분산 실시간 시스템을 위한 가상 토폴로지에서의그룹연산

나성국[0] 홍영식
동국대학교 컴퓨터공학과
{nsg94,hongys}@dgu.edu

Group membership service in the virtual topology for Distributed Real-time Systems

Sung-guk Na[0] , Young-Sik Hong
Dept of Computer Engineering, Dongguk Univ.

요 약

실시간 분산처리 시스템에서의 통신은 신뢰성이 높아야 한다. 신뢰성이 높은 그룹통신 프로토콜에서 그룹 멤버쉽을 위한 뷰 관리는 매우 중요하다. 이 논문에서는 실시간 요소를 가진 시스템에서 LAN환경을 위한 가상 링 구조에서의 그룹연산 프로토콜과 LAN의 확장인 intranet상에서의 네트웍의 효율적 사용을 위한 트리구조상에서의 프로토콜을 제안하고 실시간 시뮬레이션을 통해 평가하고 결과를 분석하여 실시간성을 제공하기 위해 필요한 요소를 확인한다.

1. 서 론

분산환경에서 이루어지는 작업의 형태는 다수의 사용자들이 서로 자원을 공유하면서 동시에 이루어지는 복잡한 형태를 띄고 있다. 이러한 환경에서 프로세스간의 일관성있는 통신을 위해서 신뢰성있는 그룹통신을 위한 연구가 되어져왔다.
그룹통신은 그룹의 뷰(view)를 근거로 이루어지기 때문에 각 멤버가 유지하는 뷰의 일관성 보장이 신뢰성을 위해서는 가장 중요하게 다루어지는 부분이다.
본 논문에서는 각 작업에 시간적 제약이 주어지는 분산실시간 시스템에서 논리적 링구조를 가진 노드간의 뷰관리와 유지를 위한 그룹관리방법과 트리구조를 가진 시스템에서의 그룹연산을 위한 프로토콜과 그 성능을 시뮬레이션하였다
본 연구에서 실시한 실험은 분산 환경을 지원하는 미들웨어인 TMO (Time-triggered Message-triggered Object) [7]을 기반으로 하였다.

2. 관련연구

그룹통신에서의 그룹연산을 위한 연구로는 그룹뷰의 신뢰성을 위한 가상동기화 모델(virtual synchrony)를 주축으로 노드간의 partition을 고려한 확장된 가상동기모델과 뷰동기시간을 줄이기위한 연성가상동기모델, 긍정적가상동기모델등의 연구와 그룹의 확장성을 고려한 Client Server model이나 Layered model 등이 연구되었다. 또한 네트워크의 효율적 사용을 위해 노드간의 어플리케이션 계층의 토폴로지(Topology)를 구성하기 위한 연구가 이루어지고 있다. 대역폭문제가 심각하지 않은 LAN환경에서는 ISIS 등에서 신뢰성을 고려한 링구조의 토폴로지가 주로 사용되었고, WAN환경에서는 대역폭 문제를 고려한 다양한 형태의 노드간의 연결이 연구되어 왔는데 IP multicast에서의 재전송트리나 End system multicast[4]등에서는 Tree형태의 토폴로지를 기본 아이디어로 채택했다.[1)]

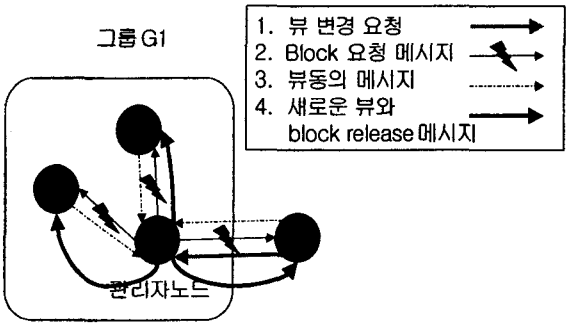
본 연구는 한국과학재단 특정기초 연구(R01-2000-000-00284-0)의 결과물임

3. 그룹연산

그룹통신에서의 그룹관련 연산은 각 멤버간의 동일한 뷰를 유지하는 것이 필수이다. 이를 보장하기위한 특성인 가상동기화를 위해서는 그룹 뷰를 수정하기전 뷰동기화중 문제를 막기위해 각 멤버 노드의 blocking과 뷰전달 확인을 위한 Ack가 필요하다. 본 절에서는 실시간 요소를 가진 LAN환경의 링구조와 확장된 환경인 Intranet환경에서의 신뢰성을 위한 가상동기화를 제공하는 그룹연산을 위한 프로토콜을 설명한다

3.1 논리적 링구조에서의 그룹연산

[그림 1]은 논리적 링구조에서 고장처리기를 가지는 실시간 시스템을 위한 그룹연산중 가입연산 과정을 보인다. 논리 링구조에서의 그룹통신은 각 노드별로 유지하고 있는 그룹뷰의 각 멤버로의 순차적 전송으로 이루어지기 때문에 각 노드의 그룹뷰를 일관성있게 보장하는 것이 가장 큰 문제이다. 일관성 유지를 위한 가상동기화를 보장하기위해서는 뷰(view)변경중에 일



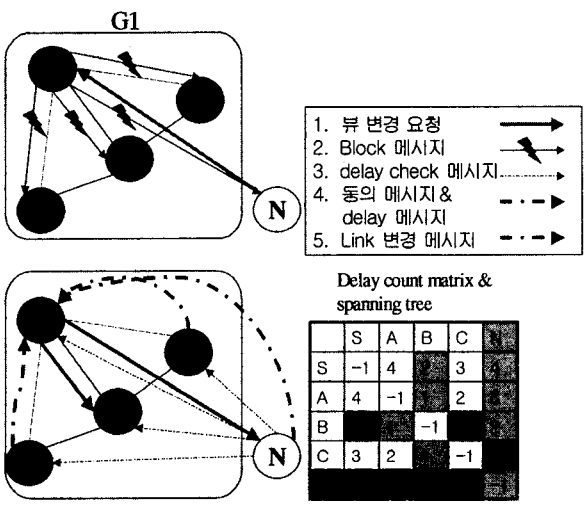
[그림 1] Join연산

어날 수 있는 간섭을 막기위해 blocking이 필요하고, 그룹관리자는 각 멤버간의 동의를 위한 Ack메시지를 각 멤버로부터 수신하게되면 release메시지 전송을 통해 뷰변경을 마친다. 실시

간환경의 신뢰성과 시간제약을 구현하기위해 본 논문에서는 ORT(official release time)[8]와 고장처리기를 사용하였다. 신뢰성있는 그룹통신을 위하여 메시지 전송자는 수신자로부터 Ack 메시지를 전송 받아야 한다. 여기에서는 메시지자체에 수신자의 큐에 유지할 시간(ORT)을 두어서 이 시간동안 고장처리기로부터 고장이 검출되지 않으면 메시지를 전달함으로써 신뢰성을 유지한다.

3.2 트리구조에서의 그룹연산

Intranet환경에서의 그룹연산은 노드간의 연결에서 대역폭문제가 있기 때문에 효율적인 토폴로지를 구성하는 것이 큰 문제이다. 그 동안 연구에서는 로컬그룹간에 유니캐스트 토폴로지를 구성하여 그 안에 효율적인 전송 경로를 구성하는 방식을 사용하였는데, 본 논문에서는 그룹안의 모든 멤버를 유니캐스트 토폴로지의 각 노드로 하여 노드사이의 IP 멀티캐스트의 라우터간의 트리를 구성하듯이 최소거리 신장트리를 구성하여 전송경로를 알리는 방식을 실험하였다.
[그림2]는 한 노드의 그룹 가입연산시의 과정을 보인것이다. 그룹연산 요청을 받은 관리자는 모든 멤버에게 blocking메시지를 전송하고 새로운 노드와의 거리를 delay check메시지를 통하여 계산하여 노드간의 거리값을 가진 행렬을 구한다. 관리자는 이 행렬을 통해 최소거리 신장트리를 계산한 후 그룹의 각 멤버간의 uplink와 downlink를 설정함으로써 가상 트리를 구성한다.



[그림 2] 트리구조에서 Join과정과 Delay Matrix

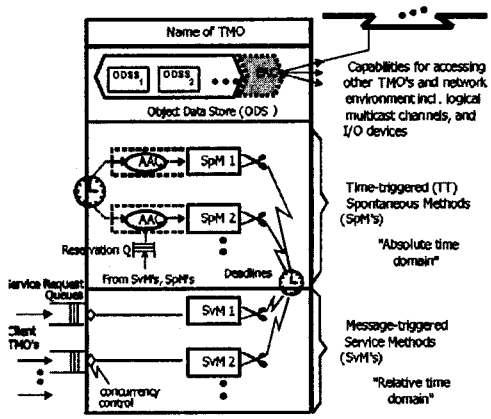
한 노드의 탈퇴는 가입연산에서 delay check과정을 생략하고 행렬에서 탈퇴 노드만 삭제한 후 최소 신장트리를 구성한다.
트리구조 안에서의 메시지 전송은 그룹연산과정에서 만들어진 가상트리에서 각 연결에 메시지를 전송함으로써 이루어진다.

3.3 고장처리

임의의 노드에서 고장이 발생했을 때 고장이 발생한 노드에 대해서 변경된 뷰나 그룹메시지의 ORT의 소진으로 Delivery되기 전에 검출되어야 신뢰성이 유지된다. 본 논문에서는 각 노드에서 uplink와 down link간에 heartbeat을 주고받음으로써 node의 고장여부를 확인한다.
주기적인 heartbeat의 count확인 과정에서 count가 부족할 경우 node의 고장으로 인지하고 고장을 인지한 노드는 그룹관리자에게 고장노드를 알려주게 되고 그룹관리자는 각 멤버에 취소(Abort) 메시지를 전달하여 메시지큐를 삭제하고 고장난 노드에 대한 뷰변경을 실시하여 이후 다른 멤버들의 오동작을 예방한다.

4. 실험 및 분석

본 논문에서 제안한 모델을 시뮬레이션하기 위해서 TMO(time-triggered message-triggered object) [7] 모델을 기본모델로 채택하였다. [그림4]와 같이 TMO모델은 세 가지 객체로 구성되어 있다. SpM(spontaneous method)이라 불리는 시간 구동 메소드는 주어진 시간 조건 AAC(automatic activation



[그림 4] TMO 시뮬레이션 모델

condition)가 만족되면 자동으로 호출되는 특징을 가지며, SvM(service method)은 외부로부터의 메시지 수신 등 이벤트에 반응하는 메소드 이다. 또한 ODSS(object data store)는 SpM과 SvM사이의 데이터공유 및 동기화의 역할을 수행하고 있다.
링구조의 실험은 노드 각 TMO객체 이름순으로 가상 ring을 미리 구성하여 실험을 하였고, 트리구조는 초기 link는 null로 시작하여 그룹참가시마다 노드간에 트리형태의 링크를 구성하였다. 트리구조 생성시 행렬을 위한 노드간의 time delay는 link에 random한 weight를 주어 링크간의 거리를 차별화 하여 실제 네트워크와 유사한 구성을 하였다.

Message	일반 multicast	80%
	Join message	10%
	Leave message	10%
HB signal		Every 1 sec
Fault		message 발생과 상관없이 random하게 node fault처리
node수		4,5,6개
ORT		200ms
실험시간		500sec

[표 1] 실험 파라미터

제안된 그룹관리기법을 실험하기위해 실험파라미터는 노드수를 4개에서 6개로 증가시키면서 실험하였고 메시지 크기는

200bytes ORT는 200ms로 실험하였다.

노드수	multicast		join		leave	
	발생횟수	평균처리시간	발생횟수	평균처리시간	발생횟수	평균처리시간
4	75	255	27	261	10	258
5	68	243	26	260	25	262
6	82	249	22	265	11	266

[표 2] Fully connected 링구조에서의 그룹연산실험

본 실험에서는 blocking과 ack사용시의 그룹변경 완료시간을 측정하였는데 [표 2]에서 보여주듯이 일반 multicast시간에 비해 10ms정도의 시간만이 더 지연되는 것으로 일관성유지를 위해서는 감수할 만한 결과를 보여준다.

노드수	multicast		join		leave	
	발생횟수	평균처리시간	발생횟수	평균처리시간	발생횟수	평균처리시간
4	84	255	21	298	8	259
5	69	254	25	305	12	262
6	72	261	27	303	10	254

[표 3] 가상 트리구조에서의 그룹연산실험

트리구조의 실험은 트리 구성과정의 지연시간과 트리를 사용한 메시지전송 시간을 측정하였다. [표 3]의 실험 결과를 통해 평균 전송시간은 크게 문제가 되지 않지만, 그룹변경시간이 기존 방식에서 보다 ORT시간을 제외하면 300%정도 지연됨을 볼 수 있다. 차후에는 그룹변경시 지연을 줄이기 위한 긍정적인 접근방안을 실험할 계획이다.

5. 결론 및 향후과제

본 논문에서는 분산 실시간 시스템에서 LAN상에서의 유용한 토폴로지인 논리링상에서의 그룹통신과 WAN이나 Intranet상에서의 그룹통신을 위한 그룹관리기법을 구성 실험하여 결과를 분석하였다. 실험을 통해 링구조나 트리구조에서의 그룹관리기법이 뷰의 일관성을 유지하는지와 뷰변경간의 지연시간이 얼마나 걸리는지를 확인하였다.

향후과제로는 논리 링구조의 효율적 구성을 위한 방법을 제안하고 실험에 적용하여 구성 시간대비 효율성을 실험할 계획이다. 트리구조에서는 트리구조 구성시 걸리는 지연시간을 단축하기 위한 기법으로 기존의 긍정적 가상동기화 모델을 적용시키는 방안을 구상중에 있다. 또한 논리 트리구성시 최악의 경우가 되는 경우를 방지하기 위해 노드간의 최장거리를 단축하는 기법이 요구되는데, shortest spanning tree에서 root노드를 검출하여 root노드로 부터의 최장거리에 있는 노드들의 링크를 조정하여 실시간 시스템의 시간제약을 위배하지 않도록 트리를 재조정하는 방법을 적용할 계획이다.

6. 참고문헌

[1] R. Viteberg, I.Keidar, G. V. Chockler, and D. Dolev. "Group Communication System Specifications : A comprehensive study. Technical report" Institute of Computer science, The Hebrew University of Jerusalem, 1999

[2] Jeremy Sussman, Idit Keidar, and Keith Marzullo, "Optimistic Virtual Synchrony.", In the 19th IEEE Symposium on Reliable Distrubted Systems, page 42-51, October 2000

[3] K.H KIM "Group Communication in Real-time computing systems : Issues and Directions" In the 7th IEEE 째가놰 on Futer Trends of DCS,page 252-258, Dec 1999

[4] Y. Chu, S. Rao, and H.Zhang. "A Case For EndSystem Multicast." In Proceedings of ACM Sigmetrics, Santaclara, CA, June 2000.

[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 with a Supervisor-based Network Surveillance Scheme and its Recovery Time Bound Analysis", Proc. IEEE CS 17th Symp. on Reliavle Distributed Systems (SRDS '98), West Lafayette, IN, 1998

[7] K. H. Kim, Chittur Subbaraman, Masaki Ishida, Jaqiang Liu, "TMO Support Library(TMOSL): Facilities for C++ TMO Programming", Univ. of California, Irvine, 2000

[8] Y.S. Hong, "Distributed object-oriented real-time simulation of the multicast protocol RFRM", Proc. of 7th IEEE int. Workshop on Object-Oriented Real-Time