Proc. 16th IAESTED Int. Conf. on Identification, Modeling and Simulation, Paris, June 1987.

ANALYSIS OF SPARSE NUMERICAL METHODS FOR DEPENDABILITY EVALUATION

Juan A. Carrasco
Departament d'Enginyeria Electrònica, ETSEIB
Universitat Politècnica de Catalunya
Diagonal 647, 08028-Barcelona, SPAIN

ABSTRACT

Homogeneous, continuous-time Markov chains are an usual mathematical tool for dependability evaluation of computer systems. Complex systems rise the large state space problem, thus making mandatory the use of efficient numerical methods. This paper analyzes state of the art sparse methods in the context of the development of an enhanced version of an existing tool. We first consider point SOR for the evaluation of the stationary probability and mean time before absortion vectors. By dynamic adjustment of the relaxation parameter a very high convergence rate is attained in many cases. The convergence is however rather poor when the Markov chain contains highly recurrent subsets. In order to improve the method we propose the integration of a novel, exact, one-step aggregation technique with block SOR. For the evaluation of the transient regime, two methods are compared: randomization and implicit integration. Scenarios in which one of them is clearly preferrable are described.

Key Words: Dependability Evaluation, Markov Chains, Sparse Numerical Methods, Aggregation Techniques, Point SOR.

I. INTRODUCTION

Dependability /1/ is widely considered as an important attribute of Computer Systems. Dependability metrics such as steady-state availability, interval availability, mean time to failure and reliability are relevant criteria for the design and validation of Computer Systems. The evaluation of these measures for Fault-Tolerant Computing Systems is a difficult problem due to the conjunction of three factors: statistical dependencies, rare events and system complexity. Statistical dependencies force the establishment of the system dynamics at the state level. Since the events to be evaluated are rare, particularly for highly dependable systems, simulation is not affordable and the models have to be solved analytically. Homogeneous, continuous-time Markov chains (referred for conciseness simply as Markov chains hereafter) are generally used because they combine good modeling capabilities with a relatively simple analysis. Finally, the complexity of the systems leads to large state spaces. We are then faced with the evaluation of large Markov chains.

There are two basic approaches against the large state space problem: avoidance and tolerance. We assume that, due to dependencies, the model is not exactly decomposable. The avoidance approach is aimed at reducing the size of the model and includes techniques such as state lumping /2/, graph pruning /3/ and approximate decomposition /4/. The tolerance approach consists in the use of sparse techniques to allow the processing of large Markov chains. It is clear that the maximum benefit is obtained from the combined use of both approaches. In this paper we deal only with the tolerance approach. More precisely, we focus our attention in sparse methods for the basic numerical problems to which the evaluation of many dependability measures can be reduced. It should be noted that this analysis is made under the perspective of the enhancement of a general-purpose modeling tool /5/. Therefore, we are interested in methods which are generally applicable and robust.

We first consider point SOR for the evaluation of the stationary probability and the mean time before absortion vectors. Next, we describe an exact, one-step aggregation technique and its use to improve the convergence of the iterative method in the cases in which point SOR is slow. Finally, we compare two state of the art methods for the evaluation of the transient regime of Markov chains: randomization and implicit integration.

II. ANALYSIS OF POINT SOR

II.1. Numerical problems

Let A be an irreducible transition rate matrix and $\bar{p}$ the stationary probability vector of a Markov chain. The elements of A are $a_{ij} = \lambda_{ij}$ for $i \neq j$, and $a_{ii} = -\lambda_i$, where λ_{ij} are transition rates from i to j and λ_i is the output rate from the state i. Since A is irreducible, the chain is ergodic and $\bar{p}$ exists and does not depend on the initial probability distribution. $\bar{p}$ can be computed by solving the homogeneous system:

$$\bar{p}^T A = \bar{0}^T \qquad (1)$$

under the restriction:

$$\bar{p}^T \bar{e} = 1 \ , \qquad (2)$$

where $\bar{e}$ is a summation column vector of all ones.

Consider now a subset S of states. Let the components of $\bar{v}_S$ be the average times spent in the states in S before an exit from the subset. Assume that A_{SS} is not singular. $\bar{v}_S$ can be computed by making the states outside S absorbing and integrating the probability vector restricted to S, $\bar{p}_S(t)$. It results:

$$\bar{v}_S^T A_{SS} = - \bar{p}_S^T(0) \ . \qquad (3)$$

Many dependability measures can be computed from the vectors $\bar{p}$ and $\bar{v}_S$. For example, the steady-state availability is obtained by adding up the elements of $\bar{p}$ associated to non-failed states, and the MTTF by taking as S the set of non-failed states and adding the elements of $\bar{v}_S$.

The systems (1) and (3) can be solved by iterative or direct methods. Direct methods have the basic problem of creating fill-in in the irreducible blocks of the matrix, thus limiting the size of the matrices which can be dealt with. Therefore, iterative methods should be used to solve very large models, as far as their convergence is reasonable. Advantage can be taken of a non trivial upper block triangular structure of A_{SS}, which appears, for example, when modeling non repairable systems (in this case the transition graph is usually acyclic) or when integrating hardware fault/repair and software fault/correction /6/. An optimal upper block triangular structure can be obtained by topologically sorting the transition graph. The algorithm proposed in /7/ has time and memory complexities O(n,m), being n the number of states and m the number of transitions. Using such an structure, the solution of (3) can be reduced to the solutions of systems on the diagonal irreducible blocks. Thus, in the following we will assume A_{SS} irreducible.

A comparison of iterative methods has been performed in /8/. Point SOR with dynamic prediction of the optimum relaxation parameter appears to be the most promising one. We refer to /9/ for a detailed exposition of the theory of iterative methods. It is convenient to scale the matrix by dividing its rows by $-a_{ii} > 0$ before applying point SOR, since this saves operations in the iterative step. In the rest of this part and in the following one it will be assumed that the matrices A and A_{SS} have been scaled and the notation Q will be used to refer to any of them.

II.2. Test of point SOR

We now present some tests using the same example as in /8/. There are however two differences. First, we use relative errors to define the convergence criterium. The relative errors in the components of the solution are estimated using an error atenuation factor, inferred during the iteration, and the differences between the most recent iterates. The convergence is considered achieved when all these errors are smaller than a specified tolerance TOL. Second, we also include results for the problem (3). As in /8/, for (1), we do not normalize the solution between iterations.

The example consists of a system with four components, two of which have failure and repair rates λ_1 and μ_1, respectively, and the other two have failure and repair rates λ_2 and μ_2, respectively. The resulting Markov chain has nine states and is depicted in Fig. 1. The model is very simple but rich enough to test the method. We present number of iterations and final inferred values for the relaxation parameter w in the tables 1 and 2. For the computation of the mean time before absortion vector, we take $S = \{1,2,3,5\}$, as it would be done to compute the MTFF assuming that the operation requires at least one component of each type. For the results given in Table 2, 1 is taken as initial state.

The results for $\bar{p}$ are analyzed first. Typically, failure rates << repair rates. As far as this holds, a noticeable fast convergence is attained. The poor performance in the cases 7, 8 and 9 can be explained by the creation of the highly recurrent subsets $\{1,3,6\}$, $\{2,5,8\}$ and $\{4,7,9\}$ due to the fact that $\lambda_2, \mu_2 >> \lambda_1, \mu_1$. These cases are pathological in the context of the model used as a test, but reflect situations which can arise in more sophisticated models incorporating performance behavior or more accurate fault modeling, e.g., intermittent faults.

The results for the evaluation of the mean time before absortion vector show that point SOR will be slow when repair rates >> failure rates and several failures

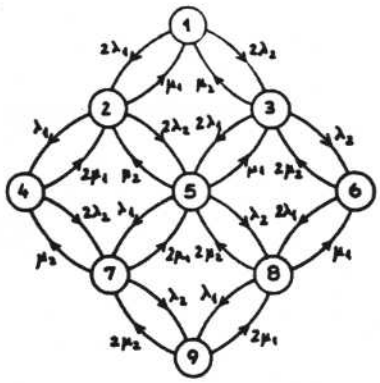

Fig. 1. Markov chain used in the tests

are needed for the system to fail. A poor convergence would also occur with coverage factors very close to 1. These are typical situations for highly reliable fault-tolerant systems. The explanation is similar to that previously given. In this case, the non-failed states 1, 2 and 3 constitute a highly recurrent subset when $\mu_i >> \lambda_i$.

It has been pointed out /10/ that state ordering has an important impact on the convergence rate. The method used in /10/ differs from the one tested in that the iterations are carried out on the system:

$$\bar{p}^T \tilde{A} = [1 \; \bar{0}^T \quad ,$$

where $\tilde{A}$ is obtained from A by replacing its first column by all ones. Thus, the probability normalization condition (2) is incorporated into the system. We have experimented this alternative formulation and found that it gives worse convergence. State ordering seems to have little impact on the convergence of the method tested.

Table 1. Test of point SOR for the evaluation of $\bar{p}$. State ordering: 1,2,3,4,5,6,7,8,9. TOL = 10**(-6).

case	λ_1	λ_2	μ_1	μ_2	nit	w
1	1	1	10	10	9	1.0261
2	1	1	100	100	6	1.0033
3	1	1	10^3	10^3	6	1.0003
4	1	1	10^4	10^4	6	1.0000
5	1	100	10^3	10^1	11	1.0296
6	1	10^3	10^4	10^4	12	1.0296
7	1	10^3	10	10^4	130	1.9062
8	1	10^4	1	10^4	531	1.9758
9	1	10^5	1	10^5	1501	1.9938

Table 2. Test of point SOR for the evaluation of $\bar{m}_S$ with $S = \{1,2,3,5\}$. State ordering: 1,2,3,5. TOL=10^{-6}.

case	λ_1	λ_2	μ_1	μ_2	nit	w
1	1	1	10	10	33	1.5795
2	1	1	100	100	87	1.8258
3	1	1	10^3	10^3	282	1.9477
4	1	1	10^4	10^4	634	1.9833
5	1	100	10^3	10^3	32	1.5414
6	1	10^3	10^4	10^4	32	1.5391

III. IMPROVING POINT SOR

III.1. Background

Highly recurrent subsets (h.r.s.'s) in Markov chains are a known problem and several techniques are at hand such as block SOR and iterative aggregation/disaggregation /11/. In all these methods systems of the form $\bar{x}_i^T Q_{ii} = -\bar{b}_i^T$ have to be solved. It is usually assumed that Q_{ii} are small enough to make feasible the use of direct methods to solve these systems. Since the submatrices are associated to h.r.s.'s, point SOR will be slow. The technique described in the next section is aimed at improving the method for this class of matrices. This will allow to define an efficient and general numerical device using only iterative methods.

III.2. The technique

We define a minimal h.r.s. as any h.r.s. which does not contain any h.r.s. different from itself. It should be pointed out that isolated states are not considered to be h.r.s.'s. Let B a minimal h.r.s. We are interested in solving:

$$\bar{x}_B^T Q_{BB} = -\bar{b}_B^T \quad . \tag{4}$$

Let a be any state in B. Since B is minimal, it follows that $R = B - \{a\}$ is not a h.r.s nor does it contain any h.r.s. Assume that a is numbered as 1. We can rewrite (4) in the form:

$$[x_1 \;\; \bar{x}_R^T] \begin{bmatrix} -1 & \bar{q}_{1R}^T \\ \bar{q}_{R1} & Q_{RR} \end{bmatrix} = -[b_1 \;\; \bar{b}_R^T] \quad .$$

In order to give intuitive insight to the technique, we will use the interpretation of $\bar{b}_B$ as initial probability vector (compare (3) and (4)). However, the validity of the technique does not depend on any property of $\bar{b}_B$ associated to this interpretation. A proof of the exactitude of the aggregation technique can be found in /12/. Under this interpretation, $\bar{x}_B$ can be viewed as the mean time before absortion vector for the subset B and the "initial probability distribution" $\bar{b}_B$. In the technique R is grouped into two states: u and r; u accounts for the initial behavior in R before an exit from the subset and u represents the stay in R after a jump from a. Next, we build the aggregated discret Markov shown in Fig. 2,a. The jump probability γ is given by:

$$\gamma = \bar{q}_{1R}^T \bar{e} \quad .$$

In order to compute α and β we solve the systems:

$$\bar{v}^T Q_{RR} = -\frac{\bar{b}_R^T}{\bar{b}_R^T \bar{e}} \tag{5}$$

$$\bar{\rho}^T Q_{RR} = -\frac{\bar{q}_{1R}^T}{\bar{q}_{1R}^T \bar{e}} \tag{6}$$

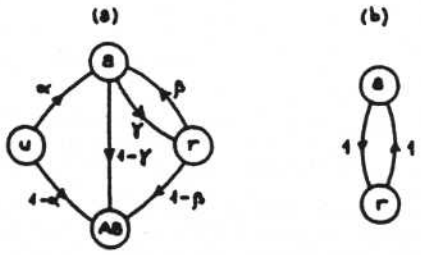

Fig. 2. Aggregated discrete Markov chains used in the proposed aggregation technique.

The elements of $\bar{v}$ can be interpreted as the average number of visits to the states in R, given that the chain is initially in this subset with initial probability distribution given by $\bar{b}_R$ after normalization. The elements of $\bar{\rho}$ are the average number of visits to the states in R since a jump from a till the exit of the subset. The jump probabilities α and β can then be computed by

$$\alpha = \bar{v}^T \bar{q}_{R1}$$

$$c_i = 1 - \sum_{j \in B,\; j \neq i} q_{ij} \qquad \text{for } i \in R$$

$$\beta = 1 - \bar{\rho}^T \bar{c}_R \tag{7}$$

The elements of $\bar{c}$ are the jump probabilities outside B. (7) is numerically better than the more obvious $\beta = \bar{\rho}^T \bar{q}_{R1}$ (consider (8) and (9) and that, since B is a h.r.s., $\beta\gamma \ll 1$). It is easy to evaluate the average number of visits to a, u and r before absortion for the aggregated discrete Markov chain. It results:

$$x_1^a = \frac{b_r \alpha + b_1}{1 - \beta\gamma} \tag{8}$$

$$x_u^a = b_r = \bar{b}_R^T \bar{e}$$

$$x_r^a = \frac{b_r \alpha + b_1}{1 - \beta\gamma}\,\gamma \tag{9}$$

In the disaggregation step, the vector $\bar{x}_B$ is computed as follows:

$$x_1 = x_1^a$$

$$x_i = x_u^a v_i + x_r^a \rho_i \qquad \text{for } i \in R$$

We have then reduced the problem of solving (4) to solving the systems (5), (6), which correspond to Markov chains without h.r.s.'s.

A similar technique can be applied to the solution of (1). Again, assume that the state to be cut off is numbered 1 and let R denote the subset of the remaining states. The aggregated discrete Markov chain is shown in Fig. 2,b. The average number of visits to the states in R before leaving the subset are computed by (6). In this case, the solution of the discrete Markov chain is trivial and a solution $\bar{x}$ is:

$$x_1 = 1$$

$$x_i = \rho_i \quad \text{for } i \in R$$

We have then reduced the problem (1) to the solution of a system (6). The advantage of the latter is that, being A irreducible, Q_{RR} is either irreducible diagonally dominant or admits a block upper triangular form with diagonal blocks satisfying the condition. Then Gauss-Seidel is guaranteed to converge /9/ and a carefull implementation of point SOR will be safe.

III.3. Application

We now show how the aggregation technique described in the previous section can be used in conjunction with block SOR without aggregation to deal with the cases in which the matrix has h.r.s.'s. We have also tested block SOR with aggregation and found that the reduction on the number of iterations is not worth the extra cost for iteration. According to the proposal for the evaluation of $\bar{p}$ recently given, we need only consider the problem (4). Assume that Q_{RR} is arranged in block form so that the diagonal blocks correspond to the h.r.s.'s. Block SOR is described by:

$$\bar{x}_i^{(k+1)^T} = (1-w)\,\bar{x}_i^{(k)^T} + wQ_{ii}^{-1}\left(\sum_{j<i} \bar{x}_j^{(k+1)^T} Q_{ji} + \sum_{j>i} \bar{x}_j^{(k)^T} Q_{ji} + \bar{b}_i^T\right)$$

The iterative step involves the solution of systems with matrices Q_{ii}. If the h.r.s.'s are minimal it is enough to apply the technique to these systems and use point SOR for the resulting (5) and (6). Otherwise, the block SOR iteration can be recursively applied to solve (5) and (6). The right-hand side of the systems (5) will be different for each iteration on the upper level. However, the solutions to these systems obtained in the previous iteration step can be used as good starting vectors, thus reducing the total number of iterations on the matrices Q_{ii}.

It remains to discuss how to detect the h.r.s.'s. After scaling the matrix, we visit all the states of the associated Markov chain and delete the transitions with rates smaller than a given threshold (say 10^{-2}). Note that, after scaling, the transition rates are the conditional jump probabilities between states. The strongly connected components of the resulting are h.r.s.'s. We proceed applying recursively the procedure to the subgraphs associated to proper strongly connected components till either the components are themselves strongly connected or are reduced to single states. At the end we have obtained a hierarchy of h.r.s.'s with minimal h.r.s.'s at the bottom level.

III.4. Tests revisited

In order to show the efficiency of the technique we revisit the tests performed in II.2 for the direct aplication of point SOR. The evaluation of $\bar{p}$ is considered first (see Table 3). In the cases 1 through 6, the Markov chain obtained after cutting off 1 does not have any h.r.s. The results are similar to those given by Table 1, showing that there is almost no price for ensuring the safeness of the method (see end of III.2). For the cases 7, 8 and 9 two h.r.s.'s result after cutting off 1: $B_1 = \{2,5,8\}$ and $B_2 = \{4,7,9\}$. The table shows for these cases the number of upper level iterations (nit) as well as the total iterations on the matrices associated to the h.r.s.'s. For the evaluation of $\bar{\pi}_S$, only the significant

Table 3. Test of improved method for the evaluation of $\bar{p}$. TOL = 10**(-6).

case	λ_1	λ_2	μ_1	μ_2	nit	nit(B_1)	nit(B_2)
1	1	1	10	10	10	-	-
2	1	1	100	100	5	-	-
3	1	1	10^3	10^3	3	-	-
4	1	1	10^4	10^4	3	-	-
5	1	100	10^3	10^3	14	-	-
6	1	100	10^4	10^4	7	-	-
7	1	10^3	10	10^4	6	14	14
8	1	10^4	1	10^4	24	46	46
9	1	10^5	1	10^5	24	46	46

Table 4. Test of improved method for the evaluation of $\bar{\pi}_S$, with S = {1,2,3,5}. TOL = 10^{-6}.

case	λ_1	λ_2	μ_1	μ_2	nit	w
1	1	1	10	10	7	1.0377
2	1	1	100	100	4	1.0076
3	1	1	10^3	10^3	3	1.0005
4	1	1	10^4	10^4	3	1.0000

cases 1, 2, 3 and 4 have been reconsidered. Cutting off state 1 leaves the Markov chain without proper h.r.s.'s and point SOR is enough in this case. As Table 4 shows, the number of iterations has been drastically reduced.

IV. TRANSIENT REGIME

Two promising sparse numerical methods to solve the transient regime are randomization /13/ and implicit integration /14/. We are interested in integrating:

$$\frac{d\bar{p}^T}{dt} = \bar{p}^T(t)\,Q\,, \quad \bar{p}(0) = \bar{p}_0\,,$$

where Q is a matrix A or A_{SS} (without scaling). Let $\Lambda = \max\{-q_{ii}\}$. i.e., the maximum output rate. Consider the discrete Markov chain $\bar{p}^*(k)$, $k \geq 0$, with transition matrix

$$Q^* = \frac{1}{\Lambda} Q + I$$

and initial probability vector $\bar{p}^*(0) = \bar{p}_0$. The randomization formula is:

$$\bar{p}(t) = \sum_{k=0}^{\infty} \bar{p}^*(k) \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} \qquad (10)$$

One of the main advantages of the method is that $\bar{p}^*(k)$ can be determined very easily and taking advantage of the sparseness of Q. In the practice, one has to approximate (10) by limiting k to a truncation value K. A good criterium is to allow a residual error added over all the components of the solution no greater than ϵ. The smaller value of K satisfying this condition is given by:

$$K(\epsilon,\Lambda t) = \min\left\{m \;/\; \sum_{k=0}^{m} \frac{(\Lambda t)^k}{k!} e^{-\Lambda t} > 1-\epsilon\right\}$$

Table 5. Truncation term $K(\Lambda t,\varepsilon)$ for randomization formula.

	ε			
Λt	10^{-3}	10^{-5}	10^{-7}	10^{-9}
0.1	2	3	4	6
1	5	8	10	11
10	21	26	30	34
100	132	145	156	166
10^3	1099	1138	1169	1195

Table 5 shows the influence of ε and Λt. Note that the residual error is the probability that more than K arrivals occur in a Poisson process with parameter Λt. For $\Lambda t >> 1$, the number of arrivals approaches asymptotically a normal distribution with mean Λt and standard deviation $(\Lambda t)^{1/2}$. Then, for $\Lambda t >> 1$, $K(\varepsilon,\Lambda t)$ is approximately equal to Λt. This allows to discuss the efficiency of randomization. Assume that the system is non-repairable and that the Markov chain only models permanent faults. In this case the t of interest will be of the order of magnitude of Λ^{-1} and randomization will be very effective. The following scenario is a repairable system and a model accounting only for permanent faults and repair. In this case, Λ will be of the order of magnitude of the fastest repair rates. We will consider two evaluation cases: availability and reliability. In the first, the steady-state regime is typically reached at a rate close to the slowest repair rates and randomization will be again typically very efficient. In the second case, with the exception of short time mission systems, the t of interest will be large compared to Λ^{-1}, being $\Lambda t = 10^4$ a typical value. In this case, randomization will be rather slow. Approximated aggregation techniques /15/ are available to deal with these cases, but the error may be coarse, for instance, when the system has a large number of components so that states with faults have not negligeable probabilities.

An alternative method is implicit numerical integration. In the first version of our modeling package /5/ we implemented a sparse version of the Gear's method. Our experience has shown us that usually between 200 and 500 integration steps are enough to solve the full transient regime, i.e., till the extinction of the slowest transient component, with relative errors of about 10^{-7}. The matrices appearing in the integration formula of the method share the properties of A_{SS} in (3). Therefore, the use of the improved SOR method proposed in III.2, 3 seems to be a valuable alternative when randomization is slow. We are currently carrying out extensive tests on an improved version of Gear's method.

V. CONCLUSIONS

Point SOR alone is effective for the evaluation of the stationary probability vector in most cases. However, it has been shown that its convergence is poor for typical instances of the evaluation of the mean time before absorption vector. The method can also find problems for the evaluation of the stationary probability vector in sophisticated models. The combination of the proposed one-step, exact aggregation technique with block SOR has improved the method so as to make it good enough for a general-purpose tool. In relation to the methods for the transient regime which have been analyzed, no general conclusion can be outlined. Randomization is undoubtly the method of choice in some scenarios, but it may be slow in some cases. Implicit integration gains attractiveness because the improved iterative method can be used. It seems that the method can be a good alternative to randomization in the scenarios in which the latter is slow. For general-purpose dependability evaluation tools, the integration of both methods with a heuristic selection policy is suggested.

Acknowledgements

The author wishes to acknowledge the encouragement received from J. Figueras in the course of this work. Dr. Goyal is greatly appreciated for providing me with detailed information on a safe implementation of point SOR. This work was supported in part by the SMART consortium (ESPRIT project, Software Technology, area 2.1) and the "Comisión Asesora para la Investigación Científica y Técnica" (CAYCIT).

REFERENCES

/1/ J.C. Laprie, "Dependable Computing and Fault-Tolerance: Concepts and Terminology," in Proc. 15th Int. Symp. on Fault-Tolerant Computing, 1985, pp. 2-11.

/2/ R. A. Howard, "Dynamic Probabilistic Systems, vol. 1," New York, John Wiley and Sons, 1971.

/3/ J. Arlat and J.C. Laprie, "Performance-related Dependability Evaluation of Supercomputer Systems," in Proc. 13th Int. Symp. on Fault-Tolerant Computing, 1983, pp. 276-283.

/4/ M. Smotherman, R. M. Geist and K.S. Trivedi, "Provably Conservative Approximations to Complex Reliability Models," IEEE Trans. on Computers, vol. C-35, no. 4, pp. 333-338, April 1986.

/5/ J.A. Carrasco and J. Figueras, "METFAC: Design and Implementation of a Software Tool for Modeling and Evaluation of Complex Fault-Tolerant Computing Systems," in Proc. 16th Int. Symp. on Fault-Tolerant Computing, 1986, pp. 424-429.

/6/ A. Costes, C. Landrault and J.C. Laprie, "Reliability and Availability Models for Maintained Systems Featuring Hardware Failures and Design Faults," IEEE Trans. on Computers, vol. C-27, no.6, pp. 548-560, June 1978.

/7/ F. Gustavson, "Finding the Block Lower Triangular Form of a Sparse Matrix," in Sparse Matrix Computations, New York, Academic Press, 1976 pp. 275-289.

/8/ W.J. Stewart and A. Goyal, "Matrix Methods in Large Dependability Models," IBM Research Report RC-11485, Yorktown Heights, November 1985.

/9/ R. S. Varga, "Matrix Iterative Analysis," Englewood Cliffs, NJ, Prentice-Hall, 1962.

/10/ J.M. Nahman, "Iterative Method for Steady State Reliability Analysis of Complex Markov Systems," IEEE Trans. on Reliability, vol. R-33, no. 5, pp. 406-409, December 1984.

/11/ W.L. Cao and W.J. Stewart, "Iterative Aggregation/Disaggregation Techniques for Nearly Uncoupled Markov Chains," Journal of the ACM, vol. 32, no. 3, pp. 702-719, July 1985.

/12/ Juan A. Carrasco, "An Exact, one-step Aggregation Technique and its Application to the Acceleration of Iterative Methods for Dependability Evaluation," Technical Report Departament d'Enginyeria Electrónica, UPC, in preparation.

/13/ D.R. Miller, "Reliability Calculation Using Randomization for Markovian Fault-Tolerant Computing Systems," in Proc. 13th Int. Symp. on Fault-Tolerant Computing, 1983, pp. 284-289.

/14/ W.L. Miranker, "Numerical Methods for Stiff Equations and Singular Perturbation Problems," Boston, D. Reidel Pub. Co., 1981.

/15/ A. Bobbio and K.S. Trivedi, "An Aggregation Technique for the Transient Analysis of Stiff Markov Chains," IEEE Trans. on Computers, vol. C-35, no. 3, pp. 803-814, September 1986.