Analysis of Replication Control Protocols

Darrell D. E. Long
University of California, Santa Cruz

darrell@cis.ucsc.edu

Abstract

In recent years many replication control protocols have been proposed, but too often these protocols are presented with insufficient evidence to demonstrate superiority over existing protocols. In this article, some simple analytical tools are presented that allow replication control protocols to be compared. A *dynamic voting* protocol is used as an example.
Keywords: fault tolerance, replication, data management.

1 Introduction

Too often replication control protocols are presented with insufficient analysis to compare them to existing protocols. Several tools exist for studying the performance of replication control protocols. These include Markov analysis (the focus of this article), combinatorial arguments and discrete event simulation.

Each of these methods provides some level of insight, but each also has its limitations. For example, Markov analysis is incapable of modeling network partitions in the most general case. Combinatorial analysis neglects recovery states where a site may be operational but unable to grant access to the data object. Discrete event simulation can model partitionable networks, but is constrained to providing only numeric results for specific system parameters.

Common measures of dependability include *reliability*, which is the probability that a replicated data object will remain continuously accessible over a fixed time period, and *availability*, which is the steady-state probability that the data object is accessible at any given moment. Availability has received much attention, in part because its analysis is more tractable than that of reliability.

In this article, Markov analysis is used to estimate both performance measures. The *optimistic dynamic voting* protocol (ODV) [4] is used as an example, though these techniques are applicable to most replication control protocols.

1.1 Optimistic Dynamic Voting

The optimistic dynamic voting protocol (ODV) is a variant of the *dynamic-linear voting* protocol (DLV) [1] that records at each access the names of the participating sites instead of a simple count. This is sufficient to ensure mutual consistency among the replicas of the data object, and makes the protocol more amenable to extensions such as *witnesses* [3] and *regeneration* [5,2]. Since ODV maintains system state information at each access, its performance, like that of DLV, is related to the access rate.

There are three pieces of information that must be maintained at each site to implement ODV. The *partition set*, representing the set of sites that participated in the last successful operation including the site, is used to determine the required quorum for the next access. The partition sets are maintained when either a read or write occurs, and are brought up-to-date when a site recovers from a failure.

The *version number* represents the number of writes accepted by the replica, and the *operation number* represents the number of successful accesses to the replica.

The sites holding a current version of the partition set, determined by the operation numbers, are called the *quorum set*. If the quorum set represents a majority of the previous quorum the access request will be granted. If there is a tie, a total ordering on the set of sites is used to decide if access will be granted.

2 Analytic Models

The system model consists of a set of sites with independent failure modes connected by a network that is assumed not to fail. When a site fails, repair is immediately initiated. Should several sites fail, the repair process will be performed in parallel. Site failures are assumed to be exponentially distributed with mean λ, and repairs with mean μ. Access requests are characterized by a Poisson process with mean κ.

Failures of the communication network are not considered since doing so would result in models with an intractable number of states. Thus, this analysis applies to environments where network partitions are impossible or have a negligible probability.

2.1 Availability

In the case of replicated data objects, availability represents the steady-state probability that the replication control protocol will allow access to the data object, and strongly depends on the replication control protocol.

Definition 2.1 *The* availability, $A_P(n)$, *of a data object consisting of n replicas managed by protocol P is the steady-state probability of the data object being in a state where the protocol will grant access.*

The availability provided by ODV is characterized by the rate at which accesses occur. When the accesses are frequent, the information about the system state is closer to the truth, improving the availability of the replicated data object.

The state transition diagram for three replicas is shown in figure 1. The states are labeled by ordered pairs where the first coordinate represents the number of sites which are believed to be available, and the second the actual number of available sites. The states marked with a bar represent states where access would be denied.

The transitions between states fall into several categories: failure, recovery and access. Transitions such as $\langle 3,3\rangle \rightarrow \langle 3,2\rangle$, represent a site failure. Transitions such as $\langle 2,2\rangle \rightarrow \langle 3,3\rangle$, represent a site recovering from a failure. Access transitions such as $\langle 3,2\rangle \rightarrow \langle 2,2\rangle$, occur when an access request is granted.

The probability of the system being in an available state $\langle i,j\rangle$ is represented by $p_{i,j}$, and of being in an unavailable state $\langle \overline{i,j}\rangle$ by $\bar{p}_{i,j}$. The resulting equations, along with the boundary condition $\sum_{i,j} p_{i,j} + \sum_{i,j} \bar{p}_{i,j} = 1$, can be solved using standard techniques.

The availability is given by the sum of probabilities of being in a state where access would be granted, and for three replicas is given by:

$$A_{ODV}(3) = \frac{\rho^3 + 3\rho^2 + 4\rho + 1}{(\rho+1)^4} - \xi(3),$$

where

$$\xi(3) = \frac{\rho^3}{(\rho+1)^4(\phi + 2\rho + 1)}$$

with $\rho = \frac{\lambda}{\mu}$ and $\phi = \frac{\kappa}{\mu}$.

For the sake of comparison, consider the availability provided by *majority consensus voting* (MCV),

$$A_{MCV}(n) = \sum_{j=n}^{\lceil n/2 \rceil} \frac{\binom{n}{n-j}\rho^{n-j}}{(1+\rho)^n},$$

for n odd [3], and by DLV with perfect system state information [4],

$$A_{DLV}(3) = \frac{\rho^3 + 3\rho^2 + 4\rho + 1}{(\rho+1)^4}.$$

Since $\lim_{\phi\rightarrow\infty} \xi(n) = 0$, the availability provided by ODV quickly converges to that of

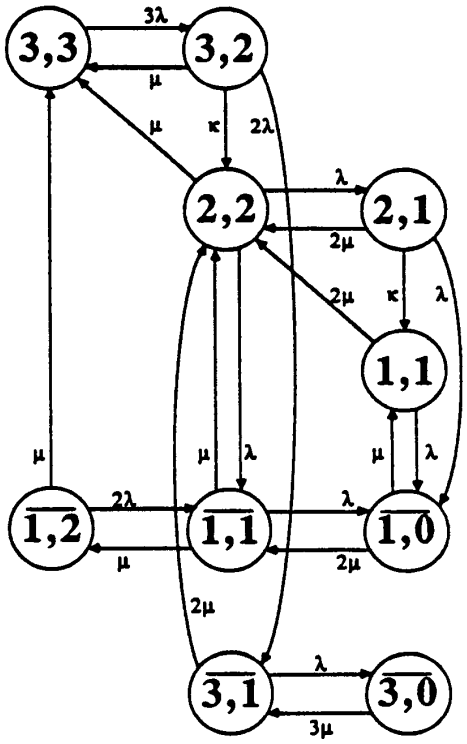

Figure 1: State Transition Diagram for Three Sites

instantaneous DLV. Even for modest access rates, the advantage of using ODV over using MCV can be seen. Even when state information is exchanged only at recovery time ODV is an improvement over MCV.

Figure 2 shows the availabilities provided with four replicas when MCV, instantaneous DLV, and ODV are employed. As expected, the performance of ODV depends on the access rate, and lies between that of the other two protocols.

A hierarchy can be established for availability afforded by each of these protocols. In general, $\mathcal{A}_{DLV}(n) > \mathcal{A}_{ODV}(n) > \mathcal{A}_{MCV}(n)$, $n \geq 3$.

2.1.1 Transient Availability

For some protocols, the transient behavior may be of interest. The transient availability of a replicated data object can be modeled using the same state transition diagram employed for availability, by relaxing the steady-state assumption and using the resulting system of explicit differential equations [7].

For example, a single host can be modeled by,

$$\frac{dp(t)}{dt} = \mu q(t) - \lambda p(t)$$
$$\frac{dq(t)}{dt} = \lambda p(t) - \mu q(t)$$

with initial conditions $p(0) = 1, q(0) = 0$. Here $p(t)$ is the probability of the host being operational at time t, and $q(t)$ of it being in a failed state. The solution to this system of equations is given by

$$p(t) = \frac{\mu}{\lambda + \mu} + \xi(t), q(t) = \frac{\lambda}{\lambda + \mu} - \xi(t)$$

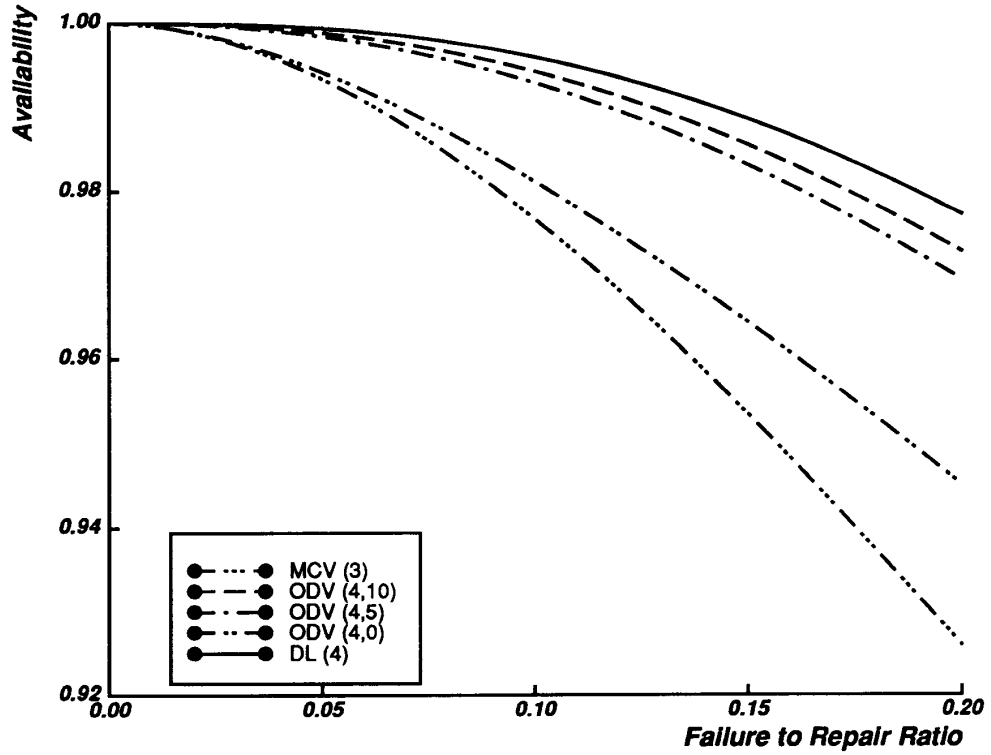

Figure 2: Availability of ODV with Four Replicas

with

$$\xi(t) = \frac{\lambda e^{-t(\lambda+\mu)}}{\lambda+\mu}.$$

When $\lambda, \mu > 0$, the $\lim_{t\to\infty} \xi(t) = 0$ converges at an exponential rate. Thus, a single site quickly reaches steady-state.

For larger systems, the resulting system of differential equation will be complex. Numerical techniques should be used to obtain the solutions since a direct method for obtaining a closed-form solution may not exist. Even if a closed-form solution could be found, it is unlikely to provide any insight into the behavior of the system.

2.2 Reliability

Reliability is a measure of the behavior of a system under transition, and its analysis is much less tractable than availability. For many applications, it is a more important measure than availability. These applications are characterized by the property that any interruption of service is intolerable.

Definition 2.2 *The* reliability $\mathcal{R}_P(n,t)$ *of a data object composed of n replicas managed by protocol P is probability that the system will operate correctly over a continuous interval of duration t given that all n replicas were operating correctly at $t = 0$.*

The same set of stochastic hypotheses are necessary to model reliability. The main difference is that steady-state behavior cannot be considered since only the first time the replicated data object becomes unavailable is of interest.

The differential equations describing the behavior of a replicated data object can be derived from the state-transition flow rate

diagrams. A replicated data object is in state 0 if it has been inaccessible at some point in the past. Thus, no transitions are permitted from state 0, since only the behavior of the system prior to the first total failure is of interest.

For small numbers of sites, closed-form solutions for the reliability of some of the protocols can be obtained from the differential-difference equations. Less tractable systems can be both simulated and solved numerically.

Systems of differential equations such as these impose a fundamental limitation on any numerical solution since there are no transitions from the failed state. The resulting systems of linear differential equations must have a zero eigenvalue, and since the remaining eigenvalues are all negative the system is "infinitely stiff" [6].

The state-transition flow rate diagram for ODV is shown in figure 3. The transitions among the available states remain the same as for the availability model, with the exception of transitions to a failed state. All failed states have been merged into a single state since only the time until the first failure is of interest.

The graph shown in figure 4, presents the reliability of four replicas managed by ODV. Several values of the access rate ϕ have been chosen, in particular, $\phi = 0, 1, 5, 10, \infty$. When $\phi = \infty$ the ODV protocol provides the same reliability as DLV with perfect state information. The value of ρ has been set at 0.05, but the relationship between protocols holds for other values as well. An important observation is that the effect of the access rate is greater on reliability than availability.

References

[1] S. Jajodia and D. Mutchler, "Dynamic voting," in *SIGMOD International Conference on Data Management*, pp. 227–238, ACM, 1987.

[2] D. D. E. Long and J.-F. Pâris, "Regeneration protocols for replicated objects," in *Proceedings 5th International Conference on Data Engineering*, (Los Angeles), IEEE, February 1989.

[3] J.-F. Pâris, "Voting with witnesses: A consistency scheme for replicated files," in *Proceedings 6th International Conference on Distributed Computing Systems*, (Cambridge), pp. 606–612, IEEE, 1986.

[4] J.-F. Pâris and D. D. E. Long, "Efficient dynamic voting algorithms," in *Proceedings 4th International Conference on Data Engineering*, (Los Angeles), pp. 268–275, IEEE, 1988.

[5] C. Pu, J. D. Noe, and A. Proudfoot, "Regeneration of replicated objects: A technique and its Eden implementation," in *Proceedings 2nd International Conference on Data Engineering*, (Los Angeles), pp. 175–187, IEEE, 1986.

[6] L. H. Shampine and C. W. Gear, "A user's view of solving stiff ordinary differential equations," *SIAM Review*, vol. 21, pp. 1–17, 1979.

[7] K. S. Trivedi, *Probability & Statistics with Reliability, Queuing and Computer Science Applications*. Englewood Cliffs, New Jersey: Prentice-Hall, 1982.

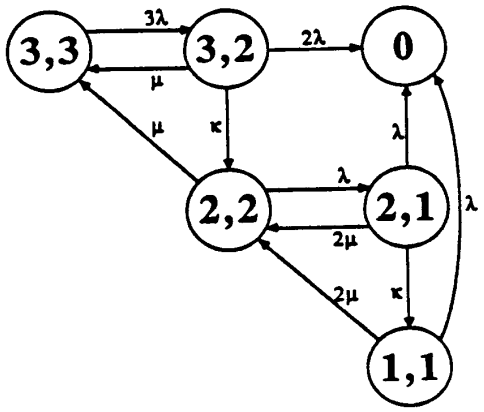

Figure 3: State Transition Diagram for Three Sites

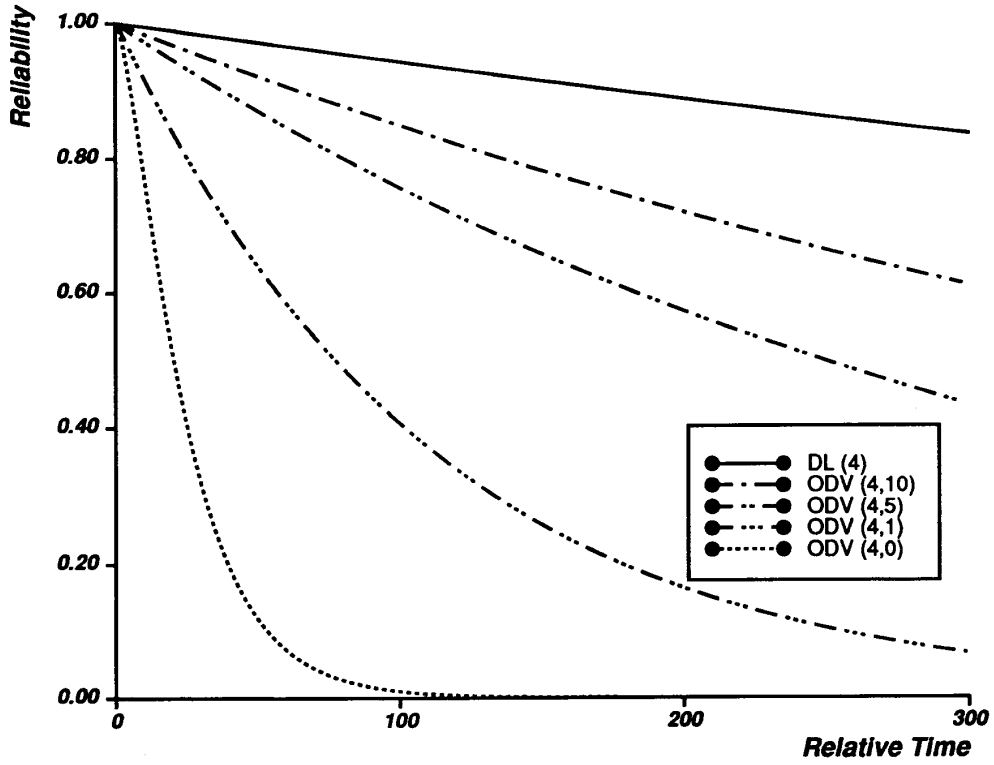

Figure 4: Reliability of ODV with Four Replicas