

Mobile IPv6의 빠른 핸드오버 기법의 성능 개선*

조경산**, 엄희용**

Performance Improvement of Fast Handovers in Mobile IPv6

Kyungsan Cho, Heeyoung Um

Abstract

IETF proposed Fast Handover mechanism to reduce the latency during which the mobile node is effectively disconnected from the Internet. In this paper, we propose three buffering strategies and a bicasting strategy to resolve packet loss and packet disordering problems in Fast Handover mechanism. Through Java-based simulation, we show that packet loss and disordering problems have been resolved by our proposal. As the result, the buffering and bicasting strategies can reduce the handover delay caused by packet loss and packet disordering, in addition.

Key Words : Mobile IPv6, Fast Handover, Buffering, Bicasting

* 이 연구는 2001학년도 단국대학교 대학연구비의 지원으로 연구되었음.

** 단국대학교 정보컴퓨터학부

1. 서론

인터넷 기술의 빠른 발전 및 이용자수의 급진적인 증가와 함께 무선통신 기술의 발전으로 무선 이동 인터넷이라는 새로운 환경이 급속하게 발전하고 있다[2][5][8].

인터넷 통신 프로토콜 IP(Internet Protocol)에서는 호스트가 인터넷 서비스를 받기 위하여 접속되는 위치가 고정된다. 인터넷에 연결된 호스트가 IP주소를 변경하지 않고 다른 서브넷으로 이동한다면 접속이 불가능하여, 데이터의 송수신이 불가능하다. IP에 이동성을 제공하기 위해서 IETF(Internet Engineering Task Force)는 Mobile IP를 제안하였다[3]. Mobile IP는 IP 주소의 변환 없이 다른 위치로 이동하여 인터넷에 연결해서 통신을 계속 할 수 있다.

Mobile IP는 현재의 IP버전인 IPv4에서 시작되었는데 새로운 버전의 IPv6가 제안됨에 따라, IPv6에 이동성을 제공하는 프로토콜 Mobile IPv6도 제안되었다[4][12].

Mobile IPv6에서 핸드오버를 효율적으로 처리하여 핸드오버 처리 지연 시간을 줄이기 위해 빠른 핸드오프 기법과 빠른 핸드오버 기법이 제안되었는데, 이 제안들에는 Mobile IP가 갖는 패킷의 손실이나 패킷의 비순서화와 같은 문제의 해결책은 포함되지 않고 있다[9][11].

본 논문에서는 빠른 핸드오버 기법에서 패킷의 손실과 비순서화 문제를 해결하고 처리지연을 줄이기 위해 버퍼링 기법과 동시 전송 기법을 빠른 핸드오버 기법에 적용하고, 이에 의한 성능 영향을 분석하고 제시한다.

본 논문의 구조는 다음과 같다. 2장에서는 IP 및 IPv6의 특성, Mobile IPv6의 개요와 특성을 각각 설명하고, 3장에서는 Mobile IP가 갖는 핸드오버의 문제점과 빠른 핸드오버 기법을 설명한다. 4장에서는 빠른 핸드오버 기법의 문제 해결을 위해 버퍼링 기법과 동시 전송 기법을 적용하고, 5장에서는 제안한 여러 기법에 의한 성능 개선을 비교하고 분석하며, 6장의 결론으로 마무리한다.

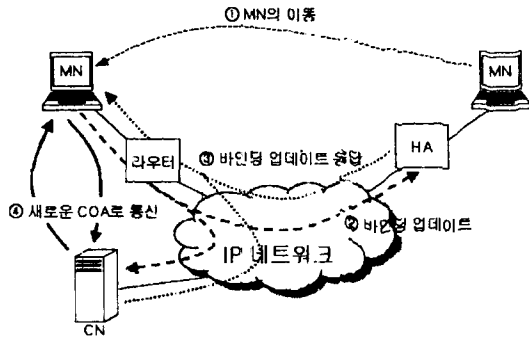
2. Mobile IP

IP는 호스트가 인터넷을 사용하기 위해서 접속되는 위치를 고정적으로 지정한다. 따라서 호스트는 접속된 네트워크로부터 유일한 IP주소를 부여받는다. 호스트가 IP 주소를 변경하지 않고 다른 서브넷으로 이동하면 호스트는 새로운 서브넷으로의 접속과 자료의 송수신이 불가능하다. IP에 호스트의 이동성을 제공하기 위해 IETF는 Mobile IP(Mobile IPv4)를 제안하였다[3]. Mobile IP의 기본적인 요소들은 MN(Mobile Node), CN(Correspondent Node), HA(Home Agent), FA(Foreign Agent), COA(Care-Of Address) 등으로 이루어진다. MN은 이동성을 가진 호스트를 말하고, CN은 MN과 통신하려는 호스트, HA는 MN이 처음 위치한 홈 네트워크에 있는 라우터로 MN이 이동 후에 MN의 현재위치에 대한 정보를 가지고 있다. 그리고 FA는 MN이 이동한 네트워크의 라우터이고, COA는 MN이 홈을 떠나 있을 때 외부 네트워크에서 통신에 사용되는 실제 주소이다. Mobile IP(Mobile IPv4)의 패킷 전송 과정은 다음과 같다. MN이 이동을 하면 새로운 네트워크의 FA로부터 COA를 부여받고 그 COA를 HA로 등록을 하면, HA는 MN으로 전송되는 패킷을 캡슐화시켜 등록된 COA로 전송하여, 이동된 MN으로 패킷이 전달된다[4].

IPv6의 등장에 따라 Mobile IPv6가 제안되었다[4]. Mobile IPv4와 달리 Mobile IPv6에는 MN이 HA와 직접통신을 하므로 FA가 필요하지 않고, 또한 보안이 필요한 모든 메시지들에 IPSec를 기본적으로 이용한다.

Mobile IPv6에서 MN의 이동에 따른 처리 과정은 <그림 1>과 같다. IPv6에서 라우터는 해당 네트워크의 prefix(전치부)값을 전송하고 이를 수신한 IPv6노드가 자신의 인터페이스 주소와 결합하여 IPv6주소를 만든다. 따라서 MN은 이 prefix를 이용해서 자신의 인터페이스 주소와 결합하여 자신이 이용할 COA를 스스로 만든다. COA는 실제로 CN에서 MN으로 패킷을 전송할 때 수신되는 주소이다. COA를 얻은 후, MN은

HA로 바인딩 업데이트 목적지 옵션을 포함한 패킷을 보내서 HA에게 자신의 바인딩을 등록한다. 바인딩 업데이트를 처리한 HA는 바인딩 승인을 MN으로 전송한다.



<그림 1> Mobile IPv6에서 MN 이동 후 처리 과정

HA에게 자신의 바인딩 정보인 COA를 등록한 MN은 COA를 IPv6 헤더의 소스 주소로 하는 패킷을 CN에게 전송할 수 있다. 패킷을 수신한 CN은 COA를 이용하는 소스 주소와 옵션으로 포함된 홈 주소를 이용하여 MN의 바인딩 정보를 등록하게 된다. 그러나, COA는 수명이 있으며, 만약 MN과 CN이 통신하는 동안에 바인딩 정보 중 COA의 수명이 거의 완료되어 간다면, CN은 MN으로 새로운 바인딩 정보를 보낼 것을 요청하는 바인딩 요청 메시지를 보낸다.

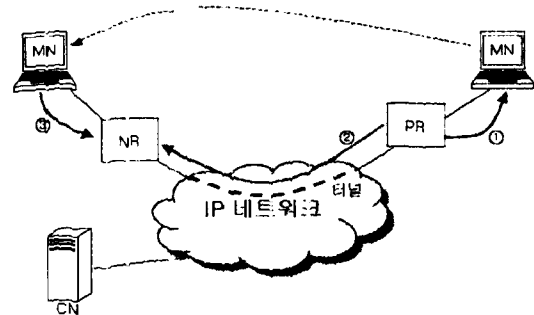
3. Mobile IPv6의 핸드오버

Mobile IP에서 MN이 현재의 서브넷에서 새로운 서브넷으로 이동하는 동작을 핸드오버(handover)라고 한다. 이 때, 현재 MN이 연결된 라우터를 이전 라우터(PR)라하고, MN이 이동한 후에 연결될 라우터를 새로운 라우터(NR)라한다. 새로운 서브넷에서 송수신을 하기 위해, MN은 새로운 주소(COA)를 부여받고 이를 HA 및 CN에게 등록해야 하는데, 이 과정이 수행되는 동안 일시적으로 통신이 중단되어 패킷이 손실되거나

또는 패킷들의 비순차적 전송이 일어날 수 있다 [1]. 패킷의 손실이나 비순서화가 일어나면 처리 지연도 길어진다.

핸드오버시 처리할 과정은 MN이 새로운 COA를 얻고 이를 등록하는 것이다. 새로운 COA를 얻는 과정에서 발생하는 지연의 원인은 이웃 발견(Neighbor Discovery), 라우터 광고와 라우터 요청, 중복 주소 발견(Duplicate Address Detection) 등의 처리이다. 핸드오버 처리 지연을 줄이기 위해 <그림 2>와 같이 처리되는 빠른 핸드오버라는 기법이 제시되었다[11].

빠른 핸드오버 기법에서는 네트워크가 MN의 이동을 감지하여 이전 라우터에게 통고하는 구조를 전제로 한다. 통고 받은 이전 라우터가 MN에게 새로운 라우터에 대한 정보를 포함한 NDR(Neighbor Discovery Redirection) 메시지를 전송함으로써, MN은 새로운 COA를 미리 얻어 바인딩 업데이트 메시지를 전송하여 핸드오버의 지연을 줄일 수 있다.



MN이 새로운 라우터로 이동하면,

- ① NDR을 MN에 전송
- ② 새로운 COA로 전송하는 길을 설정
- ③ MN엔트리를 입증하는 메시지를 전송. 엔트리가 확인되면 바인딩 업데이트를 받는다.

<그림 2> 빠른 핸드오버 기법

빠른 핸드오버 기법은 핸드오버의 처리 지연을 감소시킬 수 있으나 패킷의 손실 문제와 비순서화 문제에 대한 해결을 포함하지 않는다. 즉 핸드오버시에 새로운 링크를 구성하고 바인딩 업데이트를 승인 받기까지의 처리 시간은 줄지만,

그 동안에 MN이 수신 받아야할 패킷이 이전 라우터에서 새로운 라우터로 전송되면 새로운 라우터는 그 패킷을 MN으로 전달을 못하고 버리게 되면 패킷이 손실될 것이다. 또한, 핸드오버 처리 도중에는 메시지가 이전 라우터를 통해 전송된다. 핸드오버 직후 이전 라우터가 수신된 패킷을 새로운 위치의 MN에 전달하는데, 이들이 CN에서 전송하는 패킷보다 늦게 도착할 수 있다. 이로 인해 패킷의 비순서화가 발생한다. MAP을 갖는 계층적 mobile IPv6를 제안하고 노드의 이동에 따른 메시지를 감소시켜 지연을 줄이려는 시도가 있었지만, 이 역시 패킷의 손실 및 비순서화 문제는 해결하지 못하였다[7]

4. 버퍼링 및 동시 전송 기법 적용

빠른 핸드오버 기법은 Mobile IP에서의 처리 지연을 줄일 수 있지만, 패킷의 손실 및 비순서화 문제는 고려하지 않았다. 일반적으로 패킷의 손실 문제는 버퍼링 기법으로 해결이 가능하데, 본 장에서는 버퍼의 위치 및 관리에 따른 세 가지 버퍼링 구조와 동시 전송 기법을 Mobile IPv6의 빠른 핸드 오버 기법에 적용하여 패킷 손실을 방지하고 비순서화 문제를 해결하며 또한 패킷 전송의 지연을 줄이는 빠른 핸드오버 기법의 성능 개선을 제시한다.

4.1 이전 라우터의 버퍼링

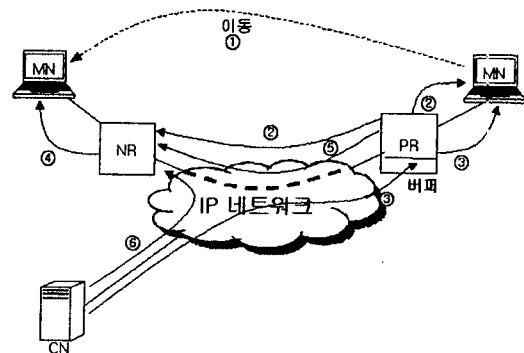
가장 간단한 구조로 핸드오버시 손실 가능성이 있는 패킷을 이전 라우터에 임시로 버퍼링하는 구조가 가능하다. 핸드오버 완료 후에 버퍼링된 패킷을 새로운 라우터를 통해 MN으로 전송하면 패킷의 손실을 예방할 수 있다.

이전 라우터에서의 버퍼링은 이전 라우터가 MN에게 NDR 메시지를 보내는 과정, 이전 라우터에서 버퍼링하는 과정, 이전 라우터의 버퍼에 저장된 패킷을 MN으로 전달하는 과정으로 진행된다.

네트워크로부터 MN의 이동성을 통고 받은 이전 라우터는 NDR 메시지를 MN에게 보내서 이동되어 접속될 새로운 라우터에 대한 정보를 알려주는 과정이 첫 단계이다.

두 번째 단계로 버퍼링 과정에서 이전 라우터는 새로운 라우터에게 MN의 이동과 버퍼링 시작을 알리고, 패킷을 버퍼링하면서 동시에 MN이 핸드오버가 완료되기 전까지 패킷을 MN에 전달한다.

핸드오버가 완료된 후에 새로운 라우터가 이전 라우터에게 버퍼링된 패킷을 요청하는 메시지를 보내면, 이전 라우터는 버퍼링된 패킷을 새로운 라우터에게 전송한다. <그림 3>은 처리 과정을 보여준다.



- ① MN의 이동
- ② NDR 메시지/ICMP 핸드오버 메시지
- ③ 버퍼링/MN으로 전달
- ④ 바인딩 업데이트/바인딩 응답
- ⑤ 버퍼링된 패킷을 NR로 전송
- ⑥ CN에서 MN으로 패킷 전송

<그림 3> 이전 라우터에서의 버퍼링 기법

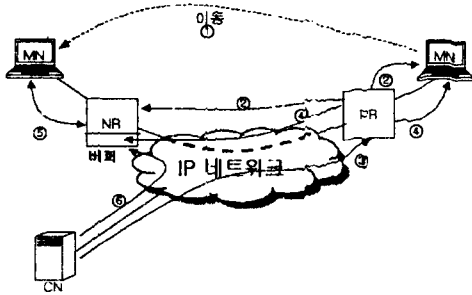
핸드오버가 시작되면 CN에서 MN으로 전송되는 패킷을 이전 라우터의 버퍼에 저장했다가, 핸드오버 후에 MN이 바인딩 업데이트로 이동 완료를 알리면 버퍼에 저장되어 있는 패킷을 MN으로 전송하므로 패킷의 손실은 해결된다. 하지만 이전 라우터의 버퍼에 있는 패킷을 MN으로 전송하는데 지연이 발생하며, 버퍼에 있는 앞

선 순서의 패킷이 CN에서 MN으로 전송한 나중 순서의 패킷보다 늦게 도착할 경우에는 패킷의 순서화가 어긋날 수 있다.

4.2 새로운 라우터의 버퍼링

앞 절의 분석과 같이 이전 라우터의 버퍼링에서는 패킷의 비순서화 문제가 생길 수 있고, 이전 라우터의 버퍼에서 새로운 라우터를 통해 MN으로 패킷을 전달하는 지연이 생긴다. 이 문제를 해결하기 위해 핸드오버 시에 새로운 라우터에서 패킷을 일시로 버퍼링하는 방법이 가능하다. 버퍼된 패킷은 핸드오버 완료 시에 MN으로 전송하여 손실과 비순서화를 방지할 수 있다.

이전 라우터가 MN에게 NDR 메시지를 보내는 과정, 새로운 라우터가 버퍼링하는 과정, 새로운 라우터의 버퍼에 저장된 패킷을 MN으로 전달하는 과정으로 수행된다. 첫 번째 단계에서는 이전 라우터가 MN에게 NDR 메시지를 보내며, 이는 이전 라우터의 버퍼링과 같다.



- ① MN의 이동
- ② NDR 메시지/ICMP 핸드오버 메시지
- ③ PR에서 MN으로 전송되는 패킷 수신
- ④ NR로 전달/NR에서 버퍼링
- ⑤ 바인딩 업데이트/바인딩 응답
- ⑥ CN에서 MN으로 패킷 전송

<그림 4> 새로운 라우터에서의 버퍼링 기법

두 번째 단계에서는 MN의 이동성을 통고 받은 이전 라우터가 새로운 라우터로 MN의 이동을 알리면 새로운 라우터는 버퍼링을 시작하

고, 이전 라우터는 CN에서 MN으로 전송되는 패킷을 MN과 새로운 라우터에 전달한다.

세 번째 단계에서 핸드오버가 완료된 후에 버퍼에 저장된 패킷을 MN으로 전송한다. 이때 CN로부터 MN으로 패킷이 전송되면 새로운 라우터는 버퍼링되었던 패킷이 모두 전송된 후에 MN으로 전송되도록 한다. 따라서 패킷들은 순서대로 전송된다. <그림 4>는 이 과정을 보여준다.

만약 MN이 새로운 라우터로 이동한 직후 다시 다른 라우터 또는 이전 라우터로 이동한 경우에는 새로운 라우터에 있는 버퍼의 효율성이 떨어질 수가 있고, 이전 라우터와 새로운 라우터 사이에 통신량이 증가한다.

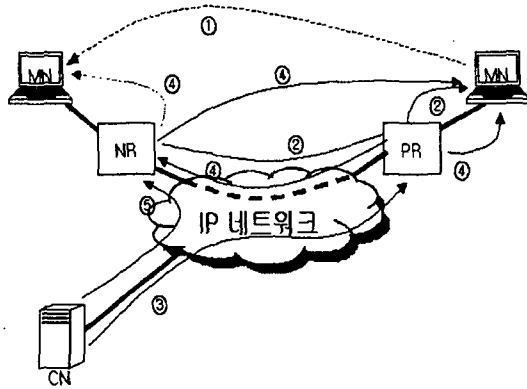
4.3 동시 버퍼링 기법

앞 절에서 이전 라우터의 버퍼링은 패킷의 비순서화가 발생할 수 있고, 새로운 라우터의 버퍼링은 MN의 빈번한 이동시에는 효율이 떨어진다고 제시되었다. 이 두 문제를 모두 해결하기 위해 새로운 라우터와 이전 라우터에서 모두 버퍼링을 하는 방법이 사용될 수 있다. MN이 즉시 이전 라우터로 다시 이동할 경우에는 새로운 라우터에 버퍼링된 패킷은 모두 무시하고 이전 라우터에 버퍼링된 패킷을 MN에게 전송한다.

따라서, 이전 라우터와 새로운 라우터의 버퍼에 모두 패킷을 버퍼링하여 MN이 빈번하게 이동했을 때 발생하는 지연을 줄일 수 있다. 하지만 이 기법은 두 라우터에 모두 버퍼를 두고 관리해야 하는 비용과 통신량이 증가한다.

4.4 동시 전송 기법

이 기법은 버퍼를 사용하는 대신에 가능한 두 경로(이전 라우터와 새로운 라우터)를 통해 동시에 패킷을 전송하여 지연과 손실을 줄일 수 있다. 동시 전송 기법은 이미 smooth handover 처리를 위해 유사한 연구가 제안된 바 있다[9][10]. <그림 5>는 빠른 핸드오버 기법에 동시 전송을 적용한 과정을 보여준다.



- ① MN의 이동
- ② NDR 메시지/ICMP 핸드오버 메시지
- ③ PR에서 패킷 수신
- ④ 동시 전송
- ⑤ CN에서 새로운 COA로 패킷 전송

<그림 5> 동시 전송 기법

핸드오버가 시작되면 이전 라우터가 MN에게 NDR 메시지를 전송하고, MN의 관련 정보와 동시 전송 시작을 새로운 라우터에게 알린다. 이전 라우터는 CN에서 MN으로 전송되는 패킷을 MN과 새로운 라우터로 동시 전송하고, MN이 핸드오버를 완료하면 새로운 라우터로만 패킷을 전송한다. MN이 연결이 되면, 새로운 라우터는 MN에게 패킷 전송을 시작한다. 바인딩 업데이트 후에도 이전 라우터로 전송되는 패킷이 있을 수 있으므로 이전 라우터는 패킷을 새로운 라우터로 일정 시간동안 전송을 해서 패킷 손실을 방지한다. 버퍼를 사용하지 않으므로 비용과 자원을 줄일 수 있으나, 두 개의 라우터를 통해 동시 전송하므로 네트워크의 통신량은 증가한다.

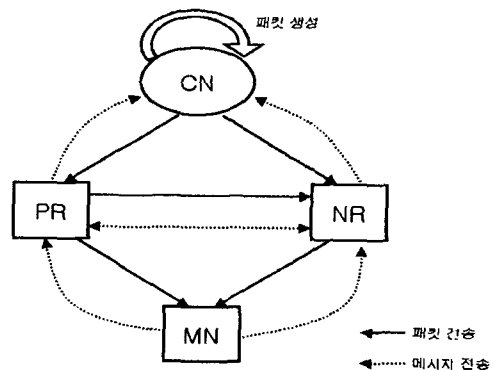
5. 성능 분석

본 장에서는 시뮬레이션을 통해 앞장에서 제안된 구조의 성능 개선 정도를 분석하여 제시한다. 시뮬레이션은 JAVA로 구현하였으며, 앞장에서

서 제안한 4가지 기법에 대해 핸드오버로 인한 패킷의 비순서화 비율, 패킷의 손실을 및 그로 인한 패킷의 지연시간과 이전 라우터와 새로운 라우터 사이의 증가된 통신량 등을 분석하였다.

5.1 시뮬레이션 환경 및 모델

시뮬레이션의 구성 및 실행에 있어 Mobile IP를 구성하는 각 노드와 핸드오버의 처리를 위한 각 사건들을 객체로서 정의할 수 있고, 상속성과 캡슐화의 객체 지향 시뮬레이션의 특성을 활용하기 편리하고, 또한 시뮬레이션 실행의 스레드 처리와 네트워크 프로그램이 쉽기 때문에 시뮬레이션은 JAVA로 구현하였다. 시뮬레이션은 빠른 핸드오버 기법을 기본 모델로 하여, CN과 MN 그리고 이전 라우터와 새로운 라우터 등은 각각 객체로 구현하였다. 시뮬레이션에서 기본이 되는 빠른 핸드오버 기법의 모델은 <그림 6>과 같다. 시뮬레이션은 <그림 2>~<그림 5>에서 제시된 바에 따라 객체로 구현된 CN, MN, PR 및 NR를 통한 패킷의 전달 과정을 구현하였고, 각 과정에서 필요한 정보를 측정하였다.



<그림 6> 시뮬레이션을 위한 기본 모델

시뮬레이션에서 고려된 기본 가정은 다음과 같다. 첫째, CN에서 이전 라우터로 패킷을 전송하는 시간과 새로운 라우터로 패킷을 전송하는 시간은 같다고 가정한다. 둘째, 라우터에서 MN

으로 패킷을 전송하는 시간은 무시한다. 셋째, CN에서 라우터로 패킷을 전송할 때, 패킷의 손실은 없다고 가정한다. 넷째, 패킷의 크기는 모두 같다고 가정한다.

5.2 성능 분석 및 결과

기존 연구와의 비교를 위하여 [1]과 동일한 환경과 조건으로 시뮬레이션을 수행하였으며, 각 기법에 대하여 20,000번의 핸드오버가 발생하도록 하였다. 제안된 구조에 의한 성능 영향만을 분석하기 위해 시뮬레이션 환경은 가능한 간단히 하였으며, 결과도 평균값만을 제시하여 비교하였다. 결과에서 비교되는 여러 기법은 다음과 같다. 기존방법은 버퍼링 기법 또는 동시 전송 기법을 적용하지 않은 빠른 핸드오버 기법이고, 기법1은 이전 라우터의 버퍼링 기법, 기법2는 새로운 라우터의 버퍼링 기법, 기법3은 동시 버퍼링 기법, 기법4는 동시 전송 기법이 적용된 경우를 각각 의미한다.

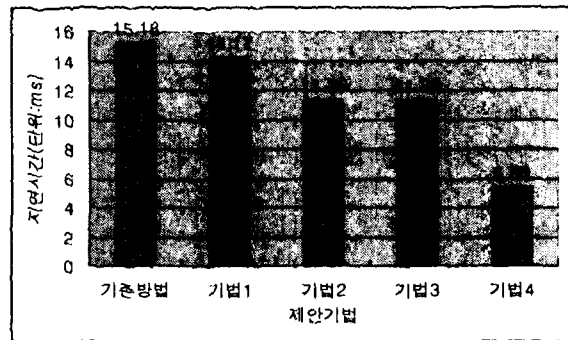
핸드오버 처리과정에서 발생한 패킷의 손실율의 분석 결과, 기존방법은 전송 지연 시간에 따라 최대 8.43%의 손실율이 나타났다. 하지만 버퍼링 기법과 동시 전송 기법을 사용한 기법1, 기법2, 기법3, 기법4에서는 패킷의 손실이 전혀 발생하지 않았다.

<표 1> 비순서화된 패킷의 비율 (단위 : %)

전송 지연시간 \ 기법	기법	기법1	기법2 기법3 기법4
	기존방법		
10(ms)	7.31	5.42	0
20(ms)	7.65	5.75	0
30(ms)	7.97	6.10	0
40(ms)	8.24	6.45	0
50(ms)	8.62	6.77	0

<표 1>은 MN까지의 전송된 패킷들 중에서 비순서화된 패킷의 비율을 분석하였다. 즉, 핸드오버 처리과정에서의 비순서화된 패킷의 비율을 구한 것이다. 이전 라우터의 버퍼링 기법(기법1)은 패킷 손실은 방지하지만, 패킷의 비순서화 문제를 완전히 해결할 수 없으며, 기법2, 기법3, 기법4는 패킷의 비순서화 문제를 해결하여 비순서화된 패킷은 언제나 0%가 된다.

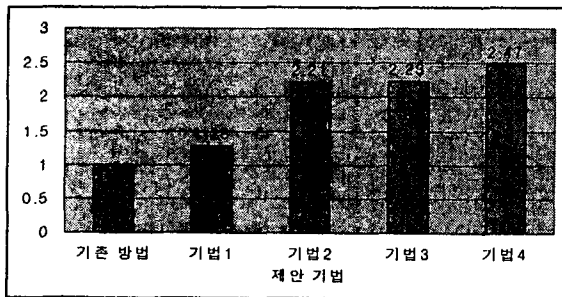
<그림 7>은 핸드오버 처리의 지연시간 분석이다. 기존방법은 패킷의 손실과 비순서화로 인해 지연시간이 증가하며 기법1, 기법2, 기법3은 버퍼링 처리시간이 지연시간에 더해진다. 여기서 기법1의 지연시간이 기법2와 기법3보다 큰 이유는 비순서화의 처리 시간과 이전 라우터에 버퍼링된 패킷을 MN에게 전송하는 시간이 더해지기 때문이다. 기법 4는 버퍼링 처리 시간이 전혀 필요하지 않으므로 가장 적은 지연을 보인다. CN과 라우터 사이의 전송지연 시간에 의한 영향은 미비한 것으로 분석된다. 손실된 패킷의 처리를 위한 상위 계층의 프로토콜(TCP등)은 고려하지 않았다.



<그림 7> 패킷의 지연시간

<그림 8>은 이전 라우터와 새로운 라우터 사이의 통신량 분석 결과이다. 제시된 값은 MN으로 전송에 의한 통신량만을 분석한 것이며, 다른 전송에 따른 네트워크에 의한 영향은 고려하지 않았다. 기존방법에서의 통신량을 1로 하였을 경우에, 각 기법들에 의한 상대적인 통신량을 표시

하였다. 기법1은 이전 라우터의 버퍼링 되어 있는 패킷을 MN으로 전달하므로 통신량이 약간 증가하였다. 기법2와 기법3은 핸드오버 도중에 이전 라우터가 패킷을 MN과 버퍼링을 위해 새로운 라우터에게 전송하는데, 중복되는 부분이 있으므로 통신량이 유사하게 증가한다. 기법4는 이전 라우터가 MN과 새로운 라우터로 동시 전송하므로 기존방법에 비해 패킷의 통신량이 더욱 증가한다.



<그림 8> 패킷의 통신량

6. 결론

최근 무선 인터넷의 수요가 증가함에 따라 효율적인 통신을 위해 Mobile IP가 제시되었다. Mobile IP에서는 핸드오버 처리에 따른 지연 문제가 일어나므로, 핸드오버 시간을 줄이기 위한 빠른 핸드오버 기법이 제안되었다. 하지만 빠른 핸드오버 기법은 핸드오버로 인한 패킷의 손실문제, 패킷의 비순서화 문제의 해결책은 포함하지 않는다. 본 논문에서는 패킷의 손실과 비순서화의 개선을 위해 이전 라우터와 새로운 라우터에서 버퍼링하는 기법과 동시 전송 기법을 빠른 핸드오버 기법에 적용하였다.

시뮬레이션 결과에 의하면, 버퍼링 기법과 동시 전송 기법이 패킷의 손실을 방지할 수 있었고 패킷의 비순서화 문제도 해결할 수 있으며, 또한 핸드오버로 인한 지연시간 역시 크게 감소되었으나, 라우터 사이의 통신량이 증가되는 문제가 있음을 보였다.

이전 라우터의 버퍼링은 구조가 간단하며 패

킷의 손실을 없앨 수 있었다. 새로운 라우터의 버퍼링은 패킷의 손실뿐만 아니라 패킷의 비순서화 문제도 해결하였지만, MN의 잦은 이동시에는 활용도가 미흡하고 이전 라우터와 새로운 라우터 사이의 통신량이 다소 증가되는 단점이 있다. 동시 전송 기법은 패킷 손실, 패킷 비순서화 문제는 해결하였지만, 통신량이 증가되는 단점이 있다. 각 기법들의 특성에 따라 성능적 영향이 다르므로, 실제 통신 환경에 적절한 버퍼링 또는 동시 전송 기법의 적용으로 빠른 핸드오버 기법의 성능 개선이 가능하다.

참고문헌

- [1] 허성진 외 4인, "이동 인터넷 환경에서 핸드오버시 패킷 비순서화로 인한 성능 저하 문제의 개선방안", 정보과학회논문지, 제 25 권, 제 12 호, pp.1424-1430, 1998.
- [2] C. Cox, "Wireless Personal Communications What is it?," IEEE Personal Communications, pp.20-35, April, 1995.
- [3] C. E. Perkins, "IP Mobility Support," Internet RFC 2002, October, 1996.
- [4] D. Johnson and C. Perkins, "Mobility Support in IPv6," draft-ietf-mobileip-ipv6-12.txt, February, 2000.
- [5] G. Forman and J. Zahorjan, "The Challenges of Mobile Computing," IEEE Computer, pp.38-47, April, 1994.
- [6] G. Krishnamurthi, R. Chalmers and C. Perkins, "Buffer Management for Smooth HandOvers in Mobile IPv6," draft-krishnamurthi-mobileip-buffer6-01.txt, March, 2001.
- [7] Hesham Soliman, "Hierarchical MIPv6 mobility management," draft-ietf-mobileip-hmipv6-02.txt, February, 2001.
- [8] Jabbari and G. Colombo, "Network Issues for Wireless Communications," IEEE Com

- munications Magazine, pp.88-98, January, 1995.
- [9] K. El-Malki and H. Soliman, "Fast Handoff in MIPv6," draft-elmalki-handoffsv6-01.txt, November, 2000.
- [10] Rajeev Koodli and Charles E. Perkins, "A Framework for Smooth Handovers with Mobile IPv6," draft-koodli-mobileip-smoothv6-02.txt, November, 2000.
- [11] R. Koodli and C. Perkins, "Fast Handovers in Mobile IPv6," draft-koodli-mobileip-fastv6-02.txt, March, 2001.
- [12] S. Deering and R. Hinden, "Internet Protocol Version 6 (IPv6) Specification," RFC 2460, December, 1998.

● 저자소개 ●

조경산



1979 서울대학교 전자공학과 학사
 1981 한국과학기술원 전기전자공학과 석사
 1988 Univ. of Texas at Austin 전기전산공학과 박사
 1988~90 삼성전자 컴퓨터부문 책임연구원
 1990~현재 단국대학교 정보컴퓨터학부 교수
 관심 분야 : 컴퓨터 시스템 설계 및 분석, 네트워크 시스템 설계 및 분석, 웹 응용, 시뮬레이션

엄희용



2000 단국대학교 전산통계학과 학사
 2002 단국대학교 전산통계학과 석사
 관심 분야 : 이동 통신, 웹 응용, 시뮬레이션