

# IP 이동성관리 프로토콜에 따른 TCP 성능 변화 분석

준회원 박창민\*, 정회원 채영수\*

## TCP Performance Analysis Over IP Mobility Management Protocols

Changmin Park\* *Associate Member*, Youngsu Chae\* *Regular Member*

### 요약

최근 들어 WiBro, mobile WiMAX, WCDMA/HSDPA 등의 다양한 무선 데이터 통신 기술이 소개됨에 따라, 이동무선환경에서 효율적인 IP 서비스를 지원하기위한 IP 이동성 관리 프로토콜에 대한 관심이 증대하고 있다. IETF에서는 표준 IP 이동성 관리 프로토콜인 Mobile IPv6(MIPv6)와 MIPv6의 핸드오버 성능을 개선한 Fast Mobile IPv6(FMIPv6), Hierarchical Mobile IPv6(HMIPv6)를 제안하였고, FHMIPv6, S-MIPv6, HIMIPv6 등 IETF의 기본 프로토콜의 장점을 결합한 새로운 프로토콜들에 대한 연구가 활발히 진행되고 있다. 본 논문에서는 이러한 이동성관리 프로토콜들의 다양한 핸드오버 성능 개선 기법이 TCP 성능에 미치는 영향을 ns-2를 사용하여 분석하였다. 시뮬레이션을 이용한 성능분석결과는 계층적 이동성관리, 선행적 핸드오버 기법 및 Simulcasting 기법을 사용한 HIMIPv6가 다양한 TCP 버전에서 MIPv6, HMIPv6, FMIPv6보다 우수한 성능을 나타냄을 보여준다.

**Key Words** : TCP, handover, Mobility Management, Mobile IP, HIMIPv6

### ABSTRACT

As the set of new mobile data networking service technologies, such as WiBro, mobile WiMAX and WCDMA-HSDPA, have been introduced in recent years, IP mobility management protocols which aim to provide seamless IP service over the mobile networking environment become one of the important building blocks for the successful mobile networks. IETF has introduced a basic mobility management protocol over IPv6 (MIPv6) and a set of enhanced protocols such as FMIPv6 and HMIPv6, followed by FHMIPv6, S-MIPv6 and HIMIPv6 which combine the benefits of the basic protocols. We have studied the impact of the various IP mobility management protocols over TCP sessions through extensive simulations with ns-2. Our simulation results show that TCP performance with the HIMIPv6 outperforms MIPv6, FMIPv6 and HMIPv6 over the various TCP protocols, due to its efficient combination of the hierarchical mobility management scheme of the HMIPv6, the predictive handover scheme of the FMIPv6 and the fast simulcasting scheme of the S-MIP.

### I. 서론

노트북, PDA, PMP와 같은 이동성 기기의 급격한 증가에 따른 이동환경에서의 인터넷 필요성이 증대

함에 따라 다양한 고속 무선 통신시스템(WiBro, Mobile WiMAX, WCDMA/HSDPA 등)이 소개되었다. 또한, 이러한 다양한 고속 무선통신시스템들을 인터넷 환경에 효율적으로 수용하기 위한 IP 이동성 관리

※ 이 논문은 2004학년도 영남대학교 학술연구조성비 지원에 의한 것임

\* 영남대학교 정보통신공학과 모바일 네트워크 연구실(fran007@yumail.ac.kr, yschae@yu.ac.kr)

논문번호 : KICS2008-02-066, 접수일자 : 2008년 02월 04일, 최종논문접수일자 : 2008년 5월 19일

프로토콜에 대한 관심도 증가하고 있다.

인터넷 표준화 기구 IETF는 IPv6망에서의 IP단말 기기 이동성을 지원하기 위해 MIPv6<sup>[2]</sup>, HMIPv6<sup>[3]</sup>, FMIPv6<sup>[4]</sup>를 제안하였다. MIPv6는 매크로 환경에서 IP 단말 이동성을 지원하기 위한 프로토콜이며, 고속 이동환경에서 끊임 없는 이동성을 지원하지 못하는 결점을 가지고 있다. 이를 개선하기 위해 HMIPv6는 MAP(Mobile Anchor point)에 의한 계층적 이동성 관리를 통해 상위 계층과 하위계층의 신호처리를 분리하여 이동성 관리의 효율성을 증대하였다. 또한, FMIPv6는 2계층 프로토콜과의 효율적인 결합을 이용하여 선행적(predictive) 핸드오버 모드를 지원함으로써 핸드오버 지연시간을 개선하였다. 이외에도 두 개 이상의 AR에 동시에 패킷을 전송하는 simulcast 기법을 이용한 S-Binding<sup>[5]</sup>, S-MIP<sup>[6]</sup>과 같은 연구들과 HMIPv6의 계층 구조를 이용한 이동성 관리의 효율성과 FMIPv6의 선행적 핸드오버 지원에 의한 끊임없는 연결 서비스의 장점을 효과적으로 접목한 HIMIPv6<sup>[11]</sup>가 제안되었다.

본 논문에서는 이러한 IP 이동성 관리기법에 사용된 다양한 성능개선 기법들이 TCP 성능에 미치는 영향을 ns-2를 이용하여 분석하였다. 또한, FMIPv6, HMIPv6와 HIMIPv6의 호환성을 개선하기 위해서 FMIPv6의 최근 개선사항을 반영한 수정된 HIMIPv6를 제안한다.

본 논문은 다음과 같이 구성되어있다. II장에서는 이동성관리 프로토콜들의 다양한 성능개선 기법의 장단점을 분석하고 FMIPv6의 최근 개선사항을 반영한 수정된 HIMIPv6를 제안한다. III장에서는 다양한 시뮬레이션 환경에서 ns-2를 이용한 이동성관리 프로토콜들의 TCP 성능분석을 수행하여, 각 프로토콜들의 성능개선기법이 TCP성능에 미치는 영향을 분석하며, IV장에서 결론을 맺는다.

## II. IP 이동성 관리 프로토콜 분석

이 장에서는 HMIPv6, FMIPv6, S-Binding, S-MIP 등 기존 IP 이동성관리 프로토콜의 장단점을 분석한 후, 이들의 장점을 결합한 HIMIPv6에 FMIPv6의 최근 개선사항을 반영하여 수정한 HIMIPv6를 제안한다.

### 2.1 IETF 및 기존 IP 이동성 관리 프로토콜

HMIPv6는 MAP이라는 지역적 이동성관리 에이전트를 모바일 노드와 HA 사이에 도입하여 MAP 도메인 내의 핸드오버(Intra-MAP 핸드오버)를 상위 계층

과 단절시킴으로써, 모바일 노드의 HA 바인딩 갱신 횟수를 줄이고 모바일 노드와 MAP사이의 상대적으로 작은 전송 지연시간에 의한 개선된 지역 바인딩 갱신 지연시간을 보여준다. 하지만, MAP 도메인 간의 핸드오버(Inter-MAP 핸드오버)에서는 바인딩 갱신 시간이 길어짐에 따라 MIPv6에 비해 성능이 떨어지는 단점도 있다.

FMIPv6는 핸드오버가 일어나기 전에 모바일 노드가 이동할 AR을 미리 예측해서 바인딩을 갱신함으로써, 실질적인 핸드오버 지연시간 및 핸드오버과정의 패킷 손실을 줄였다. 또한 핸드오버를 수행 하는 동안에는 pAR이 임시 anchor point로 동작하여 모바일 노드로 가는 패킷을 버퍼링함으로써 2계층 연결이 끊어지는 시간동안 발생하는 패킷 손실을 없앴다. 하지만 핸드오버가 수행되는 동안 패킷이 터널링 됨으로써 패킷 사이즈가 커지게 되며, 핸드오버가 이루어지는 정확한 시간을 예측하기가 어렵고, nAR에 대한 잘못된 예측(false-alarm)이 이루어졌을 경우 손실이 크다.

S-Binding은 FMIPv6에서 하나의 HoA가 하나의 CoA에만 대응되어지는 제약을 완화하여 여러 개의 CoA가 하나의 HoA에 대응할 수 있게 만들어 false-alarm에 대한 취약점을 개선한 방법이다. S-Binding에서는 특정 HoA를 목적으로 하는 패킷을 받았을 경우 해당 엔트리에 있는 모든 CoA로 패킷을 사이멀캐스팅(simulcasting)하여 준다. S-Binding은 동일 패킷을 동시에 여러 AR에 전송함으로써 네트워크 자원낭비가 발생하고 모바일 노드가 같은 패킷을 두 번 이상 받을 수 있는 단점이 있다.

### 2.2 FMIPv6 개선사항을 반영한 수정된 HIMIPv6

HIMIPv6는 현재 표준으로 제안된 HMIPv6, FMIPv6, S-Binding의 장점을 하나로 묶어 핸드오버 성능을 개선한 프로토콜이다. HIMIPv6는 핸드오버 시 발생하는 시그널링의 부하를 줄이기 위해 HMIPv6의 계층구조를 이용하였고, 연결 단락에 의한 패킷 손실을 줄이기 위해 FMIPv6의 사전처리 메커니즘을 이용하였다. 또한 FMIPv6의 문제점으로 부각된 false-alarm에 의한 성능손실을 줄이고 pAR로부터의 패킷 포워딩에 의한 지연시간을 줄이기 위해 S-Binding의 사이멀캐스트를 이용하였다.

개선된 HIMIPv6는 MAP 내부에서 일어나는 핸드오버인 Intra-MAP 핸드오버와 MAP 간에 일어나는 핸드오버인 Inter-MAP 핸드오버로 나누어진다. 먼저 Intra-MAP 핸드오버가 일어나는 과정을 살펴보면 그림 1과 같다.

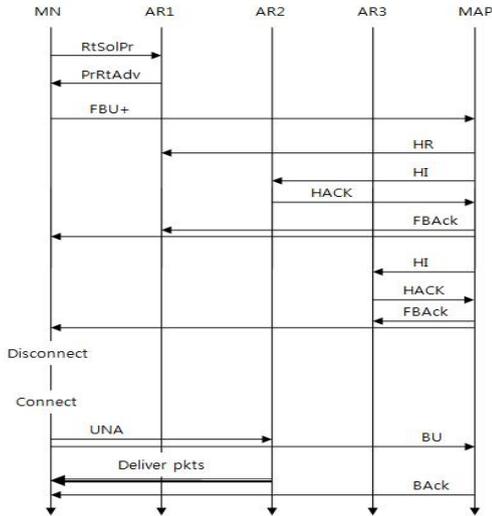


그림 1. 개선된 HIMPv6의 Intra-MAP 핸드오버

모바일 노드는 핸드오버가 필요할 경우 현재 연결되어있는 AR(pAR)과 RtSolPr/PrRtAdv를 주고받음으로써 핸드오버가 가능한 주변 AR에 대한 정보 [(AP-IP, AR-info)]를 수집한다. 이 정보를 바탕으로 모바일 노드는 하나 이상의 candidate LCoA를 생성하고 MAP에게 FBU 메시지를 전송한다. 이때 전송되는 FBU 메시지는 다수의 candidate LCoA를 사용함에 따라 FBU의 확장 개념으로 FBU+라고 명명하였다.

FBU+를 수신한 MAP는 HIMPv6에서 새로 기술된 HR(handover ready)<sup>[1]</sup> 메시지를 B flag를 1로 하여 pAR로 전송을 하고 사이멀캐스트를 시작한다. 이때 패킷이 사이멀캐스트에 의한 것임을 나타내기 위해 IP 헤더에 S flag 필드를 도입하여 값을 1로 세팅한다<sup>[1]</sup>. S flag이 1일 경우 사이멀캐스트된 패킷임을 의미한다. 또한 MAP은 사이멀캐스트 되는 모든 패킷에 시퀀스 번호를 붙이는데, 이를 위해 IP 헤더에 시퀀스 번호에 관한 옵션을 추가하였다. 시퀀스 번호를 사용하는 이유는 pAR과 nAR에서 패킷 중복 수신을 방지하기 위함이다. MAP에서 패킷을 사이멀캐스트 하기 위해서는 candidate LCoA가 유효한지를 먼저 판단해야 하는데 이를 위해서 모든 candidate AR과 HI/HACK을 교환한다. 모든 candidate AR과 HI/HACK 교환을 완료하려면 일정시간이 필요하고 pAR 역시 HR 메시지 수신 후 패킷을 버퍼링하기 때문에 이 시간 동안 모바일 노드에 전송되는 패킷이 없어 서비스가 중단되는 상황이 발생한다. 이를 방지하기 위하여 MAP은 HI/HACK를 교환하는 동안 패

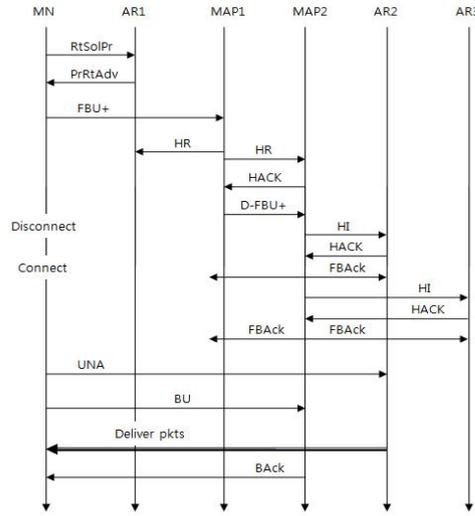


그림 2. 개선된 HIMPv6의 Inter-MAP 핸드오버

킷을 복사하여 하나는 모바일 노드로 전송하고 다른 하나는 버퍼에 저장한다. pAR에서 버퍼링을 수행하는 이유는 flase-alam 이나 ping-pong 상황에 대처하기 위해서이다<sup>[1]</sup>.

HI를 받은 AR은 LCoA의 유효성을 판단하고 HACK 전송 후 버퍼링을 시작한다. MAP은 HACK를 수신하면 버퍼에 저장된 패킷을 포함하여 사이멀캐스트를 시작한다. 개선된 FMIPv6에 의하면 2계층 연결이 회복된 후 기존에 FNA 대신에 UNa를 전송하게 된다. 이 경우를 고려하여 모바일 노드는 UNa 메시지를 먼저 nAR로 전송하고 BU 메시지를 MAP으로 전송한다. MAP은 BU를 수신한 후 사이멀캐스트를 종료하고 모든 프로토콜 과정을 종료한다.

그림 2는 Inter-MAP 핸드오버가 일어나는 과정을 보여주고 있다. Inter-MAP 핸드오버도 Intra-MAP 핸드오버에서와 같이 RtSolPr/PrRtAdv로 주변 AR의 정보를 수집한다. 하지만 Inter-MAP 핸드오버에서는 모바일 노드가 MAP정보를 필요로 하므로 [(AP-IP, AR-info, MAP-info)]를 사용하게 된다. Inter-MAP 핸드오버의 경우 모바일 노드가 이동해야 하는 MAP(nMAP)에게 직접적으로 FBU+ 메시지를 전송할 수 없기 때문에 모바일 노드는 FBU+ 메시지를 현재 연결된 MAP(pMAP)에게 전송한다. pMAP은 모든 candidate nMAP에게 HR을 보내는데 이때 T flag과 A flag을 1로 하여 보낸다. Candidate MAP 중 유효한 nMAP은 HR 메시지를 확인하고 pMAP과의 터널을 형성한 후 HRACK을

전송한다. pMAP은 형성된 터널을 통해 FBU+를 nMAP에게 모바일 노드를 대신하여 전달한다. 이때 전송되는 FBU+를 D-FBU+(delegated FBU+)로 명명하였다. D-FBU+ 메시지를 전송한 pMAP은 임시 HA가 되어 nMAP으로 패킷을 터널링 해주며, 이때 패킷에는 Intra-MAP에서와 같이 시퀀스 번호가 붙는다. 이 시퀀스 번호는 nMAP 도메인 상에서도 그대로 사용된다. D-FBU+를 받은 nMAP은 candidate AR과 HI/HACK를 교환하고 모든 candidate AR로부터 HACK를 받기 전까지 pMAP으로부터 오는 패킷을 p-buffer에 저장한다. 이후 절차는 Intra-MAP 핸드오버와 동일하다.

핸드오버를 마친 모바일 노드는 MAP이 변경 되었으므로 HA와 CN에게 BU를 전송하여 바인딩 갱신을 시도한다. 이때 일시적으로 nMAP은 한 모바일 노드에 대해 두 경로(pRCoA, nRCoA)로 패킷을 수신하게 되고 nRCoA의 패킷이 pRCoA의 패킷보다 먼저 도착하는 문제가 발생할 수 있다. HIMIPv6에서는 이를 해결하기 위해 이중 버퍼링(double buffering)을 사용한다. nMAP은 pRCoA 패킷은 p-buffer에, nRCoA 패킷은 n-buffer에 저장하고 p-buffer의 패킷을 모두 전송한 후 n-buffer의 패킷을 전송함으로써 패킷순서 뒤바뀔 문제를 해결하였다.

이중 버퍼링을 사용할 경우 nMAP은 n-buffer를 비우기 전에 p-buffer에 더 이상 패킷이 오지 않는다는 것을 확인하기 위해 일정시간이 필요한데 이 시간의 길이가 너무 짧으면 비순차적으로 패킷을 수신하는 문제가 발생하고, 너무 길면 시간을 낭비하는 문제가 발생한다. HIMIPv6에서는 타이머를 사용하여 이 문제를 해결하였다<sup>[1]</sup>.

### III. IP 이동성관리 프로토콜의 TCP 성능 분석

본 장에서는 II장에서 분석한 IP 이동성관리 프로토콜들이 TCP 성능에 어떠한 영향을 끼치는지를 분석하기 위해 ns-2를 이용한 시뮬레이션을 수행하였다. 성능 분석을 위해 mobiwan<sup>[9]</sup>의 MIPv6 ns-2 모듈을 기반으로 하여 HMIPv6, FMIPv6 및 HIMIPv6 프로토콜 모듈을 ns-2에 구현하였다.

#### 3.1 실험환경 설정

실험을 위해 총 14개의 노드를 사용하였으며 토폴로지는 그림 3과 같다. 모바일 노드는 총 3번의 핸드오버를 수행하며, AR1과 AR2 사이, AR3과 AR4 사이에서는 Intra-MAP 핸드오버가 발생하고 AR2와

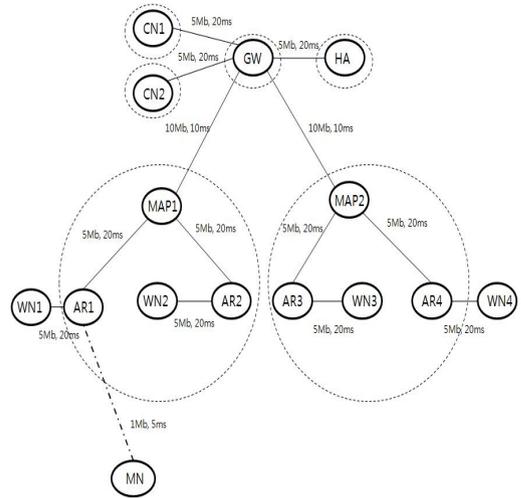


그림 3. 실험 토폴로지

AR3 사이에서는 Inter-MAP 핸드오버가 발생한다. 이때 발생하는 2계층 핸드오버는 하드 핸드오버라고 가정을 한다. 또한 2계층 연결 단락 시간은 150ms로 한다. 트래픽은 TCP 상에서의 FTP 파일을 사용하였고, 백그라운드 트래픽으로는 UDP 상에서의 CBR을 사용하였으며 패킷 사이즈는 1000 바이트로 하였다.

실험 대상은 HMIPv6와 버퍼를 사용한 FMIPv6, HIMIPv6를 사용하였고, FMIPv6에서 버퍼의 크기는 50 window size로 설정하였고 핸드오버의 시작1.5초 전에 FBU 메시지를 전송함으로써 프로토콜이 시작된다.

#### 3.2 패킷 에러가 없을 경우의 이동성 프로토콜 성능 분석

먼저 TCP Tahoe에서 이동성 프로토콜 동작에 관하여 살펴보면 TCP Tahoe에서는 패킷 손실이 발생 하였을 경우 slow-start로만 동작한다. 그림 4를 보면 Intra-MAP 핸드오버와 Inter-MAP 핸드오버 모두 핸드오버가 발생 한 이후에 slow-start로 동작하여 cwnd 값이 0부터 다시 시작하는 것을 볼 수 있다. HMIPv6에서는 2계층 연결 단락 시간동안 발생하는 패킷손실에 의한 타임아웃이 slow-start를 유발시킨다.

FMIPv6는 HMIPv6와는 다르게 2계층 단락 시간 동안에 전송 되는 패킷을 버퍼링하고 2계층 연결이 회복 된 후 수신자에게 버퍼링된 패킷을 전송함으로써 HMIPv6에서 발생했던 2계층 연결 단락 시간 동안의 패킷 손실을 예방한다. 그림 5를 보면 10초에 일어나는 핸드오버의 경우 cwnd 값이 0으로 초

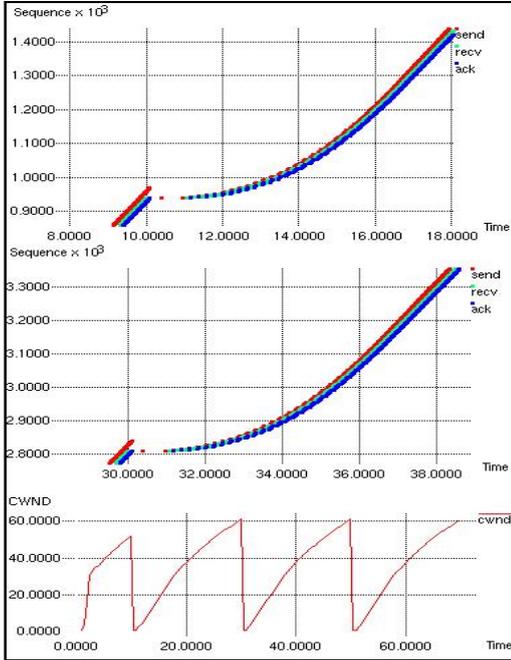


그림 4. TCP Tahoe에서 HMIPv6 동작

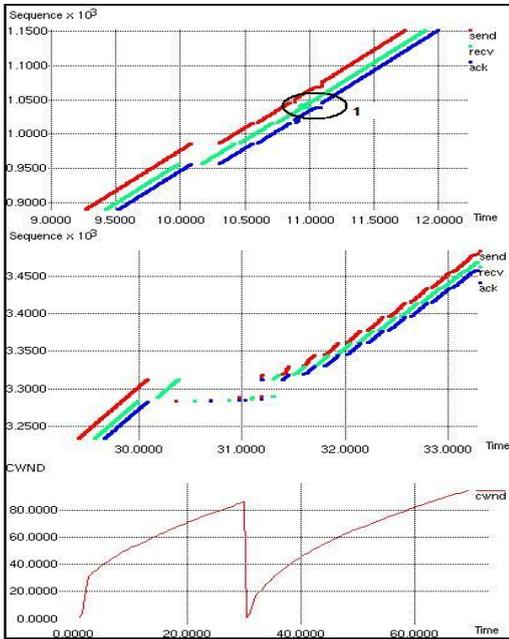


그림 5. TCP Tahoe에서 FMIPv6 동작

기화되지 않고 Predictive 모드로 동작한다. 하지만 30초에 발생하는 핸드오버에서는 Reactive 모드로 동작하는데, 그림 3의 토폴로지를 참조해 보면 30초에 일어나는 핸드오버의 경우 10초와 50초에 일

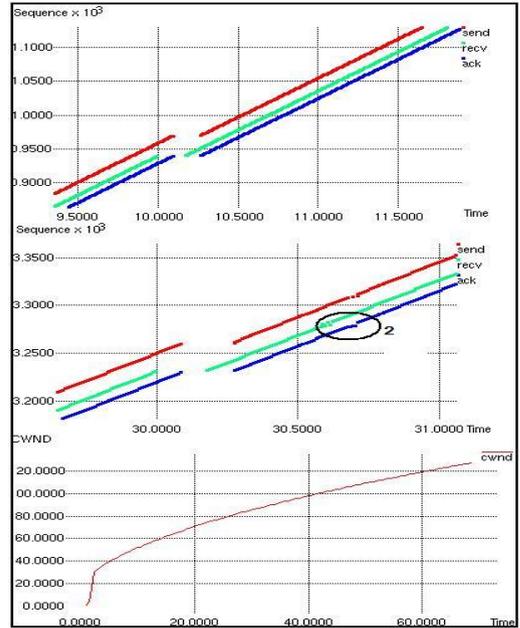


그림 6. TCP Tahoe에서 HMIPv6 동작

어나는 핸드오버 보다 20ms의 전송 지연을 더 가진다. FMIPv6는 전송 지연이 늘어남에 따라 사전 처리 시간동안 핸드오버에 필요한 모든 메시지를 전송하지 못하게 되어 Reactive 모드로 동작한다.

그림 6에서는 HMIPv6와 FMIPv6가 TCP Tahoe 상에서 패킷손실이 발생하는 것과 비교하여 HMIPv6의 경우 패킷손실 없이 seamless한 핸드오버가 이루어지는 것을 보여준다. 하지만 그림 6의 2에서 외같이 비순차적인 패킷 수신문제가 발생하는데 비순차적 패킷 수신은 그림 5의 1에서도 나타난다. 그림 5의 FMIPv6에서 비순차적인 패킷수신이 발생하는 이유는 패킷을 pAR로 터널링하기 때문인데 이를 방지하기 위해 HMIPv6에서는 이중버퍼링을 사용하였다. 이 이중버퍼링이 조기종료 될 경우 그림 6의 2처럼 비순차적인 패킷수신이 발생하게 된다. TCP Reno는 TCP Tahoe와는 3-dup-Ack이 발생하였을 때 fast-recovery로 동작한다는 점에서 차이가 발생한다. 3-dup-Ack이 발생하기 위해서는 패킷 에러가 발생해야 하는데 본 실험에서는 순수하게 핸드오버에서 발생하는 패킷 손실만을 확인하기 위하여 패킷에러를 추가 하지 않았다. 따라서 TCP Reno는 TCP Tahoe와 동일한 결과 값을 출력하게 된다. TCP Vegas의 경우 RTT값의 변화에 의해 cwnd값을 결정한다. 그림 7는 TCP Vegas에서 HMIPv6의 핸드오버를 보여준다. HMIPv6의 경

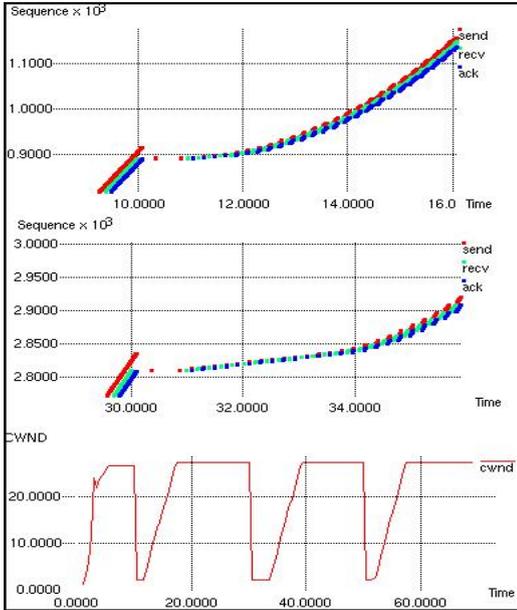


그림 7. TCP Vegas에서 HMIPv6동작

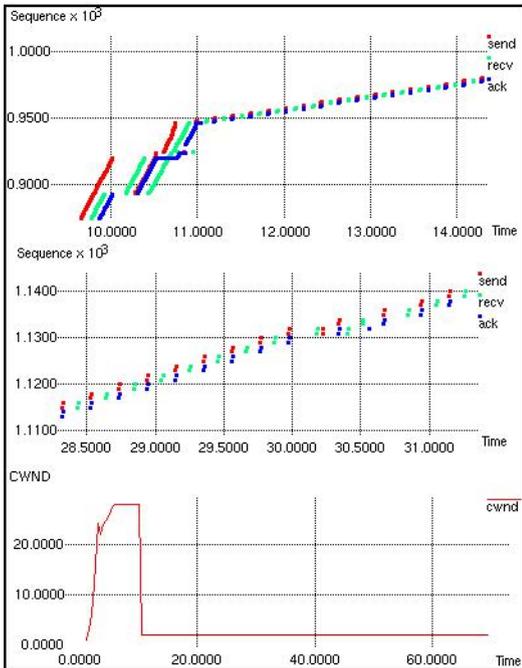


그림 8. TCP Vegas에서 FMIPv6동작

우 Inter-MAP 핸드오버, Intra-MAP 핸드오버 모두 타임아웃에 의하여 cwnd값이 초기화 되는데 Intra-MAP 핸드오버보다 Inter-MAP 핸드오버에서 회복속도가 더디게 나타난다. 이것은 그림 7의

cwnd 값에서 볼 수 있듯이 Inter-MAP 핸드오버는 Intra-MAP 핸드오버보다 링크를 회복하는데 더 많은 시간이 걸리기 때문에 발생한다.

그림 8에서는 TCP Vegas에서 FMIPv6가 동작하는데 심각한 문제가 발생하는 것을 보여 준다. 10 초에 핸드오버가 발생하여 연결이 회복 된 후 cwnd 값이 초기 값인 2에서 증가를 하지 않아 패킷 전송량이 크게 떨어졌다. FMIPv6에서는 모바일 노드가 FBBack을 수신한 경우 pAR은 nAR로 패킷을 포워딩한다. 하지만 모바일 노드는 FBBack 수신 후에도 일정시간 pAR에 머물러 있게 되어 패킷을 수신할 수 없게 되고 TCP Vegas는 전송한 패킷에 대한 Ack을 받지 못해 cwnd 값을 초기화 하게 된다. 2계층 링크가 회복 된 후에 TCP Vegas는 버퍼링 된 패킷에 대한 Ack을 모두 수신하게 되고 링크에 혼잡이 발생하지 않았다고 판단하게 되어 cwnd 값을 계속 초기 값으로 유지하게 된다.

HIMIPv6에서도 FMIPv6와 마찬가지로 핸드오버가 발생 할 경우 2계층 단락 시간동안 전송되는 패킷에 대한 버퍼링이 일어나지만 그림 9를 보면 FMIPv6에서와 같은 핸드오버에 의한 문제점이 발생하지 않는 것을 알 수 있다. HIMIPv6는 버퍼링을 할 때 FMIPv6와는 달리 사이멀캐스트를 한다. 따라서 HIMIPv6는 pAR과 연결이 끊어지기 전까지

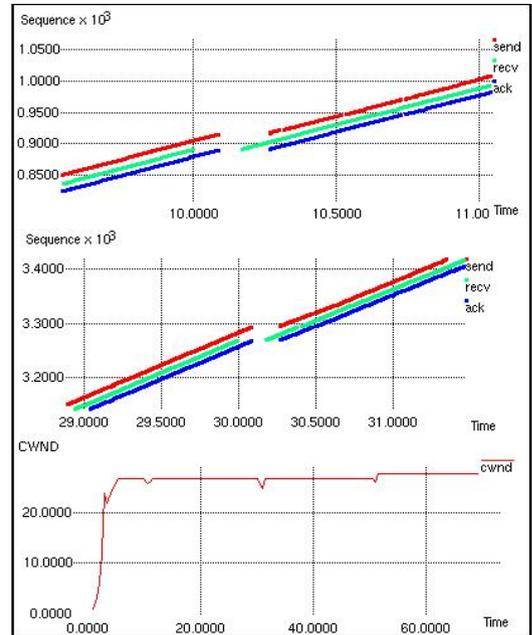


그림 9. TCP Vegas에서 HIMIPv6동작

패킷을 수신 할 수 있게 되고 TCP Vegas에서는 링크에 혼잡이 발생하지 않았다고 판단하여 핸드오버 전의 cwnd값을 유지하게 된다.

3.3 패킷 에러가 발생 할 경우의 이동성 프로토콜 성능 분석

이 실험에서는 그림 3의 토폴로지에서 AR들 (AR1, AR2, AR3, AR4)과 모바일 노드 사이의 링크에 패킷 에러를 추가 하였다. 이 때 에러율은 1%, 2%, 5%로 하였다.

표 1은 다양한 링크 에러 율에서 측정된 average TCP goodput과 average bandwidth consumption rate을 보여준다. Average TCP goodput은 실제 수신 단에서 수신한 데이터 전송률이고, average bandwidth consumption rate은 송신 단에서 수신 단으로 보낸 재전송을 포함한 데이터 전송률이다. HMIPv6의 경우 링크 에러가 발생하지 않았을 때에는 TCP Tahoe와 TCP Reno 보다 TCP Vegas가 성능이 떨어진다. 이것은 Inter-MAP 핸드오버 시 더딘 cwnd 증가에 의해 발생한 문제이다. 하지만 링크 에러가 발생할 경우 TCP Vegas가 가장 좋은 성능을 보여준다. TCP Tahoe와 TCP Reno가 패킷 손실에 의해 혼잡제어 하는 것과는 달리 TCP Vegas는 패킷의 RTT 값의 변화에 의해 혼잡제어를 하기 때문에 링크 에러가 생기더라도 TCP Vegas는 다른 두 TCP에 비해 그 영향을 적게 받는다. 또한 링크 에러가 없었을 때에는 TCP Tahoe와 TCP

Reno가 동일한 average TCP goodput 값을 보였지만 링크 에러가 발생하면서 TCP Reno에서 3-dup-Ack이 생기게 되어 TCP Tahoe보다 좋은 성능을 보여 준다.

FMIPv6의 경우 앞 절에서 설명한 TCP Vegas에서 RTT 증가로 인한 cwnd 초기화로 인하여 average TCP goodput값이 다른 TCP 에서의 average TCP goodput값 보다 현저하게 낮음을 알 수 있다. 하지만 링크 에러율이 커질수록 average TCP goodput의 차이가 작아진다. 링크 에러율이 커지게 되면 그만큼 cwnd 의 평균값이 낮아지게 되어 TCP Vegas에서 FMIPv6의 평균 cwnd 값과 비슷해지기 때문이다.

HIMIPv6의 경우 HMIPv6, FMIPv6와 비교했을 때 링크 에러율이 커지더라도 average TCP goodput의 값이 그 차이가 줄기는 하지만 꾸준히 높게 나오는 것을 볼 수 있다. 링크 에러율이 커지게 되면 핸드오버에 의한 패킷 손실 이외의 패킷 손실의 비중이 증가하게 되어 이동성 지원 프로토콜 간의 average TCP goodput 값의 차이가 줄어들지만 핸드 오버에 의한 패킷 손실이 계속 존재하기 때문에 HIMIPv6가 가장 좋은 성능을 보인다.

TCP 전송의 효율성을 측정하기 위해서 식 (1)과 같은 재전송 오버헤드 비율(Retransmission Overhead Ratio:ROR)을 정의하였다. ROR이 클 경우 재전송이 많이 발생하여 네트워크 자원의 낭비를 가져온다. 표 2에서 HMIPv6를 살펴보면 링크에러가

표 1. Average TCP goodput과 average bandwidth consumption rate

(단위 : Kbps)

|         |                     | Wireless link error rate 0% |          |           | Wireless link error rate 1% |          |           | Wireless link error rate 2% |          |           | Wireless link error rate 5% |          |           |
|---------|---------------------|-----------------------------|----------|-----------|-----------------------------|----------|-----------|-----------------------------|----------|-----------|-----------------------------|----------|-----------|
|         |                     | TCP Tahoe                   | TCP Reno | TCP Vegas | TCP Tahoe                   | TCP Reno | TCP Vegas | TCP Tahoe                   | TCP Reno | TCP Vegas | TCP Tahoe                   | TCP Reno | TCP Vegas |
| HMIPv6  | TCP goodput         | 769.28                      | 760.64   | 724.33    | 314.55                      | 347.85   | 460.95    | 228.66                      | 226.09   | 335.26    | 136.54                      | 138.88   | 172.74    |
|         | BW consumption rate | 781.5                       | 772.85   | 735.7     | 322.53                      | 354.93   | 472.54    | 236.01                      | 234.31   | 346.33    | 145.47                      | 147.91   | 184.15    |
| FMIPv6  | TCP goodput         | 888.08                      | 888.91   | 197.41    | 331.62                      | 388.95   | 260.67    | 242.61                      | 248.64   | 189.19    | 142.11                      | 146.49   | 169.62    |
|         | BW consumption rate | 890.31                      | 891.18   | 197.65    | 335.58                      | 392.49   | 263.66    | 247.71                      | 254.49   | 193.13    | 150.29                      | 154.1    | 179.19    |
| HIMIPv6 | TCP goodput         | 892.95                      | 892.93   | 916.59    | 358.88                      | 417.32   | 524.05    | 247.01                      | 267.35   | 389.6     | 148.25                      | 155.01   | 184.98    |
|         | BW consumption rate | 894.82                      | 894.8    | 918.02    | 363.18                      | 421.42   | 530.31    | 252.45                      | 272.94   | 398.03    | 156.48                      | 163.18   | 195.34    |

표 2. Retransmission Overhead Ratio

(단위 : %)

|         | Wireless link error rate<br>0% |             |              | Wireless link error rate<br>1% |             |              | Wireless link error rate<br>2% |             |              | Wireless link error rate<br>5% |             |              |
|---------|--------------------------------|-------------|--------------|--------------------------------|-------------|--------------|--------------------------------|-------------|--------------|--------------------------------|-------------|--------------|
|         | TCP<br>Tahoe                   | TCP<br>Reno | TCP<br>Vegas |
| HMIPv6  | 1.59                           | 1.61        | 1.57         | 2.54                           | 2.04        | 2.51         | 3.21                           | 3.64        | 3.3          | 6.54                           | 6.5         | 6.61         |
| FMIPv6  | 0.25                           | 0.25        | 0.12         | 1.19                           | 0.91        | 1.14         | 2.1                            | 2.35        | 2.08         | 5.76                           | 5.19        | 5.64         |
| HIMIPv6 | 0.21                           | 0.21        | 0.16         | 1.2                            | 0.98        | 1.19         | 2.2                            | 2.11        | 2.16         | 5.55                           | 5.27        | 5.6          |

0%일 경우 약 1.6%의 ROR을 보인다. 링크에러가 커질수록 ROR 값은 증가하며 링크에러가 5%인 경우 약 6.5%의 ROR을 보인다.

FMIPv6와 HIMIPv6의 경우 링크에러가 0%인 경우 매우 작은 ROR(약 0.2%) 값을 보여준다. 이는 FMIPv6와 HIMIPv6에서 핸드오버에 의한 패킷 손실이 작음을 보여준다. 링크 에러율이 증가하면 FMIPv6와 HIMIPv6 모두 ROR값이 증가하는데, 이는 링크 에러가 커지게 되면 IP 이동성 지원 프로토콜의 영향이 적어지고 패킷 에러에 의한 ROR 값이 증가하기 때문이다.

표 1과 표 2의 Average TCP goodput과 ROR 결과는 HIMIPv6가 다양한 에러율과 TCP 버전에서 가장 효율적으로 네트워크 자원을 활용하고 TCP 성능 면에서도 가장 좋은 특성을 가짐을 보여준다.

$$ROR = \frac{Average\ B/W\ Consumption - Average\ TCP\ Goodput}{Average\ TCP\ Goodput} \times 100 \quad (1)$$

#### IV. 결 론

본 논문에서는 HMIPv6, FMIPv6, HIMIPv6 등의 다양한 IP 이동성 관리 프로토콜에 관하여 살펴보고 각 IP 이동성 관리 프로토콜이 다양한 TCP에서 보이는 성능을 ns-2 시뮬레이션을 통해 분석하였다.

모든 링크 에러율과 TCP 버전에서 HIMIPv6가 가장 뛰어난 average TCP goodput과 ROR 성능을 보여주었다. FMIPv6는 TCP Vegas에서 심각한 문제가 발생하는데, 이는 핸드오버가 이루어지는 동안에 전송되는 패킷을 버퍼링 할 때 cwnd 값이 초기화 되어 증가하지 못하기 때문이다. HIMIPv6의 경우도 핸드오버가 이루어지는 동안 패킷을 버퍼링을 하지만 사이멀캐스트를 사용함으로써 FMIPv6에서 발생한

cwnd 초기화가 발생하지 않는다. 링크 에러가 발생한 경우 에러율이 커질수록 IP 이동성 프로토콜 간의 성능 차이가 줄어들지만 HIMIPv6는 지속적으로 가장 좋은 TCP 성능을 보여준다. 이는 에러율이 커져 핸드오버에 의한 패킷 손실에 대한 비중이 줄어들긴 하였지만 여전히 그 값이 성능에 영향을 미치기 때문이다.

본 논문에서는 ns-2를 이용한 시뮬레이션 성능평가를 통해 IP 이동성관리 프로토콜이 TCP 성능에 상당한 영향을 미침을 확인하였다. 구체적으로는 IP 이동성관리 프로토콜의 다양한 성능개선 기법 중 핸드오버 구간의 패킷 손실 방지를 위한 버퍼링 및 패킷 전달 지연시간을 줄이기 위한 사이멀캐스트 기법이 TCP 성능 개선에 많은 도움이 됨을 확인하였다.

#### 참 고 문 헌

- [1] 정현구, 김영탁, 맹승렬, 채영수 “HIMIPv6: 광대역 무선 통신 네트워크를 위한 효율적인 IP 이동성 관리 프로토콜”, 한국통신학회논문지 Vol 31 No.47, 2006
- [2] D. Jonhson and C. Perskins, “Mobility Support in IPv6”, IETF, RFC3775, 2004
- [3] H. Soliman, “Hierarchical Mobile IPv6 mobility management“, IETF, Internet Draft, 2004
- [4] R. Koodi, “Fast Handovers for Mobile IPv6”, IETF, RFC4068, 2005
- [5] K. Malki and H. Soliman, “Simultaneous Bindings for Mobile IPv6 Fast Handovers”, IETF, Internet Draft, 2005
- [6] R. Hsieh et al., “S-MIP: A Seamless Handoff Architecture for Mobile IP“, IEEE INFOCOM, 3, pp.1774-1784, 2003

- [7] Y. Gwon et al., “Scalability and Robustness Analysis of Mobile IPv6, Fast Mobile IPv6, Hierarchical Mobile IPv6, and Hybrid IPv6 Mobility Protocols Using a Large-scale Simulation“, IEEE Intl. conference on communication, 7, pp4087-4091, 2004
- [8] “Network Simulator version 2”, <http://www.isi.edu/nsnam>
- [9] “MobiWan: NS-2 extensions to study mobility in Wide-Area IPv6 Networks”, <http://www.inrialpes.fr/planete/pub/mobiwan>

**박 창 민 (Changmin Park)**

준회원



2006년 2월 영남대학교 정보통신공학과 학사  
2006년 3월~현재 영남대학교 정보통신공학과 석사 과정  
<관심분야> 네트워크, 모바일 네트워크

**채 영 수 (Youngsu Chae)**

정회원



1994년 2월 포항공과 대학교 전자계산학과 학사  
1996년 2월 포항공과 대학교 전자계산학과 석사  
2002년 12월 Ph.D., Georgia Institute of Technology  
2003년 3월~2004년 2월 삼성전자 통신연구소 책임연구원

2004년 3월~현재 영남대학교 전자정보공학부 조교수  
<관심분야> 모바일 네트워크, 멀티미디어 통신, 인터넷 서비스 아키텍처