Computer Science Department Technical Report
University of California
Los Angeles, CA 90024-1596

EVALUATING BOUNDS ON STEADY STATE AVAILABILITY FROM MARKOV MODELS OF REPAIRABLE SYSTEMS

Richard R. Muntz
John C.S. Lui

June 1989
CSD-890043

Evaluating Bounds on Steady State Availability from Markov Models of Repairable Systems

Richard R. Muntz
John C.S. Lui
UCLA Computer Science Department

June, 1989

Abstract

System availability is an important reliability measure for computer system designers. Most often Markov models are used in representing systems for reliability analysis. Due to the size and complex nature of the systems, the model often has an unmanageable state space and it quickly becomes impractical to even generate all the states in the system model. In this paper, we will present a method for bounding steady state availability and at the same time, drastically reduce the state space of the models that must be solved. Here we extend the work in [7] which requires an a priori decision to be made concerning the amount of detail to represent (and therfore the tightness of the bounds). This paper extends those results to allow piecewise generation of the transition matrix such that at each step the bounds can be incrementally improved. We believe the approach may be applicable more generally but at the present we require certain assumptions that are valid (and reasonable) for reliability models. It will require extensions to relax these assumptions and therefore we present the results in this context.

1 Introduction.

System availability and reliability are crucial measures for system designers, particularly life critical situations or when large financial loss is possible. In order to evaluate reliability measures for these complex systems, Markov models are often used [4]. Unfortunately, Markov models of real systems generally have a very large state space and it becomes impractical even to generate the entire transition rate matrix. Various methods have been proposed to calculate various performance and reliability measures from large Markov models. Among these are aggregation-disaggregation method for chains with *nearly completely decomposable structure* [1], the iterative aggregation-disaggregation method [11] [12], matrix-geometric based methods [8] and the successive overrelaxation method in [10].

A Markov chain is nearly completely decomposable if the interactions between groups of states are not comparable with the interactions within the groups. This is usually not the case for reliability models of computer systems and the aggregation-disaggregation method is unlikely to yield a sufficiently accurate approximation. The iteration aggregation-disaggregation method uses a method similar to the Gauss-Seidel iteration technique. One problem is that the convergence rate may be slow. Also, existence and uniqueness are difficult to show. The matrix-geometric techniques require certain regularity of structure that is often missing from reliability models. The SOR method does not provide bounds on the approximation. We propose a method that allows for efficient calculation of bounds on steady state availability by explicitly taking advantage of the properties of reliability models.

Exact reliability measures are generally not required but rather sufficiently accurate approximate results. If the approximate results can be given in terms of upper and lower bounds then this is ideal. Recently, the results reported in [7] provided a way to compute bounds on steady state availability for models of repairable computer systems. Rather than generating all the states in the system model, the approach calls for generating those states that account for most of the probability mass while all other states are grouped into a small number of aggregate states. It is shown that bounds can be obtained by suitably replacing some transition rates out of aggregate states by lower bounds on those rates and replacing others by upper bounds. This approach is "one step" in that a decision is made a priori as to how much of the transition rate matrix to expand in detail. In this paper, we extend the method to a multi-step procedure in which we successively generate more of the transition rate matrix. At each step we obtain tighter bounds on steady state availability. The following section will provide a more detailed description of the problem and the work reported in [7]. We present some background in Section 2 and Section 3, the procedure itself in Section 4 and a discussion of the results in Section 5.

2 Background.

We are interested in the availability analysis for computer/communication systems. The behavior of such a system is assumed to be specified by a continuous time, discrete state, homogeneous Markov process. Unfortunately, the characteristics of availability models (complex interactions between components, scheduling and maintenance policies, complex criteria for a system to be operational, etc.) preclude the possibility of closed form solutions in general. Thus numerical solution methods are most widely used. Yet one of the major limitations of numerical solution is the large state space cardinality of realistic models. A real life availability model can have tens of millions of states and thereby outstripping memory and processor capacities. In this paper, we present an algorithm which can provide enormous state space reduction for numerical solution and perhaps more importantly, also provide error bounds.

Many performance and reliability measures can be expressed in terms of a reward function. If $r(i)$ is the reward rate for state i, the expected reward rate, $\mathcal{M}$ can be expressed as :

$$\mathcal{M} = \sum_{i \in S} r(i)P(i)$$

with S being the state space of the Markov model and $P(i)$ being the steady state probability of state i. Availability is a special case in which the "operational" states have reward of 1 and the "non-operational" states have reward of 0.

Systems are generally designed to have a high level of availability. It is reasonable therefore to expect that during the life time of the system, most of the components are operational. With this in mind, we partition the state space of the model into $\mathcal{F}_i$, $0 \leq i \leq n$, where n is the number of system components and where $\mathcal{F}_i$ contains all the states that have *exactly* i failed components. The idea is to represent the detailed behavior of the model for $\mathcal{F}_i$, $i \leq K$ for some small value of K and approximate the remainder of the model via aggregation.

The transition rate matrix can then be viewed as shown in Figure 1 in which submatrix Q_{ii} corresponds to $\mathcal{F}_i$. In this figure the submatrices denoted by 0 contain all zero elements. This is a consequence of an assumption that there is zero probability of two or more components becoming operational at exactly the same time. Note that this does not preclude multiple repair facilities or any other common feature of reliability models. (In particular note that the case in which a dormant component that becomes active due to the repair of a second component does not violate the assumption. We simply do not consider such a dormant component as failed in the state partitions.)

We now summarize one of the results from [7]. Consider three sets of states:

$$\begin{bmatrix} Q_{00} & Q_{01} & Q_{02} & & \cdots & & Q_{0n} \\ Q_{10} & Q_{11} & Q_{12} & & \cdots & & Q_{1n} \\ 0 & Q_{21} & Q_{22} & & \cdots & & \\ 0 & 0 & Q_{32} & Q_{33} & & & . \\ . & & 0 & Q_{43} & Q_{44} & & . \\ . & \cdots & & \ddots & & \ddots & . \\ . & & & & & & \\ 0 & \cdots & & & 0 & Q_{n-1,n} & Q_{nn} \end{bmatrix}$$

Figure 1: Transition matrix.

$\mathcal{G}_0 = \mathcal{F}_0$

$\mathcal{G}_1 = \{\bigcup_{i=1}^{K} \mathcal{F}_i\}$

$\mathcal{G}_2 = \{\bigcup_{i=K+1}^{n} \mathcal{F}_i\}$

Figure 2 illustrates the transition matrix G in which G_{ii} is the principal submatrix corresponding to states in $\mathcal{G}_i$. (The submatrix shown as 0 is a consequence of the "nearest neighbor" property discussed previously.) Now construct a new transition matrix as shown in Figure 3. It is clear that G' is a stochastic matrix if G is stochastic. The relationship between the process defined by G and that defined by G' is illustrated in Figure 4. Basically, in the new process there are two sets of states corresponding to the states $\mathcal{G}_1$ of the original model. Let us call them $\mathcal{G}'_{1u}$ and $\mathcal{G}'_{1d}$ as shown in Figure 4. The idea behind this transformation can be explained as follows. Assume the system starts in the "all components up" state, i.e. $\mathcal{F}_0$. As components fail and are repaired the system will stay in states in $\mathcal{G}'_0$ and $\mathcal{G}'_{1u}$ until the first time that there are $K+1$ or more failed components. At this point the system is in a state of $\mathcal{G}_2$. However when the number of failed components falls to K, the system now enters a state in $\mathcal{G}'_{1d}$. (Now the notation is explainable; "u" stands for "going Up" and "d" stands for "going Down". As the number of failed components goes up, the system visits states in $\mathcal{G}'_{1u}$ and after $K+1$ failures have been reached, it visits the states in $\mathcal{G}'_{1d}$ as the number of failed components goes down.)

From the construction it is easy to show that the two transition matrices are such that the steady state probabilities of the original process(G), can be calculated from the steady state probabilities of the second process (G'). Specifically, if

$[\pi'_0, \pi'_{1_1}, \pi'_{1_2}, \pi'_2]$ is the solution of $\pi' G' = \pi'$

then

$[\pi'_0, \pi'_{1_1} + \pi'_{1_2}, \pi'_2]$ is the solution of $\pi G = \pi$.

There is a natural mapping of the states in G' to states in G. In terms of rewards, the reward function for states in G' is simply to assign the same reward as the associated state in G. It is clear then that the mean availabilities of the two systems are identical.

$$\begin{bmatrix} G_{00} & G_{01} & G_{02} \\ G_{10} & G_{11} & G_{12} \\ 0 & G_{21} & G_{22} \end{bmatrix}$$

Figure 2: Matrix G.

$$\left[\begin{array}{cc|cc} G_{00} & G_{01} & 0 & G_{02} \\ G_{10} & G_{11} & 0 & G_{12} \\ \hline G_{10} & 0 & G_{11} & G_{12} \\ 0 & 0 & G_{21} & G_{22} \end{array}\right]$$

Figure 3: Duplication of states. Matrix G'.

Now consider aggregation of the states $\mathcal{F}_i$, $i \geq K+1$ and $\mathcal{F}'_i$, $1 \leq i \leq K$ as illustrated in Figure 5. The corresponding transition rate matrix is shown in Figure 6. The results from [7] show that if the rates indicated by a '+' are replaced by upper bounds (on the actual values) and the rates indicated by '−' are replaced by lower bounds then bounds on the steady state availability, $\mathcal{A}$, can be expressed as:

$$\sum_{i \in D} r(i)P'(i) \quad \leq \quad \mathcal{A} \quad \leq \quad \sum_{i \in D} r(i)P'(i) + \left(1 - \sum_{i \in D} P'(i)\right)$$

where $\mathcal{D} = \mathcal{G}_0 \bigcup \mathcal{G}_1$ and

$P'(i)$ is the steady state probability for state i in the model with the indicated replacement of transition rates by bounds.

Since the above result is valid regardless of the set of operational states it follows trivially that:

$$P'(i) \leq P(i) \quad \forall \; states \; i$$

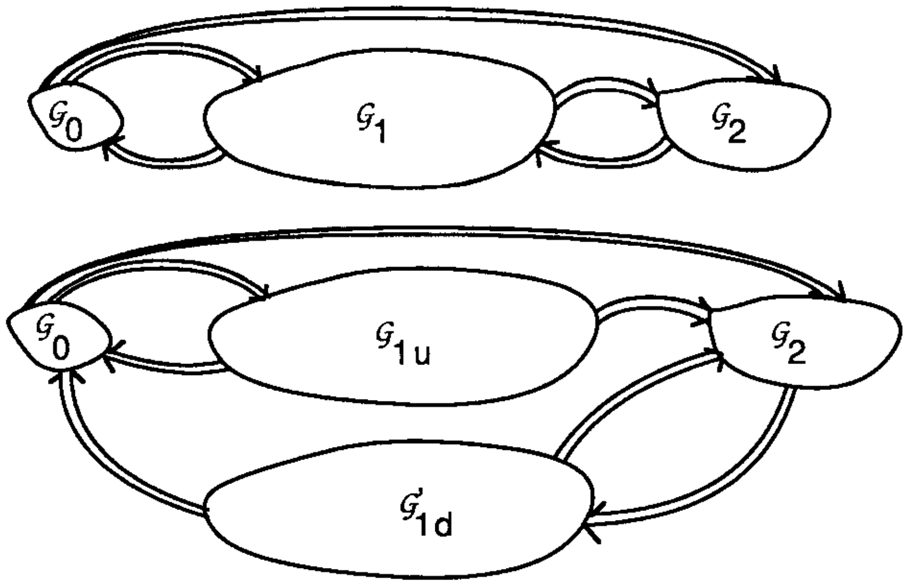

Figure 4: Relationship of G and G'.

To show this for a particular state i, simply define the reward function as:

$$\begin{aligned} r(j) &= 1 \quad j = i \\ &= 0 \quad j \neq i \end{aligned}$$

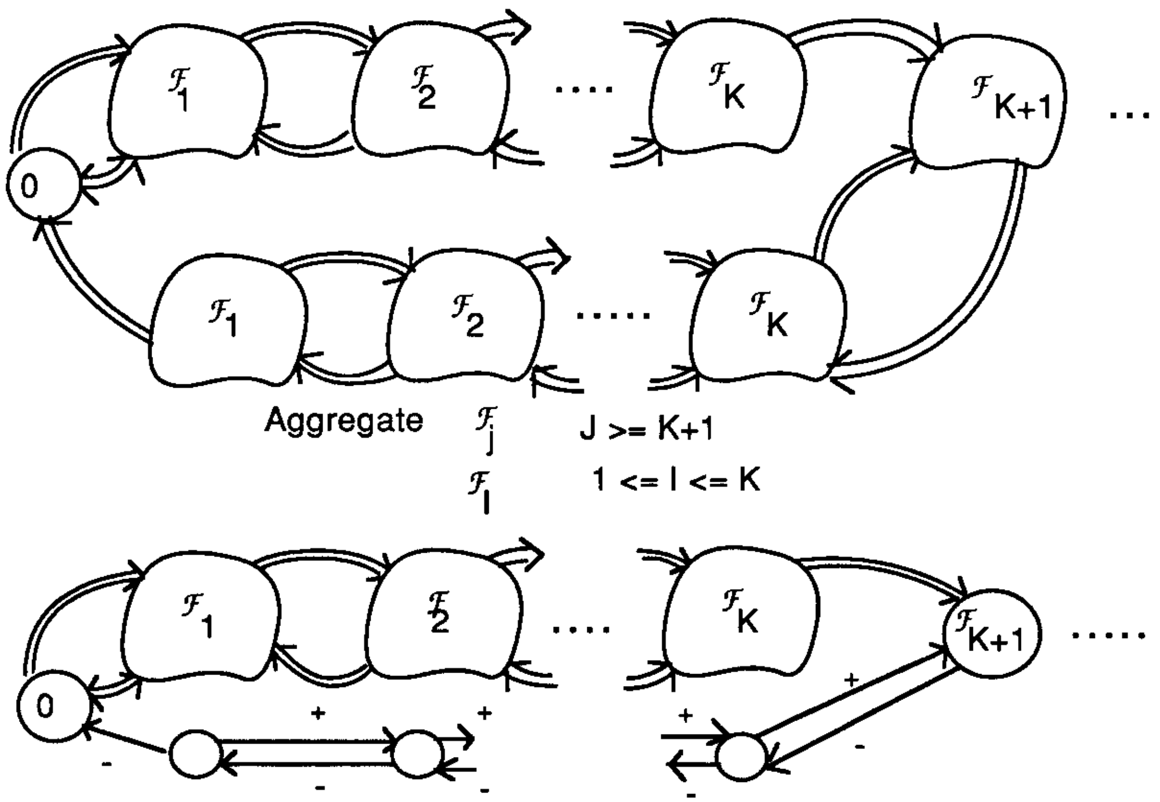

Figure 5: Aggregation of states.

$$
\left[\begin{array}{cc|ccccc}
G_{00} & G_{01} & G_{0K+1'} & & \cdots & & G_{0n} \\
G_{10} & G_{11} & G_{1K+1'} & & \cdots & & G_{1n} \\
\hline
r & 0\ldots0 & \bullet & + & \cdots & + & + \\
0 & & - & \bullet & & + & + \\
\vdots & \vdots & 0 & - & \ddots & + & + \\
 & & & 0 & - & \bullet & + \\
0 & 0\ldots0 & \cdots & & 0 & - & \bullet
\end{array}\right]
$$

Figure 6: Form of transition matrix after aggregation.

3 Preliminaries.

In the "single step" procedure described in the previous section the decision as to the dimension of the matrix G_{11} is made a priori. This implies that once the bounds have been calculated there is no means provided for utilizing the work that has been already done in solving the first portion of the matrix to further tighten the bounds. The extension that we present in this section alleviates these difficulties. It allows an incremental generation of the transition matrix. At each step a new portion of the matrix is generated. Further, at each step the results from the previous steps are used to form a transition rate matrix whose solution allows us to bound the stationary state probabilities for an additional set of states. This allows us to incrementally improve the bounds on availability.

At each step (after the first) there are three sets of states of the original model that we will need to distinguish.

$\mathcal{D}' =$ set of states for which a lower bounds on the stationary state probabilities were obtained in previous steps.
$\mathcal{D} =$ set of states which are the center of attention for this step and for which a lower bound on the stationary state probabilities will be calculated in this step.
$\mathcal{A} =$ the complement of $\mathcal{D}' \bigcup \mathcal{D}$.

Figure 7 illustrates this partitioning of the state space in terms of the transition rate matrix G. In the following we describe a short sequence of transformations to G. Each transformation is such that the stationary state probabilities for the states in $\mathcal{D}$ are always (individually) bounded from below by the corresponding state probabilities from the next model in the sequence. We start by constructing the matrix G_1 from G as illustrated in Figure 8. G_1 corresponds to a model in which the states in $\mathcal{D}$ have been replicated. The replicates (referred to as "clones") will be denoted by $\mathcal{C}$. Note that in G_1 the submatrix Q_{CC} is equal to Q_{DD}. We use the notation Q_{CC} because it enhances readability.

Notation: We use the notation π_{D/G_i} to denote the vector of stationary state probabilities for states in a subset $\mathcal{D}$ when the transition rate matrix is G_i.

In the previous section while reviewing the results from [7] a similar construction was described. From that discussion it is clear that if

$[\pi_{D'/G_1}, \pi_{D/G_1}, \pi_{C/G_1}, \pi_{A/G_1}]$ is the solution of $\pi_1 G_1 = \pi_1$

then

$[\pi_{D'/G_1}, \pi_{D/G_1} + \pi_{C/G_1}, \pi_{A/G_1}]$ is the solution of $\pi G = \pi$.

It follows immediately that $\pi_{D/G_1} \leq \pi_{D/G}$ since $\pi_{C/G_1} \geq 0$.

In the next several transformations we make use of the fact that it is possible to perform exact aggregation of a transition rate matrix. We assume that the reader is familiar with the basic aggregation/disaggregation approximation procedure as described in [1]. Later we show that exact aggregation is not actually required in the computation of the bounds. We merely use exact aggregation in the intermediate steps of the development.

$$\begin{bmatrix} Q_{D'D'} & Q_{D'D} & Q_{D'A} \\ Q_{DD'} & Q_{DD} & Q_{DA} \\ 0 & Q_{AD} & Q_{AA} \end{bmatrix}$$

Figure 7: Initial matrix, G.

$$\begin{bmatrix} Q_{D'D'} & Q_{D'D} & 0 & Q_{D'A} \\ Q_{DD'} & Q_{DD} & 0 & Q_{DA} \\ Q_{DD'} & 0 & Q_{CC} & Q_{DA} \\ 0 & 0 & Q_{AD} & Q_{AA} \end{bmatrix}$$

Figure 8: Introduction of "clone" states. Matrix G_1.

$$\begin{bmatrix} \bullet & R_{d'D} & 0 & R_{d'A} \\ Q_{Dd'} & Q_{DD} & 0 & Q_{DA} \\ Q_{Dd'} & 0 & Q_{CC} & Q_{DA} \\ 0 & 0 & Q_{AD} & Q_{AA} \end{bmatrix}$$

Figure 9: After exact aggregation of the states in $\mathcal{D}'$. Matrix G_2.

G_2 (see Figure 9) is formed from G_1 by exact aggregation of the states in $\mathcal{D}'$. We will refer to the single state which replaces $\mathcal{D}$ as d'. Since exact aggregation is assumed we have that $\pi_{D/G_2} = \pi_{D/G_1}$.

G_3 (see Figure 10) is exactly the same as G_2 except that the transitions from d' to states in $\mathcal{D}$ and $\mathcal{C}$ are modified. In G_3 the submatrices $R'_{d'D}$ and $R'_{d'C}$ are required to have non-negative elements and to be such that:

$$R'_{d'D} + R'_{d'C} = R_{d'D}$$

$$\begin{bmatrix} \bullet & R'_{d'D} & R'_{d'C} & R_{d'A} \\ Q_{Dd'} & Q_{DD} & 0 & Q_{DA} \\ Q_{Dd'} & 0 & Q_{CC} & Q_{DA} \\ 0 & 0 & Q_{AD} & Q_{AA} \end{bmatrix}$$

Figure 10: Modified rates from state d'. Matrix G_3.

A probabilistic interpretation is that the original transitions from d' to states in $\mathcal{D}$ are 'split' so that part remains to the state in D and part goes to the corresponding "clone" state in C.

From the construction of G_3 it is easy to show that if

$[\pi_{d'/G_2}, \pi_{D/G_2}, \pi_{C/G_2}, \pi_{A/G_2}]$ is the solution of $\pi G_2 = \pi$ and

$[\pi_{d'/G_3}, \pi_{D/G_3}, \pi_{C/G_3}, \pi_{A/G_3}]$ is the solution of $\pi G_3 = \pi$ then:

$\pi_{d'/G_3} = \pi_{d'/G_2}$;

$\pi_{D/G_3} + \pi_{C/G_3} = \pi_{D/G_2} + \pi_{C/G_2}$; and

$\pi_{A/G_3} = \pi_{A/G_2}$.

The result that we really want is expressed in the following theorem.

Theorem 1 $\pi_{D/G_3} \leq \pi_{D/G_2}$

Proof: The proof is given in the Appendix. □ .

Now we consider aggregation of the subsets of states in $\mathcal{C}$ and $\mathcal{A}$.

By definition

$\mathcal{D} = \bigcup_{j=L_i}^{H_i} \mathcal{F}_j$,

$\mathcal{C} = \bigcup_{j=L_i}^{H_i} \mathcal{F}'_j$

and

$\mathcal{A} = \bigcup_{j=H_i+1}^{N} \mathcal{F}_j.$

where L_i and H_i are integers associated with the i_{th} step and denote the minimum and maximum number of failed components for $\mathcal{D}$.

We form one aggregate state for each subset $\mathcal{F}_j$ in $\mathcal{C}$ and $\mathcal{A}$. The resulting matrix G_4, is shown in Figure 11. In this figure we have also interchanged the ordering of the state d' and the set of states $\mathcal{D}$. Since we have assumed exact aggregation in forming G_4 it is clear that

$\pi_{D/G_4} = \pi_{D/G_3}.$

$$\left[\begin{array}{cc|ccccc|ccccc}
Q_{D,D} & Q_{D,d'} & 0 & 0 & \cdots & \cdots & 0 & Q_{D,A_1} & Q_{D,A_2} & \cdots & \cdots & Q_{D,A_n} \\
R_{d',D} & \bullet & r_{d',C_1} & r_{d',C_2} & \cdots & \cdots & r_{d',C_J} & r_{d',A_1} & r_{d',A_2} & \cdots & \cdots & r_{d',A_n} \\
\hline
0\ldots0 & r_{C_1,D'} & \bullet & r_{C_1,C_2} & \cdots & \cdots & r_{C_1,C_j} & r_{C_1,A_1} & r_{C_1,A_2} & \cdots & \cdots & r_{C_1,A_n} \\
0\ldots0 & 0 & r_{C_2,C_1} & \bullet & \cdots & \cdots & r_{C_2,C_j} & r_{C_2,A_1} & r_{C_2,A_2} & \cdots & \cdots & r_{C_2,A_n} \\
\vdots & \vdots & 0 & r_{C_3,C_2} & \bullet & \cdots & r_{C_3,C_4} & r_{C_3,A_1} & \cdots & \cdots & \cdots & r_{C_2,A_n} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0\ldots0 & \vdots & 0 & \cdots & \cdots 0 & r_{C_J,C_{J-1}} & \bullet & r_{C_3,A_1} & \cdots & \cdots & \cdots & r_{C_2,A_n} \\
\hline
0\ldots0 & 0 & 0 & 0 & 0 & \cdots & r_{A_1,C_J} & \bullet & r_{A_1,A_2} & \cdots & \cdots & r_{A_1,A_n} \\
\vdots & 0 & 0 & 0 & 0 & \cdots & 0 & r_{A_2,A_1} & \bullet & \cdots & \cdots & r_{A_2,A_n} \\
\vdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & r_{A_3,A_2} & \bullet & \cdots & r_{A_3,A_n} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \ddots & \vdots \\
0\ldots0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & r_{A_n,A_{n-1}} & \bullet
\end{array}\right]$$

Figure 11: Aggregation of states in $\mathcal{C}$ and $\mathcal{A}$. Matrix G_4

At this point we note that π_{D/G_4} provides a lower bound on $\pi_{D/G}$. However, there are some practical problems in applying the procedure exactly as described; values for the transition rates out of aggregate states are required. To actually calculate these would require solving the original model in detail and this is what we are trying to avoid.

Comparing the matrix in Figure 6 with the matrix G_4 in Figure 11 we note that they have the same form. Therefore the result quoted in the last section applies. Specificaly, if the elements shown in Figure 12 as '+' are replaced by upper bounds on those rates and the elements shown as '−' are replaced by lower bounds then the solution for the stationary state probabilities will yield a lower bound for the state probabilities for states in D.

$$
\left[\begin{array}{cc|ccccc|ccccc}
Q_{D,D} & Q_{D,d'} & 0 & 0 & \cdots & \cdots & 0 & Q_{D,A_1} & Q_{D,A_2} & \cdots & \cdots & Q_{D,A_n} \\
R_{d',D} & \bullet & + & + & \cdots & \cdots & + & + & + & \cdots & \cdots & + \\
\hline
0\ldots0 & - & \bullet & + & \cdots & \cdots & + & + & + & \cdots & \cdots & + \\
0\ldots0 & 0 & - & \bullet & \cdots & \cdots & + & + & + & \cdots & \cdots & + \\
\vdots & \vdots & 0 & - & \bullet & \cdots & + & + & \cdots & \cdots & \cdots & + \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0\ldots0 & \vdots & 0 & \cdots & \cdots 0 & - & \bullet & + & \cdots & \cdots & \cdots & + \\
\hline
0\ldots0 & 0 & 0 & 0 & 0 & \cdots & - & \bullet & + & \cdots & \cdots & + \\
\vdots & 0 & 0 & 0 & 0 & \cdots & 0 & - & \bullet & \cdots & \cdots & + \\
\vdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & - & \bullet & \cdots & + \\
\vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \ddots & \vdots \\
0\ldots0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & - & \bullet
\end{array}\right]
$$

Figure 12: Replacement of transition rates with bounds. Matrix G_5.

In summary, $\pi_{D/G_5} \leq \pi_{D/G}$. A remaining issue is the calculation of the upper and lower bounds on the transition rates required for G_5. This issue is addressed and a detailed specification of the multi-step algorithm is given in the following section.

4 Description of the Algorithm

The bounds on the transition rates in G_5 are of three types. The rates denoted by '$+$' in Figure 12 are to be replaced by upper bounds. A simple upper bound is easily seen to be the sum of the failure rates of all components. Similarly, the rates denoted by '$-$' are to be replaced by lower bounds. A simple lower bound that suffices is the minimum repair rate of all components.

The submatrix $Q_{d'D}$ in G_5 must also contain lower bounds on the rates from d' to states in $\mathcal{D}$. If $\pi_{D'/G}$ is the vector of stationary state probabilities for states in $\mathcal{D}$ then $\pi_{D'/G}Q_{D'D}$ is the vector of transition probabilities that describe the exact transitions from d' to states in $\mathcal{D}$. It follows immediately that if

$\pi'_{D'} \leq \pi_{D'/G}$ then

$\pi'_{D'}Q_{D'D} \leq \pi_{D'/G}Q_{D'D}$

This provides the need lower bound provided that the lower bound on $\pi_{D'/G}$ is available.

In the multi-step procedure we describe next, at each step the set of states $\mathcal{D}'$ corresponds exactly to the set of states for which a lower bound has been found on previous steps. Therefore these lower bound stationary state probabilities provide the needed quantities for the following step.

4.1 Multi-step Procedure

The multi-step bounding procedure is as follows:

1. (Step 1) Generate lower bounds on the stationary state probabilities for states $\mathcal{F}_0$ through $\mathcal{F}_{H_1}$ using the results from [7]. This provides lower bounds on the state probabilities of these states as well as an initial set of bounds on steady state availability. If the bounds are tight enough then terminate.

2. (Step $i \geq 2$) Generate the portion of G corresponding to $\mathcal{F}_{L_i}$ through $\mathcal{F}_{H_i}$. Construct the matrix corresponding to G_5 described in the previous section. The submatrices Q_{DD} and $Q_{Dd'}$ are generated from the model definition. Let $\pi'_{D'}$ be the lower bounds on the state probabilities computed from previous steps. Now set the submatrix $Q_{d'D}$ equal to $\pi'_{D'}Q_{D'D}$ and solve for π_{D/G_5} which provides lower bounds on the state probabilities for states in $\mathcal{D}$. Compute upper and lower bounds on steady state availability using Equation 1 from Section 2. If the bounds are tight enough then terminate, else repeat step 2.

5 Example.

In this section, we present a simple example to illustrate the multi-step bounding procedure. We use a model of a fault-tolerant heterogeneous database system as depicted in Figure 13. The components of this system are: a front-end, two databases and three processing subsystems consisting of a switch, a memory and two processors. Components may fail and be repaired according to the rates given in Table 1. If the processor fails, it has a 0.05 probability of contaminating a database. Components are repaired by a single repair facility which gives preemptive priority to components in the order: front-end, databases, switches, processors and lastly the memories. (Ties are broken by random selection.) The database system is considered operational if the front-end is operational, at least one database is operational, and at least one processing subsystem is operational. A processing subsystem is operational if all of the components of that subsystem are operational. Also, this system is in *active breakdown* mode, meaning that components fail even when the system is non-operational.

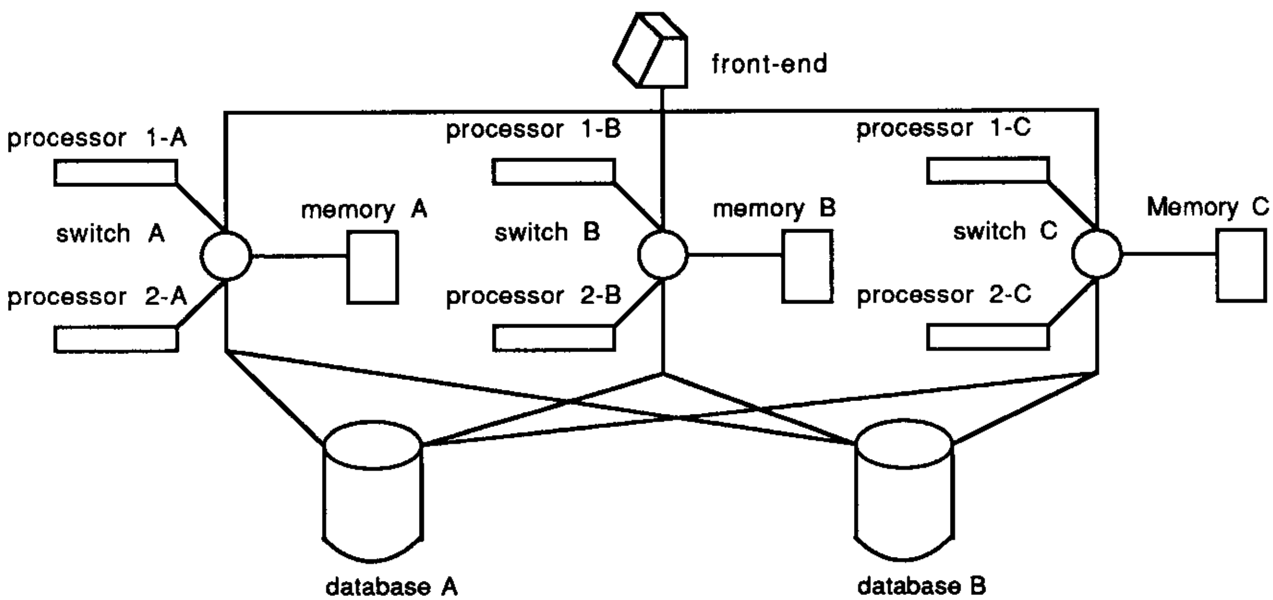

Figure 13: A fault-tolerant distributed database system.

In Table 2, we present the bounds of the steady state availability. We note that for each step, the bounds of the availability are significantly tightened. In step one we set $\mathcal{D}_1 = \{\bigcup_{i=0}^{2} \mathcal{F}_i\}$ In step two $\mathcal{D}_2 = \{\bigcup_{i=3}^{5} \mathcal{F}_i\}$ and in step 3 $\mathcal{D}_3 = \{\bigcup_{i=6}^{7} \mathcal{F}_i\}$.

6 Conclusion

We have presented a method of determining bounds on the steady state availability from the Markov model of a repairable computer system. This work is an extension of [7] which extends the one-step procedure into a multi-step procedure for evaluating the bounds of availability. At each step a tighter bound on availability can be obtained by generating more of the transition matrix of the original model.

The method not only provides the flexibility to tradeoff computational resources and error but it allows the tradeoff to be made dynamically; i.e. if at some step the bounds are not tight enough then more of the matrix can be generated and the bounds tightened without loss of the previous calculation.

Component	Mean Failure Rates	Mean Repair Rates
Front-end	1/8000	2.0
Processor 1-A	1/300	1.3
Processor 2-A	1/100	1.3
Switch A	1/550	2.0
Memory A	1/500	1.0
Processor 1-B	1/400	1.1
Processor 2-B	1/300	1.4
Switch B	1/600	2.1
Memory B	1/550	1.5
Processor 1-C	1/350	1.2
Processor 2-C	1/275	1.4
Switch C	1/650	2.5
Memory C	1/600	2.0
Database A	1/4000	2.0
Database B	1/5000	2.1

Table 1: Failure and repair rates(per hour).

Step Number	Lower Bound	Upper Bound	No of States in D
1	0.999878304	0.999939472	120
2	0.999930212	0.999938779	4823
3	0.999937012	0.999937590	11440

Table 2: Upper and lower bounds on steady state availability of the database system.

The procedure described relies on several properties of reliability models. First, to obtain reasonably tight bounds and yet generate only a small portion of the transition rate matrix requires that the stationary state probability distribution be highly skewed. Second, we utilized the assumption that the aggregated Markov process has the "skip-free to the left" property, i.e. is upper Hessenberg. In the availability modeling domain this corresponds to an assumption that there is zero probability of more than two components being repaired at the same instant. In addition, we used the nature of the application domain in choosing the definition of the aggregate states and in placing upper and lower bounds on the transitions between aggregates.

There is ongoing work investigating certain properties of the algorithm presented and compare with alternatives. For example, rather than exploring more of the state space it may sometimes be advantageous to refine the estimates of state probabilities already calculated, e.g. using the scheme in [2]. Also, with the procedure presented in this paper, there is some "unrecoverable" error at each step; i.e. that cannot be eliminated by further steps. (This is the difference between the lower bounds and the actual stationary state probabilities.) This error can be eliminated by the iterative approach. The tradeoff between the "forward only" and iterative approach remains to be investigated. It is also interesting to consider whether it is possible to provide efficient guidelines (either a priori or dynamic) to choose between the two.

It appears that there are a number of applications for which the assumption of a highly skewed stationary state probability distribution is reasonable. See for example [3,6] for example applications in probabilistic communication protocol verification. It remains to be investigated whether the other properties (e.g. "skip-free to the left") will be applicable in other domains or whether the method can be generalized to relax these conditions.

Appendix

A Proof for Theorem 1

Theorem 1. $\pi_{D/G_3} \leq \pi_{D/G_2}$

Proof : Let G be the transition rate matrix for a finite, irreducible continuous time Markov chain $\mathcal{G}$. Then $\mathcal{G}$ is uniformizable which means it can be transformed to a discrete time Markov chain with transition probability matrix P which has the same stationary probability vector π [9]. This transformation is achieved by:

$$P = I + \lambda^{-1} G$$

where λ is $\geq$ the largest absolute diagonal element of G. With this in mind, we can transform G_2 and G_3 into P_2 and P_3 with $\lambda = max(\lambda_2, \lambda_3)$ where $\lambda_2\,(\lambda_3)$ is the largest of the absolute values of the diagonal elements of $G_2\,(G_3)$.

Let us define the following notation:

$p_{i,j}$	the transition probability from state i to state j in P_2.
$p'_{i,j}$	the transition probability from state i to state j in P_3.
$d_{i,j}$	$= \; p_{i,j} - p'_{i,j}$.
$r(i)$	reward (either 0 or 1) for state i.
$T[r(i)]$	one-step expected reward of P_2 given the present state is i, i.e. $T[r(i)] \; = \; \sum_j p_{ij}\, r(j)$
$T_m[r(i)]$	one-step expected reward of P_3 given the present state is i.
$R^k(i)$	k-step ($k-1$ transitions plus initial position) accumulative reward for P_2 given the initial state is i,i.e. $R^k(i) \; = \; \sum_{l=0}^{k-1} T^l[r(i)] \qquad for\ k = 1, 2, 3, \ldots$
$R^k_m(i)$	k-step ($k-1$ transitions plus initial position) accumulative reward for P_3 given the initial state is i.
$1\{c\}$	an indicator function equal to 1 if the condition c is true, else 0.

with $T_m[r(i)]$ and $R^k_m(i)$ defined accordingly.

A special case of the results in [13] is that the expected reward for $\mathcal{G}_2$ is greater than or equal to the expected reward for $\mathcal{G}_3$ iff :

$$(T - T_m)R^k(i) \geq 0 \qquad \forall i \text{ and } k$$

Let ϕ be a 1–1 mapping from $\mathcal{D}$ to $\mathcal{C}$ which maps each state $s_D \in \mathcal{D}$ to the corresponding state $s_C \in \mathcal{C}$.

Let f be any nonnegative function applied to state i of the Markov chain. Since the only difference between G_2 and G_3 is in the rates out of d', then for any nonnegative function f :

$$\begin{aligned}(T - T_m)f(i) &= 1\{i = d'\}\Bigg\{\sum_{s_D \in D} p_{d',s_D} f(s_D) + p_{d',d'} f(d') + \sum_{a_i \in A} p_{d',a_i} f(a_i) \\ &\quad - \sum_{s_D \in D} p'_{d',s_D} f(s_D) - p'_{d',d'} f(d') - \sum_{s_C \in C} p'_{d',s_C} f(s_C) - \sum_{a_i \in A} p'_{d',a_i} f(a_i)\Bigg\}\end{aligned}$$

Since

$$\begin{aligned} p_{d',d'} &= p'_{d',d'} \\ p_{d',a_i} &= p'_{d',a_i} \quad \forall\, a_i \in A \\ p_{d',s_D} &\geq p'_{d',s_D} \quad \forall\, s_D \in D \end{aligned}$$

it follows that :

$$(T - T_m)f(i) = 1\{i = d'\}\left\{\sum_{s_D \in D} d_{d',s_D} f(s_D) + \sum_{s_C \in C} d_{d',s_C} f(s_C)\right\}$$

Since by construction of G_3 from G_2,

$$d_{d',s_D} = -\, d_{d',\phi(s_D)} \qquad \forall s_D \in D$$

we have :

$$(T - T_m)f(i) = 1\{i = d'\}\left\{\sum_{s_D \in D} d_{d',s_D}[f(s_D) - f(\phi(s_D))]\right\}$$

A sufficient condition for the above expression to be ≥ 0 is :

$$[f(s_D) - f(s_C)] \;\; \geq \;\; 0 \qquad\qquad where \;\; s_C = \phi(s_D)$$

Letting the function f be $R^k(i)$, then the condition:

$$(T - T_m)R^k(i) \geq 0 \qquad\qquad \forall i, \; k$$

is satisfied if :

$$R^k(s_D) - R^k(s_C) \geq 0 \qquad\qquad \forall s_D \in \mathcal{D} \;\; and \;\; \forall k$$

where $s_C = \phi(s_D)$.

The above sufficient conditions can be easily proved by induction.

For $k = 0$, since $R^0(*) = 0$, the inequalities hold, then for each $s_D \in \mathcal{D}$ and $s_C \in \phi(s_D)$.

Assume $R^k(s_D) - R^k(s_C) \geq 0$ for $k \leq n$. In $k = n+1$ we have:

$$\begin{aligned} R^{n+1}(s_D) - R^{n+1}(s_C) \;\; = \;\; & \Big\{ \sum_{s_D \in D} p_{s_D,s_D} R^n(s_D) + p_{s_D,d'} R^n(d') + \sum_{a_i \in A} p_{s_D,a_i} R^n(a_i) \\ & - \sum_{s_C \in C} p_{s_C,s_C} R^n(s_C) - p_{s_C,d'} R^n(d') - \sum_{a_i \in A} p_{s_C,a_i} R^n(a_i) \Big\} \end{aligned}$$

Since :

$$\begin{aligned} p_{s_D,d'} &= p_{s_C,d'} \\ p_{s_D,a_i} &= p_{s_C,a_i} \qquad\qquad \forall s_C = \phi(s_D) \;\; and \;\; \forall a_i \in A \\ p_{s_D,s_D} &= p_{s_C,s_C} \qquad\qquad \forall s_C = \phi(s_D) \end{aligned}$$

we have the following :

$$\begin{aligned} R^{n+1}(s_D) - R^{n+1}(s_C) \;\; &= \;\; \Big\{ \sum_{i_D \in D} p_{s_D,i_D} [R^n(s_D) - R^n(\phi(i_D))] \Big\} \\ &\geq \;\; 0 \qquad\qquad \Box \end{aligned}$$

References

[1] P. J. Courtois. *Decomposability — queueing and computer system approximation. Academic Press, New York, 1977.*

[2] P. J. Courtois and P. Semal. *Bounds for the Positive Eigenvectors fo Nonnegative Matrices and for Their Approximations JACM, Oct. 1984, pp. 804-825.*

[3] D.D. Dimitrijevic and M. Chen. *An Integrated Algorithm for Probabilistic Protocol Verification and Evaluation, IBM Tech. Report RC 13901, 1988.*

[4] A. Goyal, S.S. Lavenberg and K.S. Trivedi. *Probabilistic Modeling of Computer System Availability, Ann. of Oper. Res., vol. 8, pp. 285-306, 1986.*

[5] J.G. Kemeny, J.L. Snell. *Finite Markov Chains. Van Nostrand Company, 1960*

[6] N.F. Maxemchuk and K. Sabnani. *Probabilistic Verification of Communication Protocols, in Protocol Specification, Testing and Verification, VII, ed. H. Rubin and C.H. West, Elsevier, 1987, pp.307-320*

[7] R. R. Muntz,E. De Souza Silva, A. Goyal. *Bounding Availability of Repairable Computer Systems, SIGMETRICS 1989, pp. 29-38, also to appear in a special issue of IEEE-TC on performance evaluation, Dec. 1989.*

[8] M. F. Neuts. *Matrix-geometric solutions in Stochastic Models — an algorithmic approach. John Hopkins University Press, Baltimore, MD, 1981.*

[9] S.M. Ross. *Stochastic Processes. Wiley Series in Probability and Mathematical Statistics, 1983.*

[10] William J. Stewart, Ambuj Goyal. *Matrix Methods in Large Dependability Models. IBM Research Report, 11485, Nov 4, 1985.*

[11] Y. Takahashi. *Some Problems for Applications of Markov Chains, Ph.D. Thesis, Tokyo Institute of Technology. March 1972.*

[12] Y. Takahashi. *A lumping method for numerical calculations of stationary distributions of Markov chains, Research Reports on Information Sciences, Tokyo Institute of Technology, No.B-18, June 1975.*

[13] Nico M. van Dijk. *Simple bounds for Queueing Systems with Breakdowns. Performance Evaluation 8, 1988. page 117-128.*