무선 통신망에서 TCP/IP의 성능 향상을 위한 효율적인 Handoff 알고리즘

조 민 희○ 조 기 호 윤 현 수
한국과학기술원 전산학과

A Novel Handoff Algorithm for TCP/IP in Wireless Network

Minhee CHo○ Ki-Ho Cho Hyunsoo Yoon
Dept. of Computer Science, KAIST
e-mail : {mhcho,hyoon}@calab.kaist.ac.kr

요 약

무선망에서는 높은 에러율과 핸드오프로 인해 패킷 손실이 자주 발생한다. 그러나, 기존의 TCP/IP 프로토콜에서는 패킷 손실을 혼잡에 의해서 발생한 것이라 보고, 패킷 송신율을 줄임으로써 혼잡제어를 실시한다. 따라서 기존의 TCP/IP 프로토콜을 무선망에 그대로 적용했을 경우 매우 낮은 성능을 나타낸다. 본 연구에서는 무선망에서의 국부적인 재전송을 통하여 패킷 손실을 보완하고, 사용자가 셀 간을 이동할 때 핸드오프 지연시간을 줄일 수 있는 새로운 핸드오프 방식을 제안한다. 시뮬레이션을 통해서 제안된 방법의 성능을 평가한다.

1. 서론

최근 mobile computing과 wireless network에 관한 연구가 활발히 이루어지고 있으며 가까운 미래에 wireless network과 mobile computer들이 internetwork의 핵심 부분이 될 것으로 예측된다. wireless link를 이용한 통신은 제한된 대역, 큰 지연 시간, 높은 bit-error율과 빈번한 disconnetion으로 특징지어진다. 또한 사용자들이 wireless network의 셀들을 이동함에 따라 handoff가 발생할 수 있다. 따라서 wireless network과 protocol은 사용자가 요구하는 통신 품질 (QoS)을 보장해 줄 수 있도록 이러한 특성에 잘 대처해야만 한다. 특히 handoff는 하나의 base station (wired와 wireless network사이의 router)에서 다른 base station으로의 통신 상태의 전이를 수반하기 때문에, 그 지연시간으로 인해 종종 packet loss와 지연시간의 변이를 낳는다. 대부분의 network에서 handoff는 수백 ms에서 수 초 정도 지속된다.

TCP는 유선과 고정적인 호스트를 기반으로 한 network에 맞추어진 신뢰성 있는 transport protocol이다. 유선과 고정적인 호스트를 가진 망은 bit-error율이 아주 작기 때문에 TCP는 packet loss를 혼잡 (congestion)에 의해 일어난 것으로 가정한다. 즉, TCP는 평균적인 왕복 시간과 시간 편차를 계속 유지하여 평균보다 시간 편차가 4배 이상 커져도 packet의 acknowledgement를 받지 못하면 혼잡으로 가정하여 재전송을 한다. 이러한 전략은 기존 망에서는 좋은 성능을 보였다.

그러나 wireless network에서는 wireless link의 높은 bit-error율과 handoff로 인한 packet loss가 자주 발생할 수 있다. 이러한 상황에서도 기존의 TCP는 packet loss를 기존 망에서와 같은 방식으로 window 크기를 절반으로 줄이고 slow start와 같은 혼잡 제어를 수행하게 된다. 이에 따라 link의 효율은 크게 감소하고 결국 network의 처리율과 지연시간이 크게 감소한다.

I-TCP [1]는 이러한 문제점을 해결하기 위해 fixed host와 mobile host 간의 연결을 2개의 연결로 분리하여 fixed host와 base station 간의 연결은 기존의 망을 그대로 이용하므로 TCP를 그대로 이용하고 base station과 mobile host 사이의 wireless link는 이에 적합한 transport protocol을 적용하는 방식이다. 이 연구에서는 handoff는 고려하지 않았다. Balakrishnan 등의 연구 [2]는 base station에 아직 ACK를 받지 못한 TCP segment들을 caching하고서 base station과 mobile host 사이에서 packet loss를 검출하여 다시 재전송하도록 한 것이다. 이 방법은 mobile host에서도 TCP를 사용하므로 end-to-end semantics를 유지할 수 있다는 장점이 있다. 또한 handoff 시의 성능 저하를 막기 위해 IP-multicast를 응용한 multicast handoff를 제안하였다.

본 논문에서는 multicast handoff를 개선하여 handoff 지연시간을 줄일 수 있는 새로운 handoff scheme을 제안한다. 제안된 handoff scheme은 현재 연결이 설정된 base station에서 handoff할 가능성이 있는 basestation으로 ACK를 받지 못한 packet들을 forwarding하는 것이다. 이에따라 handoff 시에 packet과 basestation의 상태 변수를 forwarding하는 데 걸리는 시간을 줄일 수 있다. 또, hint를 사용할 수 없을 때도 제한된 크기의 forwarding buffer를 handoff할 가능성이 있는 base station에서 유지할 수 있다. hint를 사용할 수 있는 환경에서는 3단계의 hint를 사용함으로써 부적절한 handoff의 발생과 지연시간을 줄이도록 한다.

본 논문에서는 2장에서 관련 연구를 소개하고 3장에서 forwarding handoff 알고리즘을 제안한다. 4장에서는 simulation을 통해 multicast handoff scheme과 제안한 forwarding handoff scheme을 비교분석하고 5장에서 결론을 맺는다.

2. 관련 연구

wireless network에서 TCP의 성능이 크게 떨어지는 것으로 밝혀짐에 따라 이를 개선하려는 연구가 이루어지고 있다. 대표적인 연구로서 I-TCP scheme [1]과 Snoop module scheme [2]을 소개한다. 또, 빈번한 handoff가 통신 연결의 품질을 크게 떨어뜨리기 때문에 통신 품질에 영향을 주지 않는 handoff 알고리즘에 대한 연구가 이루어지고 있다. 본 장에서는 성능이 가장 우수한 multicast handoff 알고리즘을 소개한다.

2.1 I-TCP

Indirect TCP protocol [1]은 fixed host와 mobile host 사이의 통신 연결을 fixed host와 base station 사이의 TCP 연결과 base station과 mobile host 사이의 연결로 분리하는 방식이다. 두번째 연결의 protocol은 wireless network에 적합한 새로운 protocol이 될 수 있을 것이다. handoff 동안에는 기존의 base station에서 새로운 basestation으로 buffer를 forwarding한다. 이 방식의 장점은 wired link 와 wireless link 사이의 flow control과 congestion control을 분리함으로써 좋은 성능을 얻을 수 있다는 것이다. 그러나 다음과 같은 단점을 갖는다.

- end-to-end semantics의 결여 : 통신 연결이 두 개로 분리됨에 따라 I-TCP의 ack는 실제로 mobile host가 packet을 받았음을 보장하지 못한다.
- Application relinking : mobile host에서 새로운 protocol이 사용됨에 따라 mobile host의 application들은 새로운 protocol과 다시 link되어야 한다.
- Software overhead : packet들은 base station에서 TCP protocol stack과 새로운 protocol의 stack을 거쳐야만 한다. 이것은 packet 지연시간을 늘리게 된다.
- handoff latency : handoff 시에 base station에 있는 buffer의 내용을 새로운 base station으로 forwarding 해야 한다. 이것은 buffer의 크기에 따라 handoff latency가 길어질 수 있음을 가리킨다.

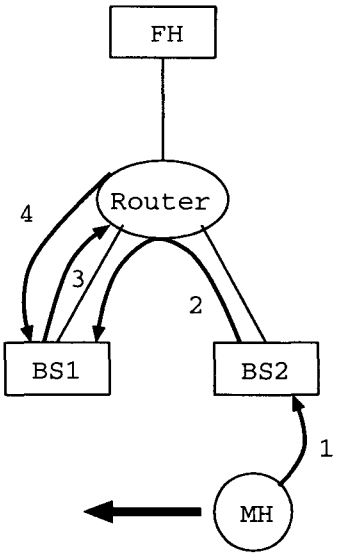

그림 1: Multicast Handoff

2.2 Snoop Module

Snoop Module [2]은 base station에서 mobile host로 향하는 packet들을 caching한다. Snoop Module은 자체내의 timer를 따로 유지하여 wireless link에서의 error로 인해 loss된 packet들을 재전송함으써 fixed host에서 timeout이 발생하는 것을 예방한다. 즉, wireless link의 높은 error로 인한 packet loss 때문에 TCP의 congestion control scheme이 작동하는 것을 막아 wireless network에서도 TCP의 성능이 떨어지지 않도록 한다. handoff 시에는 IP-multicast protocol을 이용하여 handoff hint를 받아 미리 Snoop Module의 내용을 forwarding하고 multicast join을 하도록 하여 최대한 지연시간을 줄이도록 하고 있다. 이 방식은 TCP connection의 end-to-end semantics를 그대로 유지하면서도 높은 성능을 얻는 장점을 갖는다.

2.3 Multicast Handoff

Video conference와 같은 분산 프로그램을 지원하기 위해서 대다수의 망들은 동적으로 구성원이 바뀔 수 있는 multicast connection을 지원한다. rerouting과 같은 기능들은 multicast와 같은 기능위에 몇개의 layer를 이용하여 지원될 수 있다. 따라서 handoff 동안에도 fixed host의 데이타들이 새 base station과 기존 base station으로 multicast될 수 있으며, handoff가 끝나면 기존의 base station은 multicast 연결로부터 제거될 수 있다. 이러한 multicast handoff는 기존의 handoff 알고리즘에 비해서 적은 latency와 buffering을 필요로 하는 장점을 갖는다. 또한 hint를 효과적으로 사용할 수 있도록 한

다.

그림 1은 multicast hanodff의 과정을 보여준다. 1은 mobile host가 handoff할 것임을 basestation에 hint를 주는 것이고, 2는 basestation이 새 basestation으로 packet을 forwarding하는 것이며, 3은 forwarding packet을 받은 새 basestation이 router에 multicast join을 신청하는 것이고 4는 그에 대한 acknowledgement이다.

Multicast handoff의 단점은 잘못된 hint의 사용으로 불필요한 handoff를 일으킬 수 있다는 것이다. 즉 사용자가 cell 간의 경계선에서 zigzag 움직임을 취할 때 실제로 handoff는 일어나지 않았는데도 forwarding과 multicast join이 발생할 수 있다. 또한 base station에서 buffer의 양이 클 경우 handoff의 지연시간은 rerouting보다 packet forwarding에 의존할 수 있으나 forwarding 시간에 대한 고려가 없다. I-TCP 같은 경우 base station에서의 forwarding 시간은 상당하지만 이에 대한 고려가 없다. 또, hint를 사용할 수 없는 환경에서 multicast handoff의 성능은 크게 떨어진다.

3. 제안된 Handoff Scheme

Multicast를 이용할 때라도 buffer의 양이 큰 경우에 handoff 지연시간은 forwarding 시간에 영향을 받는다. forwarding 시간을 줄이기 위해 미리 multicast를 하여 두 base station에 packet을 보존하려 하는 방법은 handoff해 갈 basestation에서 유지해야 하는 buffer의 양이 지나치게 커질 수 있다. hint를 사용할 수 없는 환경에서도 미리 multicast하는 것은 마찬가지로 buffer의 양이 지나치게 커지는 문제점을 갖는다.

이러한 문제점을 해결하기 위하여 기존의 base station에서 새로운 base station으로 아직 ack를 받지 못한 packet들을 forwarding하고 ack를 받은 packet들을 제거하는 forwarding handoff 알고리즘을 제안한다. 또, 불필요한 forwarding의 발생을 막기 위해 3단계의 hint를 이용하도록 하여 사용자의 불규칙적인 움직임으로 인한 불필요한 handoff와 forwarding을 막을 수 있도록 한다.

3.1 Forwarding Handoff Scheme

그림 ??는 forward handoff 방식을 나타낸다. snoop module과 같은 caching buffer를 유지하는 base station에서 handoff할 가능성이 있는 새로운 base station으로 아직 ack를 받지 못한 packet들을 forwarding하는 것이다. 이때 새로이 받은 ack도 함께 전송하여 이미 ack를 받은 packet들은 제거하도록 한다. 새로운 base station에서 유지해야 하는 buffer의 크기는 전적으로 기존 base station에서 forwarding하는 packet의 양에 따른다. 기존 base station은 자신의 buffer양과 packet forwarding 시간, multicast join 시간 등을 고려하여 새로운 base station으로 forwarding하는 packet의 양을 결정할 수 있다. 기존 base station은 새로운 ack를 받을 때마다 새로운 ack와 함께 새 unacked packet을 새 base

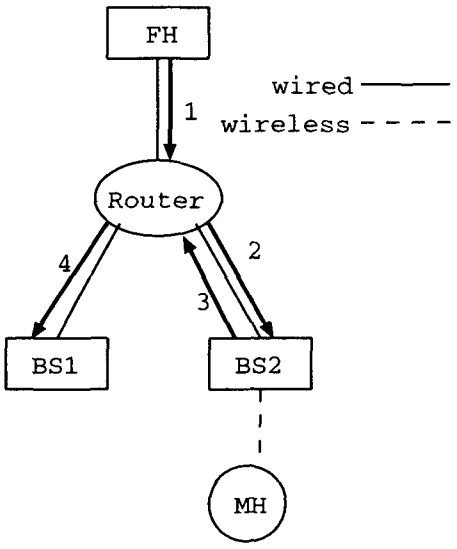

그림 2: Foward Handoff Scheme

station으로 forwarding하도록 하여 항상 buffer의 양을 일정하게 유지시키고 상태를 일치시킨다.

이 forwarding handoff를 사용함으로써 handoff 시에 mobile host는 handoff하자마자 새 base station과 통신을 재개할 수 있다. buffer에 있는 packet과 통신을 유지하는 동안에 기존 base station은 나머지 packet들을 forwarding하고 새 base station은 multicast join을 신청하도록 한다. 새 base station에 유지되는 buffer의 크기에 따라 handoff 지연시간을 이상적으로 0으로 줄일 수 있다.

3.2 3단계 hint의 사용

불필요한 hint의 발생과 forwarding을 피하기 위해 3단계 hint의 사용을 제안한다 (그림 3). 각각의 hint는 mobile host의 움직임과 base station에서 보내는 신호의 강도로부터 hystersis를 통해 얻어질 수 있으며 각 hint에서의 handoff 가능성에 따라 3단계로 분류된다. cell의 중심에서 바깥으로 움직이는 mobile host에 먼저 영향을 주는 hint는 2단계 hint이다. 이 hint를 받은 base station은 mobile host가 handoff할 가능성이 큰 base station으로 unacked packet을 forwarding하기 시작하며 base station이 새로운 ack를 받을 때마다 새로운 packet과 ack를 forwarding한다. 더욱 밖으로 움직이면 3단계 hint를 받게 되며 그러면 base station은 모든 packet의 forwarding을 시작하고 새로운 base station은 multicast join을 시작한다. 반대로 2단계 hint 후에 다시 안쪽으로 움직여서 handoff할 가능성이 어느 정도로 적어지면 1단계 hint를 받게 되고 forwarding을 그만두게 된다. 이러한 3단계 hint를 사용함으로써 불필요한 forwarding과 handoff의 발생가능성을 줄일 수 있다.

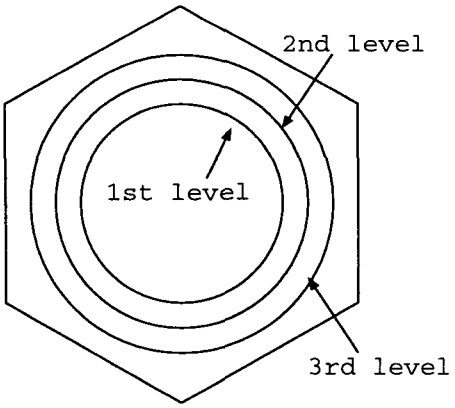

그림 3: 3단계 hint의 사용

4. 시뮬레이션

4.1 시뮬레이션 모델

본 연구의 시뮬레이션 모델은 fixed host module, mobile host module,base station module, router module, wired link module, wireless link module로 구성되어 있다. 각 모듈은 입력 버퍼 (incoming buffer)를 가지고 있고 이 입력 버퍼를 통해 패킷을 받는다. 예를 들어, A 모듈에서 B 모듈로의 패킷의 전송은 A 모듈이 B의 입력 버퍼에 패킷을 write하는 것으로 이루어지고, B는 매 시간 (time unit) 자신의 입력 버퍼를 조사하여 처리할 시간이 된 패킷을 처리하게 된다. 이 때 time unit은 0.1ms로 하였다.

본 연구에서 wired link는 최대 대역폭이 10 Mbps인 ethernet으로 가정하였고, wireless link는 mobile host당 최대 대역폭이 약 2 Mb/s인 AT&T Wavelan으로 가정하였다. 또한, 전송되는 TCP/IP 패킷의 길이는 576bytes로 가정하였다. 이 가정에 따라 계산된 각 link의 transmission delay을 비롯한 가정된 지연시간은 다음과 같다.

- **WireTrans_DELAY**
  Transmission delay of the wired link
- **WireProp_DELAY**
  Propagation delay of the wired link
- **WxTrans_DELAY**
  Transmission delay of the wireless link
- **WxProp_DELAY**
  Propagation delay of the wireless link
- **MH_DELAY**
  Time for the mobile host to acquire the wireless channel

Delay	Value
WireTrans_DELAY	0.9ms
WireProp_DELAY	2 ms
WxTrans_DELAY	4.5 ms
WxProp_DELAY	2 ms
MH_DELAY	5 ms

[표 1]

다음은 시뮬레이션 모델의 각 모듈에 대한 설명이다.

4.1.1 Fixed host module

Fixed host는 mobile host에 데이타를 보내는 source 역할을 한다. 이 때, 항상 전송할 패킷이 존재하는 infinite source인 것으로 가정한다. Fixed host 모듈은 패킷 생성 부분과 입력단자 부분의 두 부분으로 나누어져 있다. 패킷 생성 부분에서는 time-out이 되어 재전송되어야 할 패킷이 없는 때, 새로운 패킷을 생성하여 전송한다. 입력 단자 부분에서는 목적지, 즉 mobile host로부터의 acknowledgement를 받고, 마지막으로 받은 acknowledgement 패킷을 가리키는 변수의 값을 변경시킨다.

위의 두 부분 이외에도, fixed host는 패킷의 손실을 발견하기 위해 타이머를 가지고 있다. 이를 위해 각 패킷이 전송되는 시간을 기록하고 acknowledgement가 도착했을 때에는 타이머를 재시동 (restart)시킨다. 이 때, 타이머가 갖는 time-out 값은 acknowledgement를 받지 않은 첫번째 패킷이 전송되는 시간에 새로운 round-trip-time에 β를 곱한 값을 더한 것이다(본 연구에서는 $\beta = 2$[3]). Time-out 시간까지 이 패킷의 acknowledgment가 오지 않으면 이 패킷이 손실된 것으로 보고 congestion control algorithm을 작동시킨다. Time-out에서 사용되는 round trip time을 다음 식에 의해서 구해진다[3].

$$rtt(roundtriptime) = \alpha * Old_RTT + (1 - \alpha) * New_RTT_sample$$

본 보고서에는 α의 값이 0.4, 0.5, 0.7 일 경우 시뮬레이션한 결과를 보인다.

4.1.2 mobile host module

Mobile host 모듈은 두 가지 기능을 수행한다. 첫째는 source 즉 fixed host로부터 받은 패킷에 대한 acknowledgment를 보내는 일이고, 둘째는 mobile host의 이동성을 모델링하는 일이다. Mobile host가 hint를 받고 handoff를 함에 따라 packet이 forward되고 multicast join을 하게된다. multicast handoff에서는 hint를 받자마자 packet의 forwarding과 multicast join을 시작한다. 이때 각각의 operation을 위해 필요한 delay들이 감안되었다. forward handoff 시에는 3단계 hint를 받았을 때 새 basestation에는 일정량의 packet이 이미 존재한다. 따라서 나머지 packet들만을 forwarding하고 multicast

join을 시작하게 된다.

4.1.3 base station module

Base station은 wireless link를 통해 mobile host로 패킷을 전송하는 일을 한다. 이 때 wireless link 상의 패킷의 손실로 인한 TCP/IP의 성능저하를 막기 위해 base station에 snoop 모듈[?]을 둔다. Snoop 모듈에서는 local timer를 두어 wireless link상의 패킷 손실을 detect 한다. Local timer의 time-out이 발생했을 때, 캐싱해 둔 패킷을 wireless link 상으로 재전송한다. handoff 구현을 위해 current와 next 두 개의 base station을 두었다.

4.1.4 router module

Router 모듈은 current base station 또는 next base station으로 packet을 routing한다. 멀티캐스팅 handoff 시의 멀티캐스팅도 router 모듈에서 수행한다.

4.1.5 wired link module

Wired link module은 패킷이 wired link을 통해 전송될 때 생기는 전송 지연을 구현한다. Wired link 상의 전송 지연값은 [표 1]과 같다.

4.1.6 wireless link module

Wireless link module은 wireless link 상에서의 패킷의 전송 지연을 구현하며, 정해진 패킷 손실율에 따라 패킷을 손실시키는 일을 한다. wired link 상의 전송 지연값은 [표 1]과 같으며, 패킷 손실율은 0.05로 가정하였다.

4.2 시뮬레이션 결과

그림 4, 그림 5, 그림 6은 주어진 시간동안 acknowledgment를 받은 패킷의 개수를 나타낸다. 기존의 TCP/IP 프로토콜에서 패킷 손실율이 클 경우, 시간이 지남에 따라 전송율이 매우 감소되는 것과 비교해서 Snoop module를 사용한 두 경우는 시간이 지남에 따른 패킷의 전송율의 저하가 미소함을 볼 수 있다. 이는 base station에서 구현된 snoop 모듈[?]의 local retransmission의 효과로 인한 것이라 할 수 있다. 또한, 위의 결과로부터 multicast handoff에 비해 제안한 forwarding handoff 알고리즘의 성능이 우수한 것을 볼 수 있다.

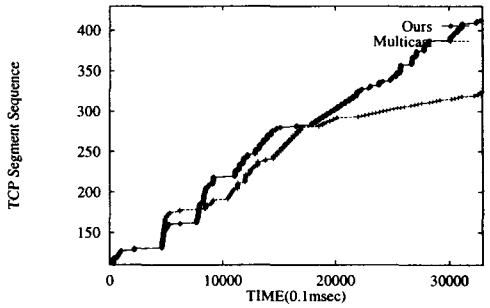

그림 4: $\alpha = 0.4$

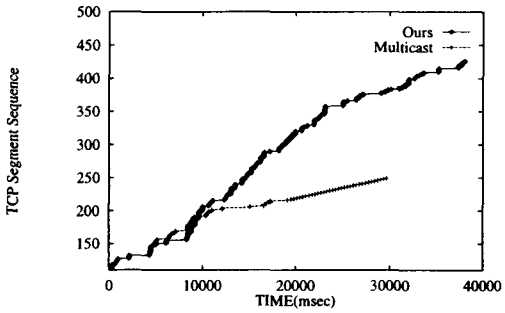

그림 5: $\alpha = 0.5$

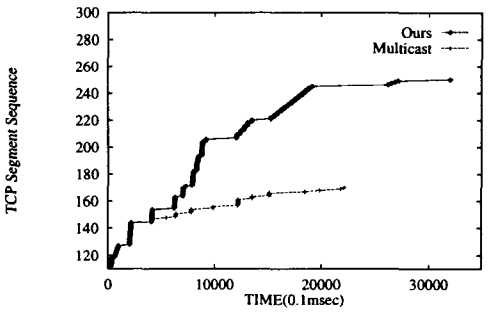

그림 6: $\alpha = 0.7$

5. 결론

본 논문에서는 wireless network에서 TCP/IP의 성능 향상을 위한 새로운 handoff 알고리즘을 제안했다. 제안된 handoff 알고리즘은 기존 base station에서 새 base station으로 ack를 받지 않은 packet들을 forwarding하여 새 base station에 일정량의 buffer를 유지하도록 해

서 forwarding으로 인한 handoff latency를 줄이도록 하였다. 또한 제안된 handoff 알고리즘의 효과적인 사용을 위하여 3단계 hint를 사용할 수 있는 방안을 제안하였다. 1단계 hint에서는 forwarding을 정지시키고 2단계 hint에서는 unacked packet의 forwarding을 시작한다. 3단계 hint에서는 모든 packet의 forwarding과 multicast join을 시작한다. 3단계 hint의 사용을 통해 불필요한 forwarding과 handoff의 발생을 줄일 수 있다.

본 논문에서는 simulation을 통하여 제안된 handoff 알고리즘의 우수성을 보였다. simulation을 위해 TCP와 snoop module이 구현되었으며 여러가지 round trip time 계산방법에 따라 ack를 받은 TCP packet들의 sequence를 비교하였다. 그 결과 제안된 handoff 알고리즘이 multicast handoff에 비하여 handoff latency를 줄일 수 있음을 알 수 있었다.

앞으로 실제 system에서 실험을 해보아야 하겠고, 새로운 base station에서 buffer 유지를 위한 mechanism이 더 연구되어야 하겠다. 또한 인접한 셀에서 반복적으로 일어나는 handoff시에 두 buffer의 관리 방법등이 더욱 연구되어야 하겠다.

참고 문헌

[1] A. Bakre and B. R. Badrinath.

[2] H. Balakrishnan et al. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks. *Wireless Networks*, 1(4):469–481, 1995.

[3] D. E. Comer. Internetworking with TCP/IP, vol. 1, Principle, Protocols, and Architectures. 1991.