멀티캐스트 전송을 위한 오류제어기법의 분류

Classification of Reliable Multicast Transport Protocols

고석주(S.J. Koh)	선행표준연구팀선임연구원
김용운(Y.W. Kim)	선행표준연구팀선임연구원
박정수(J.S. Park)	선행표준연구팀연구원
김광수(G.S. Kim)	선행표준연구팀선임연구원
김용진(Y.J. Kim)	선행표준연구팀선임연구원, 팀장

멀티캐스트 전송을 위한 오류제어기법에 대하여 많은 연구가 진행되어 왔다. 멀티캐스트 전송의 주요 고려사항은 신뢰성 및 확장성이다. 본 논문에서는 이 두 가지 문제를 해결하기 위해 제안되어 온 연구 결과를, 특히 오류제어기법 측면에서 정리 및 분류하는 데 초점을 둔다. 이러한 분류를 통해 향후 수송계층 프로토콜의 오류제어기법을 개발함에 있어서 고려해야 할 사항을 점검해 볼 수 있다.

Ⅰ. 서 론

신뢰성 프로토콜이란 통신 사용자에게 오류없는(error-free) 데이터 전송을 제공해주는 프로토콜을 말한다. 수송계층에서의 신뢰성은 주로 전송품질 저하로 인한 데이터 오류를 복구하는 오류제어기법에의해제공된다.

TCP는종단간의통신사용자사이에신뢰성을 제공하는 수송계층 프로토콜이다. TCP는 현재 유일한 신뢰성을 보장하는 유니캐스트 프로토콜로알려져있다. 사실상, 향후에도인터넷상의 유니캐스트통신의신뢰성보장을위해서대부분의 사용자는TCP를사용할것으로예상된다.

TCP의 오류제어방식은 ACK(acknowledgement) 메시지를 이용한다. 성공적으로수신된 데이터에 대하여 수신자는 ACK 메시지를 보낸다. ACK 메시지는대개다음번에받아야할패킷번호를 포함하며, 최근 메시지의 성공적인 수신을 의미한다. 패킷 번호 누락 등의 데이터 손실이 있을때, TCP는오류복구를위한재전송(retransmission) 모드로들어간다.

TCP는일종의송신자주도(sender-initiated) 오류제어방식이다. 즉, 데이터 전송 오류의 탐지 및 재전송 역할을 송신자가 책임진다. 하지만, TCP 오류제어방식을 멀티캐스트 환경에 그대로 적용하기에는 무리가 따른다. 즉, 송신자는 전송한 패킷에 대하여 각 수신자로부터 ACK 메시지를 받아야 하며, 이러한 메시지의 양은 수신자의 수가

많아질수록급격히증가한다.

이러한문제를 ACK 폭주(implosion) 문제라부르며, 수신자의 수가 많아질수록 문제는 더욱 심각하다. 즉, 송신자주도방식은확장성(scalability) 문제를 발생시킨다. 이러한 확장성 문제를 해결하기 위해 지난 10여 년 간 많은 대안들이 제안되어왔다. 하지만, TCP와는 달리, 멀티캐스트환경에서는" 단 하나의 보편적인(one-size-fits-all) 해법"은 개발되기 어려울 것으로 여겨진다. 신뢰성에 대하여 각 멀티캐스트 응용은 상이한 요구사항을 가지므로, 현재 상용화된 응용서비스만 고려하더라도, 적어도 몇 종류의 프로토콜은 있어야 할 것이다. 또한 아주 고유한 요구사항을 갖는 응용서비스는별도의신뢰성프로토콜을필요로 할지 모른다. 본 논문에서는 확장성 문제를 해결하기위해지금까지제안되어온오류제어기법들을각특징별로정리및분류하고자한다.

서문에 이어 제II장에서는 먼저 기존에 제안된 오류제어기법들을 분류 및 정리한다. 제III장에서는 제II장에서분류된방식중에가장대표적인프로토콜들을 중점적으로 살펴본다. 제IV장에서는 이러한 분류 및 정리를 토대로, 향후에 수송계층 프로토콜의오류제어기법개발시고려해야할점을살펴본다.

II. 오류제어기법의 분류

본 장에서는 현재까지 제안되어 온 신뢰성 프로토콜들의 오류제어기법들을 각 특징별로 분류한다. 관점에 따라 여러 가지 방법으로 분류할 수 있겠으나[1, 2], 본 글에서는 특히 송신자주도 오류제어기법의확장성문제를해결하기위해어떠한 방식이 사용되었는가에 초점을 두고 기존 방식들을다음과같이5가지로분류한다.

- 수신자주도(receiver-initiated)
- 토큰기반(token-based)
- 서버기반(server-based)로컬복구(local recovery)
- 트리기반(tree-based) 로컬복구(local recovery)
- 응용서비스기반(application-specific)

1. 수신자주도오류제어기법

송신자주도기법에서는 송신자 측에서 오류제어를책임지는것과는달리, 수신자주도기법은개별 수신자가 오류제어의 책임을 진다. 송신자는 수신자로부터NAK(Negative ACK)가올때까지 데이터 전송을 계속한다. 수신자로부터 NAK 메시지가 오는 경우, 송신자는 해당 데이터를 특정 수신자에게재전송한다.

각 수신자는 데이터 오류를탐지한 후, NAK 메시지를송신자에게유니캐스트로전송한다. 이 경우에 여러 수신자가 같은 NAK 메시지를 하나의 송신자에게 전송하는 문제가 발생할 수 있다. 이를 NAK 폭주라 한다. 이를 방지하기 위해 NAK 메시지를그룹전체에게멀티캐스트 하는 경우도 있다. NAK 메시지를 전체 그룹에게 멀티캐스트 하는 경우, 각 수신자는 일정 시간을 기다렸다가 NAK 메시지를 전송한다. 이는 다른수신자가 이미같은 NAK를 전송했을 수도있기때문이다. 이러한 방식으로 NAK 발생을 억제(suppression)하는메커니즘을" slotting & damping "방식이라부른다.

NAK 메시지를 멀티캐스트 하는 경우, 이는 다른 수신자에게 더 이상의 같은NAK를보내지 말라는 의미를 갖기도하며, 또한 주위의 수신자 중에서해당데이터를성공적으로수신한수신자가 있는경우에는자신에게데이터재전송을요구하

는 의미도 포함한다. 주위의 성공적인 수신자가 데이터 재전송을 하는 경우에도 역시 일정 시간을기다렸다가해당수신자에게데이터를전송하게된다. 역시또다른주위수신자의재전송을방지하기 위함이다. SRM(Scalable Reliable Multicast)[3] 방식에서는NAK 메시지전송과데이터재전송에서 모두 멀티캐스트 전송을 수행한다. 이 외에도 수신자 주도의 오류제어기법을 사용하는프로토콜로써는RAMP(Reliable Adaptive Multicast Protocol)[4] 및 MTP(Multicast Transport Protocol)[5] 등이있다.

하지만 NAK 방식 자체로써는 흐름제어(flow control) 측면에서 완전한 수송계층 프로토콜이 아니다. 송신자입장에서윈도크기(window size)를 제어하기 위해서는, 어떤 형태로든 ACK 정보를 받아야 하기 때문이다. 더욱이, 수신자로부터 ACK 메시지에 상응하는 정보가 도착해야 수신자가메모리를방출(release)할수있다. 이를위해서수신자가송신자에게주기적으로데이터수신성공을알리기 위해 주기적 폴링(periodic polling) 방식을사용하기도한다.

이처럼 수신자주도 오류제어방식에서는 모든 수신자가 오류제어의 책임을 진다. 이에 비해서 제2절에 기술될 토큰 기반 오류제어방식에서는, 토큰을가진사용자만이오류제어를담당한다.

2. 토큰 기반오류제어기법

토큰 기반 오류제어기법은 토큰을 가진 수신자만이오류제어기법을수행한다. 이러한방식의대표적인예는 RMP(Reliable Multicast Protocol)[6] 등이있다.

토큰기반오류제어방식에서는, 전체수신자그룹이 하나의 링(ring)으로 구성된다. 송신자의 데이터전송에대하여토큰을가진수신자만이송신자에게ACK 메시지를보낼수있다. ACK 메시지를 보낸 후에, 곧바로 다음 수신자에게 토큰을 건네준다. 데이터 손실을 경험한 수신자는 누구나 NAK를 송신자에게 전달할 수 있으나, 이는 흐름제어의일환으로송신자의윈도크기를감소시키기 위해 사용될 뿐, 데이터 재전송은 해당 수신자가 토큰을 얻게 되었을 때 한꺼번에 일어난다. 이처럼 토큰사용자만이 ACK를 전송하고, 손실된 데이터에대하여재전송을할수있다.

토큰기반오류제어기법은비록ACK/NAK 등의 제어 메시지를 줄일 수는 있으나, 실시간으로 오류복구를요구하는서비스에는적용하기어렵다.

3. 서버기반로컬복구기법

위의 방식에서는 송신자와 수신자가 직접적으로 오류제어에 참여했던 것과는 달리, 서버 기반 로컬복구방식에서는, 미리 선정된 하나 혹은 여러 개의 서버가 오류제어 및 데이터 재전송을 담당한다. 따라서 송신자의 처리부담을 여러 개의 서버로 분산시킬 수 있다. 이러한 방식을 사용하는 전형적인 프로토콜은 LBRM(Log-Based Receiver-reliable Multicast) 등이있다[7].

서버기반오류제어방식에서각 서버는 송신자의 모든 데이터를 기록하고 있다. 서버는 성공적인 데이터 수신에 대해 송신자에게 ACK 메시지를 보낸다. 수신자는 데이터 손실에 대해 송신자가 아닌 서버에게 NAK 메시지를 보낸다. NAK 메시지를 받은 서버는 서버에 저장해 놓은 데이터를이용하여해당수신자에게데이터를재전송한다.

이 방식은 여러 개의 서버를 두어 트래픽 부하를 분산시키는 장점이 있으나, 별도의 서버 선정

및 관리, 그리고 추가적인 메모리 할당 등의 오버헤드를요구한다.

4. 트리기반로컬복구기법

트리 기반 로컬복구방식에서는 전체 수신자 그룹을논리적인트리형태로구성하여, 각서브트리(sub-tree)에해당하는로컬그룹(local group)별로 오류제어를 담당하게 한다. 이로써 송신자의 처리 부담을 골고루분산시킬 수 있음은 물론, 많은 수신자에 대해서도 높은 확장성을 갖는다. 이러한트리기반오류제어기법을사용하는프로토콜에는TMTP(Tree-based Multicast Transport Protocol)[8] 및 RMTP(Reliable Multicast Transport Protocol)[9] 등이있다.

트리는 여러 개의 서브트리로 구성되며, 각 서브트리의 루트(root)가 오류 복구를 제어한다. 각 서브트리 안의 child는 또 다른 하위 서브트리의 루트가 될 수 있다. 이러한 제어용 트리의 구성은 새로운 멤버의 추가 혹은 기존 멤버의 탈퇴 등으로변경될수있다. 새로운멤버는" expanding ring search"(TTL 값을 확장해나가며 가장 가까운 루트를 탐색하는 방법) 기법을 이용하여 제어용 트리에가입한다.

각 child는 성공적인 데이터 수신의 경우 루트 혹은 parent에게 ACK 메시지를 보낸다. 이러한 측면에서 제어용 트리를 ACK 트리라고도 부른다. 각 서브트리의 루트는 자신이 관할하는 child들의데이터수신상태를취합하여상위루트에게 주기적으로전달한다.

데이터 손실의 경우 수신자는 자신의 parent에게 NAK를 유니캐스트로 알리거나(RMTP의 경우) 혹은 서브 트리 내에 멀티캐스트로 전달한다(TMTP의 경우). 데이터 재전송의 경우, 수신된 NAK 메시지 수에 따라 유니캐스트 혹은 멀티캐스트방식으로수행된다.

트리기반 오류제어기법은 매우 좋은확장성을 갖는다[2]. 하지만 네트워크 계층의 트리와는 별도의 제어용 트리를 구성 및 관리해야 하는 오버헤드가 요구된다. 또한 단일 세션이라 하더라도 송신자가여러명있는경우, 여러개의트리가구성되어야한다. 한편, 최근에제안된Lorax[10] 프로토콜에서는단일멀티캐스트세션에서여러송신자가 존재하는 " many-to-many" 서비스를 위해공유형(shared) ACK 트리방식을제안하였다.

5. Application-Specific 오류제어기법

모든 응용서비스에 적합한 오류제어기법을 개발하기란매우어려운일이다. 실제최근연구동향을 살펴보면, 특정 응용서비스 혹은 특정 목적에 적합한 오류제어기법을 개발하고자 하는 노력을 찾아볼 수 있다. 이에 해당하는 예로써는 MFTP(Multicast File Transfer Protocol)[11], STORM(Structure-Oriented Resilient Multicast)[12], PGM(Pretty Good Multicast 및 Pragmatic General Multicast)[13] 프로토콜, 그리고 ALF(Application Level Framing)[14] 접근방식등이다.

MFTP는 StarBurst사에의해개발되고있으며, 대용량파일전송서비스에적합한application-specific 멀티캐스트 프로토콜이다. 데이터 전송은 " pass" 단위로 이루어진다. 첫번째 pass에서 전체 파일이그룹사용자에게전달된다. 파일데이터를 성공적으로수신한멤버는송신자에게 ACK를 전송하며, 데이터 손실을 경험한 수신자는 NAK 메시지를전송한다. 두번째 pass에서, 송신자는손실된 데이터에 대하여 파일 전송을 수행한다. 역시 성공적인 데이터 수신자는 ACK를, 그렇지 않은 수신자는 NAK 메시지를 전송한다. 이러한 방식으로모든수신자가모든데이터를성공적으로 수

신할때까지데이터전송이반복된다.

STORM은 연속적인 매체서비스를 위한 프로토콜이다. 따라서 지연(delay)에 매우 민감한 서비스를 목표로 개발되었다. 전체 수신자들은 동적인(dynamic) 제어그래프(control graph)로구성된다. 트리기반방식과는 달리, 한 수신자가 여러 개의 parents를 지정하여 parent 노드들의 목록을 관리한다.

데이터손실의경우, 수신자는목록에있는 parent 노드들에게 차례대로 NAK 메시지를 전송한다. 첫번째 parent에게 응답이 없을 경우 그 다음 parent에게 NAK를 전송한다. 즉, 먼 거리에 위치한송신자에게직접적으로재전송을요청하는대신에, 가까이에 있는 여러 parents에게 데이터 재전송을 요청하는 방법이다. 따라서 그만큼의 재전송지연을단축시킬수있다.

PGM은 가장 최근에 제안된 프로토콜이다. 구현이 매우 간단하며 확장성이 매우 좋다. 트리기반방식이제어용트리를구성하여오류제어를수행한 것과는 달리, 네트워크 계층의 멀티캐스트 트리를 직접 오류제어에 사용한다. 따라서, 현행 인터넷 상에서 곧바로 구현될 수 있는 방식이다. PGM 프로토콜에 대한 상세한 기술은 다음 장에서이루어진다.

ALF(Application Level Framing)[14] 방식에서는다양한응용서비스의신뢰성을수송계층프로토콜에서 제공하기 어렵기 때문에, 각 응용 서비스 특성 및 요구사항에 맞도록 응용계층에서 해결하고자 하는 접근방식을 취하고 있다. ALF에서는 모든 데이터가“ application data" 단위로 교환된다. 각 데이터는 application-specific name space로 표현된다. 또한 각 응용 데이터는 신뢰성 요구에 따라 부호화(encoding)된다. 즉, 수송계층에서는 종단간의 데이터 전송만을 담당하고, 신뢰성제공은응용계층에서담당하고자하는방식이다.

Ⅲ. 주요 프로토콜의 오류제어기법

본장에서는지금까지논의되어 온오류제어기법 분류를 토대로, 각 분류별로 가장 대표적인 프로토콜에대하여보다상세히살펴보고자한다.

1. SRM

SRM(Scalable Reliable Multicast)[3]은 실제 현재의MBONE 상에서실험되고있으며, GlobalCast사(www.gcast.com)에 의해 개발되고 있다. 실제 MBONE의 whiteboard(wb) tool은현재SRM 프로토콜을 사용하고 있다. SRM의 목표는 다른 프로토콜과마찬가지로확장성이다. 또한기존의 TCP 동작원리를그대로수용하려고노력한다.

SRM은 수신자주도 오류제어방식을 사용한다. SRM에서그룹멤버들은, 다른수신자가데이터 손실을 경험할 때에, 오류 복구를 위해 서로 협력한다. 또한 그룹 멤버들간의 상호 협력을 통해 제어 메시지의폭주를방지하고자한다.

SRM은 오류 복구를 위해 단지 멀티캐스트 전송만을 사용한다. 그만큼 구현하기에 간단하다. SRM은 세션 메시지 교환을 통해 다른 멤버들과의 거리를 측정한다. 수신자가 데이터 손실을 탐지하면, NAK 메시지를 멀티캐스트로 전송하여 도움을 요청한다. 근처에 위치한 다른 수신자는 손실 수신자에게 도움을 줄 수 있다. 오류복구 요청 메시지의 범람을 방지하기 위해, 각 손실 수신자는 일정시간 동안 기다린 후 요청 메시지를 전송한다. 사실상이러한대기시간(waiting time)은 송신자와의거리에비례하게책정된다. 대기시간

동안에 다른 손실 수신자가 이미 요청 메시지를 보낸 경우 자신은 NAK 메시지를 전송하지 않는다. 이러한 메커니즘으로 요청 메시지를 억제하게 된다. SRM 구현을 위해 모든 수신자는 최근에 받은 일정 양의 데이터를 저장하고 있어야 한다.

데이터 손실에 상응하는 데이터 재전송도 NAK 전송과 비슷한 메커니즘으로 이루어진다. 데이터 수신에 성공한 수신자가 손실 수신자에게 데이터를 재전송하는 경우에도 일정 시간 기다린 다음 멀티캐스트 방식으로 데이터를 재전송한다. 만약 이미 다른 성공 수신자가 재전송을 한 사실을 알게 되면, 자신은 재전송을 하지 않는다. 이를 ' 재전송 억제'라 부른다.

모든 메시지가 항상 멀티캐스트로 전송된다는 점은 그룹 멤버들이 별도의 대상 지정 없이도 마음껏 협동할 수 있는 수단을 제공한다. 또한 적절한 대기시간 변수의 설정을 통해 중복되는 재전송 요청 및 재전송 트래픽을 억제할 수 있다. 하지만 모든 메시지를 멀티캐스트로 전송하기 때문에, 제어 메시지의 폭주 위험성도 그만큼 높다.

2. RMP

RMP (Reliable Multicast Protocol) [6]는 West Virginia 대학과 NASA가 연합하여 개발하는 중이며, SRM처럼 GlobalCast에 의하여 상용화되는 중이다. RMP는 토큰 기반 오류제어 메커니즘을 사용한다. 대개 수신자 그룹이 하나의 링 형태로 구성되어 토큰을 순환시킨다. 토큰을 가진 수신자가 전체 그룹에 대한 최근 패킷들을 제어한다.

성공적인 데이터 수신의 경우 토큰 사용자만 송신자에게 ACK 메시지를 보낸다. 토큰 사용자는 자신이 필요로 하는 모든 데이터를 성공적으로 수신한 다음, ACK 메시지를 보내고, 토큰을 다음 사용자에게 전달한다. 송신자는 ACK를 받을 때마다 윈도크기를 증가시킨다.

데이터 손실의 경우, 손실 수신자는 NAK 메시지를 전체 그룹에 멀티캐스트 방식으로 전송한다. RMP의 NAK 메시지는 단지 송신자의 윈도크기를 감소시키는 데 사용된다. 나중에 손실 사용자가 토큰을 가지게 되었을 때, 그동안 손실되었던 데이터를 송신자로부터 전달받는다.

RMP는 이처럼 토큰을 사용하여 그룹 사용자의 트래픽을 제어한다. 하지만 데이터 손실이 발생한 시점에서 적시에 (timely) 재전송하는 것이 아니라, 토큰을 가진 사용자만이 오류 복구를 하게 되므로, 실시간 및 대화형 서비스에는 적용하기 어렵다.

3. LBRM

LBRM (Log-Based Receiver-reliable Multicast) [7] 프로토콜은 서버 기반 오류제어를 수행한다. 여러 개의 logging 서버가 선정되며, 각 서버는 송신자가 보내는 데이터를 저장해 놓는다. 서버들끼리 계층 구조로 구성될 수도 있다. 서버는 성공적인 데이터 수신에 대하여 송신자에게 ACK 메시지를 전송한다.

한편 수신자는 데이터 오류를 탐지하는 경우, 송신자와 직접 연락을 취하는 대신에, 가까이에 위치하는 서버에게 NAK 메시지를 전송한다. 서버는 해당하는 데이터를 손실 수신자에게 유니캐스트 방식으로 재전송하든지, 아니면 로컬 그룹에게 멀티캐스트로 전송한다. 보통 로컬 그룹 내의 NAK 송신자가 많은 경우 멀티캐스트로, 반대의 경우에 유니캐스트로 데이터를 재전송한다.

이 방식은 전체 수신자 그룹을 로컬 그룹으로 조직화하여, 제어 메시지의 부하를 균등히 분산시키는 효과를 주지만, 별도의 서버를 관리 유지하고 또한 서버에 추가적인 메모리 할당이 요구

된다는단점이있다.

4. RMTP

RMTP[9]는 SRM, RMP와 함께 GlobalCast사의 연구진에 의해 개발되는 중이다. RMTP는 MBONE tunnel 환경에서 수 차례 실험되어 왔으며,' native multicast(non-tunnel)' 환경에서도 실험되고 있다. RMTP는 TMTP와 함께 트리 기반 오류제어를 수행하며, 높은 확장성을 갖는 것으로알려져있다.

RMTP는 전체 수신자 그룹을 트리 형태로 구성한다. 이러한 별도의 트리를 제어용 트리라 한다. 제어용 트리는 여러 개의 서브 트리로 구성된다. 이러한 트리는' expanding ring search' 기법을 이용하여구성된다.

RMTP는 각영역을 n-level의 계층구조로 나누어 신뢰성을 제공한다. 각 영역은 DR(Designated Receiver)를 선출한다. DR은 소속 영역의 수신자들이어떤패킷을성공적으로수신하였는지의여부에대한상태 정보를수집하여상위 DR에게 전달한다.

데이터 손실의 경우, 만약 DR이 손실 데이터를 보유하고 있다면, 곧바로 로컬 영역에게 재전송한다. 그렇지 않는 경우, 상위 DR에게 재전송 요청 메시지를 보낸다. 이러한 절차들이, 모든 수신자가 특정 window의 모든 데이터를 수신할 때까지, 각영역별로수행된다.

RMTP의 고유 특징은 계층적 구조를 사용한다는 점과, 상태 메시지를 영역별로 취합한다는 데 있다. 이러한특징으로인해송신자에게상태(status) 메시지가집중되는현상을방지할수있다. 또한 그룹 전반에 미치는 메시지의 수를 최소화할 수있다.

5. PGM

PGM(Pretty Good Multicast)[13] 프로토콜은 가장 최근에 제안된 reliable 멀티캐스트 프로토콜이며, Cisco사가주도적으로개발하고있다. IETF-RMRG(Reliable Multicast Research Group)에서 1998년2월에발표되었다.

PGM의 가장 큰 특징은 단순성과 뛰어난 확장성에있다. PGM은특히 실시간응용서비스에 적합한 방식이다. 또한 기존의 트리 기반 오류제어 방식과는 달리, 네트워크 계층의 멀티캐스트 트리를 그대로 수송계층의 오류제어에 사용한다. 한가지주목할것은 PGM의트리는, TMTP 혹은 RMTP에서사용되는로컬그룹제어을위한논리적 트리가아니라, 실제 멀티캐스트 전송 트리 그 자체라는점이다. 한편대부분의reliable 멀티캐스트 프로토콜이 UDP 위에서 동작하는 것과는 달리, PGM은 TCP, UDP 처럼 IP 위에서 동작하도록 설계되었다. 따라서 현재 인터넷 멀티캐스트 기술발전에가장근접한방식이라할수있다.

PGM에서사용되는패킷종류는다음과같다.

- ODATA(Original Content Data)
- NAK
- NCF(NAK Confirmation)
- RDATA(Retransmission Data)
- SPM(Source Path Message)

각 PGM 패킷은 해당데이터의 송신자 및 세션 정보파악을위해TSI(Transport Session Identifier)를포함한다. 따라서 PGM 라우터및 수신자들이 여러개의세션을쉽게구분할수있다. NAK 패킷만 수신자 측에서 송신자 측으로 전달되며(upstream), 나머지 패킷은 모두 downstream으로 전달된다.

PGM은 실시간 서비스 제공을 위해 " TW

(Transmit Window)'를 사용한다. 이것은 sliding window와 비슷한 것으로, 만약 TW동안 송신자가NAK 메시지를받지못하면, 해당데이터에대한복구는사실상불가능하게된다.

PGM은 전체적으로 NAK 기반 오류제어를 수행한다. 따라서 송신자에게 돌아오는 NAK의 수를 줄이는 것이 유일한 이슈가 된다. 여기에는 라우터의 도움이 필요하다. 먼저 데이터 손실을 탐지한 수신자는 가까운 PGM 라우터에게 NAK를 전송하고, NAK 메시지는 라우터에서 라우터로 멀티캐스트 트리를 따라 이동한다. 이러한 NAK 전달은NCF 혹은RDATA 메시지가발견될때까지계속된다. NCF와RDATA는이미해당메시지의재전송요청이다른수신자에의해이루어졌음을 의미한다. 이러한 경우 PGM 라우터들은 송신자에 이르는 upstream 경로를 따라 중복(duplicate) NAK 메시지를제거한다.

NAK메시지가송신자에게전달되는경로는송신자에서수신자에게 이르는 멀티캐스트 트리 경로와똑같아야한다. 이를위해ODATA 패킷전송시에SPM 메시지가삽입(interleave)되어각수신자에게 전달된다. PGM 라우터는 이 정보를 송신자로의 NAK 메시지 포워딩을 위해 사용한다. SPM은 또한 TW 안에 있는 가장 오래된 패킷이 곧 TW에서없어질 것이라는정보를각 수신자에게 알려주는 역할도 한다. 따라서 송신자는 적어도 TW가진척되는속도만큼으로 SPM을전송해야 한다(데이터패킷 하나당 적어도하나의 SPM 메시지가전송되어야한다).

PGM 라우터는 어디에서 NAK가 오는지를 알고있기때문에, RDATA의전송경로혹은전송포트를실제NAK 메시지출처로제한할수있다. 따라서불필요한재전송이일어나지않는다.

정리하면, PGM은일종의수신자주도방식의오류제어방식이다. PGM은네트워크계층의멀티캐스트트리를이용하여오류제어를수행한다. 제어용 트리의 구성 혹은 서버의 설치 등의 별도 오버헤드 없이 오류제어를 수행할 수 있다. 한편 실시간응용에부합하도록, TW 범위안에서만오류를복구한다. TW 이외의 데이터는 영구적인 손실로 간주한다. 결과적으로간단하면서도매우높은확장성을달성할수있다.

하지만 PGM은 오직 실시간 전송에 적합하고, 이에따라영구적인데이터손실이가능하다는장점이자 단점을 가진다. 또한 PGM의 구현을 위해서는, 네트워크 계층의 모든 라우터들이 PGM 고유의 패킷들을 처리할 수 있는(비록 요구되는 패킷종류가매우적다고해도) 능력이요구된다.

Ⅳ. 결론 및 향후 연구방향

지금까지기존에발표된 신뢰성 멀티캐스트프로토콜들의오류제어기법에대하여살펴보고, 그 특징에따라5가지로분류하였다. 송신자주도오류복구기법은분명히확장성에많은문제를지니고 있다. 확장성 문제를 해결하기 위해 수신자주도 혹은 토큰 기반 오류복구기법이 제안되었으며, 나아가전체수신자그룹을서버혹은트리기반으로그룹화하여로컬복구를추구하는기법들이 최근에 개발되고 있다. 한편, 응용서비스마다 다른 신뢰성 요구를 가진다는 점에 착안하여, 특정 응용서비스에 적합한 오류제어기법들에 대한 개발연구도한창진행중이다.

기존 오류제어기법에 대한 비교 · 분석결과를 토대로우리는다음과같은결론을얻을수있다.

- 첫째, NAK 기반 수신자주도방식이 ACK 기반 송신자주도방식보다는 더 좋은 확장성을 갖는다.

- 하지만, 흐름제어를 위해 어떤 형태로든 ACK 정보가 송신자에 전달되어야 한다. 이를 위한 방식으로주기적폴링방식이좋은대안으로여겨진다.
- 토큰 기반 오류제어방식은 토큰 사용자를 중심으로트래픽 부하를 분산시키는 효과를얻을 수 있지만, 근본적인 확장성 문제를 해결해주지는 않는다.
- 전체 수신자를 로컬그룹으로 조직화하여 오류 복구를 수행하는 서버 기반 혹은 트리기반방식은 수신자주도 혹은 토큰 기반 오류제어방식에 비하여더좋은확장성을제공한다.
- 모든 응용 서비스에 적합한 오류제어방식을 개발하기란매우어렵다. 따라서특정응용에적합한방식을개발하는것도좋은대안이다.

이러한분석을토대로, 확장성을위한오류제어기법으로는 트리 기반 로컬복구방식이 적합할 것으로 여겨진다. 또한 기본적으로는 NAK를 사용하며, 주기적인 폴링을 통해 송신자에게 ACK 정보를 전달할 필요가 있다. 하지만 PGM과 RMTP 방식의 비교에서 알 수 있듯이, 로컬 복구를 위한 트리로써, 제어용 논리적 트리와 실제 네트워크 계층의 멀티캐스트 트리 중에, 어느 것이 더 효율적인지는아직판단할수없다.

한편, 대부분의기존연구에서는QoS(Quality of Service) 및 그룹관리 측면에 대한 고려가부족한 편이다. 특히 응용서비스가 요구하는 지연시간 혹은 패킷 손실률 등의 정보가 오류 제어 메커니즘에 포함되어 있지 않다. 따라서 향후에 개발되는 신뢰성 멀티캐스트 프로토콜에서는, 이러한 QoS 및그룹관리정보를충분히반영한오류제어기법이제안되어야한다.

한편 ETRI에서개발되고있으며, 현재ITU-T SG7 및 ISO/IEC/JTC1/SC6 그룹에서표준화 진행중에 있는" 차세대 수송 계층 프로토콜 ECTP (Enhanced Communication Transport Protocol)"은 멀티캐스트 세션의 그룹관리 기능과 QoS 요구사항이 매우 잘 정의되어 있다. 따라서 ECTP 오류제어기법 연구에서는, 본 연구에서 얻은 결론과 함께QoS 요구사항측면이반영되어야한다.

참 고 문 헌

[1] K. Obraczka," Multicast Transport Protocols: A Survey and Taxonomy," *IEEE Communications Magazine*, Jan. 1998, pp. 94-102.

[2] Sridhar Pingali, Don Towsley, and James F. Kurose," A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols," *Performance Evaluation Reviews*, Vol. 22, 1994, pp. 221-230.

[3] S. Floyd *et al*.," A Reliable Multicast Framework for Lightweight Sessions and Application-Level Framing," *ACM SIGCOMM '95*, Oct. 1995, pp. 342-356.

[4] A. Koifman and S. Zabele," RAMP: A Reliable Adaptive Multicast Protocol," *INFOCOM '96*, Mar. 1996, pp. 1442-1451.

[5] C. Bormann *et al*.," MTP-2: Toward Achieving the S.E.R.O. Properties for Multicast Transport," *International Conference on Computer Communication and Networks '94*, Sept. 1994.

[6] Brian Whetten, Simon Kaplan, and Todd Montgomery," A High Performance Totally Ordered Multicast Protocol," *IEEE INFOCOM '95*, 1995.

[7] H. W. Holbrook, S.K. Singhal, and D.R. Cheriton," Log-Based Receiver-Reliable Multicast for Distributed Interactive Simulation," *ACM SIGCOMM '95*, Oct. 1995, pp. 328-341.

[8] R. Yavatkar, J. Griffioen, and M. Sudan," A Reliable Dissemination Protocol for Interactive Collaborative Applications," *Proceeding of ACM Multimedia '95*, 1995.

[9] J. C. Lin and S. Paul," RMTP: A Reliable Multicast Tran-

sport Protocol," *INFOCOM '96*, Mar. 1996, pp. 1414–1424.

[10] B. Levin, D. Lavo, and J. Garcia-Luna-Aceves, " The Case for Reliable Concurrent Multicasting Using Shared ACK Trees," *Proceeding of ACM Multimedia '96*, Nov. 1996, pp. 365–376.

[11] D*raft- miller-mftp-spec-03.txt, Starburst Multicast File Transfer Protocol(mftp) Specification*, IETF Internet Draft, Apr. 1998.

[12] X.R. Xu *et al.*," Resilient Multicast Support for Continuous-Media Applications," *Proceeding of NOSSDAV '97*, 1997.

[13] *Draft-speakman-pgm-spec-01.txt, PGM Reliable Transport Protocol Specification*, IETF Internet Draft, Jan. 1998.

[14] Z. Whang *et al*.," Framework for Reliable Multicast Application Design," *Proceeding of HIPPARCH '97*, 1997.

[15] *CD 14476, Enhanced Communications Transport Protocol*, ISO/IEC JTC/SC6, 1998.