

신뢰적인 멀티캐스트 전송 프로토콜을 위한 Top-Down 기반의 제어 트리 구축 방안

(A Top-down based Control Tree Construction Mechanism for Reliable Multicast Transport Protocols)

김은숙[†] 고석주[†] 강신각[†] 최종원^{††}
(Eunsook Kim) (Seok Joo Koh) (Shin Gak Kang) (Jongwon Choe)

요약 최선의 전송(Best Effort) 서비스를 제공하는 현재의 IP 멀티캐스트 서비스의 특성상 신뢰전송을 요구하는 각종 응용들의 요구를 만족시키기 위해서는 추가적인 신뢰전송 프로토콜이 필요하다. 이러한 요구에 따라 신뢰적인 멀티캐스트 전송 프로토콜에 대한 연구가 수행되고 있는 가운데, 계층적 트리를 구축하여 신뢰성을 보장하는 노력이 활발하게 진행되고 있다. 계층적 트리 기반의 방식은 높은 확장성을 보장하면서 효율적으로 신뢰성을 보장하지만 전송단계에서 효율적인 논리 트리 구축 방안이 제공되어야 한다. 논리적인 제어 트리 구축은 수신자 기반의 상향식(bottom-up) 방식이 주로 사용되어 왔으나 이 방법은 병렬적 트리 구성을 통하여 신속한 트리 구축을 할 수 있다는 장점이 있지만 제어 트리 상에 루프(loop)가 발생할 수 있다는 단점과 메시지 부하가 커지는 단점이 있다. 이에 본 논문은 하향식(top-down) 기반의 제어 트리 구축 방안을 제안한다. 하향식 방식은 단계적인 트리 구축을 통하여 루프 발생을 방지할 수 있다. 또한 성능평가를 통하여 메시지 부하를 줄일 수 있다는 것을 보였다. 본 논문은 응용의 요구 사항에 맞추어 상향식과 하향식을 선택적으로 사용할 것을 제안한다.

Abstract To meet the requirements of reliable service for various applications, a Reliable Multicast Transport Protocol should be implemented over IP Multicast where currently best-effort service is provided. Among the current researches, hierarchical tree-based mechanism has been proposed and actively studied. This mechanism is known to provide high scalability as well as reliability, but needs an additional tree configuring mechanism for building an efficient logical tree in transport layer. Bottom-up approach has been used for creating such a tree. This method has benefits from parallel tree construction for receivers, while it has some drawbacks such that it does not guarantee a loop-free tree and brings heavy message overhead during tree creation process. Therefore, this paper proposes a top-down based mechanism for constructing a control tree, which can guarantee loop-freeness by step-wise mannered tree building. From experimental simulations, it shows that the proposed mechanism has less message overhead. It is recommended that the bottom-up and the proposed top-down will be selectively used in real networks, according to the requirements of the concerned multicast applications.

1. 서론

인터넷 방송, 다중 화상회의, 소프트웨어 분배 등 다

중 전송을 이용하는 인터넷 응용이 증가하면서 IP 멀티캐스트 서비스에 대한 요구가 증가하고 있다. 그러나 현재의 IP 멀티캐스트[1]는 Best Effort 서비스를 제공하기 때문에 원격 강의, 공동 문서 작업 등 신뢰전송을 요구하는 각종 응용들의 요구를 만족시키기 위해서는 추가적인 신뢰전송 프로토콜이 필요하다. 이러한 요구에 따라 신뢰전송 멀티캐스트 프로토콜에 대한 연구가 수년간 수행되어 왔고[2, 3] 1998년 4월부터 IETF의 RMT-WG를 중심으로 이 분야의 표준화 연구가 진행

[†] 비회원 : 한국전자통신연구원 표준연구센터 연구원
eunah@etri.re.kr
sjkoh@etri.re.kr
sgkang@etri.re.kr

^{††} 종신회원 : 숙명여자대학교 전산학과 교수
choejn@sookmyung.ac.kr

논문접수 : 2001년 2월 28일
심사완료 : 2001년 9월 6일

되고 있다. IETF RMT-WG에서는 기존의 연구들을 종합하고 새로운 요구사항을 검토하여 현재 신뢰전송 프로토콜에 관한 연구들을 크게 LCT(Layered Coding Transport)[4], NORM(Nack Oriented Reliable Multicast)[5], TRACK(TRee-based ACK)[6, 7]의 세 가지로 분류하여 다양한 응용에 맞는 표준화 프로토콜을 설계하기 위한 노력을 기울이고 있다.

이 중 TRACK은 전송단계에서 계층적인 제어 트리를 구축하여 신뢰성을 보장하는 방법으로 높은 확장성을 보장하면서 효율적으로 신뢰성을 보장하는 것으로 잘 알려져 있다[8]. 그러나 이 방법을 사용하는 프로토콜의 효율성은 전송단계에서 구축되는 논리 제어 트리의 효율성에 크게 의존한다. 그러므로 동적으로 변화하는 그룹 수신자들에게 확장성 및 신뢰성을 보장하는 서비스를 제공하기 위해서는 효율적인 논리 제어 트리 구축 방안을 보장하는 것이 선행되어야 한다.

제어 트리를 구축하는 기존 연구들은 크게 상향식(bottom-up)과 하향식(top-down)으로 분류할 수 있다. [9]는 하향식 방식에 대한 연구를 발표하였으나 이 방식에서는 트리 구조상의 지역 대표 노드들의 공지 메시지(advertisement)가 전체 그룹으로 멀티캐스트 되는데 따른 메시지 오버헤드가 크다는 단점이 지적되었다. 이에 따라 [10]에서는 상향식 기반의 제어 트리 구축 방안들을 정리하여 발표하였다. 발표되어진 방법들은 수신자 기반의 멀티캐스트 그룹 운영 방식을 적용한 것으로 대표노드를 찾아가는 구체적인 메카니즘은 서로 다르지만, 각 수신자들이 동시에 자신과 가까운 서비스 노드를 찾아가는 간단한 트리 구축 방안을 제공하며 그룹에 참여한 수신자들에게 병렬적 트리 구성을 제공한다는 공통점을 가지고 있다. 그러나 상향식 기반 방식 또한 모든 수신자들이 자신의 대표노드를 구하기 위하여 보내는 질의 메시지(query)로 인하여 수신자 수가 커질수록 메시지 부하가 커지는 단점이 있다. 또한 제어 트리에 루프가 발생할 수 있다는 단점과 서비스 노드 수와 네트워크 상태 변화에 민감하다는 단점이 있다[11].

이러한 문제를 해결하기 위한 방안으로 본 논문은 기존의 하향식 방법을 개선하고 이에 기반을 둔 새로운 제어 트리 구축 방안을 제안한다. 본 연구에서 하향식에 관심을 갖게 된 것은 알려진 단점을 해결한다면 이 방법이 송신자와 서비스 노드들로부터 단계적인 트리 구축을 통하여 루프 발생을 방지할 수 있으며 네트워크의 환경에 투명한 서비스를 제공할 수 있다는 장점이 있기 때문이다. 또한 상향식 방법은 하위 네트워크 계층의 멀티캐스트 라우팅 트리를 구성할 때 수신자 기반의 상향

식이 확장성을 보장할 수 있다는 알려진 사실을 따라서 수신자 기반 방식을 채택한 것인데, 전송 계층 트리는 네트워크 계층의 멀티캐스트 라우팅 트리와 달리 송신자와 수신자간의 상태 정보 교환이 필수적인 만큼 송신자 초기화 방식의 트리 구성이 송신자에게 추가적인 오버헤드를 주지 않는다.

본 논문은 이러한 고려를 바탕으로 기존의 하향식의 공지 메시지에 제한적 멀티캐스트를 적용하고 공지 메시지의 전송 간격을 효율적으로 조정하여 기존 방식이 갖는 단점을 해소하도록 하였다. 실시한 성능평가 결과는 하향식을 기반으로 설계된 제안 기법이 기존의 상향식 기반 방식과 비교했을 때 메시지 부하를 줄일 수 있다는 것을 보였으며 현저하게 낮은 레벨의 트리를 형성함으로써 다단계의 제어 메시지 처리에 효율성을 높일 수 있음을 보였다. 그러나 단계적 트리 구성의 특성상 병렬처리를 하는 상향식 방식보다는 트리 구축시간이 길어지는 것을 볼 수 있었다. 이에, 본 논문은 응용의 요구사항에 맞추어 상향식 방식과 하향식 방식을 선택적으로 사용할 것을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서 제안된 트리 구축 기법을 소개하며, 4장에서 제안 기법에 관한 실험 결과를 보인다. 마지막으로 5장에서 결론 및 향후 과제를 살펴본다.

2. 관련연구

2장에서는 신뢰적인 멀티캐스트 전송 프로토콜의 효율적인 설계를 위한 연구 방향으로 계층 트리를 이용하는 트리 기반 ACK 프로토콜(Tree-based ACK Protocol)의 개념 및 서비스 수행 방법에 대하여 간략하게 살펴보고 이 방법을 적용하기 위하여 계층 트리를 구축하는 방법에 대한 관련 연구를 살펴보았다.

2.1 트리 기반 ACK 프로토콜

트리 기반 ACK 프로토콜 메카니즘은 전송 계층상에 논리적 제어 트리를 구축하고 하나의 멀티캐스트 그룹을 계층적 관계를 갖는 여러 개의 지역 그룹으로 나누어 지역 그룹 단위로 상태 정보를 관리한다. 그림 1에서 보는 것처럼 송신자(S)가 루트가 되는 제어 트리를 형성하고 중간 노드들 중에 RH(Repair Head)라는 서비스 노드가 있어서 자신의 지역 그룹 수신자들(R)에게 오류 제어 및 ACK 통합 등의 서비스를 제공한다. 이 때 RH 노드는 종단의 호스트 또는 서비스 사업자가 보유한 서버들이 될 수 있는데 제어 트리 형성 전에 그 역할이 명시되어야 한다. 이 과정은 SAP[12], SDR[13] 등 세션에 관한 규약에 따라 웹 등을 통해 멀티캐스트 세션

참여시에 명시되어질 수 있는 것으로 본 논문에서는 다루지 않는다.

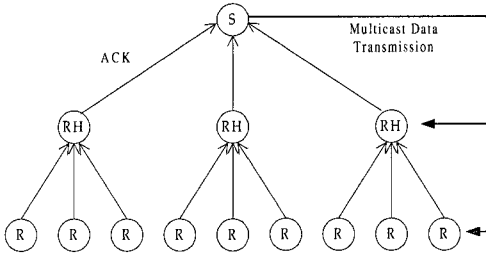


그림 1 Tree-based ACK 프로토콜의 제어 트리

제어 트리 형성이 완료되면 멀티캐스트 송신자는 데이터를 전송한다. 이 때 데이터는 멀티캐스트 라우팅 경로를 따라 전송되고 상태 정보는 제어 트리 경로를 따라 전송된다. 각 수신자는 데이터 수신 상태와 흐름 및 혼잡 제어에 대한 정보를 실은 패킷을 자신의 RH에게 전송한다. 각 RH는 수신한 정보를 바탕으로 손실 패킷의 재전송을 수행하며, 각 수신자의 상태 정보를 통합하여 자신의 상위 노드로 전송한다. 최종적으로 송신자는 자신과 직접 연결을 맺은 수신자들로부터 받은 통합된 정보를 종합하여 흐름 및 혼잡 제어를 수행할 수 있다. 이러한 오류 및 흐름 제어에 관한 기능들 외에 그룹 멤버십 관리 및 유지에 관한 기능들 또한 제어 트리를 바탕으로 수행된다.

RMTP-II[14], TRAM[15] 등이 이 기법을 기반으로 프로토콜을 설계하였다. 이들은 패킷 손실 처리나 늦은 참여 등 네트워크의 트래픽을 유발하는 오퍼레이션이 지역적으로 처리되기 때문에 높은 확장성을 보인다. 각 프로토콜들은 오류 및 혼잡 제어를 수행하는 기능에서는 대체적으로 공통성을 지니고 있으나 제어 트리를 구축하는 방법에 관한 이슈들에서는 차이점을 보이고 있다. 그러므로 다음 절에서 이들이 사용하는 트리 구축 기법을 중심으로 관련 연구를 살펴보겠다.

2.2 제어 트리 구성 방안

제어 트리의 구축은 멀티캐스트 세션 광고, RH(Repair Head) 탐색, 최적의 RH로의 바인딩(binding)의 순서로 진행된다. 세션 광고는 웹 등을 통한 out-of-band 방식으로 수행되며 이 때 그룹 수신자들은 멀티캐스트 주소와 송신자 주소 및 제어 트리를 구축하는데 필요한 정보들을 얻고, 송신자가 제어 트리 구축을 알리면 자신과 가까운 RH 탐색을 통해 제어 트리 구축에 참여하게 된다. 최적의 RH를 선택한 수신자는 해당 RH에 바인딩하고

각 RH 역시 자신의 RH 또는 송신자에게 바인딩함으로써 제어 트리를 구축한다.

이 때 RH 탐색과정을 크게 상향식과 하향식으로 나눌 수 있다. 현재 알려진 트리 구성 옵션들은 주로 상향식 트리 구성 방법으로 모든 종단 수신자들이 병렬적으로 RH가 주변에 있는지 멀티캐스트 질의를 통해 부모 RH를 찾으면서 제어 트리가 종단에서부터 상위로 확장되어 최상위 루트 노드인 송신자에게 연결되는 방법을 사용한다. 반면에 하향식에서는 RH들이 자신의 멤버를 모집하기 위하여 그룹으로 공지 메시지를 보냄으로써 상위 루트 노드부터 종단 노드로 트리가 형성되어진다.

각 방식은 각각의 장단점이 있는데, 상향식은 병렬적 트리 구성을 통하여 신속한 트리 구성을 이룰 수 있다는 장점이 있는 반면 네트워크에서 양방향 멀티캐스트가 지원된다는 가정이 있어야 한다. 또한 트리 구성이 완료되기 전에 도착하는 데이터에 대한 오류 회복은 보장되지 않는다. 반면 하향식은 단계적 트리 구성을 통하여 루프 방지가 보장된다는 장점이 있으며 초기 오류에 대한 재전송을 보장할 수 있고 단방향 멀티캐스트로서 서비스를 제공할 수 있다. 그러나 단계적 트리 구성으로 인하여 여러 단계의 트리 구성이 필요한 환경에서 긴 트리 구성 시간이 필요하다는 단점이 있다. 또한 두 가지 방법 모두 메시지 부하가 크다는 단점이 있다.

알려진 연구 중에서 [9]는 하향식 방식에 대한 연구를 발표하였으나 이 방식에서는 트리 구조상의 지역 대표 노드들의 공지 메시지(advertisement)가 전체 그룹으로 멀티캐스트 되는데 따른 메시지 오버헤드가 크다는 단점이 지적되었다. 이에 따라 [10]에서는 상향식 기반의 제어 트리 구축 방안들을 정리하여 발표하였다. 이 방법들은 RH 탐색 메커니즘에 따라 POC(Point Of Contact), GRA(Generic Router Assist), 그리고 ERS(Expanding Ring Search) 방식으로 나눌 수 있다.

POC는 제어 트리 구축을 위해 네트워크에 분포된 지정 서버들을 말한다. 각 RH 노드들은 자신의 도메인 내의 POC에 자신을 등록한다. 각 수신자들은 POC에게 현재 도메인 내의 RH 정보를 문의하고 POC는 RH 리스트를 전송해준다. 그러므로 이 방식은 POC 서버의 전개 상황에 따라 수신자들이 적절한 RH를 탐색할 수 있는 여부가 결정된다.

GRA는 라우터의 기능을 확장하여 이것의 지원을 받아 제어 트리를 구축한다. 각 라우터가 라우팅 트리 정보를 모으고 이들을 하위 스트림(downstream)의 수신자들에게 전송해주며, 상태 정보에 대한 간단한 처리를 통해 제어 트리 구축에 참여한다. 이 방법은 하위 계층

의 멀티캐스트 라우팅 트리와 비교적 일치하는 제어 트리를 구축할 수 있다는 장점이 있지만 확장된 기능을 갖춘 라우터의 전개가 우선되어야 한다는 문제가 있다. ERS는 각 수신자들이 멀티캐스트 질의 메시지를 주고받으면서 적절한 RH를 탐색한다. 각 수신자들은 'TTL-scoping'을 사용하여 자신의 이웃으로 멀티캐스트 메시지를 전송해서 해당 범위 내에 RH가 있는지 탐색한다. 이 때 TTL(Time To Live)값을 1부터 차례로 증가하면서 멀티캐스트 메시지를 전송함으로써 자신과 가까운 RH를 찾게 된다. 만일 이 메시지 범위 내에 RH가 있다면 이 메시지를 받고 이에 응답한다. 이 과정은 RH를 포함한 모든 수신자들이 동시에 수행하며 질의에 대한 응답 메시지를 받거나 TTL값이 255에 도달할 때까지 반복된다.

POC나 GRA가 특정 서버 또는 라우터의 전개가 선행되어야 하고 그 전개 상황에 성능이 크게 좌우된다는 특성이 있기 때문에 본 논문에서는 상향식 트리 구축 방안 중 ERS 방식을 본 논문에서 제안하는 방식의 성능 평가를 위한 모델로 선정하였다.

3. 하향식 기반의 제어 트리 구축 방안

앞에서 살펴본 ERS[16] 방식은 IP 헤더의 TTL정보를 이용하여 간단한 제어 트리 구축 메카니즘을 제공한다. 그러나 이 방식은 트리 구축 과정에서 모든 수신자가 멀티캐스트 질의를 통하여 트리 구축에 참여함으로써 메시지 오버헤드가 크다는 단점이 있다. 또한 일반 수신자와 RH가 동시에 트리 구축에 참여함으로써 루프가 발생할 가능성을 배제하지 못한다는 단점이 있다[11].

그러므로 본 논문은 단계적 트리 구축을 통해 루프 발생을 방지할 수 있는 하향식 트리 구축 방안을 기반으로 한 새로운 제어 트리 구축 방안을 제안한다. 1) 제어 트리 구축을 위해 송신자는 제어 트리 구축을 알리는 메시지를 데이터 채널로 멀티캐스트 한다. 이 메시지를 수신한 RH 노드들이 송신자에게 먼저 바인딩(binding)한 뒤 자신의 지역 그룹으로 제어 트리 초대 메시지를 전송하면 이 초대 메시지를 받은 각 일반 수신자들이 RH에 바인딩하여 제어 트리를 형성해 나간다.

ERS와 제안 방식의 제어 트리 형성과정에서 사용하는 메시지는 표 1과 같다. 상향식 방식에서는 표에 나열된 모든 메시지가 사용되며 본 논문에서 제한한 방식에서는 QUERY 메시지를 제외한 BEACON, QUERY,

표 1 상향식과 하향식에서 사용하는 제어 트리 형성을 위한 메시지

메시지 타입	From	To	전송 타입	설명	상향식	하향식
BEACON	S	R	M	트리 형성 알림	사용	사용
QUERY	R	RH	M	수신자들이 RH를 찾기 위해 그룹으로 보내는 질의	사용	사용 안함
ADVERTISE	RH	R	M	RH의 수신자의 질의에 대한 응답	사용	사용
BIND	R	RH	U	바인드 요청	사용	사용
ACCEPT or REJECT	RH	R	U	바인드 요청에 대한 응답	사용	사용

S: 송신자, R: 수신자, M: Multicast, U: Unicast

ADVERTISE, BIND, ACCEPT(또는 REJECT)메시지가 사용된다. 각 메시지 이름은 사용 성격에 따라 붙여진 이름이며 구체적인 프로토콜 또는 메카니즘별로 서로 다른 이름으로 불려질 수 있다.

BEACON 메시지는 송신자가 모든 수신자에게 트리 형성을 알리고 트리 형성에 대한 기본적인 정보를 전달하기 위하여 사용된다. 이 메시지는 트리 형성 과정동안 주기적으로 전송된다. QUERY 메시지는 일반 수신자들이 자신의 RH를 찾기 위하여 그룹 내로 멀티캐스트하는 메시지인데, 하향식의 단계적 제어 트리 과정에서는 이미 제어 트리의 멤버가 된 RH들이 수신자들에게 ADVERTISE 메시지를 전송하기 때문에 상향식에서 사용되는 QUERY 메시지가 불필요하다. ADVERTISE 메시지는 각 RH가 자신의 제한된 범위의 그룹으로 전송하는 메시지이다. 이것은 상향식에서는 수신자의 QUERY에 대한 응답으로 사용되며 하향식에서는 RH가 지역 그룹 수신자들에게 자신의 정보를 전달하기 위하여 사용된다. BIND 메시지는 RH와 일반 수신자들이 자신의 제어 트리에 바인드하기 위하여 송신자 또는 On-Tree 노드에게 참여요청을 하는 메시지이다. 마지막으로 ACCEPT 또는 REJECT 메시지는 RH 노드들이 자신에게 참여요청을 한 수신자들에게 응답하기 위한 메시지이다.

그림 2는 본 논문에서 제안하는 방식의 트리 형성을 위하여 송신자와 RH 노드간, RH 노드와 일반 수신자간의 메시지 교환 과정을 나타낸 것이다. 이 과정에 따라 제어 트리가 형성되는 과정을 자세히 살펴보면 다음과 같다.

(1) 송신자가 멀티캐스트 세션으로 트리 생성을 알리는 BEACON 메시지를 전송한다.

(2) 이 메시지를 수신한 RH가 송신자에게 BIND 메

1) 편의상 제안 방식을 '하향식'이라고 칭하고, 비교 대상인 기존 연구를 '상향식'이라고 통칭하여 서술하겠다.

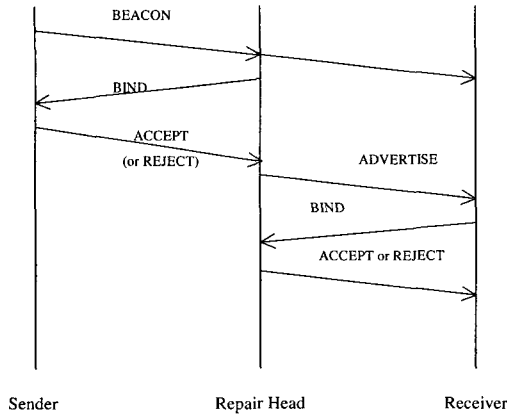


그림 2 제어 트리 형성을 위한 메시지 교환 과정

시지를 전송하고 송신자는 ACCEPT 또는 REJECT 여부에 대해 응답한다. 이 때 BIND 허락 여부는 'MaxNumChildren' 및 'MinNumRH'의 두 변수에 의해서 결정된다. 이 두 변수는 세션 정보로 기록되어 송신자의 BEACON 메시지에 실려 모든 수신자에게 전달된다. MaxNumChildren은 지역 그룹의 최대 수신자 수를 나타내며, MinNumRH는 지역 그룹 수신자들에 포함된 최소 RH 수를 나타낸다.

(3) On-tree 노드가 된 RH는 자신의 지역 그룹으로 ADVERTISE 메시지를 전송하여 자신의 이웃에 위치한 Non-Tree 노드들을 초대한다. 만일 송신자의 트리 형성 규칙(MaxNumChildren과 MinNumRH)에 위배되어 REJECT 메시지를 받은 RH 노드는 이미 On-Tree 노드가 된 다른 RH들의 ADVERTISE 메시지를 기다린다.

(4) On-Tree 노드가 된 RH들의 ADVERTISE 메시지를 받은 이웃의 Non-tree 노드들은 자신에게 전송되는 ADVERTISE 메시지에 실린 정보를 모아 자신에게 가장 가까운 RH를 선정한다. ADVERTISE 메시지를 통하여 자신의 부모 RH 정보를 얻은 수신자들은 해당 RH로 BIND 메시지를 전송한다. 이 때 부모 노드 선정은 구현시 정해질 수 있으며, ERS와 같이 Hop Count 또는 IP 주소를 기준으로 삼을 수 있다.

(5) 각 RH는 'MaxNumChildren' 과 'MinNumRH'를 기준으로 ACCEPT 또는 REJECT를 결정하여 응답한다.

이 과정은 현재 그룹에 참여해 있는 모든 노드가 On-Tree 노드가 되거나 정해진 타이머가 만료될 때까지 계속된다. 이런 방법으로 제어 트리는 그림 3과 같은 모양으로 형성되어 진다.

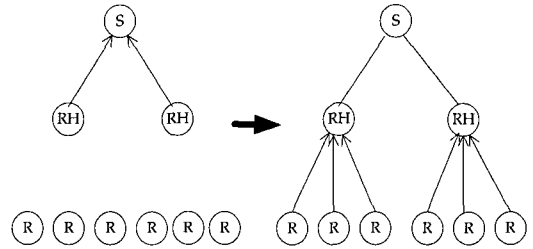


그림 3 제안 방식의 트리 형성 과정

이러한 방법에서 ADVERTISE 메시지의 전송 간격이 메시지 부하에 크게 영향을 주게 된다. 제안한 기법에서는 $T(AGT: Ack Generation Time)$ 을 정의하여 이 메시지로부터 과도한 트래픽이 발생하지 않도록 한다. $T(AGT)$ 는 다음과 같이 정의하였으며, ADVERTISE 뿐 아니라 멀티캐스트 데이터의 ACK 메시지 전송을 위하여 동시에 쓰일 수 있는 값으로 정의하였다.

$$T(AGT) = \max(1 \text{ second}, AdvCnt \times AdvSize / MaxAdvBW, AdvCnt / MaxPacketRate)$$

여기서 $AdvCnt$ 는 ADVERTISE 메시지의 수를 나타내고, $AdvSize$ 는 비트단위로 계산된 ADVERTISE 메시지의 크기를 나타낸다. 이것은 UDP와 IP 헤더를 포함한 크기이다. $MaxAdvBW$ 는 최대 가능한 ADVERTISE 대역폭을 나타낸다. 이것은 응용에서 데이터가 전송되기 전과 후에 서로 다르게 값을 제어할 수 있다. 이런 방법으로 데이터 전송이 시작된 이후에는 ADVERTISE 메시지의 전송 대역폭을 적게 변경시킬 수 있다. $MaxPacketRate$ 는 데이터 패킷의 최대 전송률을 나타낸다. ADVERTISE 메시지를 수신한 수신자들은 $T(TI: Tree Invite)$ 시간동안 이 메시지를 모아서 최적의 RH를 계산하여 바인딩을 하게 된다. 이 때 $T(TI)$ 는 $T(AGT)$ 를 이용하여 다음과 같이 정의한다.

$$T(TI) = \min(3 T(AGT), C \text{ seconds})$$

이 때 C 는 상수로서 최적의 RH를 평가하기 위한 최소 시간을 반영할 수 있도록 정하는데, 멀티캐스트 그룹의 규모에 따라 세션 시작시 값을 부여할 수 있다. 기본 값은 60초로 부여하여 최소 60초 내에 RH 평가가 종료되는 것을 보장한다.

설명한 과정에서 보이는 바와 같이 제안 기법은 단계적 트리 구성을 통하여 송신자와 RH 노드들 간의 바인딩이 먼저 이루어지고, On-Tree 노드가 된 RH들과 일

반 수신자들간의 바인딩이 차례로 이루어지기 때문에 손쉽게 루프를 방지할 수 있음을 직관적으로 보여 준다. 또한, 모든 수신자들이 제어 트리를 참여하기 위해서 그룹으로 멀티캐스트 되는 QUERY 메시지가 불필요해짐을 알 수 있다.

그러나 QUERY 메시지가 제거되었으나 ADVERTISE 메시지가 상향식과 같이 단발성이 아니고 주기적으로 발생하게 되므로 하나의 제어 메시지가 대역폭 내의 하나를 차지한다고 가정하여 메시지 부하의 감소여부를 살펴보면 다음과 같다. 측정을 위하여 제한된 하나의 지역 그룹 영역 안의 평균 수신자 수를 R 이라 하고, 부모 RH와 하나의 자식 수신자와의 평균 홉 거리를 d 라 놓는다. 또한 $T(RH)$ 를 상향식 방법에서 RH를 탐색하는데 걸리는 시간이라고 정의한다.

이 때 상향식의 경우 TTL값이 1씩 증가할 경우 하나의 수신자가 평균 $O(d)$ QUERY 메시지를 그룹으로 전송한다. 나머지 $R-1$ 개의 수신자도 같은 수의 메시지를 전송하게 되므로 평균 $O(dR)$ 의 QUERY 메시지가 전송된다. 이러한 QUERY 메시지에 대하여 RH가 유니캐스트 응답을 보내야 하므로 $O(R)$ 의 ADVERTISE 메시지가 필요하므로 $O(dR) + O(R) \approx O(dR)$ 만큼의 메시지가 교환된다. 만일 d 가 무시할 만큼 작다면 이것은 $O(R)$ 로, 그렇지 않다면 $O(R^2)$ 으로 표시될 수 있다. 초당 메시지 수를 측정한다면, $O(R) / T(FH)$ 또는 $O(R^2) / T(RH)$ 가 될 것이며 이것은 두 가지 경우 모두 이것은 수신자 수에 비례하여 메시지 부하가 증가한다는 것을 나타낸다.

하향식의 경우 RH 탐색은 RH의 ADVERTISE 메시지로 이루어지며 메시지 부하는 이 메시지의 전송 간격에 비례할 것이다. 만일 주어진 시간 안에 n 번의 메시지가 전송된다면 이 방법의 메시지 부하는 $O(n)$ 에 따른 것이다. 여기서 하향식의 경우 상향식과 같이 양방향의 교환이 필요하지 않으므로 RH탐색은 $1/2 \times T(RH)$ 에 이루어지고 이로써 초당 교환되는 메시지 수는 $1/2 \times O(n) / T(RH)$ 로 나타낼 수 있다. 이것의 의미는 메시지 부하가 수신자 수의 증가에 비례하지 않는다는 것인데, 제안 기법은 정의한 $T(AGT)$ 를 통하여 비교한 상황 방식 모델보다 적은 메시지 부하를 나타낸다는 것을 실험결과에서 보였다. 이 결과는 4장에서 서술된다.

4. 성능 평가

이번 장에서는 앞에서 설명한 상향식 기반의 ERS[16] 방법과 하향식 기반의 본 논문에서 제안한 기법에 대한

비교 평가를 실시한 시뮬레이션 결과를 설명한다. 시뮬레이션은 SMPL 라이브러리[17]를 사용한 C 코드로 실시하였다.

시뮬레이션은 100, 250, 500, 1000개의 노드를 통하여 실시되었다. 각 노드들은 홉 거리 정보를 가지고 있으며 TTL 값을 비교하여 이 값을 기준으로 RH 노드를 선정하게 된다. 그룹의 범위는 TTL 64로 지정한다. ERS 방식은 TTL 값의 증가치에 따라 메시지 오버헤드가 달라질 수 있으므로 QUERY 메시지의 TTL값을 1, 2, 5 단위로 각각 증가시킨다. $MinNumRH$ 는 $MaxNumChildren$ 의 10%로 정한다. 하나의 RH가 서비스하는 지역 노드 수는 최대 50으로 제한한다. 또한 지역 그룹내의 적절한 RH 수를 평가하기 위해서 노드 1000의 환경에서 전체 RH의 수를 50, 100, 150, 200, 250으로 증가시켰다. 즉, 최대 RH 수를 전체 노드 수의 30%까지 할당시켜 실험한 것인데, 만일 RH 수가 50%가 된다면 한 RH가 노드 1개를 서비스하는 모양이 되기 때문에 이것은 트리 기반의 서비스를 사용하는 의의를 상실하게 되므로 실험에서는 이를 30%까지 증가시킨다.

제한한 기법의 성능 평가를 위해서 사용된 매트릭스는 다음과 같다.

- 트리 형성 과정에서 교환되는 메시지 수 : 메시지 오버헤드를 측정하기 위하여 BEACON, QUERY, ADVERTISE, BIND, ACCEPT/REJECT 메시지 수를 더한다.
 - 트리 형성 시간 : 모든 노드들이 제어 트리에 연결되는 데 걸리는 시간을 측정한다
 - 트리 레벨 : 형성된 제어 트리의 효율성을 평가하기 위하여 트리의 레벨 수를 측정한다.
- 첫 번째 실험으로 그림 4는 제어 메시지 수의 측면에

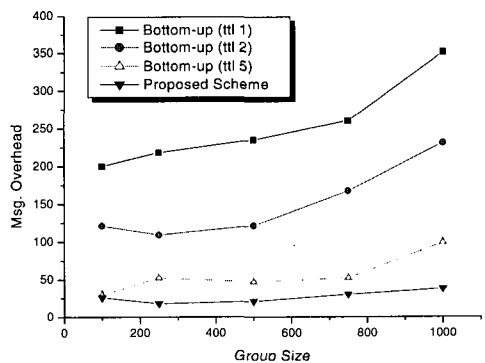


그림 4 그룹 사이즈 변화에 따른 메시지 오버헤드

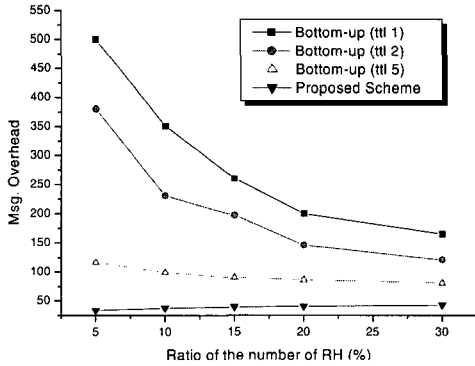


그림 5 RH 비율의 변화에 따른 메시지 오버헤드

서 제안 방법인 하향식과 ERS로 대표되는 상향식의 성능을 평가한 것이다. 서로 다른 그룹 수신자 환경에서 실험한 모든 그룹에서 하향식 방법이 제어 메시지 수를 감소시키는 것을 볼 수 있다. 이것은 앞서 설명한 대로 하향식 방법은 QUERY 메시지의 필요성을 없앴으로써 간단한 메시지 교환으로 제어 트리를 구축하며 하향식 기반의 제안 방법은 ERS와는 달리 TTL값의 증가에 영향을 받지 않는 장점이 있기 때문에 기존 방법의 주요한 단점중의 하나인 메시지 오버헤드를 줄일 수 있는 것으로 나타났다.

그림 5에서는 노드 1000개의 그룹 환경에서 전체 수신자 수 중 RH 노드가 차지하는 비율을 달리하여 메시지 오버헤드를 조사하였다. 상향식 방법은 그룹 내에서 RH 수가 증가함에 따라 성능이 조금씩 좋아지는 것을 볼 수 있다. 이것은 전체 그룹 내에 RH가 증가하면서 그룹 수신자들이 보다 빨리 RH를 찾을 수 있으므로 그만큼 제어 트리에 참여하기 위한 메시지 교환이 적어질 수 있기 때문이다. 그러나 전체 노드의 30%가 RH에 참여한 경우에서도 여전히 Top-down 방식보다는 메시지 오버헤드가 큰 것으로 나타난다. 특히 이 환경에서는 이후에 설명될 트리 레벨에 대한 실험에서 지나치게 깊은 트리를 형성하는 것으로 나타나서(그림 9 참조), 전체 노드에서 RH 수가 많아지는 것이 상향식 방식의 메시지 오버헤드를 줄이는데 도움을 주긴 하지만 전체적인 트리 효율성이 향상된다고 보여지기는 어렵다.

이 결과를 통하여 본 논문의 제안 기법은 메시지 오버헤드 측면에서 그룹 내 RH 비율의 변화에 민감하지 않은 것을 관찰할 수 있다. 그래프의 성장곡선으로 전체 수신자 수에 대한 RH의 비율이 30%를 넘어서도 제안 기법은 일정한 오버헤드를 보일 것이라는 것을 짐작할 수 있다. 그러나 전체 수신자 수에 대한 RH 노드의 비

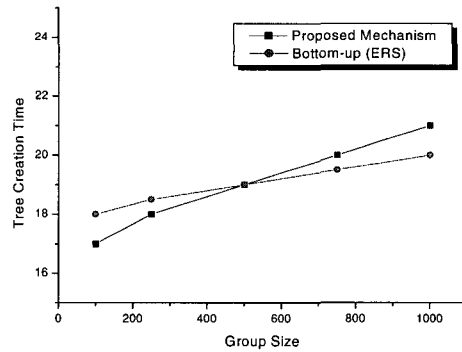


그림 6 그룹 크기 변화에 따른 트리 구성 시간

율이 30%를 넘는 것은 이 장의 앞에서 밝혔듯이 트리 기반의 장점을 저해하기 때문에 실험에서 제외되었다. 이 실험 결과는 RH 비율에 민감하지 않은 제안 기법이 어떤 환경의 네트워크에서도 고른 성능을 낼 수 있다는 장점이 있다는 것을 알 수 있다.

그림 6은 서로 다른 수의 그룹 수신자를 보유한 네트워크 환경에서 제어 트리를 구축 시간 측면에서 성능을 평가한 것을 보이고 있다. 트리 구축 시간 측면에서는 그룹 수신자 수가 증가할수록 단계적 트리를 제공하는 하향식 방법보다 병렬적 트리를 제공하는 상향식 방법에서 우수한 성능을 보이고 있다. 하향식 방법은 그룹 수신자 수 500명 미만에서는 상향식 보다 우수한 성능을 보이지만 그 이상에서는 상향식 방식에 비하여 성능이 저하되는 것을 볼 수 있다. 그러므로 하향식은 트리 구성 시간적인 측면에서만 본다면 비교적 작은 크기의 멀티캐스트 그룹에 보다 적합한 방식이라고 하겠다.

또한 그림 7은 그룹 수신자 수 1000의 환경에서 그룹

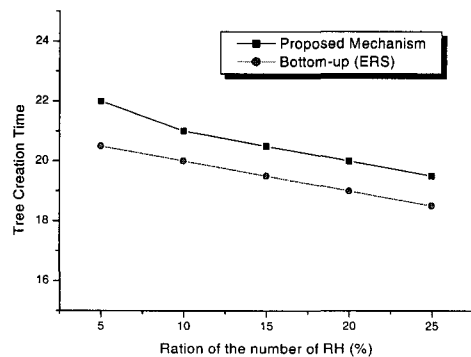


그림 7 RH 비율의 변화에 따른 트리 구성 시간

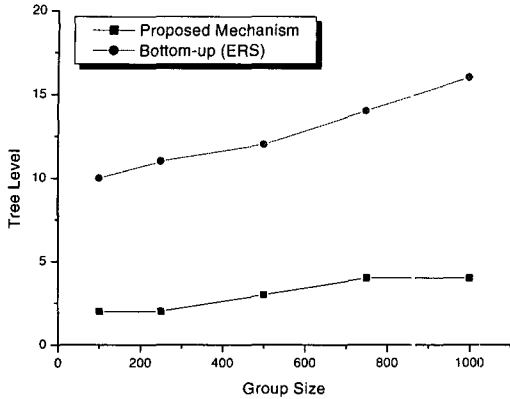


그림 8 그룹 크기 변화에 따른 트리 레벨

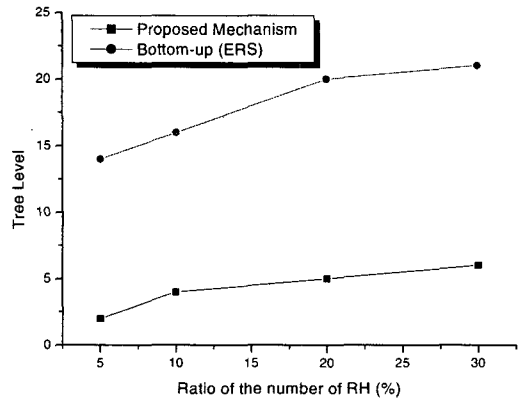


그림 9 RH 변화에 따른 트리 레벨

내 RH 수의 비율이 변화될 경우 제어 트리 구축 시간을 측정하는 것이다. 이 실험에서는 두 가지 방법 모두 RH 수가 증가함에 따라 제어 트리 구축 시간이 감소하는 것을 볼 수 있다. 이것은 일반 수신자들이 가까운 거리 내에서 짧은 시간 안에 자신의 부모 RH를 찾아 제어 트리에 참여할 수 있기 때문이다.

그림 8과 그림 9는 제어 트리가 형성된 후에 트리 레벨을 조사한 것이다. 그림 8은 서로 다른 그룹 수신자 수 아래서 형성된 제어 트리의 레벨을 보이고 있으며, 그림 9는 그룹 수신자 수가 1000일 경우 그룹 내 전체 수신자 대비 RH 수를 달리해서 실험한 결과이다. 두 가지 실험에서 모두 트리 레벨 측면에서는 제안한 하향식 방법이 현저하게 낮은 레벨의 트리를 구축한 것을 볼 수 있다. 트리 기반 방식에서 트리의 깊이가 깊어지면 메시지가 전달되는 길이가 길어지며 통합 메시지 전달에도 비효율적이므로 바람직하지 않다. 실험 결과로부터 상향식 방법은 모든 수신자들이 'greedy manner'를 가지고 전체 트리 구성과 관계없이 자신과 가까운 RH를 만나면 무조건 트리에 바인딩하는 반면, 제안한 하향식 방법은 각 RH들이 ADVERTISE 메시지를 보내고 이것으로 RH의 정보를 모은 뒤 수신자들이 적합한 RH를 찾아가는 방법을 사용하기 때문에 궁극적으로 레벨이 깊지 않은 효율적인 트리를 구축한다는 것을 알 수 있다.

위의 실험을 요약하면 다음과 같은 다음과 같은 결과를 얻어낼 수 있다.

(1) 제어 메시지 수로 측정된 메시지 오버헤드 측면에서는 제안한 하향식 기반 방식이 보다 우수한 성능을 갖는다는 것을 알 수 있다. 이 결과로부터 제안한 방식이 대역폭 자원이 적은 네트워크 환경에 보다 적합하다는 것을 알 수 있다.

(2) 트리 형성 시간 측면에서는 하향식 방법은 그룹 수신자 수 500 미만의 작은 그룹을 가진 네트워크 환경에서 우수하며 이 보다 큰 그룹에서는 상향식 방법이 우수하다는 것을 알 수 있다. 그러므로 제어 트리 형성 시간에 민감한 큰 그룹 크기를 가진 환경에서는 상향식 방법을 사용하는 것이 적합하다는 것을 알 수 있다.

(3) 형성된 제어 트리의 깊이를 평가한 결과에서는 제안한 하향식 기반 방식이 뚜렷하게 우수한 것을 볼 수 있다. 특히 제안한 기법에서는 그룹 수신자 수의 증가나 RH 수의 변화에 민감하지 않게 안정된 레벨의 트리를 형성함으로써 그룹 수신자의 분포나 참여가 일정하게 예측되지 않는 환경에서도 안정된 성능을 나타낼 수 있다는 것을 알 수 있다.

5. 결론 및 향후 연구 과제

신뢰적인 멀티캐스트 전송 프로토콜은 원격 강의, 원격 주식 정보 교환, 공동 문서 작업, 소프트웨어 분배 등 다양한 종류의 응용을 위해 필요하다. 이를 위해 계층적 트리 기반에서 서비스를 구축할 경우 신뢰성뿐만 아니라 확장성을 보장할 수 있다. 이러한 제어 트리 기반의 서비스를 구축하기 위하여 네트워크 계층의 라우팅 트리와는 별도로 전송 계층에서 논리적 제어 트리를 형성하는 것이 필요하다.

현재 제어 트리 형성을 위해서 상향식(bottom-up) 방식이 주로 사용되고 있는데, 이것은 모든 수신자들이 병렬적으로 트리 형성에 참여함으로써 수신자 주도의 간단한 제어 트리 형성 방법을 제공한다. 그러나 이 방법은 루프 방지를 보장할 수 없으며 메시지 오버헤드가 크다는 단점이 있다.

그러므로 본 논문에서는 상향식 방법의 단점을 보완

할 수 있는 하향식(top-down) 기반의 제어 트리 구축 방안을 제시하였다. 하향식 방법은 송신자와 지역 그룹 대표자인 RH들이 주도하여 제어 트리를 단계적으로 형성하여 제어 트리가 송신자로부터 점진적으로 성장해 나가기 때문에 별도의 메커니즘 없이 루프 방지를 보장할 수 있다. 또한 실험 결과를 통하여 제안 방법이 기존의 상향식 기반 방식의 단점인 메시지 오버헤드를 감소시킬 수 있으며 보다 안정된 깊이를 갖는 트리를 형성하는 것을 알았다.

그러나 제어 트리 형성 시간의 관점에서는 하향식은 그룹 크기가 커졌을 때 트리 구축 시간이 길어지는 단점을 보였다. 이것은 단계적인 트리 구축의 특성으로 인하여 병렬적 트리 구성을 하는 상향식에 비하여 긴 시간이 걸리는 것으로 보인다.

이러한 실험 결과는 제한한 하향식 제어 트리 구축 방식이 제어 트리 구축 시간이 민감하지 않고 네트워크 대역폭 자원이 풍족하지 않은 네트워크 환경에서는 상향식의 대안으로 사용될 수 있을 것이라는 것을 나타낸다. 그러므로, 본 논문에서 제한한 방식은 다양한 응용의 요구와 네트워크 환경을 고려하여 상향식 방법과 선택적으로 사용할 수 있을 것으로 보인다.

향후 보다 다양한 그룹 수신자 환경과 다양한 메트릭스를 사용하여 다양한 관점에서 상향 방식과 하향 방식을 비교 평가하며 효율적인 제어 트리 구축을 위한 연구가 지속되어야 한다.

참 고 문 헌

[1] S. Deering, "Host Extensions for IP Multicasting," RFC1112, August 1989.

[2] K. Obraczka, "Multicast Transport Protocols: A Survey and Taxonomy," *IEEE Communication Magazine*, January 1998.

[3] Reliable Multicast Links, <http://research.ivv.nasa.gov/RMP/links.html>, January 2001.

[4] M. Luby, J. Gemmell, L. Vicisano, L. Rizzo, M. Handley, and J. Crowcroft, "Layered Coding Transport: Massively scalable multicast protocol," IETF Internet Draft, draft-ietf-rmt-bb-lct-02.txt, November 2000.

[5] B. Adamson, C. Bormann, S. Floyd, M. Handley, and J. Macker, "NACK-Oriented Reliable Multicast Protocol (NORM), IETF Internet Draft," draft-ietf-rmt-pi-norm-00.txt, November 2000.

[6] B. Whetton, D. Chiu, M. Kadansky, and G. Taskale, "Reliable Multicast Transport Building Block for TRACK," IETF Internet Draft, draft-

ietf-rmt-bb-track-00.txt, November 2000.

- [7] B. Whetton, D. Chiu, S. Paul, M. Kadansky, and G. Taskale, "TRACK Architecture: A Scalable Real-time Reliable Multicast Protocol," IETF Internet Draft, draft-ietf-rmt-track-arch-00.txt, July 2000.
- [8] B. Neil Levin and J.J. Garcia-Luna-Aceves, "A Comparison of Known Classes of Reliable Multicast Protocols," *Proceedings of International Conference on Network Protocol(ICNP-96)*, 1996.
- [9] D. M. Chiu, S. Hurst, M. Kadansky, and J. Wesley, "TRAM: A Tree-based Reliable Multicast Protocol," Technical Report of SUN Microsystems, SML TR-98-66, July 1998.
- [10] M. Kadansky, B. Levine, D. Chiu, B. Whetten, G. Taskale, B. Cain, D. Thaler, and S. Koh, "Reliable Multicast Transport Building Block: Tree Auto-Configuration," IETF Internet Draft, draft-ietf-rmt-bb-tree-config-01.txt, November 2000.
- [11] C. Maihofer and K. Roethermel, "A Robust and Efficient mechanism for Constructing Multicast Acknowledgement Trees," TR-IPVR-99, July 1999.
- [12] M. Handley, "SAP : Session Announcement Protocol," IETF Internet Draft, November 1996.
- [13] M. Handley and V. Jacobson, "SDP : Session Description Protocol," RFC2327, April 1998.
- [14] B. Whetten and G. Taskale, "The Overview of Reliable Multicast Transport Protocol II," *IEEE Network*, January-February 2000.
- [15] M. Kadansky, D. Chiu, J. Wesley, and J. Provino, "Tree-based Reliable Multicast (TRAM)," IEEE Internet Draft, draft-kadansky-tram-01.txt, September 1999.
- [16] R. Yavatkar, J. Griffioen and M. Sudan, A Reliable Dissemination Protocol for Interactive Collaborative Applications, *In Proceedings of ACM Multimedia 96*.
- [17] M. MacDouall, "Simulating Computer Systems, Techniques and Tools," MIT Press, 1987.



김 은 숙

1996년 2월 숙명여자대학교 전산학과 졸업(학사). 1998년 2월 숙명여자대학교 대학원 전산학과 졸업(이학석사). 2001년 8월 숙명여자대학교 대학원 컴퓨터학과 졸업(이학박사). 2001년 3월 한국전자통신연구원 표준연구센터. 관심분야는 RMT, 멀티미디어 그룹통신, 인터넷 QoS등



고 석 주

1992년 2월 한국과학기술원 경영과학과(공학사). 1994년 2월 한국과학기술원 경영과학과(공학석사). 1998년 8월 한국과학기술원 산업공학과(공학박사). 1998년 8월 ~ 현재 한국전자통신연구원 표준연구센터. 2000년 6월 ~ 현재 ITU-T

SG7 & ISO/IEC JTC1/SC6 Project Editor. 관심분야는 인터넷 멀티캐스트, 인터넷 QoS



강 신 각

1984년 충남대학교 전자공학과 학사. 1987년 충남대학교 전자공학과 석사. 1998년 충남대학교 전자공학과 박사. 1995년 정보통신기술사. 1984년 ~ 현재 한국전자통신연구원 통신프로토콜표준연구팀장. 1997년 ~ 현재 ITU-T SG 7

Q.8 Rapporteur. 2000년 ~ 현재 VoIP 포럼 부회장. 관심분야는 멀티미디어 통신, VoIP, 정보보호



최 중 원

1984년 서울대학교 전자계산기공학과 졸업(학사). 1986년 서울대학교 대학원 전자계산기공학과 졸업(공학석사). 1992년 Northwestern University 졸업(공학박사). 1993년 ~ 1997년 숙명여자대학교 전산학과 조교수. 1997년 ~ 현재 숙명여

자대학교 전산학과 부교수. 관심분야는 분산 라우팅 알고리즘, QoS 라우팅, 멀티캐스트 라우팅 알고리즘, RMT 등임.