

논문 2006-43TC-8-2

# 이종 모바일 네트워크에서의 소규모 그룹 통신을 위한 명시적 멀티캐스트

(Explicit Multicast for Small Group Communications in Heterogeneous Mobile Networks)

김 원 태\*

(Won-Tae Kim)

## 요 약

본 논문에서는 다수의 소규모 그룹 멀티캐스트 통신을 위해 제안된 XMIP라는 명시적 모바일 멀티캐스트를 설계하고 구현한다. XMIP는 유니캐스트 라우팅 정보에 기초한 명시적 멀티캐스트의 특징을 상속받아 멀티캐스트 상태유지가 필요없는 간단한 멀티캐스트 메커니즘이다. 본 연구에서는 IETF Mobile IP의 각 이동 에이전트들을 XMIP와의 연동을 고려하여 수정하고 확장한다. 확장 홈에이전트로 전송된 XMIP 패킷들은 확장 홈에이전트의 바인딩 테이블을 참조하여 X-in-X이라는 중첩 터널을 통해 확장 지역에이전트들로 전송된다. X-in-X 터널링 메커니즘은 Mobile IP 멀티캐스트 트래픽 집중 문제를 효과적으로 해결할 수 있다. 마지막으로 CDMA2000 1X EV-DO 및 무선랜을 포함하는 이종 모바일 네트워크 환경을 기반으로 XMIP 실험망을 구축하고, 소규모 그룹 통신을 위한 다중 사용자 인스턴트 메신저를 개발하여 실험함으로써 제안하는 프로토콜들의 실효성을 검증한다.

## Abstract

We design and implement explicit mobile multicast, named XMIP, by enhancing explicit multicast for a great number of small group multicast communications. XMIP is a straightforward multicast mechanism without maintaining multicast states due to the inheritance from the explicit multicast based on a unicast routing. This research modifies and extends the functionality of each mobility agent of IETF Mobile IP for interworking XMIP. XMIP packets captured by an extended home agent are forwarded to each extended foreign agent through nested tunnels, named X-in-X tunnels, made by the binding table of the extended home agent. X-in-X tunneling mechanism can effectively solve the serious traffic concentration problems of Mobile IP multicast specifications. Finally heterogeneous mobile networks as an XMIP testbed including CDMA2000 1X EV-DO and WLAN are actually established, and a multi-user instant messenger system for small group communications is developed for verifying the feasibility of the proposed protocols.

**Keywords :** mobile, explicit multicast, QoS, small group communications

## I. 서 론

1999년 이래로, Xcast (eXplicit multicast)를 대표로 소규모 그룹 멀티캐스트 (small group multicast) 메커니즘들이 활발히 제안되었다<sup>[1-5]</sup>. 이러한 접근법들은

DVMRP와 PIM-SM 같은 기존의 멀티캐스트 라우팅 프로토콜들과는 다른 관점으로 멀티캐스트 서비스를 상품화하기 위해 출현했다. 기존 멀티캐스트 라우팅 프로토콜들이 익명의 많은 멤버를 지원하기 위해 고안되었다면, 소규모 그룹 멀티캐스트 방식들은 네로우 캐스트형의 서비스 혹은 컨퍼런스 콜 (Conference call) 등과 같은 수신자들에 대한 관리가 가능한 동시 대량 세션의 소규모 그룹 통신을 목적으로 설계되었다. Xcast는 기존 멀티캐스트 라우팅 프로토콜의 트리 구조나 멀티캐

\* 정희원, 한국전자통신연구원 임베디드SW연구단  
무선인터넷플랫폼팀  
(Mobile Internet Platform Team, Embedded  
Software Research Division, ETRI)

접수일자: 2006년5월15일, 수정완료일: 2006년8월18일

스트 라우팅 테이블 정보 유지 대신에 유니캐스트 기반의 라우팅 메커니즘을 사용하는 멀티캐스트 개념을 IP 계층에서 채택하여 네트워크 자원의 절약과 멀티캐스트에 따른 트래픽 오버헤드를 제거할 수 있다. Xcast는 기본적으로 IP 헤더 다음에 부가(extra) 헤더로서 모든 수신자들의 IP 주소를 포함한다. 다시 말하면, IPv4 혹은 IPv6 옵션 헤더로써 IP와 UDP 사이에 새로운 통신 계층이 설계되어 있다. 게다가 위에서 언급한 특징들로 인해 Xcast는 강력한 세션 컨트롤 능력을 가진다. 결국, 이로 인해 Xcast 네트워크에서는 인증과 보안에 대한 관리가 쉽기 때문에 멀티캐스트 기술의 상용화를 위해서 매우 중요한 기술적 장점이다. 그러므로, 영리를 목적으로 하는 상용 모바일 네트워크 환경에 언급한 응용 서비스들을 위해 Xcast를 채택하는 것은 적합한 선택이라 할 수 있다.

IETF Mobile IP는 HA 동작에 기반한 멀티캐스트 메커니즘을 지원하지 않지만, 이를 이동통신망에서 적용할 경우 중대한 트래픽 오버헤드 문제들이 발생한다<sup>[6][11][12]</sup>. 만약 한 HA가 멀티캐스트 그룹의 다수의 모바일 노드들을 관리해야 한다면, 터널링 메커니즘에 의해 각각의 모바일 노드로 멀티캐스트 패킷들을 전송해야 한다. 또한, 만약 서로 다른 HA들이 하나의 멀티캐스트 세션에 가입되어 통신 중인 멀티캐스트 그룹 멤버들을 서비스한다고 가정하면, 서로 다른 HA들로부터 다중의 동일 멀티캐스트 트래픽들이 하나의 FA에 집중하므로써 중복 수신 문제가 발생할 수도 있다. 따라서, 이러한 모바일 멀티캐스트 방식은 HA 시스템과 전체 모바일 네트워크의 통신 효율 면에서 있어서 매우 많은 투자 비용을 유발시킨다.

이에 본 논문에서는 XMIP (Xcast over Mobile IP)가 모바일 IP 멀티캐스트의 트래픽 오버헤드 문제를 해결하고 모바일 통신에 긍정적인 효과를 줄 것이라고 예상된다. II장에서는 관련 연구들과 Mobile IP를 동반한 Xcast의 문제점들을 요약하고 있다. 즉, HA에서의 트래픽 폭주이다. III장에서는 XMIP의 설계 원리와 프로토콜에 대해 자세히 설명할 것이다. IV장에서는 XIM (XMIP Instant Messenger)라는 다중 사용자를 위한 인스턴트 메신저 시스템을 구현하여, XMIP 터미널 아키텍처를 제시한다. V장에서는 다중망간 핸드오프 실험에서 XMIP의 유효한 평가를 위해 CDMA2000 1X EV-DO와 WLAN을 통합한 실험망을 구축하여 XIM을 통해 다양한 실험을 수행한다.

## II. 관련 연구

### 1. Xcast 프로토콜 개요

기존 멀티캐스트 라우팅이 IP 계층에서 수행된 반면에, 위에서 언급한 것과 같이 Xcast는 3.5 계층에서 수행하고, 3.5 계층 프로토콜 헤더는 모든 멀티캐스트 멤버의 주소를 포함한다<sup>[4]</sup>. 다시 말하면, Xcast 패킷을 전송하는 Xcast 라우터들은 멀티캐스트 트리 정보가 아니라, 모든 목적지 주소를 가지고 있는 Xcast 헤더 정보에 기초한다. IP 계층 다음에 위치한 Xcast 계층 때문에, 기본적으로 IP 유니캐스트 메커니즘에 의해 다중 수신자들에게 패킷을 발송한다. Xcast 라우팅 도중에는 최소의 패킷 복제가 일어나게 되며, Xcast 패킷들은 일반적인 IP 유니캐스트 패킷으로 전송된다. X2U (Xcast to Unicast)라고 불리는 이 메커니즘은 non-Xcastable 호스트도 Xcast 네트워크상에서 IP 패킷을 받을 수 있다. 그러므로 Xcast 프로토콜은 Xcast를 이해하지 못하는 수신자도 어떠한 시스템 변경없이 Xcast 망을 이용한 통신을 할 수 있도록 설계되어 있다. 하지만 Xcast는 추가적인 수행이 또한 요구된다. 이러한 Xcast는 다수의 소규모 그룹 세션을 형성하는데 효과적이다. 그림 1은 Xcast 전송방식을 이해할 수 있는 간단한 시나리오를 도시하였다.

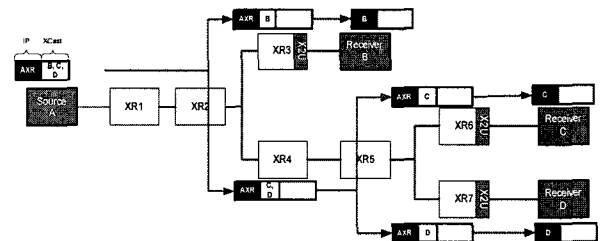


그림 1. 간단한 XCast 동작 시나리오  
Fig. 1. A simple operation scenario of XCast.

### 2. Issues of Xcast with Mobile IP

HA는 터널 인터페이스를 논리적인 네트워크 인터페이스로 생각하기 때문에, 그림 2에서 보는 것과 같이 IP 터널링 기법을 사용하는 X2U 혹은 X-in-U(Xcast in Unicast)를 수행한다. 만약 모바일 네트워크들 상에서 많은 사용자들이 성기게 (Sparsely) 분포되어 있다면, 트래픽과 컴퓨팅 오버헤드는 HA에 집중될 지도 모른다. 어떤 사람은 그것을 반복된 패킷 복제의 트래픽 폭주에 의한 트래픽 사태라고 말한다. 결국, 컴퓨팅 오버헤드는 반복된 수행에 의해 HA에서 발생할 것이다.

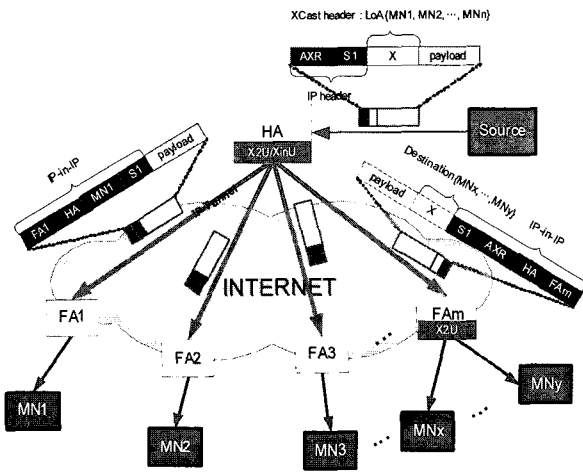


그림 2. Mobile IP와 Xcast 통합 시 문제점  
Fig. 2. The problem of the integration of Mobile IP and Xcast.

HA가 가지는 부담은 Xcast가 적용된 모바일 IP에서 매우 심각한 통신 혼잡을 야기시킬 것이다<sup>[7-9]</sup>.

### III. XMIP 아키텍처

#### 1. XMIP (Xcast over Mobile IP) 프로토콜

XMIP (Xcast over Mobile IP)는 모바일 IP와 Xcast 둘 다 포함하는 새로운 모바일 멀티캐스트 프로토콜로 평가 될 수 있기 때문에, 모바일 그룹 통신에 사용되는 네트워크 자원을 줄일 수 있고, IETF Mobile IP 멀티캐스트에 의해 발생하는 트래픽 집중 문제를 해결할 수 있는 Xcast 프로토콜의 기능을 상속한다<sup>[7]</sup>. 그리고 모바일 IP의 본질적인 요소, 즉 HA, FA 그리고 MN은 Xcast 메커니즘이 효율적으로 모바일 IP로 적용될 수 있게 수정 되어야 한다. 확장된 요소는 접두사 'x'를 가진다. 예를 들면, xHA는 확장된 HA를 의미한다. Xcast 프로토콜을 포함한 HA (xHA)는 등록된 MN으로부터 패킷들을 가로채고, 터널 인터페이스를 이용하여 CoAs로 패킷을 전송한다.

그림 3에서 xHA는 X-in-X 터널링 동작을 수행한다. xHA는 테이블에 mobility 바인딩 정보에 근거하여 X-in-X 인캡슐레이션을 한다. 그림 4는 이러한 동작의 대표적인 예이다. 그림 4에서, xHA에 등록된 MN들은 그들의 CoA로 FA-CoA를 사용하고, MN들은 xFA에 분포되어 있기 때문에, CoA는 xFA와 같은 개수다. xHA와 xFA 사이에 네트워크들은 Xcast 라우터로 구성되어있다고 가정한다. xHA가 MN으로 보낼 패킷을 붙잡을 때, xHA는 바인딩 테이블을 분석하고, CoA에

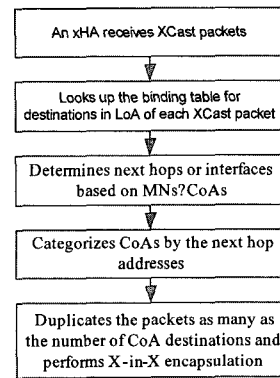


그림 3. xHA에서의 X-in-X 동작 흐름도  
Fig. 3. X-in-X flow in xHA.

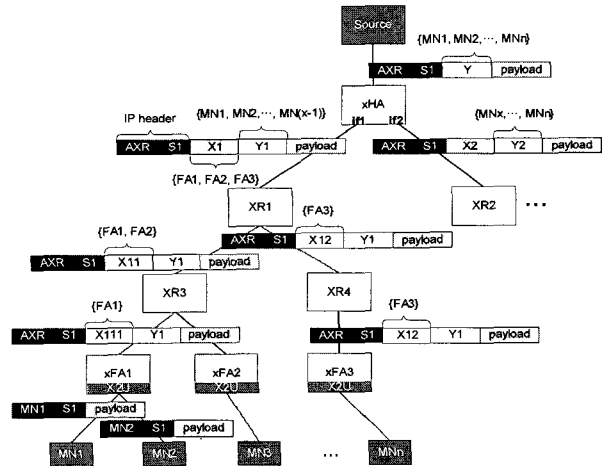


그림 4. XMIP에서의 X-in-X 터널링 메커니즘  
Fig. 4. X-in-X tunneling mechanism in XMIP.

표 1. 패킷 1의 XCast 헤더의 필드 값  
Table 1. The values of XCast header in packet 1.

항목	값			
Destinations in the inner header	MN1	MN2	MN3	MNn
Destinations in the outer header	FA1,	FA1,	FA2,	FA3

해당하는 next hop 별로 분류한다.

다음 홉(Next hop)에 의해서 CoA를 분류할 때, xFA1, xFA2, 그리고 xFA3는 xHA의 if1의 다음 홉에 속하고, 다른 것들은 if2에 의해 사용되어 질 수 있다. 두개의 독립적인 next hop 이나 인터페이스가 있기 때문에, xHA는 두 번 패킷을 복제한다. 패킷 복제 후, 위에 언급했던 것처럼 패킷의 각 비트맵(bitmap)을 바꾸고, 원래의 Xcast 패킷들 위에 CoA의 각 LoA를 포함하면서 터널 Xcast 패킷을 추가한다. 그러면 두개의 X-in-X 패킷은 각 next hop을 위해 생성된다. 각 xFA가 xHA로부터 패킷을 받으면 터널 구조를 해체하고,

이번에는 X2U 동작을 수행한다. 마지막으로 xFA는 IP 패킷을 자신의 서브넷에 존재하는 MN으로 전송한다.

## 2. xHA 구현

만약 적어도 한 개의 entry가 이동성 바인딩 테이블 (mobility binding table)에 있다면, 서브넷으로 들어온 Xcast 패킷을 포함한 모든 패킷들을 감시한다. Xcast 패킷의 경우, 그림 4에서 보이는 CoA와 next hop identification에 기초한 X-in-X 동작을 수행한다. 처음에 xHA는 각 CoA에 적당한 next hop들을 결정한다. 그 다음에는 next hop들에 기초하여 MN들의 홈어드레스를 분류한다. 기본적으로 xHA에 의해 붙잡힌 Xcast 패킷들은 인캡슐레이션 되어 X-in-X 로 전송되어야 한다. 이 과정에서, 각 MN과 일치하는 CoA들은 X-터널 헤더의 LoA로 삽입 된다. FA-CoA 나 Co-located

표 2. xHA에서의 X-in-X 처리 알고리즘  
Table 2. X-in-X processing algorithm in xHA.

```
// Performs Initialization process
Initialize XCast configurations;
Initialize Interfaces;
Import routing table;
Initialize link layer access socket resource;
Initialize XCast socket;
Subscribe AXR address;

// On receiving Xcast packets from senders, xHA performs
// X-in-X operation
while(TRUE) {
    if(receive_Xpacket == TRUE) {
        if (xchecksum(Xpacket_buffer)) DoXinX(Xpacket_buffer);
        else drop_packet(Xpacket_buffer);
    }
}

DoXinX(Xpacket_buffer) {
    allocates memory for tblDevice;
    Lookup_BindingList_xHA(LoA[.Xpacket_header]);
    Create_Destination_xFA_List(List_xFA[.Xpacket_header]);
    num = getNumber_Nexthop(LoA[.Xpacket_header]);
    for(i=num; num>0; num--){
        duplicates Xpackets_received;
        performs inner_header processing for each duplicated Xpacket;
        performs outer_header processing for each duplicated Xpacket;
        forward_XinX_packets(if_nexthop[j]);
        //j is the interface number
    }
    releases memory for tblDevice;
}
```

(CL) CoA는 터널 헤더의 LoA에 위치할 수 있다. 여러 개의 MN들이 동일한 FA CoA를 공유하는 경우도 있을 수 있기 때문에, 내부 헤더에는 여러 개의 주소가 담겨있지만, 외부 헤더는 패킷의 최종 목적지 보다 적은 주소를 가지게 될 수도 있다.

## 3. xFA 구현

X-in-X 나 X-in-U 패킷에서, 처음에 xFA는 패킷의 목적지 필드를 체크하고, 외부 헤더를 디캡슐레이션한다. 다음에 MN들이 있는지 내부 헤더의 visitor list에 등록된 주소의 유무를 확인해야 한다. 만약 패킷들이 xFA의 visitor list에 해당되지 않는 주소가 있다면, 해당 주소들은 0으로 셋팅되고 이후의 내부 Xcast 패킷 처리에 있어서는 무시된다. LoA에 등록된 주소들 때문에, 패킷은 Mobile IP 표준방식으로 전송된다.

표 3. xFA에서의 Xcast 패킷 처리 알고리즘  
Table 3. Xcast packet processing algorithm in xFA.

```
//On receiving Xcast packet at xFA
while(TRUE) {
    if(xtunFlag==FALSE) { //in case of non-tunneled Xpacket
        Lookup_VisitingList(LoA[.Xpacket_header]);
        if(there are entries for the addresses in VL)
            DoXroute(Xpacket_buffer);
    }
    else { //in case of X-in-X tunneled Xpacket
        Lookup_LoA(LoA[.Xpacket_header], xFA_address);
        if(the destination of Xpacket includes current xFA){
            performs decapsulation of the tunneled packet;
            DoXroute(Xpacket_buffer); //send to subnet or X2U
        }
        else drop the Xpacket;
    }
}

DoXroute(Xpacket_buffer){
    allocates memory for tblDevice;
    num = getNumber_Nexthop(LoA[.Xpacket_header]);
    for(i=num; i>0; i--){
        duplicates Xpackets;
        performs header processing for each duplicated Xpacket;
        if(X2U_Flag == TRUE) forward_X2U_packets(if_nexthop[j]);
        else send_Xpacket_nexthops();
    }
    releases memory for tblDevice;
}
```

4. MN에서의 XMIP 프로토콜 구현

가. 패킷의 수신

MN가 home 네트워크로부터 떠나있는 경우에도 MN은 Xcast 패킷을 받는다. 만일 MN이 FA-CoA를 사용하는 경우라면, 패킷은 FA를 통해서 전송될 것이다. CL-CoA를 사용하는 경우라면, 직접 X-in-X 패킷을 받고 터널 해제를 수행해야 한다. 이러한 경우를 제외하고, LoA안에 다른 주소들을 가지고 있는 패킷들은 0으로 채워지게(Overwrite) 되고 이후 처리는 무시되어야 한다.

나. 패킷의 송신

외부 네트워크에서 MN들은 Mobile IP 모듈에서 결정된 디폴트 라우터를 사용한다. 그러나 만일 Xcast 모듈이 다른 라우터를 지정한 경우라면, 그것이 더 높은 우선순위를 가지고 디폴트 라우터로 사용된다. 만일 Mobile IP 등록 시에 역방향 터널링(Reverse tunneling)도 등록되었다면, MN은 Xcast 패킷을 X-in-U형태로 HA에게 인캡슐레이션하여 전송할 수 있다.

IV. XMIP 터미널 구조

1. XIM Terminal

그림 5는 XIM 클라이언트를 포함한 XMIP 터미널 구조를 보여준다. 이 시스템은 4개의 모듈로 구성되어 있다. 즉, 제어와 데이터 전송부분을 담당하는 XIM 응용프로그램, Xcast 라이브러리, Mobile IP 모듈과 CDMA와 WLAN 사이에 무선 신호 강도로 자동 네트워크 스위칭을 수행하는 다중망간 네트워크 선택 관리자(Multi-homed network selection manager) 들이다.

그리고 이 모듈로부터 핸드오프 트리거링(Triggering) 이벤트는 Mobile IP 로 넘겨지고 HA로 즉시 등록

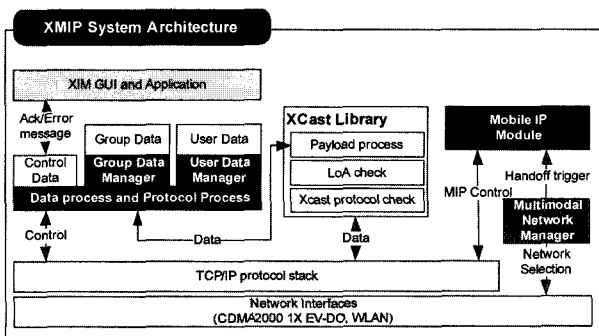


그림 5. XMIP 터미널 구조  
Fig. 5. XMIP terminal architecture.

요청을 보낸다. 이러한 동작은 L2 트리거가 아닌 응용계층 트리거 이벤트로 간주될 수 있다. 결론적으로, 이러한 이벤트 방식의 핸드오프 과정은 Mobile IP의 표준 이동성 검출 (Movement detection) 방식에 비해 빠른 핸드오버 성능을 보이게 된다.

2. 다중네트워크 관리 매니저

다중 네트워크 관리 매니저 (Multimodal Network Manager)는 망간전환 혹은 수직적 핸드오버는 끊어짐 없는 인터넷 연결성을 유지하기 위해 매우 중요한 기능 요소이다. 기존의 연구들 중에서 지능적인 다중 판단요소 기반의 선택 알고리즘인 AHP/GRA가 있다<sup>[13]</sup>.

본 논문에서는 기본적인 망간 전환알고리즘으로서 Song이 제안한 AHP/GRA 방식을 채택하였고, 구현을 통해서 그 동작을 확인하였다. AHP/GRA 방식에서는 최적의 네트워크 판단에 영향을 미치는 다중 요소들 간의 관계를 분석하고 GRC (Gray Relational Coefficient)

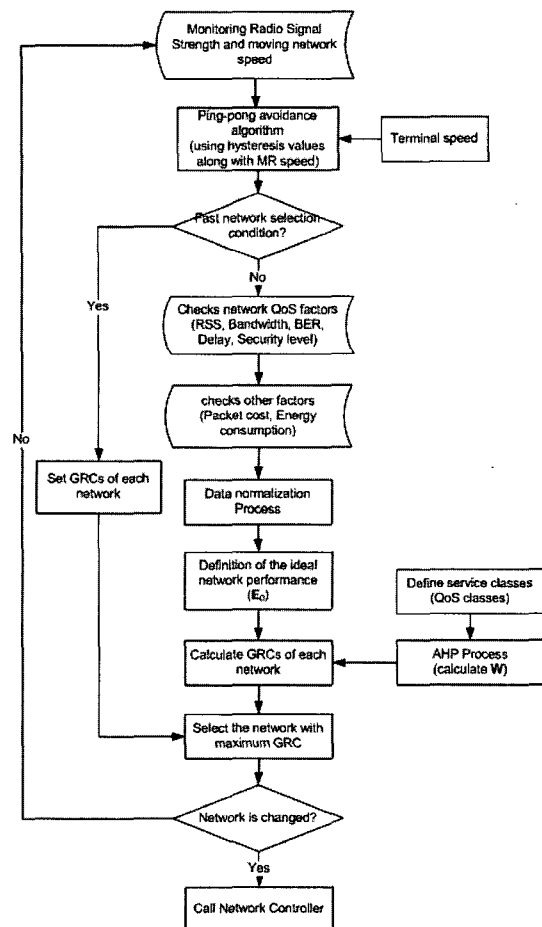


그림 6. 다중 선택요소를 기반한 다중망간 전환 알고리즘  
Fig. 6. Network switchover algorithm with multi-optimal preferences for heterogeneous networks.

를 기준으로 하여 각 옵션들을 평가한다. 옵션이란 UMTS, WiBro 및 WLAN 등 선택 가능한 네트워크를 의미한다. AHP/GRA 방식의 문제점들로는 효과적인 망선택을 위해 모바일 노드의 평풍문제를 고려하고 있지 않으며 더불어 신호세기를 다중 선택요소의 하나로 고려하고 있지 않다는 것이다. 이 알고리즘은 CDMA는 무조건 사용가능하고, 다른 옵션이 WLAN이 사용가능성 여부를 파악하여 가능하면 다른 판단요소, 즉 대역폭, BER, 보안성 및 패킷당 비용 등을 이용하는 방식을 취하고 있다.

Song이 제안한 방식과 본 논문이 제안하는 방식의 차이점은 신호세기 성분을 AHP/GRA 처리 시에 가장 높은 가중치를 갖는 주요 요소로서 평가하였고, 평풍 문제 해결을 위해 단말 속도에 따른 차별적 히스테리시스 값을 적용할 수 있는 *hysteresis\_function (terminal\_speed)* 함수를 알고리즘 내에 도입하였다.

3. XIM 개요

개발된 XIM의 목표는 XMIP가 All-IP 네트워크에서 끊임없는 멀티캐스트 서비스를 효과적으로 지원하는지를 증명하기 위한 것이다. XIM 서버는 마이크로소프트 Visual C++로 개발하여 Windows XP 상에서 동작하고, 기본적으로 XIM 사용자 정보관리 및 Xcast 그룹 세션을 관리한다. 필요시 모든 온라인 상태의 클라이언트들에게 브로드캐스팅 메시지로 긴급공지를 할 수 있다. 서버에서는 클라이언트들간 메시지 송수신에 간섭하지 않으며 오직 온라인 클라이언트들과 그들이 속한 세션 정보들을 관리하여 새로 갱신되는 사용자 정보 (닉네임, IP 주소 및 port 번호 등)를 클라이언트들에게 전송하는 관리시스템 역할만을 수행한다. 사용자 데이터는 실질적으로 Xcast 메커니즘으로 클라이언트들 사이에서 전송된다.

4. XIM 프로토콜

처음에 클라이언트들은 자신들의 등록과 생성 혹은 XIM 세션 신청을 위해서 유니캐스트 메커니즘으로 XIM 서버에 접속한다. 만약 이미 존재한다면, 그들은 세션 정보를 받고, Xcast 헤더의 LoA로 멤버의 주소들을 삽입한다. 위에서 언급했던 것과 같이, 클라이언트들로부터의 Xcast 패킷들은 각 목적지 클라이언트의 xHA로 전송되어진 다음 각 xFA까지 X-in-X 터널링 기법으로 네트워크에 퍼뜨린다. 만약 세션에 새로 온 사람들이 있다면, 서버는 Xcast 패킷 내에 Xcast 헤더

의 LoA에 직접 반영하여 세션에 연결된 모든 클라이언트들에게 그 정보를 알린다.

V. 실험 및 결과

1. 실험 환경

그림 7에서 보는 것과 같이, XMIP 실험망은 한 개의 xHA, 3 xFA, 4 MN, 1 PDSN, 3 Xcast 라우터 그리고 2 WLAN AP들로 구성되어 있다. 각 MN은 모듈모드 네트워크 인터페이스를 가지고 있고, CDMA와 WLAN 사이에서 핸드오프할 수 있다. 표 4는 XMIP 실험망의 세부적인 사항을 보여준다.

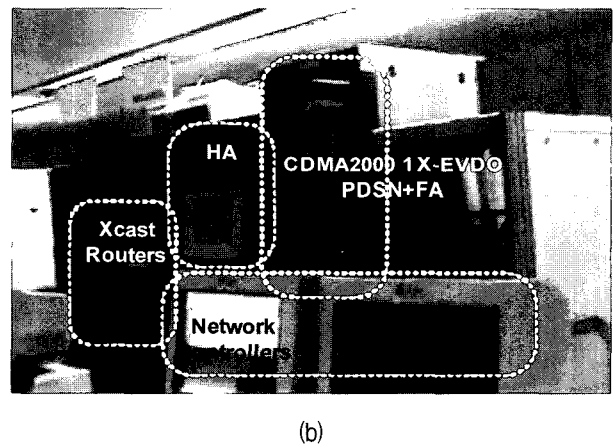
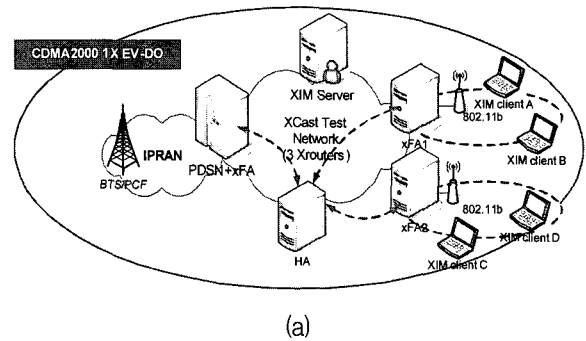


그림 7. XMIP 실험망: (a) 전체 네트워크 구성도, (b) CDMA-2000 1X EV-DO 시스템 파트  
 Fig. 7. XMIP testbed: (a) entire network topology, (b) CDMA-2000 1X EV-DO system part.

표 4. 실험 파라미터 및 적용값  
 Table 4. Experiment parameters and the values.

항목	규격	수량
Mobile IP 모빌리티 에이전트	Pentium 4 1.6GHz	4
라우터	Pentium 3 866MHz	3
XIM	Pentium 3 800MHz	4

클라이언트 시스템		
XIM s서버 시스템	홈에이전트 시스템과 시스템 공유	1
WLAN	11Mbps (IEEE802.11b)	-
WLAN PCMCIA card 및 AP	Orinoco WLAN Card/AP	-
PDSN	CDMA2000 1X EV-DO	1
BTS	CDMA2000 1X EV-DO	1
PCMCIA타입의 CDMA 모델	CDMA2000 1X EV-DO	4
CDMA 대역폭	380Kbps (최소 평균 200Kbps 보장)	-
Mobile IP 프로토콜	IETF RFC2002 호환	-
고정망 네트워크	이더넷 100BaseT	-
핸드오프 기준	WLAN과 CDMA간의 신호세기 기반 핸드오프	-

2. 실험 결과

가. Xcast 라우터 성능 실험

Xcast 라우터의 성능을 실험하기 위해서 일정수의 패킷을 라우터로 삽입하여, 패킷 손실율과 평균적인 패킷 처리 시간을 측정한다. Xcast 라우터를 중간노드로서 지정하고 패킷 발생기로부터 Xcast 패킷을 생성하여 라우터로 전송하면 Xcast 라우팅 및 헤더처리가 된 패킷을 패킷 분석기가 수신한다. 패킷 발생기는 패킷 분석기로 할당된 IP주소와 포트 번호를 삽입한다. 실험 패킷은 IP 헤더 부분의 20 바이트와 Xcast 헤더의 20 바이트 그리고, 페이로드 부분 1400 바이트로 총 1440 바이트의 길이를 가진다. 위에서 언급했던 것처럼, 실험 방법은 다음과 같다. 패킷 발생기는 패킷 분석기로 10000개의 null 패킷을 Xcast 라우터로 전송한다. raw socket을 통해 Xcast 패킷이 전송되므로, 전송을 제어, 흐름제어 및 에러 제어 등은 적용되지 않는다. 그러므로 운영체제를 포함한 시스템과 네트워크 인터페이스를 통해 최대한의 전송 성능을 낼 수 있다. 실험에서 Xcast 라우팅과 무관한 Xcast 옵션 기능은 비활성화하여 실험한다. 프로토콜 분석기에서는 Xcast 라우터에서 처리된 패킷을 수신 후 Ethereal을 통해서 분석한다. 만약 라우터의 처리 능력을 넘어서 패킷을 수신한다면, 오버플로우 현상이 발생하고 이러한 Xcast 패킷 처리용량을 Xcast 라우터의 성능 지표로 삼는다. 이는 본 실험과 동일한 전송율의 일반 IP 패킷 전송 시 전송 실패율은 거의 zero이며, 패킷 분석기의 수신에 따른 패킷 손실은 거의 없다는 사실을 사전 실험을 통해 검증했다. 동일한 전송 실험을 20회 반복한 결과를 그림 8에 도시한다.

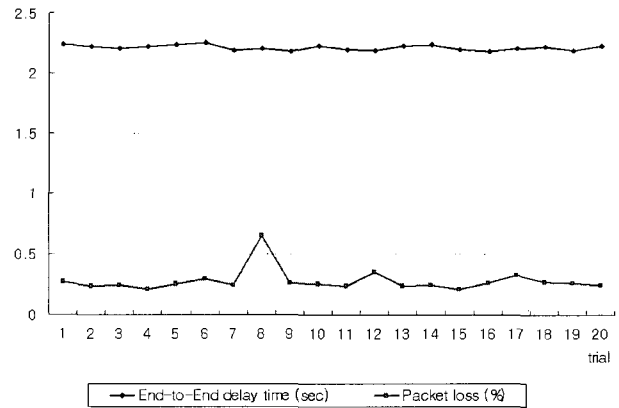


그림 8. XCast 라우터 성능평가 실험 결과  
Fig. 8. The result of XCast router performance test.

10,000개의 입력 패킷에 대해 평균적으로 27.4개의 패킷 손실로 약 0.27% 패킷 손실율이 발생하였다. 또한, 패킷 분석기에서 9972.6개의 패킷을 수신하는데 평균 2.21초의 시간이 소요되었다. 따라서 Intel P3 866MHz CPU기반의 리눅스 커널 2.4.17버전을 사용할 경우, 52.1Mbps 급의 Xcast 스트림을 0.27% 패킷 손실률 내에서 안정적으로 처리할 수 있는 수준이었다. 참고로서 100Mbps급 고속 이더넷의 실측된 성능 (Throughput)은 표준 전송속도의 70~80% 정도인 것을 고려하면, Xcast 라우터 시스템의 최대 대역폭 중 65~70% 정도를 사용하고 있다고 볼 수 있다. 본 라우터의 성능은 인터넷 전화 응용의 요구사항인 5~20%의 패킷 손실 허용 및 최대 20kbps 정도의 미디어 대역폭 요구사항을 고려하면, 2,600 회선정도의 그룹 통신형의 VoIP 응용에도 적용할 수 있는 처리량이다<sup>[10]</sup>. 본 연구는 100Mbps급의 이더넷과 리눅스기반 PC급 라우터를 이용한 소프트웨어적 라우터에 의존하는 형태이므로 실제 망에 있어서 실시간 운영체제 및 하드웨어기반 라우팅을 적용할 경우 성능은 크게 향상될 수 있으리라 사료된다.

그러나 이는 추가적 Xcast 헤더 옵션없이 단일 수신자만을 가정하였기 때문에 절대적인 성능이라고 단언할 수는 없으며 수신자 수나 Xcast 옵션 처리여부에 따라 다소간 처리 능력이 달라질 수 있다. 본 연구에서는 Xcast 라우터 성능 실험 이외에도 헤더 옵션 처리 및 분기 능력 등의 Xcast 기능성 실험 등을 수행하였으며, 이로부터 얻어진 결과들에 의해 Xcast 프로토콜 기능이 설계된 대로 정확히 수행됨을 확인할 수 있었다.

나. XMIP 프로토콜 실험

실험의 주요 목표는 XMIP망을 통한 Xcast 패킷의

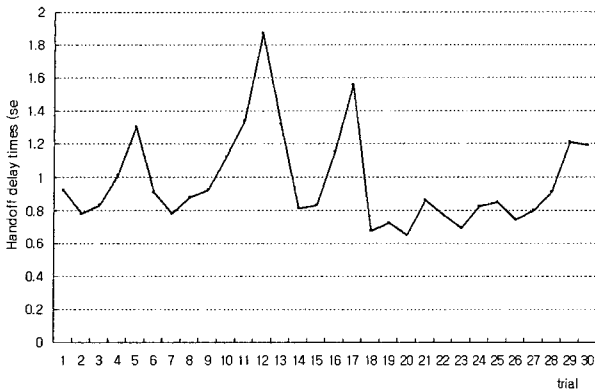


그림 9. XMIP망에서의 이종망간 핸드오버 지연 시간 (sec)

Fig. 9. The handover latency on XMIP network with heterogeneous networks(sec).

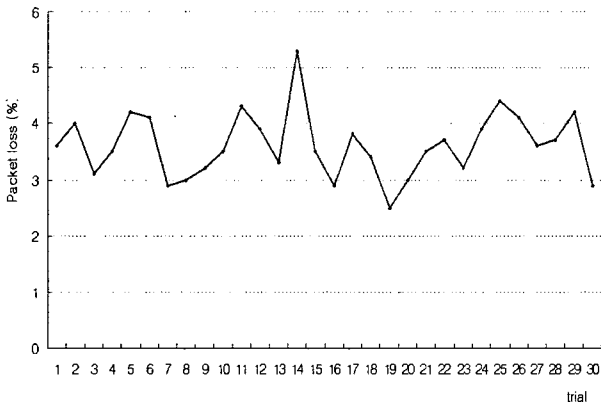


그림 10. 이종망간 핸드오버 이벤트 시 패킷 손실 비율 (%)

Fig. 10. Packet loss ratio on handover events in heterogeneous networks(%).

안정적 모바일 멀티캐스팅이다. 다시 말해서, 송신 노드로부터 xHA로 전송된 Xcast 패킷을 xHA가 각 xFA들로 X-in-X 터널을 통해 원본 Xcast 패킷을 전송하여 중단 이동단말까지 안정적으로 전송되는 것을 확인한다. 그림 9는 실험망에서 XMIP에 의해 핸드오프 시 발생하는 패킷 지연시간을 보인다.

결론적으로 평균 핸드오프 지연 시간은 0.95초로 매우 고른 핸드오프 지연처리가 이루어 졌다. 측정방법은 시스템 타이머를 이용하여 패킷이 단절 후 수신이 재개되는 시간을 측정하는 방법을 사용하였다. 그림 7-a 에서, 송신 노드와 수신 노드의 중단간 전송지연은 최대 10 hop이 가능하다. Xcast 라우터 성능 실험에서 언급한 것처럼, Xcast 라우터상의 패킷 처리 지연시간은 0.2msec/packet, 고속 이더넷에서 전송지연시간 (Transmission delay)는 약 11.5msec/packet 이고, 네트워크에서 전달지연시간 (Propagation delay)는 hop 카

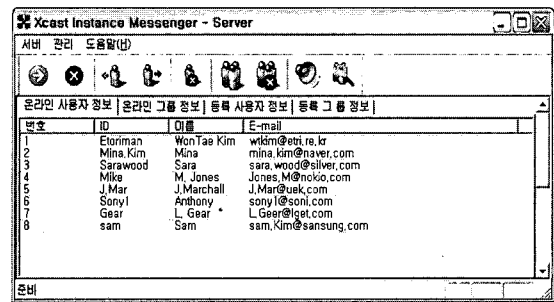
운트가 최대 10 hop이기 때문에 무시해도 좋다. 전체 지연 시간은 12msec/packet 보다 작고, 실험망에서 소스와 목적지간의 지연 시간은 무시될 수 있다.

그러므로 핸드오프 지연 시간은 주로 XMIP 프로토콜의 동작에서 발생한다. 우리의 XMIP는 Layer2 트리거링 메커니즘을 채택하고 있지 않기 때문에, 새로운 무선 지역을 센싱하는 것은 Mobile IP 원래의 이동 검출 알고리즘에 의존해야 한다. 핸드오프 처리에 대한 다른 성능 측정방법은 전송율이 150Kbps급 Xcast 트래픽을 발생시켰을 때, 핸드오프에 따른 XMIP 망에서의 패킷 손실량을 측정하였다. 그림 10은 패킷 손실율을 보여준다.

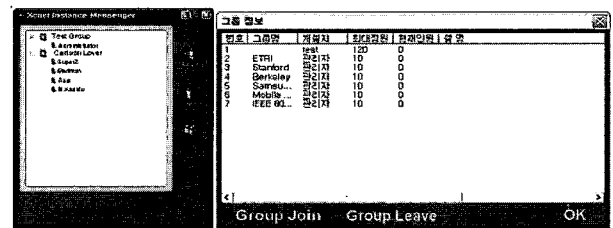
평균 패킷 손실율은 핸드오프 시 약 3.6%로 측정되었다. 멀티미디어관점에서 분석해보면, 150Kbps 대역의 QCIF급 VoD (Video on Demand)가 가능하다. 결론적으로 말하면, 평균 5.6kbps가 핸드오프 시 손실 될 수 있고 그 손실된 대역폭은 비디오 스트림에서 많아야 1 개 정도의 프레임에 영향을 주는 수준이다. 그러므로 충분한 버퍼 크기를 가진 멀티미디어 실행은 거의 끊김 없이 영상을 보여줄 수 있다.

다. XIM 실험

본 연구에서 XMIP를 기반으로한 이동 환경에서의 멀티 유저메신저로 XIM을 개발하였다. XIM 클라이언



(a)



(b)

그림 11. XIM 컴포넌트: (a) XIM 서버, (b) XIM 클라이언트 및 그룹 관리 메뉴저

Fig. 11. XIM components: (a) XIM server, (b) XIM client and group control manager.



트로부터 받은 사용자 데이터는 XMIP 네트워크상의 이동하는 다른 클라이언트로 보내진다. 패킷 손실은 XIM 실험에서 거의 관찰할 수 없었으며, XIM 클라이언트들로부터 생성된 모든 패킷들은 XMIP를 사용하는 종단 시스템까지 안정적으로 전송됨을 확인하였다. 스트리밍 서비스와는 달리 메시징 서비스의 경우 사용자의 행위패턴이 일정하지 않고 idle time이 상대적으로 길어서 핸드오프에 따른 패킷 손실이 거의 일어나지 않았다. 게다가 중단간 지연은 실시간 그룹통신이 가능할 만큼 매우 작았다. 그림 11은 XIM 클라이언트와 서버의 사용자 인터페이스를 보여준다.

## VI. 결 론

기존 멀티캐스트 라우팅 프로토콜과 응용 서비스들은 현재 인터넷상에서 광범위하게 개발되었다. 이들은 네트워크 자원을 잘 활용하기 위한 멀티캐스트 기술들의 효율성을 입증하고 있다. 그러나, 많은 멀티캐스트 라우팅 프로토콜은 확장성, 멀티캐스트 트리 유지를 위한 오버헤드, 멀티캐스트 주소 할당 문제, 그룹 관리 그리고 상용화를 위한 AAA (Account Authentication Authorization) 제공 문제 등에서 심각한 한계를 드러내고 있다. 그리고 모바일 네트워크가 진화하면서, VoIP 기반의 P2P와 푸시투톡 (PTT: Push to Talk)과 같은 그룹 통신 서비스가 엄청난 수의 소규모 그룹 세션이 생성될 수 있는 킬러 어플리케이션들의 출현 가능성이 높은 상황이다. 그러나, 기존 멀티캐스트 기술들은 이러한 종류의 서비스를 다루는데 많은 문제를 가지고 있다. 그러므로 보다 향상된 모바일 네트워크 필요성과 사용자들을 결합시키기에 적합한 멀티캐스트 메커니즘이 요구되고 있다. 비록 Xcast가 어드레싱 기법에서 발생하는 문제로 인해서 규모가 큰 그룹 통신에 있어 약점을 지니지만, 유니캐스트 라우팅에 기반한 간단하고 올바른 전송 메커니즘으로 인하여, 소규모 그룹에서는 혁신적인 멀티캐스트 기술들 중의 하나가 될 수 있다.

본 논문에서는 다중의 이종 모바일 환경에서 XMIP의 유용성과 효율성을 입증하기 위해 XMIP 주요 모듈들과 XMIP 단말 시스템을 실제로 개발했다. 이 단말 시스템은 XIM 어플리케이션, Xcast 모듈, Mobile IP 모듈 그리고 다중망간 네트워크 선택 관리자를 포함한다. 또한, HA와 FA를 각각 xHA와 xFA로 기능을 확장하고 구현하였다. xHA에 의해 캡처된 Xcast 패킷들은 xHA의 바인딩 테이블을 참조하여 각 xFA로

X-in-X 터널 인터페이스를 통해서 전송된다. 이 X-in-X 터널링 메커니즘은 IETF Mobile IP 멀티캐스트 서비스들의 트래픽 집중화 문제를 효과적으로 해결할 수 있었다. 마지막으로, XMIP 프로토콜과 XMIP 시스템의 효과적인 실험을 위해 다중의 이종 모바일 네트워크들을 기반으로 하는 실험망을 구축했고, 이 환경에서 XMIP의 프로토콜적인 특징들과 이동환경에서의 운용성을 증명했다.

## 참 고 문 헌

- [1] A. Boudani, B. Cousin, "SEM: a new small group multicast routing protocol," 10th International Conference on Telecommunications 2003, ICT2003, Vol. 1,23 Feb.-1 March 2003 Page(s):450-455.
- [2] R. Boivie, N. Feldman, C. Metz, "Small group multicast: a new solution for multicasting on the Internet," IEEE Internet Computing, Vol. 4, Issue 3, May-June 2000 Page(s):75 - 79.
- [3] S. Egger and T. Braun, "Multicast for Small Conferences: A Scalable Multicast Mechanism Based on IPv6," IEEE Communications Magazine, Jan. 2004.
- [4] R. Boivie et al., "Explicit Multicast (XCast) Basic Specification," <draft-ooms-xcast-basicpec-00.txt>, 2000.
- [5] Mozafar Bag-Mohammadi, Nasser Yazdani, and Samadian-Barzoki, "On the Efficiency of Explicit Multicast Routing Protocols," Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005), pp. 679-685, 27-30 June 2005.
- [6] Junn-Ru Lai, Wanjiun Liao, "Mobile multicast with routing optimization for recipient mobility," IEEE Transactions on Consumer Electronics, Vol. 47, Issue 1, Feb. 2001 Page(s):199 - 206.
- [7] J. W. Lee and M. K. Shin, "Explicit Multicast over Mobile IP (XMIP)," <draft-lee-xcast-mobileip-00.txt>, Nov. 2001.
- [8] M. K. Shin, K. I. Kim, D. K. Kim and S. H. Kim, "Multicast delivery using explicit multicast over IPv6 networks," IEEE Communications Letters, Feb. 2003 Page(s): 91-93.
- [9] M. K. Shin, Y. J. Kim, K. S. Park and S. H. Kim, "Explicit Multicast Extension (Xcast+) for Efficient Multicast Packet Delivery," ETRI Journal, Dec. 2001 Page(s): 202-204.
- [10] A. Croll and E. Packman, "Managing Bandwidth Deploying QOS in Enterprise Networks," page 95, Prentice Hall PTR, 2000.

- [11] Charles E. Perkins, "Mobile IP," IEEE Communications Magazine, pp.84- 99, May 1997.
- [12] Vineet Chikarmane, Carey L. Williamson, Richard B. Bunt, Wayne L. Mackrell, "Multicast support Using Mobile IP: Design Issues and Proposed Architectures," ACM/Baltzer J. Mobile Net. App., Vol. 3, No. 4, pp. 365-79, 1998.
- [13] Q.Song and A. Jamalipour, "Network selection in an integrated wireless LAN and UMTS environment using mathematical modeling and computing techniques," IEEE Wireless Communications, Volume 12, Issue 3, June 2005 Page(s):42 - 48.

---

 저 자 소 개
 

---



김 원 태(정회원)

1994년 2월 한양대학교 전자공학과 학사 졸업.

1996년 2월 한양대학교 전자공학과 석사 졸업.

2000년 8월 한양대학교 전자공학과 박사 졸업.

<주관심분야 : 유비쿼터스 컴퓨팅, 통신 아키텍처, 임베디드 시스템>