MATHEMATICAL MODELLING AND SCIENTIFIC COMPUTING, Vol. 8 (1997)

VALIDATING AN ANALYTICAL APPROXIMATION THROUGH DISCRETE SIMULATION

Jehan-François Pâris

Computer Science Department, University of Houston, Houston, TX 77204-3475
paris@cs.uh.edu

ABSTRACT

While Markov models have been extensively used to study the availability of replicated data, they cannot handle effectively network configurations where sites failures and network partitions have to be simultaneously considered. We had proposed in a previous paper a *hierarchical decomposition method* aimed at overcoming this limitation. While our method could provide closed form estimates of the availability of replicated objects whose replicas reside on networks subject to communication failures. We present here a simulation study measuring the quality of our estimates and attempting to improve upon them.

KEYWORDS

fault-tolerance, replicated systems, redundancy, voting.

INTRODUCTION

Managing replicated data can be a demanding task particularly when the replicas are stored at different sites of a computer network. Special *replication control protocols* have been devised to perform this task without user intervention and maintain the replicated data in a consistent state.

Evaluating the performance of these protocols, and especially the *data availabilities* they afford is a very important issue because they vary greatly in their complexity, their communication overhead, and the protection they provide or do not provide against network partitions. Several techniques have been used to evaluate the availability of replicated data. Combinatorial models are very simple to use (Pu et al. 1988, van Renesse and Tanenbaum 1988) but cannot represent complex recovery modes as these found in some of the most efficient replication control protocols. Simulation models can be very accurate whenever all the parameters of the modeled system are known but this is rarely the case.

As a result, stochastic models have become the method of choice for evaluating the availability of replicated data managed by protocols with complex recovery modes (Pâris 1986, Jajodia and Mutchler 1987, Ahamad and Ammar 1987). Unfortunately, stochastic models that take simultaneously into account site failures and network partitions become quickly untractable because their number of states increases exponentially with the number of failure modes being considered. As a result, all analytical studies of the availability of replicated data have either assumed that the sites could not fail or that the network could not partition.

One possible solution to this problem is the use of *simplified* Markov models. We had presented in a previous paper (Pâris 1992) a *hierarchical decomposition* method aimed at overcoming this limitation. While our method could provide closed form estimates of the availability of replicated objects whose replicas reside on networks subject to communication failures, we had no way to evaluate the accuracy of these estimates. We present here a simulation study aimed at measuring the quality of our estimates and attempting to improve upon them.

THE HIERARCHICAL DECOMPOSITION METHOD

Hierarchical decomposition reduces the complexity of the model itself by identifying parts of the system that can be studied in isolation and replaced by simpler equivalent components (Courtois 1977, Ferrari et al. 1983). This technique has been widely used in computer systems performance evaluation to solve queuing models too complex to be directly tractable. It can also be applied to the evaluation of the availability of replicated data objects (Pâris 1991).

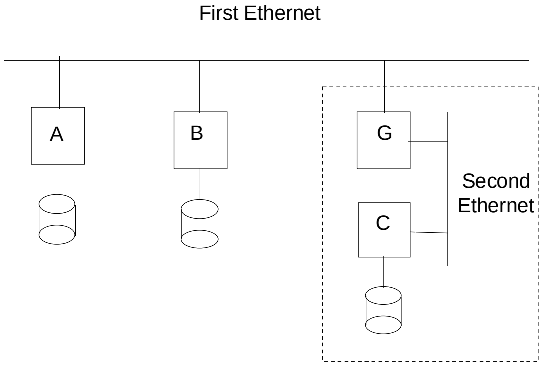

Figure 1: A small local area network with three nodes on two network segments

Many local-area networks consist of several carrier-sense segments or token rings linked by gateway machines. Figure 1 shows a very simple example of such networks: it consists of three nodes *A*, *B* and *C* located on two Ethernets linked by a gateway node *G*. Since gateways can fail without causing a total network failure, such networks can be partitioned. The key difference with conventional point-to-point networks is that sites that are on the same carrier-sense network or token ring will never be separated by a partition. We will refer to these entities as *network segments* (van Renesse and Tanenbaum 1988).

We propose to model these networks as a set of nodes and gateways with independent failure modes. When a node or a gateway fails, a repair process is immediately initiated at that site. Should several sites fail, the repair process will be performed in parallel on those failed sites. We assume that failures are exponentially distributed with mean failure rate λ, and that repairs are exponentially distributed with mean repair rate μ. The system is assumed to exist in statistical equilibrium and to be characterized by a discrete-state Markov process.

Assume now that the three nodes *A*, *B* and *C* are used to store the three copies of a replicated file *X*. Under a static voting protocol such as majority consensus voting (Ellis 1977, Gifford 1979), the replica on node *C* will only be able to be counted in quorums when the gateway *G* is operational. For all practical purposes, a failure of *G* will thus have the same effect as a failure of *C*. We propose therefore to replace the subsystem consisting of site *C* and its gateway *G* by an *aggregate site C'* that will remain operational as long as both *C* and *G* are operational. A replicated file having replicas on the two nodes *A* and *B* and the aggregate site *C'* would have the same availability as the replicated file *X* but will be much easier to investigate since we will not have to consider gateway failures.

To compute the failure rate λ' and the repair rate μ' of the aggregate site *C'*, we need to observe that *C'* will remain operational as long as both the gateway *G* and the node holding the replica *C* are both operational. The states of *C* and *G* can thus be represented by the state transition diagram of figure 2, which consist of four states numbered from $\langle 00 \rangle$ to $\langle 11 \rangle$. State $\langle 11 \rangle$ represents the state of the aggregate site when the node and its gateway are both operational. States $\langle 01 \rangle$ and $\langle 10 \rangle$ represent states when either the site or its gateway have failed while site $\langle 00 \rangle$ corresponds to a failure of both entities.

We can immediately derive the steady state probabilities for the four states:

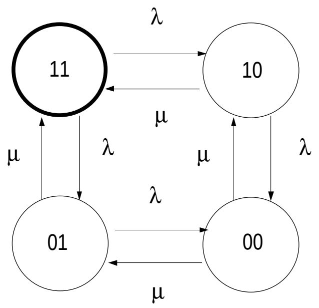

Figure 2: State transition diagram for an aggregate site consisting of one node and one gateway

$$p_{11} = \frac{\mu^2}{(\lambda+\mu)^2}, \quad p_{10} = p_{01} = \frac{\lambda\mu}{(\lambda+\mu)^2}, \quad p_{00} = \frac{\lambda^2}{(\lambda+\mu)^2}$$

Since the only available state for the aggregate site is state ⟨11⟩, the failure rate λ' and the repair rate μ' of the aggregate site *C'* are given by:

$$\lambda' p_{11} = 2\lambda p_{11}$$

and

$$\mu'(1-p_{11}) = \mu(p_{10}+p_{01})$$

which have as unique solution $\lambda' = 2\lambda$ and $\mu' = \frac{1}{1+\frac{\lambda}{2\mu}}$

This aggregation technique can be trivially extended to replicated objects consisting of an arbitrary number of replicas located on a network consisting of network segments linked by gateways. Note however that the hierarchical decomposition method assumes that replicas located on nodes that become part of a given aggregate site can never become a majority by themselves. This assumption is correct for majority consensus voting as long as an aggregate site does not contain a majority of the replicas. There are however several voting protocols that allow a minority of the replicas to have a majority of the votes. Whenever this is the case, the hierarchical decomposition method will *underestimate* the availability of the replicated data.

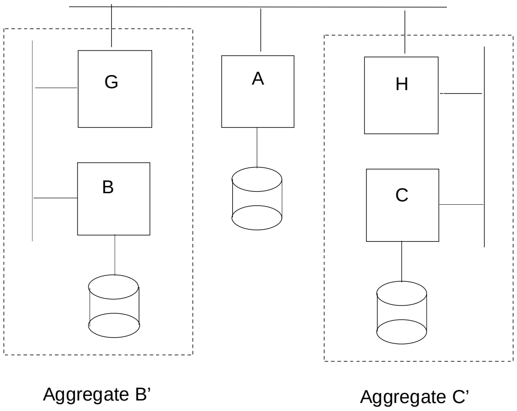

Figure 3: Three nodes A, B and C on three network segments linked by two gateways

THE VALIDATION STUDY

Being aware that our hierarchical dccomposition method could sometimes underestimate the availability of the replicated data, we decided therefore to compare the data availabilities computed through our hierarchical methods with those obtained by simulating the same configurations over a wide range of site failure and repair rates. The simulator we utilized in our study has been described in more detail elsewhere (Pâris et al. 1988). It uses the batch means method to compute 95 percent confidence intervals of the unavailability of the replicated data. In order to reduce the initial bias, we excluded all measurements collected during the first simulated 180 days. We decided to investigate three distinct replica configurations managed by two distinct voting protocols. These configurations were: (a) three replicas on one Ethernet, (b) three replicas on two Ethernets linked by one gateway(as in our previous example), and (c) three replicas on three Ethernets linked by two gateways (as on figure 3).

We selected *majority consensus voting* for its simplicity and the *dynamic-linear voting* protocol (DLV) (Jajodia and Mutchler 1987) for its excellent performance. We did not investigate configurations with two replicas since voting protocols require a minimum of three voting sites to improve upon the availability of a single copy.

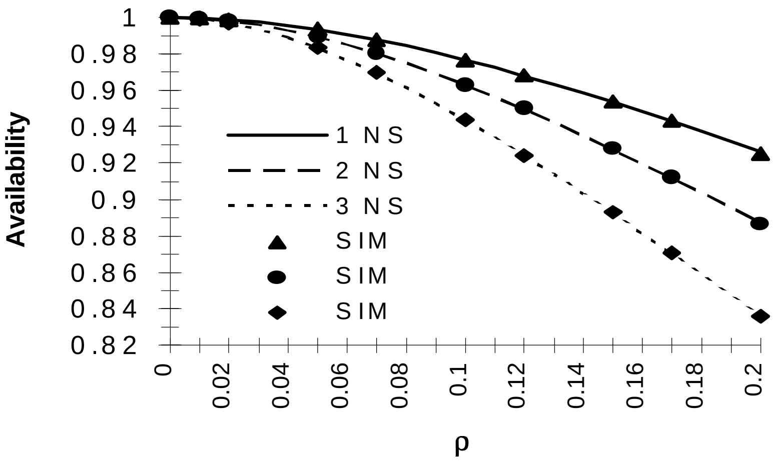

Figure 4: Compared availabilities for majority consensus voting

The results for majority consensus voting are summarized on figure 4 where the three curves correspond to the three configurations under study and $\rho = \lambda/\mu$ represents the ratio of the failure rate over the repair rate. The solid curves were obtained by substituting the proper values of the availabilities of the three replicas to p_1, p_2 and p_3 in:

$$A_{MCV}(3) = p_1 p_2 p_3 + (1 - p_1) p_2 p_3 + (1 - p_2) p_3 p_1 + (1 - p_3) p_1 p_2$$

while the isolated values were obtained through our simulator. As one can see, the results of our simulations were in perfect agreement with those derived from our hierarchical decomposition method. This was expected since none of the aggregate sites contained a majority of the replicas.

As figure 5 shows, the results for dynamic-linear voting were quite different: while the simulation results agreed with the analytical results in two of the three configurations, the simulation results for three replicas on three network segments were systematically higher than those achieved through our hierarchical decomposition method. This discrepancy results from the fact that the dynamic-linear voting protocol adjusts its quorums to reflect changes in replica availability and network connectivity. Hence the failure of any of the three replicas will result into a the establishment of a new quorum. This new quorum will consist of the replica that comes first in lexicographic ordering of the two remaining replicas. Thus, if node *A* fails first, the replica on node *B* will become a quorum by itself because *B* > *C*. The hierarchical decomposition method will then *underestimate* the availability of the replicated data because it will assume that the replica on node *B* cannot be accessed when its gateway *G* is not operational.

This can better seen on the (rather complex) state transition diagram for three replicas on three network segments under the dynamic-linear voting protocol. As figure 6 shows, the replicated data are in state ⟨12⟩ when the three nodes holding replicas and the two gateways are operational. A failure of node *A* appears on the diagram as a transition from state ⟨12⟩ to state ⟨02⟩. Since replica *B* is now a majority by itself, a failure of the aggregate site *C'* would leave the replicated data in state ⟨01⟩, which still allows access while a failure of the aggregate site *B'* would move the system to state ⟨01''⟩ , which does not allow access. We speculated first that the discrepancy between the simulation data and the results of the hierarchical decomposition method could result from the fact that once the system is in state ⟨01⟩, it should remain in this state as long as *B* remains operational regardless of the status of the gateway *G*.

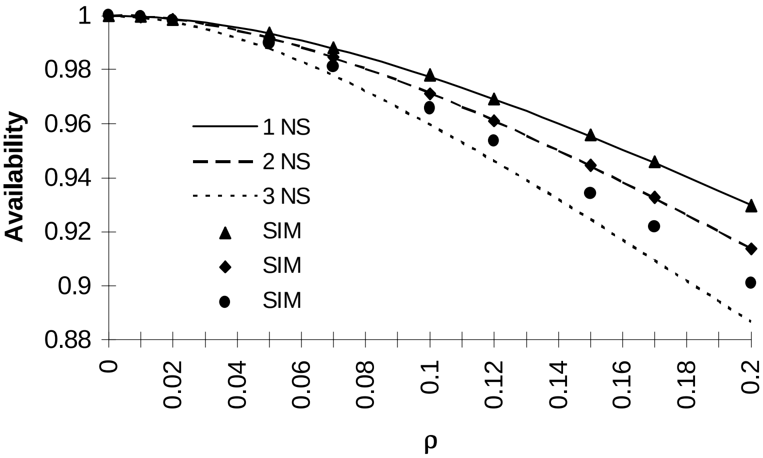

Figure 5: Compared availabilities for dynamic-linear voting

We decided to change the state transition rate between states ⟨01⟩ and ⟨00''⟩ from λ' to λ (which means *halving* it since λ' = 2λ). At our great surprise, this had no significant effect on the availability figures.

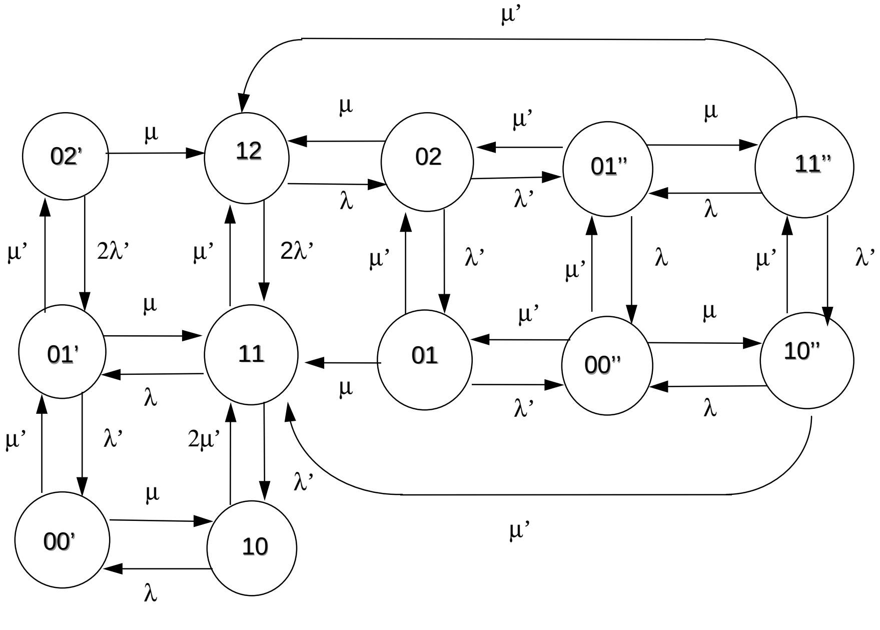

Figure 6: State transition diagram for three replicas on three segments managed by DLV

We then considered another explanation. When the system is in state ⟨02⟩ any failure of the gateway *G* will result in a failure of the aggregate site *B'* and move the system to the unavailable state ⟨01''⟩. This is not correct because a failure of *G* would leave *B* capable to continue to process requests since it already constitutes a majority by itself. We decided then to change the transition rate from state ⟨02⟩ to ⟨01''⟩ from λ' to λ. As figure 7 indicates, this brought the analytical results in much better agreement with the simulation results.

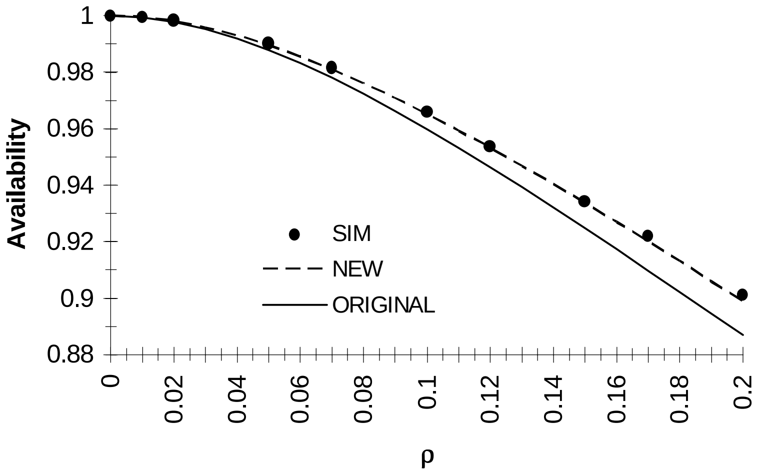

Figure 7: Tuning the Markov model of the DLV protocol for three replicas on three segments

DISCUSSION

While Markov models have been extensively used to study the availability of replicated data, they cannot handle effectively network configurations where sites failures and network partitions have to be simultaneously considered. We had proposed in a previous paper a *hierarchical decomposition method* aimed at overcoming this limitation. While our method could provide closed form estimates of the availability of replicated objects whose replicas reside on networks subject to communication failures. We have presented here a simulation study aimed at measuring the quality of our estimates and attempting to improve upon them. We gained from our study a better understanding of the limitations of our hierarchical decomposition method and one possible method to improve upon its accuracy.

REFERENCES

Pu, C., J. D. Noe and A. Proudfoot, "Regeneration of Replicated Objects: A Technique and its Eden Implementation," *IEEE Transactions on Software Engineering,* SE-14, 7 (1988), pp. 936-945.

van Renesse, R., and A. Tanenbaum, "Voting with Ghosts," *Proc. 8th Int. Conf. on Distributed Computing Systems,* (1988), pp. 456-462.

Pâris, J.-F., "Voting with Witnesses: A Consistency Scheme for Replicated Files," *Proc. 6th Int. Conf. on Distributed Computing Systems,* (1986), pp. 606-612.

Jajodia, S. and D. Mutchler, "Enhancements to the Voting Algorithm," *Proc. 13th VLDB Conf.,* (1987), pp. 399-405.

Ahamad, M., and M. H. Ammar, "Performance Characterization of Quorum-Consensus Algorithms for Replicated Data," *IEEE Trans. on Software Engineering,* SE-15, 4 (1989), pp. 492-496.

Ellis, C. A., "Consistency and Correctness of Duplicate Database Systems," *Operating Systems Review,* 11 (1977).

Gifford, D. K., "Weighted Voting for Replicated Data," *Proc. 7th ACM Symp. on Operating System Principles,* (1979), pp. 150-161.

Courtois, P. J., *Decomposability: Queuing and Computer System Applications,* Academic Press, New York (1977).

Ferrari, D., G. Serazzi and A. Zeigner, *Measurement and Tuning of Computer Systems,* Prentice-Hall, Englewood Cliffs, NJ (1983).

Pâris, J.-F., D. D. E. Long and A. Glockner, "A Realistic Evaluation of Consistency Algorithms for Replicated Files," *Proc. 21st Annual Simulation Symp.,* (1988), pp. 121-130.

Pâris, J.-F., "Evaluating the Impact of Network Partitions on Replicated Data Availability," in J.F. Meyer, R. D. Schlichting (eds.) *Dependable Computing for Critical Applications,* Springer Verlag, (1992), pp. 49-65.