

무선 LAN에서 트랜스포트 프로토콜 성능 향상을 위한 이동 IP 핸드오프 프로토콜

(A Mobile IP Handoff Protocol for Performance Enhancement of Transport Protocol over Wireless LAN)

박지현[†] 진현욱^{**} 유혁^{***}

(Jee-Hyun Park) (Hyun-Wook Jin) (Hyuck Yoo)

요약 본 논문에서는 무선 LAN(Local Area Network) 환경에서 이동 IP를 기반으로 하는 이동 호스트가 핸드오프를 수행할 때 발생될 수 있는 패킷 손실을 제거하고, 신뢰성 데이터와 멀티미디어 데이터의 특성을 모두 고려한 새로운 핸드오프 프로토콜을 제안한다. 제안된 이동 IP 핸드오프 프로토콜은 데이터 링크 계층의 정보를 이용하여 이동 호스트가 방문할 FA를 예측한 후, 방문할 FA가 패킷을 버퍼링하게 함으로써 패킷 손실을 제거하고 버퍼링된 패킷의 포워딩 시간을 최소화 한다. ns-2(network simulator-2)를 이용한 시뮬레이션 결과, 제안된 이동 IP 핸드오프 프로토콜을 사용하는 전송 프로토콜은 패킷 손실을 경험하지 않았으며, TCP의 경우 표준 이동 IP를 사용하는 TCP 보다 최대 6.25배 우수한 처리율을 보였다.

키워드 : 이동 IP, 무선 랜, 핸드오프, 데이터 링크 계층, TCP, UDP

Abstract When a Mobile IP handoff occurs, the packets in flight can be lost because they are tunneled based on out-of-date location information. In this paper, we propose an enhanced handoff protocol that achieves no packet loss during Mobile IP handoff over a wireless LAN. Our handoff protocol predicts the next foreign agent that a mobile host is to visit by using the information from the data link layer of wireless LAN. After that, when a Mobile IP handoff occurs, the current foreign agent forwards the packets destined to a mobile host to the predicted foreign agent which buffers them. This eliminates packet loss and reduces the packet forwarding delay. Our handoff protocol is simulated using the Network Simulator-2 (ns-2) and shows the substantial performance enhancement of TCP with much less overhead up to 6.2 times compared to standard Mobile IP.

Key words : Mobile IP, Wireless LAN, Handoff, Data Link Layer, TCP, UDP

1. 서론

하드웨어 기술이 발달하여 휴대형 단말기(노트북, PDA)의 보급 및 수요가 폭발적으로 증가함에 따라 기존의 IP를 기반으로 하는 데이터 통신에 무선 이동 통

신의 개념을 접목시키려는 시도가 활발하게 진행되고 있다. 이러한 흐름의 대표적인 기반 기술이 이동 IP (Mobile IP)[1]이다. 현재 무선 LAN 환경에서 이동 IP를 사용하는 호스트는 시간과 위치에 관계없이 이동하면서 인터넷에 접속하여 데이터 서비스를 받을 수 있다. 그러나 이러한 IP를 기반으로 하는 무선 이동 통신이 널리 보급되기 위해서는 많은 기술적인 문제들이 해결되어야 한다. 그 중 하나가 이동 호스트의 핸드오프 수행으로 인한 전송 프로토콜의 성능 저하이다.

무선 LAN 환경에서 이동 IP를 사용하는 호스트는 현재의 셀을 벗어나서 새로운 셀로 이동할 때, OSI(Open System Interconnection) 계층 중 두 계층에서 핸드오프를 수행해야 한다. MAC(Media Access Control) 계

· 본 연구는 2000년 정보통신연구진흥원 지원(과제번호 C1 98 229)과 2001년 정보통신부 대학 S/W 연구센터 지원사업에 의해 연구됨.

[†] 비회원 : 모토로라 연구원

erin@motorola.com

^{**} 학생회원 : 고려대학교 컴퓨터학과

hwjin@os.korea.ac.kr

^{***} 종신회원 : 고려대학교 컴퓨터학과 교수

hxy@os.korea.ac.kr

논문접수 : 2001년 1월 26일

심사완료 : 2002년 4월 9일

층에서 이루어지는 핸드오프는 새로운 셀에서 신뢰성 있는 무선 링크를 확보하기 위한 것이며, IP 계층에서 일어나는 이동 IP 핸드오프는 IP 주소의 변화 없이 이동 중에도 서비스를 계속 유지할 수 있도록 하는 호스트의 위치 투명성 제공을 목적으로 한다. 그런데 이동 IP 핸드오프 중에 이동 호스트로 전달될 패킷의 라우팅 경로가 올바르게 갱신되지 않으면 패킷 손실이 발생할 수 있다. 더욱이 IP를 기반으로 하는 대표적인 전송 프로토콜인 UDP, TCP는 이동성을 고려하지 않고 구현되어 이러한 패킷 손실에 대한 대안을 가지고 있지 않다. 따라서 기존 UDP 혹은 TCP를 사용하는 이동 호스트는 핸드오프 수행으로 예기치 않은 성능 저하를 경험할 수 있다[2][3][4].

현재 많은 연구가 위와 같은 이동 IP 핸드오프 수행으로 인한 패킷 손실 문제를 해결하기 위해서 진행되고 있고, 여러 가지 대안 중에서 패킷 버퍼링 기법이 대표적으로 사용된다[5][6][7]. 그러나 부분별한 패킷 버퍼링은 CPU 및 메모리 자원의 낭비를 초래할 뿐만 아니라, 중복된 패킷을 이동 호스트에게 전달함으로써 네트워크 대역폭을 낭비하게 된다. 또한 버퍼링된 패킷은 새로운 FA로 포워딩된 후 이동 호스트로 전달되기 때문에 큰 지연 시간으로 인하여 타임 아웃 특성이 있는 TCP 데이터와 시간 제약이 있는 멀티미디어 데이터를 사용하는 응용 프로그램의 성능 저하를 막는데 한계가 있다.

본 논문에서는 위와 같은 문제점을 해결하기 위해 새로운 핸드오프 프로토콜을 제안한다. 제안된 핸드오프 프로토콜은 패킷 손실을 제거하기 위하여 버퍼링 기법을 사용한다. 하지만 기존 연구들과는 다르게 무선 LAN(IEEE 802.11)의 데이터 링크 계층 특성을 이용하여 정확하게 버퍼링 시점을 결정하기 때문에 불필요한 패킷이 버퍼링 되는 것을 막고, 중복된 패킷 전달에 대한 우려도 제거한다. 또한 이동 호스트에게 전달될 패킷을 현재 FA가 아닌 데이터 링크 계층의 정보를 이용하여 예측된 FA에 터퍼링함으로써 패킷 포워딩 시간을 최소화 하여, 시간 제약이 있는 데이터 통신에 적용해서 실효성은 거둘 수 있도록 하였다. 물론 셀이 중첩되지 않을 경우에는 이동 호스트가 방문할 FA를 알 수 없다. 이와 같은 경우에는 현재 FA가 버퍼링을 하게 함으로써 셀 구조에 관계없이 전송 프로토콜의 성능을 보장한다.

본 논문의 구성은 다음과 같다. 본 장에 이어 2장에서는 연구 배경을 알아보고, 3장에서는 새로운 핸드오프 프로토콜을 제안한다. 4장에서는 ns-2를 이용한 시뮬레이션을 통하여 제안된 핸드오프 프로토콜의 성능을 분석한다. 그리고 5장에서는 본 논문의 결론을 맺는다.

2. 연구 배경

본 장에서는 본 연구의 배경이 되는 무선 LAN, 이동 IP의 기본 개념을 알아보고, 이동 IP 핸드오프로 인한 성능 저하를 막기 위한 기존의 연구들을 살펴본다.

2.1 무선 LAN

2.1.1 무선 LAN 구조 및 구성 요소

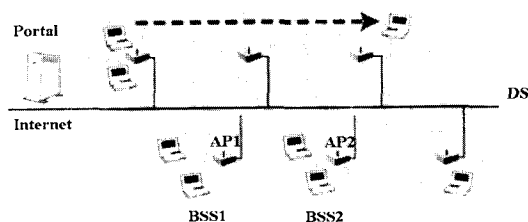


그림 1 유무선 망이 복합된 인프라스트럭처 네트워크

무선 LAN은 이동성(mobility)과 확장성(scalability)을 제공하는 데이터 통신 시스템으로 기존 유선 LAN에 비하여 구축 및 관리가 용이하며, 현재 11Mbps의 데이터 전송 속도를 제공한다. 또한 무선 LAN 상에서 이동 호스트는 케이블 없이 어느 장소에서나 고속으로 유선 LAN에 연결하여 인터넷 서비스를 받을 수 있다. 이러한 장점으로 인하여 무선 LAN은 기업 및 공공기관을 중심으로 빠르게 확산되고 있다.

현재 IEEE 802.11[8]이 무선 LAN의 물리 계층과 데이터 링크 계층에 대한 표준을 기술하고 있다. 무선 LAN은 무선 단말기뿐만 아니라 ad hoc 네트워크 혹은 유선 LAN과 연결된 인프라스트럭처(infrastructure) 네트워크로 구성될 수 있으나, 본 논문에서는 그림 1과 같이 무선 LAN과 유선 망이 복합된 인프라스트럭처 네트워크에 초점을 두기로 한다.

인프라스트럭처 네트워크는 BSS(Basic Service Set), ESS(Extended Service Set), AP(Access Point), Portal, DS(Distribution System)로 구성되어 있다[8]. AP는 유선망과 무선망 사이의 브리지로서 무선 호스트를 인터넷과 같은 기존의 유선 LAN에 연결해주며, Portal을 통하여 이동 호스트는 인터넷에 접근할 수 있다. 이때 AP를 중심으로 하나의 셀이 형성되는데 이를 BSS라고 한다. 여러 개의 BSS가 모여서 ESS를 형성하고, DS라는 분산 시스템이 이동 호스트로 전달될 패킷의 포워딩 경로를 결정해준다.

2.1.2 무선 LAN의 데이터 링크 계층

IEEE 802.11이 제안하는 무선 LAN의 MAC 프로토

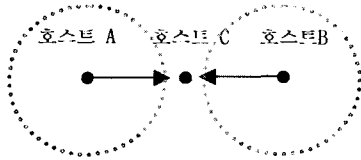


그림 2 숨은 호스트 문제

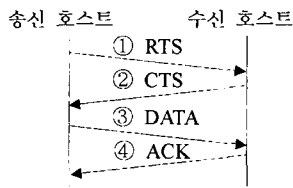


그림 3 CSMA/CA

콜은 기존 802.3 유선 LAN의 표준 기술을 따르나, 이동성 지원을 위한 요구 사항을 반영하고 무선 매체의 고유한 특성 때문에 발생하는 문제점들을 해결하기 위하여 몇 가지 새로운 기술을 도입한다.

유선 망과는 달리 무선 LAN에서 호스트의 위치는 수시로 변화하게 된다. 따라서 현재의 셀을 벗어나 다른 셀로 이동하는 호스트는 통신 링크를 재설정하기 위해서 새로운 AP를 결정해야 하는데, 이러한 과정을 MAC 계층에서의 핸드오프 혹은 로밍이라고 한다. IEEE 802.11은 seamless한 핸드오프를 위하여 scanning, reassociation과 같은 기법을 제공한다[8]. 이동 호스트는 서비스 받고 있는 AP의 시그널 세기가 특정 값보다 낮아지면 새로운 AP를 탐색하고, 가장 큰 시그널을 갖는 AP를 선택하는데 이러한 과정을 scanning이라 한다. 이와 같은 scanning 기법으로 AP가 결정되면, 이동 호스트는 reassociation 과정을 통하여 자신의 존재를 새로운 AP에게 알린다. 이후 AP는 DS에게 이동 호스트의 새로운 위치 정보를 알려주고, DS는 호스트의 위치 정보를 갱신한다.

무선 LAN은 유선 LAN과 동일하게 물리적인 매체를 공유하는 CSMA(Carrier Sense Multiple Access) 방식을 도입한다. 그러나 유선 LAN과는 달리 충돌이 빈번하게 발생할 수 있다. 예를 들면 그림 2에서 C 노드로 패킷을 전송하고자 하는 A 노드는 B 노드가 C 노드로 패킷을 전송하려 한다는 사실을 알 수 없다. 이와 같은 경우 C 노드에서 패킷들의 충돌은 불가피하다[9]. 따라서 IEEE 802.11은 이러한 문제를 해결하기 위하여 충돌 감지 방식이 아닌 충돌 회피 방식을 도입한다[10]

[11]. CSMA/CA(Collision Avoidance)의 동작 원리를 간단히 기술하면 그림 3과 같다. 데이터할 전송할 호스트는 무선 링크가 사용 중인지 아닌지를 먼저 확인한다. 만약 무선 링크가 사용 중이면, 일정 시간 기다린 후 다시 시도한다. 이후 무선 링크가 사용되지 않음이 판단되면 RTS(Request To Send)라는 제어 메시지를 브로드캐스팅 한다(그림 3-①). RTS 메시지에는 수신측 주소와 전송 시간이 적혀있는데, 이를 통하여 무선 링크를 사용하고자 하는 주변 호스트들은 얼마 동안 기다려야 하는지를 예측할 수 있다. 한편, RTS를 받은 호스트는 충돌 위험이 없을 경우에만 CTS(Clear To Send)라는 메시지를 송신 호스트에 전송한다(그림 3-②). CTS를 받은 RTS 메시지의 송신 호스트는 충돌에 대한 걱정 없이 데이터를 보내고(그림 3-③), 데이터에 대한 ACK을 기다린다(그림 3-④). 만약 일정 시간까지 ACK을 받지 못하면, 성공할 때까지 어느 횟수만큼 데이터를 재전송한다. 본 연구에서는 이와 같은 RTS/CTS, DATA/ACK 기법을 본 논문에서 제안하는 패킷 버퍼링 시점 결정에 사용한다.

2.2 이동 IP

2.2.1 이동 IP 구조

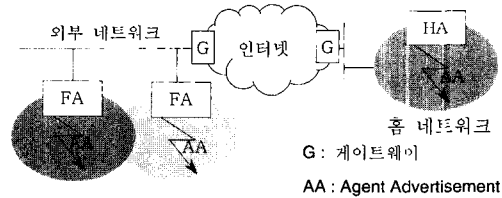


그림 4 이동 IP 구조

IETF 워킹 그룹에서 제안된 이동 IP(RFC 2002)는 IP 주소를 바꾸지 않고 호스트가 이동하면서도 데이터 통신을 할 수 있도록 한다[1]. 이를 위하여 이동 IP는 그림 4에서 보여지는 것처럼 에이전트 개념과, 호스트의 실제 위치를 나타내는 COA(Care of Address)를 사용한다. 그림 4의 HA(Home Agent)는 이동 호스트가 최초로 등록한 홈 네트워크의 라우터이며 MH로 전달되기 위한 모든 패킷은 HA를 통과한다. FA(Foreign Agent)는 이동 호스트가 방문한 네트워크의 라우터로서 이동 호스트에게 COA를 부여한다.

홈 네트워크를 떠나서 새로운 네트워크를 방문한 이동 호스트는 자신의 새로운 위치 정보, 즉, COA를 HA에게 알려야 하는데, 이러한 과정을 이동 IP 등록

(registration)이라고 한다. 이를 통하여 HA는 MH의 현재 위치를 파악할 수 있으며, MH가 외부 네트워크에 있을 때에는 패킷을 캡슐화(encapsulation)하여 FA로 포워딩 하고, FA는 이를 역캡슐화(decapsulation)하여 방문한 MH에게 전달한다.

2.2.2 이동 IP 핸드오프

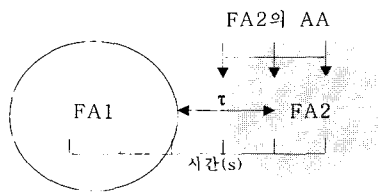


그림 5 중첩되지 않은 셀 영역에서 핸드오프 지연시간

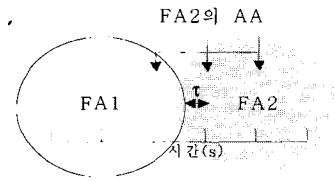


그림 6 중첩된 셀 영역에서 핸드오프 지연시간

FA는 주기적으로 COA를 실은 광고 메시지(Agent Advertisement)를 브로드캐스팅하는데, 이동 호스트는 이 메시지를 이용하여 자신의 위치를 파악할 수 있다. 이동 호스트는 새로운 FA로부터 광고 메시지를 받으면 자신이 외부 네트워크로 이동하였다고 판단하고 2.2.1에서 설명한 이동 IP 등록을 수행한다. 이와 같이 새로운 FA를 선택하여 HA로부터 등록 완료 메시지를 받기까지 일련의 과정을 이동 IP 핸드오프라 한다[1]. 그런데 이동 호스트가 이러한 이동 IP 핸드오프를 수행할 때, 갱신되지 않은 과거의 정보로 인하여 패킷 손실이 일어날 수 있다.

패킷 손실을 그림 5, 6과 같이 셀의 중첩 여부에 따라 살펴보면 다음과 같다. τ 는 이동 호스트가 그림의 FA2로부터 광고 메시지(AA)를 받을 때까지 걸린 시간이다. 그림 5와 같이 이동 호스트가 중첩되지 않은 셀을 이동할 때에는 이동할 네트워크에 대한 어떤 정보도 얻을 수 없다. 이와 같은 경우에는 이동 IP 핸드오프뿐만 무선 LAN의 MAC 계층에서의 핸드오프도 완료되지 않으며, 이동 호스트를 목적지 주소로 하는 패킷은 계속 FA1에게 전달되어 결국은 모두 손실된다. 그림 6과 같이 이동 호스트가 중첩된 영역을 지날 경우, 무선 LAN

의 MAC 계층에서 핸드오프는 완료될 수 있다. 그러나 중첩된 영역을 벗어날 때까지 새로운 FA로부터 광고 메시지를 받지 못하면, 이동 호스트는 자신의 위치 정보를 HA에게 알릴 수 없다. 이러한 경우에도 이동 호스트로 전달될 패킷은 모두 손실된다. 또한 새로운 FA를 통하여 COA를 얻었다 할지라도 HA와의 통신 오버헤드 등의 이유로 이동 IP 등록 과정에 많은 시간이 소요된다면 패킷 손실은 불가피하다.

위의 경우를 종합하여 이동 IP 핸드오프의 수행 중에 패킷 손실을 발생시킬 수 있는 지연 시간을 간단한 수식으로 표현하면 다음과 같다.

이동 IP 핸드오프 지연 시간 = τ + 이동 IP 등록 지연 시간

τ 는 앞에서 설명한 바와 같이 이동 호스트가 새로운 FA로부터 광고 메시지를 받을 때까지 걸린 시간이다.

2.3 관련 연구

2.2.2절에서 설명했듯이 이동 IP 핸드오프를 수행하는 이동 호스트는 HA에서 라우팅 정보가 갱신될 때까지 패킷을 수신할 수 없다. 그런데 IP를 기반으로 하는 전송 프로토콜은 핸드오프 수행으로 인한 패킷 손실에 대한 대안을 가지고 있지 않다. 따라서 핸드오프 수행 중에 손실된 패킷은 TCP 프로토콜의 경우 불필요한 혼잡제어 기법을 실행시키며, 시간 제약적 특성을 갖는 멀티미디어 응용 프로그램의 성능을 현저하게 저하시킨다. 따라서 현재 많은 연구가 이러한 패킷 손실을 줄이거나 제거하기 위하여 프로토콜 계층별로 진행되고 있다.

2.3.1 전송 계층에서의 해결 방안

TCP는 혼잡으로 인한 패킷 손실과 이동 IP 핸드오프 수행으로 인한 패킷 손실을 구별하지 못한다. 따라서 이동 호스트의 이동 IP 핸드오프가 완료되었어도 고정 호스트는 패킷을 바로 전송하지 않고, 기존의 TCP처럼 재전송 시간이 만료되기를 기다린다.

[5]는 의도적으로 고정 호스트에게 TCP의 Fast Retransmission을 수행하게 함으로 TCP의 성능을 향상시킨다. [5]에서 이동 호스트는 이동 IP 핸드오프 수행이 끝나면 곧바로 고정 호스트에게 3개의 중복된 Acknowledgment(ACK)을 보내고, 고정 호스트는 재전송 시간이 만료되기를 기다리지 않고 패킷을 재전송한다. 그러나 이러한 해결책은 패킷 손실을 사전에 방지하지 못할 뿐만 아니라 TCP 특성만을 고려하기 때문에 UDP에는 적용될 수 없다는 단점이 있다.

2.3.2 네트워크 계층에서의 해결 방안

계층적 FA 관리[12]는 빈번한 이동으로 인한 이동 IP 등록 오버헤드를 줄이기 위한 방법이다. 그림 7에서

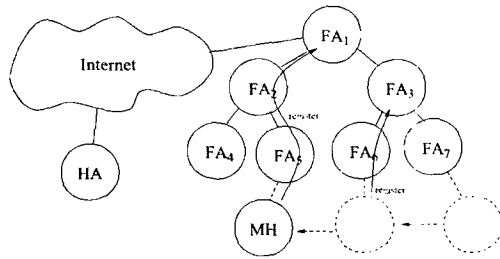


그림 7 계층적 FA 관리

기존 이동 IP를 사용하는 이동 호스트는 FA7에서 벗어나 FA6으로 이동하면 이동 IP 등록 과정을 수행한다. 그리고 HA로부터 등록 완료 메시지가 도착해야 IP 통신을 다시 시작할 수 있다. 그러나 계층적 FA 관리를 사용하는 이동 호스트는 FA3으로부터 라우팅 갱신 완료 메시지를 받으면 통신을 시작한다. 이동 호스트의 위치는 바뀌었지만 F6, F7과는 관계없이 이동 호스트로 전달되는 패킷은 항상 FA3으로 터널링되고 결국 FA3이 이동 호스트로 전달될 패킷을 FA6으로 보낼지 FA7로 보낼지 결정하기 때문이다. 이와 같은 방법은 이동 IP 핸드오프 지연 시간을 단축시킴으로써 성능을 향상시킨다. 하지만 FA의 수가 증가함에 따라 계층적으로 FA를 관리하기 어려울 뿐만 아니라 패킷 손실도 여전히 존재한다.

Smooth 핸드오프[13]는 핸드오프 수행중 패킷 손실을 줄이기 위하여 이동 호스트가 이동 IP 등록 과정을 수행함과 동시에 예전의 FA에게 패킷 포워딩 요청을 보내도록 한다. 예전의 FA는 패킷 포워딩 요청을 받으면 이동 호스트의 위치 정보를 수정하고, 이후 패킷은 모두 캡슐화해서 새로운 FA로 보낸다. 하지만 포워딩 요청이 완료될 때까지 여전히 패킷 손실이 발생한다.

Optimized Smooth 핸드오프[7]는 Smooth 핸드오프의 단점인 패킷 손실을 제거하기 위하여 버퍼링 기법을 사용한다. 현재 FA는 이동 호스트로 전달되는 모든 패킷을 버퍼링하고, 패킷 포워딩 요청이 있을 때 버퍼링된 패킷을 encapsulation 하여 새로운 FA로 보낸다. 이로 인하여 패킷 손실은 완전히 제거될 수 있다. 하지만 무분별하게 패킷을 버퍼링함으로써 버퍼 및 CPU 자원을 낭비할뿐만 아니라 이동 호스트가 받았을지도 모르는 패킷을 재전송함으로써 TCP의 경우 불필요한 혼잡 제어 기법을 수행시켜 성능 저하를 가져올 수 있다. 또한 모든 패킷은 핸드오프 완료 후에 새로운 FA로 터널링되고 이동 호스트에게 전달되므로 긴 패킷 포워딩 시간을 갖게 된다. 따라서 TCP와 시간 제약적 특성이 있는

멀티미디어 응용 프로그램에는 적용되어 실효성을 얻기 힘들다.

2.3.3 데이터 링크 계층을 이용한 해결 방안

현재 많은 연구들이 데이터 링크 계층의 정보를 이용하여 패킷 손실을 줄이려고 한다. 가장 대표적인 예가 Fast 핸드오프[14]이다. Fast 핸드오프에서 이동 호스트는 데이터 링크 계층의 특성을 이용하여 이동 호스트가 이동 가능한 FA를 예측하고, 이에 대한 정보를 HA에게 알린다. HA는 이동 호스트에 대하여 하나 이상의 바인딩 엔트리를 유지한다. HA는 이동 호스트로 패킷을 전달해야 할 경우 등록된 모든 FA에게 패킷을 전달한다. 따라서 이동 호스트는 항상 하나 이상의 FA로부터 패킷을 전달 받음으로써 패킷을 잃어버리지 않고, 또한 패킷 포워딩 지연 시간으로 인한 서비스 왜곡도 경험하지 않는다. 그러나 동일한 패킷이 여러 FA에 전달되기 때문에 네트워크 대역폭이 낭비될 뿐만 아니라, 이동 호스트의 수가 증가하면 HA의 처리 오버헤드가 급격히 증가한다. 또한 중첩되지 않은 셀 구조를 고려하지 않기 때문에 셀이 중첩되지 않았을 때는 현저한 성능 저하가 예상된다.

3. 제안된 핸드오프 프로토콜

본 논문에서 제안하는 핸드오프 프로토콜은 핸드오프 수행 중에 발생할 수 있는 패킷 손실을 제거하고 TCP와 UDP의 성능 향상을 그 목적으로 한다. 이를 위해서 데이터 링크 계층의 정보를 이용할 수 있도록 이동 IP를 확장한다. 본 장에서는 그 구체적인 동작 방식과 확장된 이동 IP가 사용하는 데이터 링크 계층의 정보를 설명한다. 이와 같이 본 논문이 이동 IP의 핸드오프 성능 향상을 위해 사용하는 데이터 링크 계층의 특성을 구체적으로 기술한 것은 큰 장점이다. Fast 핸드오프와 같이 데이터 링크 계층의 특성을 이용해야 하는 기존의 기법들은 데이터 링크 계층의 어떠한 정보를 사용해야 되는지 구체적으로 제시하지 못하고 있다.

3.1 동작 방식

제안된 핸드오프 프로토콜의 동작 방식은 다음과 같다. 이동 호스트는 무선 LAN 기술을 이용하여 핸드오프를 예측하고, 방문할 FA를 결정한 후 현재 FA에게 이와 같은 사실을 알린다(그림 8-(1)). 현재 FA는 이동 호스트가 방문할 FA에게 버퍼링을 요청하고(그림 8-(2)), 이동 호스트가 다른 셀로 이동하였음이 판단되면 이동 호스트로의 패킷 전송을 멈추고 대신에 새로운 FA에게 즉시 패킷을 터널링한다(그림 8-(3)). 새로운 FA는 패킷을 버퍼링한 후, 이동 호스트가 포워딩을 요

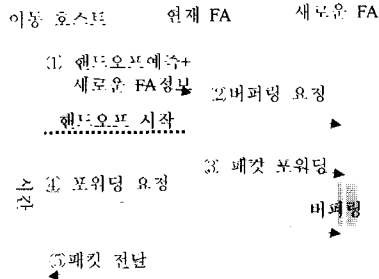


그림 8 중첩된 셀 영역에서 제안된 프로토콜

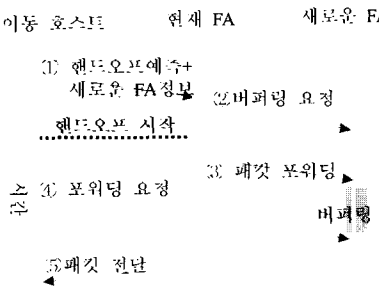


그림 9 중첩되지 않은 셀 영역에서 제안된 프로토콜

청하면 곧바로 패킷을 전달한다(그림 8 (4),(5)).

본 논문에서 제시된 버퍼링 기법은 τ 가 작고 버퍼링을 위한 버퍼의 크기가 클수록 손실되는 패킷의 수는 줄어든다[6]. 그러나 버퍼의 크기가 너무 클 경우에는 중복된 패킷 수신을 유발할 수 있다. 이를 해결하기 위해서 버퍼링의 시점을 결정할 수 있는 기법은 3.3절에서 제시하여 최소한의 버퍼로 중복 수신을 피할 수 있도록 한다.

셀이 중첩되어 있지 않은 경우에는 이동 호스트는 방문할 FA에 대한 정보를 미리 얻을 수 없고, 단지 핸드오프만을 예측할 수 있다. 이와 같은 경우에 이동 호스트는 핸드오프 발생 여부만 현재 FA에게 알리고(그림 9-①), 현재 FA는 이동 호스트가 자신의 셀 영역을 벗어나면 직접 버퍼링한다. 이후 새로운 FA를 만난 이동 호스트는 예전의 FA에게 패킷 포워딩 요청을 보낸다(그림 9 (2)). 이동 호스트로 전달될 패킷을 버퍼링하고 있던 FA는 포워딩 요청을 받은 즉시 새로운 FA에게 버퍼링 된 패킷을 터널링 하고(그림 9-③), 새로운 FA는 포워딩 된 패킷을 이동 호스트에게 전달한다(그림 9-④). 하지만 Fast 핸드오프는 중첩되지 않은 셀의 경우에는 패킷 손실을 막을 수 없다. 이는 중첩되지 않은 경우에는 이동 가능한 FA를 찾을 수 없기 때문이다.

즉, 이동 호스트로부터 핸드오프 예측이 현재 FA에게 전달될 때, 새로운 FA에 대한 정보가 포함되어 있으면 중첩된 셀에 대한 동작 방식(그림 8)을 따라서 새로운 FA에서 버퍼링을 한다. 반면, 새로운 FA에 대한 정보 없이 핸드오프 예측만 왔을 때에는 중첩되지 않은 셀에 대한 동작 방식(그림9)을 따라서 현재 FA에서 버퍼링을 한다.

3.2 데이터 링크 계층 정보를 이용한 핸드오프 예측과 방문할 FA 결정 방법

본 논문에서 제안한 핸드오프 프로토콜은 호스트의 이동성을 정확하게 파악하여 성능 향상을 이끌어낸다. 본 장에서는 핸드오프 발생 여부와 방문할 FA를 결정하는 방법을 설명한다.

2.1에서 설명했듯이 현재 IEEE 802.11은 MAC 계층에서의 핸드오프를 위하여 여러 가지 기법을 제공한다. 이동 호스트는 현재 AP의 SNR(Signal to Noise Ratio)을 주기적으로 모니터링 하다가 SNR의 값이 정해진 값 이하로 떨어지면 새로운 AP를 찾기 위하여 주변 AP에 대하여 scanning을 시작한다. 표1은 Lucent WaveLan에서 사용되는 SNR 지표이다[14]. 이후 가장 큰 SNR을 갖는 AP를 결정하고, 선택된 AP에게 reassociation 서비스를 수행한다. 본 논문은 이와 같은 IEEE 802.11의 scanning 기법 및 reassociation 정보를 이동 IP 계층에 전달함으로써 핸드오프가 발생될지 여부와 이동 호스트가 방문할 FA를 결정한다.

표 1 SNR 지표

	Radio Frequency Quality
STOP	SNR > 50%
START REGULAR	SNR < 40%
START FAST	SNR < 30%

이와 같이 데이터 링크 계층의 정보를 이용함으로써 이동 호스트가 이동할 하나의 FA만을 찾음으로써 Fast 핸드오프와 같이 여러 개의 바인딩을 관리하여 대역폭이 낭비되는 단점이 없다.

3.3 데이터 링크 계층 정보를 이용한 버퍼링 시점 결정 방법

제안된 핸드오프 프로토콜에서 FA는 자원 낭비 없이 패킷 버퍼링을 수행하기 위하여 2.1.3에서 설명한 IEEE 802.11의 CSMA/CA를 이용한다. 무선 LAN에서 데이터를 전송하려고 하는 호스트는 RTS 메시지를 전송한 후 CTS 메시지를 받아야만 데이터를 전송할 수 있다. 또한 전송한 데이터 각각에 대하여 ACK를 받아야만

다음 패킷을 전송할 수 있다. 참고로 인프라스트럭처 네트워크에서 RST/CTS, DATA/ACK는 AP와 이동 호스트간의 패킷 교환이다. 그런데 이동 호스트가 현재 AP 영역을 떠나면 RTS 메시지 혹은 DATA를 받을 수가 없기 때문에, CTS 또는 ACK를 AP에게 전송할 수 없다. 따라서 정해진 횟수만큼 RTS를 보낸 후 CTS 응답이 없거나, DATA를 재전송한 후 ACK가 없으면 이동 호스트가 현재의 셀을 벗어났다고 판단할 수 있다. 따라서 제안된 핸드오프 프로토콜에서 FA는 이동 호스트로부터 핸드오프 예측 메시지를 받은 후, RTS/CTS 혹은 DATA/ACK 교환에 오류가 발생하면 이동 호스트로의 패킷 전송을 멈추고 버퍼링 기법을 시작한다.

반면, Fast 핸드오프는 수행 시기를 결정할 때 데이터 링크 계층의 정보를 이용한다고 언급은 하고 있으나, 구체적인 방법을 기술하고 있지는 못하다. 또한 동일한 패킷을 중복 수신할 가능성이 있기 때문에 이를 방지하기 위한 별도의 기법이 필요하다. 그러나 본 논문에서 제시하는 핸드오프 프로토콜은 이동 노드가 데이터를 받지 못하기 시작한 시점을 찾아냄으로써 그 때부터 버퍼링하여 같은 패킷에 대한 중복 수신을 회피한다.

4. ns-2를 이용한 시뮬레이션 및 성능 분석

본 논문은 제안된 핸드오프 프로토콜의 성능을 검증하기 위하여 캘리포니아 버클리 대학에서 개발된 ns-2 [16]를 사용한다. ns-2는 현재 가장 널리 알려진 시뮬레이터로서 TCP, UDP, IP 뿐만 아니라 이동 IP, IEEE 802.11 등과 같은 많은 네트워크 프로토콜을 지원한다.

4.1 시뮬레이션 환경

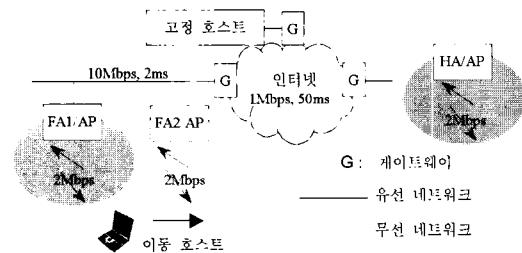


그림 10 시뮬레이션에 사용된 네트워크 구조

표 2 고정 호스트가 전송하는 패킷의 특성

프로토콜	패킷 크기	패킷 전송간격	특성
TCP	1024 바이트	0	일반 데이터
UDP	200 바이트	20 ms	오디오 데이터

그림 10은 시뮬레이션에 사용된 인프라스트럭처 네트워크의 구조 및 링크 대역폭/지연 시간을 보여준다. 이와 같은 환경에서 FA는 1분 간격으로 광고 메시지를 브로드캐스팅하고, 이동 호스트는 FA1과 FA2 사이를 이동하면서 고정 호스트가 전송하는 패킷을 수신한다. 표 2는 고정 호스트가 전송하는 패킷의 종류를 보여준다.

본 장에서는 위와 같은 시뮬레이션 환경에 표준 이동 IP, Smooth 핸드오프 프로토콜이 추가된 이동 IP, 그리고 본 논문에서 제안된 핸드오프 프로토콜이 추가된 이동 IP를 차례로 적용하고, 각각의 경우에 대하여 핸드오프시 TCP와 UDP의 성능을 측정하고 분석한다.

4.2 중첩된 셀 영역에서 TCP 성능 분석

일반적으로 중첩된 셀 영역에서는 핸드오프로 인한 패킷 손실이 발생되지 않을 것으로 생각된다. 그러나 2.2.2절에서 설명했듯이 여러 이유로 중첩된 셀 영역에서도 이동 호스트는 핸드오프를 수행하면서 패킷을 잃어버릴 수 있다.

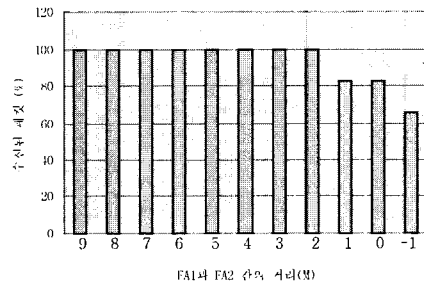


그림 11 중첩된 영역에서 패킷 수신율

그림 11은 표준 이동 IP를 탑재한 이동 호스트가 그림 10과 같은 시뮬레이션 환경에서 FA1의 영역의 중첩된 영역을 지나 FA2의 영역으로 이동할 때, 중첩 정도에 따른 패킷 수신율을 보여준다. FA1과 FA2가 충분히 중첩되었을 때의 수신율과 비교하여 중첩 정도가 1미터 (M) 이하 일 때에는 수신율이 18.2% 저하됐음을 알 수 있다. 이러한 수신율 저하는 2.2.2절의 그림 6처럼 이동 호스트가 중첩된 영역을 벗어날 때까지 FA2를 인지하지 못하여 이동 IP 핸드오프가 완료되기까지 잘못된 라우팅 정보로 인하여 많은 패킷이 손실되었기 때문이다.

그림 12는 2m/s의 속도로 이동하는 호스트가 FA1과 FA2의 중첩 영역(1미터)을 벗어나 0.18분 후에 FA2로부터 광고 메시지를 받을 경우($\tau=0.18$ 분), 이동 호스트의 TCP 패킷 수신 번호를 시간의 흐름에 따라 보여

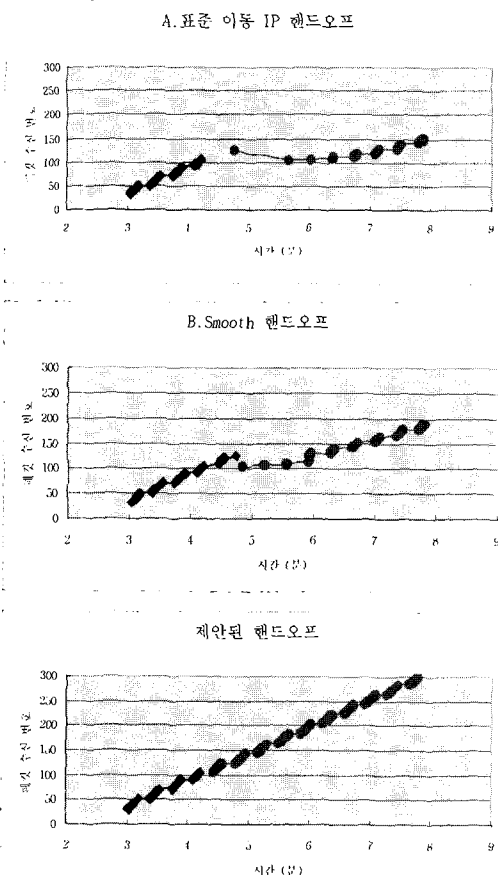


그림 12 중첩된 셀 영역에서 TCP 패킷 수신 번호 ($\tau = 0.18$ 분)

준다. 그림에서 마름모로 표시된 패킷은 FA1으로부터 수신된 패킷이며, 원으로 표시된 패킷은 FA2로부터 수신된 패킷이다.

표준 이동 IP를 사용하는 이동 호스트는 104번째 TCP 패킷을 수신하고 4.2분에 핸드오프를 시작하지만 5.65분에 105번째 패킷을 수신한다. 이러한 1.43분의 시간 공백은 앞에서 설명했듯이 잘못된 라우팅 정보로 인한 이동 호스트로 전달될 패킷의 손실과 고정 호스트의 혼잡제어 기법 수행 때문이다. 이동 호스트는 이미 FA1의 범주를 벗어났지만 핸드오프가 진행되는 동안 HA는 이동 호스트의 새로운 위치 정보를 알지 못하기 때문에 이동 호스트로 전달될 패킷을 FA2가 아닌 FA1으로 터널링하여, 결국 패킷 손실을 야기하는 것이다. 그런데 기존 TCP 프로토콜은 모든 패킷 손실을 혼잡으로 간주

한다. 따라서 고정 호스트는 이동 호스트로부터 일정 시간 내에 ACK을 받지 못하면 혼잡 제어 기법인 exponential backoff를 수행하여 중복된 ACK을 받을 때마다 재전송 타임 아웃의 크기를 두 배로 늘려 결국 1.43분 동안 패킷을 전송하지 않는다. 또한 고정 호스트는 slow start 기법을 통하여 패킷을 재전송하기 때문에 이동 호스트의 TCP 성능은 더욱 악화된다.

Smooth 핸드오프 프로토콜을 사용하는 이동 호스트는 표준 이동 IP를 사용하는 이동 호스트보다 훨씬 성능이 좋다. 이것은 이동 호스트가 FA2로부터 광고 메시지를 받으면 이동 IP 등록과 함께 포워딩 요청을 FA1으로 보내기 때문이다. 그런데 이동 호스트가 FA2로부터 받은 최초의 패킷 번호는 105번째 패킷이 아니라 111번째 패킷이다. 이것은 Smooth 핸드오프 프로토콜에서도 여전히 패킷 손실이 존재함을 의미한다. 이러한 패킷 손실로 인하여 고정 호스트는 표준 이동 IP처럼 혼잡 제어 기법인 exponential backoff와 slow start를 수행하여 이동 호스트의 TCP 성능을 저하시킨다.

본 논문에서 제안된 핸드오프 프로토콜을 적용한 TCP는 패킷 손실을 경험하지 않으며, 표준 이동 IP를 사용하는 TCP와 비교하여 거의 4.1배의 좋은 처리율을 보인다. 더욱이 이동 호스트는 4.23분에 FA2로부터 패킷을 수신하는데, 이는 제안된 핸드오프 프로토콜이 패킷 포워딩 시간을 최소화 하기 위하여 이동 호스트로 전달될 패킷을 FA1이 아닌 FA2에 저장하고 이동 호스트가 포워딩 요청을 해오면 곧바로 포워딩하기 때문이다.

그림 13은 2m/s의 속도로 이동하는 호스트가 F1과 FA2의 중첩 영역(1미터)을 지날 때, 최악의 지연 시간 후에 광고 메시지를 받았을 경우($\tau = 0.48$ 분), 시간 흐름에 따른 TCP 패킷의 수신 번호를 보여준다.

그림 13의 가장 두드러진 특징은 그림 12와는 달리 Smooth 핸드오프 프로토콜이 표준 이동 IP와 동일한 성능을 보인다는 것이다. 이러한 결과는 τ 값이 커지면 이동 호스트가 FA2의 광고 메시지를 받고 FA1에게 패킷 포워딩 요청을 보내는 것이 부의미하다는 것을 의미한다. Smooth 핸드오프 프로토콜을 사용하는 이동 호스트는 중첩된 영역을 벗어나 0.48분 후에 광고 메시지를 받고 곧바로 FA1에게 패킷 포워딩 요청을 보내지만 고정 호스트는 이미 여러 개의 패킷 손실로 인하여 exponential backoff를 수행하고 있다. 즉 고정 호스트가 패킷을 보내지 않기 때문에 중간 경로에 있는 FA1으로 패킷이 전달되지 않아 이동 호스트에게 전송할 패킷이 없는 것이다. 이것은 Smooth 핸드오프 프로토콜의 또 하나의 단점이라 할 수 있다.

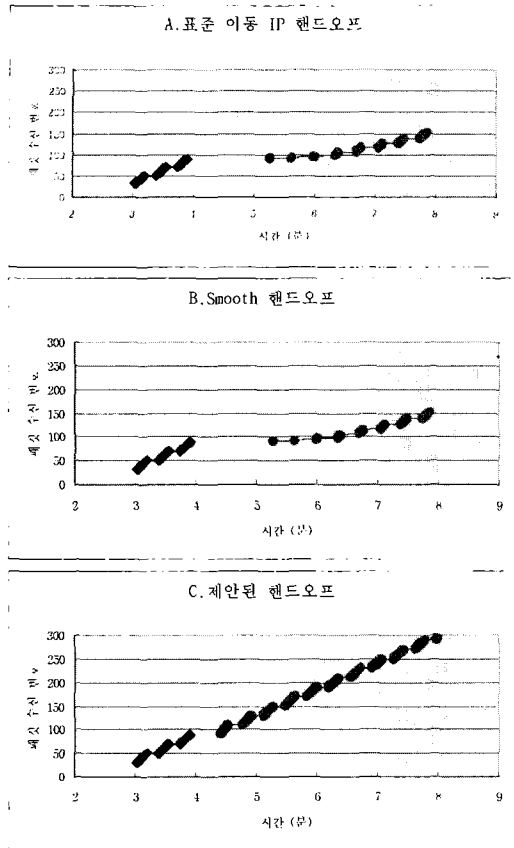


그림 13 중첩된 셀 영역에서 TCP 패킷 수신 번호 ($t=0.48$ 분)

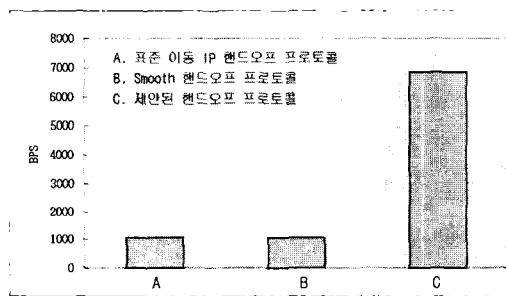


그림 14 핸드오프 수행중 TCP 처리율

반면, 제안된 핸드오프 프로토콜은 τ 값이 커짐에 따라 TCP 타임아웃이 발생하였음에도 불구하고 윈도우의 크기가 바로 복구 됨을 알 수 있다. 제안된 핸드오프 프로토콜 역시 TCP 타임아웃에 의해서 다른 두 프로토콜

과 마찬가지로 윈도우 크기가 1로 줄고, slow start를 수행한다. 하지만 FA2에서 버퍼링한 데이터들이 핸드오프가 완료되자마자 이동 호스트에게 전달되고 이들에 대한 ACK가 발생함에 따라서 윈도우 크기가 바로 회복된다. 그 결과 제안된 핸드오프는 다른 두 핸드오프 프로토콜에 비해서 그림 14와 같이 좋은 성능을 보인다. 그림 14는 핸드오프를 시작하여 표준 이동 IP 및 Smooth 핸드오프 프로토콜을 사용하는 TCP가 혼잡 제어 기법을 완료할 때까지의 TCP 처리율을 보여준다. 그림에서 알 수 있듯이 제안된 핸드오프 프로토콜은 비교 핸드오프 프로토콜보다 6.25배의 우수한 성능을 보여준다.

4.3 중첩된 셀 영역에서 UDP 성능 분석

표 3 중첩된 셀 영역에서 UDP 패킷 수신 번호($\tau=0.18$ 분)

표준 이동 IP 핸드오프		Smooth 핸드오프		제안된 핸드오프	
패킷 번호	수신된 시간	패킷 번호	수신된 시간	패킷 번호	수신된 시간
146	4.137	146	4.137	146	4.137
147	4.157	147	4.157	147	4.157
148	4.177	148	4.177	148	4.177
149	4.197	149	4.197	149	4.197
170	4.617	160	4.421	150	4.395
171	4.637	161	4.441	151	4.397
172	4.657	162	4.461	152	4.399
173	4.677	163	4.481	153	4.401
174	4.697	164	4.501	154	4.403
175	4.717	165	4.521	155	4.405

표 3은 4.1과 같은 시뮬레이션 환경에서 이동 호스트가 수신한 UDP 패킷 번호를 시간의 흐름과 함께 보여준다. 이 경우 표준 이동 IP를 기반으로 하는 이동 호스트는 20개의 UDP 패킷을 잃어버리고, 0.42분 후에 새로운 패킷을 받는다. 이것은 앞 단락에서 설명했듯이 잘못된 라우팅 정보로 인하여 이동 호스트로 전달될 패킷이 모두 FA1으로 전달되기 때문이다.

Smooth 핸드오프 프로토콜을 사용하는 이동 호스트는 등록과 동시에 FA1에게 포워딩 요청을 보내기 때문에 표준 이동 IP 핸드오프 프로토콜 보다 나은 성능을 보여주지만 여전히 10개의 패킷을 잃어버린다.

제안된 핸드오프 프로토콜은 패킷 손실도 없을 뿐만 아니라 FA1을 떠난 후 0.198분 후에 패킷을 받는데, 이것은 Smooth 핸드오프 프로토콜보다 26ms 더 빠르게 패킷을 수신하는 것이다. 이것은 버퍼링 기법을 통하여 패킷 손실을 제거하고, 이동 호스트가 방문할 FA(FA2)에 패킷을 저장함으로써 포워딩 지연 시간을 줄이기 때

문이다. 물론 제안된 핸드오프 프로토콜에도 0.198분의 지연 시간이 존재하여 멀티미디어 데이터의 시간 제약적 특성을 지켜주지 못할 수도 있다. 그러나 일반적인 멀티미디어 응용 프로그램이 사전 버퍼링(prefetching buffering) 기법을 사용하고 있음을 고려하고, 또한 MPEG에서 I 프레임과 같은 중요한 정보를 담은 패킷 손실이 멀티미디어 응용 프로그램의 성능에 치명적임을 감안할 때, 기존 핸드오프 프로토콜에 비하여 제안된 핸드오프 프로토콜은 멀티미디어 데이터의 시간 제약적 특성을 보장해 줄 수 있음을 알 수 있다.

4.4 중첩 되지 않은 셀 영역에서 TCP 성능 분석

이동 호스트가 중첩되지 않은 셀 영역에 있을 때에는 어떤 FA로부터도 패킷을 받을 수 없다. 본 장에서는 4.1절과 같은 시뮬레이션 환경에서 FA1과 FA2를 중첩되지 않게 설정하고, 이동 호스트의 핸드오프 수행으로 인한 TCP의 성능을 분석한다.

그림 15는 FA1에 있던 이동 호스트가 셀 영역을 벗어나 1.18분 후에 FA2로부터 광고 메시지를 받았을 경우($t=1.18$ 분), 시간의 흐름에 따른 TCP 패킷 번호를 보여준다. Smooth 핸드오프 프로토콜은 4.2에서 설명한 이유로 인하여 그림 13처럼 표준 이동 IP와 동일한 성능을 보인다.

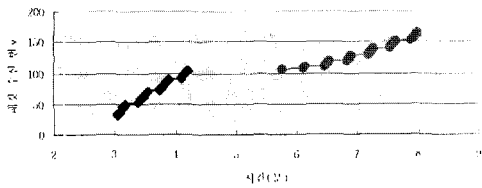
본 논문에서 제안된 핸드오프 프로토콜은 비록 끈김 현상이 있지만 셀이 중첩되어 있지 않아도 여전히 가장 우수한 성능을 보여준다. 제안된 핸드오프 프로토콜이 방문할 FA를 알 수 없을 경우 현재 FA(FA1)가 직접 이동 호스트로 전달된 패킷을 버퍼링하고, 이동 호스트로부터 포워딩 요청을 받으면 곧바로 버퍼링된 패킷을 새로운 FA(FA2)에게 터널링하기 때문이다. 이로 인하여 이동 호스트는 버퍼링된 패킷들을 받고, 곧바로 고정 호스트에게 ACK를 보냄으로써 fast retransmission을 유도하여, 혼잡 제어 기법 수행으로 인한 성능 저하를 최소화로 한다.

5. 결론

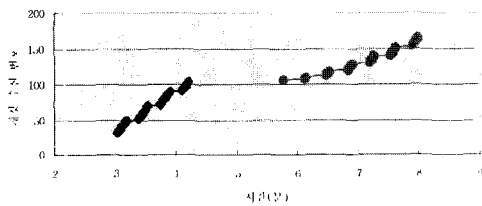
이동 IP의 핸드오프 수행으로 인하여 전송 프로토콜의 성능이 급격히 저하되고 이러한 현상의 가장 큰 원인은 잘못된 라우팅 정보로 인한 패킷 손실임은 잘 알려져 있다. 따라서 현재 많은 연구가 버퍼링 기법을 사용하여 이러한 패킷 손실을 줄이고자 한다. 하지만 무분별하게 패킷을 버퍼링함으로써 시스템 자원 및 네트워크 대역폭 낭비를 초래하며, 중복된 패킷에 대한 부가적인 처리를 요구한다. 또한 버퍼링된 패킷을 이동 호스트에게 전달하기까지 포워딩 시간을 고려하지 않아 시간 제약적 특성이 있는 멀티미디어 응용 프로그램에서는 실효성을 거두기 어렵다.

본 논문에서는 위와 같은 문제를 해결하기 위하여 새로운 이동 IP 핸드오프 프로토콜을 제안하였다. 제안된 핸드오프 프로토콜은 버퍼링 기법을 통하여 패킷 손실을 제거한다. 하지만 데이터 링크 계층의 정보를 이용하여 정확하게 버퍼링 시작 시점을 파악하기 때문에 자원 낭비 및 중복된 패킷 전송에 대한 우려가 존재하지 않는다. 또한 핸드오프 중에 이동 호스트로 전달된 패킷은 현재 FA가 아닌 데이터 링크 계층의 정보를 이용하여 예측된 FA에 버퍼링됨으로써 패킷 포워딩 시간이 최소화되어 기존의 다른 핸드오프 프로토콜에 비하여 응용 프로그램의 시간 제약적 특성을 지켜준다. 더욱이 제안된 핸드오프 프로토콜은 셀 구조를 고려하여 설계됨으로써 셀 중첩 여부에 관계없이 패킷 손실을 제거하여 전송 프로토콜의 성능을 보장한다.

A. 표준 이동 IP 핸드오프



B. Smooth 핸드오프



C. 제안된 핸드오프

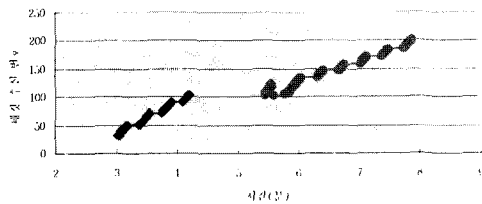


그림 15 중첩되지 않은 셀 영역에서 TCP 패킷 수신번호(t = 1.18분)

Ns-2를 이용한 시뮬레이션 결과 제안된 핸드오프 프로토콜을 적용한 전송 프로토콜은 패킷 손실을 경험하지 않았으며, TCP의 경우 표준 이동 IP를 적용한 TCP에 비하여 최대 6.25배의 우수한 성능을 보여주었다.

참 고 문 헌

- [1] Perkins E. Charles, "IP Mobility Support," RFC 2002, October 1996.
- [2] N. A. Fikouras, K. El Malki, S. R. Cvetkovic, and C. Smythe, "Performance of TCP and UDP during Mobile IP handoffs in single-agent subnetworks," Wireless Communications and Networking Conference, vol. 3, pp. 1258-1262, 1999.
- [3] N. A. Fikouras, K. El Malki, S. R. Cvetkovic, and M. Kraner, "Performance analysis of Mobile IP handoffs," Microwave Conference, Asia Pacific, vol. 3, pp. 770-773, 1999.
- [4] Anne Fladenmuller and Ranil De Silva, "The effect of Mobile IP handoffs on the performance of TCP," Mobile Networks and Applications 4, pp. 131-135, May, 1999.
- [5] R. Cáceres and L. Iftode, "Improving the performance of reliable transport protocols in mobile computing environments," IEEE Journal on Selected Areas in Communications, vol. 13, pp. 850-857, June 1995.
- [6] R. Cáceres, V and N. Padmanabhan, "Fast and Scalable Wireless Handoffs in Support Mobile Internet Audio," ACM Mobile Networks and Applications, vol. 3, no. 4, pp. 351-363, 1998.
- [7] C. E. Perkins and Kuang-Yeh Wang, "Optimized Smooth handoffs in Mobile IP," Proceedings of IEEE International Symposium on Computers and Communications, pp. 340-346, 1999.
- [8] IEEE P802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," 1997.
- [9] Scott Shenker and Lixia Zhang, "MACAW: a media access protocol for wireless LAN's," Vaduvur Bharghavan, Alan Demers Proceedings of the conference on Communications architectures, protocols and applications, pp. 212-225, 1994.
- [10] A. Marincic and D. Milovanovic, "Wireless Local Area Networks," International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, vol. 1, pp. 291-299, 1999.
- [11] Schiller and H. Jochen, "Mobile communications" Addison-Wesley Pub Co (Sd), 2000.
- [12] D. Forsberg, J. T. Malinen, and J. K. Malinen, "Distributing mobility agents hierarchically under frequent location updates," Mobile Multimedia Communications, pp. 159-168, 1999.
- [13] C. Perkins and D. Johnson, "Route Optimization in Mobile IP," draft-ietf-mobileip-optim-09.txt, February 2000.
- [14] Karim El Malki and Hesham Soliman, "Fast Handoffs in Mobile IPv4," draft-elmalki-mobileip-fast-handoffs-02.txt, July 2000.
- [15] <http://www.lucent.com>.
- [16] <http://www.isi.edu/nsnam/ns/>.



박 지 현

1999년 고려대학교 전산학 학사. 2001년 고려대학교 전산학 석사. 2001년 ~ 현재 모토로라 연구원. 관심분야는 무선 네트워크 프로토콜, CDMA.



진 현 옥

1997년 고려대학교 전산학 학사. 1999년 고려대학교 전산학 석사. 1999년 ~ 현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 기가비트 네트워크 프로토콜, 무선 네트워크 프로토콜, 운영체제.



유 혁

1982년 서울대학교 전자공학 학사. 1984년 서울대학교 전자공학 석사. 1986년 University of Michigan 전산학 석사. 1990년 University of Michigan 전산학 박사. 1986 Center for Information Technology Integration 연구원. 1986년 ~ 1988년 CAEN 연구원. 1990년 ~ 1995년 Sun Microsystems, Inc. 연구원. 1995년 ~ 2002년 고려대학교 컴퓨터학과 부교수. 2002년 ~ 현재 고려대학교 컴퓨터학과 교수. 관심분야는 운영체제, 네트워크 프로토콜, 멀티미디어.