliable systems. The improvement may be measured in terms of the standard figure of merit for Monte Carlo calculations:

\[ 1/6 T, \]

where \( T \) is the time per history, is a standard measure of Monte Carlo computational efficiency.

Improvements of more than three orders of magnitude or more are obtainable [6,7].

4. DATA DISTRIBUTIONS

The foregoing results are for point data on the failure and repair rates. Data uncertainty can also be factored into the calculations in a rational way. Each failure or repair rate is represented by a probability density function which is sampled at the beginning of a batch of \( H \) histories. The calculation is then repeated for \( N \) batches and the unreliability is determined from

\[ U_T = \frac{1}{N} \sum_{n=1}^{N} U_{Tn}, \]

where \( U_{Tn} \) is the estimate for the \( n \)th batch. Likewise, for the unavailability

\[ U_a = \frac{1}{N} \sum_{n=1}^{N} U_{an}, \]

where \( U_{an} \) is the unavailability estimate for the \( n \)th batch. The corresponding sample variances are given by

\[ S^2(U_T) = \frac{1}{N-1} \sum_{n=1}^{N} (U_{Tn} - U_T)^2, \]

and

\[ S^2(U_a) = \frac{1}{N-1} \sum_{n=1}^{N} (U_{an} - U_a)^2. \]

An important point is that variance of either result may be cast in the form [8]

\[ S^2 = \frac{1}{M} A + B, \]

provided that \( M \), the number of trials per batch, is large enough that the batch averages form a normal distribution. The first term is due to the finite number of histories per batch, and the second is due to the data uncertainty. Thus the batch sizes need be made only large enough that the second term dominates. For then the variance in the result is governed by the data uncertainty and not the finite batch size. In reliability and availability problems, where the data uncertainties are often substantial, quite moderate batch sizes often meet this criterion.

5. RESULTS

In illustrating the effects of component dependencies on system unreliability and unavailability, a model problem is defined by the fault tree shown in Fig. 1. This problem has been studied in the absence of component dependencies using both deterministic [11] and Markov Monte Carlo [6] methods. The component failure and repair rates are given in Table 1. System failure is determined by bottom-up logical evaluation of the tree [11] after each transition.

For purposes of dependency modeling the components are divided into four groups as indicated in Table 1. Using fixed-failure and repair rate data, we examine each of the dependency types individually. In all cases we calculate unreliability using a design life of \( T = 1,000 \) hours, and in the failure biasing a value of \( x = 0.9 \) is used. All quoted times are on a CYBER 205 without vectorization.
In Table 2 are shown the effects of shared repair crews on the unavailability and unreliability. In the base case there are no dependencies since each of the components has a repair crew. Note that the great increases in unreliability and unavailability in the problem result in going from two to one repair crew per set, while practically no deterioration results from going from one to none.

In Table 3 results are shown for the case where within each of the four component groups there is load sharing, such that the failure of any one component within the group increases the failure rate of the remaining components.

In Table 4 results are given for standby configurations. Component groups 1 and 2 are (2/3) systems with the third component held in standby, while component groups 3 and 4 are (1/2) standby systems. The effects of various levels of the standby failure rate, \( \lambda^s \), and of a switching failure probability of 5% are shown. In all cases \( \mu^s \), the repair rate for switching failures, is taken to be 0.5 hr.

Finally, in realistic calculations the data uncertainty is likely to be large, contributing a variance to the results that may be substantially larger than that due to the fact that only a finite number of Monte Carlo trials have been run. To demonstrate this effect the system without component dependencies is run with data uncertainties. For each component the failure and repair rates are represented by lognormal distributions. For all components the uncertainty factor in both failure and repair rates is set equal to three. The results are compared in Table 5. In the calculations without uncertainty 10,000 trials were used. With uncertainty, 1,000 batches of 25 trials per data batch were employed. With this level of uncertainty it is seen that roughly three quarters of the uncertainty in the results is due to data uncertainty, and not the limited number of Monte Carlo trials. The results are altered very little if increased numbers of batches are used.

Unless otherwise specified the foregoing calculations each consist of 10,000 Monte Carlo trials. For the example problems studied here the presence of dependencies never increases the running time by as much as a factor of two. Without data uncertainty running time is roughly ten thousandths of a second per trial or a few seconds per run. With data uncertainty, times of less than one hundredth of a second per batch of 25 typically are required. The code is also capable of treating problems in which all three dependency types are present, a... in which data uncertainty is included.

ACKNOWLEDGEMENT

This work was supported in part by the U.S. Air Force Office of Scientific Research, Contract AFOSR-84-0340.

REFERENCES


Table 1. Data for example problem

<table>
<thead>
<tr>
<th>4</th>
<th>group</th>
<th>( \lambda^s \times 10^{-5} \text{ hr}^{-1} )</th>
<th>( v^s \text{ (hr}^{-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0.26</td>
<td>0.042</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.26</td>
<td>0.042</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.26</td>
<td>0.042</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3.5</td>
<td>0.17</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>3.5</td>
<td>0.17</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>3.5</td>
<td>0.17</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>0.5</td>
<td>0.0</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>0.5</td>
<td>0.0</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>0.8</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>0.8</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Table 2. The Influence of the Number of Repair Crews on Unreliability and Unavailability

<table>
<thead>
<tr>
<th>Number of repair crews N</th>
<th>Unreliability</th>
<th>Unavailability</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Group 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>.3517x10^2</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>.3517x10^-2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>.3515x10^-5</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>.3515x10^-5</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>.3492x10^-5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>.4394x10^-7</td>
</tr>
</tbody>
</table>

Table 3. The Influence of Sharing Load on Unreliability and Unavailability

<table>
<thead>
<tr>
<th>Δλ</th>
<th>Unreliability</th>
<th>Unavailability</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>.5278x10^-4</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>.6584x10^-4</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>.7929x10^-4</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>.8778x10^-4</td>
<td></td>
</tr>
<tr>
<td>2.0</td>
<td>.1319x10^-3</td>
<td></td>
</tr>
</tbody>
</table>
Table 4. The Influence of failures in standby and of switching failures on system Unreliability and Unavailability

<table>
<thead>
<tr>
<th>p</th>
<th>0.00</th>
<th>0.05</th>
<th>0.00</th>
<th>0.05</th>
</tr>
</thead>
<tbody>
<tr>
<td>λ/λ</td>
<td>Unreliability</td>
<td>Unavailability</td>
<td>Unreliability</td>
<td>Unavailability</td>
</tr>
<tr>
<td>0.0</td>
<td>0.2949·10⁻⁵</td>
<td>23·10⁻⁷</td>
<td>0.9350·10⁻⁷</td>
<td>23·10⁻⁹</td>
</tr>
<tr>
<td>0.2</td>
<td>0.3245·10⁻⁴</td>
<td>24·10⁻⁷</td>
<td>0.9762·10⁻⁷</td>
<td>22·10⁻⁹</td>
</tr>
<tr>
<td>0.4</td>
<td>0.3563·10⁻⁶</td>
<td>24·10⁻⁷</td>
<td>1.084·10⁻⁶</td>
<td>22·10⁻⁹</td>
</tr>
<tr>
<td>0.6</td>
<td>0.3827·10⁻⁶</td>
<td>25·10⁻⁷</td>
<td>1.119·10⁻⁵</td>
<td>23·10⁻⁹</td>
</tr>
<tr>
<td>0.8</td>
<td>0.4142·10⁻⁶</td>
<td>24·10⁻⁷</td>
<td>1.263·10⁻⁵</td>
<td>23·10⁻⁹</td>
</tr>
<tr>
<td>1.0</td>
<td>0.4429·10⁻⁶</td>
<td>25·10⁻⁷</td>
<td>1.393·10⁻⁶</td>
<td>24·10⁻⁹</td>
</tr>
</tbody>
</table>

Table 5. Effect of Data Uncertainty on Unreliability and Unavailability

<table>
<thead>
<tr>
<th>Uncertainty</th>
<th>Unreliability</th>
<th>Unavailability</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>0.4394·10⁻⁶ ± 0.5·10⁻⁶</td>
<td>0.1436·10⁻⁶ ± 0.5·10⁻⁸</td>
</tr>
<tr>
<td>Yes</td>
<td>0.6708·10⁻⁶ ± 0.2·10⁻⁵</td>
<td>0.3342·10⁻⁶ ± 0.2·10⁻⁷</td>
</tr>
</tbody>
</table>

Fig. 1. Example Problem Fault Tree