Reliability of Fully & Partially Replicated Systems

Ing-Ray Chen, Member IEEE
University of Mississippi, University
Farokh B. Bastani, Member IEEE
University of Houston, Houston

Key Words — **Fault-tolerance, Replication technique, Markov process, Reliability evaluation.**

Reader Aids —
Purpose: Evaluate a new replication method
Special math needed for explanations: Probability
Special math needed to use results: Same
Results useful to: Computer designers and reliability analysts

Summary & Conclusions — **The reliability of two replication methods, partial and full, is analyzed for a 2-process/4-processor case. Three types of systems are analyzed and compared: no repair, finite repair-rate, and instantaneous repair. These systems are modeled using discrete-state continuous-time Markov chains. The condition under which a partially replicated system might yield a better reliability than that of a fully replicated system is quantified and expressed in terms of system design parameters. Partial replication is most favorable in systems without repair capability and this advantage is manifested most when the underlying hardware is unreliable.**

1. INTRODUCTION

Replication normally refers to full replication in which a module of a system is fully replicated to provide several identical copies. For example, in the Electronic Switching System [2], the CPU is duplicated and in the Fault Tolerant Multi-Processor System [3] the bus and processor modules are triplicated. We have proposed [4,5] a replication technique for process-control systems in which a module can be only partially replicated for systems where control data are smooth functions of time. Figures 1 & 2 show an example of distributing 2 process modules, a, b, among 4 processors, p_1, p_2, p_3, p_4; a/p_i means that processor p_i is allocated to process a. Both a and b are critical components and, hence, must not fail. A full replica (a or b) of a process module consumes the full processing power of a processor.

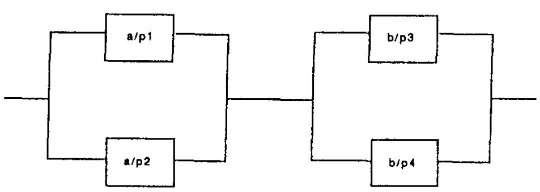

Figure 1. Full Replication

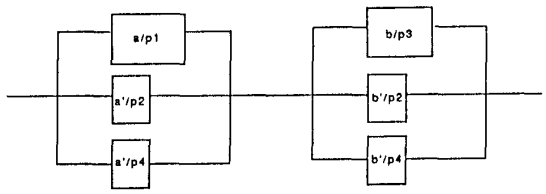

Figure 2. Partial Replication

Figure 1 shows full replication being used. Each processor is allocated to either a or b. On the other hand, figure 2 illustrates partial replication where a process module is replicated to yield one full replica (eg, a) and two partial replicas (eg, a'). For illustration, we assume that a partial replica in this case uses only 0.5 of the processing power of the processor to which it is allocated. This is achieved by restricting the processing and information flow to a partial replica in a ratio that is proportional to 0.5.

Partial replication is possible when information can be stored at various levels of detail, ranging from complete information to summarized information. A full replica has complete information and provides full control while a partial replica has only summarized information and is in standby mode. When a full replica fails, a partial replica can immediately provide temporary control based on the summarized information it has while it gradually acquires additional information to become a full replica.

Chen & Bastani [5] have demonstrated the feasibility of partial replication with a case study of a simulated chemical batch reactor. The system has to control the temperature profile in a reaction chamber to maximize the yield of a certain reactant. The full replica samples the temperature every δt seconds and adjusts the heating and cooling sources to bring the temperature to the desired level. A partial replica samples the temperature every $n \cdot \delta t$ seconds, $n \geq 2$. It is on standby to take over control when the full replica fails. In the standby mode, it needs only $1/n$ of the processing power required by the full replica. This allows several partial replicas to run on the same backup processor. When a partial replica needs to become a full replica, the processor is then devoted to that replica. This can require one or more other partial replicas on that processor to cease execution in order to meet the processing requirements of the new full replica.

In this paper, we are interested in evaluating the system reliability of fully and partially replicated systems. Continuous-time Markov chain models [1, 9] are used to describe the time behavior of these systems. Our reliability model focuses on a 2-process/4-processor case. Other cases can be analyzed similarly. Three cases are evaluated:

1. nonrepairable (zero repair-rate) system,
2. repairable systems with finite repair rate,

3. repairable systems with instantaneous repair (infinite repair-rate). □

Case #3 models the situation in which there exists an infinite number of off-line spare processors that can be used to replace failed on-line processors instantaneously. Our objective is to identify design conditions under which partial replication is preferable to full replication and vice versa.

Section 2 lists the assumptions & notation. Section 3 provides the reliability analysis of fully and partially replicated systems based on Markov chain models. Section 4 evaluates design parameters for optimizing system reliability for the same cost function.

2. ASSUMPTIONS & NOTATION

2.1 *Assumptions*

1. Repair and failure times of processors are exponentially distributed r.v.'s with a constant repair rate (for finite repair-rate) and a constant failure rate.
2. Processors are fail-stop.
3. When a processor is repaired and brought back on line, the time required to recover processes allocated to it is exponentially distributed with a constant recovery rate.
4. A full replica of a process requires the full processing power of a processor and serves as either a primary replica or a hot standby of that process [5]. This assumption is adopted to ease the reliability analysis.
5. A partial replica of a process requires half of the processing power of a processor and serves as a warm standby of that process. □

2.2 *Notation*

p_i	processor i
a,b	full replica of process module a,b
a',b'	partial replica of the process module a,b
$r(t)$	system reliability at time t
ϕ	a newly repaired processor
λ,δ	processor failure, repair rate
$\mu_{a\phi}$	recovery rate of a full replica of process a on a newly repaired processor ϕ
$\mu_{a'b'\phi}$	recovery rate of a partial replica of process a and a partial replica of process b from a newly repaired processor ϕ
$\mu_{aa'}$	recovery rate of a partial replica of process a to a full replica of process a
$\mu_{bb'}$	recovery rate of a partial replica of process b to a full replica of process b,
θ	software failure rate of a partial replica
$s_1 \rightarrow s_2$	a state transition from state s_1 to state s_2
f	failure state
$\Pr_f\{t\}$	Pr{system is in state f at time t, given that it was in the initial state at time 0} □

Other, standard notation is given in "Information for Readers & Authors" at the rear of each issue.

3. RELIABILITY MODELS

Reliability assessment of replicated systems normally falls within the following classes.

1. Systems with repair capability, eg, electronic switching systems,
   a. a failed module can be repaired with a finite repair-rate,
   b. it can be repaired instantaneously (eg, if there is a standby processor pool for replacing processors that have failed).
2. Systems without repair capability, eg, spacecraft control systems. □

The reliability analysis of a repairable system is, in general, more complicated than that of a nonrepairable system because we need to consider the repair rate. We consider these cases separately in the following subsections.

3.1 *Nonrepairable Systems*

Assumptions

1. A processor functions for an exponentially distributed time with rate λ and then fails. Once a processor fails it stays down because there is no repair capability in the system.
2. When a partial replica takes over, it takes an exponentially distributed time (the vulnerable period) to become a full replica.
3. During a partial replica's vulnerable period, there is a software failure rate, θ, representing the rate at which a partial replica fails to deal with a task which would require its data structure to be completely up-to-date. □

The failure event in assumption #1 is represented by removing the processes which are originally running on the failed processor, from the state description. For example, the failure of either p_1 or p_2 in state (a,a,b,b) is described by a state transition $(a,a,b,b) \rightarrow (a,b,b)$. This transition shows that one full replica of a has been removed from the system. It has a transition rate of $2\cdot\lambda$ since it is triggered by the failure of either p_1 or p_2.

The Markov chains in figures 3 & 4 model a fully replicated and a partially replicated nonrepairable system, corresponding to figures 1 & 2, respectively. In the fully replicated system, when a full replica (eg, a) fails, the other full replica (eg, a) takes over instantaneously. The reliability of the fully replicated system is:

$$r_{\text{full}}(t) = 1 - \Pr_f\{t\}, \text{ for the initial state } = (a,a,b,b).$$

$\Pr_f\{t\}$ can be obtained by solving the set of differential equations describing the Markov chain in figure 3, using Laplace transforms:

$$r_{\text{full}}(t) = 4\cdot\exp(-2\cdot\lambda\cdot t) - 4\cdot\exp(-3\cdot\lambda\cdot t) + \exp(-4\cdot\lambda\cdot t).$$

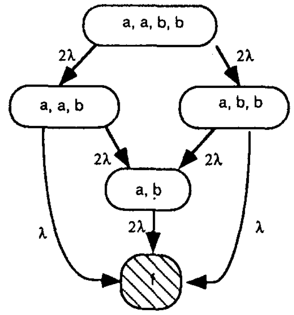

Figure 3. Full Replication Model [No Repair]

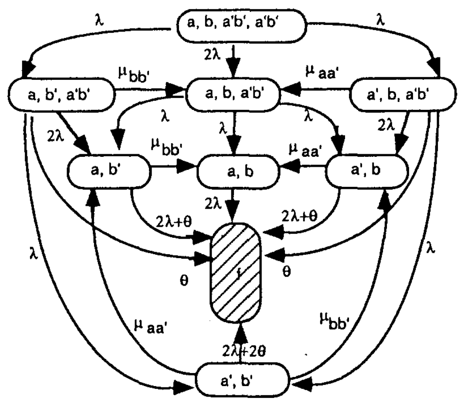

Figure 4. Partial Replication Model [No Repair]

In the partially replicated system (figure 4), when a full replica (eg, a) fails, a partial replica (eg, a') switches in to take over. Then a processor that was originally labeled $a'b'$ is selected to make a transition to a full replica, either a or b depending on whether a or b failed. For example, when a fails in state $(a, b, a'b', a'b')$ the event is described by $(a, b, a'b', a'b') \rightarrow (a', b, a'b')$. This means that a processor originally performing $a'b'$ is now locked into a recovery transition from a' to a. A partial replica taking over might involve increasing the data acquisition frequency to that of a full replica until the information in its data structures is brought completely up-to-date [5]. We represent this recovery event by a state transition with a constant transition rate. For example, when a partial replica of process a, viz, a', takes control, the event is represented by $a' \rightarrow a$ with a recovery rate of $\mu_{aa'}$. This recovery event is instantaneous for full replication since the information in a full replica's data structure is always up-to-date. The reliability of the partially replicated system is $1 - \Pr_f\{t\}$, for the initial state $= (a,b,a'b',a'b')$. $\Pr_f\{t\}$ can be obtained numerically by: 1) solving a set of differential equations describing the behavior of the Markov model in figure 4 using the Gear backward method, or 2) by using its Matrix-Geometric solution [8]:

$$\Pr_f\{t\} = \underline{\alpha} \cdot \exp(At)\underline{1}$$

$$\exp(At) = \sum_{j=0}^{\infty} A^j \cdot t^j / j!.$$

Notation

- A — $m \times m$ matrix for the Markov process with negative diagonal elements and non-negative off-diagonal elements,
- $m+1$ — a state corresponding to state f
- $\underline{\alpha}$ — $1 \times m$ initial probability vector: $(1,0,\ldots,0)$
- $\underline{1}$ — $m \times 1$ vector of 1's □

Then, even without having access to a procedure for numerically solving differential equations, $\Pr_f\{t\}$ can be computed by using an appropriate summation truncating rule to approximate $\exp(At)$ as closely as feasible.

Section 4 compares the resulting system reliability due to the use of partial replication and full replication for nonrepairable systems.

3.2 *Repairable Systems with Finite Repair-Rate*

Assumption

1. Once a processor fails, it takes an exponentially distributed time with rate δ to be repaired. □

The repair event is represented by adding a newly repaired processor, ϕ, into the state description. For example, $(a, b, b) \rightarrow (a, \phi, b, b)$ represents a transition in which p_2 (with a full replica of process a allocated to it) is just newly repaired and ready to be put on line again. Once a processor is repaired, it takes another exponentially distributed time (to load the process modules and to acquire process information) to become operational. For example, $(a, \phi, b, b) \rightarrow (a, a, b, b)$ with a transition rate $\mu_{a\phi}$ represents a transition in which a full replica of a has been recovered on a newly repaired processor ϕ. The average time that is required to perform such a transition is $1/\mu_{a\phi}$. This and other transition rates are defined in section 2.

Table 1
Full Replication Markov Chain
[Finite Repair-Rate]

State	Label	Transition
(a, a, b, b)	1	$(5, 2\lambda)$, $(6, 2\lambda)$
(a, a, b, ϕ)	2	$(5, \lambda)$, $(7, 2\lambda)$, $(10, \lambda)$, $(1, \mu_{b\phi})$
(a, ϕ, b, b)	3	$(6, \lambda)$, $(8, 2\lambda)$, $(10, \lambda)$, $(1, \mu_{a\phi})$
(a, ϕ, b, ϕ)	4	$(7, \lambda)$, $(8, \lambda)$, $(10, 2\lambda)$, $(2, \mu_{a\phi})$, $(3, \mu_{b\phi})$
(a, a, b)	5	$(9, 2\lambda)$, $(10, \lambda)$, $(2, \delta)$
(a, b, b)	6	$(9, 2\lambda)$, $(10, \lambda)$, $(3, \delta)$
(a, b, ϕ)	7	$(9, \lambda)$, $(10, 2\lambda)$, $(4, \delta)$, $(6, \mu_{b\phi})$
(a, ϕ, b)	8	$(9, \lambda)$, $(10, 2\lambda)$, $(4, \delta)$, $(5, \mu_{a\phi})$
(a, b)	9	$(10, 2\lambda)$, $(7, \delta)$, $(8, \delta)$
f	10	none

Table 2
Partial Replication Markov Chain
[Finite Repair-Rate]

State	Label	Transition
$(a, b, a'b', a'b')$	1	$(13, 2\lambda)$, $(17, \lambda)$, $(18, \lambda)$
$(a, b, a'b', \phi)$	2	$(12, \lambda)$, $(14, \lambda)$, $(15, \lambda)$, $(16, \lambda)$, $(1, \mu_{a'b'\phi})$
$(a, b', a'b', \phi)$	3	$(15, 2\lambda)$, $(17, \lambda)$, $(19, \lambda)$, $(11, \phi_{a'b'\phi})$, $(2, \mu_{bb'})$, $(25, \theta)$
$(a', b, a'b', \phi)$	4	$(16, 2\lambda)$, $(18, \lambda)$, $(19, \lambda)$, $(12, \mu_{a'b'\phi})$, $(2, \mu_{aa'})$, $(25, \theta)$
(a, b, ϕ, ϕ)	5	$(14, 2\lambda)$, $(2, 2\mu_{a'b'\phi})$, $(25, 2\lambda)$
(a, b', ϕ, ϕ)	6	$(15, 2\lambda)$, $(3, 2\mu_{a'b'\phi})$, $(5, \mu_{bb'})$, $(25, 2\lambda + \theta)$
(a', b, ϕ, ϕ)	7	$(16, 2\lambda)$, $(4, 2\mu_{a'b'\phi})$, $(5, \mu_{aa'})$, $(25, 2\lambda + \theta)$
(a', b', ϕ, ϕ)	8	$(19, 2\lambda)$, $(9, 2\mu_{a'b'\phi})$, $(7, \mu_{bb'})$, $(6, \mu_{aa'})$, $(25, 2\lambda + 2\theta)$
$(a', b', a'b', \phi)$	9	$(19, 3\lambda)$, $(20, \lambda)$, $(10, \mu_{a'b'\phi})$, $(4, \mu_{bb'})$, $(3, \mu_{aa'})$, $(25, 2\theta)$
$(a', b', a'b', a'b')$	10	$(20, 4\lambda)$, $(12, \mu_{bb'})$, $(11, \mu_{aa'})$, $(25, 2\theta)$
$(a, b', a'b', a'b')$	11	$(17, 3\lambda)$, $(20, \lambda)$, $(1, \mu_{bb'})$, $(25, \theta)$
$(a', b, a'b', a'b')$	12	$(18, 3\lambda)$, $(20, \lambda)$, $(1, \mu_{aa'})$, $(25, \theta)$
$(a, b, a'b')$	13	$(21, \lambda)$, $(22, \lambda)$, $(24, \lambda)$, $(2, \delta)$
(a, b, ϕ)	14	$(24, \lambda)$, $(13, \mu_{a'b'\phi})$, $(5, \delta)$, $(25, 2\lambda)$
(a, b', ϕ)	15	$(21, \lambda)$, $(17, \mu_{a'b'\phi})$, $(14, \mu_{bb'})$, $(6, \delta)$, $(25, 2\lambda + \theta)$
(a', b, ϕ)	16	$(22, \lambda)$, $(18, \mu_{a'b'\phi})$, $(14, \mu_{aa'})$, $(7, \delta)$, $(25, 2\lambda + \theta)$
$(a, b', a'b')$	17	$(21, 2\lambda)$, $(23, \lambda)$, $(13, \mu_{bb'})$, $(3, \delta)$, $(25, \theta)$
$(a', b, a'b')$	18	$(22, 2\lambda)$, $(23, \lambda)$, $(13, \mu_{aa'})$, $(4, \delta)$, $(25, \theta)$
(a', b', ϕ)	19	$(23, \lambda)$, $(20, \mu_{a'b'\phi})$, $(16, \mu_{bb'})$, $(15, \mu_{aa'})$, $(8, \delta)$, $(25, 2\lambda + 2\theta)$
$(a', b', a'b')$	20	$(23, 3\lambda)$, $(18, \mu_{bb'})$, $(17, \mu_{aa'})$, $(9, \delta)$, $(25, 2\theta)$
(a, b')	21	$(24, \mu_{bb'})$, $(15, \delta)$, $(25, 2\lambda + \theta)$
(a', b)	22	$(24, \mu_{aa'})$, $(16, \delta)$, $(25, 2\lambda + \theta)$
(a', b')	23	$(22, \mu_{bb'})$, $(21, \mu_{aa'})$, $(19, \delta)$, $(25, 2\lambda + 2\theta)$
(a, b)	24	$(14, \delta)$, $(25, 2\lambda)$
f	25	none

We can model the finite repair-rate by using 2 Markov chains, as shown in tables 1 & 2 which correspond to fully replicated and partially replicated systems, respectively. A state is labeled with a number. For example, table 1 has 10 states, numbered from 1 to 10. Under the *Transition* column, a "(state, state transition rate)" denotes possible outward state transitions from a given state. For example, (a, a, b, b) has 2 possible state transitions, leading to states (a, a, b) & (a, b, b), with a transition rate of $2 \cdot \lambda$ each. Hence, state 1 (a,a,b,b) has 2 possible outward state transitions: $(5, 2 \cdot \lambda)$ & $(6, 2 \cdot \lambda)$. The reliability of a fully replicated and a partially replicated system under finite repair-rate can be computed numerically from tables 1 & 2, respectively, again using the Gear backward method.

3.3 *Repairable Systems with Instantaneous Repair*

A system with instantaneous repair can be viewed as having an infinite backup processor pool. When a processor fails, it is repaired instantaneously (or in the case of an infinite backup processor pool, it is instantaneously powered-up) and ready to be put on line. Hence, unlike the previous two systems, the failure of a processor cannot be simply represented by removing the processes allocated to the failed processor in the state description, eg, $(a, a, b, b) \to (a, b, b)$ when p_1 or p_2 fails. Instead, the failure event is denoted by $(a, a, b, b) \to (a, \phi, b, b)$ since the failed processor, either p_1 or p_2, is repaired instantaneously. The transition rate of the above event, however, is still $2 \cdot \lambda$ since the repair is instantaneous. The Markov chain shown in figure 5 models the fully replicated system with instantaneous repair, with state (a, a, b, b) representing the initial state, while the Markov chain shown in table 3 models the partially replicated system with $(a, b, a'b', a'b')$ representing the initial state.

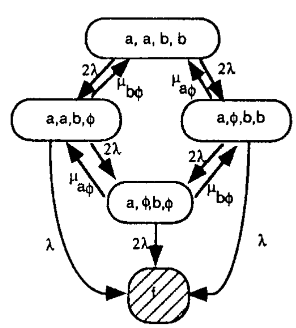

Figure 5. Full Replication Model [Instantaneous Repair]

Table 3
Partial Replication Markov Chain Model
[Instantaneous Repair]

State	Label	Transition
$(a, b, a'b', a'b')$	1	$(2, 2\lambda)$, $(3, \lambda)$, $(4, \lambda)$
$(a, b, a'b', \phi)$	2	$(1, \mu_{a'b'\phi})$, $(5, \lambda)$, $(6, \lambda)$, $(7, \lambda)$
$(a, b', a'b', \phi)$	3	$(6, 2\lambda)$, $(8, \lambda)$, $(11, \mu_{a'b'\phi})$, $(2, \mu_{bb'})$, $(13, \theta)$
$(a', b, a'b', \phi)$	4	$(7, 2\lambda)$, $(8, \lambda)$, $(12, \mu_{a'b'\phi})$, $(2, \mu_{aa'})$, $(13, \theta)$
(a, b, ϕ, ϕ)	5	$(2, 2\mu_{a'b'\phi})$, $(13, 2\lambda)$
(a, b', ϕ, ϕ)	6	$(3, 2\mu_{a'b'\phi})$, $(5, \mu_{bb'})$, $(13, 2\lambda + \theta)$
(a', b, ϕ, ϕ)	7	$(4, 2\mu_{a'b'\phi})$, $(5, \mu_{aa'})$, $(13, 2\lambda + \theta)$
(a', b', ϕ, ϕ)	8	$(9, 2\mu_{a'b'\phi})$, $(7, \mu_{bb'})$, $(6, \mu_{aa'})$, $(13, 2\lambda + 2\theta)$
$(a', b', a'b', \phi)$	9	$(8, 3\lambda)$, $(10, \mu_{a'b'\phi})$, $(4, \mu_{bb'})$, $(3, \mu_{aa'})$, $(13, 2\theta)$
$(a', b', a'b', a'b')$	10	$(9, 4\lambda)$, $(12, \mu_{bb'})$, $(11, \mu_{aa'})$, $(12, 2\theta)$
$(a, b', a'b', a'b')$	11	$(3, 3\lambda)$, $(9, \lambda)$, $(1, \mu_{bb'})$, $(13, \theta)$
$(a', b, a'b', a'b')$	12	$(4, 3\lambda)$, $(9, \lambda)$, $(1, \mu_{aa'})$, $(13, \theta)$
f	13	none

4. RELIABILITY COMPARISON OF FULLY & PARTIALLY REPLICATED SYSTEMS

System reliability of fully and partially replicated systems (as modeled in section 3) are compared in this section.

1. We show for each case that there is a set of parameter values under which a partially replicated system has a better reliability than that of a fully replicated system.
2. We analyze the effect of parameters on system reliability and their relationship with each other.
3. We conclude that among the three cases, partial replication is most favorable when the system is nonrepairable, ie, when hardware resources are scarce and there is no repair capability in the system. □

Processes *a* & *b* both consume the full processing power of a processor.

Assumptions

1. $\mu = \mu_{aa'} = \mu_{bb'}$.
2. The average time required to recover a full replica from a newly repaired processor is about twice as much as that required to recover a full replica from a partial replica. This is reasonable because the replication level is 0.5 for a partial replica. That is, we let $\mu_{a\phi} = \mu_{b\phi} = \mu_{a'b'\phi} = \mu/2$. λ is selected to go from $2.77 \cdot 10^{-4}$, 1 hour per failure, to $2.77 \cdot 10^{-5}$, 10 hours per failure. This range shows meaningful differences in reliability between the partially replicated and fully replicated systems. □

We adopt the following two strategies to analyze the effect of parameters on system reliability:

1. All parameters vary proportionately.
2. Only one parameter varies while all others are kept constant.

4.1 *Analysis of Strategy #1*

Strategy #1 is used to see how parameters are interrelated when they vary proportionately. For that purpose, the ratio of the processor failure rate (λ) to the processor repair rate (δ) or the replica recovery rate (μ), is 0.1, while the ratio of the processor failure rate to the software failure rate (θ) varies from 0.1 to 1, ie, $10 \cdot \lambda = x \cdot \theta = \mu = \delta$, with x ranging from 1 to 10. The intent of this strategy is to reveal the cross-over points at which one replication technique provides a better reliability than the other.

Figures 6 - 8 show the difference in reliability between the partially replicated and fully replicated systems, $r_{\text{partial}}(t) - r_{\text{full}}(t)$, for nonrepairable, finite repair-rate, and instantaneously repairable systems, respectively. When $10 \cdot \lambda = \theta = \mu = \delta$ $(x=1)$, all three cases show that the reliability of the fully replicated system is better than that of the partially replicated system for all λ values under consideration. The advantage of full replication over partial replication disappears, however, as θ becomes comparable in magnitude to λ, ie, for $x \geq 5$. In this case, the partially replicated systems provide better reliability than the fully replicated systems as λ increases. The 2 reasons for this are:

1. State transitions that could lead to system failure in partial replication are mostly due to θ rather than λ. This is in contrast to full replication where system failure is exclusively due to hardware failures.
2. The number of states from which state transitions could lead to system failure in partial replication is less than that of full replication. Consequently, the reliability of the partially replicated systems will decline by a lesser extent than that of the fully replicated systems as λ increases, since increasing λ only increases θ by the same order of magnitude. Conversely, when θ is an order of magnitude higher than λ (eg, $10 \cdot \lambda = \theta = \mu = \delta$), the partially replicated systems suffer more from increasing λ. The reason is that this increases θ by an extra order of magnitude (ie, 10 times) and the probability of state

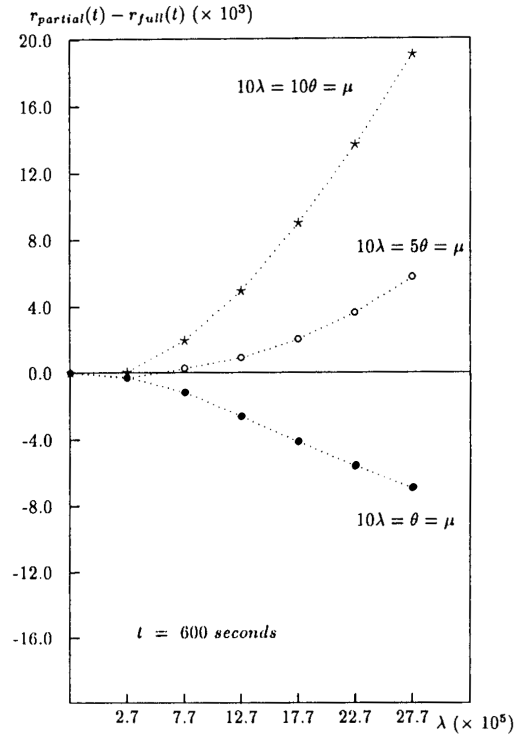

Figure 6. Difference in Reliability [No Repair]

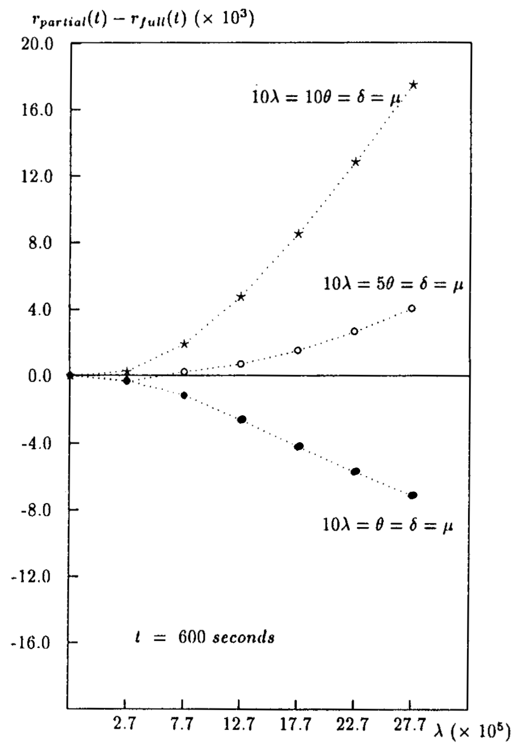

Figure 7. Difference in Reliability [Finite Repair-Rate]

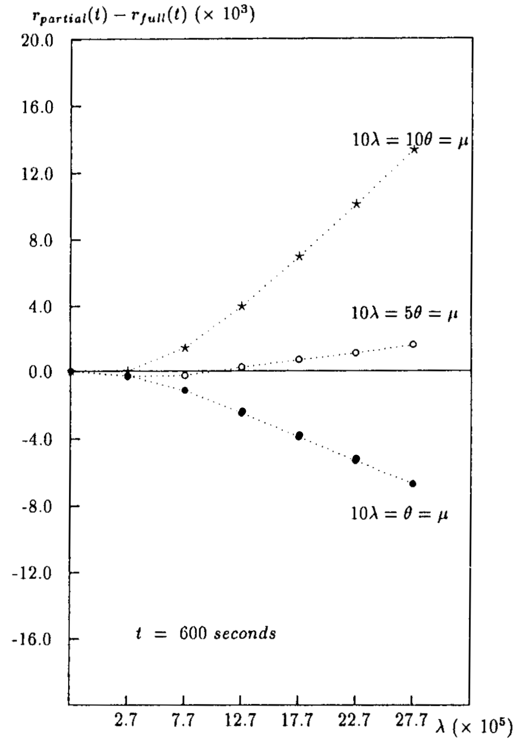

Figure 8. Difference in Reliability [Instantaneous Repair]

transitions that can lead to system failure for partially replicated systems is greatly increased. These arguments explain why in figures 6 - 8 when $10 \cdot \lambda = \theta = \mu$, the $r_{\text{partial}}(t) - r_{\text{full}}(t)$ decreases as λ increases, while when $10 \cdot \lambda = 5 \cdot \theta = \mu$, the $r_{\text{partial}}(t) - r_{\text{full}}(t)$ increases as λ increases. In these cases, λ and all other parameters increase proportionately. Below we apply strategy #2 to see the absolute effect of λ. □

Figure 9 demonstrates that when θ is comparable in magnitude to λ, the benefit of partial replication over full replication is manifested most when there is no repair in the system.

4.2 *Analysis of Strategy #2*

The effect of θ, of course, is not surprising since θ relates to partial replication only. What is surprising, however, is the effect of the recovery rate (μ) on system reliability. Recall that the recovery rate is the rate at which the recovery of a replica can be performed on an on-line processor. Since the replication level for a partial replica is 0.5, the time required to recover a full replica from a newly repaired processor is about twice as much as that required for recovering a full replica from a partial replica. Hence, we have set $\mu_{bb'} = \mu_{aa'} = \mu$, and $\mu_{a\phi} = \mu_{b\phi} = \mu_{a'b'\phi} = \mu/2$. With this provision, it seems that the recovery rate should have equal effects on both the fully and partially replicated systems because their recovery rates are proportionally comparable. This is, however, not the case.

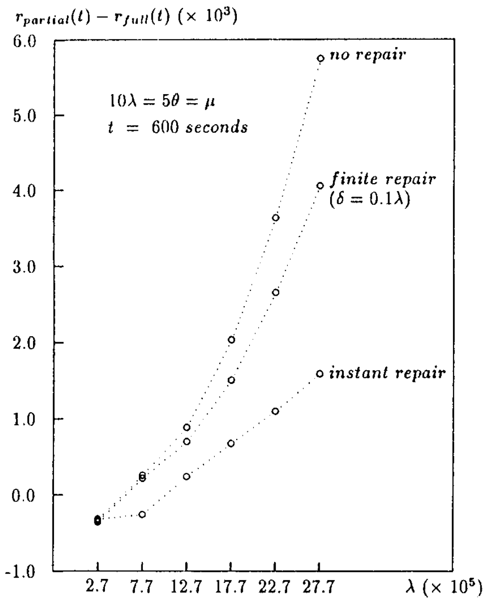

Figure 9. Difference in Reliability [A Comparison]

In the partial replication scheme, all states other than the initial state $(a, b, a'b', a'b')$, the final states f, state $(a, b, a'b')$, and state (a, b) $(25-4=21$ states) can attempt to perform $a' \rightarrow a$ or $b' \rightarrow b$ transitions. In the full replication scheme, only 5 out of 10 states can perform $\phi \rightarrow a$ or $\phi \rightarrow b$ transitions. For example, state (a, a, b) cannot recover to state (a, a, b, b) without first going to state (a, a, b, ϕ), ie, the failed processor must be repaired first before recovery can take place. In partial replication, however, state $(a, b', a'b')$, for example, can recover to state $(a, b, a'b')$ without having to wait for the failed processor to be repaired, ie, there is no need to go first to state $(a, b', a'b', \phi)$ in order to perform replica recovery.

While the above rationale suggests that partial replication is more favorable when the value of μ is large, it also implies that when the repair rate becomes larger, the advantage of partial replication over full replication regarding μ is less important since there are no *wait* states in either replication scheme. This is indeed the case. Figure 10 shows that in the system with instantaneous repair $(\delta=\infty)$, the effect of μ is least important among all 3 cases. The advantage of partial replication over full replication *vis-a-vis* μ, on the other hand, increasingly manifests itself as the repair rate decreases and is at the maximum point when $\delta=0$ (no repair).

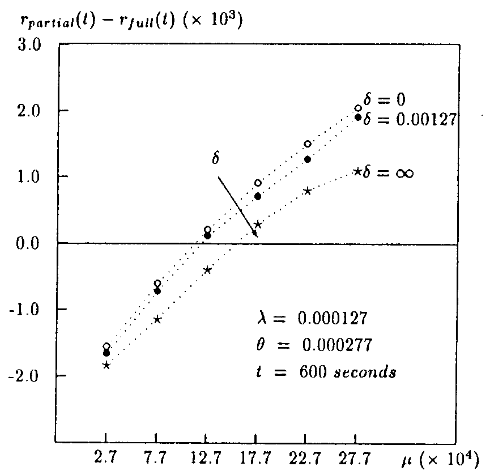

Figure 10. Difference in Reliability As a Function of μ & δ

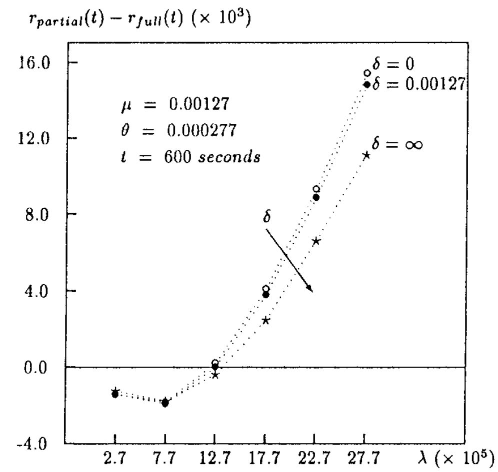

Figure 11. Difference in Reliability As a Function of λ & δ

Figure 10 shows that in all three cases, the reliability of the partially replicated system can benefit more from an increase of the recovery rate than the fully replicated system. In fact, the reliability of the partially replicated system becomes better than that of the fully replicated system when μ exceeds a threshold value. One explanation of this phenomenon is that the number of states from which a replica can recover (due to μ) in partial replication is much larger than that of full replication. For example, for finite repair-rate systems, it is 21 *vs* 5 after excluding the initial state, the failure state, and states from which no replica recovery (no transitions due to μ) is possible.

With the above result, which shows that an increase in μ favors partial replication, especially when δ is low, it is conceivable that if we fix μ and increase δ, then the fully replicated system will benefit more from the increase in δ than the partially replicated system (since the advantage of partial replication due to μ becomes less important when δ is high). This is indeed the case and is illustrated in figure 11. Hence, the effects of μ and δ are inversely related. Since typically a system is not likely to have instantaneous repair, the effects of μ and λ appear to favor the adoption of partial replication over full replication. Furthermore, the advantage of partial replication over full replication is augmented when the underlying hardware is unreliable. Figure 11 summarizes what we have described so far. It shows that partial replication is most favorable in systems with no repair capability, viz, when $\delta = 0$, and this advantage is manifested most when the underlying hardware is unreliable.

We thus conclude that partial replication is most favorable when θ is of the same order of magnitude (within a factor of 10) as λ and there is little or no repair capability in the system. This favorable situation is most likely when the underlying hardware is unreliable and the software recovery rate (μ) is high.

ACKNOWLEDGMENT

We are pleased to thank the three anonymous referees for their many useful criticisms and suggestions. Their detailed comments have greatly improved the quality of the paper. This work was supported in part by the US National Science Foundation under Grant CCR-9110816.

REFERENCES

[1] S.M. Ross, *Introduction to Probability Models*, 4th Edition, 1989; Academic Press.

[2] W. N. Toy, "Fault-tolerant design of local ESS processors", *Proc. IEEE*, vol 66, 1978 Oct, pp 1126-1145.

[3] A. L. Hopkins, T. B. Smith III, J. H. Lala, "FTMP - A highly reliable fault-tolerant multiprocessor for aircraft", *Proc. IEEE*, vol 66, 1978 Oct, pp 1221-1239.

[4] F. B. Bastani, C. V. Ramamoorthy, "Distributed fault-tolerant process-control programs", *FJCC* '87, Dallas, 1987 Oct.

[5] I.-R. Chen, F. B. Bastani, "Telescopic replication in hierarchically structured process-control programs", *Proc. 22nd Hawaii Int'l Conf. System Sciences*, 1989 Jan, pp 1055-1063.

[6] I.-R. Chen, *An AI-based architecture of self-stabilizing fault tolerant distributed process-control programs*, PhD Thesis, 1988 Dec; Dept. Computer Science, University of Houston.

[7] IMSL Inc, *MATH/Library - Fortran Subroutines for Mathematical Applications*, 1987, pp 640-652.

[8] M. Neuts, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*, 1981; John Hopkins University Press.

[9] L. Kleinrock, *Queueing Systems*, vol II, 1976, pp 56-79; Wiley-Interscience.

AUTHORS

Dr. Ing-Ray Chen; Dept. of Computer & Information Science; Weir Hall 302; University of Mississippi, Mississippi 38677 USA.

Ing-Ray Chen (M'90): For biography, see IEEE Trans. Reliability, vol 40 1991 Aug, p 369.

Dr. Farokh B. Bastani; Dept. of Computer Science; University of Houston; Houston, Texas 77004 USA.

Farokh B. Bastani (M'82): For biography, see IEEE Trans. Reliability, vol 39, 1990 Jun, p 216.

Manuscript TR89-169 received 1989 October 14; revised 1991 February 15; revised 1991 August 21.

IEEE Log Number 08260

◄TR►

FROM THE EDITORS FROM THE EDITORS FROM THE EDITORS FROM THE EDITORS FROM THE EDITORS FROM THE EDITORS

Opinion Items

Do you want to sound off on a reliability topic?

Then DO IT!

Write a "Letter to the Editor".

The Editorial staff welcomes your comments on material published in this Transactions (eg, technical papers, editorials, and book reviews) as well as on topics of interest to Transactions readers. If the correspondence concerns technical matters it will be sent to the original authors and/or other capable people for their comments. The correspondent need not make any changes because of those "referee" comments, but is permitted to do so. Opinion items are labeled as such.

Comments on technical errors, with a reply by the original author, will usually be published as a short note. Interaction between the correspondent and the original author is encouraged in order to reduce the area of disagreement as much as feasible.

Material which is judged not suitable as a referred paper can sometimes be abbreviated and published as an opinion item.

Opinions will be edited as to length and some rather broad limits of propriety — at the sole discretion of the Editorial staff.

◄TR►