

# 모바일 환경에서 패킷 리오더링 개선기법에 관한 연구

한병진, 이종혁, 정태명  
성균관대학교 전자전기컴퓨터공학부  
e-mail : {bjhan, jhlee, tmchung}@imtl.skku.ac.kr

## A Study on the Mechanism for Reducing the Packet Reordering Phenomenon in Mobile Environments

Byung-Jin Han, Jong-Hyook Lee, Tai-Myoung Chung  
Electrical and Computer Engineering, Sungkyunkwan University.

### 요 약

패킷 리오더링은 패킷을 보낸 순서와 목적지에 도착한 순서가 다른 현상을 말한다. 모바일 환경에서는 모바일노드의 이동에 따라 패킷이 전달되는 경로가 바뀌기 때문에 빈번하게 패킷리오더링이 발생한다. 본 논문은 FMIPv6 환경에서 모바일노드가 핸드오프 할 때 발생하는 패킷리오더링 현상을 개선하기 위한 기법을 제안한다. 제안하는 기법은 대응노드에 버퍼를 두어 패킷리오더링을 유발하게 되는 패킷을 버퍼링 하였다가 새로운 경로가 설정된 후 새로운 경로를 통해 버퍼링 된 패킷을 전송하여 패킷 리오더링 현상을 개선한다. 또한, 본 논문에서는 수신 측인 모바일노드가 패킷의 끊김을 겪지 않고 패킷 리오더링 현상을 개선하기 위하여 정확한 버퍼링 시점을 계산한다.

### 1. 서론

패킷 리오더링은 패킷을 보낸 순서와 목적지에 도착한 순서가 다른 현상을 말한다. 패킷리오더링은 불필요한 재전송을 야기하고 네트워크의 성능을 저하시킨다. 또한 순서에 민감한 정보의 경우 패킷 리오더링에 의해 정확한 정보를 전달받지 못하게 된다.

한편 휴대형 기기와 무선통신기술의 발달로 이동 중에도 인터넷을 사용할 수 있는 Mobile IP가 등장하였다 [1]. 하지만 Mobile IP에서는 모바일노드의 이동에 따라 패킷의 전달경로가 바뀌기 때문에 패킷 리오더링이 빈번하게 발생하여 성능을 저하시킨다 [2].

모바일 환경에서 패킷 리오더링이 발생하는 경우는 크게 두 가지 경우로 볼 수 있다. 첫 번째 경우는 라우팅최적화 (Routing Optimization) 기법에서 발생한다 [3]. 라우팅최적화는 패킷 전달 시 반드시 홈에이전트를 통해야 하는 삼각라우팅을 줄이기 위하여 등장한 것으로, 라우팅최적화 기법을 통해 경로가 재설정 될 때, 홈에이전트를 통해 전달되던 패킷이 최적화된 경로를 통해 전달되는 패킷보다 늦게 도착하는 패킷 리오더링이 발생한다. 두 번째 경우는 FMIPv6 (Fast Handover for Mobile IPv6) 기법의 핸드오프 과정에서 발생한다 [4]. FMIPv6는 Mobile IP에서 모바일노드가 핸드오프 할 때 새로운 네트워크에서의 주소획득을 위한 작업시간지연으로 인해 핸드오프 지연시간이 길어지는 단점을 극복하기 위해 등장하였다. FMIPv6는

주소획득 작업을 핸드오프 이전에 미리 수행하고, 액세스라우터(Access Router)가 제공해주는 터널링 기법과 버퍼링 기법을 이용하여 핸드오프 지연에 따른 패킷손실을 줄여주는 기능이 있다. 하지만, 모바일노드가 새로운 네트워크로 이동한 후 새로운 경로를 통해 전송 받는 패킷이 액세스라우터에 의해 터널링 되어 전송되던 패킷보다 먼저 도착하는 패킷 리오더링이 발생한다. 본 논문은 FMIPv6 기법에서 발생하는 패킷 리오더링 현상에 초점을 맞추고, 이를 해결하기 위한 대응노드의 버퍼링 기법을 제안한다.

본 논문은 다음과 같이 구성된다. 2 장에서는 관련 연구를 통해 패킷리오더링의 영향과 FMIPv6에서의 패킷리오더링의 문제점에 대해 자세히 살펴보고, 그러한 문제점을 해결하기 위한 해결방안을 3 장에서 제시한다. 4 장에서는 제안하는 기법의 고려사항에 대해 논의하고, 5 장에서 본 논문의 결론을 맺는다.

### 2. 관련연구 및 문제제기

#### 2.1. 패킷 리오더링

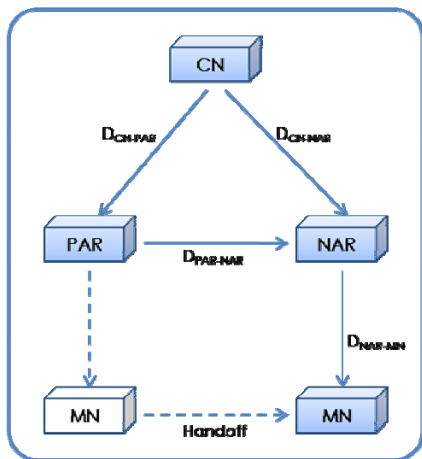
패킷들의 순서가 바뀌어 목적지에 도착하는 패킷 리오더링은 네트워크 환경 및 성능에 큰 영향을 끼치는 요인이다. 화상회의와 같이 동영상정보를 전송하는 응용프로그램에서는 UDP가 주로 사용되는데, UDP 같이 패킷의 순서에 상관하지 않는 전송계층 프로토콜들은 순서가 바뀌어 오는 패킷을 처리할 수 있는 방법이 없다. 따라서 네트워크계층인 IP 단에서 패킷 리오더링이 발생하게 되면 동영상의 끊김이나, 순서의 뒤바뀜이 일어난다.

"본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음"  
(IITA-2007-C1090-0701-0028)

한편 일반적인 목적이나 데이터 전송 등을 위한 응용프로그램에서는 신뢰성을 보장하는 TCP를 주로 사용하는데, TCP는 신뢰성을 보장하기 위해 손실되거나 리오더링된 패킷을 재전송하여 응용프로그램에 정확한 정보를 전달해 준다. TCP에서는 네트워크의 효율성을 높이기 위해 혼잡제어 (Congestion Control)을 수행한다. 이 혼잡제어 메커니즘에서는 3개의 중복응답 (Duplicate Ack)을 받으면 네트워크가 혼잡하여 패킷이 손실되었다고 판단하고 혼잡윈도우 (Congestion Window)를 줄이고, 손실되었다고 판단되는 패킷을 재전송한다. 하지만, 손실된 패킷과 리오더링된 패킷을 구분하기는 쉽지 않다. 즉, 패킷리오더링에 의한 중복응답과 패킷 손실로 인한 중복응답을 구분하기는 쉽지 않다. 따라서 패킷리오더링에 의한 중복응답에 대해 혼잡제어 메커니즘이 동작하여 패킷을 재전송하고 혼잡윈도우를 줄인다. 이처럼 패킷 리오더링은 패킷의 순서가 바뀐 것일 뿐 패킷을 손실한 것이 아님에도 불구하고 패킷을 재전송하고 혼잡윈도우를 줄이게 되어 엄청난 성능상의 손해를 가져온다 [5][6].

2.2. 문제 제기

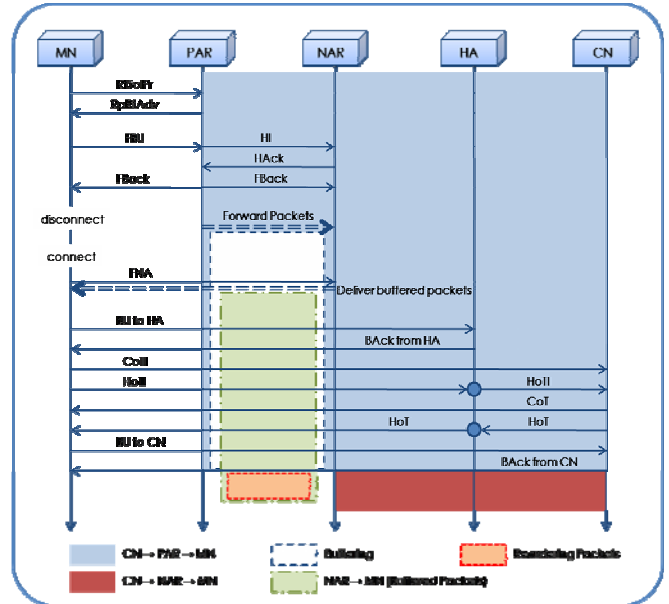
서론에서 언급하였듯이, 모바일 환경에서 패킷 리오더링은 새로운 경로로 도착하는 패킷이 이전경로를 통하여 도착하는 패킷보다 먼저 도착할 때 발생하게 되는데, 그 원인은 (그림 1)에서 보는 것과 같다. (그림 1)은 FMIPv6에서 두 경로의 전송지연의 차 때문에 패킷 리오더링이 발생하는 상황을 보이고 있다.



(그림 1) 패킷 리오더링 발생 이유

대응노드가 모바일노드에게 패킷을 전달하는 도중에 모바일노드가 핸드오프 하게 되면 이전 액세스라우터 (pAR)는 새 액세스라우터 (nAR)에게 터널을 생성하고, 그 터널을 이용해 핸드오프가 완료될 때까지 이전 액세스라우터에게 오는 패킷을 새 액세스라우터에게 터널링 하여 보내준다. 한편, 핸드오프가 완료되면, 대응노드는 새 액세스라우터를 통하여 모바일노드에게 패킷을 전달하게 된다. 편의상 전자를 제 1경로라고 하고 후자를 제 2경로라고 하자. 이때, 제 1경로를 통해 전달되는 패킷은  $D_{CN-PAR} + D_{PAR-MN}$  +

$D_{NAR-MN}$ 의 시간 지연을 겪게 되고, 제 2 경로를 통해 전달되는 패킷은  $D_{CN-NAR} + D_{NAR-MN}$ 의 시간 지연을 겪게 된다.  $D_{CN-PAR}$ 와  $D_{CN-NAR}$ 이 같다고 가정하면, 두 경로로 도착하는 패킷의 지연시간 차이  $D_{PAR-NAR}$ 만큼의 시간 동안 패킷 리오더링이 발생하게 되는데, 이 차이는 다음 (그림 2)의 상세한 FMIPv6 메시지 교환 절차를 살펴보면 더욱 극명하게 드러난다.



(그림 2) FMIPv6 메시지 교환절차

모바일노드가 홈에이전트와 대응노드에게 바인딩업데이트 메시지를 보내기 전부터 이전 액세스라우터는 새 액세스라우터에게 터널을 생성하여 패킷을 버퍼링하고 포워딩 한다. 이 경로로 오는 패킷은 대응노드에 대한 바인딩업데이트가 끝난 다음에도 전달되어 오는 것을 (그림 2)를 통해 확인 할 수 있는데 바로 그 부분에서 패킷 리오더링이 발생한다.

대부분의 패킷 리오더링 개선기법은 TCP 계층에서 작업을 수행한다. 하지만 모바일 환경의 IP 계층 특성 때문에 발생한 패킷 리오더링은 IP 계층에서 처리하는 것이 바람직하다. 즉, Mobile IP 스택 상에서 해결해야 한다. 따라서 본 논문은 앞서 제기한 문제에 초점을 두고 FMIPv6에서 발생하는 패킷 리오더링을 FMIPv6 상에서 해결하는 방안을 3장에서 소개한다.

3. 제안하는 기법

제안하는 패킷 리오더링 해결 기법의 가장 기본적인 아이디어는 다음과 같다. 우선 패킷 리오더링을 해결할 엔터티를 선정한다. 패킷 리오더링을 해결할 엔터티에 버퍼를 생성한다. 버퍼링을 수행할 시점과 지속 시간을 계산한다. 해당 엔터티가 버퍼링된 패킷의 순서를 재조정 하여 보내면 패킷의 수신자는 패킷 리오더링을 겪지 않게 된다. 제안하는 기법을 각각의 단계별로 살펴보면 다음과 같다.

3.1. 엔터티 선정

FMIPv6 에서 모바일노드의 핸드오프 시 발생하는 패킷 리오더링은 (그림 2)를 통해 확인 하였다. 패킷 리오더링을 해결하기 위해서는 전송경로상의 엔터티 중 하나가 패킷의 순서를 바로잡아주는 기능을 가져야 한다. 패킷의 순서를 바로잡아주기 위해서는 패킷들을 임시로 저장할 공간이 필요한데, 여기서 버퍼의 필요성이 생긴다. 두 방향으로 전송되는 패킷을 동시에 받아서 패킷의 순서를 바로잡을 수 있는 엔터티는 모바일노드, 새 액세스라우터, 그리고 대응노드이다. 각각에 대해서 버퍼링을 통한 패킷 순서 바로잡기 기능의 가능 여부를 따져보자.

모바일노드는 양방향으로 전송되는 패킷을 받을 수 있다. 하지만 동시에 패킷의 도착지이기도 하다. 모바일노드의 IP 계층으로 패킷이 들어오는 순간 패킷리오더링이 발생하기 때문에 버퍼를 통해 패킷 리오더링을 수정할 시간적 여유가 없다.

새 액세스라우터는 양방향으로 전송되던 패킷이 합쳐지는 부분으로 버퍼를 통해 패킷 리오더링을 수정할 수 있다. 하지만 버퍼링을 시작하는 순간을 포착하기 위해 패킷의 내용을 확인하는 작업이 동반되어야 하는데 이는 대응노드와 모바일노드간에 전송되는 메시지를 각종 보안위협에 노출 시킬 수 있어 패킷 리오더링을 바로잡는 곳으로 적합하지 않다.

마지막으로 대응노드는 패킷을 전송하는 곳이기 때문에 양방향의 패킷을 모두 확인 할 수 있고 모바일노드와 통신하는 주체이기 때문에 보안위협과 상관없이 패킷의 내용을 확인할 수 있다. 따라서 버퍼를 이용해 패킷 리오더링을 바로잡을 수 있는 엔터티로는 대응노드가 적합하다.

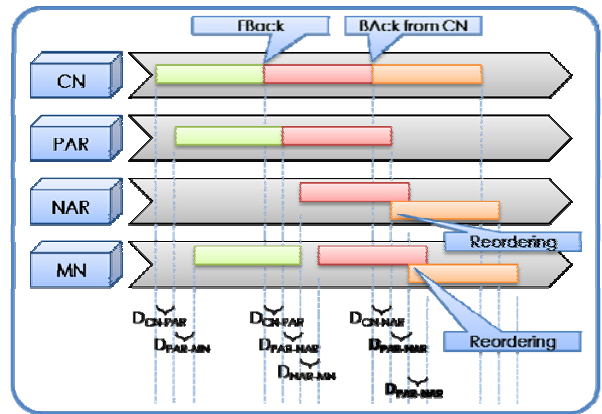
### 3.2. 버퍼링 시점의 계산

패킷 리오더링을 바로잡을 엔터티로 대응노드를 선정했다. 이 절에서는 대응노드에서 버퍼링을 시작할 시점을 계산해 낸다. 버퍼링을 시작할 시점과 지속할 시간은 (그림 1)에서 보았던  $D_{PAR-NAR}$  과 대응노드가 모바일노드의 바인딩 업데이트를 끝내는 시점 (대응노드가 모바일노드에게 BACk 을 보내는 시점)에 밀접한 연관을 가진다.

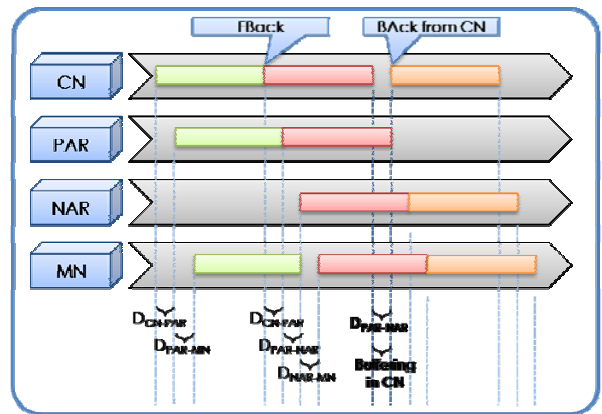
(그림 2)에서 보였듯이 패킷 리오더링은  $D_{PAR-NAR}$  만큼의 시간지연 때문에 발생한다. 패킷 리오더링을 막기 위해, 대응노드는 전송경로가 바뀌기  $D_{PAR-NAR}$  시간 전부터 패킷을 버퍼링 하여 전송경로가 바뀐 뒤에 제 2 경로로 보내게 한다. 제 2 경로로 보낸 패킷은 제 1 경로보다  $D_{PAR-NAR}$  시간만큼 빨리 도착하므로 수신 측에서는 패킷의 끊김 없이 수신할 수 있다. (그림 3)과 (그림 4)는 위 과정을 설명하고 있다.

자세히 설명하자면, 바인딩 업데이트가 끝나기 전, 즉 경로가 바뀌기 전 마지막  $D_{PAR-NAR}$  시간부터 모바일노드에게 전송할 패킷을 버퍼링 한다 (Mobility Header 를 포함한 패킷은 버퍼링에서 제외). 그 후, 대응노드는 바인딩 업데이트가 끝나 모바일노드에게 BACk 을 보내는 순간부터, 즉 경로가 바뀐 다음부터

버퍼에 들어있는 패킷을 제 2 경로로 전송한다. 버퍼가 완전히 비기 전까지 모바일노드로 보내는 패킷은 계속 버퍼링 하고, 버퍼가 모두 비는 순간 버퍼링을 중지한다. 버퍼에는 전송해야 할 패킷이 순서대로 들어있으므로 순서대로 제 2 경로를 통해 전송하면 된다.



(그림 3) 패킷 리오더링이 발생하는 상황

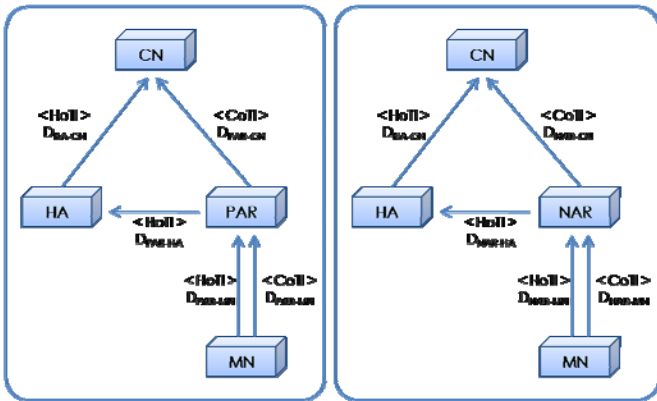


(그림 4) 버퍼링을 통한 리오더링 예방

이와 같이 동작하기 위해 대응노드는 두 가지 시간 정보를 알아야 한다. 하나는  $D_{PAR-NAR}$  이고 다른 하나는 바인딩업데이트가 끝나 전송경로가 바뀌는 시점이다. 먼저 후자를 구하도록 하자. 전송경로가 바뀌는 시점은 대응노드가 바인딩 업데이트를 수행하는데 걸리는 시간  $D_{BU}$  를 기록하게 함으로써 쉽게 구할 수 있다. 대응노드가 바인딩업데이트 메시지를 받은 시간에 바인딩업데이트 수행시간  $D_{BU}$  를 더한 시간이 바로 전송경로가 바뀌는 시점이다.

다음으로 대응노드가  $D_{PAR-NAR}$  을 알아내는 방법을 살펴보자. 원래  $D_{PAR-NAR}$  은 이전 액세스라우터와 새 액세스라우터간의 지연시간이다. 따라서 대응노드는 그 시간을 알 수 없다. 하지만 다음과 같은 절차를 통해 계산할 수 있다. (그림 5)는 이전 액세스라우터와 새 액세스라우터에서 바인딩업데이트를 수행하는 과정의 일부다. 이전과 같이  $D_{CN-PAR}$  과  $D_{CN-NAR}$  이 같다고 가정하면,  $D_{HA-PAR}$  과  $D_{HA-NAR}$  은 HoTI 가 도착한 시간과 CoTI 가 도착한 시간의 차로 구할 수 있다. 이때,  $D_{PAR-NAR}$  은  $D_{HA-PAR}$  과  $D_{HA-NAR}$  의 차에 근사 한다. 따라서 대응노드는  $D_{PAR-NAR}$  를 다음 (식 1)과 같이 구

할 수 있다.



(그림 5) PAR 과 NAR 에서의 바인딩업데이트 과정

$$D_{PAR-NAR} \approx |D_{HA-PAR} - D_{HA-NAR}| \quad (식 1)$$

대응노드는  $D_{HA-PAR}$  를 얻기 위해서 바인딩 업데이트를 수행할 때마다 HoTI 와 CoTI 메시지 수신시간의 차이를 저장해 두어야 한다.

### 3.3. 동작과정

이상을 통해 버퍼링을 수행할 엔터티를 선정하고 버퍼링을 시작할 시점을 계산하였다. 즉, 대응노드는 바인딩 업데이트에 걸리는 시간  $D_{BU}$  와 이전 액세스 라우터에서 HoTI 와 CoTI 메시지 수신시간의 차이를 알고 있다. 이를 기반으로 패킷 리오더링을 위한 버퍼링 기법의 동작과정을 설명한다.

모바일노드가 핸드오프를 위해 HoTI 와 CoTI 메시지를 보내면 (식 1)을 통해  $D_{PAR-NAR}$  을 구한다. 이후 대응노드가 바인딩 업데이트 메시지를 받으면 바인딩 업데이트가 끝나는 시간을 계산해 낸다. 대응노드는 이 두 가지 정보를 이용해 바인딩업데이트가 끝나는 시간으로부터  $D_{PAR-NAR}$  시간 전부터 모바일노드에게 보내는 패킷을 버퍼링 하기 시작한다. 이후 바인딩업데이트가 끝나면 제 2 경로를 통해 버퍼링 된 패킷을 순서대로 전송한다. 버퍼가 완전히 빌 때까지 모바일노드로 보내는 패킷은 버퍼를 통해서 전송된다.

### 4. 토론

본 논문은 대응노드의 버퍼를 이용하여 FMIPv6 환경에서 모바일노드의 핸드오프로 인한 패킷 리오더링 현상을 막는 기법을 제안하였다. 버퍼를 사용하기 위해 이전 액세스 라우터와 새 액세스 라우터간의 지연시간을 계산하고 바인딩업데이트가 끝나는 시간도 계산했다. 하지만 보다 원활한 동작을 위해 고려해야 할 점이 몇 가지 있다. 첫 번째는  $D_{BU}$  와  $D_{PAR-NAR}$  의 상관관계에 대한 논의고, 두 번째는 버퍼의 크기에 대한 논의다.

첫 번째 논의사항은 대응노드의 버퍼링시점은  $D_{BU}$  와  $D_{PAR-NAR}$  의 상관관계에 따라 달라져야 한다는 것이다.  $D_{BU}$  와  $D_{PAR-NAR}$  는 네트워크의 상황과 대응노드

의 컴퓨팅 파워 그리고 기타 여러 요인에 의해 달라질 수 있다. 만일  $D_{BU}$  가  $D_{PAR-NAR}$  보다 크거나 같다면 대응노드의 버퍼링 시점은 앞서 살펴본 것처럼 바인딩업데이트가 끝나기  $D_{PAR-NAR}$  시간 전이 된다. 하지만  $D_{BU}$  가  $D_{PAR-NAR}$  보다 작으면 대응노드는 바인딩업데이트 메시지가 도착한 순간부터 버퍼링을 시작해야 한다. 게다가  $D_{BU}$  가  $D_{PAR-NAR}$  보다 작으면 제안한 기법의 개선효과가 줄어든다.

두 번째 논의사항은 대응노드에 도입할 버퍼의 크기에 대한 고려사항이다. 제안한 기법에서는 바인딩업데이트가 끝나면 버퍼링을 계속하면서 버퍼에 있는 패킷을 순서대로 전송한다. 만일 버퍼의 크기가 충분하지 않다면 버퍼에 저장되지 못하고 버려지는 패킷이 생기게 된다. 하지만 버퍼에 저장되는 패킷의 양은 패킷이 버퍼에 도착하는 속도와 패킷의 크기 그리고  $D_{PAR-NAR}$  의 크기에 따라 달라지기 때문에 이에 대한 정확한 분석과 예측이 필요하다.

### 5. 결론

본 논문은 FMIPv6 기반의 모바일 환경에서 모바일노드의 이동에 따라 패킷이 전달되는 경로가 바뀌어 패킷 리오더링이 빈번하게 발생하는 현상을 개선하기 위한 기법을 제안했다. 제안하는 기법은 모바일노드와 통신하는 대응노드에 버퍼를 두어 패킷 리오더링을 유발하는 패킷을 버퍼링 한 후, 새로운 경로가 설정 되면 새로운 경로로 전송하여 패킷 리오더링을 줄인다. 또한, 본 논문은 정확한 버퍼링 시점을 계산하기 위해 이전 액세스 라우터와 새 액세스 라우터간 지연시간과 바인딩 업데이트 지연시간을 구했다. 또한 이 두 지연시간의 상관관계에 따라 버퍼링 시점이 달라짐을 밝혔다.

추후 연구 과제로 두 지연시간의 상관관계를 상세히 밝히고 대응노드가 가져야 할 버퍼의 크기를 명확하게 계산해 내는 작업을 수행할 것이다.

### 참고문헌

- [1] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.
- [2] D. S. Eom, M. Sugano, M. Murata, and H. Miyahara, "Performance Improvement by packet buffering in mobile IP based networks", IEICE Trans. Communication, Vol. E83-B, No. 11, pp. 2501-2512, Nov. 2000.
- [3] J. Arkko, C. Vogt, and W. Haddad, "Enhanced Route Optimization for Mobile IPv6", RFC 4866, May 2007.
- [4] R. Koodli, "Fast Handovers for Mobile IPv6", RFC 4068, Jul. 2005.
- [5] K.-C. Leung, O. K. Li and D. Yang, "An Overview of Packet Reordering in Transmission Control Protocol (TCP): Problems, Solutions, and Challenges," IEEE Trans. on Parallel and Distributed Systems, Vol. 18, No. 4, pp. 552-535, Apr. 2007.
- [6] S. Bohacek, J. P. Hespanha, J. Lee, C. Lim, and K. Obraczka, "A New TCP for Persistent Packet Reordering", IEEE/ACM Trans. on Networking, Vol. 14, No. 2, pp. 369-382, Apr. 2006.