

서버라우터를 이용한 멀티캐스트 활용

최원혁*, 이광재**, 최준*, 김정선*
한국항공대학교 전자공학과*
서남대학교 전기 전자 멀티미디어 공학부**

The Use of Multicast for Server Router

won-hyuck choi, Kwang-Jae Lee, joon choi, Jung-Sun Kim
Department of Avionics, Hankuk Aviation University
Department of Electrics, Electronics and Multimedia, Seonam University
rbooo@korea.com

요약

IP 멀티캐스트는 다자간 통신을 위한 효과적인 방법이지만 신뢰성 보장과 FTP나 Telnet등 TCP제공하는 다양한 전송프로토콜을 지원하지 않는다. 본 논문에서는 멀티캐스트의 UDP와 TCP계층의 전송 프로토콜을 동시에 사용할 수 있는 신뢰성 있는 멀티캐스트 서버라우터를 제안 한다. 신뢰성 있는 멀티캐스트가 반드시 고려해야 할 확장성과 오류회복, 흐름제어를 위해, 그룹의 확장성과 오류회복에 적합한 기존의 SRM방식을 이용하였다. SRM방식은 신뢰성 있는 다-대-다 멀티캐스트 패킷을 전달한다. 각 참여자는 동시에 수신자와 송신자 역할을 할 수 있다. 모든 그룹 멤버는 낮은 주파수의 세션 메시지를 교환하고 그룹내의 주변 가입자에 대한 정보를 얻게 되며 패킷 손실 판단에 필요한 지연시간을 측정하고 복구하는 방식이다. UDP의 트래픽 기반은 멀티캐스트에 적합한 CBR과 SRM을 사용하였다. 본 연구에서는 멀티캐스트의 UDP 패킷과 TCP 패킷을 동시에 보낸 수신율과 멀티캐스트의 UDP만 보낸 수신율을 지연이 가장 큰 멀티캐스트 수신자측에서 패킷 번호와 수신율을 측정하고, 시뮬레이션을 통해서 검증을 통하여 기존의 멀티캐스트 전송방법과 비교하였다.

1. 서론

최근 멀티미디어 워크스테이션, 분산시스템, 고속통신 시스템이 크게 발전하고 있다. 이들의 결합은 분산 멀티미디어 시스템에 대한 연구개발을 촉진시켰으며, 워크스테이션 뿐 만 아니라 개인용 컴퓨터에서도 텍스트 외에 음성, 화상, 오디오, 이미지, 그래픽 등의 신호를 송수신할 수 있게 되었다. 이에 따라 화상회의의 가상현실, 원격 진단시스템, 주문형 비디오 등과 같은 여러 용용들에서 각종 데이터 종류를 이용하는 멀티미디어 통신이 급격한 일반화의 양상을 보이게 되었다. 또한 시스템 및 네트워크 성능에 따라 사용자들의 기대 수준이 높아지면서 요구하는 서비스 특성이 다양해 졌으며, 각각의 특성들이 상호 복합적으로 적용되어 사용자의 요구사항을 충족시켜 주고 있다. 이러한 서비스는 네트워크와 수송계층에서 이루어지는 데이터 전송의 하부 구조로 볼 수 있다. 각종 서비스 특성들을 지원할 수 있는 기능이 전송 프로토콜에

정의 및 구현되어 있을 때, 이에 기반 하여 사용자의 요구 사항을 충족시켜주는 응용서비스를 개발할 수 있다. 필요하다면 수송계층의 상위계층에서 적절한 역할을 수행할 수 있어야 하며, 이에 따라 궁극적으로 사용자가 각종 서비스 특성을 제공받는 멀티미디어 응용 서비스를 활용할 수 있게 된다. 각종 멀티미디어 응용 서비스들이 다자간 응용서비스를 요구함에 따라 이의 실현을 위해 기존의 IPv4 네트워크에서는 RFC1112[1]를 통하여 그룹관리 프로토콜로서 IGMP와 함께 IP 멀티캐스트 기능을 제공하고 있다. 차세대 IP 프로토콜로서 일컬어지는 IPv6에서는 IGMPv6 와 함께 멀티캐스트 기능을 제공한다. IP 멀티캐스트를 통해 하나의 송신자는 그룹에 가입해 있는 모든 참여자에게 멀티캐스트 데이터를 제공해 줄 수 있으며, 이는 네트워크의 상위 계층인 수송계층 전송 서비스를 위한 기본 기능이라 할 수 있다. 현재 일-대-일 통신 서비스로부터 일-대-다 또는 다-대-다 통신서비스에서 신뢰성을 요구하는 새로운 요구가 많아지고 있지

만, IP 멀티캐스트의 신뢰성만으로는 이러한 서비스를 제공할 수 없는 상황이기 때문에 수송계층에서 신뢰성을 제공하고자 하는 방식이 제안된다. 또한 멀티캐스팅 라우터 자체가 UDP와 같은 데이터 그램 단위의 그룹 엔트리만을 처리하기 때문에 TCP에서 사용되는 일대일 방식의 전송 프로토콜 서비스는 멀티캐스트 라우터에서 사용이 제한적이다. 본 연구에서는 서버라우터를 이용한 각 전송 프로토콜의 데이터 전송을 위한 공정하고 실용적인 대역폭 사용하기 위해서 UDP와 TCP를 각 정해진 시간에 송신하여 서버라우터로부터 공정한 대역폭 전송할 수 있는 라우터를 시뮬레이션 상에서 구현하고 성능을 평가한다. 한편, 이후 본 논문의 2장에서는 신뢰성 있는 멀티캐스트의 요구 조건과 전송 프로토콜에 대해서 알아보며, 3장에서는 본 논문에서 제안한 서버라우터에 대해서 설명한다. 4장에서는 3장에서 제안한 방식에 대한 시뮬레이션 결과와 성능을 분석하고 결론을 내린다.

2장 신뢰성 있는 멀티캐스트

신뢰성을 제공하는 멀티캐스트를 위해 제안된 방법으로는 크게 SRM, RMPT, PGM 방식으로 나눌 수 있다. SRM 방식은 NACK 발생 가능성 즐여 최소한의 NACK가 송신자에게 전달되도록 하는 방식이며 RMPT, PGM 방식은 수신자를 다수의 그룹으로 나누어 관리하는 계층적 그룹기반의 방식[18]이다. 신뢰성 있는 멀티캐스트 요구조건에는 확장성, 오류제어, 응답 패킷 타입으로 나눌 수 있다. 신뢰성 있는 멀티캐스트의 확장성은 일-대-다 통신에서 대단히 많은 수 송신자가 존재한다고 할 때 모든 수신자가 신뢰성을 제공받기 위해서 수신 패킷에 대한 응답을 송신자에게 전달, 이는 송신자의 동작 상태를 포화상태로 이르게 하여 송수신 중단 상태를 초래할 수 이있다. 따라서 그룹의 많은 참여자가 존재하여도 참여자의 수와는 무관해야 한다. 오류제어는 사용된 제어구조가 어떤 방식이냐에 따라서 제어동작이 결정된다 오류제어 방식에는 중앙 집중식 오류제어 와 분산 오류제어 두 가지 방법으로 나눌 수 있다. 현재 멀티캐스트 오류제어 방식에는 송신자의 오류복구 부담을 감소시키는 분산식 오류제어 방식이 주류를 이루고 있다. 분산식 오류제어 방식은 다시 지역그룹을 구성하는지 여부에 따라서 구분될 수 있으며 재전송이 지역그룹 내에서만 발생하는 지역그룹 방법은 불필요한 재전송을 감소시키는 방법으로서 효과적인 방법이라 할 수 있다. 신뢰성 있는 멀티캐스터는 오류를 제어하기 위해서 여러

가지 응답 패킷 타입을 사용한다. 현재 제안된 신뢰성 있는 멀티캐스트는 ACK, NACK, ACK+NACK, SACK 등의 응답 메시지를 사용하는지에 따라서 멀티캐스트 성능이 달라질 수 있지만 응답 형태는 그룹구조 형성에 매우 의존적인 특성을 가지고 있기 때문에 그룹구조를 논하지 않고 별개로 어떤 응답 메시지 방식이 좋다는 결정을 내리기는 어렵다. NACK 기반의 방식은 손실이 발생하였을 때 NACK가 그룹에 멀티캐스트 되며 NACK를 수신한 가입자는 동일한 패킷에 대한 NACK를 발생하지 않는 방식으로서 두 가지 조건이 만족되는 상태에서 NACK는 ACK 보다 효율적인 방식이라 할 수 있다. 첫째로 NACK 전송에 신뢰성이 보장되는 상태로서 NACK의 손실 비도가 높은 망에서는 효과적인 방식이 될 수 없는 특징이 있다. 둘째로 수신자간의 패킷 교환 지연시간이 송신자와 수신자사이의 지연시간과 비교하여 작을 때만 정상적인 동작을 할 수 있다. 또한 NACK 기반의 방식은 비교적 큰 버퍼를 필요로 하는 단점이 있다. ACK 기반의 방식은 ACK에 대한 추가적인 처리가 없다면 ACK 폭주를 송신자에게 가져올 수 있는 심각한 단점이 있으며 NACK에 비해 상대적으로 더 많은 메시지를 발생한다. 그러나 망 내부 가입자 또는 라우터의 속도와 큰 버퍼를 가진다면 ACK 폭주를 예방할 수 있어 송신자에게 유니캐스트와 유사한 환경을 제공할 수 있다.

2.1 서버라우터의 TCP와 UDP 요청

서버라우터는 두 가지 의미로 설계되었다. 멀티캐스트 전송과 TCP 전송의 적합성과 멀티캐스트와 TCP의 원활한 소통의 호환성을 측정할 수 있다. 서버라우터에서 우리가 logical tree 저장 패킷을 송신자의 버퍼들로부터 수신을 하고 그리고 timer를 놓고 재배치를 호출하고 다른 패킷을 버퍼로 저장한다. 송신자는 또한 재배치 timer를 배치하여 다른 패킷을 전송한다. 다른 수신자는 명확한 확인(ACK)을 보내며 혹은 부정응답(NACK)은 트리의 부모로 보낸다. 받은 패킷들은 ACKs로 보고하고 그리고 잃어버린 패킷을 NACKs로 보고한다. 송신자는 모든 차식으로부터 ACK로 받았을 때 버퍼를 버린다. 반면에, 송신자는 패킷을 유니캐스트를 통하여 NACK로 보고 받았을 때 재전송을 하는데 패킷을 잃어버리거나 혹은 만약 받지 못했다면 ACK의 패킷이 logical tree 전에 timer의 조합과 함께 패킷은 소멸시킨다. TCP와 UDP의 혼잡현상을 해결하기 위해서 서버라우터의

TCP에서는 패킷이 수신자들에게 전송되었을 때 전송된 바이트들을 평가하기 위해 TCD(transmission control determine)를 사용하였다. 멀티캐스트로 전송되는 패킷의 혼잡도를 평가하기 위해서 UDD(user datagram determine)를 사용하였다. 서버라우터는 항상 전송된 데이터 패킷의 총계Dt0l(Data total)에서 TCD와 UDD의 값의 차이를 비교하여 혼잡정도를 평가하여 혼잡정도가 심할 때 멀티캐스트 패킷을 먼저 처리할 수 있도록 하였다. 이때 TCP의 상대 지연의 패킷량은 송신자로부터 계속 수신된 feedback에 의거하여 UDD가 유지된다. 이는 송신자의 전송이 ACK의 패킷에 의하여 계속 노드가 유지된다.

3장 서버라우터 개요

멀티캐스트의 가장 큰 장점은 같은 데이터의 중복 전송으로 인한 네트워크의 대역폭의 낭비를 줄일 수 있다. 그리고 네트워크의 혼잡상황이 발생할 가능성을 줄일 수 있다. 즉, 멀티캐스트는 네트워크 사용의 효율을 극대화할 수 있다. 멀티캐스트를 네트워크 상에서 모두 지원한다고 가정하면, 네트워크 사용의 효율성을 높이기 때문에 비용 적인 감소는 매우 크다. 또한 송신자 쪽에서는 하나의 데이터를 보내면 되기 때문에 크기가 큰 멀티미디어 데이터도 동시에 여러 수신자에게 보낼 수 있고, 송신자 서버의 부하도 크게 줄일 수 있다. 이러한 특성은 인터넷방송, 그리고 인터넷을 통한 화상회의 등에 필요한 스트리밍 서비스를 가능하게 한다. 그러나 전송계층에서의 멀티캐스트 프로토콜은 UDP 기반의 서비스를 한다. 그래서 TCP전송에서의 FTP(와 Telnet같은 신뢰성 있는 프로토콜을 전송하지 못하며 보다 다양한 네트워크환경에서 사용이 제안되어있다. 즉 라우터가 다양한 전송 방식을 가지지 못하는 단점이 있다. 본 논문은 앞에서 제기된 기존의 네트워크 상에서의 라우터의 다양한 전송프로토콜 지원문제를 해결하고 서버라우터(Server Router)를 이용한 신뢰성 있는 네트워크 환경에서의 전송을 위한 방식을 제안한다. 실제 네트워크 환경에서는 UDP뿐만 아니라 TCP프로토콜 응용계층의 사용자 명령에 따라 한 시스템에 있는 파일을 다른 시스템으로 전송하고 터미널이나 개인용 컴퓨터로 원격 컴퓨터에 접속하여 사용하는 FTP, Telnet와 같은 일-대-일 패킷 전송방식과 여러 처리에서 효과적인 TCP의 공격적인 프로토콜의 다양한 트래픽을 고려해야 한다. 서버라우터를 사용하면 송신자에서 수신자 쪽으로 TCP와 UDP 프로토콜의 전송 특징을 동시

에 만족할 수 있고 네트워크 환경에서 프로토콜을 좀더 효과적으로 사용될 수 있다.

3.1 서버라우터의 구성

서버라우터는 송신자로부터 전송되는 데이터의 전송방식을 확인한 후 라우터는 각 방식에 따른 전송을 시작한다. 서버쪽 라우터에서 전송방식에 따른 알고리즘은 다음과 같다. 멀티캐스팅부분은 멀티캐스팅 스위칭, 멀티캐스터 그룹의 엔트리를 저장할 수 있는 공간과 Join할 수 있는 공간설정과 시뮬레이션에서 정의된 멀티캐스트 트래픽 CBR, SRC을 사용하였고 IP 패킷의 플래그가 0 이될 때 멀티캐스트가 실행되도록 하였다. 그리고 신뢰성 있는 멀티캐스트를 사용하기 위해 SRM의 프로토콜을 사용하였다. TCP 전송 트래픽으로 FTP, Telnet를 사용하였고 TCP가 전송된 다음 ACK의 전송을 위해서 TCP point를 사용, 패킷이 수신자에게 전송된 뒤 ACK를 다시 전송하도록 하였다. 서버라우터서 TCP와 멀티캐스트 전송인 UDP를 동시하기 위해 두 개의 버퍼를 두었다. 이중 멀티캐스트인 경우 그룹을 관리하고 처리해야하기 때문에 보다 큰공간 두어야한다. 그리고 상대 지연시간을 측정하기 위해 송신 쪽에서 전송된 패킷은 라우터 자체에 Timer를 설치해서 재전송이 가능하게 하였다. 이 때 패킷은 수신자와 시간에 의하여 전송되고 송신자는 자식노드로부터 ACK의 수신을 예상한다. 이는 시간 주기의 측정이 어려운 문제 때문이다. 왜냐하면 송수신자 사이에 clock이 다르기 때문이다. 송신자는 timer가 소멸하여도 패킷은 재전송을 가능하다. 그리고 멀티캐스트 전송인 경우 다시 재 전송된 패킷을 timer의 재배치 해야하는데 이것을 서버라우터의 멀티캐스트 처리 부에서 재배치하도록 하였다. TCP 처리인 경우 패킷들은 손실에 의해 자식과 재전송 단지 그것들은 결정 할 수 있다. 불필요한 재전송은 피하기 위해서 서버라우터의 TCP 처리 부에서는 다른 ACK를 보내는 시간 중에 송신자에서 전송하는 마지막 패킷을 전송하고 버퍼를 이용할 수 있다.

3.1.1 서버라우터의 TCP 전송

서버라우터 TCP 전송은 송신자의 재전송 timer에 의해 다른 패킷에 의해 수신한다. timer는 패킷 전송 주기 사이의 시간을 의미한다. 패킷은 수신자와 시간에 의하여 전송되고 송신자는 자식으로부터 ACK의 수신을 예상한다. 이 시간 주기는 측정이 어려운데 왜냐하면 송신자와 수신자 사이에 clock이 다르기 때문

이다. 이 어려움을 극복하려면 clock 가치 사이의 차이점을 정의, ACK 할 때 자식으로부터 패킷을 수신, clock의 가중치, 수신자의 사용, 비슷한 패킷을 전송 할 시점이 중요하다. 전송시간은 TCP가 다른 데이터 패킷이 운반되는 시간이다. 그리고 다른 ACK운반은 전송시간 마지막 패킷을 수신하고 재전송을 하는 RTT와 비슷한 방법을 사용한다. 이를테면 최소의 알맞은 RTD와 일반적인 RTD는 수신자를 이용하여 다수의 패킷들을 통과 자식노드의 송신자로부터 평가한다.

3.1.2 서버라우터의 멀티캐스트 전송부

멀티캐스트부의 유니캐스트 전송은 송신자에서 수신자로 패킷이 전송될 때 서버라우터 멀티캐스트 그룹을 단시간 저장한다. 서버라우터의 멀티캐스트 흐름제어를 하기 위해 전송하는 패킷의 양을 수용력에 맞게 수신자에 많은 수의 버퍼들을 유용하게 사용하기 위해 송신자가 많은 데이터를 보내지 않게 지킨다. 즉 TCP와 멀티캐스트가 동시에 전송될 때 서버라우터에서 데이터량을 조절한다. 수신자는 다수의 버퍼 사용을 서버라우터를 통해 유용한 송신자에게 전달한다. 최소의 버퍼를 수신자에서 사용하기 위해 우리는 평균 버퍼를 정의한다. 송신자는 평균 버퍼를 TCP의 윈도우로 보낸 후 버퍼수가 증가하면 윈도우 의해 패킷량을 조절해서 흐름제어를 한다. 멀티캐스트의 혼잡제어 순차적 프드백 방식의 NACK이다. 그므로 송신자에 의하여 새로운 패킷을 받을 때 NACK를 수신자에서 송신측으로 보낸다. 그리고 만약 NACK가 교착상태의 경우에는 윈도우 사이에의 패킷에 의해 몇 개의 패킷을 전송 교환할 수 있다. 그리고 하나의 서버라우터에 IP 주소 중 멀티캐스트와 TCP를 인식할 수 있는 IP필드를 삽입하여 멀티캐스트와 TCP가 동시에 전송되더라도 서버라우터에서 멀티캐스트와 TCP를 구분할 수 있게 된다.

4장. 시뮬레이션 환경 및 결과

4.1 시뮬레이션 환경

시뮬레이션은 200초간 실시하였으며 수신자를 수를 변화시키면서 수신율을 측정하였다. 시뮬레이션 포트폴로지는 그림2와 같은 구조를 이용하였다. 그룹A의 경우 N1 N4까지의 네트워크 망을 사용하고 N2에서 그룹A2의 U1으로 서버라우터를 통해 이동된다. 또한 그룹A1의 자노드 N4에서도 그룹A1의 자노드 N4에서도 그룹A3의 G1으로 서버라우터를 사용해서

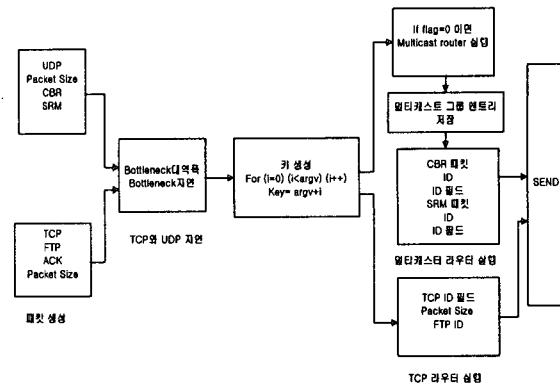


그림 1 서버라우터 구성도

이동하는 범위가 큰 네트워크 토플로지를 사용하여 좀더 실질적인 네트워크 구조를 설정하고 각 그룹들은 서버라우터로 공유하고 있다. 송신자 Sender에서는 TCP와 멀티캐스트 전송을 동시에 전송한다. 실험 1에서는 TCP와 UDP를 전송했을 때 각 그룹에서 전송된 데이터 전송률의 변화와 그룹에서 TCP 전송을 증가 시켰을 때의 데이터 전송률 변화를 살펴본다 그리고 TCP, UDP 데이터가 동시에 전송될 때 멀티캐스팅 방식에서 TCP와 UDP가 정당하게 대역폭을 공유할 수 있음을 증명하였다. 신뢰성 있는 멀티캐스트를 위해서 SRM방식을 도입하여 멀티캐스터를 좀더 효과적으로 사용하였고 TCP환경에서는 FTP, Telnet을 사용하였다. 시뮬레이션 도구는 TCP/IP 환경을 잘 반영하는 ns-2 (Network Simulator-II)를 사용하였고 시뮬레이션의 공정성을 기하기 위해 랜덤 값을 이용하여 각 방식별 100번을 반복 수행한 후 평균을 결과값으로 채택하였다.

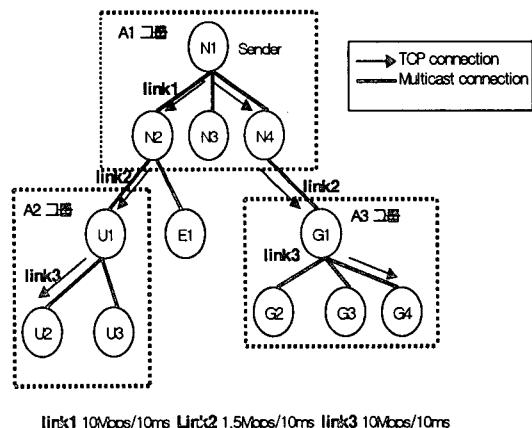


그림 2 네트워크 토플로지

4.2 시뮬레이션 결과

실험1은 각 그룹이 시간에 따른 데이터 전송의 변화를 관찰한다. 대역폭과 지연시간(propagation delay)은 좀더 실질적인 네트환경을 구축하기 위해 송신 측 그룹A1에서 하위 그룹으로 보낼 때 지연시간과 대역폭은 10ms, 1.0mb 공유된 라우터와 라우터 사이는 10ms, 0.5mb, 수신 측은 각각 10ms, 1.0mb로 다양하게 두었다. 이는 SRM 기반의 서버라우터 특성상 지연시간이 다양하고, 서버라우터와 라우터 사이에서 혼잡으로 인한 손실이 일어나게 하기 위함이다. 그럼 3는 서버라우터 상에서 TCP와 멀티캐스트가 동시에 전송 때와 멀티캐스트만 전송될 때 수신자수에 따른 수신율 변화를 나타내는 그래프이다. 그래프에 볼 수 있듯이 멀티캐스트와 TCP를 동시에 전송한 멀티캐스트의 수신율이 유사하게 전송됨을 알 수 있다.

실험2는 TCP와 멀티캐스트 패킷을 동시에 전송했을 때의 TCP수신자와 멀티캐스트 수신자가 받은 패킷 번호의 누적된 결과이다. 아래 그래프에서 나타나듯이 TCP와 멀티캐스트 수신자의 수신특성의 차이가 크지 않음을 볼 수 있다.

4.3 결론 및 향후 연구

현재 다양한 전송방식의 라우터가 설계되고 있고 많은 응용프로그램이 개발되어 서비스되고 있으며 이에 따라 많은 연구들이 수행되고 있다. 이러한 다양한 서비스를 제공해주는 하부 구조가 네트워크와 수송계층에서의 데이터 전송 부분이라 할 수 있다. 다양한 서비스 특성을 지원할 수 있는 기능이 전송 프로토콜에 구현되어 있을 때 이를 기반으로 하여 사용자는 응용 서비스를 개발할 수 있다. 그러나 현재의 멀티미디어를 주축으로 한 통신방식에서 멀티캐스트 방식은 신뢰성과 다양한 전송 방식을 제공하지 못하는 단점이 있다. 본 연구에서는 서버라우터를 사용한 다양한 방식의 전송계층의 프로토콜인 TCP의 패킷과 멀티캐스트전송 방식인 UDP 패킷을 동시에 전송할 수 있는 신뢰성 있는 멀티캐스트의 새로운 방식을 제안하였다. 즉, 현 멀티캐스트에서 UDP 패킷만을 전송하는 전송 프로토콜방식과 달리 서버라우터에서는 TCP의 IP 패킷을 처리할 수 있는 라우터 처리부분을 두고 TCP와 멀티캐스트의 UDP 패킷을 동시에 처리하여 수신자에게 전송할 수 있도록 하는 네트워크 전송 환경을 구축하였다. TCP의 경우 송신 측의 TCP 생성 패킷번호에 포인터를 두고 TCP 패킷의 넘버를 3개씩 나누어 응답 패킷인 ACK를 이용한 흐름제어를 할 수 있

도록 하였다. 패킷의 넘버를 TCP와 멀티캐스트가 전송된 패킷 넘버를 구분하는 방식으로 측정하였다. 서버라우터의 병목현상 문제에 대해서 지연시간과 대역폭의 크기를 두어 좀더 효과적인 네트워크 환경을 구축하였다. 멀티캐스트의 평균 수신율과 TCP, 멀티캐스트 조합의 평균 수신율이 성능평가에서 서로 유사함을 보였다. 그리고 TCP의 패킷과 멀티캐스트전송의 UDP 패킷의 평균 수신율 비교에서도 서로 큰 차이를 보이지 않았다. 또한 SRM 방식의 신뢰성 있는 멀티캐스팅을 함께 사용하고 멀티캐스팅의 단점인 흐름제어를 탄력적으로 해결함으로서 신뢰성을 높일 수 있는 장점을 보여주었다.

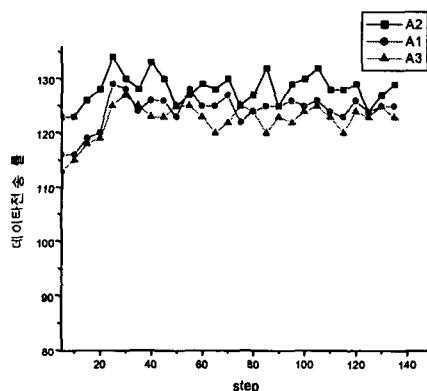


그림 3 네트워크 그룹에서 데이터 전송률

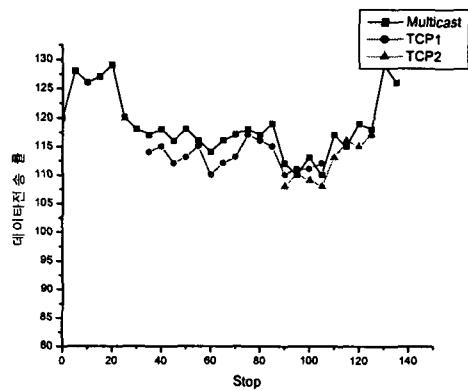


그림 4 TCP 증가에 따른 데이터 전송률

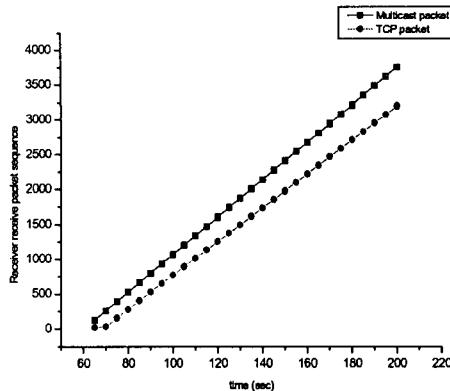


그림 5 멀티캐스트와 TCP의 패킷 넘버

of INFOCOM' 96, IEEE, March 1996.

- [8] S. Bhattacharyya, D. Towsley, and J. Kurose, "The Loss Path Multiplicity Problem in Multicast Congestion Control", Proc. IEEE INFOCOM' 99, pp.856-863, New York, USA, March 1999.
- [9] S. K. Kasera, S. Bhattacharyya, M. Keaton, D. kiwior, S. Zabele, J. Kurose, and D. Towsley, "Scalable Fair Reliable Multicast Using Active Services", IEEE Network, vol.14, no.1, January/February 2000
- [10] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance". IEEE/ACM Trans. Networking, vol.1, no.4, pp.397-413, August 1993.

References

- [1] B.N. Levine, D.B. Lavo, J.J. Garcia-Luna-Aceves, The case for reliable concurrent multicasting using shared ack trees, in: Proceedings of Multimedia 1996, ACM. New York, 1996, pp.365-376
- [2] S. Paul, k.k. Sabnani, J.C. Lin, S. Bhattacharyya, Reliable multicast transport protocol (RMTP), Proceedings of INFOCOM' 96, IEEE, March 1996.
- [3] M.Hofmann, Adding scalability to transport level multicast, in: Proceedings of ThirdCOST237 Workshop-Multimedia Telecommunications and Applications, Springer, Berlin, 1996.
- [4] S. Floyd, Requirements for congestion control for reliable Multicast, The Reliable Multicast Research Group Meeting in Cannes, September 1997.
- [5] L. Vicisano, L Rizzo, J. Crowcroft, TCP-like congestion control for layered multicast data transfer, in: Proceedings of INFOCOM' 98, IEEE, April 1998.
- [6] D. DeLucia, K. Obraczka, Multicast feedback suppression using representatives, in: Proceedings of INFOCOM' 97, IEEE, April 1997
- [7] E. Zegura, K. Calvert, S. Bhallacharjee, How to model an internetwork, in: Proceedings