

UMTS 시스템에서 동적 무선 환경 변화에 따른 TCP 성능 향상 기법

김 남 기[†] · 박 인 용^{††} · 윤 현 수^{†††}

요 약

도시 인구 집중과 이동 인구의 증가로 이동 통신 시스템은 90년대 이후 급격한 발전을 하고 있다. 그러나 현재 널리 사용되고 있는 2세대 이동 통신 시스템은 음성에만 치중하여 개발되었고 시장이 거의 포화상태에 이르렀다. 따라서 앞으로 다가올 UMTS (Universal Mobile Telecommunication Services) 시스템과 같은 3세대 이동 통신 시스템에서는 무선 인터넷과 같은 본격적인 데이터 서비스를 위한 패킷 데이터 통신이 중요한 부분으로 떠오르고 있다. 그런데 Web이 TCP를 근간으로 하기 때문에 대부분의 인터넷 트래픽이 TCP 트래픽으로 구성되어 있고 따라서 UMTS 시스템에서 효율적인 TCP 트래픽 전송은 패킷 데이터 서비스 성능에 매우 중요한 역할을 할 것이다. 지금까지 무선 네트워크에서 TCP 성능 향상을 위한 기법들은 많이 연구되어 왔는데 UMTS 시스템에서는 그 중 링크 계층 재전송 기법을 사용한다. 하지만 이 기법이 UMTS 시스템에서 TCP 성능에 정확히 어떠한 영향을 미치는 지에 대한 연구는 아직 미진한 상태이다. 특히 동적인 무선 자원의 변화에 따른 TCP 성능 분석은 거의 진행되어진 바 없다. 따라서 본 논문에서는 UMTS 시스템에서 무선 환경 변화에 따라 TCP 성능이 어떻게 변화하는지 실험을 통해 분석해 본다. 그리고 무선 환경이 동적으로 변화하는 상황에서 TCP 성능 저하를 최소화 할 수 있는 TCP 성능 향상 기법을 제안한다. 실험 결과에 의하면 UMTS 시스템에서 무선 환경이 변화할 때 TCP 성능이 크게 저하되는데 이 때 제안된 TCP 성능 향상 기법을 적용할 경우 UMTS 시스템에 큰 부담을 주지 않으면서 무선 자원을 절약하고 TCP 성능 저하를 줄일 수 있음을 알 수 있었다.

TCP Performance Improvement Scheme on Dynamic Wireless Environment over UMTS System

Namgi Kim[†] · Inyong Park^{††} · Hyunsoo Yoon^{†††}

ABSTRACT

The mobile telecommunication system has been growing exponentially after 1990s due to the high population in a city and the growth of mobile user. In this time, the current mobile system mainly concentrates on the voice communication. However, in the next generation, mobile users want to get very diverse services via mobile terminal such as the Internet access, web access, multimedia communication, and etc. For this reason, the next generation system, such as the UMTS (Universal Mobile Telecommunication Services) system, has to support the packet data service and it will play the major role in the system. By the way, since the Web service is based on TCP, most of the Internet traffic is TCP traffic. Therefore, efficient transmission of TCP traffic will take very important role in the performance of packet data service. There are many researches about improving TCP performance over wireless network. In those schemes, the UMTS system adapts the link layer retransmission scheme. However, there are rarely studies about the exact performance of the link layer retransmission scheme in the face of dynamic changes of wireless environment over the UMTS system. The dynamic changes of wireless environment, such as wireless bandwidth, can degrade TCP performance directly. So, in this paper, we simulate and analyze the TCP performance in the UMTS system with dynamic wireless environments. Then, we propose a simple scheme for minimizing TCP performance degradation. As a result of simulation, we can find that when wireless environment is changed dynamically, the probability of TCP timeout is increased, and the TCP performance is degraded very much. In this situation, the proposed simple scheme shows good performance. It saves wireless resources and reduces the degradation of TCP performance without large overhead of the base station.

키워드 : Wireless TCP, UMTS, 3세대 이동 통신(3G Mobile Communication)

1. 서 론

도시 인구 집중과 이동 인구의 증가로 인해 이동 통신

시스템은 1990년 대 이후 급격한 성장을 해오고 있다. 이동 통신 시스템을 세대 별로 살펴 보면 1세대의 아날로그 셀룰러 시스템과 현재 사용하고 있는 GSM, cdmaOne(IS-95), US-TDMA(IS-136) 등의 2세대 디지털 시스템으로 분류할 수 있다. 1, 2세대 시스템은 음성 통신을 무선으로 가능하게 만드는데 역점을 두었다. 하지만 이제는 음성 서비스 만으로는 시장이 포화 상태에 이르렀고 소비자들도 이동 통

* 본 연구는 첨단정보기술 연구센터를 통하여 과학재단의 지원을 받았다.

† 준 회 원 : 한국과학기술원 대학원 전자전산학과

†† 정 회 원 : 금오공과대학교 컴퓨터공학부 교수

††† 정 회 원 : 한국과학기술원 전산학과 교수

논문접수 : 2002년 12월 14일, 심사완료 : 2003년 10월 20일

신을 통한 텍스트 메시지, 데이터 네트워크와 같은 새로운 서비스에 대해 많은 관심을 두고 있다. 이러한 추세에 따라 3세대 이동 통신 중 하나인 UMTS(Universal Mobile Telecommunication Services) 시스템은 빠른 데이터 전송 속도와 새롭고 유연한 통신 능력을 통해 멀티미디어 데이터 통신이 가능하도록 설계되고 있다. 즉 무선 단말을 통해 사용자에게 다양한 인터넷 서비스를 제공하고 이를 통해 새로운 시장 창출하고자 하는 것이 3세대 UMTS 시스템의 중요한 목표가 되고 있는 것이다[11, 12]. 이러한 목표를 달성하기 위해서는 UMTS 시스템에서 무선 단말까지 패킷 데이터 서비스를 효과적으로 제공해 줄 수 있어야 한다. 왜냐하면 멀티미디어 데이터와 인터넷 데이터 모두 패킷 데이터로 이루어져 있고 인터넷 및 멀티미디어 서비스는 이 패킷 데이터 서비스를 근간으로 제공되기 때문이다. 따라서 UMTS 시스템에서 패킷 데이터 서비스는 3세대 이동 통신 시스템의 성공 여부를 가름할 매우 중요한 역할을 할 것이다.

한편 1990년 이후 웹으로 인해 인터넷 사용이 크게 증가하면서 인터넷 트래픽도 크게 늘어 났는데 인터넷 트래픽의 대부분을 차지하는 Web, telnet, ftp 등의 응용 프로그램 트래픽은 모두 TCP(Transport Control Protocol)를 이용하고 있다. 이로 인해 인터넷 트래픽은 80% 이상이 TCP 트래픽으로 구성되어 있는데 이 추세는 인터넷 시장에서 웹이 차지하는 부분이 줄지 않는 한 앞으로도 계속되리라 예상된다[1]. 그러므로 효과적인 패킷 데이터 서비스를 제공하기 위해서는 반드시 UMTS 시스템 상에서 TCP 트래픽을 어떻게 처리하고 또 동적인 무선 환경에 의해서 TCP 성능이 어떻게 변화하는지에 알아볼 필요가 있다.

그런데 TCP는 유선 망에 최적화되어 있는 프로토콜이다. 유선 망에서는 대부분의 데이터 손실이 노드 과부화에 의한 버퍼 오버플로우로 발생한다. 따라서 TCP는 패킷이 손실되었다고 판단되면 이를 노드 과부화로 해석하고 데이터를 보내는 속도를 줄이는 메커니즘을 가지고 있다. 그러나 무선 망에서는 노드 과부화 뿐만 아니라 링크 에러로 인해 데이터가 자주 사라지게 되는데 TCP는 이를 구분할 방법이 없다. 그러므로 TCP를 무선 망에 그대로 적용할 경우 많은 성능 저하를 가져 오는데 이를 극복하기 위해 많은 연구가 진행되었고 UMTS 시스템에서는 무선 구간에서 링크 에러로 인한 데이터 손실을 막기 위해 LL MR-ARQ(Link Layer Multiple Rejects ARQ) 기법[2]을 사용한다. 하지만 LL MR-ARQ 기법이 UMTS 시스템 환경에서 무선 환경의 변화에 따라 정확히 TCP 성능에 어떤 영향을 미치는지에 대해서는 아직 연구되어진 바가 많지 않다. 무선 환경에서 TCP 트래픽 전송은 링크 에러로 인한 데이터 손실 뿐만 아니라 대역폭 변화와 같은 무선 자원의 변화, 지연 시간 변화, 무선 구간 에러율 변화 등 여러 다른 요인으로

인해서도 크게 영향 받을 수 있는데 이러한 무선 환경 변화는 TCP 성능에 직접적인 영향을 미친다. 특히 무선 대역폭 변화에 의해 TCP Timeout이 발생하게 되면 UMTS 시스템에서 증폭된 데이터 전송으로 인한 무선 자원 낭비와 TCP 성능 저하가 발생하게 된다.

따라서 본 논문에서는 UMTS 시스템 상에서 대역폭 변화를 중심으로 무선 환경 변화에 따른 TCP 성능을 실험을 통해 평가 분석해 본다. 그리고 동적인 무선 환경에서 TCP 트래픽을 효율적으로 전송할 수 있는 간단한 TCP 성능 향상 기법을 제안한다. 이 기법은 지금까지 연구된 기법에 비해 BS(Base Station)에 부담을 거의 주지 않기 때문에 UMTS 시스템에 쉽게 적용될 수 있으리라 예상된다.

본 논문의 구성은 다음과 같다. 제 2장에서는 무선 구간에서 TCP 트래픽을 전송할 때 발생하는 문제점들과 기존의 해결책들을 알아 본다. 제 3장에서는 UMTS 시스템에서 대역폭 변화와 같은 무선 환경의 변화에 의해 TCP가 어떻게 반응하는지 실험을 통해 평가 분석한다. 그리고 제 4 장에서 무선 환경이 변화할 때 TCP 트래픽을 효율적으로 전송할 수 있는 TCP 성능 향상 기법을 제안한다. 마지막으로 제 5장에서 결론을 맺고 향후 연구 계획에 대해 언급한다.

2. TCP와 UMTS 시스템

2.1 TCP와 무선 구간

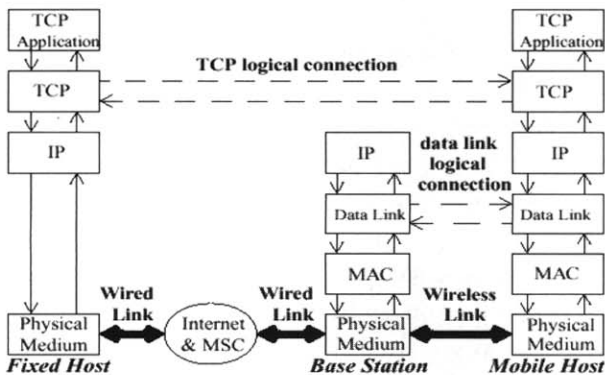
TCP sender는 패킷이 사라졌다고 판단되면 이를 링크 에러로 인한 데이터 손실로 보지 않고 중간 노드에서 Congestion이 발생한 것으로 생각하고 Congestion Window 크기를 줄임으로서 패킷 전송 속도를 떨어 드린다[3]. 이것은 유선 망에서는 링크 에러율이 매우 낮기 때문에 가능한 가정이다. 하지만 무선 망은 유선 망에 비해 매우 높은 BER을 가지고 있어서 무선 구간의 링크 에러로 인한 패킷 손실 확률이 유선 망에 비해 매우 크다. 그러므로 무선 구간이 있는 경로를 TCP 패킷이 지나가고 있을 때 무선 구간의 링크 에러로 인해 패킷이 사라진다면 TCP sender는 이를 중간 노드에서 Congestion이 발생한 것으로 오인하게 되고 패킷 전송 속도를 떨어 드리는 오동작을 하게 된다. 이러한 오동작은 TCP 전송 성능을 떨어뜨리게 되는데 무선 구간에서는 링크 에러가 빈번히 발생하므로 TCP의 전송 성능은 크게 저하 된다[4, 5].

TCP를 바꾸지 않고 무선 구간의 링크 에러로 인한 TCP 성능 저하를 막기 위한 기법으로는 Link-Layer protocol, Split-connection Protocol, Snoop Protocol 등이 있다[6, 13].

Split-connection Protocol은 TCP 연결을 유선 구간과 무선 구간을 따로 만들어 연결하는 방법이다. 즉 BS(Base

Station)에서 유선 연결과 무선 연결을 잇는 방법으로 무선 연결은 높은 BER을 감당할 수 있게 새로운 프로토콜로 구성된다. 이 방법은 무선 구간에서 데이터 전송 성능을 최대화할 수 있는 장점이 있지만 BS에서 연결마다 연결 상태를 저장하고 유지 관리해야 하고 TCP의 종단간 신뢰성 프로토콜이라는 특성이 깨어 지는 단점이 있다. 실제로 다른 기법들에 비해 그리 나은 성능을 보이지 못한다[6]. Split-connection Protocol 방법으로는 Indirect-TCP[7]가 대표적이다.

Snoop Protocol은 고정 단말에서 무선 단말까지 하나의 TCP 연결로 유지하면서 BS에서 무선 구간의 링크 에러에 의한 TCP 성능 저하를 막도록 조정하는 방법이다. BS에서 Snoop Agent는 TCP 연결을 통해 전송되는 패킷을 양방향 모두 모니터하고 있다. 그리고 고정 단말에서 BS를 통해 무선 단말까지 패킷이 전달 될 때 ACK가 전달되지 않은 패킷을 캐쉬에 저장한다. 그러다가 Duplicated ACK나 Local Timeout에 의해 패킷 손실이 발생했다고 생각되면 Snoop Agent가 그에 해당하는 패킷을 무선 단말에게 재전송하고 ACK는 없애 버리는 방법이 Snoop Protocol이다[8]. Snoop Protocol을 사용하면 불필요한 Fast Retransmission과 Congestion Control을 막을 수 있으며 무선 구간을 포함하는 TCP 연결의 성능을 크게 향상시킬 수 있다. 하지만 BS에서 Snoop Agent가 패킷을 중간에 저장, 관리해야 하고 TCP 연결마다 상태를 관리해야 하며 Transport 계층에 해당하는 TCP 헤더를 Snoop Agent가 분석해야 하기 때문에 BS가 매우 복잡해 지는 단점을 안고 있다.



(그림 1) TCP 연결을 위한 개략적인 통신 구조

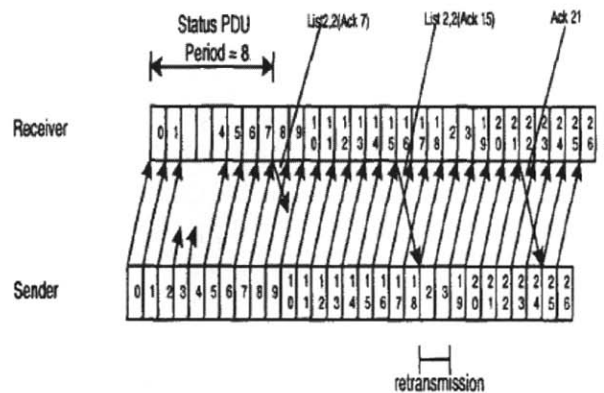
Link-Layer protocol은 상위 계층에서 무선 링크 에러를 자주 감지하지 못하도록 링크 계층에서 데이터를 재전송하거나 패킷에 대한 정보를 함께 보내서 최대한 복원하는 기법이다. Link-Layer protocol은 기존의 계층 네트워크 구조에 적합하고 상위 계층과 독립적으로 동작하며, TCP 연결마다 상태를 관리하지 않아도 되는 장점을 가진다. 즉 가장

쉽고 효과적으로 무선 TCP의 성능을 향상시킬 수 있다. Link-Layer Protocol 기법으로는 FEC(Forward Error Correction) 과 ARQ(Automatic Repeat reQuest) 방법 등이 있는데 현재 UMTS 시스템에서는 ARQ를 이용한 링크 계층 재전송 방법(Link-Layer Retransmission) 중의 하나인 LL MR ARQ(Link Layer Multiple Rejects ARQ) 기법을 사용한다. (그림 1)은 Link-Layer Protocol에서 무선 구간을 포함하는 TCP 연결을 위한 통신 구조를 개략적으로 나타낸 것이다.

2.2 UMTS 시스템에서 데이터 전송

현재 3GPP에서는 UMTS 시스템 상에서 무선 구간 TCP 성능을 향상시키기 위해서 링크 계층 재전송 기법(LL ARQ)을 사용하도록 제안하고 있다[2]. 3GPP 표준안에 따르면 UMTS 시스템에서 RLC(Radio Link Control) 프로토콜은 AM 모드(acknowledged mode)를 사용할 경우 In-sequence Delivery나 Out-of-sequence Delivery를 수행할 수 있고 Selective Repeat, Go Back N, Stop-and Wait ARQ[14, 15] 등의 재전송을 사용한 에러 보정 기능을 제공해 데이터 전송을 보장한다. 만약 RLC에서 데이터 전송이 불가능 할 경우에는 상위 계층에 전송 실패를 알려 준다.

3GPP 표준에서는 UMTS 시스템에서 어떠한 종류의 링크 계층 재전송 기법을 사용하라고 구체적으로 지정해 놓지는 않았다[2]. 하지만 보편적으로 무선 구간에서는 여러 가지 재전송 기법 중에서도 SR(Selective Repeat) ARQ이 좋은 성능을 나타낸다. 특히 받지 못한 PDU에 대한 정보를 몰아 두었다가 주기적으로 하나의 STATUS PDU에 실어 보내는 LL MR ARQ(Link Layer Multiple Rejects ARQ) 기법은 무선 자원 까지 고려한 효과적인 방법으로 UMTS 시스템에서 보편적으로 사용될 수 있는 링크 계층 재전송 기법이다. (그림 2)는 MR ARQ 방법의 예를 보여 주고 있다[9].



(그림 2) Multiple Reject ARQ Operation[9]

지금까지 TCP를 무선 구간에 의해 발생하는 TCP의 문제점과 무선 구간의 신뢰성을 증가 시켜서 TCP 성능을 향상시키는 기존 연구에 대해 살펴 보았다. 또 마지막으로 현재 UMTS 시스템에서 사용하는 링크 계층 재전송 기법의 동작 방법에 대해 알아 보았다. 하지만 기존의 연구는 무선 구간의 높은 링크 에러율에 의한 데이터 손실로 발생하는 TCP 성능 저하만을 최소화하기 위해 제안되어졌다. 즉 사용자 변화, 무선 구간 에러율 변화, 대역폭 변화 등 동적인 무선 환경 변화에 따른 TCP 성능 변화에 대한 연구는 거의 진행되어지지 않고 있다. 따라서 본 논문에서는 UMTS 시스템에서 LL MR ARQ를 사용할 때 대역폭 변화로 대표되는 동적인 무선 환경 변화가 TCP 성능에 미치는 영향을 실험을 통해 분석하고 이를 개선할 수 있는 방법을 제안한다.

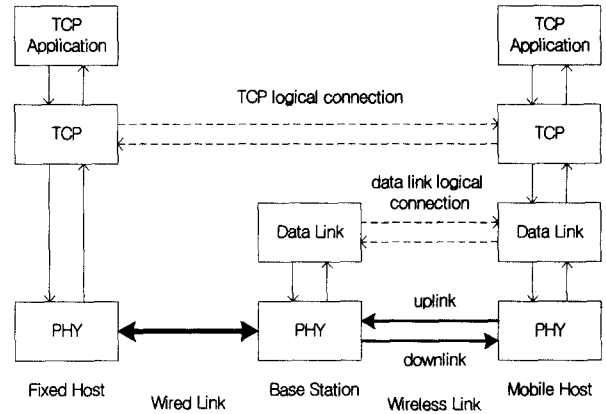
3. UMTS 시스템 상에서 무선 환경 변화에 따른 TCP 성능 분석

무선 환경에서 TCP 트래픽 전송은 링크 에러로 인한 데이터 손실 뿐만 아니라 대역폭 변화와 같은 무선 자원의 변화, 지연 시간 변화, 무선 구간 에러율 변화 등 여러 다른 요인으로 인해서도 크게 영향 받는다. 특히 무선 환경 변화는 대역폭의 변화로 대표될 수 있는데 이러한 대역폭 변화는 TCP 트래픽 전송에 많은 영향을 미친다. UMTS 시스템의 무선 구간에서 사용자에게 제공해 줄 수 있는 대역폭은 제한적이며 유선 망에 비해 크게 부족하다. 특히 서비스 중간에 사용자의 증가와 같은 무선 환경 변화로 인한 대역폭 감소는 TCP로 하여금 Timeout을 발생시킬 수 있다. Timeout이 발생하면 TCP는 Timeout이 발생한 데이터부터 다시 재전송하게 되는데 이러한 중복된 데이터 전송은 값비싼 무선 자원의 낭비를 가져 온다. 따라서 무선 환경이 변화하더라도 무선 자원의 낭비 없이 효율적으로 TCP 트래픽을 전송하는 기법이 절실히 요구 된다.

그러나 지금까지 무선 TCP에 관한 연구는 링크 에러로 인한 데이터 손실을 보상하는 방법에 관해서만 치중 되어 있었고 무선 환경 변화에 따른 효율적인 TCP 트래픽 전송에 대해서는 거의 진행되어진 것이 없다. 따라서 본 장에서는 UMTS 시스템에서 대역폭 변화와 같은 무선 환경의 변화에 의해 TCP가 어떻게 반응하는지 실험을 통해 평가 분석한다.

3.1 실험 모델

실험에 사용한 시스템 환경은 (그림 3)과 같다.



(그림 3) 실험에 사용된 Internet Service를 위한 UMTS 시스템 구조

실험 모델에서 사용한 변수 값은 <표 1>과 같다.

<표 1> 실험 모델에서 사용한 변수 값

Parameters		Values
TCP Version		Reno
TCP Receiver Window Size		4096, 16384Bytes
Initial TCP Timeout value		3s
TCP Packet Size (MSS + Header)		(512 + 40)Bytes, (1024 + 40)Bytes
Wired Link	Propagation Delay	10ms
	Bandwidth	10Mbps
	Error Rate	0%
Wireless Link	Propagation Delay	10ms, 80ms
	Bandwidth	64Kbps~2Mbps
	FER (downlink)	10%, 30%
	FER (uplink)	0%
	Delivery Sequence	In Sequence Delivery

UMTS 시스템 상에서 TCP 성능 실험을 위해 이벤트 기반 시뮬레이터인 NIST Network Simulator[10]를 사용하였다. TCP 버전은 Reno를 사용하였다. TCP Receiver Window 크기는 환경에 따라 4096이나 16384Bytes를 사용하였다. 4.3BSD에서는 4096을 사용하고 있으나 연구 결과에 의하면 지금과 같은 인터넷 유선 망 환경에서는 16384Bytes로 할 경우 TCP 성능이 더 좋아진다고 되어 있고 Solaris 2.2, 4.4BSD, AIX 3.2 등에서는 실제로 4096보다 큰 8192, 16384 등을 사용하고 있다[3]. 유선 구간에 10ms 지연 시간을 두었다. 10ms는 매우 큰 대도시를 포함할 수 있을 정도의 지연 시간이다. 무선 구간에서는 10ms와 80ms 지연 시간을 두었다. UMTS 시스템에서는 한 프레임의 최소 길이가 10ms이고 interleaving에 따라 80ms까지 지연 시간을 가질 수 있기 때문이다. 무선 대역폭은 최고 2Mbps에서 최소 64Kbps로 가정하였다.

3.1.1 링크 계층 모델(LL MR-ARQ)

UMTS 시스템에서 링크 에러에 의한 데이터 손실을 막기 위해서 LL MR-ARQ 기법을 사용한다. LL MR-ARQ 기법은 다음과 같은 3가지 경우 송신자 LL(Link Layer)에게 ACK에 해당하는 STATUS Report를 보낸다[2].

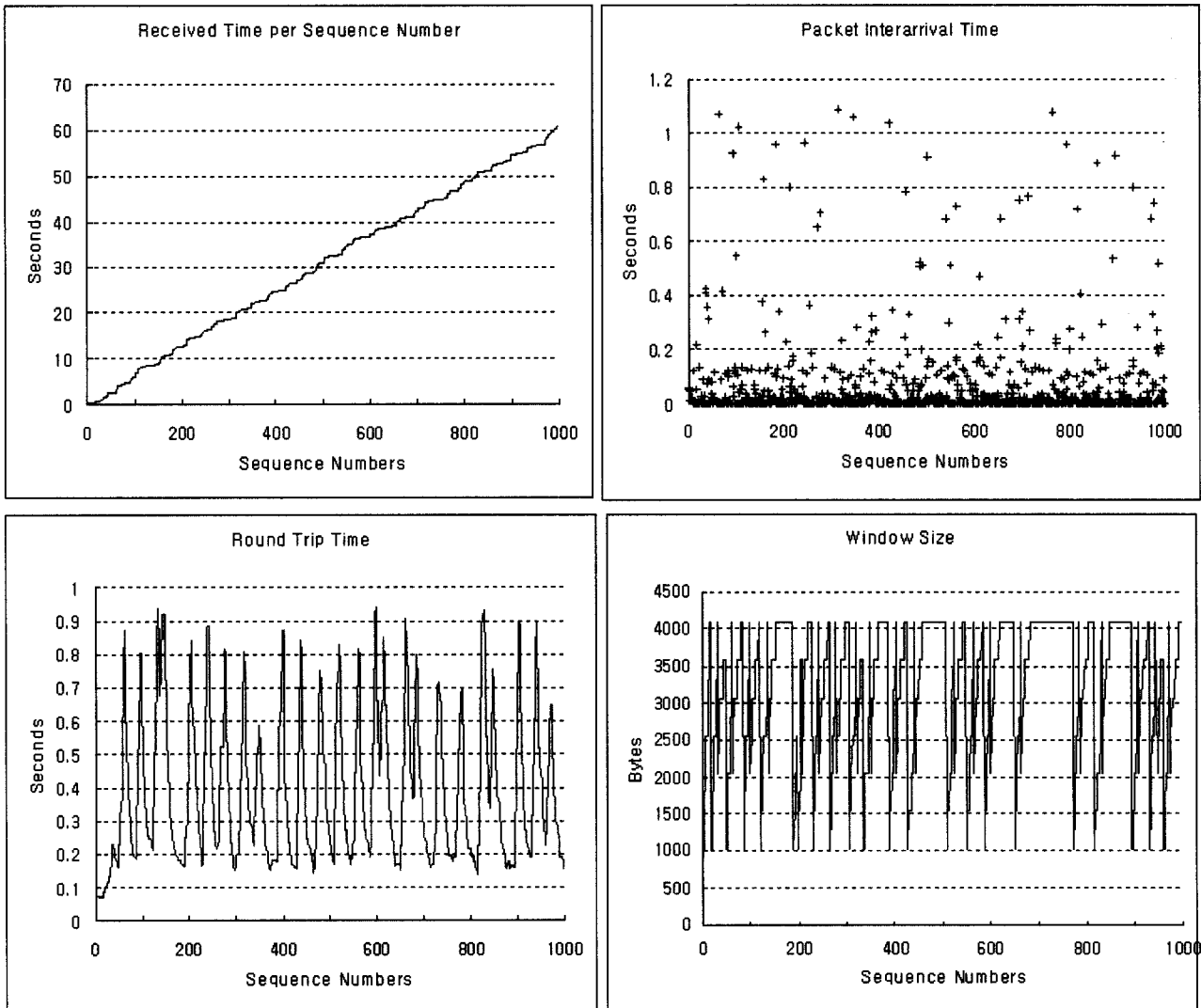
- Detection of missing data
- Timer based STATUS transfer
- The EPC(Estimated PDU Counter) mechanism

즉 링크 에러로 인한 데이터 분실을 감지하거나 마지막 STATUS Report를 보내고 일정 시간이 지났다면 STATUS Report를 보낸다. 그리고 EPC Mechanism을 통해 재전송되고 있는 데이터에 의한 STATUS Report 발송을 방지한다. 그리고 다음과 같은 경우 STATUS Report를 보내지 않게 한다[2].

- STATUS prohibit
- The EPC Mechanism

즉 한번 STATUS Report를 보냈다면 일정 시간 동안은 STATUS Report를 보내지 않게 함으로써 불필요하게 중복되는 데이터 전송을 방지한다. 이렇듯 LL MR-ARQ 기법은 무선 자원을 최대한 효율적으로 사용하려고 노력하는데 하는데 이 중 보편적으로 발생하는 STATUS Report 전송은 Timer based STATUS 전송이라 볼 수 있다 [9].

본 논문에서는 UMTS 시스템에서 사용하는 LL MR-ARQ 기법을 모사하기 위해 Optimized Timer based ACK 전송 방법을 사용하였다. 즉 실험에서는 무선 단말과 기지국 사이의 거리가 고정되어 있으므로 정확한 지연 시간(propagation delay)을 알 수 있다. 이를 통해 최적화된 주기로 ACK를 보내는 방법이 Optimized Timer Based ACK 전송



(그림 4) d10, e0.1, bw384_64, rwnd4096, mss512, r1

이다. 실험에서는 이 Optimized Timer Based ACK 전송을 통해 MR-ARQ 기법을 모사하였다. 실험에서 ACK transfer 주기는 무선 구간의 지연 시간의 3배 - Round Trip Time의 1.5배 - 로 하였다.

3.2 무선 자원 변화에 따른 실험 결과

3.2.1 대역폭 변화 주기에 따른 실험 결과

(그림 4)는 무선 지연 10ms, 무선 구간 FER 10%, TCP MSS 512Bytes, TCP Receiver Window Size 4096Bytes에서 무선 대역폭이 384Kbps와 64Kbps를 반복해서 변화할 때 그 반복 주기를 1초로 놓고 1000개의 TCP 패킷을 받는 실험의 결과이다.

TCP 수신자가 1000개의 패킷을 받는데 RT(Received Time)는 60.5초가 걸렸다. PIT(Packet Interarrival Time)는 평균 60.4ms였고 최고치는 1074.3ms였다. RTT(Round Trip Time)는 평균 391.0ms였고 최고치는 943.5ms였다. WS(Window Size)는 TCP Timeout이 일어날 때마다 1024로 떨어지고 Fast Retransmit이 일어날 때마다 그 전 WS의 절반으로 떨어졌다. 실험 결과 평균 69.3Kbps의 TCP Throughput을 보여 주었다. 그리고 RT 증가 폭이 대역폭의 변화에 따라 달라지는 것을 알 수 있다. 또 WS의 변화를 통해서 21번의 TCP Timeout이 일어났음을 알 수 있다. 마지막으로 대역폭이 바뀔 때마다 RTT값과 PIT의 변화 폭이 매우 큰 것도 알 수 있다. 이렇듯 대역폭의 변화는 TCP 성능에 많은 영향을 미친다. 특히 TCP Timeout이 발생하게 되면 TCP는 과거에 보낸 데이터를 다시 재전송하게 되어 무선 자원의 낭비를 가져 오게 되는 단점을 지니게 된다.

<표 2>는 이전 실험과 같은 환경에서 대역폭 변화 주기를 바꾸어가며 실험한 결과이다. <표 2>의 ②는 이전 실험과 같은 환경에서 반복 주기를 0.1초로 바꾸어 놓고 실험한 결과이다. RT는 52.2초가 걸렸다. PIT는 평균 52.1ms였고 최대치는 552.0ms였다. RTT는 평균 403.3ms였고 최고 714.0ms까지 증가 하였다. WS는 한 번 4096Bytes에 도달한 후 떨어지지 않았다. 이를 통해 우리는 대역폭 주기가 아주 짧으면 무선 링크의 대역폭 변화에 의해 발생하는 무선 지연 시간의 변화를 TCP가 Timeout을 발생시키지 않고 포용할 수 있음을 알 수 있다. 실험 결과 평균 TCP Throughput은 78.5Kbps였다.

<표 2>의 ③은 이전 실험과 같은 환경에서 반복 주기를 10초로 바꾸어 놓고 실험한 결과이다. RT는 50.0초가 걸렸다. PIT는 평균 50.0ms였고 최고 7958.3ms였다. RTT는 평균 387.2ms였고 최고 8175.4ms까지 증가 하였다. WS는 대부분 4096Bytes를 유지했으며 TCP Timeout은 3번 일어났다. 실험 결과 평균 83.9Kbps의 TCP Throughput을 보여 주었다. 대역폭 변화 주기를 늘리면 TCP Timeout이 대역폭이 변화할 때만 발생하므로 발생 횟수가 줄어 든다. 그리고 줄어든 만큼 TCP 성능은 나아진다. 하지만 대역폭이 변화할 때 PIT와 RTT의 최고 값이 크게 증가됨을 알 수 있다. 이는 수신자 WS 4096일 때 대역폭이 64Kbps일 경우 대역폭에 비해 WS가 너무 커서 BS에서 데이터가 누적되어 쌓이기 때문이다.

지금까지 대역폭 변화에 따른 TCP 성능을 실험해 보았다. 실험 결과 대역폭 주기 변화가 아주 짧으면 하나의 TCP 패킷이 전송되는 동안 여러 번의 대역폭 변화가 일어나게 되고 그 변화에 의한 영향이 RTT 값에 반영된다. 따라서 TCP Timeout이 발생되지 않는다. 그러나 대역폭 변화 주기가 커지면 대역폭이 줄어 링크 환경이 악화될 때 TCP는 급격한 영향을 받게 된다. 이로 인해 TCP는 급격한 환경 변화에 적응하지 못하고 TCP Timeout을 발생시킨다. 하지만 대역폭 변화 주기가 매우 길어지면 일정 시간 동안 무선 환경이 변화되는 횟수가 줄게 되어 TCP Timeout이 발생하는 빈도 수가 줄어들게 되고 그에 따라 전체적인 TCP 성능도 나아진다. 그러므로 대역폭 변화 주기가 대역폭 축소로 인한 무선 환경 악화를 TCP가 수용할 수 없을 정도로 큰 것들 중에서 가장 작을 때 TCP 성능은 가장 나빠진다.

3.2.2 대역폭 변화에 따른 실험 결과

다음 실험들은 무선 지연 10ms, 무선 구간 FER 10%, TCP MSS 512Bytes, TCP Receiver Window Size 16384Bytes, 무선 대역폭 2Mbps인 환경에서 1초 주기로 무선 대역폭이 좋을 때와 나쁠 때 실험한 결과들이다. <표 3>은 대역폭 변화를 바꾸어가며 실험한 결과들을 보여 준다.

<표 3>의 ①은 무선 환경이 나쁠 때 무선 대역폭이 384 Kbps인 경우, TCP 패킷을 1000개 보내는 실험의 결과이다. RT는 10.7초가 걸렸다. PIT는 평균 10.7ms였고 최고

<표 2> 대역폭 변화 주기에 따른 실험 결과

	RT (sec)	PIT (msec)		RTT (msec)		Throughput (Kbps)
		Avg	Max	Avg	Max	
① d10, e0.1, bw384_64, rwnd4096, mss512, r1	60.5	60.4	1074.3	391.0	943.5	69.3
② d10, e0.1, bw384_64, rwnd4096, mss512, r0.1	52.2	52.1	552.0	403.3	714.0	78.5
③ d10, e0.1, bw384_64, rwnd4096, mss512, r10	50.0	50.0	7958.3	387.2	8175.4	83.9

〈표 3〉 대역폭 변화에 따른 실험 결과

	RT (sec)	PIT (msec)		RTT (msec)		WS (Bytes)	Throughput (Kbps)
		Avg	Max	Avg	Max		
① d10, e0.1, bw2048_384, rwnd16384, mss512, r1	10.7	10.7	627.6	160.5	721.6	8925.4	392.0
② d10, e0.1, bw2048_144, rwnd16384, mss512, r1	14.5	14.5	770.8	201.1	1046.1	7741.9	289.3
③ d10, e0.1, bw2048_64, rwnd16384, mss512, r1	18.0	18.0	1058.5	207.5	1172.8	6641.7	233.0

627.6ms까지 증가하였다. RTT는 평균 160.53ms였고 최고 721.6ms까지 증가 하였다. WS는 평균 8925.4Bytes였다. 실험 결과 TCP Throughput은 평균 392.0Kbps였다. WS는 최대치로 증가한 후 TCP Timeout이 생길 때마다 급격히 떨어졌는데 한 번 떨어진 이후엔 최대치로 증가하지 전에 TCP Timeout이 발생하여 다시 떨어지는 모습을 보여 주고 있다.

〈표 3〉의 ②는 무선 환경이 나쁠 때 무선 대역폭이 144 Kbps인 경우 실험한 결과이다. RT는 14.5초가 걸렸다. PIT는 평균 14.5ms였고 최고 770.8ms까지 증가하였다. RTT는 평균 201.1ms였고 최고 1046.1ms까지 증가 하였다. 실험 결과 TCP Throughput은 평균 289.3Kbps였다. WS는 평균 7741.9Bytes였다. PIT와 RTT의 평균 값과 최고 값이 이전 실험에 비해 늘어 난 것을 알 수 있다. 이것은 대역폭이 줄어들었을 때 보낼 수 있는 데이터 양이 이전 실험보다 줄어 그 동안 데이터가 전송되지 않기 때문이다. WS는 이전 실험과 마찬가지로 최대치로 증가한 후 TCP Timeout이 생길 때마다 급격히 떨어졌는데 한 번 떨어진 이후 다시 떨어지기까지 구간이 짧아 졌다. 그 이유는 환경이 나쁠 때 대역폭이 많이 감소했기 때문에 대역폭이 좋아진 다음부터 다시 감소할 때까지 받는 ACK의 개수가 이전보다 작아 지기 때문이다. ACK의 수가 작으면 WS는 충분히 회복하기 전에 다시 떨어진다.

〈표 3〉의 ③은 무선 환경이 나쁠 때 무선 대역폭이 64Kbps인 경우 실험 결과이다. RT는 18.0초가 걸렸다. PIT는 평균 18.0ms였고 최고 1058.5ms까지 증가하였다. RTT는 평균 207.5ms였고 최고 1172.8ms까지 증가 하였다. WS는 평균 6641.7Bytes였다. TCP Throughput은 233.0Kbps였다. 환경이 나쁠 때 무선 대역폭이 더욱 줄어들었으므로 이전 실험에 비해 더욱 나쁜 TCP 성능을 보여 주고 있다. PIT와 RTT의 최고 값도 가장 컸고 WS의 평균도 가장 작았다. 이것은 대역폭이 줄어들었을 때 TCP 패킷 전송량이 아주 작기 때문이다. 그리고 무선 환경이 개선되어 대역폭이 다시 회복되더라도 대역폭이 나쁠 때 전송되었던 패킷 수가 작아서 돌아오는 ACK의 양이 줄어들고 줄어드는 ACK 양은 송신자에서 시간 당 TCP가 전송하는 패킷 수를 줄이기 때문이다.

지금까지 무선 환경이 악화될 때 가질 수 있는 무선 대

역폭을 변화시키면서 실험해 보았다. 〈표 2〉는 지금까지의 실험에서 나온 결과를 표로 정리한 것이다. 실험에서 알 수 있듯이 환경이 심하게 악화되어 무선 대역폭이 크게 줄어들 수록 낮은 TCP 성능을 보였다. 그 이유는 첫째로 무선 대역폭이 심하게 줄 수록 줄어들었을 때 전송할 수 있는 데이터량이 줄어들었기 때문이고 둘째로 무선 대역폭이 다시 회복된 직후 보낼 수 있는 데이터량이 이전에 대역폭이 줄어들었을 때 보낸 데이터량에 영향을 받기 때문이다. 즉 줄어드는 시간 동안 보낸 패킷이 적으면 대역폭이 다시 회복된 직후 송신자가 받을 ACK 수가 줄게 되고 이 때문에 회복된 대역폭을 충분히 사용해 데이터를 보내는데 시간이 걸리기 때문이다.

3.2.3 무선 자원 변화에 따른 TCP 성능 전체 분석

첫째로 무선 대역폭 변화 주기가 변할 때 TCP 성능이 어떻게 변화되는지 실험하였다. 이 경우 무선 환경의 변화가 매우 빨라서 하나의 TCP 패킷이 전송되는 동안 대역폭이 여러 번 바뀌게 되면 그 변화가 TCP RTT 값에 반영되고 RTT 값을 바탕으로 RTO 값을 정하는 TCP mechanism에 의해 TCP Timeout이 발생되지 않는다. 하지만 대역폭 변화 주기가 일정 간격 이상이 되면 하나의 패킷에 그 변화가 모두 반영되지 못하고 대역폭이 악화되는 순간 데이터 전송이 이전처럼 빠르게 되지 못한다. 이로 인해 일정 시간 안에 TCP 패킷이 전송되지 못하면 TCP ACK가 TCP RTO 값 내에 송신자로 도달하지 못하고 결국 TCP Timeout이 발생하게 된다. TCP Timeout이 발생하면 TCP 성능은 크게 떨어진다. 한편 대역폭 변화 주기가 계속 커진다면 단위 시간 안에 대역폭이 악화되는 시점이 발생하는 빈도가 줄고 이로 인해 TCP Timeout이 발생하는 빈도도 줄어 든다. 따라서 대역폭 변화 주기가 TCP Timeout이 발생시킬 만큼 충분히 크면서 가장 작은 값을 가질 때 TCP 성능은 가장 많이 저하된다.

둘째로 무선 환경 악화로 무선 대역폭이 줄어들 때 그 폭이 변화가 TCP 성능에 어떠한 영향을 미치는지 실험해 보았다. 실험 결과 무선 환경의 악화되는 폭이 클 수록 TCP 성능이 더욱 더 떨어지는 것을 알 수 있었다. 그 이유는 첫째, 무선 대역폭이 줄때 크게 줄어들면 그 시간 동안 보낼 수 있는 데이터량이 적기 때문이고 둘째, 다시 무선 환경이

좋아져서 대역폭이 늘어난 직후 TCP가 대역폭을 충분히 사용하기 위해서는 시간이 필요한데 그 시간이 대역폭이 줄었다가 늘어난 폭이 클 수록 더 오래 걸리기 때문이다.

지금까지 무선 환경의 변화로 인한 대역폭 변화가 TCP 성능에 어떠한 영향을 미치는지 실험을 통해 분석해 보았다. 다음 장에서는 무선 환경이 변화할 때 TCP 트래픽을 효율적으로 전송할 수 있는 간단하면서 UMTS 시스템에 쉽게 적용할 수 있는 TCP 성능 향상 기법을 제안한다.

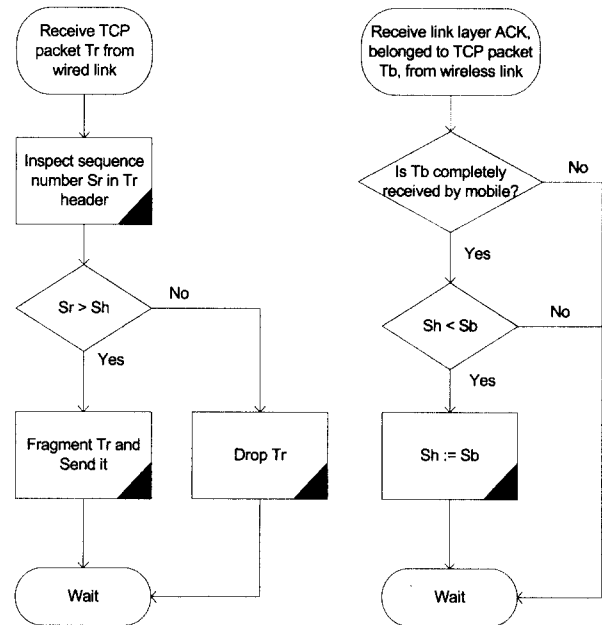
4. 효율적 TCP 트래픽 전송을 위한 TCP 성능 향상 기법

무선 환경 변화 등으로 인해 TCP Timeout이 발생하게 되면 송신자는 Timeout이 발생한 TCP 패킷부터 다시 재전송하게 된다. 하지만 BS가 데이터를 버리지 않는 한 재전송되는 패킷들은 이미 단말로 전송되었거나 BS 버퍼에 저장되어 전송되기를 기다리고 있다. 그러므로 재전송된 패킷을 다시 단말로 보내는 것은 값 비싼 무선 자원을 이용해 중복된 데이터를 전송하는 낭비를 가져 온다.

무선 구간에서 이러한 TCP 성능 저하를 막기 위한 기법들은 이미 많이 연구 되어져 왔다. 하지만 대부분의 기법이 BS에 너무 많은 것을 유지 관리하기를 원하거나 비현실적으로 TCP 수신자를 바꾸려는 접근을 취하고 있기 때문에 실제 시스템에 적용하기는 거의 불가능하다[6]. 다시 말해서 BS에 많은 부담을 주지 않고서는 무선 자원 변화에 따른 TCP Timeout 발생은 막을 수 없다.

따라서 본 장에서는 TCP Timeout이 발생했을 때 성능 저하를 최대한 막고 무선 자원을 절약하여 TCP 트래픽을 효율적으로 전송할 수 있는 TCP 성능 향상 기법을 제안한다. 이 기법은 BS에 많은 부담을 주지 않기 때문에 실제 UMTS 시스템에 쉽게 적용될 수 있으리라 생각한다. 이 기법의 동작 방식은 다음과 같다. 우선 BS에 데이터가 도착하면 이 것이 TCP 패킷인지 아닌지 판단하고 아니라면 그대로 전송한다. 그리고 만약 TCP 패킷이 도달했다면 이 패킷의 Sequence Number가 이전에 BS가 보낸 경험이고 수신자가 잘 받았던 패킷 중 가장 큰 Sequence Number보다 큰지 판단한다. 크다면 전송한다. 그러나 작거나 같다면 재전송되는 패킷이므로 BS에서 패킷을 버린다. 즉 이미 수신자가 받은 데이터라면 BS에서 무선 자원을 사용해서 전송하지 않고 미리 걸러 내는 것이다. 이러한 TCP 성능 향상 기법을 사용하게 되면 TCP Timeout으로 인해 발생하는 중복된 데이터 전송을 무선 구간에서 막을 수 있게 되어 무선 자원을 절약할 수 있다. 그리고 재전송되는 TCP 패킷이 단말에 도착하지 않으므로 단말은 중복된 ACK를 보내지 않게 된다. 그 결과 송신자에서 중복된 ACK에 의

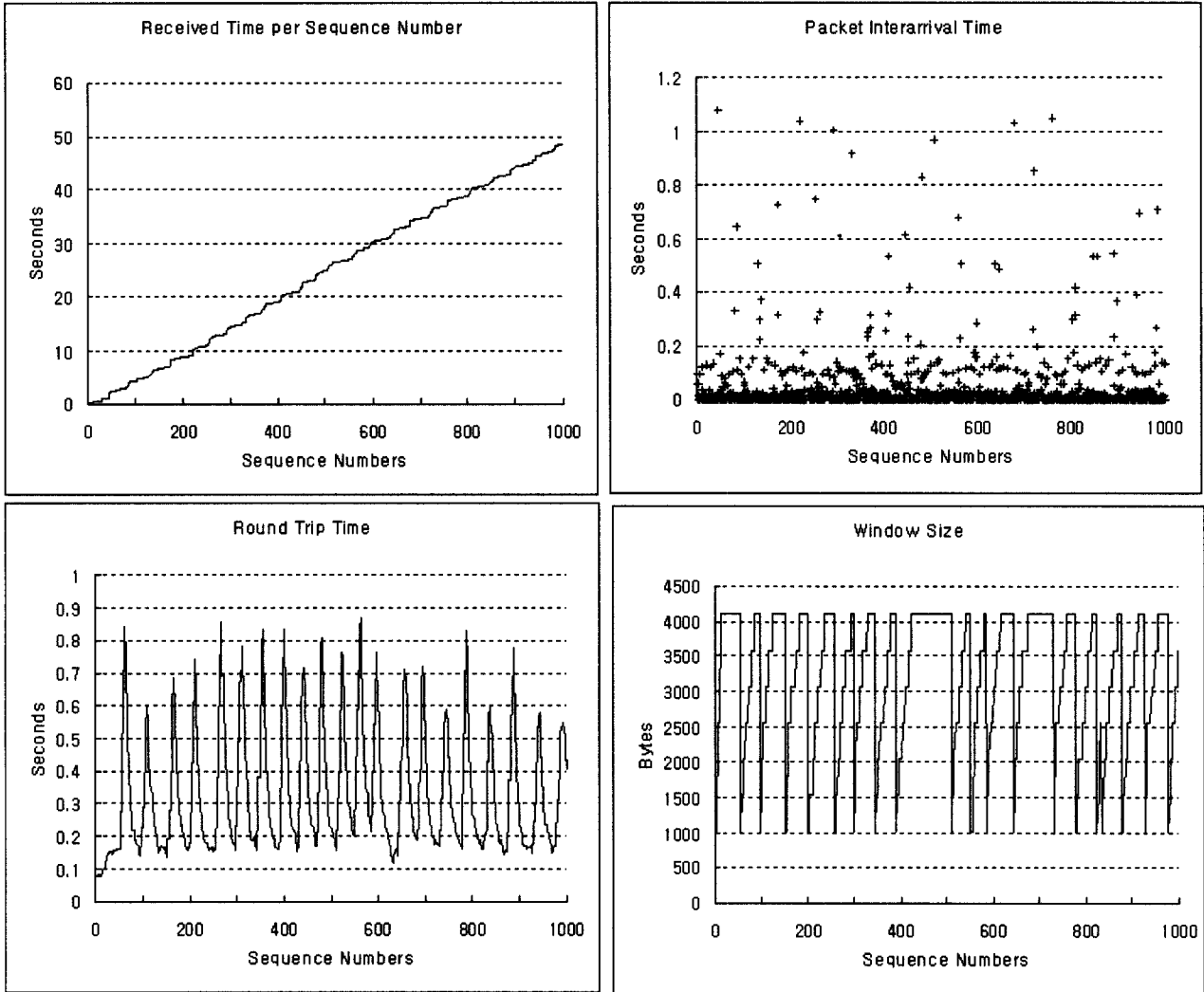
한 Fast Retransmit이 발생하지 않아 불필요하게 TCP 수신자 측의 Congestion Window를 감소시키지 않게 되어 TCP 성능을 향상시킬 수 있다. 다만 BS에서 TCP 헤더의 Sequence Number를 보고 그 중 가장 높은 것을 저장해야 하는 부담은 있다. (그림 5)는 간단한 TCP 성능 향상 기법에 대한 알고리즘을 도식적으로 나타낸 것이다.



(그림 5) BS에서 TCP 성능 향상을 위한 기법

4.1 TCP 성능 향상 기법을 도입한 후 실험 결과

(그림 6)은 무선 지연 10ms, 무선 구간 FER 10%, TCP MSS 512Bytes, TCP Receiver Window Size 4096Bytes에서 무선 대역폭이 384Kbps와 64Kbps를 1초를 주기로 반복해서 변화할 때 TCP 성능 향상 기법을 도입해 1000개의 TCP 패킷을 받는 실험의 결과이다. TCP 수신자가 1000개의 패킷을 받는데 RT는 48.5초가 걸렸다. PIT는 평균 48.5ms였고 최고치는 1076.9ms였다. RTT는 평균 333.8ms였고 최고치는 872.6ms였다. WS는 TCP Timeout이 일어날 때마다 1024로 떨어지고 Fast Retransmit은 일어나지 않았다. 실험 결과 평균 84.5Kbps의 TCP Throughput을 보여주었다. TCP 성능 향상 기법을 적용하지 않았을 경우 TCP Throughput이 69.3Kbps 정도로 측정되었으므로 이 기법을 적용하면 약 20% 정도의 성능 향상을 가져 온다는 것을 알 수 있다. PIT와 RTT 값도 이 기법을 적용할 때 조금 더 작게 나오는 것을 알 수 있다. 그리고 이 기법을 적용하지 않았을 경우 중복된 TCP 패킷 수신으로 인해 수신자에서 중복 TCP ACK를 발송하고 송신자에서 이 중복 ACK로 인해 Fast Retransmit에 의한 WS 감소 현상이 있었다. 하지만 TCP 성능 향상 기법을 적용하면 TCP 패킷을 BS

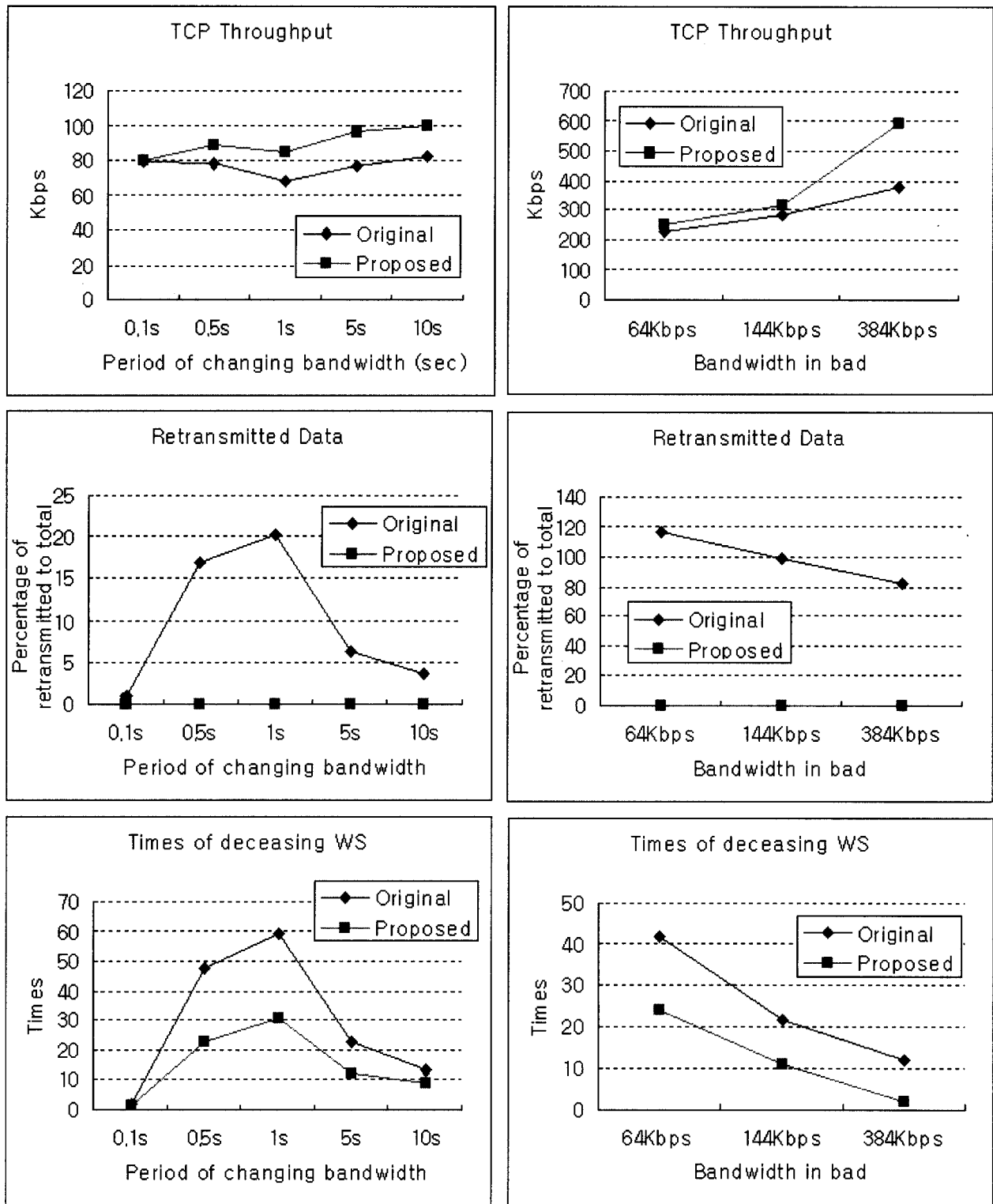


(그림 6) d10, e0.1, bw386_64, rwnd4096, r1에서 TCP 성능 향상 기법을 적용한 후 실험 결과

에서 재전송하지 않으므로 TCP Timeout 이후 중복된 ACK에 의한 Fast Retransmit이 발생하지 않는다.

(그림 7)은 대역폭 변화 주기와 줄어드는 대역폭의 크기를 변화시켜 가면서 실험한 결과이다. TCP Throughput은 받은 데이터량을 받은 시간으로 나눈 것으로 TCP 성능을 알려 주는 지표이다. 실험에서 대역폭 변화 주기를 어떻게 바꾸어도 TCP 성능 향상 기법을 적용할 경우 더 좋은 성능을 보였다. 특히 대역폭 변화 주기가 적당히 큰 경우 가장 좋은 성능을 보였는데 이는 대역폭 변화 주기가 TCP mechanism이 적용 못할 만큼 적절히 큰 값을 가질 때 TCP Timeout이 가장 많이 발생하기 때문이다. 또 감소 대역폭 크기를 바꾸어 가며 실험했을 때 감소 대역폭 크기가 작을 수록, 즉 감소되는 대역폭 값이 클 수록 TCP 성능 향상 기법의 성능 개선 효과가 크게 나타났다. TCP 성능 개선은 단말에 재전송된 데이터량과 WS가 줄어드는 횟수를 통해서도 알 수 있다. TCP 성능 향상 기법을 적용하지 않을 경

우 단말에 재전송되는 데이터량은 대역폭 변화 주기가 적절히 클 때, 그리고 감소 대역폭 크기가 작을 때 가장 많았다. 이에 반해 TCP 성능 향상 기법을 적용하게 되면 BS에서 재전송되는 데이터는 제거하게 됨으로 단말에 중복되어 전송되는 데이터는 존재하지 않게 된다. WS 크기가 줄어드는 횟수로도 TCP 성능을 측정할 수 있는데 TCP 성능 향상 기법을 적용할 경우 적용하지 않은 경우에 비해 대략 1/2 정도로 WS가 줄어드는 횟수가 준다. 그 이유는 단말에 중복된 데이터를 전송하지 않기 때문에 단말이 중복된 TCP ACK를 만들지 않고 송신자에서 중복된 ACK로 인한 Fast Retransmit이 일어나지 않기 때문이다. Fast Retransmit이 일어나게 되면 WS는 TCP Reno는 이전 값의 절반으로 TCP Tahoe는 2 패킷 크기로 떨어진다. 이상의 결과를 통해 우리는 무선 환경의 변화로 인해 대역폭이 변화할 때 TCP 성능 향상 기법을 사용하면 TCP 성능이 크게 개선됨을 알 수 있다.



(그림 7) 대역폭 변화 주기와 감소 대역폭 크기에 따른 실험 결과

5. 결론 및 향후 계획

지금까지 UMTS 시스템에서 무선 자원 변화에 따른 TCP 성능 변화를 분석해 보았다. 또 BS에 많은 부담을 주

지 않으면서도 TCP 성능을 적절히 향상시킬 수 있는 간단한 TCP 성능 향상 기법을 제안하고 이 성능을 실험을 통해 분석해 보았다.

실험을 통해 무선 환경 변화에 의해 대역폭이 변화될 경

우 UMTS 시스템에서 TCP 성능이 적지 않게 저하되는 것을 알 수 있었다. 특히 변화 주기가 TCP Timeout mechanism이 적용할 수 없을 정도로 크고 동시에 그 주기가 짧을 수록 TCP Timeout이 빈번히 발생하여 TCP 성능을 크게 저하시켰다. 그리고 상태가 좋지 않을 때의 대역폭의 크기가 작을 수록 TCP 성능이 많이 떨어졌다. 이러한 동적인 무선 환경에서 TCP Timeout은 BS에 많은 부담을 주지 않고서는 피할 수 없는 일이다. 따라서 TCP Timeout으로 인한 TCP 성능 감소를 최소한으로 줄일 수 있는 간단한 TCP 성능 향상 기법을 제안했다. 즉 TCP 송신자가 중복된 데이터를 전송했을 때 이를 BS에서 제거하는 기법을 제안했는데 이 방법을 사용할 경우 값 비싼 무선 자원을 사용해 중복된 데이터를 불필요하게 보내는 것을 방지 할 수 있었다. 그리고 단말에서 중복된 패킷 수신을 막음으로써 송신자의 TCP Congestion Window가 불필요하게 감소하는 것을 막아 효율적인 TCP 트래픽 전송이 가능케 하였다. 하지만 이 기법은 BS에서 TCP 헤더에서 Sequence Number를 관찰해야 하고 그 중 가장 높은 값을 유지해야 하는 부담이 따른다. 그러나 이 기법은 하나의 Sequence Number만을 유지하기 때문에 Snoop[8] 처럼 BS에 많은 부담을 요구하지 않고 TCP 성능을 개선시킬 수 있어서 실제 UMTS 시스템에 적용하기에 적절하다고 판단된다. 실제로 무선 구간의 TCP 성능 향상을 위해 많은 연구가 있었으나 대부분 BS에 너무 많은 부담을 주거나 비현실적으로 TCP 송신자 프로토콜 변경을 요구하기 때문에 현실에 적용하기가 거의 불가능하였다.

따라서 앞으로는 본 논문에서 제안한 TCP 성능 향상 기법을 실제 UMTS 시스템에 적합하도록 구현해 볼 예정이다. 특히 비정상적인 상태나 Handoff와 같은 상황에서 제안된 방법을 어떻게 적용해야할지 연구해 볼 계획이다.

참 고 문 헌

- [1] Kevin Thompson, Gregory J. Miller and Rick Wilder, "Wide-Area Internet Traffic Patterns and Characteristics," IEEE Network, Nov./Dec., 1997.
- [2] 3GPP, Technical Specification Group Radio Access Network, "RLC Protocol specification," 3G TS 25.322 version 3.0, October, 1999.
- [3] W. Recharad Stevens, "TCP/IP Illustrated, Volume I," Addison-Wesley, 1996.
- [4] Ramon Caceres and Liviu Iftode, "Improving the Performance of Reliable Transport Protocols in Mobile Computing Environments," IEEE Journal on Selected Areas in Communications, Vol.13, June, 1995.
- [5] T. V. Lakshman and Upamanyu Madhow, "The Performance of TCP/IP for Networks with High Bandwidth-Delay Products and Random Loss," IEEE/ACM Transactions on Networking, Vol.5, No.3, June, 1997.
- [6] Hari Balakrishnan, Venkata N. Padmanabhan, Srinivasan Seshan and Randy H. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links," IEEE Transactions on Networking, Vol.5, No.6, Dec., 1997.
- [7] A. Bakre and B. R. Badrinath, "I-TCP : Indirect TCP for mobile hosts," In Proceedings of the 15th International Conference on Distributed Computing Systems, Vancouver, BC, pp.136-143, May, 1995.
- [8] H. Balakrishnan, S. Seshan and R. H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," ACM Wireless Networks, Vol.1, Dec., 1995.
- [9] Yooyeol Yoon, Seongssoo Park and Paul S. Min, "Performance Analysis of Multiple Rejects ARQ at RLC(Radio Link Control) for Packet Data Service in W-CDMA System," Wireless Communication and Networking Conference, 2000.
- [10] Nada Golmie, et al, "The NIST ATM/HFC Network Simulator," http://w3.antd.nist.gov/Hsntg/prd_atm-sim.html, Dec., 1998.
- [11] A. Samukic, "UMTS universal mobile telecommunications system : development of standards for the third generation," IEEE Transaction on Vehicular Technology, Vol.47, Issue 4, pp.1099-1105, Nov., 1998.
- [12] L. Bos and S. Leroy, "Toward an all-IP-based UMTS system architecture," IEEE Network, Vol.15, Issue 1, pp. 36-45, Jan./Feb., 2001.
- [13] A. Natani, et al., "TCP for Wireless Networks," survey paper on <http://www.utdallas.edu/~sudha/surveypapers.html>, Nov., 2001.
- [14] S. Lin and D. J. Costello, Jr., "Error Control Coding : Fundamentals and Applications," New Jersey : Prentice-Hall, 1983.
- [15] S. Lin, D. J. Costello, Jr. and M. J. Miller, "Automatic-Repeat-Request Error-Control Schemes," IEEE Communication Magazine, Vol.22, No.12, pp.5-17, Dec., 1984.



김 남 기

e-mail : ngkim@camars.kaist.ac.kr

1997년 서강대학교 컴퓨터학과 학사

2000년 한국과학기술원 전자전산학과 석사

2000년~현재 한국과학기술원 전자전산학과

박사 과정 재학중

관심분야 : 컴퓨터 네트워크, 인터넷, 이동통신, Adhoc망



박인용

e-mail : iypark@se.kumoh.ac.kr
1990년 연세대학교 전산학과 학사
1992년 한국과학기술원 전산학과 석사
2002년 한국과학기술원 전자전산학과 박사
2002년 한국과학기술원 정보전자연구소
연수연구원

2002년~현재 금오공과대학교 컴퓨터공학부 전임강사
관심분야 : 컴퓨터 네트워크, 분산처리



윤현수

e-mail : hyoon@camars.kaist.ac.kr
1979년 서울대학교 전자공학과 학사
1981년 한국과학기술원 전산학과 석사
1981년~1984년 삼성전자 연구원
1988년 오하이오 주립대학 전산학 박사
1988년~1989년 AT&T Bell Labs. 연구원

1989년~현재 한국과학기술원 전산학과 교수
관심분야 : Adhoc망, 암호학, 상호연결 네트워크, 병렬 컴퓨터