

NACK 기반 신뢰적 멀티캐스트 프로토콜의 설계 및 평가

Design and Evaluation of NACK Based Reliable Multicast Protocol

정 선 화* 박 석 천**
Sun-Hwa Jung Seok-Cheon Park

요 약

멀티캐스트 기술을 인터넷과 같은 거대한 네트워크에 적용하기 위해서는 멀티캐스트 데이터그램을 수신자 그룹에게 효율적이고 신뢰성있게 전달하는 것이 필수적이다. 따라서 본 논문에서는 IETF를 중심으로 연구중인 RMT 프로토콜의 취약점을 제시하고 망의 혼잡으로 인해 재전송이 발생하는 경우 이를 개선하기 위해 구성요소의 기능을 재 설계함으로써 라우터에서 복구 및 재전송 처리를 할 수 있는 신뢰적 멀티캐스트 프로토콜을 설계하고 구현 및 평가하였다. 손실을 겪은 수신자가 보낸 재전송 요청을 지역으로 제한하고 멀티캐스트 라우터를 통해 손실을 복구하도록 하여 NACK 신호와 재전송이 지역 멀티캐스트 그룹으로 폭주하는 것을 방지하였다. 멀티캐스트 라우터에 버퍼링 기능을 가지도록 하여 재전송 요청시 이를 포워딩함으로써 재전송에 따른 지연 시간을 기존의 복구 방식보다 감소시킬 수 있도록 하였다. 이와 같은 기능에 적합한 신뢰적 멀티캐스트 프로토콜의 메시지와 동작절차를 설계하였으며, 이를 토대로 제안한 멀티캐스트 프로토콜을 유닉스 환경에서 C 언어를 사용하여 구현하고 시험하여 기존의 방식보다 우수한 성능을 나타냄을 확인하였다.

Abstract

Multicast protocols are developed in order to support group communications efficiently. However, there still exist some issues to be solved for deploy multicast protocol in the public Internet. This paper refers problem of existing Reliable Multicast Protocols and redesigns component function, and proposes enhanced reliable multicast transport protocol. This paper proposed a improved multicast transport scheme in NACK based reliable multicast. This scheme is much faster than by sender-initiated or receiver-initiated recovery and latency is smaller. Designed components are implemented in UNIX environment using C programming language. Then the protocol was evaluated performance through simulation. As the results, proposed protocol is better than existing protocols in both of transmission delay and packet loss. Especially, proposed protocol in this paper can be used in multicast services needed high reliability.

Key words : Reliable Multicast Protocol, RMT(Reliable Multicast Transport), NACK(Negative ACK)

1. 서 론

멀티캐스트 기술은 인터넷 사용자의 급속한 확산과 인터넷 서비스의 대중화로 인해 높은 부가 가치를 지니는 다양한 응용 서비스 보급을 더욱 촉진시킬 것으로 전망된다[1]. 이러한 인터넷 멀티캐스트 기술은 멀티캐스트 응용 서비스 자체와, TCP와 상응되는 수송계층에서의 멀티캐스트 신뢰성 제공 기술 그리고 네트워크 계층에서의 멀

티캐스트 라우팅 기술로 분류할 수 있다. 이 중 멀티캐스트 신뢰성 제공 기술은 멀티캐스트 데이터그램을 수신자 그룹에게 효율적이고 신뢰성 있게 전달하기 위한 방식으로 인터넷과 같은 거대한 네트워크에 멀티캐스트를 적용하기 위해서는 필수적인 기술이다. 유니캐스트 전송을 기반으로 하고 있는 기존의 인터넷에서는 이를 위하여 TCP (Transmission Control Protocol)를 사용하고 있지만, TCP에서와 같은 제어 방식을 멀티캐스트에 적용할 경우 재전송 요구의 폭주 및 불필요한 재전송으로 인한 망의 혼잡을 초래할 수 있다. 이러한 문제를 해결하기 위해 IETF (Internet Engineering

* 준 회 원 : 경원대학교 대학원 컴퓨터공학(박사과정)
sunny@connet.kyungwon.ac.kr(제1저자)
** 종신회원 : 경원대학교 컴퓨터공학과 정교수
scpark@kyungwon.ac.kr(공동저자)

Task Force)에서는 RMT (Reliable Multicast Transport) 워킹 그룹(WG)을 구성하여 신뢰적인 멀티캐스트 전송에 대한 연구를 진행중이지만, 멀티캐스트의 다양한 네트워크 및 수신자 특성 정보를 고려한 해법 제시가 어려워 아직까지 만족할 만한 그리고 대규모 인터넷에 쉽게 적용할 만한 해법은 제시되지 못하고 있다[2][10].

현재까지 RMT WG에서 연구되고 있는 방식들은 다양한 멀티캐스트 응용에 대한 요구조건에 따라서 ACK (Acknowledgment) 기반, NACK (Negative ACK) 기반, FEC (Forward Error Correction)를 사용하는 ALC (Asynchronous Layered Coding) 프로토콜의 3가지 클래스로 구분되며, 신뢰성 제공을 위해 종단간 제어 방식을 중심으로 문제를 해결하고 있다. 그러나 망의 전송 기술이 향상됨에 따라 대부분의 패킷 손실은 망의 혼잡에 기인하는데 이들 방법 중 대부분은 종단 호스트에서 문제를 해결하고자 함으로써 오히려 망의 혼잡을 가중시키는 결과를 초래할 수 있다[3][4].

따라서 재전송의 폭주를 방지하고 망의 혼잡을 감소시킴으로써 보다 향상된 성능을 제공할 수 있는 신뢰적 멀티캐스트 프로토콜 구조의 고안이 필요하다. 본 논문에서는 이를 위하여 IETF를 중심으로 연구중인 RMT 프로토콜을 분석하여 문제점을 제시하고 구성요소의 기능을 재설계함으로써 망의 혼잡으로 인해 재전송이 발생하는 경우 라우터에서 복구 및 재전송을 처리할 수 있는 신뢰적 멀티캐스트 전송 프로토콜을 설계하고 구현하였다. 손실을 겪은 수신자가 보낸 재전송 요청을 지역으로 제한하고 멀티캐스트 라우터를 통해 손실을 복구하도록 하여 NACK 신호와 재전송이 전역 멀티캐스트 그룹으로 폭주하는 것을 방지하였다. 또한 제안한 방식은 멀티캐스트 라우터에 버퍼링 기능을 가지도록 하여 재전송 요청시 이를 포워딩함으로써 재전송에 따른 지연 시간을 기존의 송신자 중심(sender-initiated)이나 수신자 중심(receiver-initiated)의 복구 방식보다 감소시킬 수 있도록 하였다[5]. 이와 같은 기능에 적합한 신뢰

적 멀티캐스트 프로토콜의 메시지와 동작 절차를 설계하였으며, 이를 토대로 프로토콜을 유닉스 환경에서 C 언어를 사용하여 구현하고 혼잡 상황 발생시 재전송에 대한 성능 분석을 수행하였다.

2. 신뢰성 제공을 위한 멀티캐스팅 기술

인터넷 멀티캐스팅 기술은 네트워크 계층에서의 멀티캐스트 라우팅 기술과 수송계층에서의 신뢰성 제공기술로 분류할 수 있다. 멀티캐스트 라우팅은 인터넷 데이터 패킷(packet)이 목적지에 도달하기까지 경유하게 되는 전송 경로를 설정하는 메커니즘을 의미한다. 수송계층에서의 신뢰성 제공기술이란 기존 TCP 처럼 오류제어 및 혼잡제어를 통해 멀티캐스트 종단 사용자간의 데이터 전송의 신뢰성을 제공하는 기술이다[6].

본 절에서는 현재 IETF를 중심으로 연구가 계속 진행중인 멀티캐스트 신뢰 전송 프로토콜의 구조와 동작 절차를 분석하고 멀티캐스트 전송 방식에서 신뢰성을 제공하기 위한 요소 기술을 분석한다.

2.1 RMT 관련기술

RMT 제어 메커니즘은 오류제어, 흐름제어 및 혼잡제어 기술로 나눌 수 있다.

오류제어 기술은 수신자의 입장에서 손실된 멀티캐스트 전송 패킷을 복구하는 절차를 의미하며, 수신자가 패킷의 손실 여부를 탐지하는 오류탐지, 송신자 혹은 다른 수신자에게 손실된 패킷의 재전송을 요구하는 재전송 요구, 그리고 이를 토대로 손실된 패킷을 다시 전송하는 재전송 기능으로 나누어 볼 수 있다. TCP와는 달리 멀티캐스트에서 신뢰성 제공이 어려운 이유는 여러 수신자들이 서비스에 참여함으로써 손실 패킷의 재전송 요구시 많은 수신자가 동시에 송신자에게 재전송 요구 패킷을 보내는 경우, ACK 혹은 NACK 폭주 문제가 발생할 수 있기 때문이다.

흐름제어는 멀티캐스트 수신자의 버퍼 혹은 패킷 처리능력을 초과하지 않도록 송신자의 송신속도를 제어하는 기법이다. TCP와는 달리 멀티캐스트 환경에서는 여러 수신자를 고려해야 하므로, 어느 수신자의 버퍼용량을 그룹의 대표값으로 정해야 할지 등의 문제를 고려해야 한다[7][8].

혼잡제어는 송신자의 트래픽이 네트워크에 과부하되지 않도록 규정한다. 특히 하나의 패킷이 여러 수신자에게 전달되는 멀티캐스트 전송에서는 혼잡제어의 중요성이 더욱 커진다. 또한 네트워크가 혼잡상태에 있을 때 송신자의 트래픽 발생을 줄여서 데이터 손실을 방지하도록 하며, 불필요한 트래픽 발생을 억제하는 역할을 한다.

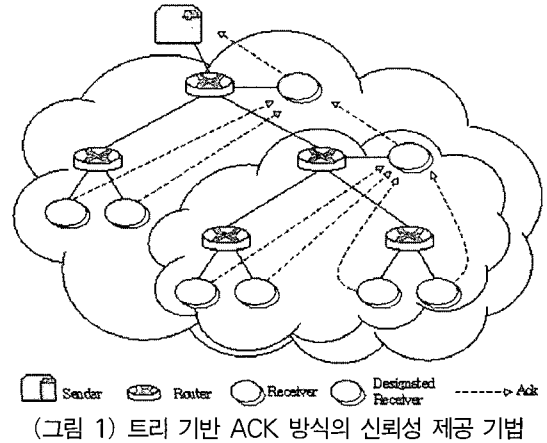
2.2 신뢰성 제공을 위한 주요 RMT 프로토콜

멀티캐스트 전송의 신뢰성 제공을 위한 제어기술과 기술개발 관련 이슈를 해결하기 위해 다양한 방식들이 제안되어 왔으며, 이 중 현재까지 알려진 주요 RMT 기법들은 다음과 같이 분류할 수 있다.

(1) 트리 기반 ACK 방식

멀티캐스트 사용자를 연결하는 수송계층의 논리적 트리를 구성하고, 트리의 계층적 구조를 이용하여 오류를 복구하는 기법이다. 트리를 통해 각 수신자들은 부모-자식 관계를 형성할 수 있으며, 각 부모 노드들은 자식노드들에 대한 오류복구 기능을 제공한다. 부모 노드가 재전송을 요구하는 패킷을 가지고 있지 않은 경우에는 상위 부모 노드에게 재전송을 요청하여 결국 데이터 송신자에게 재전송 요구가 전달될 수 있다.

그림 1은 수송계층에서의 논리적 트리에 대한 예를 보여주고 있다. 그림에서 각 부모 노드는 DR (Designated Receiver)로써 표현되며 각각의 자식 노드로부터 ACK 패킷 및 손실정보를 받은 뒤 오류를 복구하고 자신의 서브 그룹에 대한 데이터 송수신 상태 정보를 상위 부모 노드에게 전달



(그림 1) 트리 기반 ACK 방식의 신뢰성 제공 기법

한다. 이러한 상태 정보에 입각하여 송신자는 흐름 및 혼잡제어를 수행하게 된다.

트리기반 오류제어에서는 이처럼 수많은 수신자들을 하나의 트리 구조로 연결하여 확장성 문제를 해결하고자 한다. 하지만 전체 그룹을 하나의 트리로 구성하는 방법에 대한 효과적인 해법이 제시되지 못하고 있으며, 특히 어떤 수신자를 부모노드로 정할 것인지 등의 이슈가 해결되지 못하고 있다. 또한 망의 혼잡제어를 위해 각 수신자의 정보를 송신자에게 보내게 되는 경우 해당 제어 패킷이 트리를 따라 송신자에게 전달되는 동안 일정시간의 지연이 발생한다는 문제점도 지니고 있다. 이 방식의 대표적인 프로토콜로는 RMTP (Reliable Multicast Transport Protocol), TRAM (Tree based Reliable Multicast) 프로토콜 등이 있다[3].

(2) NACK 기반 신뢰성 제공 방식

오류 재전송을 요구하는 수신자는 NACK 패킷을 전체 그룹에 전송하며 가까이 있는 성공적인 수신자가 있을 경우, 이러한 NACK 패킷에 응답한다. 이 경우 같은 NACK 패킷을 여러 수신자들이 동시에 발생시켜 NACK 패킷이 폭주할 우려가 있으므로, 각 수신자는 타이머(timer)를 이용하여 적절한 시간동안 다른 NACK 패킷이 이미 발생되었는지를 파악해야 하며, 이러한 기법을 NACK 억제 기법이라 한다.

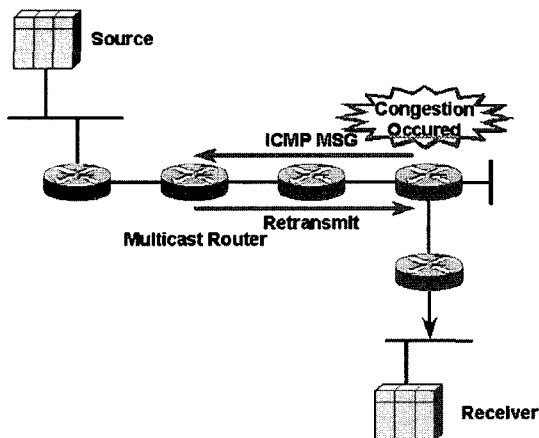
NACK 기반 오류제어에서는 오류복구 기능을 송신자가 아닌 가까이 있는 수신자의 도움으로 해결하여 확장성을 높이고자 한다. 이 방식에서는 특히 NACK 억제를 위해 사용되는 타이머의 동작이 전체 성능에 영향을 주며 특히 모든 수신자에게 멀티캐스트 송신 능력을 요구한다. 또한 ACK 패킷의 기능 중의 하나인 송신버퍼의 방출(release 혹은 flush)기능이 없다는 문제점도 지니고 있다. 이 방식의 대표적인 프로토콜로는 SRM(Scalable Reliable Multicast) 방식이 있다[9].

3. 신뢰성 향상을 위한 NACK 기반 멀티캐스트 프로토콜의 설계

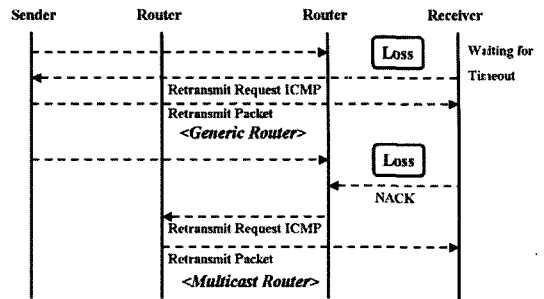
3.1 멀티캐스트 프로토콜의 동작 절차 설계

본 논문에서 제안한 프로토콜은 그림 2와 같은 동작 원리로 손실된 패킷을 복구함으로써 멀티캐스트 전송에 있어 보다 향상된 신뢰성을 제공할 수 있다. 송신자로부터 전송되는 데이터들이 망의 중간 라우터에 대량 집중됨으로써 혼잡이 발생하게 되면 혼잡의 발생 때문에 손실되는 패킷들은 송신자로부터 재전송하여 복구하지 않고 멀티캐스트 라우터에서 바로 복구를 할 수가 있다.

복구 과정은 라우터에서 혼잡으로 인하여 손실



(그림 2) 제안한 Multicast Protocol의 동작 절차



(그림 3) 일반적인 RMT protocol과 제한한 RMT protocol의 비교

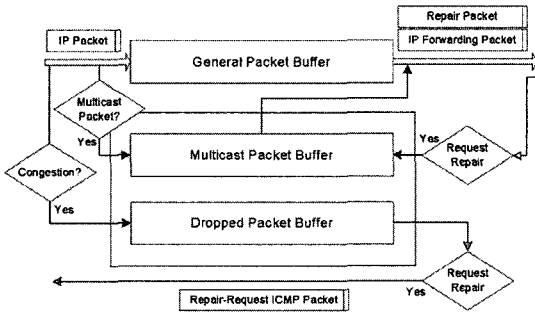
이 발생되면 손실되는 패킷의 IP 헤더부분을 저장한다. 혼잡이 해제된 후에 이전 홉의 라우터에게 ICMP 메시지를 전송하여 패킷의 재전송을 요청한다. 재전송을 통해 패킷을 복구한 후에 다시 멀티캐스팅을 수행한다. 그림 3은 기존의 멀티캐스트 라우팅 방식과의 차이점을 나타낸다.

이러한 과정을 수행하기위하여 멀티캐스트 라우터에서는 멀티캐스트 패킷을 저장하기 위한 선형 버퍼와 손실된 패킷의 IP 헤더 부분만을 저장하는 버퍼가 추가된다. 또한 멀티캐스트 라우터가 IP 데이터그램의 옵션필드에 자신의 IP 주소를 저장할 수 있도록 하여 지나올 때마다 이 필드의 주소를 업데이트 한다. 이것은 재전송 요청시 바로 이전의 멀티캐스트 라우터 주소를 목적지 주소로 하여 재전송 요청 메시지를 보낼 수 있게 한다.

3.2 멀티캐스트 프로토콜의 기능 구조 설계

라우터에서 신뢰성 있는 멀티캐스트 전송을 하기 위해서는 버퍼 관리기능, 손실 패킷 관리기능, 그리고 ICMP 처리기능이 있어야 한다. 제안한 멀티캐스트 프로토콜에서는 현재 버퍼의 혼잡 상태를 파악하고, 선형 버퍼에는 멀티캐스트 패킷을 저장한다. 혼잡이 발생하면 손실되는 패킷들의 헤더부분은 손실 패킷 버퍼에 링크드 리스트 형태로 삽입된다.

IP 헤더는 목적지 주소, 송신자 주소 등의 필드를 포함하고 있으므로 패킷을 구분할 수 있는

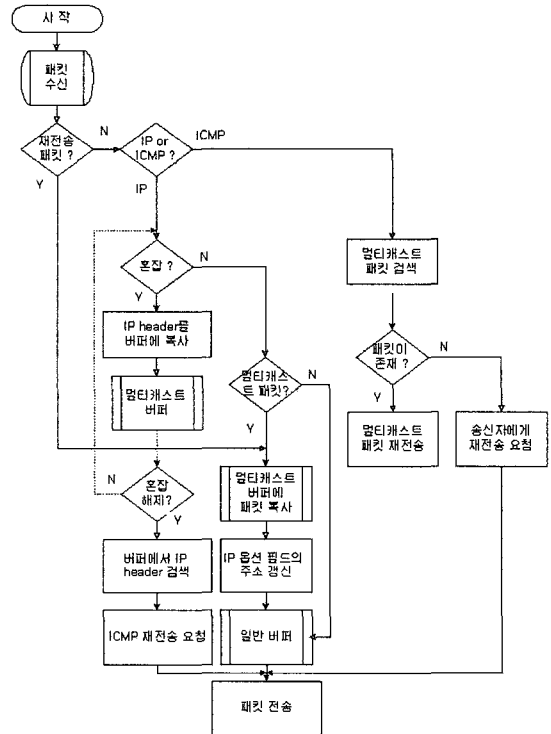


(그림 4) Multicast Router ICMP Handling 모델

```

receiveICMP()
if requestRepair
    search DroppedPacketList
    generate ResendICMP
    send ResendICMP
elseif requestRetransmit
    search MulticastPacketBuffer
    if requestPacket is exist
        reransmitPacket
    else
        requestRepair
    
```

(그림 5) Multicast Router ICMP Handling 알고리즘



(그림 6) 멀티캐스트 프로토콜의 전체 동작 알고리즘

정보를 갖고 있다. 나중에 혼잡이 해제되면 이전 홉의 라우터에게 IP 헤더를 포함한 ICMP 메시지를 송신해서 재전송 요구를 한다. 그림 4는 제안한 멀티캐스트 프로토콜의 처리 절차를 나타내며, 재전송을 요청할 때 ICMP 패키지를 생성하고, 다른 라우터로부터 재전송을 요청받았을 때 ICMP 메시지에 포함된 헤더의 송신자 주소, 목적지 그룹 주소가 일치하는 패킷을 검색하여 전송하는 기능을 담당한다. 그림 5는 Multicast Router ICMP Handling 알고리즘을 나타낸 것이다.

멀티캐스트 라우터에서 전송의 성능을 향상시키기 위해 위에서 설계한 기능 구조에 따른 멀티캐스트 프로토콜의 전체 동작 알고리즘은 그림 6과 같다.

3.3 멀티캐스트 프로토콜의 메시지 형식 정의

멀티캐스트 전송의 성능 향상을 위해 필요한

(표 1) IP Datagram Router Option 메시지의 필드 구성

필드	내용
Code	물리 네트워크의 제한 등에 따라 IP 패킷을 분할하여 보낼 경우 사용되는 필드로 라우터에 의해 지정된 IP 패킷의 고유 식별 번호(Identification)를 표시
Length	8비트의 부호없는 정수로 IP 패킷의 길이를 octet 단위로 표현
MR IP Addr.	멀티캐스트 패킷 전송 시 라우터에 의존하는 처리를 하기 위한 경로 기록으로 경우유하는 멀티캐스트 라우터의 주소를 지정

메시지 형식을 IP Datagram Router Option, ICMP Repair Request, Retransmission Datagram Option의 세 가지 필드로 정의하였으며 이 메시지 필드는 수신자와 멀티캐스트 라우터 또는 멀티캐스트 라우터 간의 통신을 통해 패킷 재전송 요청과 복구를 처리하기 위해 사용된다. 각각을 나타내면 표 1, 표 2, 표 3와 같다.

(표 2) ICMP Repair Request 메시지의 필드 구성

필드	내용
Code	Code 0 : 수신자의 재전송 요청 Code 1 : 멀티캐스트 라우터의 복구 요청
Length	8비트의 부호없는 정수로 IP 패킷의 길이를 octet 단위로 표현
Checksum	IP 프로토콜 헤더 자신의 내용이 바르게 되었는지를 검출하는 필드
Identifier	물리 네트워크의 제한 등에 따라 IP 패킷을 분할하여 보낼 경우 사용되는 필드로 IP 패킷의 고유 식별 번호(Identification)
Sequence No.	순서 번호
IP Header	멀티캐스트 라우터에 의한 복구 요청시 버퍼에 저장된 손실된 패킷의 IP Header

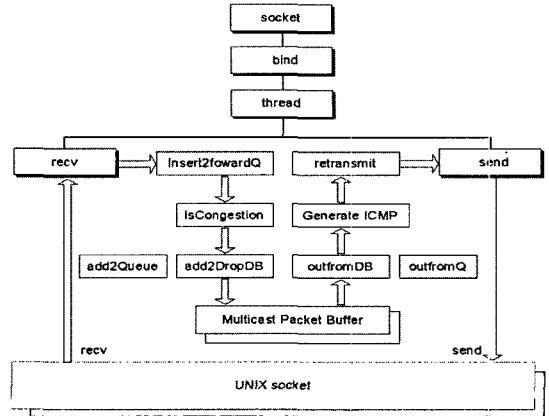
(표 3) Retransmission Datagram Option 메시지의 필드구성

필드	내용
Code	물리 네트워크의 제한 등에 따라 IP 패킷을 분할하여 보낼 경우 사용되는 필드로 라우터에 의해 지정된 IP 패킷의 고유 식별 번호(Identification)를 표시
Length	8비트의 부호없는 정수로 IP 패킷의 길이를 octet 단위로 표현
IP Address	멀티캐스트 패킷 재전송 시 라우터의 복구 요청에 대한 응답일 경우 요청 라우터의 유니캐스트 주소가 지정되며, 수신자의 재전송 요청일 경우 수신자가 속해 있는 멀티캐스트 그룹 주소를 지정

4. 멀티캐스트 프로토콜의 구현

4.1 멀티캐스트 프로토콜의 구현 모델 및 프리미티브

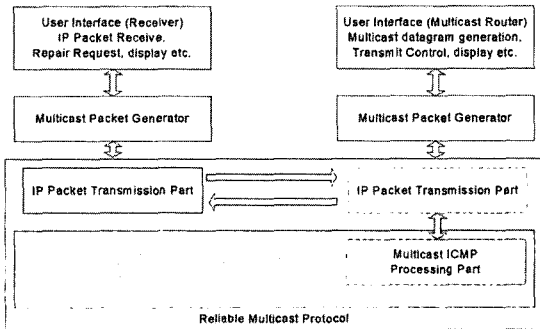
멀티캐스트 프로토콜의 구현을 위하여 유닉스 상에서 소켓을 사용하여 각 프로세스간에 port를 할당한 후, 송신과 수신 부분의 쓰레드를 생성하였다. 두 쓰레드는 패킷 버퍼를 공유하여 사용하며, recv 부분은 수신 패킷을 버퍼에 삽입하고 send 부분은 수신되어 버퍼링된 패킷에 대한 전송을 담당하는 역할을 한다. 멀티캐스트 프로토콜의 전체적인 구현 모델은 그림 7과 같고, 구현한 프리미티브의 각 기능은 표 4와 같다.



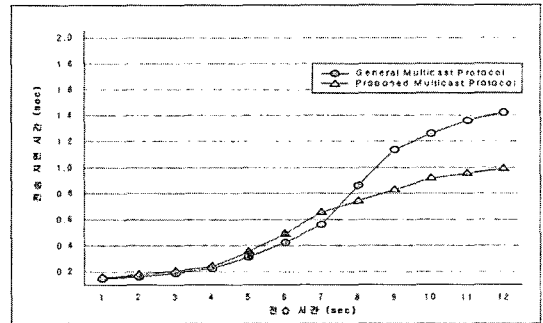
(그림 7) 멀티캐스트 프로토콜의 전체적인 구현 모델

(표 4) 구현 함수의 프리미티브

함수	설명
add2Queue	파라미터로 수신된 패킷이 들어가게 되며, 이 패킷이 멀티캐스트 패킷인지를 가리키는 플래그가 들어감으로써 QoS의 지원에 있어서 일반 데이터와의 차별성을 둔다.
outfromQueue	현재 큐에 있는 데이터 중 큐의 head가 가리키는 곳에 있는 패킷을 리턴한다. 수신 후 포워딩 모듈이 전송할 때 큐에 버퍼링된 패킷을 꺼내는데 사용한다.
add2DropDB	혼잡으로 인한 패킷의 손실 발생 시 손실 패킷 리스트에 손실 처리되는 패킷의 IP 헤더가 들어가게 된다. 파라미터로는 손실 패킷이 IP 헤더를 인자로 받으며, 멀티캐스트 패킷에 대해서만 동작한다.
outfromdropDB	혼잡 해제 시 상위의 라우터에게 재전송을 요청하는 멀티캐스트 ICMP 메시지를 발생시키기 위해 손실 처리된 패킷의 IP 헤더를 얻어낸다.
isCongestion	패킷 버퍼의 상태를 검사하여 설정한 혼잡상태의 값보다 클 경우 현재의 상태를 혼잡상태로 설정한다.
recv	송신자로부터 패킷을 수신받는다.
send	패킷 버퍼의 수신자 그룹 주소를 검사하여 패킷을 포워딩 한다.
Generate Resend ICMP	멀티캐스트 ICMP 패킷을 생성하는 역할을 하며, outfromDropDB의 수행을 통해 이전에 손실 처리된 패킷의 IP 헤더를 받아들여 멀티캐스트 ICMP 패킷을 생성한다.
Retransmit	멀티캐스트 ICMP 패킷을 수신하였을 경우, 자신의 버퍼를 검색하여 IP 헤더의 필드 중 주소 필드와 sequence 번호가 일치하는 패킷을 찾아내어 다시 전송하는 기능을 한다.



(그림 8) 실험 시스템 구성도



(그림 9) 혼잡상황 발생에 따른 패킷의 전송 지연

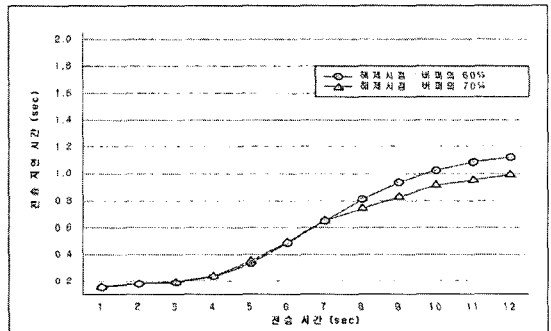
4.2 멀티캐스트 프로토콜의 시험

NACK 기반의 에러제어 방식을 수용하면서 멀티캐스트 라우터에서 재전송 요청을 처리할 수 있도록 성능을 제한한 멀티캐스트 프로토콜을 유닉스 환경에서 C언어로 구현하고 시험 모델을 구성하여 성능을 분석하였다.

본 구현에서는 오류검출 능력에 가장 뛰어난 방법인 국부 시험 방법을 이용하여 구현 프로그램의 적합성 시험을 수행하였고, 또한 실제 멀티캐스트 패킷을 주고 받는 모의 시험을 통하여 구현한 멀티캐스트 프로토콜의 동작 기능을 확인하였다. 실험 모델 구성은 그림 8과 같다.

4.3 멀티캐스트 프로토콜의 성능 분석

혼잡 상황에 따른 멀티캐스트 프로토콜의 성능 분석을 위하여 멀티캐스트 패킷이 경유하는 경로에 임의의 혼잡을 발생시켜 수신측에서 패킷의 평균 전송 지연 시간을 측정하여 평가하였다. 각 노드간의 전송속도는 25ms에서 전송노드의 가변적인 특성을 반영하기 위하여 30%의 가감을 두었으며, 멀티캐스트 패킷 버퍼는 최대 1,000개의 패킷을 저장할 수 있도록 하였다. 혼잡상황 발생에 따른 멀티캐스트 패킷의 전송 지연 시간 변화와 혼잡상황의 해제 시점에 따른 멀티캐스트 프로토콜의 패킷 전송 지연시간을 그림 9와 그림 10에 각각 나타내었다.



(그림 10) 해제 시점에 따른 패킷의 전송 지연

위의 결과를 보면 제안한 방안은 멀티 캐스트 라우터에서 버퍼링을 통해 라우터의 혼잡에 따른 데이터의 손실을 방지하고 패킷의 재전송 및 복구 경로를 단축시킴으로서 전송 지연시간을 감소시킬 것이며 재전송에 따른 제어 메시지 및 트래픽을 지역으로 한정시킴으로써 폭주 발생의 위험을 줄여 신뢰적 멀티캐스트 프로토콜의 성능을 향상시킬 것으로 사료된다

5. 결 론

현재 멀티캐스트 서비스에 대한 요구사항은 많은 반면, 수송계층에서의 멀티캐스트 신뢰성 제공 기술은 확장성 문제, 혼잡제어 등으로 인해 효과적인 해법을 찾지 못하고 있는 실정이다.

따라서 본 논문에서는 일시적인 혼잡으로 인하여 손실되는 패킷을 인접 라우터들 사이의 간단한 동작으로 복구함으로써 End-to-End 방법으로

인한 복구 지연을 줄이고, 폭주 발생의 위험을 줄임으로써 멀티캐스트 전송의 신뢰성을 향상시킬 수 있는 NACK 기반의 멀티캐스트 프로토콜을 제안하였다.

구현을 위하여 유닉스 환경에서 C언어를 사용하였고 설계한 동작 절차에 따라 멀티캐스트 패킷의 전송 및 복구가 정상적으로 이루어짐을 확인하였다. 멀티캐스트 프로토콜의 신뢰 전송에 대한 성능 분석을 수행하기 위하여 네트워크 시험 모델을 가상으로 구성하고, 파라미터 값을 다양하게 하여 전송 지연 시간과 패킷 처리율에 대한 성능을 분석한 결과 기존의 방식보다 우수한 성능을 나타냄을 확인하였다. 본 논문에서 설계 및 구현한 NACK 기반의 신뢰적 멀티캐스트 프로토콜의 성능 향상에 관한 연구는 향후 다양한 멀티캐스트 응용의 요구에 대해 높은 신뢰성을 제공하는 멀티캐스트 전송기술로 활용될 수 있을 것으로 사료된다.

참 고 문 헌

[1] 이동만, 윤원용, "Survey on Reliable Multicast over the Internet", 정보처리학회지, 제8권 제5

호, Sep. 2001.

- [2] 고석주, 강신각, "멀티캐스트 신뢰전송 기술 및 표준화 동향", 주간 기술동향, ETRI, May 2000.
- [3] Katia Obraczka, "Multicast Transport Protocols : A Survey and Taxonomy," *IEEE Communications Magazine*, Jan. 1998.
- [4] Seok Joo Koh and et al., "Reliable Multicast Transport Building Block: Tree Auto-Configuration," *Internet Draft*, Mar. 2001.
- [5] Bhattacharyya and et al., "An Overview of SSM Deployment," *Internet Draft*, May 2000.
- [6] Thomas A. Maufer, *Deploying IP Multicast in the Enterprise*, Prentice Hall, 1997.
- [7] Whetten and et al., "RMT Building Blocks," *RFC 3048*, Jan. 2001.
- [8] Luby and et al., "RMT BB Forward Error Correction Codes," *Internet Draft*, Jul. 2001.
- [9] B. Adamson and et al., "NACK-Oriented Reliable Multicast Protocol," *Internet Draft*, Jul. 2001.
- [10] Whetton B., et al., *The Reliable Multicast Transport Protocol*, IETF *Internet Draft*, draft-whetton-rmt-ii-00.txt, Apr 1998.

● 제 자 소개 ●



정 선 화

1993년 경원대학교 전자계산학(공학사)
1995년 경원대학교 교육대학원 전자계산학(교육학석사)
1997년 성일정보산업 고등학교 교사
2001년~현재 : 경원대학교 대학원 컴퓨터공학(박사과정)
관심분야 : 차세대 이동통신, NGcN, 유비쿼터스 컴퓨팅 etc.
E-mail : smw0@hitel.net



박 석 천

1977년 고려대학교 전자공학(공학사)
1982년 고려대학교 컴퓨터공학(공학석사)
1989년 고려대학교 컴퓨터공학(공학박사)
1979년~1985년 금성통신 연구소
1991년~1992년 Univ. of California, Irvine Post Doc.
1992년~1994년 경원대학교 산업기술연구소장
1988년~현재 : 경원대학교 컴퓨터공학과 정교수
관심분야 : NGcN, 유비쿼터스 컴퓨팅, 멀티미디어 통신, 위성 통신망, IMT-2000 etc.
E-mail : scpark@kyungwon.ac.kr