TMR 실시간 제어 시스템의 내고장성 기법 및 신뢰도 해석

Reliability Analysis and Fault Tolerance Strategy of TMR Real-time Control Systems

곽 성 우*, 유 관 호
(Seong Woo, Kwak and Kwan Ho, You)

Abstract : In this paper, we propose the Triple Modular Redundancy (TMR) control system equipped with a checkpoint strategy. In this system, faults in a single processor are masked and faults in two or more processors are detected at each checkpoint time. When faults are detected, the rollback recovery is activated to recover from faults. The conventional TMR control system cannot overcome faults in two or more processors. The proposed system can effectively cope with correlated and independent faults in two or more processors. We develop a reliability model for this TMR control system under correlated and independent transient faults, and derive the reliability equation. Then we investigate the number of checkpoints that maximizes the reliability.

Keywords : reliability analysis, triple modular redundancy, rollback recovery, fault tolerance, real-time control system

I. Introduction

In recent years, control of systems using digital computers has been drastically increased due to the availability of inexpensive, powerful computers and the increasing needs to control more sophisticated processes. Therefore when computers are involved in controlling dangerous and life-critical systems such as nuclear reactors or aircraft, failure or malfunction of computers may lead to enormous disaster. These life-critical applications often require control in real time, that is, control actions have tight timing requirements(Control System Deadline : CSD), and violation of timing requirements invalidates the usefulness of the control action. Thus computers(or processors) deployed in these life-critical control applications require stringent reliability specifications. It is usually met by imposing fault tolerance to controller computers.

Among faults that cause some control system failures, transient faults are becoming more important in recent years. More than 90% of field failures are reported as being caused by transient faults[1]. Transient faults occurring independently in each processor are called independent faults. They are usually caused by internal factors. In contrast, transient faults affecting several processors simultaneously are called correlated faults. Correlated faults are caused mostly by external factors such as EMI. These faults are especially important in the area of aerospace, which is characterized by an environment containing significant electromagnetic and elementary particle radiation. For example, airborne computers can be disrupted by lightning strikes. Transient faults can be handled by hardware, time, or information redundancies. Triple Modular Redundancy(TMR) is one of the most popular hardware fault-tolerance methods[2]-[5]. Errors generated by any single faulty module are masked out through a simple voter. The TMR strategy is useful for tolerating independent faults in one processor, but it suffers from independent faults in two or more processors and correlated faults. Checkpoint scheme using time redundancy is also a fault tolerance technique commonly used for transient faults[6]-[12]. In this scheme, the intermediate states of a task are saved periodically in a secure device at each checkpoint time. If an error is detected, the saved states are restored and the task re-executed from the checkpoint(called rollback). Thus, checkpoint can greatly reduce the probability of incorrect outputs due to transient faults, and so make the control task to be more reliable.

In this paper, we propose a TMR system called as the TMTR(Triple Modular Temporal Redundancy) system, which is equipped with a checkpoint scheme. The TMTR system can cope with both correlated faults and independent faults in any processors. We develop a reliability model for this TMTR system under independent and correlated faults, and then analyze the reliability. The mission-time reliability of the TMTR control system is derived. Finally, we find the optimal number of checkpoints that maximizes the reliability. This paper is organized as follows. In Section 2, the TMTR system is discussed, and reliability of this system is analyzed in Section 3. Numerical results are provided in Section 4. We conclude with Section 5.

II. TMTR Control System

As a real-time control system executes the same task periodically at each sampling time, the operation in one sampling interval is repeated in other sampling intervals. Also, a real-time controlled plant has a deadline within which the periodic task must be finished. Missing a deadline leads to a system failure such as crash in airplane(dynamic failure[2]).

Real-time control systems are often used in harsh environments, and subject to many transient faults while in operation. Checkpoint enables a reduction in the recovery time from a transient fault by saving intermediate states of a task in a reliable storage facility, and then, on detection of a fault,

* 책임저자(Corresponding Author)
논문접수 : 2004. 3. 19., 채택확정 : 2004. 7. 20.
곽성우 : 계명대학교 전자공학과(ksw@kmu.ac.kr)
유관호 : 성균관대학교 정보통신공학부(khyou@skku.ac.kr)
※ 본 연구는 2003 학년도 계명대학교 비사연구기금으로 이루어졌음.

restoring from a previously stored state. Whereas inserting more checkpoints and reducing the interval between them reduces the re-processing time after faults, checkpoints have associated costs, and inserting extra checkpoints increases the overall cost and the task execution time. Thus, a trade-off between the re-processing time and the checkpoint overhead leads to an optimal checkpoint placement strategy that optimizes certain performance measures.

1. Basic Assumptions

Consider a control system that is characterized by a periodic control task with the following assumptions.

A.1 : Checkpoints can be inserted anywhere in the periodic task.

A.2 : Faults within a checkpoint interval are detected at each checkpoint time.

A.3 : The deadline of a control task is equal to its sampling period.

A.4 : Transient faults arrive as a Poisson process with rate λ and disappear with rate μ.

A.5 : Correlated faults affect all the processors simultaneously.

A.6 : Faults always cause errors.

In practical, it may be difficult to insert checkpoints anywhere in the task, that is, difficult to maintain checkpoint intervals equal. However, analysis under the assumption of equal checkpoint interval (Assumption A.1) can give insight on how checkpoints should be inserted for best reliability. A.2 means that any faults can be detected through voting in the TMR system. A.3 represents that we consider a strict control system. A.4 is assumed for simple analysis. A.5 and A.6 are conservative assumptions: Faults may not cause errors and correlated faults may affect only one or two processors.

2. TMTR System Structure & Operation

The TMTR system utilizes both the space and the temporal redundancy. By using the TMR(the space redundancy), faults in a single processor can be masked and faults in multiple processors can be detected through 2 out-of 3 voting. Also, the checkpoint strategy (the temporal redundancy) is used to recover from correlated faults. Fig. 1 shows the basic hardware configuration of this system, and Fig. 2 is the time sequence of operation. T means a sampling period, which is equal to a deadline, and Δ is the checkpoint interval. Three processors send voting data to a voter(see Figs. 1 and 2) at each checkpoint time. The voter gives two outputs; voting information and the majority. Voting information contains that there are no errors on three processors, or there is one faulty processor, or there is no majority data(voting failure has occurred). The majority is the voted results.

Each processor gets the voting information from the voter and determines whether to rollback or to save the majority data into a secure memory. All processors rollback to the latest checkpoint when the voting information contains that voting failure has occurred. In case of successful voting(no error or only one processor error), each processor synchronizes its voted data with that of other processors by fetching the majority results, and saves the majority to its secure memory.

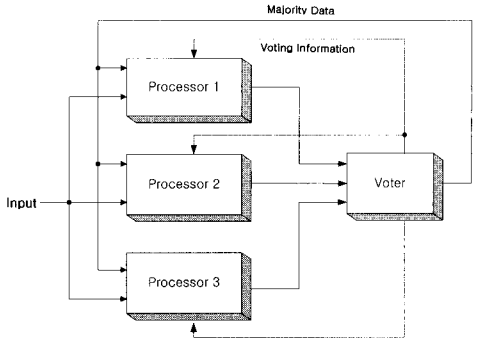

Fig. 1. Hardware configuration of the TMTR System.

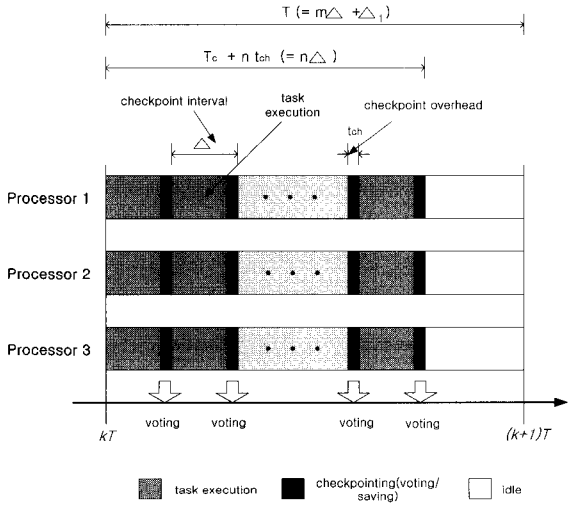

Fig. 2. Time operation of the TMTR system.

Therefore, errors in any single processor can be masked through voting, and errors in multiple processors can be recovered through rollback.

Let T_c be the required time to execute the periodic task. Then, if there are n checkpoints in T_c, the checkpoint interval Δ and the allowable maximum number of checkpoints m that can be inserted in one sampling period (deadline) are obtained as follows.

$$\Delta = \frac{T_c}{n} + t_{ch} \tag{1}$$

$$m = \left\lfloor \frac{T}{\Delta} \right\rfloor \tag{2}$$

where $\lfloor x \rfloor$ is the largest integer which does not exceed x, and t_{ch} is the checkpoint overhead. Also the remaining time (Δ_1) after m checkpoints in T can be derived as: $\Delta_1 = T - m \cdot \Delta \quad (0 \le \Delta_1 < \Delta)$. Note that owing to the checkpoint overhead, the execution time of a task is increased by $n \cdot t_{ch}$ after placing n checkpoints. Fig. 2 shows Δ, T and Δ_1. For the successful execution of a task, at least n

checkpoint intervals should be fault free among the m checkpoint intervals available in one T. In other words, the TMTR system can overcome faults in up to($m-n$) checkpoint intervals.

III. Reliability Analysis of TMTR Control System

1. Reliability Model for the TMR Control System

Consider a control system that is characterized by a single processor. This control system reads sensor values, computes control inputs using a suitable algorithm, and outputs the results to plants. These steps are repeated in each fixed sampling period, T. Thus, a mission of the control system consists of sampling periods. In an environment of transient faults, each sampling period of the control system can be represented by three states, '0'(up), '1'(down), and F, as shown in Fig. 3. Here, the state '0'(up) represents that there is no fault at the beginning time of a periodic task (fault free at time t=kT) and the periodic task is executed correctly and finished within the deadline, that is, the period from $(k-1)T$ to kT. The state

'1'(down) means that there are some faults at the beginning time of a periodic task(faulty at t=kT) while the periodic task is executed correctly and finished within the deadline. The state 'F' means that the control task is either executed incorrectly or not finished within the period from $(k-1)T$ to kT.

Since we assumed transient faults of exponential distributions with occurrence rate λ and recovery rate μ, and a strict real-time control system, the single processor real-time control system under transient faults can be modeled with the three-state discrete time Markov chain as shown in Fig. 4. This Markov chain evolves with sampling periods t=kT,(where k=1, 2...). From this Markov model, we can find the stochastic state of a control system after a mission time,

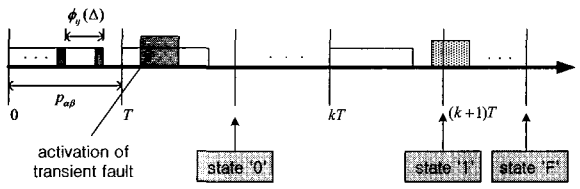

Fig. 3. State description of a single processor control system.

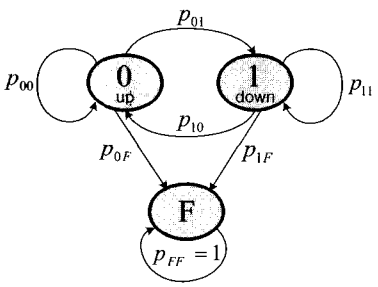

Fig. 4. Markov model of a single processor control system under transient faults.

and hence the control system reliability. The reliability of a control system over a period of operation(the mission) is the probability that its entire critical workload executes on time successfully over the period.

For a TMR system, we should consider two types of transient faults. They are independent faults and correlated faults. Independent faults affect each processor independently, whereas correlated faults affect three processors simultaneously. Because the model of a single processor control system under transient faults needs two states excluding the failed state as in Fig. 4, a TMR system model under independent and correlated faults needs sixteen states ($2^3 \times 2$) plus one failure state: 2^3 states (2 states for each processor) for independent faults, and 2 states for correlated faults.

However, if all three processors have the same hardware/software components and configurations, the number of states for independent faults can be reduced to four states, because all the processors have the same fault occurrence and recovery rates. The four states, denoted as '0', '1', '2', '3', represent the number of processors which are under independent faults at t=kT. Therefore, the TMR control system under independent and correlated transient faults can be modeled with a nine-state discrete-time Markov chain.

We denote each state of the TMR control system as (i, j) except the failed state 'F'. The first element 'i' (i $\in$ {'0' (up), '1' (down)}) represents the state of the TMR control system under correlated faults. The '1' (down) means that the TMR control system is affected by correlated faults at t=kT, and the '0'(up) means that the TMR control system is free from correlated faults at t=kT. The second element 'j' (j $\in$ {'0'(0-down), '1' (1-down), '2' (2-down), '3' (3-down)}) indicates state under independent faults: 'j' represents the number of processors affected by independent faults at t=kT. Fig. 5 shows the Markov model. In state (i,j), the periodic task is executed successfully within the deadline. The state 'F' represents that the TMR control system has failed due to missing a deadline. For example, the state (1,2) means that the TMR control system is affected by correlated faults, at t=kT (down), and also two of three processors by independent faults (2-down), while the periodic task is executed successfully

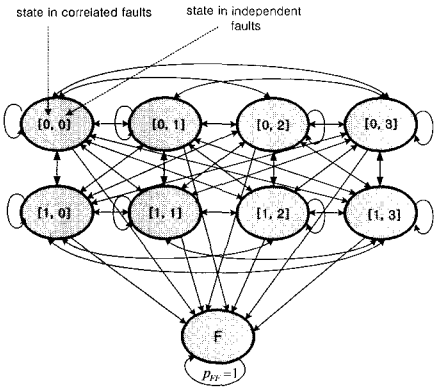

Fig. 5. Markov model of the TMR control system under correlated correlated and Independent transient faults.

within the deadline.

2. Reliability of the TMTR Control System

To find the reliability of the TMTR control system over a mission, we should find the stochastic states after the mission. The stochastic states can be derived with the knowledge of transition probabilities in the model of Fig. 5. Let $\phi_{ii'}(\Delta)$ (i, i' $\in$ {'0'(up), '1'(down)}) be the probability of staying in state i' at t=Δ given the initial state i at t=0 for a single processor system, then they can be obtained as follows [13].

$$\phi_{00}(\Delta) = \frac{\lambda}{\mu+\lambda}e^{-(\mu+\lambda)\Delta} + \frac{\mu}{\mu+\lambda} \tag{3}$$

$$\phi_{01}(\Delta) = -\frac{\lambda}{\mu+\lambda}e^{-(\mu+\lambda)\Delta} + \frac{\lambda}{\mu+\lambda} \tag{4}$$

$$\phi_{10}(\Delta) = -\frac{\mu}{\mu+\lambda}e^{-(\mu+\lambda)\Delta} + \frac{\mu}{\mu+\lambda} \tag{5}$$

$$\phi_{11}(\Delta) = \frac{\mu}{\mu+\lambda}e^{-(\mu+\lambda)\Delta} + \frac{\lambda}{\mu+\lambda} \tag{6}$$

Also $\phi_0(\Delta)$, which is defined as the probability that there are no faults within time interval [0, Δ], is obtained as follows.

$$\phi_0(\Delta) = e^{-\lambda\Delta} \tag{7}$$

Next, we define Δ-step transition probabilities of the TMTR system as follows.

- $S_{(i,j)(i',j')}$: Transition probability of a checkpoint interval Δ from state (i,j) into (i',j') without voting failure.
- $u_{(i,j)(i',j')}$: Transition probability of a checkpoint interval Δ from state (i,j) into (i',j') with voting failure.
- $h_{(i,j)(i',j')}$: Transition probability of an interval Δ from state (i,j) into (i',j').

Here, i (i'), j (j') represents states under correlated and independent faults respectively (i,i' $\in$ {'0'(up),'1'(down)} j,j' $\in$ {'0'(0- down),'1'(1-down),'2'(2-down),'3'(3-down)}).

Let S, U, and H be the matrices that have their elements of $S_{(i,j)(i',j')}$, $u_{(i,j)(i',j')}$, and $h_{(i,j)(i',j')}$ respectively. Then we can find the following relation.

$$U = H - S \tag{8}$$

Fortunately, elements of the S matrix, $S_{(i,j)(i',j')}$, have nonzero value only for (i,j), (i',j') $\in$ {(0,0), (0,1)}, and all zeros for other states because these states can not produce successful voting. The S matrix is shown in Fig. 6. Because $S_{(0,0)(0,0)}$ is the probability that no correlated faults exist within an interval Δ and independent faults at most one processor with both beginning and end state of an interval Δ of '0', it can be obtained as follows.

$$S_{(0,0)(0,0)} = {}^c\phi_0(\Delta)\cdot[{}^i\phi^3{}_0(\Delta) + 3\cdot({}^i\phi_{00}(\Delta) - {}^i\phi_0(\Delta))\ {}^i\phi^2{}_0(\Delta)] \tag{9}$$

Here, superscript c(i) means that parameters of correlated (independent) faults, that is, λ_c (λ_i), μ_c (μ_i) are used for the calculation of $\phi_{\cdot\cdot}(\cdot)$. $\phi_{00}(\Delta) - \phi_0(\Delta)$ is the probability that faults exist within an interval Δ with both beginning and end state of '0'.

Similarly, $S_{(0,0)(0,1)}$, $S_{(0,1)(0,0)}$ and $S_{(0,1)(0,1)}$ can be derived as follows.

$$S_{(0,0)(0,1)} = {}^c\phi_0(\Delta)\cdot 3\,{}^i\phi_{01}(\Delta)\cdot{}^i\phi^2{}_0(\Delta) \tag{10}$$

$$S_{(0,1)(0,0)} = {}^c\phi_0(\Delta)\cdot{}^i\phi_{10}(\Delta)\cdot{}^i\phi^2{}_0(\Delta) \tag{11}$$

$$S_{(0,1)(0,1)} = {}^c\phi_0(\Delta)\cdot{}^i\phi_{11}(\Delta)\cdot{}^i\phi^2{}_0(\Delta) \tag{12}$$

The H matrix is shown in Fig. 7. Note that the element of each transition probability is product of two transition probabilities, transition probabilities of independent faults (${}^i\hat{P}_{jj'}$) and correlated faults (${}^i\phi_{ij}(\Delta)$). Also each of the four divided sections in Fig. 7 has the same transition probabilities for independent faults. Thus the elements of H matrix, $h_{(i,j)(i',j')}$, can be easily derived as follows:

$$h_{(0,j)(0,j')} = {}^c\phi_{00}(\Delta)\cdot{}^i\hat{P}_{jj'} \tag{13}$$

$$h_{(0,j)(1,j')} = {}^c\phi_{01}(\Delta)\cdot{}^i\hat{P}_{jj'} \tag{14}$$

$$h_{(1,j)(0,j')} = {}^c\phi_{10}(\Delta)\cdot{}^i\hat{P}_{jj'} \tag{15}$$

$$h_{(1,j)(1,j')} = {}^c\phi_{11}(\Delta)\cdot{}^i\hat{P}_{jj'} \tag{16}$$

Transition probabilities due to independent faults ${}^i\hat{P}_{jj'}$ are in Table 1. The U matrix can be obtained from Eq. 8.

We define W(i) as a transition probability matrix that there is an unsuccessful transition (voting failure at checkpoint time) for consecutive (i-1) Δ -steps and the successful transition at i- th Δ -step, then it is:

$$W(i) = U^{i-1}S \tag{17}$$

Let Q be the transition probability matrix of one sampling period where the periodic task completes its execution successfully, then Q can be obtained as follows.

$$Q = \sum_{i1=1}^{i1f}\sum_{i2=1}^{i2f}\cdots\sum_{in=1}^{\inf} W(i1)W(i2)\cdots W(in)H_{\Delta 1} \tag{18}$$

where

$$i_{1f} = m - n + 1$$

$$i_{jf} = m - n + j - \sum_{k=1}^{j-1} i_k, \qquad j \geq 2$$

$$\Delta_1 = T - \Delta\cdot\sum_{k=1}^{n} i_k$$

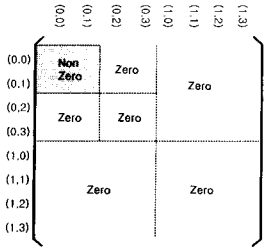

Fig. 6. S Matrix.

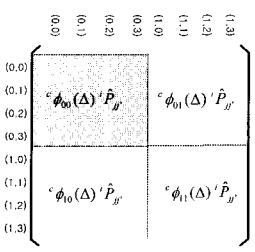

Fig. 7. H Matrix.

Table. 1. Transition Probabilities ${}^{i}\hat{P}_{jj'}$.

Transition Probabilities	Transition Probabilities
${}^{i}\hat{P}_{00} = {}^{i}\phi^{3}_{00}(\Delta)$	${}^{i}\hat{P}_{10} = {}^{i}\phi_{10}(\Delta)\,{}^{i}\phi^{2}_{00}(\Delta)$
${}^{i}\hat{P}_{01} = 3\,{}^{i}\phi^{2}_{00}(\Delta)\,{}^{i}\phi_{01}(\Delta)$	${}^{i}\hat{P}_{11} = {}^{i}\phi_{11}(\Delta)\,{}^{i}\phi^{2}_{00}(\Delta) + 2\,{}^{i}\phi_{10}(\Delta)\,{}^{i}\phi_{01}(\Delta)\,{}^{i}\phi_{00}(\Delta)$
${}^{i}\hat{P}_{02} = 3\,{}^{i}\phi_{00}(\Delta)\,{}^{i}\phi^{2}_{01}(\Delta)$	${}^{i}\hat{P}_{12} = 2\,{}^{i}\phi_{11}(\Delta)\,{}^{i}\phi_{01}(\Delta)\,{}^{i}\phi_{00}(\Delta) + {}^{i}\phi_{10}(\Delta)\,{}^{i}\phi^{2}_{01}(\Delta)$
${}^{i}\hat{P}_{03} = {}^{i}\phi^{3}_{01}(\Delta)$	${}^{i}\hat{P}_{13} = {}^{i}\phi_{11}(\Delta)\,{}^{i}\phi^{2}_{01}(\Delta)$
${}^{i}\hat{P}_{30} = {}^{i}\phi^{3}_{10}(\Delta)$	${}^{i}\hat{P}_{20} = {}^{i}\phi^{2}_{10}(\Delta)\,{}^{i}\phi_{00}(\Delta)$
${}^{i}\hat{P}_{31} = 3\,{}^{i}\phi_{11}(\Delta)\,{}^{i}\phi^{2}_{10}(\Delta)$	${}^{i}\hat{P}_{21} = {}^{i}\phi_{01}(\Delta)\,{}^{i}\phi^{2}_{10}(\Delta) + 2\,{}^{i}\phi_{11}(\Delta)\,{}^{i}\phi_{10}(\Delta)\,{}^{i}\phi_{00}(\Delta)$
${}^{i}\hat{P}_{32} = 3\,{}^{i}\phi_{10}(\Delta)\,{}^{i}\phi^{2}_{11}(\Delta)$	${}^{i}\hat{P}_{22} = 2\,{}^{i}\phi_{11}(\Delta)\,{}^{i}\phi_{10}(\Delta)\,{}^{i}\phi_{01}(\Delta) + {}^{i}\phi_{00}(\Delta)\,{}^{i}\phi^{2}_{11}(\Delta)$
${}^{i}\hat{P}_{33} = {}^{i}\phi^{3}_{11}(\Delta)$	${}^{i}\hat{P}_{23} = {}^{i}\phi_{01}(\Delta)\,{}^{i}\phi^{2}_{11}(\Delta)$

Here $H_{\Delta 1}$ is the transition probability matrix of an interval Δ_1, which can be obtained from H by substituting Δ with Δ_1.

Let $\pi_{(i,j)}(t)$ be the probability of staying in state (i,j) at time t. We define the state vector $\Pi(t) = [\pi_{(0,0)}(t) \;\; \pi_{(0,1)}(t) \;\; \cdots \;\; \pi_{(1,3)}(t)]$. From the Markov model of Fig. 4, the state after k sampling periods, that is time interval [0,kT], can be obtained as:

$$\Pi(kT) = \Pi(0)Q^{k} \tag{19}$$

Since the reliability after t=kT is $\pi_{(0,0)}(t) + \pi_{(0,1)}(t) + \cdots + \pi_{(1,3)}(t)$, we have

$$R(kT) = \Pi(0)Q^{k}[1 \;\; 1 \;\; \cdots 1]^{T} \tag{20}$$

where $\Pi(0) = [1 \;\; 0 \;\; \cdots \;\; 0]$.

Here $\Pi(0)$ is the initial probability of the TMR control system at each state: We assumed fault free state at the initial time.

IV. Numerical Results

Fig. 8 shows how the system reliability varies with the workload of periodic task in a sampling period(the portion of periodic task in the sampling period) and the number of checkpoints. The checkpoint overhead is assumed to be 1 % of T, checkpoint interval T is 0.1, the mission is 10^4 sampling periods, fault occurrence rates are $\lambda_i = 10^{-3}$, $\lambda_c = 5\times10^{-4}$, and the duration parameters are $\mu_i = 500$, $\mu_c = 100$. The and the duration parameters are $\mu_i = 500$, $\mu_c = 100$. The system reliability decreases as the increase of workload because of the less idle time for rollback recovery. The envelop reliability(the reliability connecting each local maximum) at first improves as the number of checkpoint is increased, but later drops off because the more checkpoints mean the more checkpoint overhead. There is a finite optimal number of checkpoints such that before this checkpoints envelop reliability increases with the increase of checkpoints and after that envelop reliability decreases because of the increased checkpoint overhead. Note that even though envelop reliability increases(decreases) before(after) the optimal number of checkpoints, reliabilities jitter along with the number of checkpoints. This jittering phenomenon is caused by the property of TMTR control system that detects errors at checkpoint times not at their occurrences. Readers can find detailed discussions of this jittering phenomenon in our previous work[12]. The optimal number of checkpoints also can be obtained by using the algorithm proposed in our previous work. Fig. 9 shows the variation of reliability to checkpoint overhead. The reliability decreases with the increase of overhead because of the increased portion of checkpoint overhead in idle time.

Figs. 10 and 11 are simulation results with the variation of μ and λ respectively. The reliability decreases with the decrease of recovery rate μ as can be seen in Fig. 10. For rapid recovery rates (large μ), the optimal number of checkpoint does not change. However, it changes for slow recovery rates(small μ). The reliability decreases with the increase of fault occurrence rate λ. However, from several simulations we can find that the optimal checkpoints that maximizes the reliability does not change with the variation of λ as can be seen in Fig. 11.

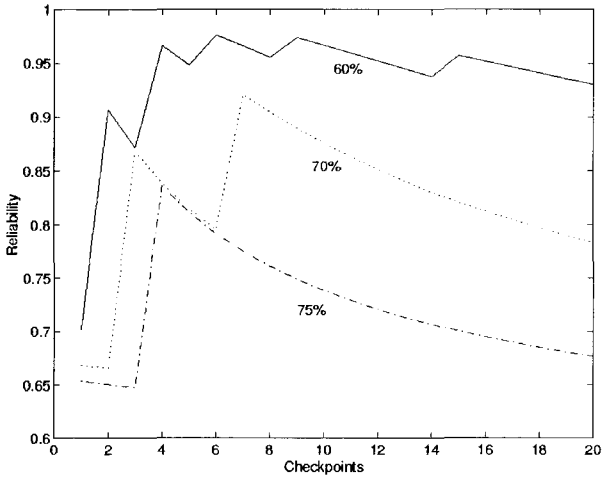

Fig. 8. Reliability vs. workload and checkpoints. ($\lambda_i = 10^{-3}$, $\lambda_c = 5\times10^{-4}$, $\mu_i = 500$, $\mu_c = 100$, t_{oh} = 1%, load = 60%, 70%, 75%)

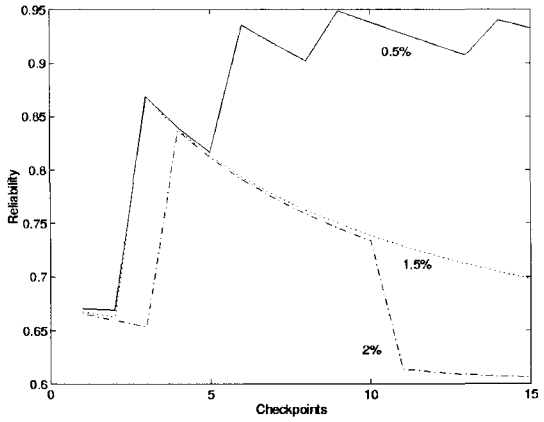

Fig. 9. Reliability vs. overhead (t_{oh}) and checkpoints.
($\lambda_i = 10^{-3}$, $\lambda_c = 5\times10^{-4}$, $\mu_i = 500$, $\mu_c = 100$, load = 70%, t_{oh} = 0.5%, 1.5%, 2%)

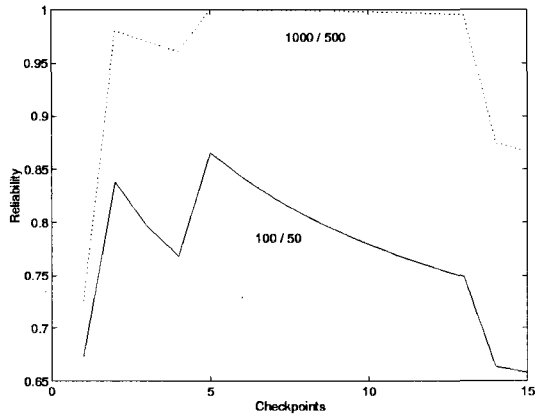

Fig. 10. Reliability vs. μ and checkpoints.
($\lambda_i = 10^{-3}$, $\lambda_c = 5\times10^{-4}$, t_{oh} = 2%, load = 60%, μ_i / μ_c = 1000/500, 100/50)

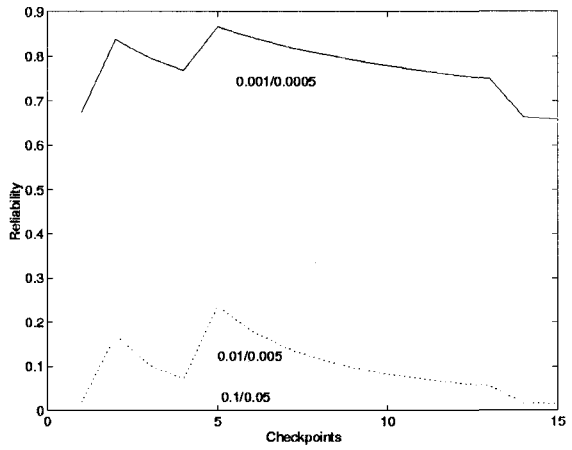

Fig. 11. Reliability vs. λ and checkpoints.
(μ_i = 500 , μ_c = 100 , t_{oh} = 2%,load = 60%, λ_i / λ_c = 0.001/0.0005, 0.01/0.005, 0.1/0.05)

V. Conclusion

The TMR structure is commonly used in many fault tolerant systems. It utilizes the space redundancy to mask and tolerate faults. The checkpoint technique, which utilizes the temporal redundancy, also has been adopted to recover from transient faults. In this study, we proposed the TMR control system that is equipped with the checkpoint strategy(TMTR control system). In this system, faults in a single processor are masked and faults in two or more processors are detected at each checkpoint time by voting three processor's results. The conventional TMR system cannot overcome faults in two or more processors. However the TMTR control system recovers from faults in two or more processors through rollbacks to the latest checkpoints. Thus the proposed TMTR control system effectively copes with both independent and correlated faults. We developed a model for this TMTR control system under correlated and independent transient faults. Assuming that the occurrence and disappearance of transient faults are modeled with an exponential probability distribution, the system reliability equation over a mission time is derived. Under the proposed control system, we showed the optimal number of checkpoints that maximizes the mission time reliability.

References

[1] H. Kim and K. G. Shin, "Design and Analysis of an Optimal Instruction Retry Policy for TMR Controller Computers", *IEEE Tr. Computers, vol.* 45, pp. 1217-1225, 1996. 11.

[2] C. M. Krishna and A. D. Singh, "Optimal Configuration of Redundant Real-Time Systems in the Face of Correlated Failure", *IEEE Tr. Reliability,* vol. 44, pp. 587-594, 1995. 12.

[3] M. Kameyama and T. Higuchi, "Design of Dependent-Failure-Tolerant Microcomputer System Using Triple-Modular Redundancy", *IEEE Tr. Computers,* vol. C-29, pp. 202-205, 1980. 2.

[4] H. Kim and K. G. Shin, "Sequencing Tasks to Minimize the Effects of Near-Coincident Faults in TMR Controller Computers", *IEEE Tr. Computers,* vol. 45, pp. 1331-1337, 1996. 11.

[5] Y.-H. Lee and K. G.. Shin, "Design and Evaluation of a Fault-Tolerant Multiprocessor Using Hardware Recovery Blocks", *IEEE Tr. Computers,* vol. C-33, pp. 113-124, 1984. 2.

[6] Krishna and A. D. Singh, "Reliability of Checkpointed Real-Time Systems Using Time Redundancy", *IEEE Tr. Reliability,* vol. 42, pp. 427-435, 1993. 9.

[7] R. Geist, R. Reynolds, and J. Westall, "Selection of a Checkpoint Interval in a Critical-Task Environment", *IEEE Tr. Reliability,* vol. 37, pp. 395-400, 1988. 10.

[8] K. G. Shin, T.-H. Lin, and Y.-H. Lee, "Optimal Checkpointing of Real-Time Tasks", *IEEE Tr. Computers,* vol. C-36, pp. 1328-1341, 1987. 11.

[9] A. Ziv and J. Bruck, "An On-Line Algorithm for Checkpoint Placement", *IEEE Tr. Computers,* vol. 46, pp. 976-984, 1997. 9.

[10] J. W. Young, "A First Order Approximation to the Optimal Checkpoint Intervals", *Comm. of the ACM,* vol. 17, pp. 530-531, 1974. 11.

[11] E. Gelenbe and D. Derochette, "Performance of Rollback Recovery Systems under Intermittent Failures", *Comm. of the ACM,* vol. 21, pp. 493-499, 1978. 6.

[12] S. W. Kwak, B. J. Choi and B. K. Kim, "Optimal Checkpointing Strategy for Real-Time Control Systems under Faults with Exponential Duration", *IEEE Tr. Reliability,* vol. 50, no. 3, pp. 293-301, Sep. 2001.

[13] S. W. Kwak and B. K. Kim, "Task Scheduling Strategies for Reliable TMR Controllers using Task Grouping and Assignment", *IEEE Tr. Reliability,* vol. 49, no. 4, pp. 355-362, Dec. 2000.

Seong Woo Kwak

Seong-Woo Kwak was born on March 10, 1970. He received the B.S, M.S and Ph. D. degrees all in electrical engineering from KAIST, Korea, in 1993, 1995 and 2000 respectively. From 2001 to 2002, he was a research professor at the Satellite Technology Research Center in KAIST, where he was involved in the development of the satellite system, STSAT-1. Since 2003, he has been a faculty member of the Department of Electric Engineering at the Keimyung University, Daegu, Korea. His research interests are in the areas of intelligent control, fault-tolerant system, satellite system design and real-time system.

Kwan Ho You

He received his B.S. and M.S. degrees in electrical engineering from KAIST in 1993 and 1996, and received his Ph.D. Degree from the University of Minnesota in 2000, respectively. He had worked as a faculty at Texas A\&M University-Kingsville. In 2001, he joined the School of ICE. at Sungkyunkwan University. His research interests are in nonlinear optimal control, sensor fusion, and estimation theory.