

# 내장형 AVTMR 시스템의 하드웨어 및 소프트웨어 신뢰성 분석

정회원 김 현 기\*

## Hardware and Software Dependability Analysis of Embedded AVTMR(All Voting Triple Modular Redundancy) System

Hyunki Kim\* *Regular Member*

### 요 약

본 논문에서는 신뢰성을 평가하는 데 있어서 소프트웨어 및 하드웨어 측면을 고려한 통합된 마코브 모델링(Markov modeling)으로 AVTMR(All Voting Triple Modular Redundancy) 시스템의 신뢰성을 분석한다. 본 시스템의 모델링은 하드웨어의 경우에 고장율이 시불변 특성을 가지며, 소프트웨어 경우에는 시 가변 특성으로 모델링되어 AVTMR 시스템 과 단일 시스템에 대한 신뢰성 비교를 한다. 특히, 소프트웨어적인 특성은 G-O/NHPP 기법을 이용하여 분석이 되며, AVTMR 시스템의 전체적인 특성을 소프트웨어 및 하드웨어적인 관점에서 고장을 따른 특성을 이해할 수 있게 된다. 평가된 AVTMR 은 임베디드 통신 시스템, 항공기 등의 결함 허용 시스템에 요구되는 스펙에 맞도록 설계를 하기 위한 기반을 제시한다.

**Key Words** : Fault-Tolerant System; Reliability; Safety; AVTMR; G-O/NHPP; Markov modeling

### ABSTRACT

In this paper, the unified Markov modeling of hardware and software for AVTMR(All Voting Triple Modular Redundancy) system is proposed and the dependability is analyzed. In hardware case, a failure rate is fixed to no time varying parameter. But, in software case, failure rate is applied with time varying parameter. Especially, the dependability(Reliability, Availability, Maintainability, Safety) of software is analyzed with G-O/NHPP for Markov modeling.

The dependability of single and AVTMR system is analyzed and simulated with a unified Markov modeling method, and the characteristic of each system is compared according to failure rate. This kind of fault tolerant system can be applied to an airplane and life critical system to meet the requirement for a specific requirement.

### I. 서 론

산업사회의 발전에 따라 인간의 생명과 밀접한 시스템에서 높은 신뢰도(Reliability)와 안전도(Safety) 및 결함(Fault)을 허용(Tolerance)할 수 있는 시스템이 요구되어 왔다. 이러한 문제를 해결하기 위해 다수결 보터(Majority voter)를 이용하는 최초의 결함

허용 시스템(Fault tolerant system)이 아폴로의 유도 시스템에 적용이 되었으며, 인간의 생명과 관련된 시스템인 항공기, 철도 시스템, 통신 시스템 등에 사용되어 왔다<sup>[1][2]</sup>.

이러한 결함허용 시스템을 개발하기 위해 고장의 연구가 필요하게 되었고, 결함이 오류를 발생시키고, 오류가 시스템의 고장을 발생시킨다는 것을 알게

\* 팬택 해외 마케팅 본부 상품기획팀 상품기획 팀장 (r48019@naver.com)

논문번호 : KICS2009-03-094, 접수일자 : 2009년 3월 11일, 최종논문접수일자 : 2009년 6월 12일

되었다. 즉, 결함허용 시스템(Fault tolerant system)이란, 결함이 발생했을 때, 시스템이 오류나 고장으로 전이되는 것을 방지하는 시스템이라 볼 수 있다. 결함을 방지하는 기법으로는 결함 회피(Fault avoidance)와 결함허용(Fault tolerance)이 있다. 결함 회피는 전자소자의 질을 향상시켜서 고장율(Failure rate)을 작게 하고, 시스템에 대한 완벽한 테스트를 통해서 시스템을 구성하는 방법으로, 실질적으로 구현하기 어려운 시스템이다. 왜냐하면, 시간이 지나 감에 따라 전자소자의 질은 떨어지고, 예상치 못한 경우에 대해서 완벽한 테스트를 한다는 것은 어려운 일이기 때문이다. 결함허용 시스템(Fault tolerant system)은 시스템에서 결함이 발생하더라도 정상적인 동작이 계속 유지되는 시스템이다. 그래서, 결함허용 시스템은 결함회피 시스템보다 시스템을 개발하는데 있어서 많은 장점을 가지게 된다. 일반적으로 결함허용 시스템은 여분을 가지고 있는 구조로 되어있고, 결함이 발생을 하여도 정상적인 동작을 멈추지 않고 동작을 하는 특성을 가지게 된다. 본 논문에서 설계된 AVTMR의 하드웨어 결함허용 기법은 소프트웨어 결함허용 시스템보다 시간이 중요시되는 시스템에 적용되며, 하드웨어 결함허용 시스템은 버스 레벨로 데이터를 비교하고, 보팅(voting)을 하는 특성을 가지게 된다<sup>[14]</sup>.

이렇게 설계된 시스템에 대한 신뢰성을 평가하기 위해서, 기존에 발표된 논문에서는 하드웨어의 관점에서만 평가를 하였던 것에 소프트웨어의 고장율을 포함한 신뢰성을 평가하기 위해, 즉 하드웨어와 소프트웨어의 통합 모델링으로 시스템의 신뢰성을 평가한다<sup>[15-17]</sup>. 하드웨어의 경우는 고장율이 상수의 특성을 가지고, 소프트웨어는 고장율이 시변 특성을 가지는 것으로 모델링을 하였다. 특히 소프트웨어 기법은 비동차형 포이송 분포인 G-O/NHPP를 이용하여 하드웨어 및 소프트웨어의 통합된 모델링을 이용하였다. 이렇게 모델링된 AVTMR 시스템과 단일 시스템은 고장율에 따른 신뢰성 특성을 분석한다.

이렇게 제안된 시스템은 높은 신뢰도, 안전도가 요구되는 시스템인 항공기, 철도, 선박, 통신 시스템의 요구를 만족할 수 있도록 적용이 될 수 있다.

## II. 통합 모델링

본 논문에서 사용하는 마코브 모델링(Markov Modeling)은 조합모델의 단점을 극복하고 시간의 상태에 따른 시스템 천이를 이용하여 하드웨어 및

소프트웨어를 통합하여 마코브 모델을 이용하여 시스템을 모델링 한다. 시스템 고장의 상태 변화는 이산 시스템 모델로 표현이 되며, 그에 따른 시스템의 신뢰도를 평가한다. 본 논문에서는 다음의 두가지 조건을 가정하여 시스템을 모델링 한다.

- 1) 시스템의 모든 모듈이 정상적으로 바르게 동작하는 상태로 시작을 한다.
- 2) 동일시간에 한 개의 고장만 발생할 수 있다.

### 2.1 하드웨어 모델링

하드웨어 모델링 기법은 확정된 고장율(failure rate)를 가지고, 마코브 모델링(Markov Modeling)을 사용한다. 하드웨어 마코브 모델링은 다음과 같은 조건으로 모델링을 하게 된다.

- 1) 시스템이 동작하는 순간  $t_0$ 에 전체 시스템은 정상동작을 한다.
- 2) 사용된 부품의 고장율은 상수이다.

### 2.2 소프트웨어 모델링

본 논문에서는 소프트웨어 모델링을 위해서 G-O/NHPP 기법을 사용한다. 기본적으로 소프트웨어의 신뢰성이 예측되는 단계는 최종 개발 완료된 후에 시스템을 평가하게 된다. 그러므로, 소프트웨어 수행도는 최종 평가의 기간과 밀접한 관련이 있다.

$$\Pr\{N(t) = y\} = \text{poim}[y; m(t)], y = 0, 1, 2, \dots; \quad (1)$$

$$m(t) = a[1 - \exp(-bt)]; \quad (2)$$

$$\lambda_{sw}(t) = dm(t)/dt = ab \cdot \exp(-bt); \quad (3)$$

즉, 소프트웨어 신뢰도 모델의 결함 개수는 최종 평가 및 상용화되어 동작하는 동안 적용된다. 이런

표 1. G-O/NHPP 수식에 사용된 기호 정리

기 호	의 미
$\Pr\{N(t) = y\}$	G-O/NHPP 확률밀도 함수
$\text{poim}(\cdot; \mu)$	포이송 분포 확률밀도 함수
$m(t)$	시간 t까지 결함의 평균값
$f_{t_0}$	소프트웨어 고장율의 확률 밀도 함수
$l_r$	r 번째 시간 간격
a	결함의 평균값
b	결함 검출율

결함 개수의 예측을 G-O/NHPP 기법을 사용해서 예측하게 되며, 초기 설계시에 목표설정에 따른 시스템 개발을 유도 할 수 있도록 분석을 할 수 있다.

기본적으로 G-O/NHPP는 식 (1)에 나타난 것과 같이 포아송 분포를 따르게 된다. 식 (2)에서 나타나 있는  $m(t)$ 는 시간  $t$ 까지 결함의 평균값으로 지수함수로 표시되며,  $a$ 는 최종 결함의 평균값으로 시간이 오래 갈수록 결함의 평균값은 이 값으로 수렴을 하는 특성을 가지고 있으며, 식(3)은 소프트웨어의 고장율을 나타내고 있다.

2.3 통합 모델

본 논문에서 사용되는 통합 모델은 하드웨어 및 소프트웨어 신뢰도를 평가하는 방법으로 마코브 모델링을 이용한 통합모델의 기본적인 제약 조건은 다음과 같다.

- 1) 하드웨어 결함은 시간에 따른 지수 분포 함수의 특성을 가진다.
- 2) 소프트웨어 고장은 G-O/NHPP의 모델로 표현된다.
- 3) 시간 간격( $t, t + \Delta t$ )에서 발생하는 고장의 확률은  $\lambda(t) \cdot \Delta t$ 로 평가된다.

$$\begin{aligned}
 & f_{i_0}(l) \cdot \Delta l \\
 &= \text{poim}[r-1, m(l)] \cdot \lambda(t) \cdot \Delta l \\
 &= 1 - \lambda(t) \cdot \Delta l \cdot \sum_{r=1}^{\infty} \text{poim}[r-1, m(l)] \\
 &= 1 - \lambda(t) \cdot \Delta l \cdot \text{poif}[\infty, m(l)] \quad (4) \\
 &= 1 - \lambda(t) \cdot \Delta l \\
 &= 1 - \lambda(t) \cdot \Delta t
 \end{aligned}$$

소프트웨어에서 발생하는 고장에 대한 마코브 모델링(Markov Modeling)은 식 (4)로 나타낼 수 있다.  $f_{i_0}$ 는 소프트웨어의 확률 밀도 함수로 시간의 함수인  $\Delta l$ 의 곱은 소프트웨어 상태 천이 확률이 된다. 이 수식으로 포아송 분포를 이용한 방정식을 유도하면 식(4)의 값이 된다<sup>[11]</sup>.

그러므로 식(4)는 G-O/NHPP 특성을 가지는 소프트웨어 고장율에 따른 상태 천이를 나타내고 있으며, 소프트웨어의 고장율은 시변 함수로 표현되는 것을 알 수 있다. 이와 유사한 방법으로 하드웨어의 고장율에 따른 하드웨어 상태 천이 확률은 식(5)과 같이 나타낼 수 있다.

$$1 - \lambda_{HW} \cdot \Delta t \quad (5)$$

결국 하드웨어 소프트웨어 통합 모델링에 식(4)와 식(5)을 사용하며, 소프트웨어의 고장율은 시간에 따른 변화를 나타내는 것으로 볼 수 있으며, 하드웨어 고장율은 시불변 값을 가지며 신뢰도 함수가 유도될 수 있다는 것을 알 수 있다.

III. 시스템 통합 모델링

3.1 단일 시스템 모델링

단일 시스템의 상태 천이도는 그림 1과 같이 나타낼 수 있다. 즉, 정상적인 동작인  $P_0$ , 고장난 상태인  $P_f$  두 가지 상태로 표현될 수 있다.

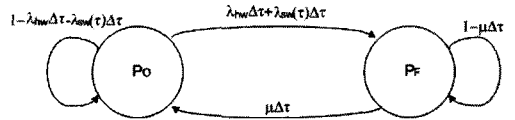


그림 1. 단일 시스템 모델링

그림 1에서 보는 바와 같이 단일 시스템 모델링의 하드웨어/소프트웨어 고장율은 각각  $\lambda_{HW}, \lambda_{SW}(t)$ 로 표현될 수 있으며, 정상동작에서 고장이 발생할 경우인 하드웨어, 소프트웨어의 고장율에 따른 상태 천이로 표현되며, 고장난 상태에서 정상동작으로 전이되는 수리율은  $\mu$ 로 표시된다.

3.2 AVTMR 시스템 모델링

다수결 보터를 CPU 보드에서 3중화하여 한 개의 보터가 고장이 발생해도 시스템에서 결함이 발생하지 않고 정상적인 동작을 할 수 있는 AVTMR 시스템이 그림 2에 나타나 있다. 각각의 시스템은 CPU를 가지고 있고, 입/출력보드를 가지고 있다.

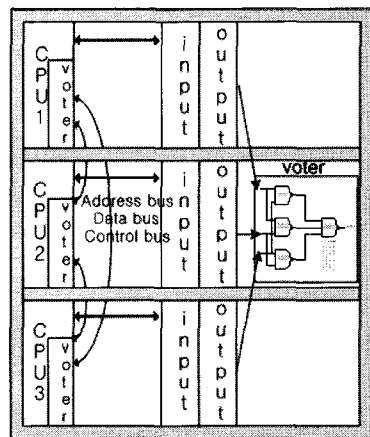


그림 2. 보터와 AVTMR 시스템 구조

CPU와 각각의 입/출력보드에 전달하는 데이터는 CPU에 있는 내부 보터에 의해서 어드레스 버스, 데이터 버스, 제어버스의 비교가 이루어진다. 출력 보터인 경우에는 3개의 출력 보터에서 최종 한 개의 출력을 내야 하므로 출력 보터를 사용하고 있다. AVTMR 시스템의 신뢰도를 평가하기 위하여 그림 2에 대한 시스템 마코브 모델을 그림 3과 같이 구성하였다. 그림 3의 고장율에서 첨자 P는 프로세서를 의미한다. 즉, 3P는 3개의 CPU가 정상적인 동작을 하는 것을 나타내고, 상태 변화에 의해서 2P가 되는 경우는 프로세서 1개가 고장이 나는 경우를 나타낸다. 3V는 3중화 보터를 의미하고, 2V는 3중화 보터 중 한 개가 고장이 나는 경우를 나타낸다.

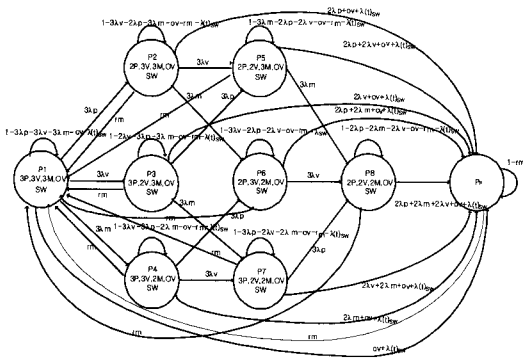


그림 3. AVTMR 마코브 모델

표 2. 모델링에 사용된 기호의 정리

기 호	의 미
$\lambda_p$	프로세서 고장율
$\lambda_m$	프로세서를 제외한 고장율 (메모리, 주변장치/입출력)
$\lambda_v$	내부보터의 고장율
$\lambda_{ov}$	출력보터의 고장율
$\lambda_{sw}(t)$	소프트웨어 고장율
$\mu$	수리율

표 3. AVTMR의 상태에 따른 동작상태

	CPU (P)	주변장치 (M)	내부보터 (V)	출력보터 (OV)	소프트웨어 (SW)	동작 상태
P1	세계정상	세계정상	세계정상	정상	정상	정상
P2	한개고장	세계정상	세계정상	정상	정상	정상
P3	세계정상	세계정상	한개고장	정상	정상	정상
P4	세계정상	한개고장	세계정상	정상	정상	정상
P5	한개고장	세계정상	한개고장	정상	정상	정상
P6	한개고장	한개고장	세계정상	정상	정상	정상
P7	세계정상	한개고장	한개고장	정상	정상	정상
P8	한개고장	한개고장	한개고장	정상	정상	정상
Pf	독립적으로 같은 모듈이 두 개의 고장이 났을 경우			고장	고장	고장

표 4. 각 신뢰성 방정식 테이블

RAMS	방정식
Reliability	Single: $P_o(t)$ AVTMR: $P_1(t) + P_2(t) + P_3(t) + P_4(t) + P_5(t) + P_6(t) + P_7(t) + P_8(t)$
Availability	Single: $P_o(t)$ AVTMR: $P_1(t) + P_2(t) + P_3(t) + P_4(t) + P_5(t) + P_6(t) + P_7(t) + P_8(t)$
Maintainability	$1 - e^{-\mu t}$
Safety	Single: $P_o(t)$ AVTMR: $P_1(t) + P_2(t) + P_3(t) + P_4(t) + P_5(t) + P_6(t) + P_7(t) + P_8(t)$

3M은 CPU를 제외한 나머지 소자의 정상적인 상태를 의미하고, 2M은 한 개의 모듈이 고장난 상태를 나타내고, Ov는 출력 보터를 의미한다<sup>[9][10]</sup>.

소프트웨어는 AVTMR의 각 모듈이 같은 소프트웨어를 탑재하기 때문에 한 개의 오류가 발생하면 고장이 발생하는 것으로 시스템을 구성하였다. 각 상태에서 시스템이 고장이 발생을 하였을 때는 같은 수리율을 가지는 것으로 시스템의 마코브 모델을 구성하였다. 즉, 그림 3에서는 각 상태에서 3개의 시스템이 정상 동작하는 상태로 같은 수리율로 전이가 되도록 구성이 되어 있다. 표 3은 전체적인 AVTMR 시스템의 동작 상태로 각각의 상태에 따른 시스템의 동작상황을 나타낸다.

#### IV. 신뢰성 시뮬레이션

본 논문에서 마코브 모델링을 이용해서 유도된 하드웨어/소프트웨어 통합 모델링 방정식을 이용해서 단일 시스템과 AVTMR 시스템이 각각의 고장율에 따른 시스템 신뢰성을 살펴보도록 한다. 그림 1과 3에 나타난 단일 시스템과 AVTMR 시스템의 소프트웨어 및 하드웨어 고장율의 특성을 가지는 상태 천이도를 이용해서 RAMS(Reliability, Availability, Maintainability, Safety) 즉, 신뢰도 안전도, 가용도, 유지보수도를 구할 수 있다. 하드웨어 고장율은 단일 시스템 및 AVTMR의 설계에 사용된 시스템 고장율<sup>[9][10]</sup>을 사용하였으며, 소프트웨어 고장율은  $a=2, a=10, b=0.001, b=0.0001$ 에 따른 시뮬레이션을 하였다. 특히, 마코브 모델로 유도된 식은 Runge-Kunta를 이용해서 시스템 분석을 하였다.

##### 4.1 신뢰도

그림 4, 그림 5는 AVTMR 시스템과 단일 시스템의 신뢰도를 나타내고 있다. 기본적으로 하드웨어의 고장율은 시간에 따라 변경되지 않은 상수의 특

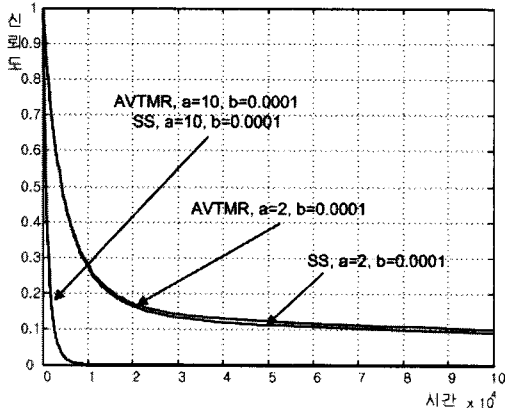


그림 4. 신뢰도(a=2, b=0.0001 or b=0.001)

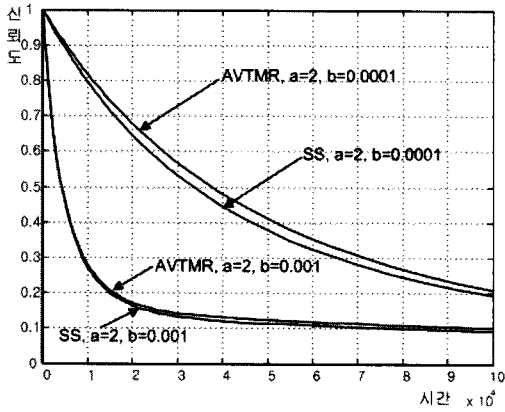


그림 5. 신뢰도(a=2 or a=10, b=0.0001)

성을 가지므로 시간에 따라 변수가 변성이 되는 소프트웨어의 고장율에 따른 변화를 보는 것에 중점을 두었다. 특히, 소프트웨어의 고장율의 변수인  $a$  인 결함의 평균값과  $b$ 인 결함 검출율에 대한 분석을 해보았다. 기본적으로 AVTMR시스템이 같은 조건하에 단일 시스템(SS)보다 신뢰도의 특성이 우수하다는 것을 볼 수 있다. 그림 4는 결함 검출율 변경에 따른 신뢰도의 특성을 나타내고 있다. 즉, 결함 검출율이 높을 수록 신뢰도가 낮게 나타나는 특성을 볼 수 있다. 그림 5는 결함의 평균값에 따른 신뢰도의 분석을 나타내고 있는데, 결함의 평균값이 작을수록 신뢰도가 높은 특성을 가지게 된다. 그러므로, 시스템 설계시에 각각의  $a$ ,  $b$ 의 목표를 설정하여 그에 따른 소프트웨어 설계를 하면 전체 시스템의 신뢰도 분석이 가능하여 요구되는 시스템 설계를 할 수 있게 된다. 그림 4는 결함 검출율에 변경에 따른 신뢰도의 특성을 나타내고 있다. 즉, 결함 검출율이 높을 수록 신뢰도가 낮게 나타나는 특성을 볼 수 있다. 그림 4는 결함의 평균값에 따른

신뢰도의 분석을 나타내고 있는데, 결함의 평균값이 작을 수록 신뢰도가 높은 특성을 가지게 된다. 그러므로, 시스템 설계시에 각각의  $a$ ,  $b$ 의 목표를 설정하여 그에 따른 소프트웨어 설계를 하면, 전체 시스템의 신뢰도 분석이 가능하게 된다.

#### 4.2 가용도

시스템의 가용도도 마찬가지로 소프트웨어 변수에 따른 특성을 분석하였다. 그림 6와 그림 7에서 볼 수 있듯이 하드웨어 및 소프트웨어 특성을 이용한 가용도도 단일 시스템보다 더 우수한 특성을 가진다는 것을 알 수 있다.

그림 6에서는 결함 검출율에 따른 시스템의 특성을 나타내고 있다. 결함 검출율이 적을 수록 시스템의 가용도가 높아지는 것을 알 수 있으며, 시스템의 초기 3000시간 까지는 결함 검출율이 적을 경우 가용도가 높게 나타나지만 그 이후 부터는 더 높은 가용도를 가진다는 것을 알 수 있다. 그림 7은 결함의 평균값에 따른 가용도의 변화를 나타내고 있다.

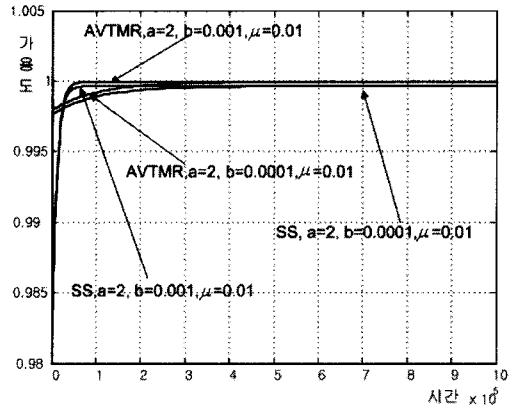


그림 6. 가용도(a=2, b=0.001 or 0.0001, μ=0.01)

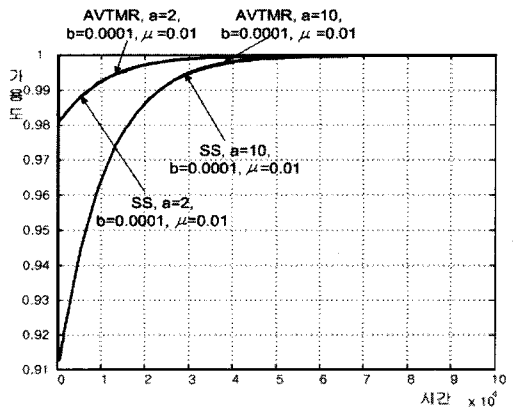


그림 7. 가용도(a=2 or 10, b=0.0001, μ=0.01)

결함의 평균값이 적을 수록 항상 더 높은 가용도를 나타내고 있다는 것을 알 수 있다.

### 4.3 유지보수도(Maintainability)

그림 8은 유지보수도(Maintainability)를 나타내고 있다. 이 시뮬레이션의 결과는 실질적으로 어떠한 동작을 하는 시스템에 적용될 때, 시스템이 수리가 되어서 정상적인 동작을 할 수 있는 유지보수도(Maintainability)의 결과이다. 그림 8은 단일 시스템 또는 AVTMR 시스템이건 고장이 발생하였을 때, 수리율(repair rate)에 따라서 시스템의 유지보수도가 달라지는 것을 나타내고 있다. 즉, 수리율이 클수록 다시 말해서 수리를 하는 시간이 짧을수록 시스템의 유지 보수도(Maintainability)가 우수하다는 것을 알 수 있다. 그러므로, 시스템의 가용도를 높이기 위해서는 유지 보수도가 높을수록 좋은 가용도를 가진다는 것을 알 수 있다.

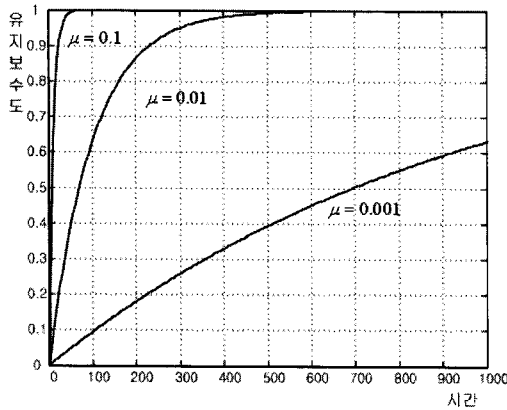


그림 8. 유지 보수도

### 4.4 안전도

본 논문에서는 안전측 동작의 특성을 가지는 하드웨어 설계가 되어 있지 않으므로 신뢰도와 같은 결과를 가지게 되므로 그림 4, 그림 5와 같은 특성을 가지게 된다.

## V. 결 론

본 논문에서는 결함 허용 시스템인 제안된 AVTMR 시스템과 단일 시스템의 신뢰성 분석을 하드웨어와 소프트웨어 고장율 기법을 적용하여 분석을 하였다. 하드웨어는 고장율은 MILSpec-217F를 기준으로 상수로 평가하였으며, 소프트웨어는 G-O/NHPP 방식을 이용하여 소프트웨어의 고장율

이 시간의 함수로서 시스템 분석을 하였다. 이러한 방법으로 단일 시스템에 비해서 AVTMR 시스템의 신뢰도, 가용도, 안전도가 우수하다는 특성을 시스템 분석을 통해서 알 수 있었다. 특히, 상수의 특성을 가지는 하드웨어의 고장율 변수 보다 소프트웨어의 고장율 변수인 평균 결함의 개수 및 결함 검출율 함수에 따른 시스템의 신뢰성이 변경되는 특성을 알 수 있었다. 즉, 평가된 시스템에서 신뢰도의 경우 결함의 개수가 작을 수록 더 높은 신뢰도를 가지고, 결함 검출 밀도가 작을 수록 높은 신뢰도의 특성을 가진다는 것을 알 수 있었다. 또한, 가용도 측면에서는 평균 결함 검출수가 작을 수록 좋은 가용도를 가진다는 것을 알 수 있었으며, 결함 검출 밀도가 높을 수록 좋은 가용도를 가지지만 일정 시간 즉, 3000시간 까지는 낮은 결함 검출 밀도가 더욱 우수한 특성을 가진다는 것을 알 수 있었다. 본 논문에서는 AVTMR 시스템의 하드웨어와 소프트웨어의 특성을 포함한 시스템의 신뢰성을 분석하여 우수한 시스템 특성을 갖기 위해서는 결함의 최종 평균값이 작고, 결함 검출을 낮출 수록 좋은 시스템 특성을 가진다는 것을 알 수 있었다. 기본적으로 하드웨어는 상수의 고장율을 가지므로, 시스템의 신뢰성을 목표로 하는데 있어서 소프트웨어의 변수값을 설정하여 특정한 시스템 신뢰성을 갖도록 설계를 하기위한 목표를 설정하기 위한 방법을 제시하였다.

## 참 고 문 헌

- [1] Barry W. Johnson, "Design and Analysis of Fault-Tolerant Digital Systems", Addison Wesley Publishing Company, 1989.
- [2] Daniel P. Siewiorek and Robert S. Swarz, "Reliable Computer System" Second Ed, Digital Press, 1992.
- [3] Hyunki Kim, "The Design and Evaluation of All Voting Triple Modular Redundancy", The Annual Reliability and Maintainability Symposium, Seattle, WA, USA, January, 2002.
- [4] Dhiraj K. Pradhan, "Fault-Tolerant Computer System Design", Prentice Hall, 1996.
- [5] 김현기, 이기서, "AVTMR 시스템 설계 및 RAM 평가", 제12호, Vol.25, 한국통신학회 2000.
- [6] 김현기외 2인, "AVTMR과 듀얼 듀플렉스 시스템 비교에 관한 연구", 제26호, 6호, 한국통신학

회 논문지, 2001.

- [7] Stephen R.Welke, "Reliability Modeling of Hardware/Software Systems", *IEEE Trans. On Reliability*, Vol.44, No.3, 1995 September.
- [8] 김현기의 2인, "보터의 구조에 따른 TMR 시스템의 신뢰도 평가에 관한 연구", *전기학회 춘계학술대회*, 1998.
- [9] Jeffrey A. Clark and Dhiraj K.Pradhan, "Reliability Analysis of Unidirectional Voting TMR systems through Simulated Fault-Injection" *IEEE Tran. on Reliability*, Vol.38, No.7, 1992 July.
- [10] JOHN F. WAKERLY, "Microcomputer Reliability Improvement Using Triple Modular Redundancy", *PROCEEDING OF THE IEEE*, Vol.64, No.6, JUNE 1976.

김 현 기 (Hyunki Kim)

정회원



1993년 2월 광운대학교 공과대학 제어계측공학과 학사

1995년 2월 광운대학교 대학원 제어계측공학과 석사

2001년 6월 광운대학교 대학원 제어계측공학과 박사

2006년 8월 고려대학교 경영대학원(MBA) 경영학 석사

1999년 6월~2002년 8월 모토로라 선임 연구원

2002년 9월~2004년 10월 텔슨전자 책임연구원

2005년 1월~2009년 7월 팬택 중앙연구소 수석연구원 US4팀장

2009년 7월~ 현재 팬택 해외 마케팅 본부 상품기획팀장

<관심 분야> 결합허용 시스템, 이동 통신, 영상처리, 휴대 인터넷, 실시간 운영체제, 모바일 브라우저, 모바일 플랫폼, Emerging Device