REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

(2)

1. AGENCY USE ONLY (Leave blank)
2. REPORT DATE
3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
NUMERICAL EVALUATION OF PERFORMABILITY AND JOB COMPLETION TIME IN REPAIRABLE FAULT-TOLERANT SYSTEMS.

5. FUNDING NUMBERS
AFOSR-84-0132
and
AFOSR-84-0140

6. AUTHOR(S)
V.G. Kulkarnil, V.F. Nicola, R.M. Smith, and K.S. Trivedi

AEOSR-TR- 90 1009

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Computer Science
Duke University
Durham, NC 27706

8. PERFORMING ORGANIZATION REPORT NUMBER
61102F
2304/K3

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR
Bld 410
Bolling AFB DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Fault-tolerant computer systems change their level of performance (e.g., mode of operation or service rate) in response to different events such as failure, degradation or repair. We present a unified model for the analysis of job (task) completion time and the accumulated service (reward) until a given time (also known as performability). In prior work, the evaluation of the distribution of performability was restricted to nonrepairable systems (represented by acyclic Markov chains). In this paper, we describe an algorithm for the numerical evaluation of the distributions of performability or job completion time, in repairable fault-tolerant systems (represented by cyclic Markov chains). We demonstrate the feasibility of our techniques by means of numerical examples.

14. SUBJECT TERMS

15. NUMBER OF PAGES

16. PRICE CODE

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

NSN 7540-01-280-5500

Standard Form 298 (890104 Draft)

Proc IEEE [illegible] Symp. on Fault-Tolerant Comp Systems

NUMERICAL EVALUATION OF PERFORMABILITY AND JOB COMPLETION TIME IN REPAIRABLE FAULT-TOLERANT SYSTEMS*

V. G. Kulkarni[†], V. F. Nicola[‡],
R. M. Smith[‡] and K. S. Trivedi[‡]

[†] Curriculum in Operations Research and Systems Analysis,
University of North Carolina, Chapel Hill, NC 27514, USA.
[‡] Department of Computer Science, Duke University, Durham, NC 27706, USA.

Abstract

Fault-tolerant computer systems change their level of performance (e.g., mode of operation or service rate) in response to different events such as failure, degradation or repair. We present a unified model for the analysis of job (task) completion time and the accumulated service (reward) until a given time (also known as performability). In prior work, the evaluation of the distribution of performability was restricted to nonrepairable systems (represented by acyclic Markov chains). In this paper, we describe an algorithm for the numerical evaluation of the distributions of performability or job completion time, in repairable fault-tolerant systems (represented by cyclic Markov chains). We demonstrate the feasibility of our techniques by means of numerical examples.

Keywords. Degradable/Repairable Systems, Markov Reward Processes, Numerical Methods, Performability Modelling, Task Performance, Task-Oriented Reliability.

1. Introduction

The increased reliability requirements of present day systems have caused fault-tolerant and degradable systems to become more important. For these systems, it is important to introduce measures that reflect both performance and reliability of the system. Several authors have developed models for the evaluation of reliability, performability and program performance (e.g., task completion time measures). This paper is an attempt to unify these different models with a single model which is useful for assessing the behaviour of degradable/repairable computer systems.

As pointed out by Meyer [15], distributed and multiple processor systems are generally characterized by three main features: concurrency, fault-tolerance and degradable performance, furthermore, real-time systems must also possess the timeliness property. Traditional system-oriented reliability/availability models have covered the fault-tolerance aspect [19]. Job-oriented reliability models have catered to the fault-tolerance and timeliness aspects simultaneously [2,11]. System-oriented performability models have included both fault-tolerance and degradable performance in system evaluation [5,7,10,14]. The unifying aspect of this paper is that fault-tolerance, degradable performance and timeliness are addressed simultaneously. Concurrency and timeliness issues are addressed elsewhere [18].

In the model we develop, changes in the structural state of the computer system caused by different events are described by a stochastic process (referred to as the 'structure-state' process). Associated with each structure state is a reward rate (e.g., service rate or throughput). It is particularly useful for our unifying analysis to consider the execution of a particular job on the system. In this case the reward rate represents the service rate (e.g., the number of instructions executed per unit time). It is obvious that the completion time of the job is affected by the preemptions and the possible variations in the service rate due to changes in the structure state of the system. If the job service is always resumed after preemptions, then the completion time of a given job and the cumulative service (measure or reward) until a given time are dual measures so that the distribution of one allows us to compute the distribution of the other. This is a key observation because the analysis of the completion time yields the distribution of the cumulative measure (accumulated reward or performability) which can be appropriately specialized to obtain different system-oriented measures as will be shown in Section 2.

Howard [9] studied the expected accumulated reward until a given time in Markov and semi-Markov reward processes. Other authors [5,7,14] have studied the distribution of the accumulated reward in acyclic reward processes, under the assumption that preemptions do not result in a loss of work. Iyer et al. [10] considered the evaluation of the moments of the accumulated reward in cyclic Markov reward processes. In [12,13] we have considered the analysis of the job completion time in the presence of different types of preemptions, with the possibility of loss of all work (in which case the job has to be repeated). In the present paper, the distribution of the accumulated reward in cyclic and acyclic Markov reward processes is derived as a special case of the more general analysis of job completion. We give an algorithm for the numerical evaluation of the distributions of the job completion time and performability measures in repairable fault-tolerant systems. The work presented in this paper is a significant contribution, since the earlier work was restricted to nonrepairable systems.

In Section 2, we introduce the mathematical model, some definitions and notations. The transform solution of the completion time and the cumulative measure is derived in Section 3. An algorithm for the numerical evaluation of these measures is presented in Section 4. In Section 5, we give nontrivial numerical examples to demonstrate the use of the solution technique and the feasibility of the algorithm developed.

2. The Basic Model and Definitions

Consider a particular job to be processed on a given computer system. The work requirement of the job is a random variable B, and is measured in work units (e.g., the number of instructions to be executed). It has the distribution function $G(x) = P(B \leq x)$ and the LST (Laplace Stieltjes Transform)

* Work supported by Air Force Office of Scientific Research under grants AFOSR-84-0132 and AFOSR-84-0140 by Army Research Office under contract DAAG29-84-0045 and by National Science Foundation under grant MCS-830200

$\tilde{G}(s) = E(e^{-sB})$. It is assumed that $G(0+) = 0$.

The stochastic process, $\{Z(t), t \geq 0\}$, which describes the behaviour of the system in time (the structure-state process) is a time-homogeneous continuous-time Markov chain ($CTMC$). $Z(t)$ is the state of the system at time t. This stochastic process is assumed to be independent of the work requirement of the job. At any given time, the system can be in one of n states. In state i the system serves the job at a rate $r_i \geq 0$, $1 \leq i \leq n$. The set of states $1,2,...,n$ is canonically partitioned into $k+1$ sets; namely, $S_T, S_{C1}, S_{C2}, ..., S_{Ck}$, such that S_T is a set of transient states and S_{Ci}, $1 \leq i \leq k$, is a closed set of recurrent states. (If the system enters a closed set of states, it stays there forever.) A recurrent set is called a "failure" set if the reward rate is equal to zero in all its states, otherwise it is called a "nonfailure" set. If the system enters a failure set, it stays there and offers no more service (system failure). On the other hand, if the system enters a nonfailure set, it stays there and the job will eventually complete.

Let q_{ij}, $1 \leq i,j \leq n$, $i \neq j$, be the infinitesimal transition rate from state i to state j. $Q = [q_{ij}]$, $1 \leq i,j \leq n$, is the n by n generator matrix, where

$$q_{ii} = -q_i = -\sum_{\substack{j=1 \\ j \neq i}}^{n} q_{ij} .$$

Note that row sums of Q are equal to zero (i.e., $Q\underline{e} = \underline{0}$, where $\underline{e}$ is the n by 1 vector with all elements equal to one and $\underline{0}$ is the n by 1 zero vector).

Now let us introduce some important performance measures that will be used throughout the paper.

Cumulative measure, $Y(t)$, is the total reward gained in all structure states until time t (in this paper we also refer to it as the accumulated reward or performability), i.e., $Y(t) = \int_0^t r_{Z(h)}\, dh$. Y ($= \lim_{t \to \infty} Y(t)$) is the cumulative measure until system failure [1,20]. $Y(t)$ can be specialized to define the following job- and system-oriented measures.

Job completion time, $T(x)$, is the time needed to complete a job whose work requirement is x units of work. T denotes the completion time of a job that requires a random amount of work, B. Since $Y(t)$ represents the useful work done on the job until time t, it is a nondecreasing function and has piecewise continuous paths. It follows that $T(x) = \min\{t \geq 0: Y(t) = x\}$ and $T = \min\{t \geq 0: Y(t) = B\}$. The analysis of the job completion time has been considered for special cases in [2,6,16].

Probability of omission failure, $\eta(x)$, is defined to be the probability that the system fails before the completion of a job that requires x units of work. Thus

$$\eta(x) = P(Y(t) < x, \text{ for all } t \geq 0) = P(T(x) = \infty)$$

If η denotes the probability that the system fails before the completion of a job with random work requirement, then

$$\eta = P(Y(t) < B, \text{ for all } t \geq 0) = P(T = \infty) .$$

A related measure is the dynamic failure probability in real-time systems. For a hard deadline d, it is given by $\eta = P(T > d)$, and is readily obtained from the distribution function of T.

[1] $(^{\sim})$ denotes the LST, i.e., the Laplace transform of a probability density function, and $E(.)$ is the expectation operator

System reliability, $R(t)$: let X be the time until system failure, i.e., the time until the structure-state process enters a failure set of states. If we set $r_i = 1$ for all states i that are not contained in any failure set of states, then

$$R(t) = P(X > t) = \lim_{\tau \to \infty} P(Y(\tau) > t) .$$

Total "up" or "down" time until time t, $U(t)$ or $D(t)$: the system is said to be "up" if it is in a state i with $r_i > 0$, otherwise it is said to be "down". $U(t)$ (or $D(t)$) is defined to be the total time the system spends in "up" (or "down") states until time $\min\{t, X\}$, where X is the system's life time. Clearly if we set all $r_i > 0$ to 1, then

$$P(U(t) \leq x) = P(Y(t) \leq x)$$

and, since

$$U(t) + D(t) = \min\{t, X\} ,$$

it follows that

$$P(D(t) \leq x) = P(Y(t) \geq \min\{t, X\} - x) .$$

Interval availability, $A_I(t)$, is defined to be the fraction of time the system spends in "up" states in the interval $(0,t)$, i.e., $A_I(t) = U(t)/t$. The distribution of the interval availability has been a subject of recent investigations [8].

In Section 3, we derive double-transform equations for the distributions of the job completion time and the cumulative measure, and in Section 4 we describe an algorithm for the numerical evaluation of these distributions. In the remainder of this section we introduce some notations that will be used later. Define the distribution functions

$$F_i(t,x) = P(T(x) \leq t \mid Z(0) = i), \quad x \geq 0, \ 1 \leq i \leq n .$$

$$F(t,x) = P(T(x) \leq t), \quad x \geq 0,$$

$$F_i(t) = P(T \leq t \mid Z(0) = i), \quad 1 \leq i \leq n .$$

$$F(t) = P(T \leq t)$$

and the LST's

$$F_i^{\sim}(s,x) = E(e^{-sT(x)} \mid Z(0)=i), \ x \geq 0, \ 1 \leq i \leq n. \quad (2.1)$$

From the independence of $\{Z(t), t \geq 0\}$ and B it follows that

$$F^{\sim}(s,x) = E(e^{-sT(x)}) = \sum_{i=1}^{n} F_i^{\sim}(s,x)\, P(Z(0)=i), \ x \geq 0. \quad (2.2)$$

$$F_i^{\sim}(s) = E(e^{-sT} \mid Z(0)=i) = \int_0^{\infty} F_i^{\sim}(s,x)\, dG(x), \ 1 \leq i \leq n, \quad (2.3)$$

$$F^{\sim}(s) = E(e^{-sT}) = \sum_{i=1}^{n} F_i^{\sim}(s)\, P(Z(0)=i) . \quad (2.4)$$

The omission failure probability, η, follows from

$$\eta = P(T = \infty) = 1 - \lim_{s \to 0} F^{\sim}(s) . \quad (2.5)$$

3. The Transform Solution

We give the transform solution for the distribution of the job completion time in Theorem 3.1. In Theorem 3.2 we present a useful dual relationship between the cumulative measure and the completion time.

Let us first define the following transforms [1]

$$Y^*(u,t) = E(e^{-uY(t)}), \quad Y^{**}(u,s) = \int_0^\infty e^{-st}\, Y^*(u,t)\, dt ,$$

$$F^*(s,x) = E(e^{-sT(x)}), \quad F^{**}(s,u) = \int_0^\infty e^{-ux}\, F^*(s,x)\, dx .$$

The following notations will be used

$$\underline{F}^{**}(s,u) = [F_1^{**}(s,u), F_2^{**}(s,u), \ldots, F_n^{**}(s,u)]^T ,$$

$$\underline{Y}^{**}(u,s) = [Y_1^{**}(u,s), Y_2^{**}(u,s), \ldots, Y_n^{**}(u,s)]^T ,$$

$$\underline{r} = [r_1, r_2, \ldots, r_n]^T \quad \text{and} \qquad R = diag\,[r_1, r_2, \ldots, r_n] ,$$

where the superscript T denotes transpose.

Theorem 3.1:

The double transforms $F_i^{**}(s,u)$, $1 \le i \le n$, satisfy the following equations

$$F_i^{**}(s,u) = \frac{r_i}{s+q_i+r_i u} + \sum_{\substack{j=1 \\ j\ne i}}^{n} \frac{q_{ij}}{s+q_i+r_i u} F_j^{**}(s,u). \tag{3.1}$$

Since the matrix $[sI + uR - Q]$ is invertible, Equations (3.1) can be written in a matrix form as follows

$$\underline{F}^{**}(s,u) = [sI + uR - Q]^{-1}\, \underline{r} \tag{3.1a}$$

where I is the identity matrix. The proof is given in [12].

The following theorem presents a useful dual relationship between the cumulative measure at a given time and the completion time of a given job. As a result, knowing the distribution of $T(x)$ allows us to determine the distribution of $Y(t)$. Consequently, system-oriented measures such as system reliability, interval availability and others can be determined by appropriately specializing the reward rates in different structure states (as discussed in Section 2).

Theorem 3.2:

The distribution function of the cumulative measure, $Y(t)$, is related to the distribution function of the completion time $T(x)$, as follows

$$P(Y(t) < x) = 1 - P(T(x) \le t) \tag{3.2}$$

and the corresponding double transforms are related as follows

$$Y^{**}(u,s) = \frac{1}{s}\,[1 - uF^{**}(s,u)] . \tag{3.3}$$

Proof:

It is clear that

$$P(Y(t) < x) = P(T(x) > t) ,$$

since these are the probabilities of two identical events, and Equation (3.2) follows. Multiplying both sides of Equation (3.2) by e^{-ux} and integrating with respect to x, we get

$$Y^*(u,t) = 1 - u \int_0^\infty e^{-ux}\, P(T(x) \le t)\, dx .$$

Multiplying by e^{-st} and integrating with respect to t, Equation (3.3) follows. *Q.E.D.*

We can rewrite Equation (3.3) in a matrix form as follows

$$\underline{Y}^{**}(u,s) = [sI + uR - Q]^{-1}\, \underline{e} \tag{3.3a}$$

where we made use of Equation (3.1a) and the relation $Q\underline{e} = \underline{0}$. Equation (3.3a) was derived in [17] using a different approach.

4. An Algorithm for Numerical Computations

In this section we consider the evaluation of the distribution function of the cumulative measure,

$$Y_i(x,t) = P(Y(t) \le x \mid Z(0) = i), \quad 1 \le i \le n .$$

from Equation (3.3a).

Let $A = [sI + uR - Q]$, then by Cramer's rule we have

$$Y_i^{**}(u,s) = \frac{N_i(u,s)}{D(u,s)}, \quad 1 \le i \le n , \tag{4.1}$$

where $D(u,s)$ is the determinant of A, and $N_i(u,s)$ is the determinant of A with the ith column replaced by $\underline{e}$. Clearly, $D(u,s)$ and $N_i(u,s)$ are polynomials in u and s. Therefore for a fixed value of u, $Y_i^{**}(u,s)$ is a rational function in s, say $N_i(s)/D(s)$. Once the roots $s_1(u), s_2(u), \ldots, s_n(u)$ of the polynomial $D(s)$ are determined, we obtain the partial fraction expansion of $Y_i^{**}(u,s)$ and then invert analytically with respect to s. These roots are precisely the eigenvalues of $[Q - uR]$, and are determined numerically using orthogonal transformations and the QR algorithm [21]. Let $D(s)$ be given by $\prod_{j=1}^{d} (s - s_j(u))^{m_j}$, where each of its distinct roots: namely, $s_j(u)$, $1 \le j \le d$, has multiplicity m_j. Then we can write the partial fraction expansion as follows

$$Y_i^{**}(u,s) = \frac{N_i(s)}{D(s)} = \sum_{j=1}^{d} \sum_{k=1}^{m_j} a_{ijk}(u)\,(s - s_j(u))^{-k} \tag{4.2}$$

We then choose n values of s that are not too close to any $s_j(u)$, $1 \le j \le d$, and substitute in Equation (4.2). From the resulting linear system of equations, the n unknowns $a_{ijk}(u)$ are uniquely determined. Now we invert $Y_i^{**}(u,s)$ analytically with respect to s, to get

$$Y_i^*(u,t) = \sum_{j=1}^{d} \sum_{k=1}^{m_j} \frac{a_{ijk}(u)}{(k-1)!}\, t^{k-1} e^{s_j(u)t} \tag{4.3}$$

Once we have $Y_i^*(u,t)$, we invert numerically with respect to u as follows.

For notational simplicity we let $Y_i^*(u,t)$ be $V(u)$ and $Y_i(x,t)$ be $v(x)$. This will avoid the confusion between the subscript i and the radical $i = \sqrt{-1}$ that we need below. We secure the inverse of $V(u)$ with the inversion formula

$$v(x) = \frac{1}{2\pi i} \int_{a-i\infty}^{a+i\infty} e^{ux}\, V(u)\, du . \tag{4.4}$$

Now let $g(x)$ be a periodic function whose period, 2τ, is the interval of interest, such that $g(x) = e^{-ax}\, v(x)$, $0 \le x \le 2\tau$. The parameter a is chosen so that $g(x)$ is a bounded function. An approach to approximating $v(x)$ over the interval $0 \le x \le 2\tau$ is to obtain the Fourier series expansion of the function $g(x)$.

[1] (*) denotes the Laplace transform of a function

$$v(x) = \frac{e^{ax}}{\tau}\left[V(a) + \sum_{k=1}^{\infty}\left[\mathrm{Re}(V(a + k\pi i/\tau))\cos(k\pi x/\tau)\right.\right.$$

$$\left.\left. - \mathrm{Im}(V(a + k\pi i/\tau))\sin(k\pi x/\tau)\right]\right]. \qquad (4.5)$$

The above series is approximated by the first m terms. It has characteristically slow convergence, but it can be accelerated by continued fraction methods such as Wynne's ϵ algorithm, or the quotient-difference algorithm of Rutishauser with a remainder estimate suggested by De Hoog [3]; we use the latter approach.

We now give the algorithmic structure for the computation of $Y_i(x,t)$ for a system with n states.

```
A:  for ( m values of u )
    {
        determine s_j(u)  ( QR algorithm )                          O(n^3)
B:      for ( n values of s ≠ s_j(u), 1 ≤ j ≤ d )
        {
            determine N_i(u,s) ( Evaluate determinant )             O(n^3)
            determine e_ijk(u) ( Solve linear system )              O(n^3)
        }
    }
C:  for ( p different values of t )
    {
D:      for ( m values of u )
        { Evaluate (4.3) for a given time t }                       O(n)
E.      for ( q different values of x )
        {
            Evaluate finite approximation
            to (4.5), for a given (x,t) pair                        O(m)
        }
    }
```

Since the number of repetitions of the inner loop B is $O(n)$, clearly the algorithm requires $O(n^4 m)$ time to compute $Y_i(x,t)$ for a given (x,t) pair. Because most of the computational effort occurs before step C, additional values of $Y_i(x,t)$ can be computed cheaply. $Y_i(x,t)$ may be evaluated for q additional values of x at an increase of only $O(qm)$ computation time, since just loop E must be recomputed for the new values of x. To obtain $Y_i(x,t)$ for a different value of t and q different values of x requires only that loop D be performed m times and loop E be performed q times. The computational burden for the new t value is thus $O(mn + mq)$. For example, if we wish to obtain the values of $Y_i(x,t)$ for p values of t and q values of x for each value of t (as on a rectangular grid) then the pq values could be determined in $O(n^4 m + mp(n+q))$ time. It should be noted that the storage requirement is independent of the number of (x,t) pairs for which $Y_i(x,t)$ is evaluated. The matrix operations in loop B require $O(n^2)$ storage. The $e_{ijk}(u)$ evaluated for the $O(mn)$ values of $s_j(u)$ are needed to perform loop C, and m values of $Y_i(u,t)$ evaluated from (4.3) are required in loop E. Hence the total space requirement is $O(n^2 + nm)$. Accurate results are readily obtained when 80 terms are used to approximate (4.5) ($m = 80$). In the next section we give two numerical examples illustrating the feasibility of the above algorithm.

5. Examples

First, we consider a fault-tolerant multiprocessor system with finite buffer stages. A similar two-processor system (without repair) was considered by Meyer [14], and was extended by Iyer et al [10] to include repair. In [10] they describe a numerical algorithm to compute the moments (rather than the distribution) of performability. In our example we use the numerical technique described above to obtain the distribution of performability. For N processors and b buffer stages, the system is modelled as an $M/M/N/N+b$ queueing system. Jobs arrive at rate Λ and are lost when the buffer is full. The job service rate is Θ. Processors fail independently at rate λ and are repaired singly with rate μ. Buffer stages fail independently at rate γ and are repaired singly with rate τ. Processor failure causes a graceful degradation of the system (the number of processors is decreased by one). The system is in a failed state when all processors have failed or any buffer stage has failed. No additional processor failures are assumed to occur when the system is in a failed state. The model is represented by a *CTMC* with the state-transition diagram shown in Figure 1. At any given time the state of the system is (i,j) where $0 \leq i \leq N$ is the number of nonfailed processors, and j is zero if any of the buffer stages is failed, otherwise it is one. An appropriate reward rate in a given state is the steady-state throughput of the system with the given number of nonfailed processors (the throughput formula is a well known result [19]). The reward rate is zero in any system failure state.

We evaluate the distribution of performability, $Y(t)$, given that the system started with all its processors and buffers operational, for utilization period of 10 hours. The number of processors is eight, each with a failure rate $\lambda = 0.01$ per week and a repair rate $\mu = 0.1686$ per hour. The individual buffer stage failure rate $\gamma = 0.22$ per week and its repair rate $\tau = 0.1666$ per hour. Jobs arrive at rate $\Lambda = 170$ per hour and the service rate for a single processor is $\Theta = 20$ jobs per hour. In Figure 2 we plot the distribution of performability for different numbers of buffer stages. We observe that fewer buffer stages provide a lower maximum accumulated reward but a more favourable distribution of $Y(t)$ (i.e., lower values of $P(Y(t) \leq x)$, for a given x less than the maximum possible reward). The run time of our algorithm for this example (with an underlying Markov chain of 16 states) on a VAX/750 is 100 seconds. The distribution of the "up" time, $U(t)$, can similarly be evaluated by setting the reward rates in all nonfailed states to one. The complementary distribution of the interval availability, $A_I(t)$ ($= U(t)/t$), is plotted in Figure 3 for different numbers of buffer stages. The interval availability, $A_I(t)$, is lower for more buffer stages. this is due to the increase in the total buffer failure rate (notice that with more buffer stages the interval availability is not affected by the increased reward rates in "up" states). The reliability of the system, $R(t)$, can be determined by disallowing repair from all system failure states and evaluating the complementary distribution of the "up" time for infinite utilization period. This is plotted in Figure 4 for different numbers of buffer stages.

Let us now consider an example to compute the distribution of the job completion time on a two-processor (degradable/repairable) system. The system is subject to total failure (due to imperfect coverage [19,20] or exhaustion of processors). The processor failure rate is $\gamma = 0.2$, the processor repair rate is $\tau = 4.0$ and the coverage factor is $c = 0.99$. The *CTMC* representing the system is shown in Figure 5, in which the states II, I and 0 represent a system with two, one and no operational processors, respectively. It is further assumed that the job can be divided into parallel subtasks, so that if both processors are operational the service rate is increased by a factor r, $1 \leq r \leq 2$. Let the service rate in state I be $r_I = 1$, then the service rate in state II is $r_{II} = r$. We choose $r = 1.6$. Consider the execution of a job with work requirement equal to x on the system. In Figure 6 we compare the distribution of the job completion time, when executed on different systems, namely, a single processor system, a two-processor system with and without repair. In other words, we study the effect of redundancy and repair in fault-tolerant systems. The favourable effect of redundancy is obvious, and the improvement resulting from repair is clearly significant (this is reflected in a reduced job completion time and a higher probability of successful job completion; the latter is the asymptotic value of the distribution function of the completion time). In Figure 7 we show the effect of the coverage factor in a two-processor system with and without repair. It is interesting to remark that the probability of successful job completion is more sensitive to variations in the coverage factor in the presence of repair than in the absence of repair. The favourable effect of repair is again obvious.

6. Conclusions

In this paper, we have presented a unified modelling approach to the combined evaluation of performance and reliability of degradable/repairable fault-tolerant systems. The structure-state process of the system is modelled as a *CTMC*, in which transitions occur in response to events such as failure, repair or system degradation. A reward rate (or performance measure) is associated with each structure state.

We give the transform solution for the distribution of the job completion time, and relate it to the transform solution of performability. This is a useful dual relationship, since it enables us to derive from the analysis of the completion time other measures such as performability, system reliability, up/down time, and the distribution of interval availability.

We have developed an algorithm for the numerical evaluation of the distributions of performability and the completion time of a job from their corresponding transform solutions. This is a significant step towards the evaluation of repairable fault-tolerant computer systems.

References

[1] Beaudry, M. D., "Performance-Related Reliability for Computing Systems," *IEEE Trans. Comput.*, Vol. C-27, No. 6, 1978, pp. 540-547.

[2] Castillo, X. and Siewiorek, D. P., "A Performance-Reliability Model for Computing Systems," *Proc. 1980 Int. Symp. on Fault-Tolerant Computing*, Kyoto, Japan, 1980, pp. 187-192.

[3] De Hoog, F. R., Knight, J. H. and Stokes, A. N., "An Improved Method for Numerical Inversion of Laplace Transforms," *SIAM J. Stat. Comput.*, Vol. 3, No. 3, 1983, pp. 357-366.

[4] Donatiello, L. and Iyer, B. R., "Analysis of a Composite Performance Reliability Measure for Fault Tolerant Systems," IBM Research Report, RC-10325, January 1984.

[5] Furchtgott, D. G. and Meyer, J. F., "A Performability Solution Method for Degradable Nonrepairable Systems," *IEEE Trans. Comput.*, Vol. C-33, No. 6, 1984, pp. 550-554.

[6] Gaver, D. P., "A Waiting Line with Interrupted Service, Including Priorities," *J. Royal Statistical Society*, Series B24, 1962, pp. 73-90.

[7] Goyal, A. and Tantawi, A. N., "Evaluation of Performability ... Acyclic Markov Chains," IBM Research Report, RC-10529, May 1984.

[8] Goyal, A. and Tantawi, A. N., "Numerical Evaluation of Guaranteed Availability," *Proc. 1985 Int. Symp. on Fault Tolerant Computing*, Ann Arbor, Michigan, 1985.

[9] Howard, R. A., *Dynamic Probabilistic Systems, Vol. II: Semi-Markov and Decision Processes*, John Wiley, New York, 1971.

[10] Iyer, B. R., Donatiello, L. and Heidelberger, P., "Analysis of Performability for Stochastic Models of Fault-Tolerant Systems," IBM Research Report, RC-10719, September 1984.

[11] Krishna, C. M. and Shin, K. G., "Performance Measures for Multiprocessor Controllers," in *Performance '83*, A. K. Agrawala and S. K. Tripathi (eds.), North-Holland, 1983, pp. 229-245.

[12] Kulkarni, V. G., Nicola, V. F. and Trivedi, K. S., "On Modelling the Performance and Reliability of Multi-Mode Computer Systems," to appear in *Int. J. of Systems and Software*, 1986.

[13] Kulkarni, V. G., Nicola, V. F. and Trivedi, K. S., "The Completion Time of a Job on a Multi-Mode Systems," in review, *J. Appl. Prob.*, 1986.

[14] Meyer, J. F., "Closed-Form Solution of Performability," *IEEE Trans. Comput.*, Vol. C-31, No. 7, 1982, pp. 648-657.

[15] Meyer, J. F., "Performability Modelling of Distributed Real-Time Systems," in *Mathematical Computer Performance and Reliability*, G. Iazeolla, P. J. Courtois and A. Hordijk (eds.), Elsevier Science Publishers B.V. (North-Holland), 1984, pp. 361-372.

[16] Nicola, V. F., "A Single Server Queue with Mixed Types of Interruptions," to appear in *Acta Informatica*, 1986.

[17] Puri, P. S., "A Method for Studying the Integral Functionals of Stochastic Processes with Applications: I. Markov Chain Case," *J. Appl. Prob.* 8, 1971, pp. 331-343.

[18] Sahner, R. and Trivedi, K. S., "Performance and Reliability Analysis Using Directed Acyclic Graphs," in review.

[19] Trivedi, K. S., *Probability and Statistics with Reliability, Queueing and Computer Science Applications*, Prentice-Hall, Englewood Cliffs, N. J., 1982.

[20] Trivedi, K. S., "Modelling and Analysis of Fault-Tolerant Systems," in *Modelling Techniques and Tools for Performance Analysis*, D. Potier (ed.), North-Holland, Amsterdam, 1985.

[21] Wilkinson, J. H., and Reinsch, C., *Handbook for Automatic Computation, Vol. II Linear Algebra*, Springer-Verlag, 1971.

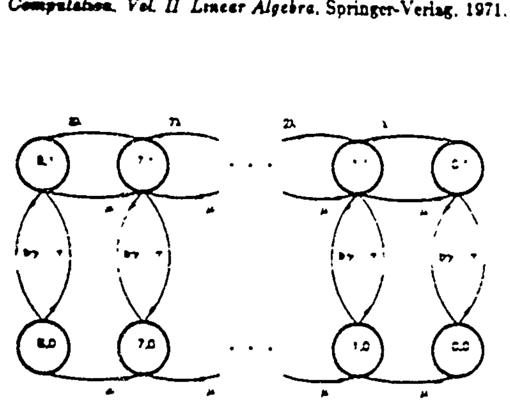

Fig. 1. State-transition diagram for an 8-processor system

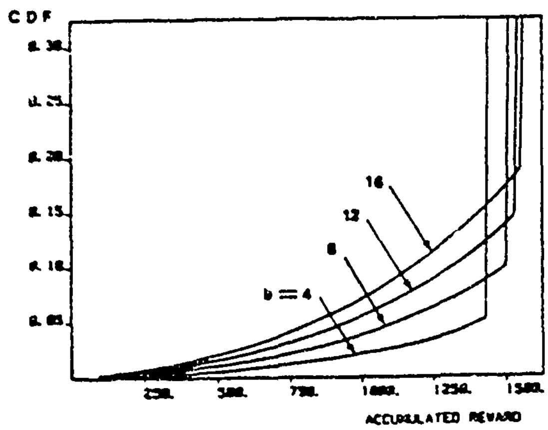

Fig. 2. The distribution of the accumulated reward, $Y(T)$, for different numbers of buffer stages (T = 10 hours)

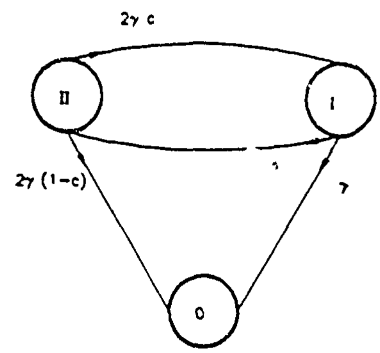

Fig. 5. State-transition diagram for a two-processor system

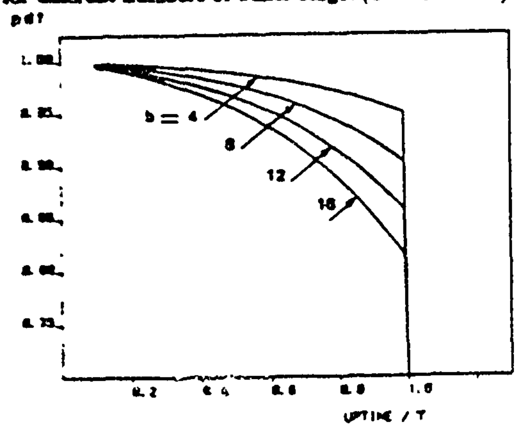

Fig. 3. The distribution of the interval availability, $A_I(T)$, for different numbers of buffer stages (T = 10 hours)

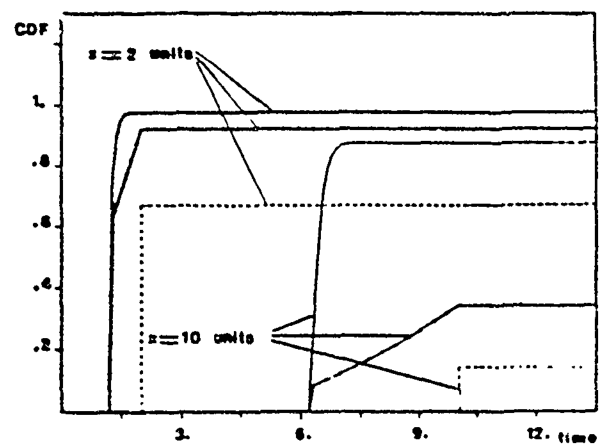

Fig. 6. The effect of redundancy and repair on the distribution of the completion time

I- Two-processor system with repair (————)

II- Two-processor system without repair (——+——)

III- Single processor system (- - - - - - - -)

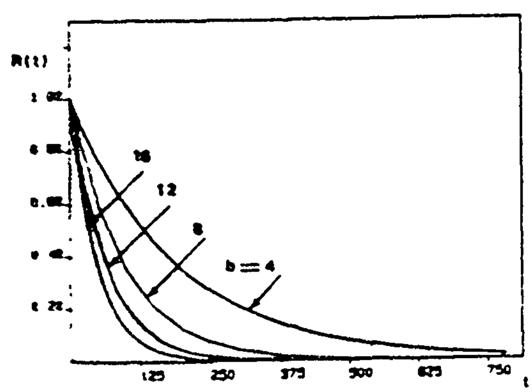

Fig. 4. The system reliability, $R(t)$, for different numbers of buffer stages

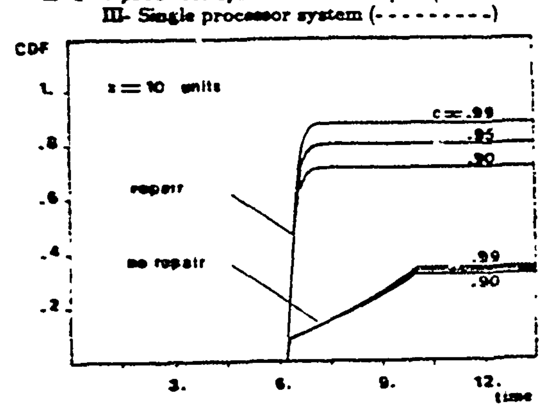

Fig. 7. The effect of the coverage factor on the distribution of the completion time