

대리송수신자 개념을 이용한 신뢰성 있는 멀티캐스트 전송기법

안 병 호[†] · 조 국 현^{††}

요 약

멀티캐스트 전송은 다자간 응용들의 전송서비스를 제공하기 위한 중요한 전송 방식이다. 멀티캐스트 전송환경에 대한 최근의 많은 연구는 다양한 문제점들을 도출하고 있다. 이러한 문제들 중 대표적인 문제가 신뢰성(reliability) 및 확장성(scalability)을 보장할 수 있는 멀티캐스트 전송서비스 문제이다. 본 논문은 최근의 신뢰성 있는 멀티캐스트 전송방식에서 발생하는 문제점들을 분석해 보고, 특히 신뢰성 보장을 위한 멀티캐스트 전송시 문제점을 해결하기 위해서 대리 송수신자(ASRC)의 개념을 이용한 새로운 통신 구조를 제안한다. 이를 위해서 본 논문에서는 일반적인 송신자(sender-initiated) 및 수신자 위주(receiver-initiated)의 전송프로토콜에 제안된 구조를 적용할 수 있는 방법을 제안하고, 제안된 기법의 성능을 멀티캐스트 송수신자의 시스템 처리요구양(processing requirement)과 최대처리율(maximum throughput)측면에서 비교 분석하여 대리송수신자 개념의 타당성을 검증한다.

A Reliable Multicast Transfer Method Using Agent Sender & Receiver Concept

Byoung-Ho Ahn[†] · Kuk-Hyun Cho^{††}

ABSTRACT

A Multicast transfer is a critical delivery method to provide a transport service to multipeer applications. The various problems on the multicast transfer environments have been occurred from the results of current research. One of these problems is the multicast transport service issue to guarantee reliability and scalability. First, this paper presents the related research of the reliable multicast transport methods, and then proposes a new transfer architecture using the Agent Sender and Receiver Concept(ASRC) to solve a reliable multicast transfer issue. We also propose a method to apply the proposed architecture(ASRC) to the well-known sender-initiated and receiver-initiated transport protocol. In order to validate the proposed ASRC architecture, the applied sender and receiver system is compared and analyzed over the processing requirement and maximum throughput.

1. 서 론

최근의 초고속 통신망과 컴퓨터 처리 기술의 발달 및 멀티미디어 기술의 향상은 기존의 통신망 구조를

급격하게 변화시키고 있다. 이러한 정보통신 분야의 급격한 변화는 다양하게 요구되는 정보통신 사용자 요구들을 만족시키기 위한 각종 응용들의 등장과 밀접한 관계가 있다. 이러한 다양한 응용들은 기존의 전송서비스로는 제공하기 어려운 다자간 화상회의 및 공동작업 등 다양하게 등장하고 있다. 멀티캐스트 통신구조의 개념은 특정 송신자의 정보를 받고자 하는 수신자

[†] 정 회 원 : 충청대학 전자계산과 교수
^{††} 정 회 원 : 광운대학교 전자계산학과 교수
논문접수 : 1998년 8월 5일, 심사완료 : 1998년 12월 7일

들에게만 통신할 수 있도록 하는 방식으로 이러한 다자간 통신 응용들의 전송서비스를 제공하기 위해 필수적인 전송 방식이다.

또한 이러한 새로운 응용들은 다수 참여자들간의 사용자 정보의 분산을 필연적으로 다루고 있으며, IP 멀티캐스트[1]의 대중화 및 인터넷(internet)을 이용한 MBONE(Multicast backBONE)의 과급으로 기하학적인 확장을 야기 시키고 있다. 또한 통신그룹의 증가에 따라 신뢰성 있고 확장성있는 오류제어 및 트래픽제어를 포함하는 효율적인 전송 관리체계는 더욱 필수적으로 요구되는 것이다.

최근 연구들은 이 기종 망에서 광범위한 멀티캐스트 응용들을 만족하기 위한 전송계층 프로토콜들을 활발하게 개발하고 있다[3, 4, 5, 6]. 어떤 연구들은 그룹 통신의 특별한 측면들만을 기술하고 특정한 사용자환경들에만 초점을 맞추고 있다. 또한 다른 연구들은 멀티캐스트 통신구조에서 발생하는 문제들을 국소적으로 해결하고자 별도의 프로토콜들을 개발하고 있다. 그러나, 이러한 기존방식들은 멀티캐스트 통신환경의 문제를 해결하기 위한 방안을 제안하고 있으나, 기존 시스템간 효율성, 상호운용성 및 이식성 문제들과 같은 또 다른 문제들을 야기하고 있다.

그러므로, 본 논문에서는 기존의 멀티캐스트 통신구조의 신뢰성 및 확장성 문제를 해결하기 위한 새로운 전송 프로토콜 개발등의 접근방식과는 달리 기존의 IP 멀티캐스트 통신환경에서 신뢰성있고 확장성을 고려할 수 있는 새로운 전송 기법을 제안한다.

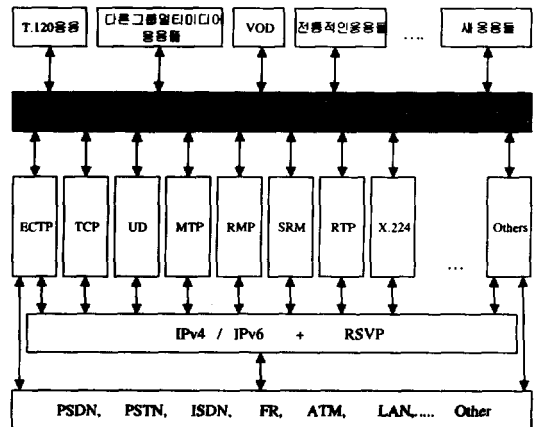
본 논문은 멀티캐스트망의 특정한 전송 프로토콜에 대한 검증은 다루지 않고 있으며 일반적인 전송 프로토콜로 알려진 송신자 위주(sender-initiate)와 수신자 위주(receiver-initiate) 전송 프로토콜의 기존 개념에 별도의 멀티캐스트 트래픽 제어용 시스템을 이용하는 대리송수신자개념(ASRC : Agent Sender & Receiver Concept)제안하고, 적용하여 이들의 기능을 재정의함으로써 전체 시스템의 전송량 측면을 분석한다.

이를 위해서, 본 논문은 먼저 기존의 신뢰할 수 있는 멀티캐스트 전송기법들을 기술하고, 이에 대한 문제점을 분석한 후에 본 논문에서 제안한 대리송수신자 개념(ASRC)의 배경, 기본 원리, 구조 및 구성요소의 각 기능들을 기술한다. 다음으로 기존의 송신자위주 전송기법, 수신자위주 전송기법에 본 논문에서 제안한 ASRC를 적용한 구조의 동작을 기술한다. 또한, 본 논

문에서 제안한 ASRC 적용시 구성요소 각각에 대해서 시스템의 처리요양과 최대 처리율을 분석한후, 이에 대한 결과를 분석한 다음 마지막으로 본 논문의 결론과 향후 연구과제들을 기술한다.

2. 신뢰성 있는 멀티캐스트 전송

멀티캐스트 통신환경의 연구들은 대부분 크게 두가지 측면에서 진행되고 있다. 즉, 첫 번째로 MOSPF, DVMRP, PIM-DM/SM 등과 같은 효율적인 IP 멀티캐스트 경로배정을 위한 프로토콜들의 연구이며, 두 번째로 이를 이용하여 신뢰성(reliability)과 확장성(scalability) 등을 보장하기 위한 전송계층 프로토콜의 연구 개발이다. (그림 1)은 현재까지 알려진 멀티캐스트 통신구조의 프로토콜 스택을 보이고 있다.



(그림 1) 멀티캐스트 통신환경을 위한 잘 알려진 프로토콜 스택

(Fig. 1) A well-known protocol stack for multicasting

(그림 1)의 프로토콜 구조는 기존의 일대일(point-to-point) 전송과는 다른 다자간 전송환경의 응용들과 이를 위한 최근까지 연구된 통신 프로토콜들의 스택을 보인 것이다[6].

이러한 전송 프로토콜들은 멀티캐스트 통신환경에서 송신자와 수신자들간의 트래픽 폭주 및 흐름제어는 물론 정확한 데이터 전송을 보장하기 위해서 필요한 것이다. 그러므로 완전한 신뢰성 보장의 통신 서비스의 개발은 수신자의 상태를 송신자에게 되돌리는 전송(소위 "피드백 정보"라고 함) 또는 그 반대의 전송이 요구된다.

피드백 정보의 전송기법들은 크게 두 가지로 고려되고 있다. 즉, 송신자위주와 수신자위주의 기법이 그것이다[7, 8, 10]. 먼저 송신자 위주 기법은 송신자가 특정 데이터 패킷을 멀티캐스트 통신 그룹의 수신자들에게 전송한 후 이에 대한 수신 여부 신뢰성을 위해서 모든 수신자들에게 수신했음을 알리는 상태정보(Positive Acknowledgement) Ack를 받는 것이다. 이때 반송된 Ack의 양으로 인해 발생하는 문제가 잘 알려진 송신자 폭주(Sender Implosion)문제이다. 많은 연구들이 이를 해결하기 위한 또 다른 방법들을 제안하고 있다. 한편 이와는 달리 송신자 측의 Ack정보 처리의 부담을 줄이고 이 부담을 수신자들에게 분산하려고 하는 기법이 소위 수신자위주 기법(Receiver Initiated)이다. 이것은 송신자는 데이터 패킷을 송신하고 수신자들이 분실된 패킷의 발견이나, 오류 패킷의 발견을 책임지고 이에 대한 상태정보 NAK(Negative Acknowledgement)를 전송하여 재전송을 요구한다. 이 기법에서도 통신그룹의 크기가 증가함에 따라 수신자들의 동시 다발적인 NAK의 전송으로 소위 NAK 폭주(NAK Implosion)문제를 발생시킬 수 있다. 최근의 멀티캐스트 통신환경에 대한 연구과제들은 이러한 문제들을 어떻게 해결하는가에 초점을 맞추고 있으며 상당한 진척이 있었다.

이러한 폭주 문제들을 해결하기 위해서 연구개발자들의 의도에 따라 서로 다른 절차를 사용하는 여러 가지 멀티캐스트 프로토콜들이 설계되어왔다[3, 4, 5, 6, 10]. 예를 들어, MTP(Multicast Transport Protocol)와 RMP(Reliable Multicast Protocol)같은 프로토콜들은 총체적 순서위주(Totally ordered) 멀티캐스트 전송을 제공하기 위해서 중앙 집중의 제어기법을 실현하고 있다. 또한 상기 기술한 일련의 상태 정보의 폭주로 인한 문제를 해결하기 위한 것이 XTP(eXpress Transport Protocol) 프로토콜이며, 이것을 확장하여 수신자 위주 기법을 구현하여 완전 신뢰성 서비스는 어렵지만 멀티캐스트 그룹의 참여자들의 확장성 문제를 위한 것이 SRM(Scalable and Reliable Multicast)이다. 또한 수신자들의 집단을 다시 일련의 부 집단(Subgroup)으로 분리하여 각 부집단의 제어를 이용하여 폭주문제를 해결하고 전송 비용을 줄이려고 하는 LGC(Local Group Concept)[9] 등이 있다.

그러나, 이러한 멀티캐스트 전송 프로토콜들은 다음과 같은 문제들로 인해 점점 복잡하고 다양화될 수밖에

에 없는 실정이다.

- 원격 화상회의 및 공동 작업등 멀티캐스트 응용을 위한 전송 프로토콜 내의 제어 및 관리 메카니즘의 복잡성 증가
- 멀티캐스트 그룹 참여자들의 일관성 있는 상태관리를 위한 구조상의 복잡성 증가
- 데이터 송수신 및 제어를 하나의 메카니즘으로 모두 처리하려고 하는데 따른 송신측과 수신측의 처리부담 증가
- 각 그룹 참여자들의 상태 정보 파악을 위한 정보 교환으로 인한 트래픽 문제 및 이를 처리하기 위한 부담의 증가
- 송신측의 전송 데이터 버퍼 관리(유지)에 따른 자원의 낭비

다시 말하면, 멀티캐스트 통신환경의 송신자와 수신자들간의 순수 데이터 전송이외의 메카니즘 처리에 필요한 부하의 증가에 따른 효율성 및 융통성이 구조적인 측면에서 전체 시스템 성능 저하의 또 다른 요인이 되는 것이다.

3. 대리송수신자 개념(ASRC)

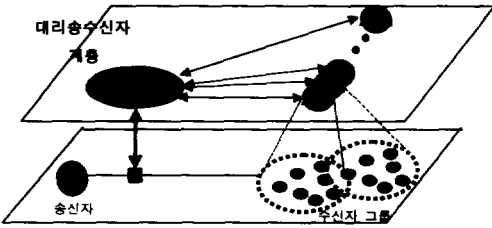
여기서는 멀티캐스트 망에 ASRC를 적용하여 송신자와 수신자들간의 관계를 알아보고 ASRC의 구성 요소들의 기능을 설명한다.

3.1 ASRC의 개념 및 구조

ASRC 개념은 멀티캐스트 통신환경에서의 송수신자들간의 순수 데이터 교환 이외의 제어 문제로 인한 부담을 별도로 생각하자는 것이다. 즉, ASRC는 다자간 통신환경의 망에서는 송수신자간의 순수한 의미의 데이터 전송만을 수행하도록 하고, 이에 따라 발생하는 각종 관리 및 제어 기능은 독립적으로 분리하여 별도의 제어측면에서 관할하여, 신뢰성 있는 다자간 통신환경의 트래픽 부하의 분산 및 제어의 전문성을 도모하도록 하기 위함이다.

멀티캐스트 통신환경에서 각 송수신자들의 기존의 방식과는 달리 ASRC개념은 각 수신자들의 분실 패킷 재전송을 담당하는 대리 송신시스템을 두어 이에 대한 제어 및 관리를 담당하도록 한다. 이것은 먼저 사용자

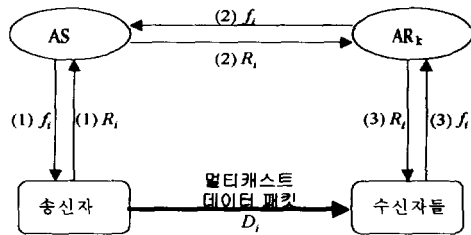
의 용용과 관련되는 순수한 데이터 전송과 제어용 데이터의 전송을 독립시킴으로서 사용자 망의 송수신 측에서의 재전송에 대한 부담을 제거하려고 하는 것이다. 이를 위한 ASRC의 논리적 구조는 (그림 2)와 같다.



(그림 2) 멀티캐스트 통신환경의 ASRC 논리적 구조
(Fig. 2) The ASRC logical architecture of multicasting environments

(그림 2)에서 ASRC 개념의 적용은 기존의 멀티캐스트 통신구조에 대리송수신자 기능의 계층을 추가 적용하고, 일반적인 송신자 및 수신자 위주의 전송프로토콜의 구조는 그대로 적용한다. 즉, 순수 다자간 데이터 전송을 담당하는 멀티캐스트 망 계층과 이에 대한 제어를 담당하는 대리송수신자 계층을 논리적으로 분리하는 계층 구조이다.

여기서 대리송수신자는 다른 수신자들과 마찬가지로 송신자로부터 전송된 데이터를 멀티캐스트 그룹의 일원으로 수신한다. 단지 송신자는 자신의 전송 데이터에 대한 신뢰성을 대리송수신자로부터만 확인한다. 또한 대리수신자들도 다른 수신자들과 같은 방법으로 수신에 참여한다. 이때 대리 수신자는 자신이 제어하는 수신자 부 그룹의 참여자들에게 재전송 책임을 지고 대리송수신자는 대리수신자들의 재전송 책임을 진다. (그림 3)은 제안된 ASRC 논리적 구조를 기반으로 각 구성요소들간의 전송 동작을 보여 주고 있다.



(그림 3) ASRC 구조의 피드백 제어모델
(Fig. 3) The feedback control model of ASRC architecture

ASRC 구조를 따른 멀티캐스트 전송환경에 대한 데이터 전송 및 제어모델은 다음과 같은 절차로 실행된다.

- 먼저 해당 멀티캐스트 그룹에 대리 송신자(AS)와 대리수신자(AR_k)를 등록하고 설정한다.
- 송신자는 멀티캐스트 데이터패킷(D_i)을 모든 수신자들에게 전송하고, 각 수신자들로부터의 제어용 응답을 받는 대신에 대리송수신자로부터만 전송패킷에 대한 응답(1)f_i을 받아 모든 수신자들이 수신한 것으로 간주한다.
- 한편, 수신자들의 피드백정보(3)f_i는 대리수신자에게 전달하고 자신이 처리할 수 있다면 재전송(3)R_i 하고, 아니면 대리 수신자는 자신의 주소 그룹 수신자들을 대신하여 대리송수신자에게 피드백 정보(2)f_i를 전달한다.
- 대리 송신자는 자신에게 전달된 피드백 정보(2)f_i에 대한 처리 후, 재전송 패킷(2)R_i을 전달한다.
- 여기서 피드백 정보(f_i)의 종류는 적용되는 전송 프로토콜의 동작을 따른다.

ASRC 구조는 신뢰성 있는 멀티캐스트 전송을 위해서 기존의 데이터 경로와 제어 경로를 분리함으로써 송신자가 부담하는 모든 수신자들로부터의 피드백 정보들을 분산 처리한다.

3.2 구성요소 기능

여기서는 제안된 ASRC 구조를 구성하는 각 요소들의 기능들을 기술한다.

3.2.1 대리송수신자(Agent Sender)

- 대리송수신자는 송신자를 대신하여 모든 수신자들의 피드백정보를 처리하고, 패킷의 재전송 기능을 담당한다.
- 대리송수신자는 현재의 멀티캐스트 그룹에 속한 모든 참여자들에게 자신의 주소를 멀티캐스트하여, 자신의 존재를 알린다. 즉, 참여 그룹의 멀티캐스트 주소를 이용하여 데이터를 수신한다. 단, 이때, 대리송수신자와 송신자간에는 신뢰성이 보장되어야 한다.

- 수신자들이 송신자로부터 수신된 데이터 패킷에 대한 수신여부의 응답 및 재전송책임은 송신자대신 수행한다. 이로서, 송신자는 신뢰성 있게 데이터 전송이 이루어진 것으로 보고 자신의 버퍼를 비울 수 있다. 이것은 송신자의 버퍼를 유지하는 시간의 감소를 의미하는 것이다. 그 이유는 송신자의 입장에서는 대리송신자로부터의 응답만을 받으면, 신뢰성 있게 전송되었다고 가정하기 때문이다. 단, 이때의 버퍼관리를 담당하는 대리송신자는 시스템상의 오류가 없는 신뢰할 수 있는 시스템으로 가정하며, 이러한 가정은 본 논문의 진행상 문제가 없는 것임을 주의하여야 한다.
- 그러므로, 대리송신자는 송신자로부터 신뢰성 있게 수신한 패킷들에 대한 관리 책임을 진다.
- 또한 대리송신자는 대리수신자에게 데이터의 재전송 책임을 진다.
- 이러한 대리송신자의 전송 동안에 원래 송신자는 계속 자신의 데이터 패킷의 전송만을 실행하고, 관리와 연관된 일은 모두 이에게 맡긴다.

3.2.2 대리 수신자

대리 수신자는 [9]에서 제안한 LGC(Local Group Concept)의 개념을 도입한 것으로 각각의 국소 그룹에 포함된 수신자들의 대표적인 국소 그룹 제어기(lgc)의 기능을 사용할 수 있다. ASRC구조의 대리수신자는 다음의 기능을 수행한다.

- 대리수신자는 국소의 수신자 그룹의 피드백제어 정보들을 모아 처리하고 대리송신자에게 이들을 대표하여 피드백 정보를 대리송신자에게 전송함으로써 피드백 정보의 양을 줄인다.
- 수신자는 대리송신자로부터 자신의 대리수신자를 지정 받는다.
- 대리수신자는 자신의 참여자들에 대한 재전송 책임을 진다. 이때 재전송이 어려우면 대리 송신자에게 재전송 받는다.

대리 수신자에 대한 관리는 일반적으로 각 수신자들과 가장 가까운 대리 수신자들에게 할당하는 대표적인 국소 그룹 제어기(lgc) 할당 방식을 사용한다[9]. 수신자와 대리수신자에 대한 할당 및 관리를 위해서 대리송신자는 이들에 대한 거리 정보와 수신자의 등록

정보를 별도로 관리한다.

3.2.3 송신자

송신자의 멀티캐스트 데이터는 대리송신자의 피드백 정보를 기반으로 전송된다. 이 때 송신자는 기존의 방식과는 달리 대리송신자로부터의 피드백 정보만을 기본으로 다음 데이터 패킷의 전송 여부를 결정한다.

3.2.4 수신자

수신자는 해당 멀티캐스트 그룹으로 전송된 송신자의 데이터 패킷을 수신하고, 이에 대한 피드백 정보는 송신자에게 전송하지 않고 자신의 대리수신자에 전송한다. 수신자의 오류발생 시, 재전송 책임은 먼저 대리수신자에게 있고, 최종적으로 대리송신자가 특정 수신자의 재전송 패킷을 전송한다. 이렇게 하여, 송신자는 모든 수신자들로부터의 피드백 정보에 대한 처리 부담을 최소화한다.

4. ASRC 구조의 전송동작 적용

여기서는 대표적인 일반적 전송 프로토콜인 송신자위주(A), 수신자위주(N) 프로토콜 및 이들 각각에 ASRC 개념을 도입한 전송 기법을 각각 "AA(Asrc on A)"와 "AN(Asrc on N)"이라고 하며, 이들에 대한 기본적인 동작을 기술한다.

송신자위주와 수신자위주의 프로토콜은 [8]에서 기술한 프로토콜을 이용한다. 먼저 송신자위주 프로토콜(A)의 개략적 동작은 다음과 같다.

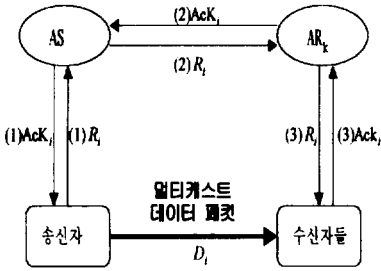
- 오류복구는 분실된 패킷들만을 재전송하는 선택적반복(Selective-Repeat)방식을 채택한다.
- 송신자가 전송하거나 재전송할 때 마다, 모든 수신자들에게 브로드캐스트하고 타이머를 시작한다.
- 수신자가 정확하게 패킷을 수신할 때, Ack를 일대일(point-to-point) 연결로 전송한다.
- 타이머가 만료될 때 마다, 송신자는 대응하는 패킷을 다시 브로드캐스트하고 새로운 패킷전송보다 높은 우선순위로 전송한다.

수신자위주 프로토콜(N)의 개략적 동작은 다음과 같다.

- 송신자는 모든 패킷들을 브로드캐스트한다.
- 송신자는 새로운 패킷들의 전송보다 재전송을 우선적으로 처리한다.
- 수신자는 분실 패킷을 발견할 때 마다 일대일 채널을 통해서 NAK를 전송하고 타이머를 설정한다.
- NAK로 재전송을 요구한 이전 패킷의 수신 없이 타이머가 끝나면 분실된 패킷으로 간주하여 다시 요청한다.

- 수신자는 수신된 패킷을 받고 자신의 대리수신자에게 Ack(3)Ack_k를 전송한다.
- 수신자의 분실된 패킷은 일차적으로 대리수신자가 재 전송한다(3)R_k.
- 대리수신자는 수신자의 재전송 요구에 재전송을 수행하며, 이것이 불가하거나 자신만의 분실 패킷 발생 시 대리송신자에게 응답 Ack(2)Ack_i를 전송하지 않는다.
- 대리송신자의 타이머 값의 만료 시 응답이 없는 대리수신자에게 재 전송한다(2)R_i.

(그림 4)는 제안된 ASRC개념을 데이터 전송을 위한 송신자 위주 전송 프로토콜에 적용한 ASRC 전송기법(AA)에 대한 전송절차 모델을 보이고 있다.



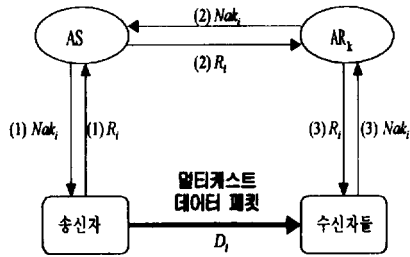
(그림 4) 송신자 위주의 전송기법에 적용된 ASRC 모델 (Fig. 4) The ASRC model applied on a sender-initiated transfer

다음은 (그림 4)의 송신자 위주의 전송 기법에 적용된 AA의 전송 동작이다.

- 대리송신자와 대리수신자는 멀티캐스트 그룹에 등록 및 설정한다.
- 송신자는 모든 데이터 패킷을 멀티캐스트(D_i)하고 타이머를 설정한다.
- 송신자는 대리송신자로부터만 (1)Ack_k를 수신한다. 이때 타이머가 만료될 때까지 Ack를 수신하지 못하면 대리송신자에게 일대일로 패킷을 재 전송한다(1)R_i.
- 대리 송신자는 송신자로부터 패킷 수신 후 타이머를 설정하고 Ack를 송신자에게 전송한다.
- 이때 타이머 설정은 대리수신자들로부터의 Ack를 처리하기 위한 것이다.

여기서 송신자의 특정 데이터 패킷의 전송 시 설정하는 타이머 값은 전송 패킷의 응답을 처리해야하는 대리송신자와 대리수신자의 타이머설정에 이용되도록 하기 위해서 대리송신자와 대리수신자에게 전달될 수 있는 것으로 가정한다. 이 가정은 멀티캐스트 전송 프로토콜의 데이터 패킷의 헤더정보를 이용할 수 있을 것이며, 본 논문에서는 특정 프로토콜의 패킷 형식의 내용은 다루지 않기로 한다.

(그림 5)는 제안된 ASRC개념을 데이터 전송을 위한 수신자 위주 전송 프로토콜에 적용한 ASRC 전송기법(AN)에 대한 전송절차에 대한 모델을 보이고 있다.



(그림 5) 수신자 위주의 전송기법에 적용된 ASRC 모델 (Fig. 5) The ASRC model applied on a receiver-initiated transfer

다음은 (그림 5)의 수신자 위주의 전송 기법에 적용된 AN의 전송 동작이다.

- 대리송신자와 대리수신자는 멀티캐스트 그룹에 등록 및 설정한다.
- 송신자는 모든 데이터 패킷(D_i)을 멀티캐스트한다.
- 송신자는 대리송신자의 데이터 분실 시에만 (1)Nak_k를 수신 받아 처리한다.

- 대리 송신자는 분실된 패킷의 발견 시 송신자에게 (1)Nak,를 전송하고, 타이머를 설정한다.
- 수신자들은 오류발견시 대리 수신자에게 (3)Nak,를 전송하고, 타이머를 설정한다.
- 대리수신자는 수신자의 재전송 요구에 재전송을 수행하며 (3)R,, 이것이 불가하거나, 자신만의 분실 패킷 발생 시 대리 송신자에게 (2)Nak,를 전송하여 요구하고 타이머를 설정한다.

5. 성능 및 결과 분석

여기서는 본 논문에서 제안한 ASRC 구조를 송신자 위주와 수신자위주의 전송 프로토콜에 적용한 AA와 AN 방식에 대한 성능을 각 경우의 시스템 처리요구양 (processing requirement)과 최대 처리율(maximum throughput) 측면에서 분석해 본다. 여기서 최대처리율은 송신자, 수신자, 대리송신자 및 대리수신자 측에서 하나의 패킷을 처리하는데 소요되는 최소 처리시간을 검출하여 그 역으로 계산된다. 또한 본 논문의 성능 분석은 [7][8][10] 등에서 사용한 비교 모델을 근간으로 사용한다.

5.1 가정

본 논문에서는 제안된 AA 및 AN에 대한 최대 처리율의 분석 시 다음과 같은 가정을 기본으로 하고 있다.

- 먼저 앞서 기술한 바와 같이 대리송신자 및 대리 수신자는 멀티캐스트 망의 순수 참여자의 일부 기능이 아닌 별도의 시스템들로 구성된다. 이것은 최근 처리 시스템의 성능향상 및 가격인하로 가능하다.
- 또한 이 시스템들은 처리성능 상 신뢰할 수 있는 시스템이다.
- 대리 송수신자들의 처리율은 전체 멀티캐스트 망의 참여자들과는 별도의 시스템에서 병행 처리되므로 본 논문의 성격상 최대 처리율 분석 시에는 적용되지 않으며, 이것은 단지 기존의 송신자와 수신자의 처리요구양 부담의 전이량이다.
- 또한 전송된 패킷의 피드백 정보 Ack와 Nak 패킷은 분실되지 않는 것으로 가정한다. 이것은 다음에

사용할 수식을 단순화시키기 위한 것으로 이 가정은 본 논문의 내용에는 무관하다.

5.2 용어

본 논문의 성능 분석 시 사용되는 기본적인 용어는 [7][8][10]에서 사용하는 용어와 같으며, 본 논문에서 확장 정의된 용어를 <표 1>과 같이 사용한다.

또한 여기서 전체 멀티캐스트 그룹 G는 n명의 참여자들로 구성되며, $G = \{m_1, m_2, \dots, m_n\}$ 으로 표시하자. 모든 수신자들은 G에 속하며, 대리수신자가 관리하는 수신자 부 그룹(subgroup) $G_i = \{1, 2, \dots, K\}$ 로 표시하고 $G \supseteq G_i$ 이며, 이때 $K(\geq 1)$ 는 각 국소 부그룹의 참여자수로서 해당 부그룹의 크기이다. 또한 멀티캐스트 그룹에 대한 부그룹의 수를 W라고 한다.

5.3 최대 처리율 분석

본 논문에서 송신자위주 및 수신자위주의 전송 프로토콜에 대한 처리량 성능 분석의 결과는 [7][8][10]의 결과를 따르고, 여기서는 이들 각각에 제안된 ASRC 전송기법을 적용한 AA와 AN측면에 대한 분석에 중점을 두어 다룬다.

5.3.1 AA 전송기법

여기서는 위에서 기술된 동작의 실행을 위해서 처리되는 처리요구양을 각 시스템 측면에서 알아본다. 먼저, AA 전송기법에서 송신자의 처리요구양은 다음의 (1)과 같다.

$$X^{AA} = X_f + X_p(1) + X_a(1) + \sum_{m=2}^K (X_i(m) + X_p(m)) \quad (1)$$

이 수식에서 첫 번째 항은 송신자의 상위계층에서 받은 데이터를 패킷으로 구성하기 위한 것이고, 두 번째와 세 번째 항은 하나의 패킷을 처리하여 전송하고 대리송신자로부터 이에 대한 Ack를 받아 처리하기 위한 시간이다. 마지막 항은 송신자의 전송 패킷을 대리송신자가 받지 못한 경우 Ack의 타이머 만료를 처리하고 재전송을 위한 시간이다. 이때 M_r 은 대리송신자도 수신자들의 일종으로 취급하기 위한 것이다. (1)식에 기대값을 취하면 (2)와 같은 수식이 될 것이다.

(표 1) 표기의 의미
(Table 1) The meaning of symbols

R	수신자 수 : $R = \ G\ - 1$.
G	멀티캐스트 그룹 : n
X_r	상위계층로부터 데이터를 받아 데이터 패킷을 생성하는데 걸리는 시간
X_p	하나의 패킷의 전송을 처리하는데 걸리는 시간
X_a, X_n, X_i	송신측에서 하나의 ACK, Nak 및 타이머만료를 수신하여 처리하는데 걸리는 시간
X_t, Y_t	각 송수신측에서 타이머를 처리하는데 걸리는 시간, 여기에는 타이머 동작 시키거나 중단, 또는 실행시키기 위한 시간들이 포함된다.
Y_p	새로운 수신된 패킷을 처리하는데 걸리는 시간
Y_f	상위 계층에 데이터를 전달하는데 걸리는 시간
Y_a, Y_n	수신측에서 ACK 또는 Nak를 생성하여 전송하는데 걸리는 시간
p	수신측에서의 패킷 분실 확률. 수신자들의 분실확률은 서로 독립적이다.
L^A, L^N	패킷 마다 수신자 r 에 의해서 보내진 ACK 또는 Nak 의 수
L^A_{ar}, L^N_{ar}	대리수신자로부터 보내진 Ack, Nak의 수
L^A_k, L^N_k	국소그룹의 수신자들로부터 보내진 Ack, 또는 Nak의 수
M_r	패킷 한 개에 대해 모든 수신자 r 이 성공적으로 받도록 하기 위한 전송 횟수
M	패킷 한개에 대해 모든 수신자들이 성공적으로 받도록 하기 위한 전송 횟수 ; $M = \max(M_r)$
T_p	초당 순수 네트워크 통과 시간. i.e., 단방향 링크를 통과하는데 걸리는 순수 지연시간
X^w, Y^w, AS^w, AR^w	송신자, 수신자, 대리송신자 및 대리수신자에서 패킷당 처리시간. $w \in \{A, N, AA, AM\}$
$\Lambda^w_s, \Lambda^w_r, \Lambda^w_{as}, \Lambda^w_{ar}, \Lambda^w$	송신자, 수신자, 대리송신자, 대리수신자 및 전체시스템의 처리율. $w = \{A, N, AA, AM\}$

$$E[X^{AA}] = E[X_p] + E[M_r]E[X_p] + E[X_a] + (E[M_r] - 1)E[X_i] \quad (2)$$

여기서 M_r 은 수신자 r 이 특정 패킷을 정확하게 수신하는데 요구되는 전송의 수로 정의되며 송신자는 대리송신자를 r 로 생각할 수 있을 것이다. 그러므로,

$$P[M_r \leq m] = 1 - p^m, m = 1, \dots \text{이며, } E[M_r] = \frac{1}{(1-p)}$$

가 된다. 이것을 (2)식에 적용하면,

$$E[X^{AA}] = E[X_p] + E[M]E[X_p] + E[X_a] + \frac{p}{(1-p)} E[X_i] \quad (3)$$

그러므로, $E[X^{AA}] \approx O(1/(1-p))$ 가 되며, 수신자의 수(R)와는 무관하다. 이때 처리율은 $\Lambda_X^{AA} = 1/E[X^{AA}]$ 이다.

한편, 대리송신자의 처리요구량은 다음의 (4)와 같다.

$$AS^{AA} = Y_{a(1)} + \sum_{m=2}^M (X_r(m) + X_p(m)) + \sum_{i=1}^{L^A} X_a(i) \quad (4)$$

수식 (4)에서, 첫 번째 항은 대리송신자가 송신자로 Ack를 전송하는데 걸리는 처리시간이고, 두 번째 항은 대리수신자들로부터의 Ack에 대한 타이머 만료와 이에 대한 처리시간이며, 마지막 항에서 L^A_{ar} 는 대리수신자로부터 Ack 수이다. 이에 대한 기대치를 구하면, 식 (5)와 같다.

$$E[AS^{AA}] = E[Y_a] + (E[M] - 1)E[X_i] + E[X_p] + E[L^A_{ar}]E[X_a] \quad (5)$$

여기서, $E[L^A_{ar}] = WE[M](1-p)$ 이고, 이를 대입하면 (6)과 같다.

$$E[AS^{AA}] = E[Y_a] + (E[M] - 1)(E[X_i] + E[X_p]) + WE[M](1-p)E[X_a] \quad (6)$$

식 (6)에서, 수신자들의 수 $R=W$ 이므로, [7,8]의 정의에 의해서,

$$E[M] = \sum_{i=0}^{\infty} \binom{W}{i} (-1)^{i+1} \frac{1}{(1-p)^i} \quad (6.1)$$

그러므로,

$$E[M] = \alpha \left(1 + \frac{p}{1-p} \ln W \right) \quad (6.2)$$

이고, (6.1)과 (6.2)를 식 (6)과 함께 고려하면,

$$\begin{aligned} E[AS^{AA}] &\in \alpha W \left(1 + \frac{p}{(1-p)} \ln W \right) (1-p) \\ &\in \alpha W (1-p) + W p \ln W \end{aligned}$$

이다.

그러므로 대리송신자의 처리요구량은 대리 수신자의 수 W 에 의존한다. 이때 W 는 별도의 시스템으로 구축되므로 융통성 있는 시스템 구조를 가질 수 있을 것이다. 이것은 또한 기존의 송신자위주 전송프로토콜에서 송신자 처리량의 부담을 대리송신자로 전이할 수 있는 처리량을 의미한다.

다음은 AA전송 기법에서 수신자의 처리요구량을 알아 본다. 이때 수신자는 수신자 본연의 처리면에서, 송신자위주 전송 프로토콜과 같을 것이다. 그러므로, (7)식을 자연스럽게 얻을 수 있다.

$$\begin{aligned} Y^{AA} &= Y_r + P(M_r=1) Y_p(1) + P(M_r>1) \sum_{i=2}^{\infty} Y_p(i) + \\ &P(M_r=1) Y_a(1) + P(M_r>1) \sum_{i=2}^{\infty} Y_a(i) \quad (7) \end{aligned}$$

(7)식에서 첫 번째 항은 상위계층에서 데이터를 받아 패킷으로 구성하고 처리하며, 두 번째와 세 번째 항은 새로운 패킷의 올바른 수신을 위한 처리시간이다. 또한 네 번째와 다섯 번째항은 올바르게 수신된 패킷에 대한 Ack 응답처리시간을 의미한다. (7)식의 수신자 처리요구량에 기대치를 취하면,

$$E[Y^{AA}] = E[Y_r] + E[M](1-p)E(Y_p) + E[M](1-p)E(Y_a) \quad (8)$$

식 (8)은 [7,8,10]에 의해서, $E[Y^{AA}] = \alpha(1-p + p \ln K)$ 이다. 그러므로, 이 수신자 처리요구량은 국소그룹의 수신자 수 $K(\geq 1)$ 에 의존하며, 이것은 수식을 간단히 하기 위해서 $K = \lceil \frac{R}{W} \rceil$ 로 정의한다.

다음으로 AA전송기법의 대리수신자의 처리요구량을 알아보자. 이것은 AA전송기법의 수신자 처리요구

량에 자신의 국소그룹에 속한 수신자들의 Ack를 처리하기 위한 시간과 자신의 수신자들에게 재전송하기 위한 시간이 요구된다. 이것을 수식으로 표시하면, 식(9)과 같으며, 이것의 기대치를 취한 것이 식(10)이다.

$$AR^{AA} = Y^{AA} + \sum_{j=1}^{L_s^A} X_s(j) \quad (9)$$

이때, L_s^A 는 K 개의 국소 그룹 수신자로부터 보내진 Ack 수이므로, $E[L_s^A] = KE[M](1-p)$ 이다. 이를 식(9)에 기대치를 취하여 대입하면,

$$\begin{aligned} E[AR^{AA}] &= E[Y^{AA}] + E[L_s^A]E[X_s] + \\ &E[M](E[X_r] + E[X_p]) \end{aligned}$$

이고,

$$\begin{aligned} E[AR^{AA}] &= E[Y^{AA}] + KE[M](1-p)E[X_s] + \\ &E[M](E[X_r] + E[X_p]) \quad (10) \end{aligned}$$

이다.

이 전송기법에서 구성 시스템의 전체처리율 Λ^{AA} 는 송신측, 수신측에서 패킷 당 처리요구량의 최소값으로 주어진다.

즉,

$$\Lambda^{AA} = \min\{\Lambda_s^{AA}, \Lambda_r^{AA}\} \quad (11)$$

여기서 $1/\Lambda^{AA} \in \alpha(1-p + p \ln K)$ 이므로, $p \rightarrow 0$ 일 때는 $\alpha(1)$ 이며, $p \rightarrow \alpha$ 일 때는 $\alpha(\ln K)$ 이다.

5.3.2 AN 전송기법

여기서는 상기에서 기술한 NAK기반의 수신자위주의 전송 프로토콜에 ASRC 개념을 도입한 전송기법("AN" 전송기법이라고 함)에서 송신자, 수신자, 대리송신자 및 대리수신자의 성능을 처리요구량 측면에서 같은 방법으로 분석해 본다. 이 기법에서도 [7]에서 기술하고 있는 수신자 위주의 전송 프로토콜의 송수신 측 처리요구량과 처리율 측면을 이용하지만, 대리송신자와 대리수신자를 이용할 경우의 좋은 성능변화를 짐작할 수 있다.

여기서는 수신자위주의 전송 프로토콜에서 송신자와 수신자 처리요구량과 전체 시스템 처리율만을 알아

보고 대리송신자와 대리수신자의 처리요구양은 위와 같은 방법을 간단히 사용할 수 있으므로 생략한다.

먼저, 송신자의 처리요구양은 하나의 패킷을 전송하기 위한 처리량과 대리송신자로부터의 NAK를 처리하기 위한 처리시간으로 생각할 수 있다. 그러므로, 이에 대한 수신자의 표현은 (12)와 같다.

$$X^{AN} = X_r + X_p(1) + \sum_{i=3}^{L_n} X_n(i) + \sum_{m=2}^{M_r} X_p(m) \quad (12)$$

이에 대한 기대치를 취하면, 식(13)과 같다.

$$E[X^{AN}] = E[X_r] + E[M_r]E[X_p] + E[L_n^*]E[X_n] \quad (13)$$

여기서 $E[L_n^*]$ 은 대리송신자로 부터의 재전송 요구를 위한 NAK의 기대되는 수이므로, $M_r - 1$ 로 표현되고, $E[M_r] = \frac{1}{(1-p)}$ 이므로, 이를 (13)식에 대입하면 식(14)와 같다.

$$E[X^{AN}] = E[X_r] + \frac{1}{(1-p)} E[X_p] + \frac{p}{(1-p)} E[X_n] \quad (14)$$

그러므로, $E[X^{AN}] \in O(\frac{1}{(1-p)})$ 가 된다. 이것의 의미로 보면, 기존의 수신자위주 프로토콜에서 송신자의 처리요구양 $O(1 + pR/(1-p))$ 에서 수신자수 R에 의존하지만, AN 전송 기법의 송신자 처리요구양은 수신자의 수와는 무관함을 보인다.

다음으로 수신자의 처리요구양을 알아보자. 이것은 [7]의 송신자위주 전송 프로토콜에서의 수신자 처리요구양과 같다. 그러므로, 식 (15)로 정의된다.

$$Y^{AN} = Y_r + P(M_r=1)Y_p(1) + P(M_r>1) \sum_{m=1}^{M_r} Y_p(m) + P(M_r>1) \sum_{m=2}^{L_n} Y_n(m) + P(M_r>2) \sum_{m=3}^{L_n} Y_r(m) \quad (15)$$

식 (15)에서 두 번째와 세 번째 항은 새로운 패킷의 올바른 수신을 위한 것이고, 네 번째와 다섯 번째 항은 이에 대한 오류 및 분실발생 시 Nak 리턴과 타이머 만료를 위한 처리요구양이다.

식 (15)에 기대치를 취하면, 식 (16)과 같다.

$$E[Y^{AN}] = E[Y_r] + E[M](1-p)E[Y_p] + P(M_r>1)(E[M_r|M_r>1]-1)E[Y_p] + P(M_r>2)(E[M_r|M_r>2]-2)E[Y_p] \quad (16)$$

식 (16)에서 [7,8]에 의해서 다음의 (17)로 정의된다.

$$E[Y^{AN}] = E[Y_r] + E[M](1-p)E[Y_p] + pE[Y_n]/(1-p) + p^2E[Y_r]/(1-p) \quad (17)$$

여기서 수신자의 처리요구양은 $E[Y^{AN}] \in O(1-p + plnk)$ 로 제한된다. 이때 처리율은 $\Lambda^{AN} = 1/E[Y^{AN}]$ 이다. 또한 전체 시스템의 최대처리율은 $\Lambda^{AN} = \min\{\Lambda_s^{AN}, \Lambda_r^{AN}\}$ 이다.

그러므로, $1/\Lambda^{AN} \in O(1-p + plnk)$ 로 제한된다. 이것은 $p \rightarrow 0$ 일 때는 $O(1)$ 이며, $p \rightarrow$ 상수일 때는 $O(\ln k)$ 로 제한된다. 이것은 기존의 수신자위주 전송프로토콜의 전체 처리율과 같다. 그러므로, 대리송수신자를 이용한 AN전송기법도 기존의 수신자위주의 전송 프로토콜과 최소한 같거나 그 이상의 성능을 볼 수 있다.

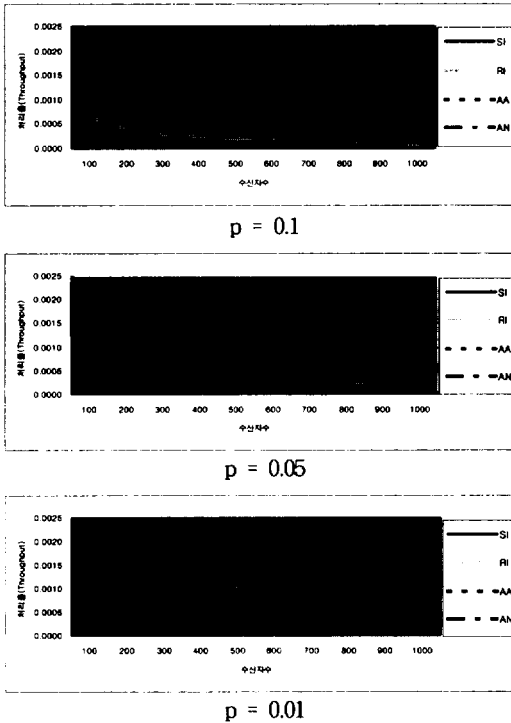
5.4 결과 분석

여기서는 이제까지 기술한 상기의 성능분석에 대한 결과를 분석하고 기존의 송수신위주의 전송 프로토콜과 ASRC전송 기법의 처리율을 도식적으로 비교 검토해 본다. 이러한 결과를 유도할 때 사용된 각 유도식의 매개변수들을 처리하기 위해서 각 매개변수를 <표 2>와 같이 적용한다. 이것은 [10]에서 사용한 펜티엄 PC에서 각각의 처리시간들을 반복적으로 실험한 측정치를 이용한다. 또한 대리수신자수 W는 64로 가정하면, 각 그룹의 수신자수는 $\lceil \frac{R-1}{W} \rceil$ 로 주어진다.

<표 2> 처리 시간들(microsecond 단위)
<Table 2> processing time

X_p	502	13.8
X_m, X_a	87	5.8
Y_p	487	12.6
Y_m, Y_a	86	5.1
X_p, Y_r	32	1.5
X_p, Y_f	22	1

<표 2>의 매개변수를 이용하여 처리율을 비교하면 (그림 6)과 같은 결과를 얻을 수 있다.



(그림 6) 손실율에 따른 처리율 비교

(Fig. 6) throughput Comparisons on a several loss rates

(그림 6)에 따르면, 기본적으로 AA와 AN 전송 기법은 기존의 송수신자 위주의 전송 프로토콜에 대한 처리율에 비해 현격한 차이를 볼 수 있다. 이 결과에서 AA와 AN 전송 기법의 처리율은 손실율(p)에는 거의 무관함을 보이고 있다. 또한, 여기서도 AA보다는 AN 전송 기법의 처리율이 다소 좋은 것을 알 수 있다. 이러한 결과는 별도의 시스템용 제어 구조는 멀티캐스트 망의 송수신자들의 최대처리율을 현저히 줄일 수 있음을 보이고 있다.

6. 결 론

본 논문에서는 신뢰성 있는 패킷 전송을 위한 기존의 프로토콜에 대한 새로운 전송 기법을 제안하고 송수신자들의 최대처리율 측면에서 분석하였다. 이를 위해서 본 논문에서는 ASRC 개념의 기본 개념, 원리 및 구조를 제안하고, 이를 일반적인 전송 프로토콜인 송신자 및 수신자위주의 프로토콜에 적용하고, 이에 따른 동작

을 기술하였다. ASRC 개념은 멀티캐스트 망의 송수신자는 순수한 의미의 데이터 송수신만을 수행하도록 하고, 이에 따라 발생하는 각종 오류에 따른 재전송 제어 기능은 독립적으로 분리하여 별도의 제어측면에서 관리함으로써, 신뢰성 있는 다자간 통신환경의 트래픽 부하의 분산 및 제어의 전문성을 도모하도록 하자는 것이다.

분석 결과, [7]에서 분석한 기존의 송신자 위주 전송 프로토콜에서 최대처리율을 상당 부분 향상시킬 수 있었다. 또한 수신자 위주 프로토콜에서의 최대 처리율에 대해서는 같거나 향상된 결과를 얻을 수 있었다.

이러한 결과는 멀티캐스트 환경내의 신뢰성 보장을 위한 각종 메카니즘의 개발 보다 별도의 신뢰할 수 있는 제어 시스템의 필요성과 타당성을 보인 것이다.

또한 본 논문에서 제안된 ASRC 전송 기법은 멀티캐스트 통신환경을 위한 별도의 시스템들로 구성된 구조적인 관리플랫폼의 개발에 대한 타당성을 입증한 것이다. 이로서 본 논문에서는 "멀티캐스트 통신환경에서 데이터송수신과 이에 따른 제어 및 관리의 기능을 상호 독립시킬 수 있다"는 결론을 기본으로 향후의 연구 과제를 추출할 수 있었다.

또한 멀티캐스트 통신환경에서는 신뢰성 보장을 위한 송신자와 수신자의 제어정보 폭주로 인한 문제뿐만 아니라 확장성(scalability)문제, QoS(Quality of Service)관리 문제, 순서제어문제(Total Ordering), 그룹관리 문제, 세션관리문제 등 다양한 문제들을 남겨 두고 있다. 물론 이에 대한 많은 연구들이 진행되고 있는 실정이다.

그러므로, 본 연구의 향후 연구 과제는 멀티캐스트 통신환경에서 사용자 노드들의 기능에 무관하게 기존 문제들을 통합적으로 제어하고 관리하기 위한 구조에 대한 연구, 이에 대한 관리 플랫폼설계 및 개발에 대한 연구이다. 이를 위해 망관리 시스템 영역의 연구가 병행되어야 하며, SNMP 관리 프로토콜과 멀티캐스트 통신환경을 위한 관리 대상의 MIB 구축 등의 연구도 진행할 예정이다. 또한 멀티캐스트 통신환경에 대한 관리영역의 연구도 필요할 것이다.

이러한 향후 연구과제는 다양하고 새로운 장점을 가질 수 있을 것이다.

참 고 문 헌

[1] T.Maufer, C.Semeria, "Introduction to IP Multi-

cast Routing," IETF, draft-ietf-mboned-intro-multicast-03.txt, Oct. 1997.

[2] RFC 1112, "Host Extensions for IP Multicasting," S.E.Deering, Aug. 1989.

[3] RFC 1889, "RTP-A Transport Protocol for Real-time Applications," Audio-video Transport Working Group, H.Schulzrinne, S.Casner, R.Frederrick, and V.Jacobson. Jan. 1996.

[4] RFC 1301, "Multicast Transport Protocol," S. Armstrong, A.Freier, and K.Mareullo. Feb. 1992.

[5] Todd Montgomery, "Design, Implementation, and Verification of Reliable Multicast Protocol," MS Thesis, West Virginia University, 1994.

[6] ISO/IEC JTCl/SC6, "Consolidated Text out of ECTS FCD ballot voting,"

[7] S.Pingali, D.Towley, and J.F.Kurose, "A Comparison of sender-initiated and Receiver-initiated reliable multicast protocols," Performance Evaluation Review, Vol.22, pp.221-230, May. 1994.

[8] Brain Heil Levine, "A Comparison of known classes of Reliable Multicast protocols," MS thesis, University of California, June 1996.

[9] Markus Hofmann, "A Generic concept for Large-Scale Multicast," Proceeding of 1996 International Zurich Seminar on Digital Communications, Springer Verlag, Feb. 1996.

[10] Sneha K.Kasera, Jim Kurose, and Don Towley, "Scalable Reliable Multicast Using Multiple Multicast Groups," CMPSIC Technical Report, Oct. 1996.

[11] ITU-T, M.3010, "Principles for a Telecommunications Management Network".

[12] RFC 1157, "A Simple Network Management Protocol(SNMP)," J.case, et. al. May. 1990.



안 병 호

e-mail : bhahn@cccc.chch-c.ac.kr
 1986년 광운대학교 전자계산학과 졸업(이학사)
 1988년 광운대학교 대학원 전자계산학과(이학석사)
 1998년 광운대학교 대학원 전자계산학과(박사수료)

1988년~1994년 한국전자통신연구소(ETRI) 선임연구원
 1994년~현재 충청대학 전자계산과 교수
 관심분야 : 망관리, 멀티캐스트 통신구조, 고속통신망, 성능평가 등



조 국 현

e-mail : khcho@cs.kwangwoon.ac.kr
 1977년 한양대학교 전자공학과(공학사)
 1984년 일본 동북대학교 대학원(공학박사)
 1984년~현재 광운대학교 교수, 광운대학교 이과대학 학장, 전자계산소장 역임

1999년 현재 전산대학원장, 한국정보과학회 이사, 개방형 컴퓨터 통신연구회 부회장 역임, 현재 개방형 컴퓨터 통신연구회 회장
 관심분야 : 정보통신망 관리, 분산처리, 멀티 캐스팅 및 정보통신분야의 표준화