

분산 실시간 시스템을 위한 고장탐지

김태규⁰ 노진홍 홍영식
동국대학교 컴퓨터공학과
(ashes, jhno, hongys)@kiss.or.kr

Fault Detection for Distributed Real-time Systems

Tae-Gyu Kim⁰, Jin-Hong No, Young-Sik Hong
Dept of Computer Engineering, Dongguk Univ.

요 약

실시간 분산처리 시스템에서의 통신은 신뢰성이 높아야 한다. 신뢰성이 높은 그룹통신 프로토콜을 개발하기 위해서는 발생 가능한 고장들을 알아내고, 고장 발생 시에 고장을 탐지하고 적절한 처리를 할 수 있어야 한다. 기존의 ack 및 time-out 에 의한 탐지방식은 수신되지 않는 ack에 대한 부하로 인하여 실시간 시스템에 있어서는 적합하지 않다. 따라서 본 논문에서는 분산 실시간 시스템에 적용 가능한 고장 탐지 기법을 실험하고 그 결과를 분석하여 후에 신뢰성 있는 메시지 전송 및 데드라인을 보장할 수 있도록 한다.

1. 서 론

인터넷 기반 네트워크 컴퓨팅의 급속한 성장과 초고속통신망의 개통으로 실시간 분산처리 시스템에 대한 수요가 가속화되고 있다. 따라서 정보기술산업 분야에서는 현재 사용되고 있는 기법보다 수배 또는 수십 배 더 효율적인 실시간 분산처리 시스템 개발기법을 필요로 하고 있다. 실시간 시스템은 실제로 설계되기 이전에 시스템이 기능적으로 올바르게 동작하여 정확한 값을 산출할 뿐만 아니라 주어진 제약조건을 위배하지 않음을 보장하여야 한다. 따라서 실시간 시스템 설계 시 여러 모델링 기법의 사용이 제안되어 왔으며 실시간 시뮬레이션 기법도 연구되고 있다.

이러한 네트워크 컴퓨팅 환경에서 신뢰성 있고 안정적인 분산 실시간 처리 응용 프로그램의 개발을 위해서는 분산 실시간 처리시스템에 관련된 전반적인 분야의 발전이 필요하며, 특히 그 중에서도 실시간 분산처리 시스템의 고장 탐지기능은 매우 중요하다.

따라서 본 논문에서는 분산 실시간 시스템에서 사용 가능한 고장 탐지 기법을 실험하였다. 본 연구에서 실시한 실험은 분산 환경을 지원하는 시스템 설계모델인 TMO (Time-triggered Message-triggered Object) [7]을 기반으로 하였다.

2. 관련연구

SNS(Supervisor-based Network Surveillance)[1]는 NS(Network Surveillance)를 실시간 환경에 맞도록 중앙 집중식 감독자(supervisor)를 사용하여 고장을 감지하는 방법이다. SNS는 일대일 네트워크 구조를 가지고 있으며 주기적으로 이웃 노드에게 heartbeat 메시지를 전송하면서 자기 자신도 이웃 노드로부터 고동 메시지를 전송 받는 방법이다. 이웃 노드로부터 heartbeat 메시지가 없을 경우 이웃 노드를 고장으로 간주하여 감독자에게 고장 처리를 위해 suspicion report를 제출하고, 감독자는 고장노드를 모든 노드들에게 알려준다. SNS는 시스템

에 고장난 노드가 발생하였을 때나 시스템의 일부만이 고장처리중일 때에 고장이 아닌 정상 노드를 빠르게 알 수 있다. 또한 고장처리 후의 새로운 시스템의 구성을 빠르게 할 수 있으며 다양한 고장 유형을 탐지할 수 있다는 장점이 있다.

PSTR (Primary-Shadow TMO Replication) [2]은 DRB/PSP(primary-shadow active replication principle)의 방법[3]과 TMO 구조[4][5]를 통합시킨 방법이다. PSTR은 서로 다른 주 TMO(primary TMO)와 보조 TMO(shadow TMO)가 네트워크 상에 존재하여 PSTR 본부(station)를 구성한다. 주 TMO와 보조 TMO는 클라이언트의 요청을 함께 받아 다음과 같은 방법으로 처리한다. 주 TMO는 요청을 받았는지를 보조 TMO에게 통보하고, 정상적으로 요청이 처리가 되었는지를 보조 TMO에게 통보한다. 마지막으로 주 TMO가 처리 결과를 클라이언트에게 응답하는데 성공했는지를 통보한다. 만약 일정 시간 안에 주 TMO가 요청을 못 받거나 처리를 하지 못하는 경우, 혹은 처리 실패를 보조 TMO에게 통보한 경우나 응답 통보를 하지 못한다면 보조 TMO가 처리한 작업을 대신 클라이언트에게 응답하게 된다. 이 방법은 실시간 환경에 적합하게 시간 적시성(timeliness)을 제공할 수 있다는 장점이 있다.

3. 링 기반 고장 탐지 기법

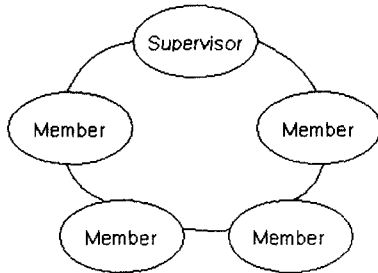
실시간 분산처리 시스템에서의 통신은 신뢰성이 높아야 한다. 신뢰성이 높은 그룹통신 프로토콜을 개발하기 위해서는 발생 가능한 고장들을 알아내고, 고장 발생 시에 고장을 탐지하고 적절한 처리를 할 수 있어야 한다. 즉, 실시간 분산처리 시스템에서 발생하는 그룹통신 메시지는 수신 프로세서들 모두에게 전달되었는가 아니면 메시지 자체가 무시되어야 한다.

고장 감내 그룹통신을 위해서 실시간 처리 시스템에서 발생 가능한 고장의 유형은 다음과 같이 4가지로 분류된다.

- 프로세서의 고장
 - 입력 통신 처리기의 고장
 - 출력 통신 처리기의 고장
 - 통신 네트워크의 고장
- 프로세서의 고장은 물리적으로 노드가 고장난 것을 의미하

며, 입력 통신 처리기의 고장은 수신자가 메시지를 받지 못하는 것을 의미하고 출력 통신 처리기의 고장은 이와 반대로 송신자가 메시지를 전달하지 못하는 것을 의미한다. 통신 네트워크의 고장은 노드와 노드를 연결하는 통신 링크(communication link)의 고장을 의미한다. 4종류의 고장들은 모두 노드 고장으로 간주되어 처리된다고 가정한다.

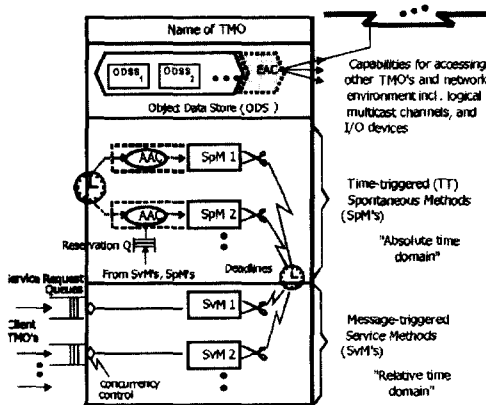
고장을 감지하기 위해 모든 노드는 다음 그림과 같이 논리적 링으로 구성되어 있다. 논리적 링 정보를 통해 2개의 이웃 노드들을 결정하고 이웃 노드들끼리 고장 여부를 주기적으로 검사한다.



[그림 1] 논리적 링으로 구성된 시스템 구조

모든 노드들은 자신의 존재를 알리기 위해 이웃 노드에게 주기적으로 heartbeat 시그널을 전송한다. 반대로 모든 노드들은 이웃 노드로부터 주기적으로 heartbeat 시그널을 수신하고 일정 기간동안 heartbeat 시그널이 도착되지 않는다면 이웃 노드를 고장으로 간주하게 되어 감독자(supervisor)에게 suspicion report를 전송한다. Suspicion report를 수신한 감독자는 해당 노드의 고장을 인식하고 이러한 고장 사실을 정상 노드들에게 전달한다.

이와 같은 프로토콜 엔진의 시뮬레이션을 위해서 본 연구에서는 TMO(time-triggered message-triggered object) 모델을 기본모델로 채택하였다. [그림2]와 같이 TMO모델은 세 가지 객체로 구성되어 있다. SpM(spontaneous method)이라 불리는 시간 구동 메소드는 주어진 시간 조건 AAC(automatic



[그림 2] TMO 시뮬레이션 모델

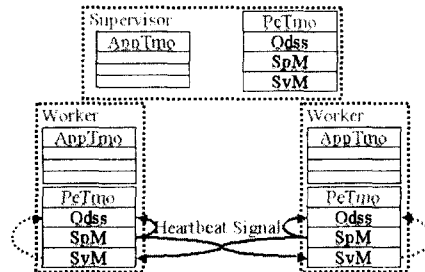
activation condition)가 만족되면 자동으로 호출되는 특징을 가지며, SvM(service method)은 외부로부터의 메시지 수신 등

이벤트에 반응하는 메소드이다. 또한 ODSS(object data store)는 SpM과 SvM사이의 데이터공유 및 동기화의 역할을 수행하고 있다.

고장 검출을 위한 시뮬레이션 모델은 TMO 모델을 사용하여 작성되고, 고장 검출을 위해 APP TMO에 독립적으로 PE TMO가 heartbeat signal을 전송한다. PE TMO가 전송하는 heartbeat signal은 다음과 같이 두 가지 종류로 나누어 볼 수 있다.

- 일반 노드끼리의 heartbeat 시그널 교환
 - 일반 노드와 감독자와의 heartbeat 시그널 교환
- 먼저 일반 노드끼리의 heartbeat 시그널을 교환하기 위해서 다음 그림과 같이 TMO 실시간 처리 프로그램을 작성한다.

PE TMO는 주기적으로 heartbeat 시그널을 이웃 노드에게 전송을 하는데, 이는 주어진 시간마다 실행되는 PE TMO SpM이 담당하여 PE TMO SpM이 실행될 때 heartbeat 시그널을 이웃 노드에게로 전송한다. 그러면 이웃 노드의 PE TMO SvM이 heartbeat 시그널을 수신하고 이를 PE ODSS에 저장한다. 그리고 주기적으로 PE TMO SpM이 PE ODSS를 조사하여 이웃 노드의 heartbeat 시그널이 수신되었는지 체크한다. 그러므로 각 PE TMO들이 하는 작업은 다음과 같다.



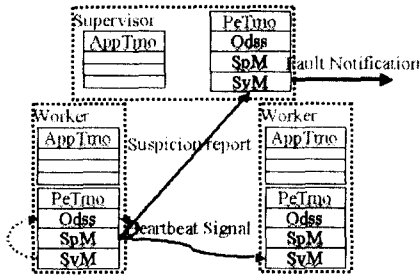
[그림 3] 노드끼리의 heartbeat signal 교환

모든 PE TMO SpM은 주기적으로 실행될 때마다 heartbeat 시그널을 이웃 노드로 전송하는 작업과 수신된 heartbeat 시그널을 PE ODSS에서 체크하는 작업을 번갈아 가면서 실행한다. 모든 PE TMO SvM은 이웃 노드가 보낸 heartbeat 시그널을 수신하고 PE ODSS에 저장한다.

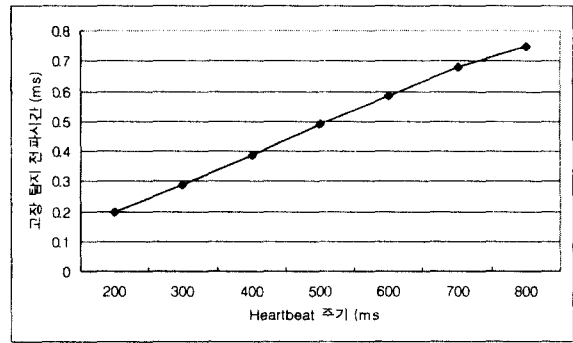
만약 heartbeat 시그널이 수신되지 않은 경우라면 다음 그림과 같이 처리를 한다. PE TMO SpM이 PE ODSS를 검사하는 도중 heartbeat 시그널이 수신되지 않은 것을 체크하면 supervisor에게로 suspicion report를 전송한다. Supervisor의 PE TMO SvM은 suspicion report를 수신한 후 해당 노드의 고장을 인식하게 되고, 정상 노드들에게 해당 노드의 고장을 알리며 고장 노드를 제외한 논리적 링 정보를 정상 노드들에게로 전달한다. 그러므로 일반 노드가 고장이 발생했을 경우 각 PE TMO들이 하는 작업은 다음과 같다.

모든 PE TMO SpM에서는 Heartbeat 시그널을 PE ODSS에서 체크하고 이웃 노드가 고장난 사실을 발견하면 감독자 노드에게로 suspicion report를 전송한다.

감독자 노드의 PE TMO SvM에서는 그룹 멤버가 보낸 suspicion report를 수신하고 고장난 노드를 제외한 논리적 링 정보와 그룹 정보를 멤버들에게 전달한다.



[그림 4] 일반 노드의 고장검출



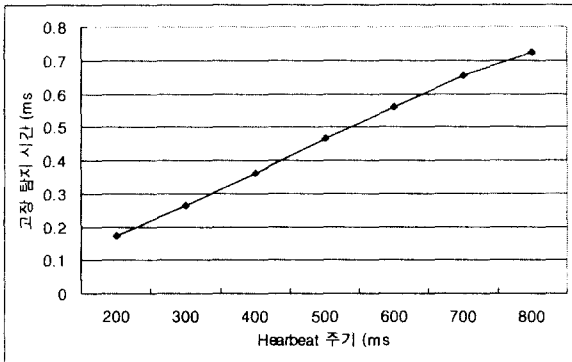
[그림 6] 고장 탐지 전파 시간

4. 실험 및 분석

본 논문에서 제안한 단순 고장탐지 기법을 위해서 Pentium III PC 2대 및 윈도우즈 2000 및 TMO 미들웨어[7] 상에서 실험하였다. 기타 실험에 사용된 매개변수는 다음과 같다.

노드 개수	8
SpM 활성화 주기	50ms
노드의 고장	SpM 활성화 횟수의 10%

[그림 5]는 각 노드에서 송신하는 heartbeat 시그널의 주기에 따라 실제 고장이 탐지되는 시간을 나타내고 있다.



[그림 5] 고장 탐지 시간

실험 결과에서는 heartbeat 시그널의 주기가 짧을수록 빠른 탐지시간이 나타남을 보이고 있다. 그러나 잦은 heartbeat 시그널의 송/수신은 각 노드에서의 부하를 발생시키게 되는 단점을 생각할 수 있다.

[그림 6]은 이웃 노드에 의하여 고장이 탐지된 사실이 감독자 노드를 통해 다른 노드에 전파되기까지의 시간을 나타내고 있다.

고장 사실이 다른 노드에 전파되는 시간은 신뢰성 있는 메시지 송수신에 있어서 중요한 의미를 갖는다. 왜냐하면 각 노드들이 이러한 시간만큼 다른 노드의 고장에 대해 고려하기만 한다면 전체 노드에 대한 고장이 없다고 가정할 수 있기 때문이다. 또한 이것은 메시지 송수신 시 고장 사실이 전파될 때 필요한 시간 안에 다른 노드의 고장 여부를 확인한다면, 모든 노드가 고장이 없는 상태에서 메시지 교환을 할 수 있다는 의미로 해석할 수 있다.

5. 결론

본 논문에서는 분산 실시간 시스템을 위한 고장 탐지 기법을 실험하고 그 결과를 분석하였다. 실험목적은 고장탐지 방법을 단순화하여 실시간 환경에서 사용 가능성을 확인하고자 하였다. 실험에서 고장 탐지 시간과 다른 노드들이 노드의 고장을 인지하는데 걸리는 시간을 측정하였으며 이것은 후에 통신 프로토콜 상에서 유용한 매개변수로 사용이 가능할 것으로 본다.

향후 과제로는 감독자의 고장 및 선출 알고리즘에 대한 실험과, 발생할 수 있는 고장의 종류를 다양화하여 보다 신뢰성 있는 실험을 해 보아야 할 것이다. 또한 실험결과를 토대로 하여 제안된 고장탐지기법을 실시간 그룹통신 프로토콜에 적용하여 실험할 계획이다.

참고문헌

- [1] K. H. Kim and Chittur Subbaraman, "Dynamic Configuration Management in Reliable Distributed Real-Time Information Systems", IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, 1999
- [2] K. H. Kim, and Subbaraman, C., "Fault-Tolerant Real-Time Objects", Communications of the ACM, Vol. 40, No. 1, January 1997, pp.75-82
- [3] K. H. Kim "Action-Level Fault Tolerance" Ch 17 in Sang H. Son ed., 'Advances in Real-Time Systems', Prentice Hall, 1994
- [4] K. H. Kim, et al., "Distinguishing Features and Potential Roles of the RTO.k Object Model", Proc. WORDS '94 (IEEE Computer Society Workshop on Object-oriented Real-Time Dependable Systems), Dana Point, Oct.1994, pp.36-45
- [5] K. H. Kim, "Object Structures for Real-Time Systems and Simulators", IEEE Computer, August 1997, pp.62-70
- [6] K. H. Kim and Chittur Subbaraman, "An Integration of the Primary-Shadow TMO Replication Scheme and its Recovery Time Bound Analysis", Proc. IEEE CS 17th Symp. on Reliable Distributed Systems (SRDS '98), West Lafayette, IN, 1998
- [7] K. H. Kim, Chittur Subbaraman, Masaki Ishida, Jiaqiang Liu, "TMO Support Library(TMOSL): Facilities for C++ TMO Programming", Univ. of California, Irvine, 2000