

무선 네트워크에서 링크 상태 정보를 이용하는 TCP 효율 향상을 위한 패킷 스케줄링 메커니즘

김성철[†], 이진영^{**}

요 약

본 논문에서는 무선 네트워크에서 모바일 호스트가 수신 호스트인 경우, 기지국에서 패킷을 전송할 때에 목적지 모바일 호스트까지의 링크 상태를 미리 예측한 후에 예측된 정보를 이용하여 전송 효율을 높일 수 있는 새로운 패킷전송 메커니즘을 제안한다. 제안된 패킷전송 메커니즘에서는 패킷을 전송하기 전에 미리 예측된 정보를 이용하여 패킷 스케줄링이 이루어지기 때문에 전체 패킷 전송 시간을 줄일 수 있는 장점을 가진다. 뿐 만 아니라 링크 상태가 좋지 않은 목적지 호스트로 향하는 패킷을 전송하지 않고 임시 FIFO 버퍼에 저장한 후에 나중에 전송함으로써, 기존의 방식에서 발생할 수 있는 HOL(Head Of Line) 블로킹 문제를 해결할 수 있다는 장점도 가진다.

Packet Scheduling Mechanism to Improve TCP Throughput Using Link State Information in Wireless Networks

Seong-Cheol Kim[†], Jin-Young Lee^{**}

ABSTRACT

In this paper we propose a new packet scheduling mechanism to improve TCP throughput and fairness with estimated link state information in wireless networks, in which mobile hosts are receivers. Since a priori estimated link state information is used in this proposed mechanism, the entire packet transmission time can be reduced. In other words, packets scheduling procedure is performed before packets are transmitted using estimated link state. The proposed mechanism also has an temporary FIFO queue to store packets which are on the head of queue and can not be transmitted due to bad link state. Using this temporary FIFO queue, HOL(Head of Line) problem of FIFO queue can be resolved.

Key words: TCP(전송 제어 프로토콜), Scheduling(스케줄링), Wireless Networks(무선 네트워크)

1. 서 론

무선망의 출현으로 기존의 인터넷 환경은 유무선

이 통합된 망으로 변화하고 있다. 이러한 유무선 통합 망에서는 모바일(mobile) 사용자들의 연결들을 유지하기 위해서 일정한 수준의 서비스 질(QoS: Quality of Service)을 만족시켜야 한다. 이러한 문제들을 해결하기 위해 무선 환경에서의 라우팅을 위한 기법들과 단말기의 이동성에 대응할 수 있는 기법 그리고 무선 환경에서 각종 프로토콜들의 성능향상을 위한 여러 기법들이 제안되었다. 그러나 이러한 네트워크 계층에 대한 문제 이 외에 종단간(End-to-End) 신뢰성 있는 연결을 보장하기 위한 TCP (Transmission Control Protocol)의 유효성에 대한

※ 교신저자(Corresponding Author): 김성철, 주소: 서울시 종로구 홍지동 7(110-743), 전화: 02)2287-5315, FAX: 02)396-5704, E-mail: sckim@smu.ac.kr

접수일: 2004년 5월 25일, 완료일: 2004년 9월 1일

[†]정회원, 상명대학교 소프트웨어학부 부교수

^{**}정회원, 상명대학교 컴퓨터학과

(E-mail: jylee@smu.ac.kr)

※ 이 논문은 2004년도 상명대학교 소프트웨어미디어연구소의 연구비 지원 하에 수행되었음.

검증은 해결해야 할 과제로 남아있다.

오늘날 통신환경에서 널리 사용되고 있는 TCP는 유선망과 고정된 호스트로 이루어진 기존의 유선 네트워크에 적합하며 유선망의 특성을 적용한 것이다. 따라서 무선 구간의 제한된 대역폭, 높은 지연, 잦은 비트 에러, 일시적인 연결 두절 및 핸드오프 등과 같은 특징을 가진 무선망(wireless network)에 이를 그대로 적용하게 되면 오히려 불필요한 메커니즘의 사용으로 인해 종단간(end-to-end) 처리율(Throughput)의 성능 저하를 가져오게 된다. 이런 처리율의 성능 저하는 TCP가 유선망의 낮은 비트 에러율(error rate)때문에 통신망에서 발생하는 패킷 손실을 기본적으로 혼잡에 의한 것으로 생각하여 혼잡 제어 메커니즘으로 패킷 손실을 처리하기 때문에 발생한다. TCP 송신자는 패킷 손실을 발견하면 먼저 전송 윈도우 크기를 줄이고 손실된 패킷을 재전송한다. 또한 혼잡 제어나 회피 메커니즘을 초기화하고 재전송을 위한 타이머의 값을 증가시킨다. 하지만 무선 네트워크에서의 패킷 손실은 대부분 혼잡에 의해서가 아니라 핸드오프 시의 연결이 끊기거나 높은 비트 에러율에 의해 발생하게 된다. 그러므로 앞에서 언급한 TCP의 패킷 손실 회복 메커니즘을 그대로 무선 네트워크에 적용시키면 오히려 불필요한 성능 저하를 가져오게 된다. 그리고 이러한 유선망에서의 TCP 프로토콜을 무선 환경에 그대로 적용될 경우, 유선 구간의 대역폭 이용 면에서 불필요한 감소를 초래하며 처리율의 저하 및 지연을 증가시켜 전체망의 성능을 저하시키게 된다.

이러한 무선 환경에서의 TCP 성능 저하를 해결하기 위하여 다양한 방법들이 이루어지고 있다[1-5]. 이런 방법들에는 연결 분할(Split connection)방식, 빠른 재전송 방식, 링크 계층 재전송 방식 등이 있다. 이러한 방식들은 유선망과 무선망을 구분하여 무선망과 관련된 손실을 고정 호스트(FH: Fixed Host)에 숨김으로써 성능을 향상시켰다.

이처럼 무선망에서의 TCP 성능을 향상시키기 위하여 여러 방법들이 제안되어 왔지만 여러 채널이 통합되는 BS(Base Station)에서의 버퍼링 문제 또한 매우 중요하다. 무선 환경에서는 구현이 간단한 FIFO(First-In First-Out) 큐 패킷 스케줄링을 사용하고 있다. 그러나 이 FIFO 큐 방식에서는 버스트 에러 발생 시, 다른 패킷의 전송을 막는 HOL(Head

Of Line) 블로킹(blocking)이라는 문제점이 발생한다. 즉, 한 개의 버스트 에러 상태인 모바일 호스트(MH: Mobile Host)가 다른 모바일 호스트들의 전체 처리율까지 감소시켜 대역폭 할당에 있어서 불공평성을 가져와 전체 처리율의 성능이 저하된다. 이러한 무선망에서 발생하는 성능 저하를 향상시키기 위하여 다양한 스케줄링이 제안되어 왔는데 그중 하나가 [8]에서 제안한 CSDP(Channel State Dependent Packet) 스케줄링 방식이다. 이 방식은 전체 처리율을 증가시키고 불공평성 문제도 감소시키는 장점을 가진다. 반면에 이 방식은 모바일 호스트의 빈번한 이동이 있을 경우, 즉 셀(Cell) 내의 모바일 호스트의 수가 수시로 변하게 되는 경우에 버퍼할당을 자주 해주어야 하는 문제와 동시에 많은 호스트가 이동할 경우, 버퍼의 이용률이 낮아질 수 있다는 단점을 가진다. 본 논문에서는 이러한 문제를 해결하기 위해 미리 링크 상태를 예측해서 HOL에 있는 패킷이 bad 상태인 링크로 향할 경우에 별도의 큐에 임시로 저장한 후에 처리하는 방안을 제안한다. 제안된 방안은 FIFO 큐의 장점을 가지면서 적은 변형을 통하여 버퍼 관리 문제를 해결하여 더 큰 성능향상을 기대할 수 있는 장점을 가진다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 채널상태의존 패킷 스케줄링 방식에 대해 설명하고, 3장에서는 제안하는 TCP 효율 향상 메커니즘에 관하여 설명한다. 4장에서는 시뮬레이션 환경 및 실험 결과에 대한 비교 및 분석 후 5장에서는 결론을 맺고자 한다.

2. 관련 연구

무선망을 통해 데이터를 전송할 때 잡음, 간섭 등의 무선 에러 특성으로 인해 패킷 손실이 크다. 따라서 무선망에서의 정확한 데이터 전송을 위해서는 물리 계층에서 발생하는 에러율을 감소시켜야 한다. 이를 위해 상위 계층인 데이터 링크 계층에서는 에러 제어 메커니즘이 필요하다. 대표적인 에러 제어 메커니즘의 하나는 손실된 원래의 데이터를 부가적 정보로부터 복원하기 위하여 원래의 정보에다 부가적 정보를 함께 전송하는 FEC(Forward Error Correction) 방식이다[6,7]. 그러나 FEC 방식은 데이터 이외에 추가로 에러 제어 코드를 함께 전송해야 하기 때문

에 전송 효율이 다소 떨어지는 단점이 있지만, 에러가 발생할 경우 재전송을 요구하지 않아 시간 지연을 피할 수 있는 장점이 있다. 일반적으로 무선 네트워크에서 주어진 RTT(Round Trip Time)와 패킷 손실률 P와 TCP의 효율 η 의 관계는 다음 식으로 주어진다[6,7].

$$\eta = \min \left\{ \frac{1}{RTT} \sqrt{\frac{3}{2P}}, \frac{W_{\max}}{RTT} \right\} \quad (1)$$

여기서 W_{\max} 는 TCP 소스의 최대 혼잡 윈도우의 크기이며, 패킷 손실률 P는 다음과 같이 주어진다.

$P = \text{Prob}\{N\text{개의 단위크기를 가진 한 개의 프레임에서 } K\text{개 이상이 손실될 경우}\}$

$$= \sum_{i=0}^{K-1} \binom{N}{i} (1-p)^i p^{N-i} \quad (2)$$

여기서 p 는 무선 링크 상에서 비트 에러율로서 패킷 손실률 P와는 다르다. 또한 FEC 블록코드는 (N, K)로 표현되며, N은 전송되는 전체 크기이며, K는 원래의 데이터, N-K는 에러 검출을 위해 사용되는 추가 정보(Redundancy)를 의미한다. 그러므로 전송되는 전체 블록의 크기 $N=K+R$ 이며, FEC의 상대적인 양은 N/K 이다. 전송되어지는 원래 패킷에 FEC 코드를 추가함으로써 TCP 패킷의 손실률은 감소되지만, FEC의 양이 어느 시점 이상이 되면 효율이 감소된다. 위의 식에서 TCP의 효율은 패킷 손실률에 반비례함을 알 수 있는데 p 가 변함에 따라 TCP 효율의 변화에 대한 것이 그림 1에 보여준다. 여기에서 RTT(Round Trip Time)는 평균 왕복 시간으로 560 ms로 설정했고, 무선 링크 대역폭인 μ 는 3000 units/s를 사용하였다.

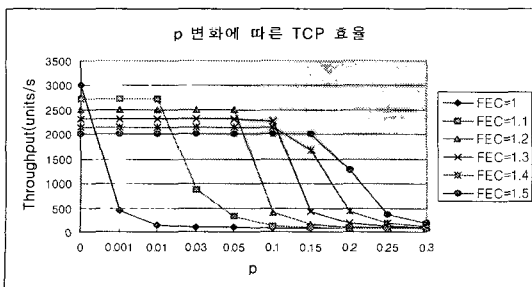


그림 1. 에러율 p의 변화에 대한 TCP 효율의 변화

위의 그림 1에서 볼 수 있듯이 전송되어지는 데이터에 첨가된 FEC 코드의 길이가 길수록 에러의 변화에 적게 영향을 받으며, 또한 최대 TCP 효율이 줄어듦을 알 수 있다. 즉, FEC의 코드 길이가 증가할수록 에러의 변화율에 의한 TCP 효율의 변화는 줄어든다. 따라서 이러한 특성을 이용하여 네트워크 상황에 따라 순간순간 에러율을 예측하여 적절한 FEC 코드를 부가하면, 네트워크 성능을 향상시킬 수 있음을 알 수 있다. 따라서 본 논문에서는 네트워크의 링크 상태를 예측하여, 이를 패킷전송에 적용함으로써 보다 효율적인 전송을 할 수 있는 메커니즘에 대하여 살펴보기로 한다.

일반적으로 인터넷 망에서 소스 호스트에게 네트워크에서 일어난 에러 발생을 알리는 방법으로 TCP 레벨의 타임아웃 메커니즘을 사용한다. 즉, 소스 호스트에서는 패킷을 전송한 후에 수신 호스트로부터 확인응답 신호를 받은 후에야 데이터 전송이 성공적으로 이루어졌음을 안다. 만약 일정 시간 후에도 수신 호스트로부터 확인응답 신호를 받지 못하는 경우에는 네트워크에서 에러가 발생한 것임을 알고 재전송하게 된다. 그러나 이 방법에서의 타임아웃 시간은 패킷 전송의 왕복시간(round trip time)보다 크게 설정하기 때문에 손실된 패킷에 대한 반응이 느리다는 단점을 가진다. 따라서 전송 에러 발생을 빠르게 소스 호스트에게 알림으로써 효율을 증대시키는 방법은 링크 레벨에서의 타임아웃을 사용하는 것이다. 현재 사용 중인 대부분의 무선 랜 카드의 드라이버에서는 패킷을 보내기 위해 일반적으로 FIFO 큐를 사용하고 있다. 이 FIFO 큐를 사용하는 스케줄링 방식은 일반적으로 패킷 손실이 임의로 일어날 경우에는 좋은 성능을 보이지만 패킷손실이 버스트(burst)하게 일어나는 환경에서는 네트워크 성능을 저하시키는 여러 문제점을 가지게 된다[9,10]. FIFO 스케줄링 방식에서는 버퍼의 맨 앞에 위치한 HOL(head of line) 패킷을 선택해 전송을 시작하고, 맨 앞에 있는 패킷이 전송을 완료한 후에는 다음 순서의 패킷이 큐의 맨 앞에 위치하여 다음 전송을 준비한다. 맨 앞에 위치한 패킷을 먼저 전송해야 하는 FIFO 스케줄링 특성 때문에 다음과 같은 문제점을 가지게 된다. 예를 들어, 여러 모바일 호스트에게 서비스를 제공하고 있는 어느 기지국에서 모바일 수신 호스트들에게 패킷을 전달하는 경우를 살펴보자. 만약 FIFO 큐의 맨

앞에 있는 패킷이 자신의 목적지인 수신 호스트까지 전송되어지는 과정에서 무선 링크 상에서의 에러로 인하여 전송이 실패한다고 가정하자. 이 경우에 소스 호스트는 모바일 호스트로부터 확인응답 신호(Ack)를 받지 못하기 때문에 전송을 위하여 큐에 저장되어 있는 다음 패킷을 전송하지 못하고, 전송된 패킷에 대한 Ack를 받을 때까지 계속 기다린다. 타임아웃이 발생하면 소스 호스트는 재전송을 시도하게 되는데, 링크 상태가 나쁘게 되면 똑같은 일이 반복되게 된다. 따라서 FIFO 큐의 다음 전송순서에 있는 패킷은 바로 앞의 패킷에 대한 수신 측으로부터의 Ack를 받을 때까지 전송을 할 수 없는 HOL 블로킹 현상으로 인하여 전송할 수 없는 경우가 발생하게 된다. 결국 어떤 하나의 모바일 호스트까지의 링크상의 전송 에러로 인해 나머지 에러가 없는 모든 호스트들로의 전송도 이루어질 수 없게 되므로 네트워크 성능이 감소되는 현상을 나타나게 된다. 이러한 FIFO 큐에서 HOL 블로킹 문제를 그림 2에서 보여주고 있다.

이러한 FIFO 큐를 사용하는 경우에 발생하는 HOL 블로킹 문제를 해결하기 위한 연구가 많이 이루어졌는데, 특별히 [2]에서 제안된 채널상태 의존 패킷(Channel State Dependent Packet: DSCP) 스케줄링 방식이 제안되었다. 이 방식에서는 해당 기지국(base Station: BS)을 통과하는 모든 연결에 대해 각각의 FIFO 버퍼를 따로 할당하고, 최종 목적지인 모바일 호스트까지 링크의 상태에 따라 패킷 스케줄링을 함으로써 HOL 블로킹 문제를 해결한다. 즉, HOL 블로킹이 발생할 경우 소스 호스트는 신뢰성 있는 전송을 위하여 계속 재전송 함으로 인하여 지연이 발생하였으나, CSDP 방식에서는 HOL 블로킹이 일어난 링크에 대한 전송을 중지하고, 전송에러가 발생하지 않고 상태가 좋은 다음 목적지로의 패킷을 전송한다. 이를 위하여 CSDP 스케줄링 메커니즘은 모바일 호스트 각각으로 향하는 독립된 버퍼와 링크 상태

를 모니터링 하는 링크 상태 모니터(LSM: Link State Monitor), 그리고 목적지 모바일 호스트로의 패킷 전송을 위한 패킷 디스패처(dispatcher) 등 3가지 요소들로 구성되어 진다. CSDP 메커니즘의 동작원리는 다음과 같다. 먼저 BS의 HOL에 있는 패킷이 목적지 모바일 호스트로 전송되어진다. 이 때 패킷을 전송한 BS로 전송되어진 패킷의 목적지 모바일 호스트로부터 일정 시간 이내에 링크 계층으로부터의 Ack가 전송되지 못하면, 이는 그 목적지 모바일 호스트까지의 링크가 나쁜 상태(bad state)로 인식되게 된다. 여기에서 나쁜(bad) 상태란 무선 링크 상에서 패킷 손실 특성을 2-state Markov 모델[9]을 이용하여 표현할 때, 무선링크의 에러율로 인하여 패킷 전송에 문제가 생기는 상태를 말한다. 따라서 링크상태 모니터는 그 호스트로의 링크를 bad 상태임을 나타내기 위하여 그 해당 버퍼를 마크(mark)하게 되고, 적절한 스케줄링 알고리즘에 따르는 다음 버퍼의 HOL에 있는 패킷을 전송한다. 따라서 하나의 FIFO 버퍼에서 발생하는 HOL 블로킹에 의한 네트워크의 효율 저하를 줄이게 된다. 이 CSDP 메커니즘에서 링크상태 모니터에서 링크 상태가 bad 혹은 good 인가를 확인하는 것은 BS에서 해당 링크로 패킷을 전송한 이후에 그 링크로부터 링크 계층에서의 Ack가 일정 시간 이내에 오는가의 여부에 따르게 된다. 이처럼 CSDP 메커니즘에서는 흐름(flow) 당 버퍼관리를 제공함으로써 HOL 문제를 해결한다. 그러나 이러한 흐름 당 버퍼관리 방식은 다음과 같은 단점을 가진다. 어떤 네트워크에서 모바일 호스트의 빈번한 이동이 있을 경우, 즉 특정 셀(Cell) 내의 모바일 호스트의 수가 수시로 변하게 되는 경우에 버퍼할당을 자주 해주어야 하는 문제와 동시에 많은 호스트가 이동할 경우, 버퍼의 이용률이 낮아질 수 있다. 이러한 현상은 다음과 같은 이유 때문에 발생할 수 있다. 패킷 손실이 버스트로 일어나는 무선 환경에서 모바일 호스트의 핸드오프(hand-off)로 인하여 셀 내의 모바일 호스트의 수가 수시로 변할 수 있다. 이로 인하여 버퍼의 이용률이 낮아질 수 있다. 즉, TCP 프로토콜을 사용하는 네트워크에서 대역폭의 이용이 사용자간에 공평하게 이루어지는 성질을 가진다. 예를 들어, 처음 어떤 셀에 N개의 목적지 모바일 호스트가 있을 경우, 동일한 환경 하에서 유선 네트워크의 소스 호스트로부터 하나의 모바일 호스트까지 할당되

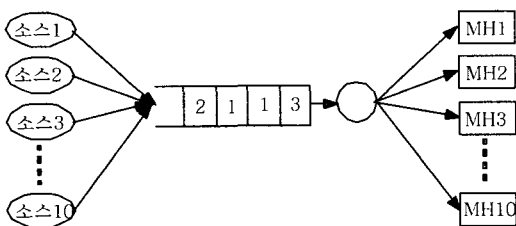


그림 2. FIFO 큐에서의 HOL 블로킹

는 유선 링크 대역폭의 크기는 전체 대역폭의 1/N이다. 그러나 어떤 셀에서 최악의 경우 (N-1)개의 모바일 호스트가 이동한다면, 즉, (N-1) 개의 목적지가 변한다면, 동일한 유선 링크에서 평균 (N-1)/N의 대역폭 크기가 사용을 못하고 낭비하게 된다. 결국 모바일 호스트의 이동이 잦은 셀의 경우, 소스 호스트로부터 목적지 호스트까지의 흐름 당 버퍼의 할당 방식에서는 효율 저하라는 단점을 가지게 된다. Q따라서 본 논문에서는 이러한 문제점을 해결하는 새로운 패킷 스케줄링 메커니즘을 제안된다.

3. 제안된 링크 상태 정보를 이용한 패킷 스케줄링 방안

일반적으로 라우터에서 버퍼의 크기는 네트워크의 특성에 맞추어 적절하게 설정이 되어야한다. 만약 버퍼의 크기가 너무 작으면 패킷의 손실이 크게 되어 네트워크의 성능이 저하될 것이고, 너무 크게 되면 패킷 손실은 줄어들지만 지연이 크게 되어 실시간 트래픽의 경우에는 치명적일 수가 있다. 능동적인 버퍼 관리 메커니즘인 RED(Random Early Detection)를 사용하는 라우터에서 적절한 버퍼 크기는 라우터의 버퍼에 하나 이상의 버퍼가 들어있는 능동적 연결(active connection)에 따라 버퍼를 할당한다. 즉, 흐름 당 버퍼 관리는 연결의 수가 커질수록 복잡해진다. 본 논문에서 제안된 링크 상태 정보를 이용한 스케줄링 메커니즘의 동작은 다음과 같다. 먼저 링크 상태 예측 메커니즘을 통하여 만들어진 각 링크의 상태 정보가 기지국(BS)에 목적지 MH로의 링크별로 벡터 형태로 저장되어진다. 이러한 정보는 과거의 링크 상태(good 혹은 bad)를 기억하여 기억 상태 비트를 저장하여 사용하는 방법과 직접 기지국과 모바일 호스트간의 신호 교환을 이루어지는 방안, 혹은 [11]에서 제안한 것과 같이 가상 소스(Virtual Source)를 사용하는 방법 등이 있다. 그러한 방법에 대한 것은 이 논문에서 다루어지는 영역 밖의 것이므로, 본 연구에서는 이러한 메커니즘이 가능하다고 가정한다. 이들 메커니즘을 사용하여 얻은 예측 정보를 통한 효율적인 스케줄링 방안에 대해 고찰하고자 한다. 제안된 방식에서는 스케줄러에 임시 큐를 둔다. 이 임시 큐의 활용은 다음과 같다. 만약에 HOL에 있는 패킷이 bad 상태인 링크로 향할 경우에는 그

패킷을 임시 큐에 저장해 놓고, HOL 다음에 있는 패킷을 전송함으로써 HOL 블로킹으로 인한 지연을 막을 수 있게 된다. 임시 큐에 저장되어진 패킷은 전송되어질 링크가 good 상태가 되면 우선적으로 전송되어진다. 제안된 스케줄링 메커니즘은 구현이 간단한 원래의 FIFO 패킷 큐와 bad 링크로 향하는 패킷을 임시로 저장하는 크기가 작은 임시 버퍼, 그리고 BS 내의 상태 모니터로 이루어진다. 이에 대한 구성은 그림 2에 보여준다. 두 개의 FIFO 큐 중 하나는 기존의 FIFO 큐이고 다른 하나는 임시 큐이다. 기존 큐는 고정된 호스트에서 BS으로 진입하는 패킷들을 순서대로 두는 큐이고, 임시 큐는 기존 큐보다는 크기가 작은 큐로서 HOL 문제가 발생 시 기존 큐에서 모바일 호스트로부터 링크 계층의 Ack를 기다리고 있는 패킷을 임시 큐에 저장해 두었다가 링크 상태가 good 상태가 되면 전송을 한다. BS의 상태 모니터는 BS과 모바일 호스트 간에 링크 상태를 확인하기 위한 비콘(beacon) 신호를 벡터로 저장해 두어 각각의 모바일 호스트의 링크 상태를 알린다. 아래의 그림에서 HOL에 있는 큐는 모바일 호스트 3으로 전송되어지는 패킷이다. 그러나 모바일 호스트 3으로 향하는 전송 링크가 bad 상태이므로, 이 패킷을 임시 큐로 옮기고, 그 다음의 있는 모바일 호스트 1로 향하는 패킷을 전송한다. 모바일 호스트 1로 향하는 전송 링크의 상태는 good 이므로, 기지국에서는 전송 지연 없이 계속하여 패킷을 전송할 수 있다. 즉, HOL 블로킹 문제를 해결할 수 있다.

이 새로운 스케줄링 메커니즘은 기존의 FIFO 버퍼의 장점을 가지면서 적은 변형을 통해 HOL 문제를 해결할 수 있는 장점을 가진다. 뿐만 아니라 CSDP 프로토콜은 일단 패킷을 목적지로 전송한 후에 Ack의 수신 여부에 따라 링크의 상태를 예측하지만, 본 논문에서 제안된 알고리즘은 미리 링크 상태

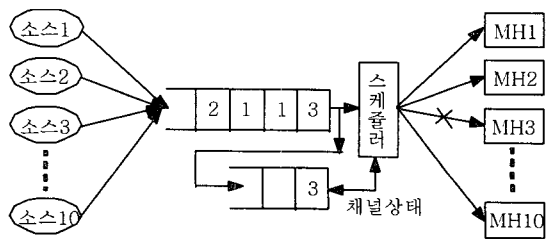


그림 3. 제안된 버퍼링 메커니즘

를 예측하여 패킷의 전송 여부를 결정하기 때문에 보다 효율적인 전송을 얻을 수 있다. 제안된 알고리즘은 다음과 같다.

Step 1: Estimate Link states

Save the information

Step 2: If (link state to destination is bad)

Put the packet into temp buffer

Schedule to send next packet

Step 3: Retry to send packets in temp buffer

그러나 제안된 메커니즘에서 해결해야 할 것은 ‘임시 버퍼로 옮겨진 패킷을 얼마 만에 재전송을 해야 하는가’이다. 즉, 임시 큐로 옮겨진 패킷은 해당 목적지 호스트로 향하는 링크의 상태가 좋으면 재전송하게 되는데 이러한 재전송 시간을 정하는 것이 중요하다. 본 논문에서는 이들 임시 큐에 저장된 패킷을 재전송하는 데 다음과 같은 두 가지 방식을 사용하여, 이들에 대한 성능을 비교한다.

가) 고정된 시간(Fixed Time)을 적용할 경우

이 방식은 네트워크 상황을 고려하지 않고 패킷 전송 시간 혹은 평균 RTT(Round Trip Time)를 고려하여 충분한 전송시간만큼의 일정 시간 이후에 임시 버퍼에 저장되어 있는 패킷을 재전송 하는 방식이다. 이 방식의 특성으로는 동작이 간단하기 때문에 구현이 간단하다는 것이지만, 네트워크 상태를 고려하지 않고 정해진 시간에 재전송이 이루어지기 때문에 상대적으로 네트워크 상태가 혼잡하지 않고 부하의 변화가 적은 네트워크에 적절하다는 것이다.

나) 링크의 평균 bad 상태 시간을 적용할 경우

이 방식에서는 링크의 전송 상태에 따라 재전송을 하는 방식이다. 즉, 전송링크가 확률적으로 평균 bad/good 상태를 반복한다는 것을 고려하여 전송하는 방식이다. 모바일 호스트가 기지국에서 멀어지거나 혹은 핸드오프 등에 의하여 전송링크 상태가 bad인 경우에는 기지국에서 전송되어지는 패킷이 손실될 확률이 크므로 링크가 good 상태로 변환 이 후에 보내는 것이 에러 없는 전송이 될 것이다. 따라서 어떤 전송 링크가 bad이기 때문에 전송을 하지 않고 임시 버퍼에 들어 온 패킷을 링크의 평균 bad 상태 시간 동안에는 버퍼링 하고, 그 이후에 보내게 된다. 이 방식에서는 링크의 상태에 따라 즉, 네트워크의

상태에 따라 적응적으로 패킷의 전송이 이루어지기 때문에, 네트워크 자원의 효율적인 사용과 전체 전송 효율을 높일 수 있다는 장점을 가진다. 이 방식에서 시간적으로 변하는 무선링크 채널은 2-state 마르코프 체인(Markov chain)을 사용하여 해석할 수 있다. 일반적으로 에러가 있는 무선 전송로는 Gilbert 모델인 ON-OFF 방식이 많이 사용되어지는데 이 링크의 상태 모델은 아래의 그림 4와 같다. 그림에서 0 상태는 “Good” 상태를, 1은 “Bad” 상태를 나타낸다. 이러한 Markov 모델에서 각 링크 상태에 대한 정상상태 확률은 다음과 같이 표현되어 진다.

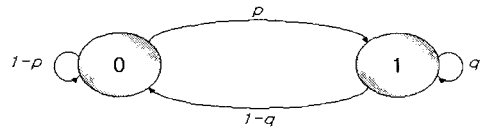


그림 4. 무선링크에 대한 Gilbert 2-state 손실 모델

$$\pi_B = p/(1-q+p), \quad \pi_G = (1-q)/(1-q+p) \quad (3)$$

여기서 π_B 와 π_G 는 각각 링크의 “Bad” 상태와 “good” 상태의 정상상태 확률을 나타낸다. 위의 식을 통하여 각 상태에서의 평균 지연시간을 구할 수 있다. 즉, Bad, Good 상태의 평균 길이를 각각 L_B 와 L_G 로 표현한다면, 다음의 식으로부터 구할 수 있다.

$$L_B = 1/(1-q), \quad L_G = 1/p \quad (4)$$

본 연구에서 이들 기간은 해석의 용이함과 기존의 연구들에서의 결과에 따라 지수함수(Exponential) 분포를 가진다고 가정한다. 위에서 구한 식을 통하여 얻은 값을 사용하여, 임시 큐에 저장되어 있는 패킷을 재전송하게 된다.

4. 성능 비교

제안된 메커니즘의 성능 비교를 위하여 위의 그림 3과 같은 네트워크 환경을 설정하였다. 즉, 그림에서 볼 수 있듯이 10개의 소스 호스트와 10개의 모바일 호스트가 하나의 기지국을 통하여 연결되어 있다. 따라서 모든 호스트에서 전송되어진 패킷은 하나의 기지국을 통하여서 서로 다른 모바일 호스트로 전송되어진다. 기존의 다른 알고리즘과 본 논문에서 제안된

알고리즘과의 성능비교를 위하여 사용되어진 파라미터들은 다음과 같다. 시뮬레이션 모델은 C++을 사용하여 구현하였다.

본 연구에서는 기지국의 FIFO 큐에서의 오버플로우로 인한 패킷의 재전송 때문에 일어나는 지연을 고려하지 않기 때문에, 기지국의 FIFO 큐의 크기는 무한 크기를 가진다고 가정한다. 또한 고정된 시간을 적용한 경우는 2RTT 값인 33.6 ms를 사용하였고, ON-OFF 버스트를 사용한 경우는 bad 상태에서의 에러율은 0.8, 그리고 good 상태에서의 에러율은 0.0으로 가정하였다. 또한 아래의 실험에서 표 1에서 보는 바와 같이 평균 Good 상태 기간과 Bad 상태 기간을 대략 10배 차이가 나도록 선택하였다.

그림 5는 기존의 FIFO 스케줄링 방식과 CSDP 방식, 그리고 본 논문에서 제안한 두 가지 스케줄링 메커니즘들과의 효율 비교를 보여준다. 그림에서 x축은 TCP를 사용하는 FTP 세션의 수를 나타내고, y축은 하나의 TCP 세션에서 단위 시간(초) 당 전송되어진 평균 데이터의 양 즉, 효율을 나타낸다. 그림에서 볼 수 있듯이 FIFO 방식은 HOL 블로킹으로 인하여 다른 세 개의 메커니즘에 비하여 효율의 저하를

표 1. 사용되어진 네트워크 파라미터들

유선링크 대역폭	1 Mbps
유선링크 전파지연시간	1.6 ms
무선링크 대역폭	1 Mbps
최대세그먼트 크기(MSS)	512 바이트
수신윈도우 크기	8 K
FTP 크기	800 K
TCP 연결 수	1~10
평균 Bad 상태기간	100~1,000 ms
평균 Good 상태기간	1~10 sec

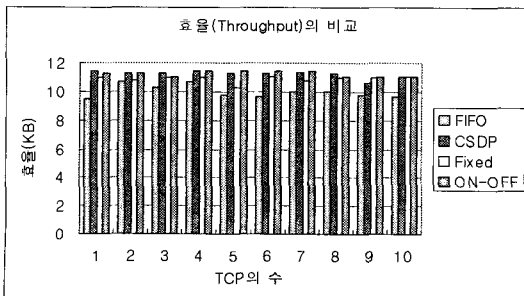


그림 5. 제안된 메커니즘의 성능 비교

보인다. 뿐 만 아니라 링크 상태를 예측하여 전송하는 세 가지 스케줄링 방식에서 Fixed 방식은 다른 방식에 비하여 약간의 낮은 효율을 가진다. 이것은 Fixed 방식에서는 HOL으로 인하여 임시 큐에 들어있는 패킷을 단순히 일정시간 후에 전송하기 때문에 발생한다. 아래의 그림 7은 각 방식에서 하나의 TCP 소스가 800 KB 크기의 데이터를 전송하는데 걸리는 평균 지연을 보여주고 있다.

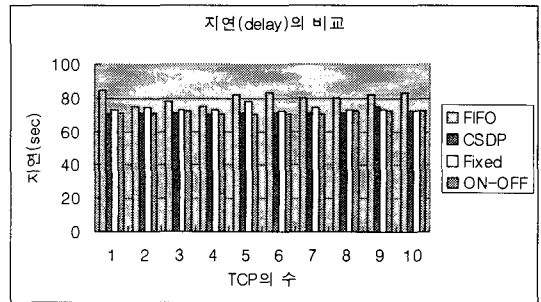


그림 6. 제안된 메커니즘의 지연(delay) 비교

위의 그림 5와 6에서 볼 수 있듯이 제안된 방식은 재전송이 빈번한 에러율이 많은 네트워크 환경일수록 더욱 더 효과적임을 알 수 있다.

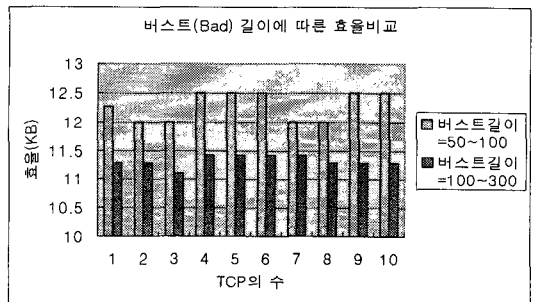


그림 7. 버스트의 길이에 따른 효율의 비교

위의 그림 7은 ON-OFF 스케줄링 방식에서 여러 버스트 길이에 따른 효율의 비교를 보여 준다. 효율 비교를 위하여 각 TCP 소스들은 동일한 버스트 길이를 갖지 않고 임의의 값을 가진다. 위의 그림에서 볼 수 있듯이 Bad 버스트의 길이가 길수록 낮은 효율을 얻는다. 따라서 제안된 메커니즘은 사용되어지는 파라미터들에 많은 의존도를 가진다. 따라서 이에 대한 추후 연구가 필요하다.

5. 결 론

무선망의 기지국에서 모바일 호스트로 패킷을 전송하는 경우 구현이 간단한 FIFO 큐를 사용한다. 그러나 무선 데이터링크의 전송 특성과 숨은 터미널 문제, 그리고 이동성들로 인한 링크 상태의 변화로 말미암아 HOL 블로킹 문제를 가지게 된다. 이러한 문제를 해결하기 위하여 링크 상태 예측하여 이에 따르는 전송 메커니즘이 필요하다. 본 논문에서 제안된 새로운 스케줄링 방식은 링크 상태 예측 메커니즘을 통하여 얻은 링크 상태 정보를 이용하여 기존의 FIFO 큐에 새로운 임시 FIFO 큐를 첨가하여 HOL 문제를 해결하며, 네트워크 적응적인 방법이므로 전송 효율을 높일 수 있다는 장점을 가진다. 또한 제안된 메커니즘은 구조가 간단할 뿐 만 아니라 기존의 TCP 프로토콜이나 모바일 호스트의 특별한 수정 없이 동작할 수 있고, 적절한 파라미터의 설정으로 TCP의 성능을 향상시킬 수 있는 장점을 가진다. 그러나 제안된 방식은 사용되는 파라미터들과 예측 메커니즘에 많은 영향을 받기 때문에 이에 대한 추후 연구가 필요하다.

참 고 문 헌

- [1] A. Baker and B. R. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts," Proceedings of distributed Computing System, pp 136-143, June 1995.
- [2] K. Brown and S. Singh, "M-TCP: TCP for Mobile Cellular Networks," *ACM Computer Communications Review (CCR)*, pp.19-43, vol. 27, NO. 5, Jul. 1997.
- [3] H. Balakrishnan, S. Seshan, E. Amir, and R. H. Katz, "Improving TCP/IP Performance over Wireless Networks," *Proc. IEEE Distributed Computing Systems*, pp.365-373, May 1997.
- [4] K. Ratnam and I. Matta, "WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links," Proceedings of the third IEEE Symposium on Computers and Communications, pp.74-78, June 1998.
- [5] Chunlei Liu, Raj Jain, "Using Congestion Coherence to Enhance TCP over Wireless Links," *IEEE/ACM Transactions of Networking*, June 2001.
- [6] Benyuan Liu, Dennis L. Goeckel and Don Towsley, "TCP-Cognizant Adaptive Forward Error Correction in Wireless Network," *IEEE INFOCOM 2002*.
- [7] Chadi Barakat and Eitan Altman, "Bandwidth tradeoff between TCP and link-level FEC," in *Processing of IEEE International Conference on Networking*, Colmar, France, Jul 2001.
- [8] P. Bhangwat, P. Bhattacharya, A. Krishna and S. Tripathi. "Using Channel State Dependent Packet Scheduling to improve TCP throughput over wireless LANs," *Wireless Networks*, pp.91-102, 1997.
- [9] G. T. Nguyen, R. H. Katz, B. Noble and M. Satyanarayanan. "A Trace-based Approach for Modeling Wireless Channel Behavior," *Proceeding of the Winter Simulation Conference*, pp.597-604, December 1996.
- [10] D. Duchamp and N. Reynolds, "Measured performance of a wireless lan," in 17th Conference on Local Computer Networks, IEEE Press, pp. 494-499 Sep. 1992.
- [11] M. Barry, Andrew T. Cambell, and Andra Veres, "Distributed Control Algorithms for Service Differentiation in Wireless Packet Networks," *IEEE INFOCOM 2001*.



김 성 철

1995년 6월 뉴욕 Polytechnic Univ. 박사
1994년 7월~1997년 2월 (주) 삼성전자 수석연구원
1997년 3월~현재 상명대학교 소프트웨어학부 부교수

관심분야: 멀티미디어 통신, 인터넷 프로토콜, 네트워크 성능분석, 무선랜



이 진 영

1998년 상명대학교 정보과학과 졸업(학사)
2000년 상명대학교 정보처리학과 졸업(석사)
2000년~현재 상명대학교 컴퓨터학과(박사)

관심분야: WLAN, Multimedia communication, High-Speed Network, Ubiquitous Computing, and Internet QoS