

# 하향식 수직적 핸드오버를 위한 무선 TCP 성능 향상 기법과 성능 분석

준회원 김 호 진\*, 종신회원 이 수 경\*\*

## Improving TCP Performance for Downward Vertical Handover

HoJin Kim\* Associate Member, SuKyoung Lee\*\* Lifelong Member

### 요 약

본 논문에서는 이질적 무선망 중에서 3G 셀룰러 망에서 WLAN으로 넘어가는 하향식 핸드오버에 초점을 맞추고 단말의 이동성에 따른 TCP 전송 프로토콜의 성능을 향상시키기 위한 기법을 제안한다. 하향식 핸드오버를 전후로 한 링크의 특성은 급격히 변화하게 되는데 특히 사용 가능한 대역폭이 급격하게 증가하고 RTT (Round-Trip-Time)는 감소하게 된다. 이 때 핸드오버 전후로 전송되는 패킷의 순서는 뒤바뀌게 되어 TCP는 false fast retransmission을 함으로써 TCP의 성능을 저하시킨다. 본 논문에서는 이러한 문제를 해결하기 위한 새로운 리오더링 (reordering) 기법을 제안하여 핸드오버로 인한 TCP 성능 저하 현상을 막고자 한다.

**Key Words :** TCP, Reordering, Downward Vertical Handover, False Fast Retransmission

### ABSTRACT

Interconnecting wireless local area networks (WLANs) with third generation (3G) cellular networks has become an issue of great interest. However, a Vertical Handover (VHO) causes an abrupt change in link bandwidth. Due to such a change, TCP triggers unnecessary fast retransmission during a Downward VHO (DVHO) from a cellular network to a WLAN, causing throughput degradation. Thus, we propose a new reordering mechanism for DVHO that suppresses unnecessary retransmission due to the spurious duplicate acknowledgments. We analytically investigate the throughput of TCP in the literature and our proposed scheme. Through the numerical and simulation results, it is shown that our proposed TCP achieves better performance in terms of throughput, compared with Nodupack with SACK.

### 1. 서 론

이질적 무선망이란 서로 다른 특성을 가지는 무선 액세스망이 서로 연동되어 운영되는 무선망을 의미한다. 이동 단말은 서로 다른 이질적 망에 접속하여 사용자의 사용시간이나 트래픽 특성에 따라 적합한 망을 선택하여 사용할 수 있다. 특히 3G/4G 셀룰러 망과 WLAN의 연동은 데이터 전송률과 커

버리지 (coverage) 측면에서 상호간의 장단점을 보완할 수 있지만 셀룰러 망에서 WLAN으로 이동하는 하향식 핸드오버 (Downward Vertical Handover) 시에는 링크 대역폭이 급격하게 증가하고 RTT (Round-Trip-Time)이 감소하게 되어 TCP의 성능을 저하시키게 되므로, 사용자가 고속 데이터 서비스를 이용하기 위해서는 이질적 무선망을 효과적으로 연동할 수 있는 전송 메커니즘이 필요

\* 본 논문은 Optical Internet Research Center (R11-2000-074-02001-0) 지원으로 수행되었습니다.

\* 연세대학교 컴퓨터과학과 무선네트워킹연구실(hojin@winet.yonsei.ac.kr), \*\* 연세대학교 컴퓨터과학과 교수(sklee@cs.yonsei.ac.kr)

논문번호 : KICS2007-08-358, 접수일자 : 2007년 8월 8일, 최종논문접수일자 : 2007년 10월 12일

하다. 패킷들은 RTT가 서로 다른 두 링크로 전달되어 송신측에서 패킷을 전달한 순서와 다르게 수신측으로 패킷이 도착하는 리오더링(reordering) 문제가 발생하게 된다. 잘못된 순서로 패킷이 전달되면 TCP는 이를 링크의 congestion 상황으로 인식하여 패킷이 손실되었다고 판단한다. 따라서 패킷이 손실되지 않았음에도 불구하고 그림 1에서처럼 잘못된 순서로 전달된 패킷에 대해 dupacks (duplicate Acknowledgements)를 신호로 보냄으로써 패킷을 다시 전송할 것을 요청하여 false fast retransmission 문제를 일으킨다. false fast retransmission이 발생하면 TCP는 congestion control에 따라 cwnd (congestion window size)와 ssthresh (slow start threshold) 값을 감소시켜 TCP 전송률(throughput)을 떨어뜨려 전체 네트워크의 효율성을 떨어뜨린다.

본 논문에서는 순서가 뒤바뀐 패킷의 도착으로 인한 불필요하게 발생하는 false fast retransmission을 막고 이로 인한 TCP 성능저하 현상을 막기 위해 새로운 리오더링 기법을 제안하고자 한다. TCP 리오더링 기법에 대해서 기존에 제시된 연구 [1]과 [2]에서는 일정 개수의 dupacks를 받을 때까지는 패킷을 재전송하지 않도록 dupthresh (dupack threshold)를 조절함으로써, 일시적으로 발생할 수 있는 false fast retransmission 문제를 해결한다. 그러나 dupthresh의 값을 지나치게 높게 설정하게 되면 패킷이 손실된 경우 timeout, 재전송의 지연, congestion control의 지연 등의 문제점 [5]을 야기할 수 있다. [3]에서는 핸드오버가 발생할 때 dupacks를 보내지 않음으로써 congestion control이 작동하지 않도록 하는 방식을 제안하고 있지만 dupacks의

개수를 정하는 방법을 제안하고 있지 않아 실제 환경에서 적용하기 어렵다. 따라서 본 논문에서는 false fast retransmission 문제를 막기 위해 dupacks를 보내지 않는 리오더링 기법을 적용하고 송신측에서 적합한 전송률을 조절하여 새 링크의 네트워크 효율성을 높일 수 있도록 제안한다.

본 논문의 구성은 다음과 같다. II절에서는 하향식 핸드오버에서의 리오더링 기법을 소개하고 III절에서는 모의실험을 통해 TCP의 성능 향상을 확인한 후에 IV절에서 결론을 맺는다.

## II. TCP 리오더링 기법

하향식 핸드오버 시에는 RTT 값이 급격하게 감소함에 따라 리오더링 문제가 발생하게 된다. 핸드오버 이전에 셀룰러 망을 통해 전송되었으나 아직 수신측에 도착하지 않은 패킷을 in-flight 패킷이라고 하고, WLAN에서 전송되어 순서가 뒤바뀐 채로 도착하는 패킷을 out-of-order 패킷이라 하자.

본 논문에서 제안하는 TCP 방식을 TCP-D라고 할 때 TCP-D는 dupthresh 값을 늘려주는 기존의 방식 대신에 in-flight 패킷이 모두 도착할 때까지 dupacks 대신 정상적인 ACK (Acknowledgment)를 보냄으로써 fast recovery [4]알고리즘이 작동하지 않도록 한다. 이러한 방식은 dupthresh를 크게 늘렸을 때의 위험을 방지하면서 동시에 기존에 제시된 기법들과 동일한 효과를 낼 수 있게 한다.

이 때 TCP 수신측에서는 패킷의 sequence number를 이용하여 셀룰러 망을 통해 전송되고 있는 in-flight 패킷의 개수를 측정하여, 이 값만큼의 패킷이 오는 동안에는 dupacks 대신 정상적인 ACK를 보내도록 한다. 즉, 핸드오버가 이루어지는 순간 수신측이 송신측에게 핸드오버가 일어났음을 알려주면 이때부터 TCP 송신측은 제안한 리오더링 기법을 적용한다. 이 때 측정된 in-flight 패킷의 개수를 N이라고 하자.

N의 개수를 포함하여 수신측에서 송신측으로 하향식 핸드오버 시 필요한 정보를 전달해주기 위해 TCP 수신측은 핸드오버 이후 전송된 첫 번째 패킷에 대한 ACK를 통해 전달할 수 있도록 TCP 헤더 옵션 정보를 추가적으로 구현한다. 그림 2는 TCP 헤더 옵션에서 구현된 필드를 보여준다. HO 필드는 핸드오버가 일어났음을 알려주는 플래그, N은 핸드오버 시 발생한 in-flight 패킷의 개수, Available Bandwidth는 수신측에서 측정된 새 링크에 대한 사용 가능한 대역폭에 대한 정보를 각각 나타낸다.

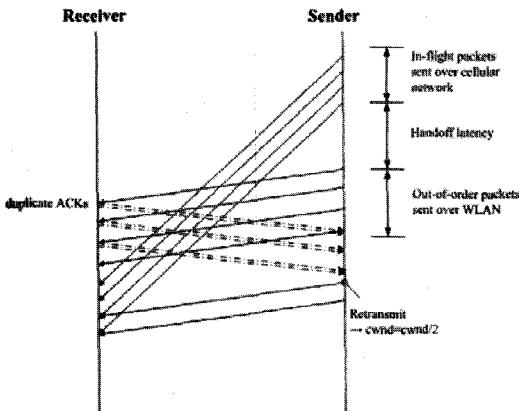


그림 1. False Fast Retransmission

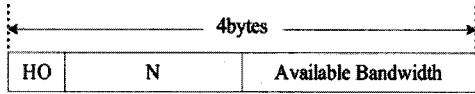


그림 2. ACK의 TCP 헤더옵션

N은 핸드오버 전후의 수신측이 받은 패킷의 sequence number의 차를 이용하여 in-flight 패킷의 개수를 구할 수 있는데, 이 값은 앞서 설명했듯이 dupacks를 보내지 않는 패킷의 개수로 설명된다. 또한 새 링크에 대한 사용 가능한 대역폭을 구하는 것은 핸드오버 이후에 TCP가 새 링크의 대역폭을 보다 효율적으로 사용할 수 있게 하기 위한 것인데, 이 값을 측정하는 방식은 [6]에서 제안하는 방식을 이용하여 TCP 수신측은 핸드오버 지연시간 동안 일정량의 표본 패킷들을 새 링크에 보냄으로써 사용 가능한 대역폭을 측정하여 구할 수 있다.

이처럼 순서가 뒤바뀐 패킷들의 리오더링 문제를 해결하더라도 두 링크의 급격한 변화로 인하여 핸드오버 시 TCP의 성능을 저하시키는 premature timeout 문제가 발생할 수 있다. 이는 하향식 핸드오버에서 RTT 값이 짧아짐으로써 retransmission timer 값도 새로 측정된 RTT 값에 의해 업데이트되면서 발생하는 문제이다. 즉, in-flight 패킷들이 모두 도착하지 않았음에도 불구하고 새로 옮겨간 WLAN에서 전송받은 패킷들에 의하여 타이머 값이 짧아지게 되고, 이로 인하여 in-flight 패킷에 대하여 premature timeout이 발생하여 송신측에서는 불필요하게 패킷을 재전송하게 된다. 동시에 TCP는 congestion control을 발생시켜 TCP 전송률을 저하시키게 된다. 이 문제를 해결하기 위해 TCP-D는 핸드오버 시 셀룰러 망에서 오던 패킷에 대한 ACK를 모두 받을 때까지 이전 링크의 타이머 값을 저장함으로써 timeout이 일어나는 것을 막을 수 있도록 한다. 이 때 셀룰러 망의 retransmission timer 값을  $rT$ 라고 하고,  $rT$  타이머는 in-flight 패킷이 모두 도착할 때까지만 유지하도록 한다.

사용자 단말이 셀룰러 망을 통해 마지막으로 받은 패킷을  $d_i$ , WLAN을 통해 처음으로 받은 out-of-order 패킷을  $d_j$ 라고 하자. 리오더링 기법이 적용되는 동안 송신측과 수신측은 다음과 같이 작동한다.

**TCP 수신측**

- 수신측에서 L2 (Layer 2) trigger를 받게 되면 TCP 송신측에 핸드오버 사실을 알리기 위해 ACK 패킷 안의 TCP 헤더 옵션의 HO 필드를 통해 해당 정보를 전송한다.
- WLAN을 통해 받은 패킷이 out-of-order 패킷일 경우, 수신측은 송신측에게 in-flight 패킷의 개수( $N=j-i-1$ )를 측정하여 ACK 패킷안의 TCP 헤더 옵션을 통해 알려준다.
- in-flight 패킷의 개수를 지속적으로 측정하기 위해  $d_i, d_{i+1}, \dots, d_{j-1}$  사이에 있는 패킷을 받을 때마다 N을 1씩 감소시킨다.

**TCP 송신측**

- 송신측에서  $d_j$  패킷에 대한 ACK를 받아서 WLAN에 대한 RTT 값을 측정할 수 있게 되고 retransmission timer 값을 업데이트 해준다. 동시에 셀룰러 망에서 사용하던  $rT$ 를 저장한다.  $rT$  값은 in-flight 패킷이 모두 전송되고 나면 리셋 된다.
- $rT$  타이머가 종료되지 않고, in-flight 패킷이 남아있는 동안에는 (즉,  $N>0$ ) 송신측은 congestion control을 작동시키지 않고 WLAN을 통해 지속적으로 패킷을 전송한다.
- in-flight 패킷이 더 이상 남아있지 않은 경우에는 (즉,  $N=0$ ) 기존의 TCP 메커니즘을 따른다.

그림 3은 TCP-D에서 제안한 리오더링 기법을 요약하고 있다.

이 때 리오더링 문제를 해결하는 과정에서 핸드오버 지연시간이나 링크의 특정한 트래픽 문제로 인하여 패킷의 손실이 발생할 수 있다. 따라서 패킷 손실이 발생할 경우 TCP-D가 어떻게 작동하는지 다음 네 가지 경우로 나누어 보면 다음과 같다.

case 1. 셀룰러 망을 통해 전송되는 in-flight 패킷이 손실되는 경우,  $rT$ 값을 리셋하고, WLAN을 통해서 손실된 패킷을 재전송한다. 단 이 경우의 패킷 손실은 핸드오버 지연시간으로 인하여 발생한 손실일 확률이 크므로 송신측에서는 congestion control을 작동하지 않는다. 즉, 손실된 in-flight 패킷을 재전송하고 나서도 cwnd와 ssthresh의 값을 그대로 유지시킨다.

case 2. WLAN을 통해 전달되는 패킷이 손실되는 경우, WLAN의 타이머를 리셋하고 재전송한다. in-flight 패킷이 전송되고 있는 중이고 핸드오버 직후 발생하는 패킷 손실은 case 1의 경우와 같이 링크의 congestion 상황이기 보다는 핸드오버 지연시간으로 인한 손실일 가능성이 더 크므로 손실된 패킷에 대해 재전송을 하되, congestion control은 작동하지 않는다.

case 3. in-flight 패킷이 모두 도착한 이후, WLAN을 통해 전달되는 패킷이 손실되는 경우, 기본 TCP 메커니즘에 따라 손실된 패킷을 재전송하고 cwnd와 ssthresh의 값을 줄인다.

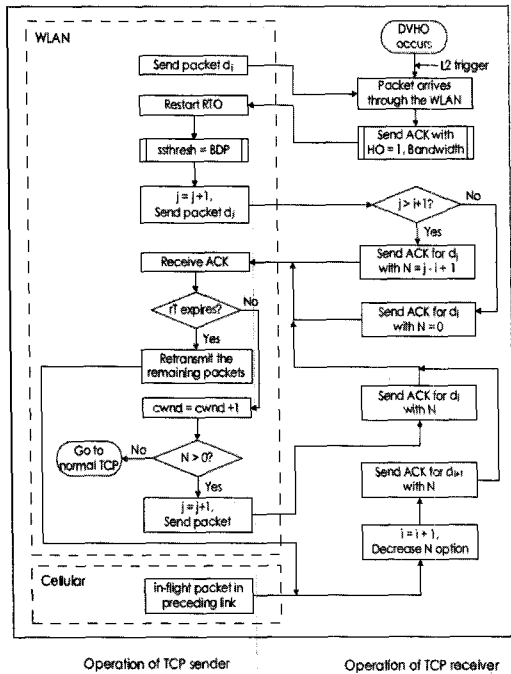


그림 3. TCP-D에서 제안한 리오더링 메커니즘

### III. 시뮬레이션 결과

III절에서는 시뮬레이션을 통해 TCP-D 메커니즘의 성능 향상을 알아보기 위해 리오더링이 발생하는 동안 dupacks를 보내지 않는 메커니즘인 [3]과 비교하고자 하며, 비교 메커니즘은 'Nodupack with SACK'으로 명명하도록 한다.

시뮬레이션은 NS-2.28<sup>[7]</sup>의 확장 모듈을 이용하여

테스트 하였다. [8]에서 설정한 파라미터 값을 참조하며, WLAN은 각각 2Mbps, 11Mbps의 대역폭을 가지고, RTT는 각각 100msec, 20msec로 고정하였다. 셀룰러 망 UMTS 망을 모델로 하여 384Kbps의 대역폭을 가지며 이때의 RTT는 300msec으로 설정하였다. 본 논문에서 제안하는 셀룰러 망에서 WLAN으로 일어나는 하향식 수직적 핸드오버에서는 L2 trigger가 일어나는 지연 시간을 포함하여 약 300msec<sup>[9]</sup>의 핸드오버 지연시간을 갖는다고 가정한다. 발생하는 FTP 트래픽은 1000bytes의 패킷 사이즈로 설정하였다<sup>[8]</sup>.

그림 4는 핸드오버 전후로 관찰되는 송신측의 윈도우 크기의 변화를 보여준다. 384Kbps의 대역폭을 갖는 셀룰러 망에서 2Mbps의 대역폭을 갖는 WLAN으로 이동하는 경우인 그림 4의 (a)를 살펴 보면 10sec에서 핸드오버가 발생하고 TCP-D는 false fast retransmission을 해결하여 불필요한 congestion control을 막고 slow start 메커니즘에 따라 작동하게 된다. ssthresh가 BDP (Bandwidth Delay Product)<sup>[10]</sup>에 이르고 난 뒤에는 congestion avoidance 메커니즘으로 전환한다. 반면 Nodupack with SACK 메커니즘에서는 셀룰러 망에서의 cwnd가 그대로 유지되어 WLAN으로 핸드오버가 이루어진 후에도 해당 대역폭에 적합한 전송률을 갖지 못하게 된다. 그림 4의 (b)인 셀룰러 망에서 11Mbps의 대역폭을 갖는 WLAN으로 이동하는 경우도 마찬가지로 TCP-D는 제안된 리오더링 기법과 slow start 메커니즘에 따라 적합한 전송률을 더 빠르게 찾고 있음을 알 수 있다.

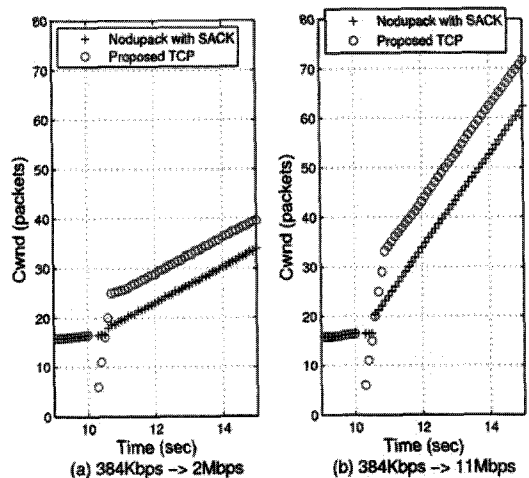


그림 4. TCP-D의 시간(sec)당 cwnd (packets) 성능 향상 정도

그림 5와 그림 6은 TCP-D와 Nodupack with SACK 메커니즘의 핸드오버 전후로 측정된 TCP 송신측이 보낸 패킷의 sequence number, TCP 송신측이 받은 ACK의 sequence number, TCP 수신측이 받은 데이터의 sequence number를 보여주고 있으며 각각 384Kbps의 셀룰러 망에서 2Mbps의 WLAN으로 이동하는 경우와 11Mbps의 WLAN으로 이동하는 경우에 따라 두 메커니즘의 성능을 비교하였다.

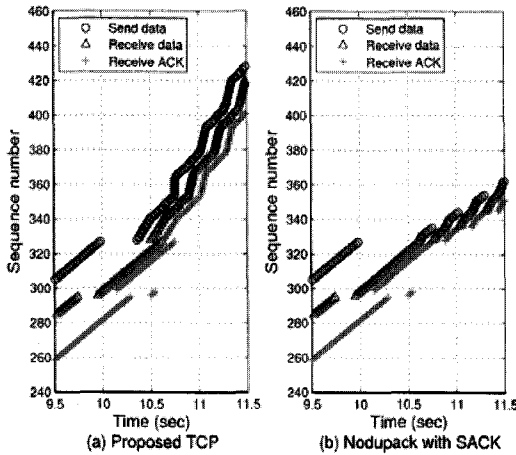


그림 5. 384 Kbps → 2Mbps 이동시 TCP-D 메커니즘의 시간(sec)당 보낼 수 있는 패킷의 양 비교

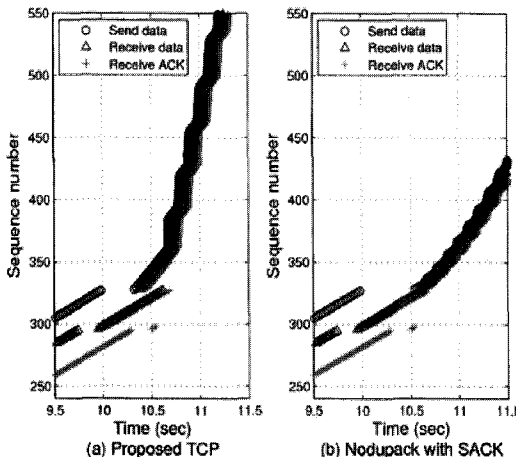


그림 6. 384 Kbps → 11Mbps 이동시 TCP-D 메커니즘의 시간(sec)당 보낼 수 있는 패킷의 양 비교

그림 5의 (a)에서 보여주고 있는 TCP-D 메커니즘에서는 핸드오버 지연시간이 10초에서 10.3초까지

지속되는 동안 셀룰러 망으로 전송된 패킷은 아직 도착하지 않았고, WLAN에서는 그 다음 시퀀스 넘버를 가진 패킷이 전송되기 시작한다. in-flight 패킷이 10.8 초까지 도착하는 동안 WLAN에서 오는 out-of-order 패킷을 전송 받고 있고, TCP-D 메커니즘에서는 WLAN으로 전송된 패킷에 대해 dupacks를 보내는 대신 정상적인 ACK를 보낸다. 결과적으로 리오더링 기법이 적용되는 동안에도 지속적으로 cwnd를 증가시킬 수 있도록 한다. 이 때 핸드오버 이후의 cwnd는 slow start 메커니즘에 따라 작동하며 WLAN의 상대적으로 큰 대역폭에 맞추어 빠르게 전송률을 조절함으로써 전체 네트워크의 효율성을 높이고 있음을 알 수 있다. 그림 5의 (b)에서 보여주고 있는 Nodupack with SACK 메커니즘에서는 out-of-order 패킷에 대하여 dupacks를 보내지는 않지만 해당 패킷에 대한 ACK도 전송하지 않으므로 리오더링 문제를 해결하고 있는 동안에는 셀룰러 망에서의 전송률을 유지하고 있어 TCP의 성능 향상을 기대할 수 없다. 또한 수직적 핸드오버 이후 cwnd의 시작 크기는 핸드오버 이전의 크기를 그대로 가져옴으로써 핸드오버가 끝난 이후에도 송신측의 전송률은 대역폭이 크게 변화한 것을 인식하지 못하고 셀룰러 망의 대역폭에 맞추어 패킷을 전송하고 있어 WLAN에서 사용가능한 대역폭에 맞는 전송률을 찾을 때까지 긴 시간이 걸릴 것으로 추측되며 따라서 TCP-D 메커니즘에 비해 성능이 떨어짐을 알 수 있다.

그림 6에서도 그림 5와 마찬가지로 TCP-D의 성능 향상 정도를 볼 수 있다. 특히 그림 6의 (a)가 (b)에 비해 그림 5에서보다 전송률 측면에서 볼 때 더 큰 성능 향상이 있음을 알 수 있는데 증가된 사용가능한 대역폭 (11Mbps)에 따라 시간당 보낼 수 있는 패킷의 수가 증가하였기 때문이다. 즉, Nodupack with SACK 메커니즘에서는 in-flight 패킷이 모두 도착할 때까지 TCP 전송률을 증가시킬 수 없기 때문에 그만큼의 시간동안 성능저하가 발생하게 되어 TCP-D와의 성능 격차가 더 벌어지기 때문이다.

표 1은 시뮬레이션 측정 시간동안 TCP-D와 Nodupack with SACK 메커니즘의 전송속도를 측정된 결과이다. 이 때 TCP-D와 Nodupack with SACK 메커니즘의 핸드오버 이후의 전송률은 핸드오버 직후부터 cwnd가 안정적인 상태로 들어가고 사용가능한 최대 대역폭에 이르러 패킷 손실이 발생하기까지의 시간 동안 전송한 패킷의 양을 구하여 이에 대한 평균값을 비교하였다.

표 1. 하향식 핸드오버 직후 평균 전송률

	384Kbps → 2Mbps	384Kbps → 11Mbps
TCP-D	0.54 Mbps	1.77 Mbps
Nodupack with SACK	0.19 Mbps	0.56 Mbps

TCP-D는 Nodupack with SACK 메커니즘에 비해 2Mbps의 WLAN로 이동할 때와 11Mbps의 WLAN으로 이동할 때, 각각 최대 184%, 216%까지 전송률이 증가하였음을 알 수 있다. 그림 5와 그림 6에서 확인했던 것처럼 2Mbps의 WLAN으로 이동할 때보다 대역폭이 더 큰 11Mbps의 WLAN으로 핸드오버가 발생할 때 더 큰 성능 향상 효과가 있음을 알 수 있다.

이상에서 살펴본 것처럼 TCP-D는 리오더링 기법을 통하여 리오더링 문제 및 premature timeout 문제를 해결하여 TCP의 성능을 저하시키는 false fast retransmission 현상을 막음으로써 기존 연구에 비해 이질적 무선망에서의 TCP 성능을 향상시키고 있음을 알 수 있다.

#### IV. 결론

셀룰러 망과 WLAN의 통합망에서 이동하는 단말은 하향식 핸드오버 시 링크의 대역폭이 급격하게 변화할 때 TCP의 성능이 저하되는 문제가 발생한다. 이 때 리오더링 문제 및 premature timeout으로 발생하는 불필요한 false fast retransmission을 막기 위해 본 논문에서는 새로운 TCP 전송 메커니즘을 제안하였다. 특히 서로 다른 링크의 특성을 갖는 이질적 무선망사이의 수직적 핸드오버에서 발생할 수 있는 문제를 해결하기 위해 제안된 리오더링 기법은 핸드오버 시 발생하는 리오더링 문제를 해결하고 이에 따라 발생하는 premature timeout을 사전에 방지하면서 동시에 하향식 핸드오버에 적합하게 설계된 cwnd의 조절을 통하여 기존에 제안된 리오더링 기법의 단점을 보완하고 TCP의 성능 저하 현상을 해결하여 네트워크의 전체적인 효율성을 증가시키고 있다.

#### 참고 문헌

[1] E. Blanton and M. Allman, "On Making TCP More Robust to Packet Reordering",

ACM Computer Communication Review, Vol. 32, No.1, pp. 20-30, Jan. 2002.

[2] C. Ma and K. Leung, "Improving TCP Robustness under Reordering Network Environment", *Proc. IEEE GLOBECOM'04*, Vol. 2, pp. 828-832, Nov. 2004

[3] W. Hansman and M. Frank, "On Things to happen during a TCP Handover", *Proc. IEEE LCN'03*, pp. 109-118, October 2003.

[4] B. Forouzan, "TCP/IP Protocol Suite", 3rd Edition, Mc GrawHill, 2007

[5] M. Zhang, B. Karp, S. Floyd, and L. Peterson, "RR-TCP: a reordering-robust TCP with DSACK", *Proc. the 11th IEEE International Conference on Network Protocols*, pp. 95-106, Nov. 2003.

[6] H. Fu, L. Lin, and W. Jia, "Efficient wireless link bandwidth detection for IEEE 802.11 Networks", *IEEE ICC'05*, May 2005.

[7] The Network Simulator - ns2,  
<http://www.isi.edu/nsnam/ns>

[8] N. Wang, Y. Wang and S. Chang, "A Fast Adaptive Congestion Control Scheme for Improving TCP Performance during Soft Vertical Handoff", *Proc. IEEE WCNC'07*, pp. 3641-3646, Mar. 2007.

[9] M. Bernaschi, F. Cacace, G. Iannello, S. Za, and A. Pescapè, "Seamless internetworking of WLANs and cellular networks: architecture and performance issues in a Mobile IPv6 Scenario", *IEEE Wireless Communications*, Vol. 12, No. 3, pp. 73-80, Jun. 2005.

[10] J. Hoe, "Improving the Start-up Behavior of a Congestion Control Scheme for TCP", *ACM SIGCOMM'96*, Aug. 1996.

김 호 진 (HoJin Kim) 준회원  
2006년 2월 연세대학교 컴퓨터과학과 졸업  
2006년 3월~현재 연세대학교 컴퓨터과학과 석사과정  
<관심분야> TCP, Mobile IP, Power saving

이 수 경 (SuKyoung Lee) 종신회원  
2005년 9월~현재 연세대학교 컴퓨터과학과 교수  
<관심분야> 이종망 통합, Optical Network, Mobile IP, Ad-Hoc Network, 4G, WLAN