

모바일 ad-hoc 네트워크에서 TCP 성능 분석

정회원 남 호 석*, 준회원 조 술*, 이 태 훈*, 정회원 김 준 년*

TCP Performance Analysis over Mobile ad-hoc Networks

Ho-Seok Nam* *Regular Member*, Sol Cho*, Tae-Hoon Lee* *Associate Members*,
Jun-Nyun Kim* *Regular Member*

요 약

무선 데이터 통신 기술의 발전으로 모든 IP 기반의 네트워크는 유선과 무선이 혼합된 형태의 네트워크로 진화해 나갈 것이다. TCP는 연결형의 신뢰성 있는 전송계층 프로토콜로서 모든 유선 네트워크의 실질적 표준으로 사용되고 있다. 그러나, TCP의 혼잡제어 알고리즘이 비트 에러, 링크의 단절, 경로의 변화로부터 혼잡을 구분하지 못하기 때문에 TCP는 모바일 ad-hoc 네트워크의 성능 저하를 초래한다. 본 논문에서 TCP의 이론적인 특징을 고찰하고, 모바일 ad-hoc 네트워크에서 TCP의 성능이 저하됨을 ns2를 이용하여 분석하였다.

Key Words : TCP, MANET, Mobile computing, Congestion

ABSTRACT

With the development of wireless data communication technology, all IP-based network will become compositions of wired and wireless networks. TCP is a connection-oriented, reliable transport protocol and has been used as de facto standard in most wired networks. Because TCP's congestion control algorithm could not distinguish congestion from BER, link failure and frequent route changes, TCP shows a poor performance over mobile ad-hoc networks. In this paper, the theoretical feature of TCP was studied and the performance of TCP over mobile ad-hoc networks was analyzed with ns2.

I. 서 론

최근 무선통신기술과 네트워크의 분야의 급격한 발전으로 무선 환경에서 모바일 디바이스 간의 통신 서비스에 대한 요구가 증대되고 있다. 이러한 무선 데이터 통신의 발전은 새로운 분야의 발전 가능성을 제시하고 있으며 대표적인 분야가 센서 네트워크이다. 센서 네트워크의 특징은 기존 인프라를 사용하지 않고 무선 센서 노드들 사이에 자율적인 통신 경로를 설정하는 ad-hoc 네트워크라는 점이다. 또한 센서 네트워크를 포함해서 앞으로 모든 IP 기반의 네트워크는 유선과 무선이 혼재하

는 이중의 네트워크로 구성될 것이다. TCP/IP는 현재 IP 기반 네트워크의 실질적 표준으로 자리 잡았다. TCP는 연결형의 신뢰성 있는 서비스를 제공하는 전송 계층 프로토콜이다.^[1] TCP는 본래 BER(bit error rate)이 낮고 대부분의 패킷 손실이 혼잡에 의해서 발생하는 유선 환경 기반으로 설계되었다. 따라서 기존 TCP를 높은 BER, 노드의 이동에 의한 잦은 링크의 단절, 경로의 재설정 및 네트워크의 파티션이 빈번하게 발생하는 ad-hoc 네트워크에 적용하면 패킷 손실을 감지했을 때 혼잡 회피 단계로 들어가 심각한 성능 저하를 발생시킨다. 따라서 본 논문에서는 ad-hoc 네트워크 환경에

* 중앙대학교 전자전기공학부 컴퓨터통신연구실 (hsnam@bc.ac.kr), (sol5061@hanmail.net), (gaodayo@gmail.com), (jnk@cau.ac.kr)
논문번호 : KICS2007-09-415, 접수일자 : 2007년 9월 13일, 최종논문접수일자 : 2007년 12월 4일

서 TCP 전송 모델 성능을 분석하였다.

II. TCP의 혼잡제어

TCP는 연결 지향의 신뢰성 있는 전송 계층 프로토콜로 흐름 제어와 오류 제어 기능을 제공한다. 그 중 TCP의 흐름 제어는 슬라이딩 윈도우를 이용한다. cwnd와 rwnd를 비교하여 작은 값을 현재의 윈도우 값으로 설정한다.^[2]

2.1 TCP의 재전송

손실 또는 손상된 패킷을 처리하기 위해 TCP는 재전송 방식을 사용한다. 이 때 재전송 타이머를 이용하게 되는데 재전송 타이머의 값은 보통 RTT(round trip time)의 2배로 설정하게 된다. 송신측은 전송된 데이터에 대해서 RTO(retransmission time out) 동안 ACK를 받지 못하거나 3개의 중복된 ACK를 받게 되면 해당되는 데이터를 재전송하게 된다.

2.2 혼잡 제어

인터넷은 네트워크와 라우터들의 집합으로 이루어져 있다. 송신지에서 출발한 패킷은 여러 개의 라우터를 거친 후에 최종 목적지에 도달하게 된다. 라우터는 버퍼를 가지고 있어서 수신한 패킷을 저장하고 처리한 후 적절한 출력 포트로 전송한다. 혼잡은 라우터가 패킷을 처리할 수 있는 속도보다 더 빨리 패킷을 수신하는 경우에 발생한다. 통신망이 혼잡하면 전송된 패킷이 손실된다. 따라서 TCP는 이러한 현상을 방지하기 위해서 혼잡 윈도우를 이용한 저속 시작, 혼잡 회피, 신속 복구와 신속 재전송을 사용한다.

TCP 연결이 설정 되면 혼잡 윈도우의 크기를 1로 설정한다. 그 후에 전송된 각 세그먼트에 대한 ACK를 수신 할 때 마다 윈도우의 크기를 ACK를 수신한 데이터 숫자의 2배로 증가 시킨다. 저속 시작은 혼잡 윈도우의 크기가 TCP가 설정한 ssthresh(slow start threshold)값과 같아질 때까지 계속된다.

혼잡이 발생하기 전에 미리 혼잡을 예방하기 위해서 cwnd의 크기를 저속 시작의 지수 형태로 증가하는 속도를 줄여야 한다. 따라서 저속 시작 알고리즘이 시작되어 cwnd의 크기가 ssthresh가 같아지면 TCP는 혼잡 회피 단계로 들어가게 된다. TCP가 혼잡 회피 단계로 들어가게 되면 하나의 ACK를 수신 할 때마다 cwnd의 크기를 한 개 만

큼 증가 시킨다. 혼잡 회피 단계는 혼잡을 감지하거나 cwnd가 수신 윈도우의 값에 도달 할 때 까지 계속된다.

만약 혼잡이 발생하게 되면 TCP는 cwnd의 크기를 감소시킨다. TCP가 혼잡을 감지하게 되면 ssthresh값을 현재 혼잡 윈도우 값의 반으로 설정하고 cwnd를 1로 설정하여 저속 시작 단계로 들어가게 된다.

TCP가 혼잡을 감지하였을 때 RTO 값이 초과된 것이 아니라 중복된 3개의 확인 응답을 받게 되면 해당하는 데이터를 바로 재전송 하고 RTO 값을 다시 설정한다. 이때는 cwnd를 1로 설정하지 않고 현재 값의 반으로 설정한 후 혼잡 회피 단계로 들어간다.

III. 모바일 ad-hoc 네트워크에서의 TCP

전송채널이 무선인 ad-hoc 네트워크 환경에서는 유선과 다르게 많은 변수가 존재한다. 무선 링크는 자유공간을 전송채널로 사용하고 있어서 날씨, 건물과 같은 장애물, 다중경로 간섭현상 등과 같은 제어할 수 없는 많은 요소들이 존재한다.^[3] 또한 무선 노드들은 제한된 라디오 커버리지를 가지고 있으며 노드들의 이동에 의한 빈번한 핸드오프 문제도 발생하게 된다. 따라서 모바일 ad-hoc 네트워크 환경은 높은 BER과 노드들의 이동에 따른 링크의 단절과 경로의 변화, 네트워크 파티션 등의 특징을 나타내게 된다.^[4] 모바일 ad-hoc 네트워크에서 TCP는 앞에서 언급한 특성에 의한 패킷 손실이나 지연된 응답을 모두 네트워크의 혼잡으로 취급하기 때문에 매우 취약한 단점을 가지고 있다.

3.1 BER

모바일 ad-hoc 네트워크 환경은 높은 BER을 가진다. BER은 TCP 데이터 세그먼트나 ACK 패킷을 손상시키는 결과를 가져온다. 만약 전송된 세그먼트의 ACK가 지정된 RTO에 도착하지 않았을 때, 송신측의 TCP는 해당 세그먼트를 재전송하면서 RTO를 지수적으로 증가시키고 cwnd는 줄이게 된다. 결국 에러가 반복되어 발생하게 되면 송신측의 cwnd 크기는 1로 줄어들게 되고, 이것은 송신측의 throughput을 매우 낮게 만든다.^{[5],[6]} 높은 BER은 FEC(forward error coding)이나 ARQ를 이용해서 해서 어느 정도 해결할 수 있으나 오류정정이 필요 없을 경우 무선대역의 많은 부분을 낭비하게 된다.

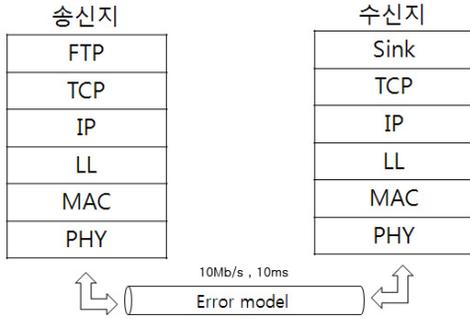


그림 1. BER 성능 분석 모델

3.2 경로 재설정

모바일 ad-hoc 네트워크 환경에서 각 노드들은 이동하기 때문에 송신지에서 목적지로 가는 경로는 자주 변화하게 된다. DSDV는 주기적인 컨트롤 메시지의 교환을 통해 변경된 라우팅 테이블을 갱신하게 된다.^[7] DSR, AODV 및 TORA는 필요시 경로탐색 메시지를 통해 경로를 재설정 한다.^{[8],[9],[10]} 만약 새로운 경로가 이전 경로의 RTO보다 긴 시간의 RTO를 필요로 할 경우, 송신측의 TCP가 이전의 RTO로 패킷을 전송하면 타임아웃이 발생하게 된다. TCP는 타임아웃이 발생하게 되면 혼잡으로 간주하기 때문에 저속 출발과 혼잡회피 알고리즘에 의해 cwnd를 조절하게 되고, 이는 throughput을 작게 만드는 원인이 된다. 빈번한 경로의 변화에 의해 이러한 상황이 자주 발생하게 되면, 송신측의 cwnd는 수신측에서 광고되어진 윈도우 크기보다 훨씬 작은 크기의 패킷만 보낸다.

3.3 네트워크 파티션

모바일 ad-hoc 네트워크 환경에서 각 노드들은 순간적으로, 또는 RTO보다 긴 시간동안 네트워크가 파티션 될 수 있다. 만약 네트워크가 파티션 된 상태라면 송신측의 TCP에서 보낸 패킷들은 모두 손실 될 것이고, 송신측은 이를 혼잡으로 간주하여 혼잡 회피 단계를 수행하게 될 것이다. TCP는 네트워크 파티션을 알 수 없기 때문에 타임 아웃된 같은 패킷을 지속적으로 재전송하게 될 것이고, 송신측 TCP는 경로가 재설정 될 때까지 RTO를 지수함수적으로 증가하게 된다.

3.4 다중 경로

TORA와 같은 라우팅 프로토콜은 송신지와 목적지 사이의 여러 경로를 가지고 있기 때문에 수신측

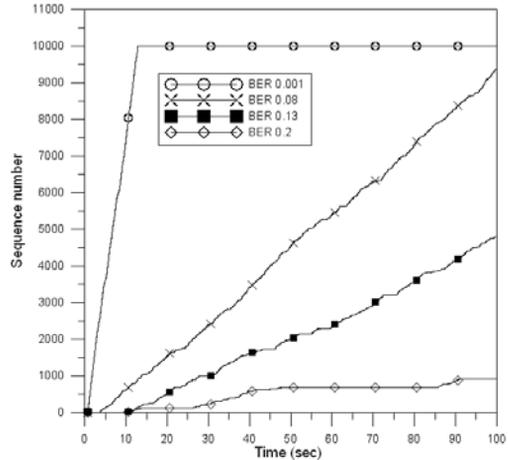


그림 2. 시간에 따른 순서 번호

에서 받은 패킷들의 순서번호는 순차적이지 않다. 패킷들의 순서가 순차적이지 않을 경우에 수신측은 중복된 ACK를 발생하고, 송신측이 중복된 ACK를 3개 이상 받게 되면 혼잡제어를 실행한다. 따라서 무선 노드들의 움직임이 많아져서 루트의 변화가 빈번하게 발생하게 되면 TORA의 성능은 떨어지게 된다.

IV. 모바일 ad-hoc 네트워크에서 TCP 성능분석

본 논문에서는 모바일 ad-hoc 네트워크에서 TCP 성능 분석을 하기위해서 각각의 설정된 시나리오를 작성해서 네트워크 시뮬레이터인 ns2를 이용하였다.^[11]

4.1 BER에 따른 TCP 성능 분석

모바일 ad-hoc 네트워크의 높은 BER로 인한 TCP의 성능을 분석하기 위해서 그림 1과 같이 FTP를 이용하여 대역폭이 10Mbps이고, 시간 지연이 10ms의 링크를 통해 10Mbyte의 파일을 전송한다고 가정하였다. 송신지와 수신지의 링크는 가상의 무선 환경을 구현하기 위해서 에러 모델을 삽입하였다.^[12] Throughput은 송신지에서 보낸 패킷의 수를 나타내고 goodput은 수신지에 제대로 전달된 패킷의 수를 나타낸다.

그림 2는 BER이 순서 번호 증가에 어떤 영향을 미치는가를 보여주고 있다. 에러율이 거의 없다고 볼 수 있는 0.001의 경우에는 10 Mbyte의 파일의 전송이 15초가 되지 않아 완료되었다. 하지만 BER이 0.08, 0.13, 0.2인 경우에는 시뮬레이션 시간 동안 파일을 완전히 전송하지 못했다. 그림에서 순서 번호가 증가하지 못하고 거의 수평을 이루는

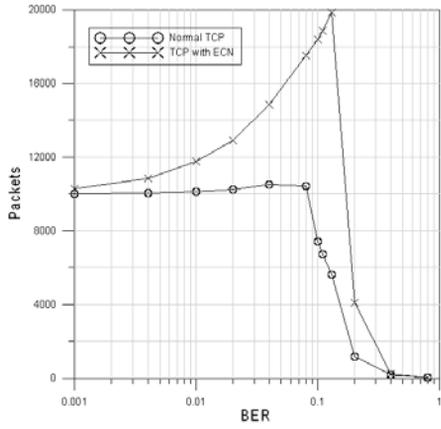


그림 3. BER에 따른 패킷 전송량

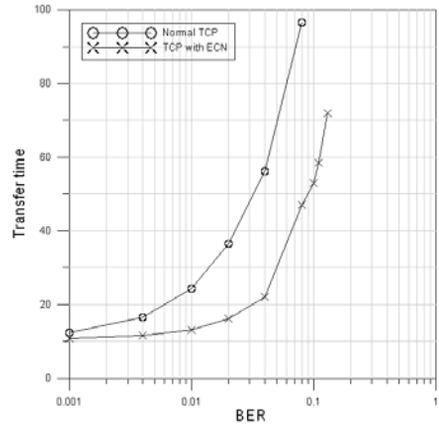
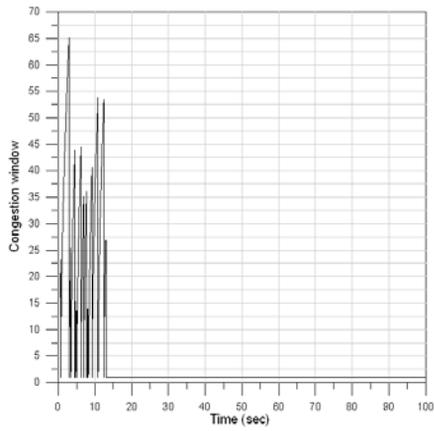
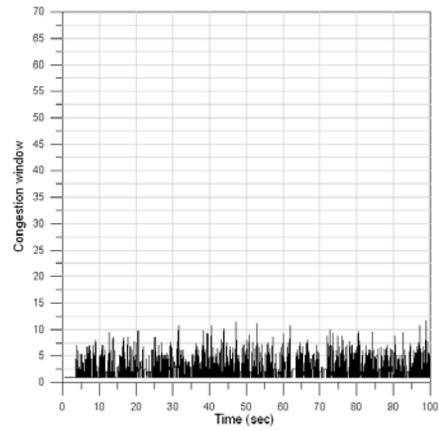


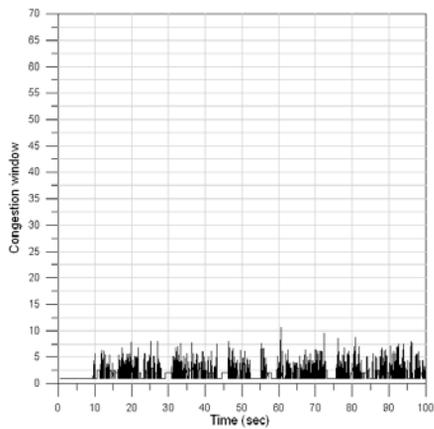
그림 4. BER에 따른 전송시간



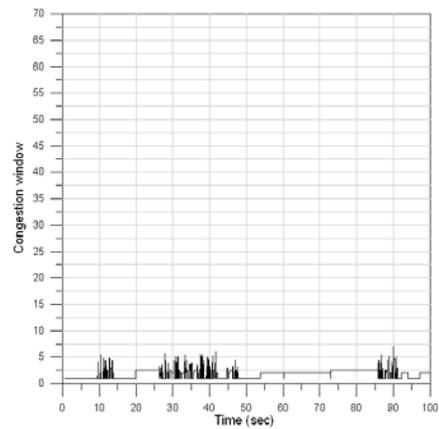
(a) BER = 0.001



(b) BER = 0.08



(c) BER = 0.13



(d) BER = 0.2

그림 5. BER에 따른 혼잡 윈도우의 변화

구간은 세그먼트가 전송 중에 손실되었거나 제대로 전송된 세그먼트에 대한 ACK가 손실되어 TCP가 해당 세그먼트를 계속 재전송하고 있는 것을 알 수 있다.

그림 3은 BER이 패킷 전송량에 어떤 영향을 미치는가를 보여주고 있다. BER이 증가함에 따라서 기존 TCP에서는 전송량이 0.1 이하부터 크게 감소한다. 크로스레이어의 일부 개념인 ECN(explicit congestion notification)을 사용하였을 때 throughput에 대하여 성능이 향상됨을 알 수 있다.

그림 4는 BER에 따라 10Mbyte의 데이터를 FTP로 보낼 때 전송시간을 도시하였다. 에러율에 따라 전송 시간이 지수적으로 증가함을 보여주고 있으며 기존 TCP 환경에서는 에러율이 0.1을 초과하면 사실상 전송시간이 무한대에 가까워지면서 파일 전송이 불가능해진다는 것을 보여준다. 하지만 ECN을 이용한 경우에는 기존 TCP 환경보다 더 짧은 전송시간을 보여주고 있으며 BER이 0.1이 초과한 몇몇의 경우에도 파일 전송이 가능하다는 것을 보여주고 있다.

그림 5는 각 BER에 따라 cwnd의 변화량을 보여주고 있다. 그림 5 (a)의 경우는 거의 에러가 발생하지 않으므로 10 Mbyte의 데이터를 다 전송한 후

cwnd가 최소값인 1로 설정된다. 그림 5(b) ~ 그림 5(d)에서는 BER이 높아질수록 손실된 세그먼트 수가 많아지고 TCP는 세그먼트의 손실을 혼잡으로 간주하고 cwnd는 점차 작아진다. BER이 0.2일 때는 cwnd가 7을 넘어서지 못하고 반복적으로 혼잡 감지와 혼잡 회피가 일어나고 있다. 따라서 TCP의 급격한 성능저하가 발생함을 알 수 있다.

4.2 링크의 단절과 경로의 재설정

링크의 단절과 경로의 재설정을 시뮬레이션 하기 위해서 그림 6과 같이 물리적 크기가 300m×800m인 환경에서 4개 노드가 FTP로 데이터를 송수신하는 모델을 작성하였다. 시뮬레이션이 시작되면 송신지는 10초 후에 그림 6 (a)와 같이 FTP로 데이터를 전송한다. 50초 후 그림 6 (b)와 같이 노드 2가 노드 0의 라디오 커버리지 밖으로 벗어나 링크가 단절된다. 120초 후 그림 6 (c)와 같이 노드 3의 이동에 의해 링크가 다시 연결되고 새로운 네트워크 토폴로지를 형성한다.

그림 7과 그림 8은 그림 6 (a)의 상태에서 노드의 이동이 없을 때의 순서번호와 cwnd의 변화를 나타낸다. 그림 7과 그림 8에서 노드의 이동에 의한 링크 단절이 없기 때문에 순서번호는 선형적으로

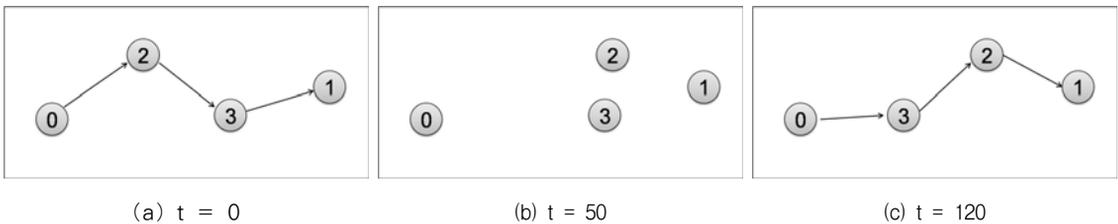


그림 6. 링크의 단절과 경로의 재설정

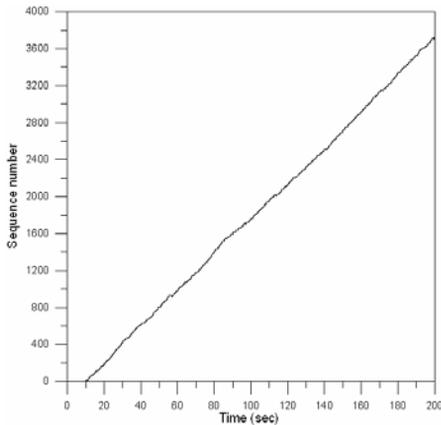


그림 7. 순서번호

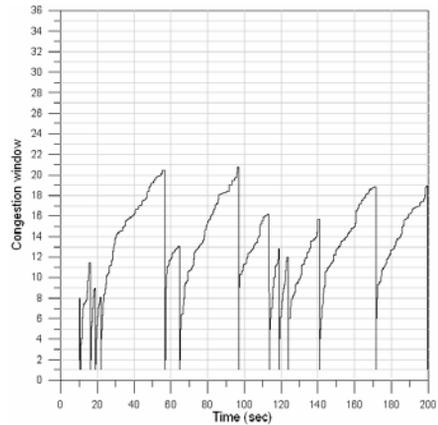


그림 8. 혼잡 윈도우

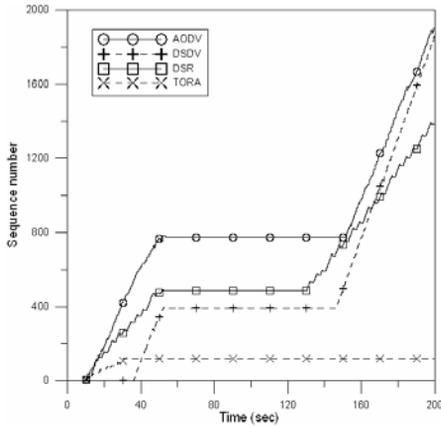
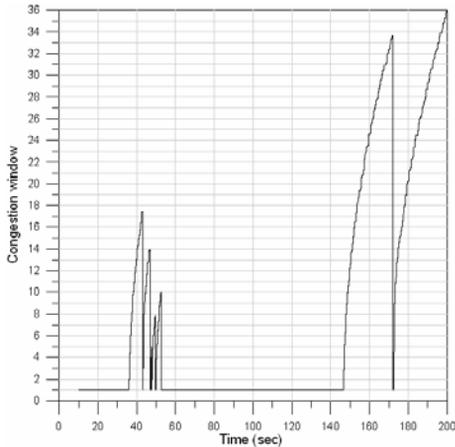


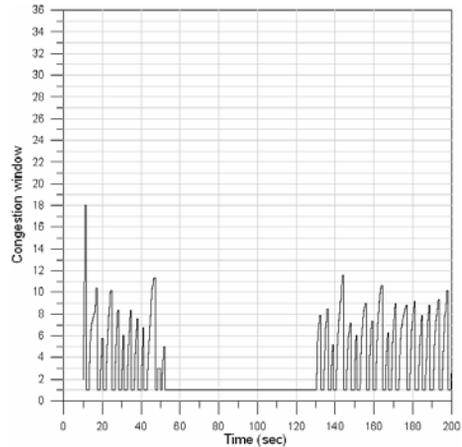
그림 9. 라우팅 프로토콜 순서번호

로 증가하게 되고, cwnd는 일반적인 TCP의 혼잡제어 알고리즘을 나타낸다.

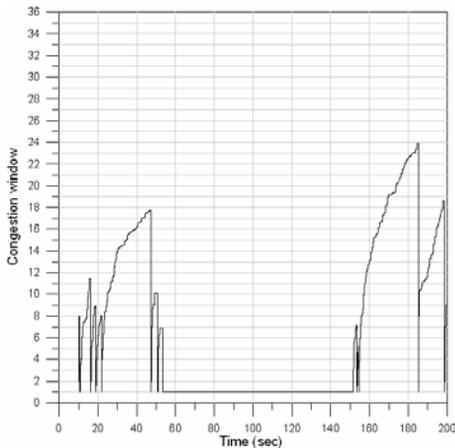
그림 9는 각 라우팅 프로토콜의 순서번호를 비교한 것이다. DSDV는 전체 네트워크 토폴로지를 구성한 후에 데이터를 전달하기 때문에 다른 프로토콜과 달리 전송 시작 시간이 약간 늦어지게 된다. 그림 6 (b)에서 노드 0과 노드2의 링크가 단절되므로, TCP의 순서번호는 증가를 멈추게 된다. 노드 3과 노드 0의 링크가 연결되는 그림 6 (c)상태가 되면 순서번호는 다시 증가하게 된다. TORA는 새로운 경로 설정 시에 링크 역전에 의한 반사, 전파 등에 의해 다른 라우팅 프로토콜보다 많은 라우팅 오버헤드를 발생시키고 일시적인 루프의 발생 때문에 노드3이 노드0의 라디오 커버리지 안에 들어왔지만 데이터의 교환이 일어나지 않았다.



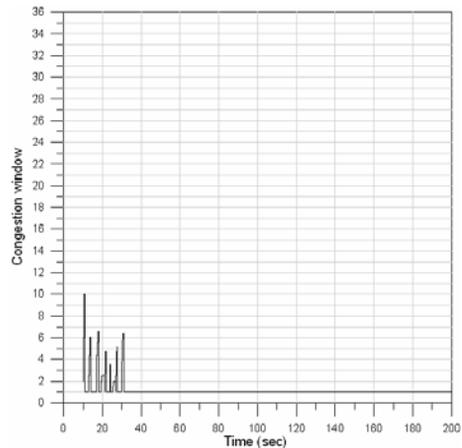
(a) DSDV



(b) DSR

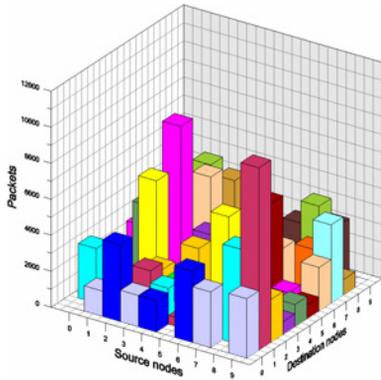


(c) AODV

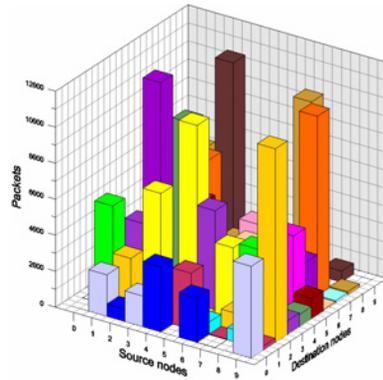


(d) TORA

그림 10. 라우팅 프로토콜에 따른 혼잡 윈도우

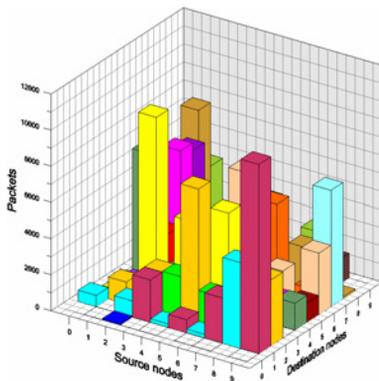


(a) 정지시간 100초

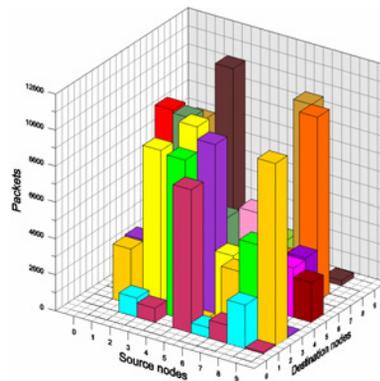


(b) 정지시간 500초

그림 11. AODV throughput



(a) 정지시간 100초



(b) 정지시간 500초

그림 12. TORA throughput

그림 10 (a) ~ 그림 10 (d)는 각 라우팅 프로토콜별 cwnd의 변화를 나타낸다. 각 그림에서 볼 수 있듯이 패킷 손실이 발생하여 재전송이 일어나게 되면, TCP는 ssthresh값을 현재 값의 반으로 하고 cwnd를 1로 설정하여 저속 시작 단계로 들어가게 된다. 그림 6에서와 같이 노드 0과 노드 2의 연결이 단절된 50초 지점부터 cwnd는 1로 변하고, 재연결이 된 후에는 cwnd가 1부터 지수증가 하는데 이것은 TCP가 링크가 단절된 것을 혼잡으로 인식하는 것에 따른 결과이다.

4.3 네트워크 파티션 및 다중 경로

앞에서 설명한 모바일 ad-hoc 네트워크 환경에서 네트워크 파티션 및 다중경로에 따른 TCP의 성능평가를 위해서 물리적 크기가 500m×1000m 환경에서 10개의 노드가 정지시간 100초와 500초에 대한

시나리오 파일을 작성해서 시뮬레이션을 수행하였다. 각 노드의 최소속도는 0.5m/s 이고 최대속도는 20 m/s로 설정하였다. 물리적 크기를 500m×1000m으로 설정한 것은 의도적으로 잦은 링크의 단절과 네트워크 토폴로지의 변화를 주기위해서 선택하였다. 시뮬레이션 수행 시간 동안 각 노드는 일대일 대응관계에 의해 임의의 시간에 90번의 TCP 연결을 설정하였으며 10MByte의 파일을 FTP로 전송한다고 가정하였다.

그림 11은 라우팅 프로토콜이 AODV인 경우 송신지 노드와 목적지 노드 사이의 throughput을 측정하는 것으로 노드의 이동이 잦을 수록 송신지 노드의 전체적인 throughput은 감소하는 것을 확인할 수 있다. 이는 앞에서 언급한 대로 새로운 경로가 이전 경로 보다 긴 시간의 RTO를 필요로 하지만 타임아웃이 발생해서 혼잡 회피 알고리즘에 의해

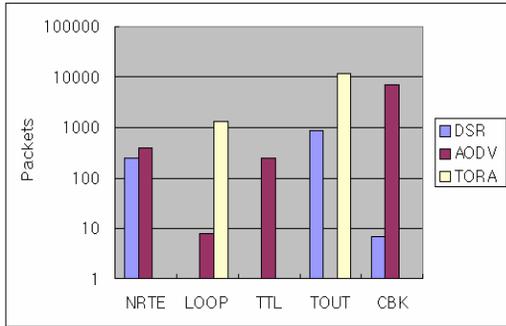


그림 13. 패킷의 손실

cwnd를 조절하게 되고 전체적인 throughput을 작게 만드는 데서 비롯된다. 또한 그림 11(b)와 같이 정지 시간이 500초 일 때 일부 노드 사이에서는 네트워크가 파티션이 되었지만 송신지는 노드는 혼잡을 감지한 것으로 판단하고 RTO를 지속적으로 증가시켜서 throughput이 급격하게 감소한 것을 볼 수 있다.

그림 12는 라우팅 프로토콜이 TORA인 경우 송신지 노드와 목적지 노드 사이의 throughput을 측정 한 것이다. 전체적인 결과는 앞에서 설명한 AODV와 같으나 잦은 링크의 단절과 경로의 변화가 많은 노드들 간의 TCP 연결에서는 TORA가 여러 경로를 가지고 있기 때문에 수신측에서 중복된 ACK를 발생으로 인한 혼잡제어를 실행한 결과이다.

그림 13은 중지시간 100일 때 각 라우팅 프로토콜에 대한 손실된 패킷의 이유를 나타낸다. NRTE는 송신지 노드와 목적지 노드 사이에 경로가 존재하지 않은 경우로 네트워크 파티션이 TCP 성능을 저하시키는 것을 확인 할 수 있다. LOOP는 송신지 노드와 목적지 노드 사이에 루프가 존재하는 것으로 TORA의 같은 경우는 잦은 링크 역전 알고리즘에 의해 일시적인 루프가 많이 발생하는 것을 알 수 있다. TTL은 AODV에서 라우팅 정보에 life-time을 설정하고 life-time이 지나면 폐기되는 것을 나타내고 TOUT은 패킷이 소멸한 것을 나타낸다. CBK는 라우팅 프로토콜이 MAC 계층까지 전달되지 못한 것으로 이는 802.11 MAC 계층에서 인접 노드들 간의 충돌로 인한 것이다.

V. 결 론

본 논문에서는 모바일 ad-hoc 네트워크 환경에서 TCP의 성능이 저하됨을 고찰하였다. BER과 노드의 이동에 의한 링크가 단절될 경우, TCP의 혼잡 제어

알고리즘이 불필요하게 cwnd를 감소시켜 무선 대역폭의 낭비를 가져온다. 또한, 모바일 ad-hoc 네트워크에서 네트워크 파티션 및 다중경로의 중복된 ACK에 대해서 TCP는 혼잡으로 판단하여 전체적인 성능저하가 발생하였다. 따라서, 무선 환경에서 TCP 성능 개선을 위해 여러 가지 형태의 TCP들이 고안되어졌다. 분할 연결 모델을 사용하여 중간의 기지국이 에러가 발생 시에 재전송을 대신 해주는 I-TCP와 ECN과 ICMP메시지를 통해 네트워크의 상태를 파악해 송신측의 TCP 상태를 조절하는 ATCP가 있다. 이외에는 리액티브 혼잡 제어를 사용하는 TCP-Reno와 TCP-New Reno가 있고 프로액티브 혼잡 제어를 사용하는 TCP-Vegas, TCP-Veno, TCP-Westwood, TCP-Jersey등이 있다. 하지만 위의 TCP는 기존 유선 네트워크와의 완벽한 호환성을 기대하기 어렵다. 앞으로 모든 IP 기반의 네트워크는 무선과 유선이 결합된 이종의 네트워크로 진화할 것이다. 그러므로 TCP가 기존 유선 네트워크와 호환성이 보장되고 모바일 ad-hoc 네트워크에서 원활한 성능을 수행하기 위해서는 크로스 레이어 개념을 도입하여 하위 계층에서 전송 계층으로의 정보전달을 이용한 TCP의 효율적인 혼잡 제어 알고리즘에 대한 연구가 필요하다.

참 고 문 헌

- [1] Behrouz A. Forouzan, *TCP/IP Protocol Suite 3rd edition*, McGRAW-Hill p.278-344
- [2] W. Richard Stevens, *TCP/IP Illustrated, Vol.1*, Addison-Wesley p.297-322
- [3] Ye Tian, Kai Xu, Nirwan Ansari, "TCP in wireless environments: problems and solutions", *Communications Magazine*, IEEE Volume 43, Issue 3, March 2005 P.S27-S32
- [4] Jian Liu, Suresh Singh, "ATCP: TCP for mobile ad hoc networks", *Selected Areas in Communications, IEEE Journal on* Volume 19, Issue 7, July 2001 P.1300-1315
- [5] A. Bakre and B. R. Badrinath, "I-TCP: Indirect TCP for mobile hosts", in Proc. 15th Int. Conf. Distributed Computing Systems, Vancouver, BC, Canada, June 1995, PP.136-143
- [6] K. Brown and S. Singh, "M-TCP : TCP for mobile cellular networks", *ACM Comput.*

Commun. Rev., vol. 27.no.5, pp.19-43, 1997.

- [7] C. E. Perkins & P.Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", in *Computer communications Review*, Oct 1999
- [8] David B. Johnson, David A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks, in *Mobile computing*", 1996
- [9] Perkins, C.E. and Royer, E.M., "Ad-Hoc On-Demand Distance Vector Routing", *IEEE WMCSA '99*, New Orleans, LA, Feb. 1999
- [10] V. Park & S. Corson, "Temporally-Ordered Routing Algorithm(TORA)", <http://www.ietf.org/internet-drafts/draft-ietf-manet-tora-spec-03.txt>, *IETF Internet draft*, Nov 2000, Work in progress
- [11] The VINT Project, *The NS Manual*, 2007
- [12] Fabienne Lefevre, Guillaume Vivier, "Understanding TCP's behavior over wireless links", *Communications and Vehicular Technology*, 2000. *SCVT-200. Symposium* on 19 Oct. 2000 P.123-130

남 호 석 (Ho-Seok Nam)

정회원



1993년 2월 중앙대학교 전자 공학과 졸업

1995년 2월 중앙대학교 전자 공학과(석사)

1995년 2월~2000년 국방과학연구소 4 체계 연구부

2005년~현재 중앙대학교 전자전기공학부 박사수료

<관심분야> 무선통신, 센서네트워크, USN

조 솔 (Sol Cho)

준회원



2006년 2월 중앙대학교 전자전기공학부 졸업

2006년 3월~현재 중앙대학교 전자전기공학부(석사과정)

<관심분야> 센서네트워크, USN

이 태 훈 (Tae-Hoon Lee)

준회원



2007년 2월 중앙대학교 전자전기공학부 졸업

2007년 3월~현재 중앙대학교 전자전기공학부(석사과정)

<관심분야> 센서네트워크, USN

김 준 년 (Jun-Nyun Kim)

정회원



1988년~현재 중앙대학교 전자전기공학부 교수

2000년~2006년 ISO/IEC JTC 1/SC 6 Chairman

2002년 개방형 컴퓨터 통신 연구회(OSIA) 회장

2004년~2005년 한국통신학회 학술이사