

모바일 인터넷에서 이동성 패턴을 이용한 핸드오프 기법

김황래^{1*}

A Hand-off Technique Using Mobility Pattern in Mobile Internet

Hwang-Rae Kim^{1*}

요약 본 논문에서는 Mobile IPv6에서 핸드오프 시에 발생하는 패킷의 손실과 비순서화의 문제를 해결하기 위하여 이동성 패턴을 이용한 핸드오프 기법을 제안하였다. 이동 노드의 이동 범위를 미리 설정하여 그룹을 만들고, 그룹에 버퍼 서버를 두어 이동 중에 서비스를 받게 함으로써 패킷의 손실과 비순서화를 감소시켰다. 안정적인 핸드오프에서 발생하는 비순서화를 IP 레벨에서 방지하고, 상위계층인 TCP에서의 재전송을 최소화하여 성능을 향상하였다. 모의실험 결과 노드의 이동성 패턴을 이용하여 효율적으로 패킷을 전송하였으며, 패킷의 손실이 감소되고, 빠른 핸드오프 동작을 보였다.

Abstract Mobile IPv6 generates the loss of packets and out of sequencing when hand off. In this paper, We propose a improved hand off techniques using the mobility pattern of mobile nodes. As making group by presetting the moving range of mobile nodes, and putting buffer server in the group, the packet loss and out of packet sequence can be reduced. The proposed method prevents the out of packet sequence in IP level which can be happened in the stable state, minimizes the packet re-send in TCP level.

In the simulation, the proposed hand off techniques transmits packets efficiently by using the mobility pattern of mobile nodes.

Key Words : Mobile Internet, Mobile IP, Hand off, Mobility pattern

1. 서론

하드웨어 기술이 발달하여 휴대형 단말기의 보급 및 수요가 폭발적으로 증가함에 따라 기존의 IP(Internet Protocol)를 기반으로 하는 데이터 통신에 무선 이동 통신의 개념을 접목시키려는 시도가 활발하게 진행되고 있다. 이러한 흐름의 대표적인 기반 기술이 Mobile IP이다[1]. IP를 기반으로 하는 무선이동 통신이 널리 보급되기 위해서는 많은 기술적인 문제를 해결해야 한다. 그 중 하나가 이동 노드의 핸드오프(Hand-off) 수행으로 인한 전송 프로토콜의 성능 저하이다.

Mobile IP를 사용하는 호스트는 현재의 서브 네트워크를 벗어나서 새로운 서브 네트워크로 이동할 때 핸드오프를 수행해야 한다. Mobile IP 핸드오프 중에 이동 노드로 전달될 패킷의 라우팅 경로가 올바르게 갱신되지 않

으면 패킷 손실이 발생할 수 있다. 이로써 노드는 핸드오프 수행으로 예기치 못한 성능 저하가 발생 한다[2].

Mobile IP에서 핸드오프 수행으로 인한 패킷 손실 문제를 해결하기 위해서 IETF 워킹그룹을 중심으로 연구가 진행되고 있다. 그 중에서 패킷 버퍼링 기법이 대표적으로 사용된다[3][9]. 그러나 무분별한 패킷 버퍼링은 CPU 및 메모리 자원의 낭비를 초래할 뿐만 아니라, 중복된 패킷을 이동 노드에게 전달함으로써 네트워크 대역폭을 낭비하게 된다.

본 논문에서는 Mobile IPv6에서 이동 노드의 이동성 패턴 값을 정하고 버퍼서버 라우터를 두어 패킷을 버퍼링한 후 이동성 패턴을 이용하여 이동 노드로 패킷을 전달하도록 하였다. 제안한 방법에서는 안정적인 핸드오프에서 발생하는 패킷 비순서화와 패킷의 손실을 IP레벨에서 방지하고 상위 계층인 TCP 계층에서 패킷 재전송을 최소화하여 성능을 향상하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 Mobile IP에서의 기본적인 핸드오프와 Mobile IPv6

¹공주대학교 컴퓨터공학부

*교신저자 : 김황래(plusonc@kongju.ac.kr)

에서의 핸드오프인 안정적인 핸드오프와 빠른 핸드오프(Fast Hand-off)를 설명하고 문제점을 기술한다. 3장에서는 이동성 패킷을 이용한 핸드오프 기법을 제안하였다. 4장에서는 제안한 방법의 성능평가를 기술하고, 마지막으로 5장에서는 결론 및 향후 연구 과제를 기술한다.

2. 관련 연구

Mobile IP에서 성능을 개선하기 위해서는 라우팅의 최적화와 더불어 핸드오프 시에 발생하는 패킷의 손실을 최소화하는 메커니즘이 제공되어야 한다. 인터넷 액세스를 수행하는 노드의 이동으로 인해 인터넷 접속점이 변경되면 네트워크상에서 사용되는 현지주소(COA : Care Of Address)값도 변경된다. 이 경우, 이동 노드는 등록 절차를 통해 이 사실을 자신의 HA(Home Agent)에 알리게 되고, HA는 관련 이동 노드에게 이 사실을 통보하게 된다[2]-[4].

2.1 Mobile IP에서의 핸드오프

Mobile IP에서 이동 노드는 자신의 홈 네트워크를 벗어나 새로운 FA(Foreign Agent)의 네트워크로 진입하게 되면, 에이전트 발견 절차에 의해 네트워크 변경 사실을 인지하게 된다. 이동 노드가 서브 네트워크의 변경 사실을 인지하면 COA를 획득하고, FA를 통해 등록을 시도하게 된다. 등록 요구를 수신한 FA는 이동 노드의 인터넷 접속점이 변경된 사실을 HA에 알려주게 되고, HA는 터널의 종단점이 되는 COA를 새로운 FA의 주소로 변경하게 된다. 따라서 IP 패킷은 새로 등록이 완료된 후에, HA에 의해 새로운 COA로 캡슐화되어 전달되며, 이전의 FA에 저장되는 이동 노드의 바인딩 엔트리 정보는 생명주기가 초과되면 해제된다.[4]-[6]. 이러한 Mobile IP에서의 핸드오프 동작이 그림 1에 나타나 있다.

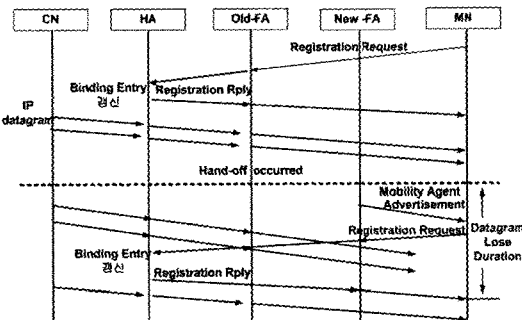


그림 1. Mobile IP에서의 핸드오프 절차

대응노드(CN : Correspondent Node)에서 이동노드(MN : Mobile Node)으로 IP 패킷을 전송할 경우 전송 지연시간은(D_{IP})은 다음과 같이 계산된다.

$$D_{IP} = RT_{CN,HA} + RT_{HA,FA} + RT_{FA,MN} + PT_{HA} + PT_{FA}$$

$RT_{CN,HA}$: CN에서 HA로의 라우팅에 소요되는 시간

$RT_{HA,FA}$: HA에서 FA로 터널을 통해 전송되는 시간

$RT_{FA,MN}$: FA에서 이동 노드로 IP 패킷 전달에 소요되는 시간

PT_{HA} : HA에서 바인딩 엔트리 검사 및 터널링을 위한 캡슐화에 소요되는 처리시간

PT_{FA} : FA에서 패킷의 디캡슐레이션(decapsulation) 및 바인딩 엔트리 검사 등에 소요되는 처리 시간

또한, 핸드오프로 인한 패킷의 손실은 핸드오프 발생 시간부터 이동 노드의 등록 완료시까지 발생하게 된다. 따라서, 데이터 손실이 발생하는 시간 간격(DT_{loss})은 다음과 같다.

$$DT_{loss} = DT_{reg} + DT_{COA}$$

DT_{reg} : 등록 요구를 전송하고 응답을 수신하기까지의 시간 간격

DT_{COA} : 핸드오프 발생시부터 에이전트 통보 메시지를 수신하여 COA를 획득할 때까지의 시간 간격 중에서 작은 값.

2.2 Mobile IPv6에서의 핸드오프 문제점

2.2.1 핸드오프 시 패킷의 지연

노드가 다른 라우터 영역으로 이동할 때 핸드오프 지연이 발생한다. 발생하는 지연 시간에는 핸드오프 시에 이동 노드가 새로운 라우터 영역으로 이동하였음을 인지하게 되기까지의 시간, 새로운 서브네트워크에서 사용할 COA를 얻기까지의 시간, 주변의 라우터를 발견하여 이웃 발견 캐쉬(Neighbor Discovery Cache)를 갱신하는데 걸리는 시간과 주소의 중복을 막기 위한 중복 주소 발견 시간 등이 있다[4][7].

2.2.2 핸드오프 시 패킷 손실

빈번하게 발생하는 핸드오프의 과정에서 CN/HA로부터 오는 패킷의 손실이 발생하게 된다. 그림 2는 패킷손실 예를 보여준다. CN/HA에서 패킷1,2,3,4를 PR (Previous Router)를 통해서 이동 노드에게 전달하고 이동

노드가 새로운 지역으로 핸드오프 한 것을 모르고 CN/HA는 패킷을 PR로 전송하게 된다. 새로운 지역으로 핸드오프 한 이동 노드는 CN/HA로 NR(New Router)를 통해서 바인딩 업데이트를 한다. 그 이후의 패킷인 7, 8은 NR를 통해서 이동 노드에게 패킷을 전달한다. 패킷 5, 6번은 이동 노드의 핸드오프로 인해 손실된다[8].

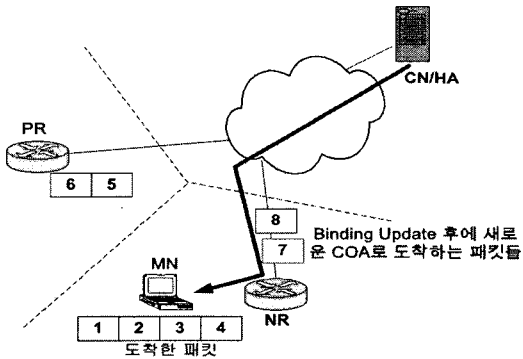


그림 2. 핸드오프 후 이동 노드의 패킷 손실

2.2.3 패킷의 비순서화

이동 노드가 현재 접속한 PR에서 NR로 이동할 때 PR에서 버퍼링을 할 수 있다면 핸드오프 시의 패킷 손실을 줄일 수 있다. 그러나 CN/HA는 패킷이 바인딩 업데이트 후에 NR로 패킷을 보낸다. 이전 패킷이 PR을 거쳐서 오는 동안 바인딩 업데이트 후 보낸 패킷이 먼저 도착하는 상황이 발생할 수 있다. 그림 3에서 보여주는 것처럼 먼저 보낸 패킷 5, 6번이 PR에서 버퍼링된 후에 다시 NR로 전송되는 과정에서 CN/HA이 전송한 패킷 7, 8번이 NR를 통해서 이동 노드에게 먼저 전달되고, 그 이후에 패킷 5, 6번이 이동 노드에게 전달되면 패킷 비순서화가 발생하게 된다[10][11].

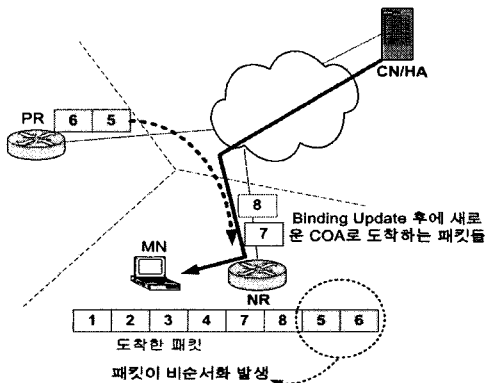


그림 3. 핸드오프 후 수신된 패킷의 비순서화

3. 이동성 패턴을 이용한 핸드오프 기법

3.1 제안한 이동성 패턴을 이용한 핸드오프

본 논문에서는 이동성 패턴을 이용하여 패킷을 안정적으로 전달할 수 있는 방법을 제안하였다. 이동 노드는 자신이 이동한 자료인 이동한 홉 카운터를 가지고 이동성 패턴을 정하며, 이동 노드가 30회 이동한 데이터의 평균이다. 이동성 패턴 값은 이동 노드가 셀을 이동할 때 지나온 셀을 의미한다. 이동성 패턴 값이 $1 \leq X \leq 7$ 인 경우는 이동 노드가 이동성 패턴이 7개의 셀을 이동하면서 패킷을 전달받는 것을 말한다. 이동성 패턴 값이 $7 \leq X \leq 14$ 인 경우는 14개의 셀을 이동하면서 패킷을 전달받는 것, 그리고 이동성 패턴 값이 $14 \leq X \leq 21$ 인 경우는 21개의 셀을 이동하면서 패킷을 전달받는 것을 말한다.

이동 노드의 이동성 패턴 값이 ($1 \leq X \leq 7$)인 경우 Mobile IPv6 기반의 이동 네트워크에서 패킷의 손실과 비순서화를 최소화하고 이동성 패턴을 이용하여 핸드오프 시 이동 노드에게 전달되는 패킷을 안정적으로 보낼 수 있도록 한다. 동작 과정을 보면 그림 4와 같다.

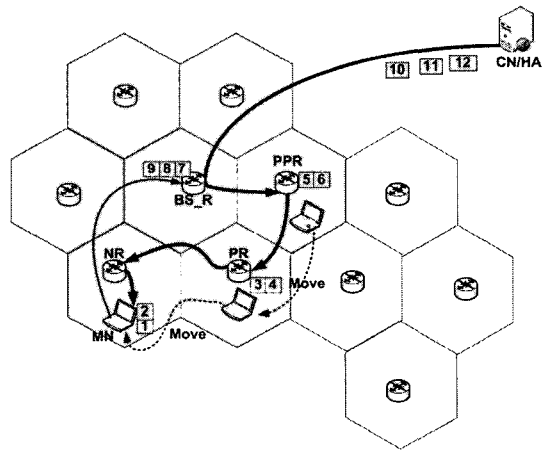


그림 4. 이동성 패턴 값 ($1 \leq X \leq 7$)

이동 노드의 이동성 패턴 값이 ($7 \leq X \leq 14$)인 경우에는 그림 5와 같이 동작한다. 이동성 패턴 값이 변하여 다른 그룹으로 이동할 때 이전 그룹에서 패킷을 전송 받던 이동 노드는 새로운 연결설정을 해야 한다. 노드가 이동하면 이전 그룹의 BS_R에게 CN/HA에서 전송 중인 패킷을 현재 그룹의 BS_R에게 전송하고, 그 패킷을 이동 노드에게 전달한다.

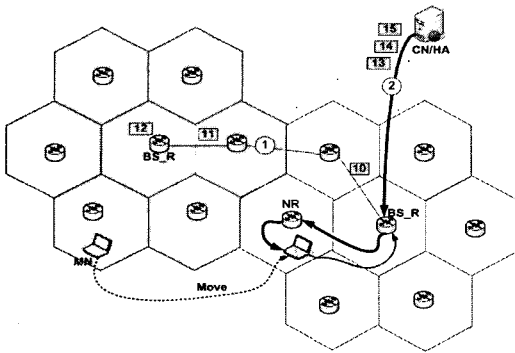


그림 5. 이동성 패턴 값(7≤X≤14)

이동 노드가 새로운 그룹으로 이동할 때 현재 위치한 그룹에서 핸드오프가 발생하는 것을 감지한다. 이후 이동한 그룹에서 바인딩 업데이트 메시지와 현재 갖고 있던 이동성 패턴 값을 수정하여 HA에게 보내고 BS_R에게 할당된 COA를 등록한다.

이동 노드의 이동성 패턴 값이 (14≤X≤21)인 경우는 그림 6과 같이 동작한다. 본 논문에서 제안한 방안에 따라 이동 노드와 BS_R이 동작한다.

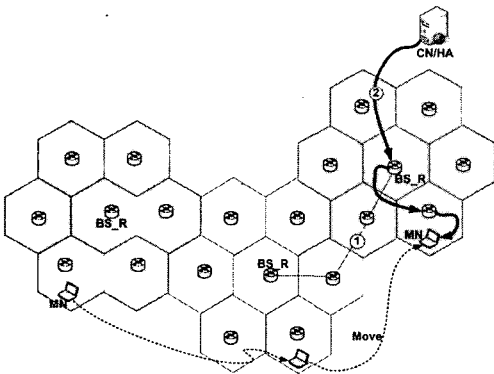


그림 6. 이동성 패턴 값(14≤X≤21)

3.2 이동성 패턴을 이용한 핸드오프 알고리즘

그림 7은 이동 노드가 패킷을 수신할 때의 버퍼 동작을 나타낸다. 전달받을 바인딩 업데이트 메시지를 HA에게 보내면 COA를 이용해 DAD(Duplicated Address Detection) 검사와 이동성 패턴 값 검사를 한다. MN이 위치한 라우터는 BS_R에게 버퍼링을 요구하고 BS_R은 이동 노드의 버퍼를 할당하고 버퍼를 초기화한다. BS_R은 HA/CN에서 패킷을 받아 버퍼에 저장한다. BS_R은 이동 노드가 위치한 라우터에 패킷을 전송하면 라우터는 패킷을 MN에게 전송한다.

```

dad Pattern_check()
while(1) {
    if(getmessage(binding_update){
        if(DAD) then {
            pattern check
            Request (Router, BS_R, Buffering)
            BS_R : Buffer[i]
            Send (HA/CN, BS_R, Packet)
            BS_R : DQ ← Packet
            Send (BS_R, Router, Packet)
            Send (Router, MN, Packet)
        }
    }
}
    
```

그림 7. 이동 노드가 패킷을 전송 받을 때 버퍼의 동작

그림 8은 이동 노드가 패킷전송 중 이동할 때의 버퍼 동작을 나타낸다. 노드가 패킷을 전송 중일 때 다른 라우터 영역으로 이동하면 BS_R은 패킷을 CN/HA로부터 받아 버퍼에 저장하고 바인딩 업데이트 메시지를 보내며, DAD 검사를 한다. 이동 노드가 위치한 현재의 라우터는 BS_R에게 버퍼링을 요구한다. BS_R은 이전 라우터에 패킷을 보내고 이전 라우터는 이동 노드가 위치한 현재의 라우터로 패킷을 보낸다. 현재의 라우터는 패킷을 이동 노드에게 전송한다.

```

move MN_check()
if (Send Packet, move MN){
    Buffer = Send(CN/HA, BS_R, Packet)
    if (SendMsg(binding_update)){
        if(DAD)
            Request(Router, BS_R, Buffering)
            Send (BS_R, Router1, Packet)
            Send (Router1, Router2, Packet)
            Send (Router1, MN, Packet)
        }
    else (not send binding_update)
        Buffer = Send(CN/HA, BS_R, Packet)
        Wait
    }
}
    
```

그림 8. 이동 노드가 패킷전송 중 이동할 때 버퍼의 동작

그림 9는 이동 노드의 그룹 값이 변할 때의 버퍼 동작을 나타낸다. 노드가 다른 그룹으로 이동할 때 바인딩 업데이트 메시지를 HA에 보낸다. HA는 DAD 검사하고, 현재 라우터는 이동한 BS_R에게 버퍼링을 요구한다. BS_R은 버퍼 할당과 초기화를 하고 이전 BS_R은 전송 받던 패킷을 인접한 라우터를 통하여 이동 노드의 현재 BS_R에게 패킷을 전송한다. 현재 BS_R은 이동 노드에게 패킷을 전송한다. 이동 노드가 최악의 이동시 모든 연결설정

을 버리고 이동 노드는 COA를 등록하고, 빠른 핸드오프를 적용한다.

```

move ogroup_check()
if(move other group MN){
    SendMsg (MN, HA, binding_update Msg)
    if(HA, DAD)
        Repeust (N_Router1, BS_R, Buffering)
        BS_R : Buffer[i]
        Send (BS_R1, Router1, Packet)
        Send (Router1, Router2, Packet)
        Send (Router2, BS_R2, Packet)
        Send (BS_R2, N_Router1, Packet)
        Send (N_Router1, MN, Packet)
        Send (HA, BS_R2, Packet)
    }
    else(no group){
        disconnect all
        Fast handoff
    }
}
    
```

그림 9. 이동 노드가 그룹 값이 변할 때 버퍼의 동작

그림 10은 위 모듈들을 종합한 핸드오프 성능평가를 위한 알고리즘이다.

```

if (MN이 최초 연결설정 시 핸드오프 처리)
{
    if(이동성 패턴 값이 0,1,2이때)
        {call dad Pattern_check();}
    else if (이동성 패턴이 없는 MN){
        call MIPv6 hand-off();
        MN이 30회 이상할 때까지 기다림
    }
    else(MN이 30회 이상 움직였을 때){
        Save Pattern value
        call dad Pattern_check();
    }
}
/* 전송중일 때 MN이 이동중일 때 핸드오프 처리*/
if (패킷 전송중 MN이 다른 라우터 영역으로 이동)
{ call move MN_check();}
else(그룹 값이 없는 MN이 다른 라우터 영역으로 이동)
    { call MIPv6 hand-off();}
/* 전송중 다른 그룹으로 MN이 이동시 핸드오프 처리*/
if(패킷 전송중 MN이 다른 그룹으로 이동)
    {call move ogroup_check();}
else if(그룹 값이 없는 MN 버퍼서버 라우터가 있는 영역으로 이동)
    {call MIPv6 hand-off();}
else(MN이 최악의 이동){
    연결 설정을 재설정;
    이동성 패턴에 추가하지 않음;
    call MIPv6 hand-off();
}
    
```

그림 10. 성능 평가를 위한 핸드오프 알고리즘

4. 성능 평가

본 장에서는 제한한 이동성 패턴을 이용한 핸드오프의 성능 평가 및 분석을 하였다. 본 논문에서는 Mobile IPv6 기반의 무선 인터넷 환경을 기반으로 하였으며, CN이 전달하는 패킷 전송 중에 이동 노드가 핸드오프를 할 때 발생하는 패킷의 지연 및 손실과 제한한 방안에서의 패킷 손실 및 지연을 계산하여 분석하였다.

- 우선 시뮬레이션을 위하여 다음과 같이 가정하였다.
- 첫째, 이동 노드와 이동성 에이전트의 네트워크 계층은 IETF에서 제한한 "Mobile IPv6 Support" 의 내용에 따라 동작한다.
 - 둘째, 기존의 핸드오프 기법과 이동성 패턴을 이용한 핸드오프 기법은 패킷을 순차적으로 전송한다.
 - 셋째, 기존의 핸드오프 기법과 이동성 패턴을 이용한 기법에서 노드가 새로운 그룹으로 이동할 때 등록 절차에 수행되는 시간은 동일한 조건하에 동작하므로 패킷 전송시간에서 제외한다.
 - 넷째, 네트워크의 연결은 TCP를 전제로 연결하며, 노드가 이동할 때 패킷의 손실과 비순서화가 일어나고, 핸드오프가 발생한다.

시뮬레이션 환경은 다음과 같다.

- 인텔펜티엄 2.4GHZ,
- 메모리 512MB,
- HDD 40GB,
- OS : Linux kernel 2.4.18-4 (WOW Linux 7.3)
- 시뮬레이션 툴 : NS-2(Network Simulator-2) 버전 NS-2.1b7a
- 사용언어: C/C++과 TCL/TK

이동 노드는 하나로 설정하였고 3가지 패턴 값을 갖고 각각을 그룹으로 설정하여 3개의 그룹으로 나누었다. 이동 노드가 라우터에서 패킷을 전송 받을 때 버퍼 서버 라우터의 버퍼링 시간은 셀을 증가시키면서 측정하였다. 시뮬레이션에 사용된 파라미터들은 표 1과 같다.

표 1. 시뮬레이션에 사용된 파라미터

라우팅 프로토콜	TCP
패킷의 크기	1 Kbyte
이동 노드 수	1개
유선링크 전송률	10 Mbps
무선링크 전송률	5 Mbps
버퍼의 크기	100 Kbyte

4.1 시뮬레이션

이동 노드가 다른 라우터 지역으로의 핸드오프 시에 패킷의 전송을 보여주는 시뮬레이션 과정은 그림 11과 같다.

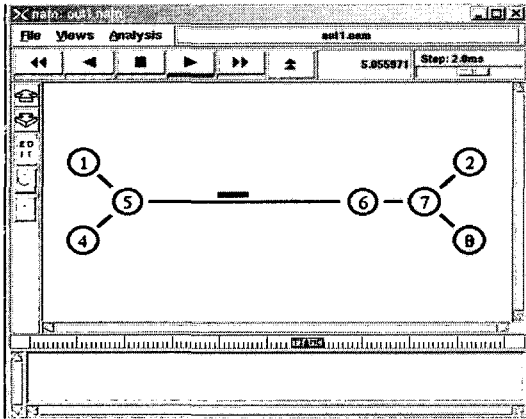


그림 11. 시뮬레이션 동작

그림 12는 시뮬레이션 동안 핸드오프가 7번 발생할 때 노드가 다른 라우터 영역(그룹)으로 이동하는 과정에서 생기는 지연을 측정하였다. 제안한 방법에서는 선행한 주소 중복지연 검사를 통해서 빠른 핸드오프가 가능함을 알 수 있다. 기존의 방법과 제안한 방법에서 차이가 나는 지연은 이동 노드에게 보내는 바인딩 업데이트 시의 지연으로 볼 수 있다.

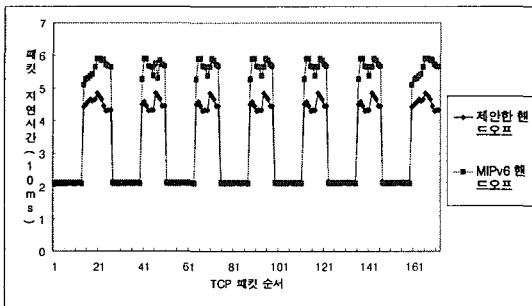


그림 12. 이동 노드의 핸드오프 시 지연

그림 13은 이동 노드가 일정 시간동안 발생하는 핸드오프를 변화시키면서 평균 패킷 지연 시간을 계산한 것이다. 중복 주소 발견 지연을 먼저 선행함으로써 패킷 지연을 줄일 수 있었다.

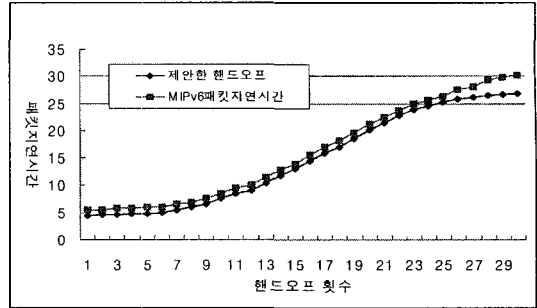


그림 13. 이동노드의 핸드오프 횟수에 따른 패킷지연시간

그림 14는 핸드오프 시에 발생하는 패킷의 손실을 측정하였다. 제안한 방법에서는 버퍼링을 통해서 핸드오프 시 패킷 손실을 감소시킬 수 있었다.

이상과 같이 제안한 핸드오프 방식과 Mobile IPv6의 핸드오프 기법에서의 핸드오프 시의 지연 및 패킷손실을 측정하고 제안한 방안인 이동성 패턴을 이용한 핸드오프 기법이 성능의 향상이 있음을 알 수 있다.

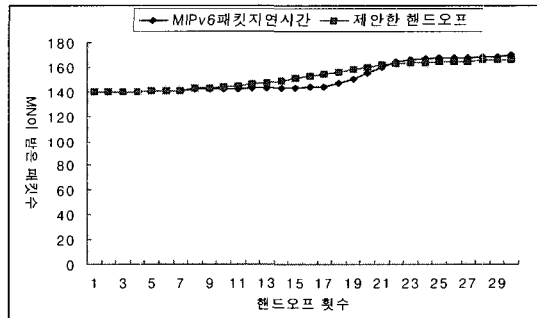


그림 14. 이동노드의 핸드오프 시 패킷의 손실

5. 결론 및 향후 연구 과제

본 논문에서는 Mobile IPv6기반의 이동 네트워크에서 노드의 이동시 핸드오프 성능 향상을 위해서 기존의 핸드오프 기법에 이동성 패턴을 적용한 핸드오프의 기법을 제안하였다. 이동 노드의 이동 범위를 미리 설정하여 패턴 및 그룹을 만들고, 그룹에 버퍼 서버를 두어 이동 중 서비스를 받게 하였다. 이동성 패턴을 이용함으로써 패킷의 손실을 감소시켰으며, 기존의 핸드오프에서 발생하는 비순서화를 IP 레벨에서 방지하고, 상위계층인 TCP에서의 재전송을 최소화하여 성능을 향상하였다.

시뮬레이션을 통하여 노드가 이동할 때 패킷전송 시 패킷 지연 및 비순서화가 감소함을 보여 기존의 핸드오

프 기법보다 성능이 향상되었음을 보였다.

향후 연구 과제로는 제안한 방법을 실제 네트워크에 적용하여 평가하는 것이며, 버퍼 서버를 두어 핸드오프 성능개선에 소요되는 비용과 이동노드의 잦은 이동시의 핸드오프 성능을 개선하는 것이다.

참고문헌

[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] 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.

[4] E. Gustafsson, et al., Mobile IP Regional Tunnel Management, Internet Draft, IETF, Aug. 1999.

[5] Charles E. Perkins, IP encapsulation within IP, RFC 2003, May. 1996.

[6] Charles E. Perkins, Minimal Encapsulation within IP, RFC 2004, Oct. 1996.

[7] IP Routing for Wireless/Mobile Hosts(mobileip) Working Group Homepage, <http://www.ietf.org/html.charters/mobileip-charter.html>

[8] R. El-Malki and H. Soliman, "Fast Handoff in MIPv6," "draft-elmlki-handoff-01.txt, November, 2000.

[9] G. Krishnamurthi, R. Chalmers and C. Perkins, "Buffer Management for smooth Handover in MobileIPv6" draft-krishnamurthi-mobileip6-01.txt, March, 2001.

[10] Rajeev Koodli, Fast Handovers for Mobile IPv6, IETF draft-ietf-mobileip-fast-mip6-06.txt, Mar. 2003.

[11] Greg Daley et al., Movement Detection Optimization in Mobile IPv6, IETF draft-daley-mobileip-movedetect-01.txt, May 2003.

[12] Hesham Soliman et al., Hierarchical Mobile IPv6 Mobility Management, IETF draft-ietf-mobileip-hmip6-07.txt, Oct. 2002.

[13] Thierry Ernst et al., Mobile Networks Support in Mobile IPv6, draft-ernst-mobi-ipv6-network, Mar. 2002.

[14] T. Knivton et al., Mobile Router Tunneling Protocol, draft-knivton-mb-rtr-03.txt, Nov. 2002.

김 황 래(Hwang-Rae Kim)

[정회원]



- 1982년 9월 : 중앙대학교 전자계산학과 (이학사)
- 1991년 2월 : 중앙대학교 컴퓨터공학과 (공학석사)
- 1994년 2월 : 중앙대학교 컴퓨터공학과 (공학박사수료)
- 1983년 3월 ~ 1994년 2월 : 한국전자통신연구원
- 1994년 3월 ~ 현재 : 공주대학교 컴퓨터공학부 교수

<관심분야>

컴퓨터네트워크, 네트워크보안, 무선인터넷, 네트워크 생존성 관리