

TCP 송수신자간의 큐사용률 추정을 이용한 송신자 기반의 패킷손실 구별기법

박미영*, 정상화**, 이윤성***

A Sender-based Packet Loss Differentiation Algorithm based on Estimating the Queue Usage between a TCP sender/receiver

Mi-Young Park*, Sang-Hwa Chung**, Yun-Sung Lee***

요약

멀티홉 무선 네트워크에서 TCP가 동작하는 경우, 유선 네트워크와 무선 네트워크의 서로 다른 특성으로 인하여 TCP의 심각한 성능저하가 초래된다. 이것은 TCP가 무선오류로 인해 발생하는 패킷손실을 혼잡으로 인해 발생한 패킷손실로 간주하여 데이터 전송률을 불필요하게 감소시키기 때문이다. 이러한 성능저하를 피하기 위해서 혼잡손실과 무선손실을 구별하는 많은 기법들이 연구되어 왔으나, 이들 기법들은 무선손실에 대한 탐지정확도가 기대만큼 높지 않거나, 무선손실에 대한 탐지정확도가 높으면 혼잡손실에 대한 정확도가 낮아지는 경향을 보인다. 본 논문은 혼잡손실에 대한 탐지정확도의 희생을 최소화하면서, 무선손실에 대한 탐지정확도를 높이는 송신자 기반의 패킷손실 구별기법을 제안한다. 본 기법은 네트워크 혼잡과 상호 관련성이 높은 TCP 송·수신자간의 큐 사용률을 추정하고, 추정된 큐 사용률과 특정 임계값을 비교하여 혼잡손실과 무선손실을 구별한다. 네트워크 시뮬레이터인 QualNet을 이용한 실험에서는 기존 기법과 제시된 기법간의 혼잡손실에 대한 탐지정확도와 무선손실에 대한 탐지정확도를 구분하여 비교평가하고, 홉 수 증가에 따른 성능향상을 비교평가 한다. 실험 결과는 멀티홉 무선 네트워크상에서 본 기법이 가장 높은 탐지정확도를 가질 뿐만 아니라 TCP의 성능을 가장 높게 향상시킴을 보인다.

▶ Keyword : 멀티홉 무선 네트워크, 패킷손실 구별기법, TCP, 큐 사용률

Abstract

When TCP operates in multi-hop wireless networks, it suffers from severe performance degradation due to the different characteristics of wireless networks and wired networks. This is because TCP reacts to wireless packet losses by unnecessarily decreasing its sending rate assuming the losses as congestion losses. Although several loss differentiation algorithms (LDAs) have been proposed to avoid such performance degradation, their

• 제1저자 : 박미영 교신저자 : 정상화

• 투고일 : 2010. 10. 21, 심사일 : 2010. 11. 19, 게재확정일 : 2010. 12. 07.

*부산대학교 컴퓨터공학과 연구교수 ** 부산대학교 컴퓨터공학과 교수 ***부산대학교 컴퓨터공학과 박사과정

※ 이 논문은 부산대학교 자유과제 학술연구비(2년)에 의하여 연구되었음

detection accuracies are not high as much as we expect. In addition the schemes have a tendency to sacrifice the detection accuracy of congestion losses while they improve the detection accuracy of wireless losses. In this paper, we suggest a new sender-based loss differentiation scheme which enhances the detection accuracy of wireless losses while minimizing the sacrifice of the detection accuracy of congestion losses. Our scheme estimates the rate of queue usage which is highly correlated with the congestion in the network path between a TCP sender and a receiver, and it distinguishes congestion losses from wireless losses by comparing the estimated queue usage with a certain threshold. In the extensive experiments based on a network simulator, QualNet, we measure and compare each detection accuracy of wireless losses and congestion losses, and evaluate the performance enhancement in each scheme. The results show that our scheme has the highest accuracy among the LDAs and it improves the most highly TCP performance in multi-hop wireless networks.

▶ Keyword : multi-hop wireless networks, packet loss differentiation algorithm, TCP, queue usage

1. 서론

유선 네트워크상의 신뢰성 있는 데이터 전송 프로토콜로써, 가장 광범위하게 사용되는 프로토콜은 TCP (Transmission Control Protocol)[1, 2]이다. 동적으로 끊임없이 변화하는 네트워크 환경에서, 안정적인 데이터 전송을 보장하는 TCP는 현재 인터넷 트래픽의 90%이상을 차지하고 있다. 이러한 TCP를 기반으로 하는 웹브라우저, 이메일, 파일전송 등과 같은 인터넷 애플리케이션들이 무선 네트워크로 이동함으로써, TCP는 무선 네트워크상에서도 가장 널리 사용될 전송 프로토콜로 주목받고 있다.

그러나 무선 네트워크에서 TCP가 동작하는 경우, 유선 네트워크와 무선 네트워크의 서로 다른 특성으로 인하여 TCP의 성능이 심각하게 저하된다. 특히 무선 네트워크에서 홉 수가 증가할수록 TCP의 성능은 비례적으로 감소한다[3, 4]. 이러한 성능저하는 패킷손실이 네트워크의 혼잡으로 초래된다는 TCP의 기본 전제조건에서 비롯된다. 다시 말해서, 패킷손실을 알리는 세 개의 연속된 중복 ACK (Acknowledgement)가 수신되면, TCP는 혼잡으로 인하여 패킷이 손실되었다고 판단하고, 자신의 전송속도를 절반으로 줄인다[1]. 비록 이러한 전제 조건이 전송에러가 거의 없는 유선 네트워크에서는 잘 동작하였으나, 잡음, 페이딩, 간섭 등에 의한 높은 패킷 손실률을 가지는 무선 네트워크에는 적합하지 않다.

잡음, 간섭 등과 같은 무선오류 (wireless errors)로 인해 발생한 패킷손실에 대한 TCP의 비람직한 동작은 자신의 전송속도를 줄이지 않고, 단지 손실된 패킷을 재전송하는 것이다. 그러나 기존의 TCP는 모든 패킷손실을 네트워크 혼잡으로 초래되었다고 간주하고, 무선오류로 인해 발생한 패킷손실에 대해

서도 자신의 혼잡윈도우 (congestion window)를 불필요하게 감소시켜 TCP의 성능저하를 초래한다. 이러한 성능저하를 피하기 위해서는 TCP가 혼잡으로 인해 발생한 패킷손실과 무선 오류로 인해 발생한 패킷손실을 구분하여, 원인에 따라 데이터 전송률을 적절하게 제어해야 한다. (이하 논문에서는 네트워크 혼잡으로 인해 발생한 패킷손실을 혼잡손실이라고 하고, 무선 오류로 인해 발생한 패킷손실을 무선손실이라 칭한다.)

무선 네트워크상의 TCP 성능을 개선하기 위한 기존의 패킷손실 구별 알고리즘 (LDA, Loss Differentiation Algorithm)으로는 Veno[5], West[6], JTCP[7], 그리고 RELDS[8] 등이 있다. 그러나 이들 기법은 무선손실에 대한 탐지정확도가 기대만큼 높지 않거나, 혼잡손실에 대한 탐지정확도가 크게 감소하는 경향이 있다. 특히, 이 기법들은 마지막 홉인 TCP 수신자만 무선으로 통신하는 셀룰러 네트워크 (cellular network)를 대상으로 제안되었기 때문에, 멀티홉 무선 네트워크(multi-hop wireless network)에서 탐지정확도가 어느 정도인지 평가되지 않았다.

본 논문은 혼잡손실에 대한 탐지정확도의 희생을 최소화하면서, 무선손실에 대한 탐지정확도를 높이는 송신자 기반의 중단간 패킷손실 구별기법(a sender-based end-to-end loss differentiation algorithm)을 제안한다. 본 기법은 네트워크 혼잡과 상호 관련성이 높은 TCP 송/수신자간의 큐 사용률을 추정하고, 추정된 큐 사용률과 특정 임계값을 비교하여 혼잡손실과 무선손실을 구별한다.

본 기법은 네트워크의 혼잡상태를 판단하기 위해서, 패킷이 전송되면서 거치는 모든 중간 노드상의 큐를 하나의 큐로 가정하여 큐 사용률을 추정한다. 패킷손실을 알리는 세 개의 연속된 중복 ACK가 수신되면, TCP 송신자는 추정된 큐 사용률과 특정 임계값을 비교하여 손실의 원인을 판단한다. 즉, 추정된 큐 사용률이 특정 임계값(50%) 이상이면 손실된 패킷을

혼잡손실로 판단하고, 그렇지 않으면 무선손실로 판단한다.

본 기법이 추정한 큐 사용률은 네트워크의 혼잡과 상호 관련성이 높기 때문에, 혼잡과 관련된 패킷손실과 그렇지 않는 패킷손실을 보다 더 정확하게 구별할 수 있는 이점이 있다. 또한, 본 기법은 TCP 수신자나, 중간 노드들로부터 도움을 필요로 하지 않는 장점이 있다. 즉, 본 기법은 TCP의 시맨틱을 그대로 유지하면서 TCP의 송신자 측에서만 동작하므로, 현재의 네트워크에 손쉽게 적용할 수 있는 이점이 있다.

본 실험에서는 네트워크 시뮬레이터인 QualNet 4.5[9]를 이용하여 본 기법과 기존의 LDA 기법들 간의 탐지정확도와 성능향상을 비교평가 하였다. 실험에서는 본 기법이 추정한 큐 사용률과 실제 큐 사용률 간에 높은 상관성이 있음을 보이고, 기존의 기법들과의 비교에서 가장 높은 탐지정확도를 가짐을 보인다. 또한 본 기법을 TCP에 적용했을 때, 기존의 기법들보다 가장 높게 TCP의 성능을 향상시킴을 보인다.

이어지는 2절에서는 기존의 LDA 기법들을 간단히 소개하고, 3절에서는 큐 사용률을 추정하여 패킷손실의 원인을 탐지하는 본 기법을 제시한다. 그리고 4절에서는 본 기법과 기존의 LDA 기법을 비교 실험한 결과를 보이고, 마지막으로 5절에서 결론을 내린다.

II. 연구 배경

멀티홉 무선 네트워크상에서 무선손실로 인한 TCP 성능저하를 해결하고자, 기존의 많은 연구들은 혼잡손실과 무선손실을 구별하는 다양한 방법들을 모색하였다. 이들 접근방식을 크게 두 가지로 구분하면, 네트워크 계층(Network layer)의 도움을 받아 패킷손실의 원인을 구별하는 방식[2]과 전송계층(Transport layer)에서 유효한 정보만을 이용하여 패킷손실의 원인을 구별하는 방식[5, 6, 7, 8]이 있다. 이들 중에서, 네트워크 계층의 도움을 받는 방식은 TCP의 시맨틱(semantic)을 지지 못하기 때문에, 현재의 네트워크 환경에 적용하는데 많은 어려움이 있다. 반면 네트워크 계층의 도움 없이, 전송계층에서 패킷손실의 원인을 구별하는 방식은 TCP의 시맨틱을 그대로 유지하는 장점이 있다. TCP의 시맨틱을 그대로 유지하는 기존 연구로는 Venol[5], West[6], JTCP[7], 그리고 RELDS[8] 등이 있다.

TCP Venol[5]는 Vegas[10]의 기법을 이용하여 버퍼에 축적된 패킷의 수 (N)를 추정하고, 패킷손실의 원인을 판별한다. Venol에서 N은 아래와 같이 계산된다.

$$N = \frac{cwnd}{SRTT} \times (SRTT - RTT_{min}) \dots \dots \dots (1)$$

식 (1)에서 *cwnd*는 혼잡윈도우 (congestion window) 크기를 나타내며, *SRTT*는 EWMA (exponentially weighted moving average)로 계산된 *RTT* (round trip time)이다. 패킷이 손실되면, Venol은 식 (1)에서 계산된 *N*과 특정 임계값 β 를 비교한다 (Venol은 실험을 통해서 β 로 3을 제시하였다). 이때, *N*이 β 보다 작으면 무선손실로 간주하고, *N*이 β 보다 크면 혼잡손실로 간주한다.

Yang[6]이 제안한 TCP Westwood+에서는 패킷손실의 원인을 결정하기 위해서 Cen과 Voeler가 제안한 Spike 기법[2]을 사용하였다. 다만 Spike 기법에서는 패킷이 송신자에서부터 수신자까지 전송되는데 걸리는 시간인 *ROTT* (relative one-way trip time)를 사용한 반면에, Yang은 TCP 송신자 측에서 *RTT*를 사용하였다. *RTT*를 이용하여, Yang은 TCP 연결 상태를 검사하여, 혼잡상태에서 패킷이 손실되면 혼잡손실로 탐지하고, 그렇지 않으면 무선손실로 탐지한다. TCP 연결 상태를 검사하는데 필요한 두 개의 임계값은 *B_{spikestart}*와 *B_{spikeend}*이며, 이들은 다음과 같이 계산된다.

$$B_{spikestart} = RTT_{min} + 0.4 \times (RTT_{max} - RTT_{min}) \dots \dots \dots (2)$$

$$B_{spikeend} = RTT_{min} + 0.05 \times (RTT_{max} - RTT_{min}) \dots \dots \dots (3)$$

식 (2)와 (3)에서 *RTT_{min}*과 *RTT_{max}*는 측정된 *RTT*들 중에서 최소값과 최대값을 나타낸다. 측정된 *RTT*가 *B_{spikestart}*보다 크면, TCP 연결은 spike 상태가 된다. 그리고 spike 상태일 때, *RTT*가 *B_{spikeend}*보다 작으면 TCP 연결은 spike 상태를 떠나게 된다. 이 기법은 spike 상태의 모든 패킷손실을 혼잡손실로 간주하고, non-spike상태의 패킷손실을 무선손실로 간주한다.

Wu와 Chen이 제시한 a jitter-based TCP (JTCP)[7]는 무선손실과 혼잡손실을 구분하기 위해서, 한번의 *RTT* 내에서 TCP의 평균 jitter 값(*J_r*)을 계산한다. 이때 *J_r*값은 큐 상태를 나타내며 다음과 같이 계산된다.

$$J_r = \frac{(R_{newest} - R_{oldest}) \times (S_{newest} - S_{oldest})}{R_{newest} - R_{oldest}} \dots \dots \dots (4)$$

식 (4)에서 *S_{newest}*와 *R_{newest}*는 가장 최근에 ACK된 패킷이 송신된 시간과 수신된 시간을 나타내며, *S_{oldest}*와 *R_{oldest}*는 각각 한 번의 *RTT*내에서 가장 오래된 패킷의 송신 시간과 수신시간을 나타낸다. TCP 송신자가 연속된 세 개의 중복 ACK를 수신하면, 이 기법은 세 개의 중복된 ACK를 받는 시간이 한 번의 *RTT*보다 크고, 이때 계산된 *J_r*이 *cwnd* 역수 값보다 크지를 비교한다. 만약에 두 조건이 모두 만족되면, 이 기법은 해당 손실을 혼잡손실로 간주하고, 그렇지 않으면 무선손실로 간주한다.

Lim과 Jang이 제안한 RELDS (robust end-to-end loss differentiation scheme) [8]는 RTT 최소값과 건본값으로 구성된 함수를 이용하여 혼잡손실과 무선손실 구별한다. 패킷 손실의 원인을 구별하는 함수는 다음과 같다.

$$\frac{RTT - SRTT}{RTT_{dev}} > 2 \times \left(\frac{RTT_{min}}{RTT} \right)^3 - 1 \dots \dots \dots (5)$$

$$(RTT_{dev} = (3/4)RTT_{dev} + (1/4)|RTT - SRTT|)$$

패킷이 손실됐을 때, 위의 주어진 식 (5)의 조건이 만족되면, 이 기법은 혼잡손실로 간주하고, 그렇지 않으면 무선손실로 간주한다.

이러한 기존 기법들은 무선손실을 탐지하여, TCP가 불필요하게 데이터 전송률을 감소시키는 것을 방지함으로써, 무선 네트워크상의 TCP 성능을 향상시킨다. 그러나 이 기법들의 무선손실 탐지정확도가 여전히 낮거나, 혼잡손실의 탐지정확도가 매우 낮아지는 경향이 있다. 특히 이 기법들은 마지막 홉인 TCP 수신자만이 무선으로 통신하는 셀룰러 네트워크 (cellular network)를 대상으로 제안되었기 때문에, 이들 기법의 정확도가 멀티홉 무선 네트워크에서는 어떻게 변화하는지 평가되지 않았다.

III. 제안된 패킷손실 구별기법

본 논문에서는 멀티홉 무선 네트워크에서도 혼잡손실과 무선손실에 대한 탐지정확도가 모두 높은 송신자 기반의 패킷손실 구별기법을 제안한다. 본 기법은 혼잡손실과 무선손실을 정확하게 구별하는 것보다는 패킷손실과 네트워크 혼잡과의 관련여부를 알아내는 데 초점이 있다. 왜냐하면, 네트워크 혼잡상태에서도 무선오류로 인하여 패킷이 손실될 수 있으며, 이 경우 TCP는 해당 패킷을 혼잡손실로 간주하여, 혼잡을 제어할 필요가 있기 때문이다. 따라서 본 연구는 혼잡으로 인해 발생한 패킷손실을 식별하고, 혼잡으로 인해 발생한 패킷손실이 아닌 모든 손실을 무선손실로 간주한다.

네트워크상의 혼잡은 큐의 사용률과 상관성이 깊다. 큐 사용률이 어떤 임계값을 초과하는 경우, 큐 정책에 의해서 패킷들은 의도적으로 버려진다. 따라서 본 기법의 핵심 아이디어는 패킷이 전송되면서 거치는 모든 중간 노드상의 큐를 하나의 큐로 가정하여 큐 사용률을 추정하는 것이다. 그리고, 추정된 큐 사용률이 설정된 임계값을 초과하는 경우, 손실된 패킷을 혼잡손실로 간주하고, 임계값 이하이면 무선손실로 간주한다.

TCP의 시멘틱을 유지하면서 큐 사용률을 추정하기 위해서

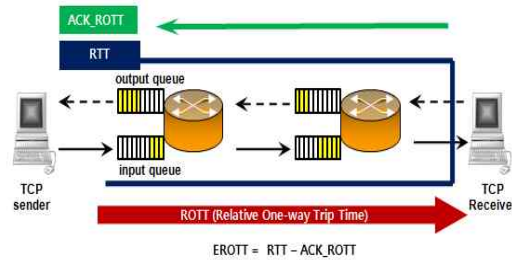


그림 1. EROTT 측정
Fig. 1. Measuring EROTT

는, 혼잡 윈도우 크기 (cwnd), RTT (round trip time), ROTT (relative one-way trip time)와 같은 전송 계층에서 측정이 가능한 정보만을 사용해야 한다. 기존 연구 [11]는 실험을 통하여, ROTT가 RTT보다 큐의 지연시간을 보다 정확하게 반영함을 보였다. 즉, RTT는 데이터 패킷이 전송되는 정방향의 큐 지연과 ACK 패킷이 전송되는 역방향의 큐 지연을 모두 포함하는 반면, ROTT는 정방향의 큐 지연만을 포함하기 때문에 큐 사용률의 측정에 더 적합하다[11].

그러나 TCP 송신자 측에서는 ROTT를 측정할 수 없을 뿐만 아니라, 이전 연구에서 보였듯이 RTT를 단순히 절반으로 나누어서 ROTT를 계산할 수 없다[11]. 본 연구에서는 TCP 수신자가 측정된 ROTT를 ACK가 보내질 때 마다 TCP 송신자로 보내는 시도를 하였다. 이 경우, TCP 송신자가 수신한 ROTT는 이미 오랜 시간이 지난 정보로 변하여, TCP 수신자 측에서 측정된 ROTT 만큼 유용하지 않았다.

따라서 본 기법에서는 TCP 송신자 측에서 아래의 식을 이용하여 ROTT를 추정하였다.

$$EROTT = RTT - ACK_ROTT \dots \dots \dots (6)$$

식 (6)에서 ACK_ROTT는 ACK 패킷이 수신자에서부터 송신자까지 전송되는 시간을 나타내며, 본 기법이 TCP 송신자 측에서 추정된 ROTT를 EROTT (estimated relative one-way trip time)로 나타내었다. 식 (6)을 그림으로 표현하면 그림 1과 같다. 그림 1은 식 (6)에서 사용된 RTT, ACK_ROTT, ROTT 그리고 EROTT간의 관계를 보여 주고 있다. 그림에서와 같이 RTT는 정방향과 역방향에서의 모든 큐 지연시간을 포함한다. 이때 정방향의 큐 지연시간 (ROTT)만을 알기 위해서는 측정된 RTT에서 역방향의 큐 지연시간 (ACK_ROTT)을 제거하면 된다. 즉, RTT에서 ACK_ROTT를 제거함으로써 ROTT에 근접한 EROTT를 측정할 수 있다.

그림 2는 측정된 EROTT를 이용하여 TCP 송신자가 큐 사용률을 추정하는 알고리즘을 보인다. 알고리즘에서 사용된

minEROTT와 maxEROTT는 추정된 EROTT들 중에서 가장 작은 EROTT와 가장 큰 EROTT를 나타내며, minEROTT는 큐 사용률이 1%일 때의 EROTT, maxEROTT는 큐 사용률이 100%일 때의 EROTT라 가정한다.

변수 *estimated_queue*는 EROTT를 이용하여 추정된 큐 사용률을 나타내며, 1%에서 100% 사이의 값을 가진다. 그리고 변수 *gap_max_min*는 maxEROTT와 minEROTT간의 간격을 수치화 한 것으로, 두 변수 minEROTT와 maxEROTT가 적절한 최소/최대 EROTT를 가졌는지를 판단하는 기준으로 사용된다.

즉, TCP 연결이 시작된 직 후에는 minEROTT와 maxEROTT가 적절한 최소값과 최대값을 가지지 못한다. 그러나 시간이 지남에 따라, 이들 두 변수는 추정된 EROTT들 중에서 큐 사용률을 잘 반영하는 적절한 최소값과 최대값을 가지게 된다. 따라서 EROTT를 이용하여 큐 사용률을 측정하기 위해서는 maxEROTT와 minEROTT가 언제 적절한 최대/최소 값을 가지는지를 알 필요가 있다.

이를 위해서, 알고리즘의 1번째 줄에서 maxEROTT와 minEROTT간의 간격(*gap_max_min*)을 측정한다. 그리고 알고리즘의 3번째 줄에서, maxEROTT가 minEROTT보다 3배 이상인지를 비교한다. 본 알고리즘에서는 *gap_max_min*이 3보다 크면 두 변수 maxEROTT와 minEROTT가 적절한 최대/최소값을 가졌다고 가정한다. 이때, 알고리즘의 4번째 줄에서와 같이, maxEROTT에 대한 현재 추정된 EROTT의 비율을 계산하여, 큐 사용률 (*estimated_queue*)를 추정한다.

만약에 maxEROTT가 minEROTT보다 3배 이상 크지 않으면, TCP 연결은 여전히 시작 시점에 있다고 가정하고, 알고리즘의 6번째 줄에서와 같이, 큐 사용률이 $\beta\%$ 이상을 넘지 않도록 하였다. (β 에 대한 적절한 값을 찾기 위해서, 여러 값을 대상으로 실험을 진행하였고, 30%로 설정했을 때 가장 좋은 결과를 보였다.)

알고리즘의 9번째 줄에서는 전송중인 데이터의 양 (*flightsize*)을 큐 사용률 계산에 사용한다. 이것은 네트워크상에 전송 중인 데이터의 양이 많을수록, 큐 사용률 또한 커지는 것을 반영하기 위해서이다. 즉, 알고리즘의 9번째 줄에서는, 전송 중인 패킷이 하나씩 늘 때 마다, 큐 사용률은 1%씩 증가하도록 하였다.

알고리즘의 10번째 줄에서는 EROTT를 이용하여 혼잡 시작점을 계산한다. 다시 말해서, 추정된 큐 사용률 (*estimated_queue*)이 50%에서 60%사이일 때의 EROTT에 대한 평균값을 계산하여, 해당 값을 혼잡 시작점(*congestion_start_point*)으로 설정하였다. 따라서, 혼잡손실과 무선손실을 구별하기 위한 기준

```

01:  $gap\_max\_min = \frac{\max EROTT}{\min EROTT}$ 
02:
03: if ( $gap\_max\_min > 3$ )
04:    $estimated\_queue = \frac{EROTT}{\max EROTT} \times 100$ 
05: else
06:    $estimated\_queue = \frac{EROTT}{\min EROTT} \times \beta$ 
07: end if
08:
09:  $estimated\_queue = estimated\_queue + \frac{flightsize}{packetsize}$ 
10:  $congestion\_start\_point =$  the average of EROTTs
    when  $estimated\_queue$  ranges from 50% to 60%

```

그림 2. 큐 사용률 추정 알고리즘
Fig. 2. Algorithm to estimate the queue usage

으로는 추정된 큐 사용률(*estimated_queue*)과 혼잡 시작점 (*congestion_start_point*)이 사용된다.

즉, TCP 송신자가 패킷손실을 알리는 세 개의 연속된 중복 ACK를 받으면, 본 기법은 추정된 큐 사용률 (*estimated_queue*)이 설정된 임계값 (50%) 이상인지, 또는 현재 측정된 EROTT가 혼잡 시작점(*congestion_start_point*)보다 큰지를 비교한다. 만약에 두 조건들 중에 하나라도 만족되면, 본 기법은 네트워크가 혼잡한 상태라 간주하고, 손실된 패킷을 혼잡손실로 탐지한다. 만약에 *estimated_queue*가 50%이하이고, EROTT가 *congestion_start_point*보다 작으면, 손실된 패킷을 무선손실로 간주한다.

IV. 시뮬레이션 및 결과분석

1. 시뮬레이션 환경 및 시나리오 설계

본 실험에서는 네트워크 시뮬레이터인 QualNet 4.5[9]를 이용하여, 본 기법과 기존 LDA 기법들을 비교·평가하였다. 우선, 다양하고 동적으로 변화하는 네트워크 환경을 반영하여 LDA 기법들을 평가하고자, 홉 수, 패킷 손실률 등과 같은 네트워크 파라미터 값을 서로 다르게 설정함으로써 255개의 서로 다른 시나리오를 설계하였다. <표 1>은 모든 시나리오에서 공통적으로 사용된 네트워크 파라미터를 나타내며, 그림 3은 실험에 사용된 멀티홉 무선 네트워크 토폴로지를 보인다.

본 실험에서는 제안된 기법이 멀티홉 무선 네트워크에서도 혼잡손실과 무선손실에 대한 탐지정확도가 모두 높은 패킷손실 구별기법임을 보이기 위해서, 실험의 세부 목표를 다음과



그림 3. 토폴로지
Fig. 3. topology

같이 설정하였다. 1. 패킷 손실을 증가 또는 홉 수 증가에 따른 각 LDA 기법의 혼잡손실 탐지정확도와 무선손실 탐지정확도를 비교하고 평가한다. 2. 각 LDA 기법을 TCP Reno에 적용했을 때 어느 정도의 성능향상을 가져오는지 비교하고 평가한다.

이를 위해서 25개의 서로 다른 시나리오를 실험 의도에 따라 세 개의 소그룹(W, C, M 그룹)으로 재구성하였다. W 그룹은 패킷 손실률, 또는 홉 수 증가에 따른 각 LDA 기법의 무선손실 탐지정확도를 비교하기 위한 시나리오들로 구성된다. 따라서, W 그룹에 속한 모든 시나리오에는 무선오류로 인해 발생한 패킷손실만 존재하며, 패킷 손실률은 최저 1%에서 최고 6%정도 이다. W 그룹의 시나리오 실험을 통해서 각 LDA 기법이 얼마나 많은 무선손실을 정확히 탐지하는지를 살펴본다 (TCP Reno의 무선손실 탐지정확도는 0%).

C 그룹은 패킷 손실률, 또는 홉 수 증가에 따른 각 LDA 기법의 혼잡손실 탐지정확도를 비교하기 위한 시나리오들로 구성된다. 따라서, C 그룹에 속한 모든 시나리오는 혼잡으로 인해 발생한 패킷손실만 존재하며, 패킷의 손실률은 최저 1%에서 최고 15% 정도이다 (서로 다른 수준의 혼잡을 발생하기 위해서, 네트워크상에 흐르는 TCP 흐름의 수를 1개부터 25개로 점진적으로 증가시켰다). C 그룹의 시나리오 실험을 통하여 각 LDA 기법들이 얼마나 많이 혼잡손실 탐지정확도를 희생시키는지를 살펴본다 (TCP Reno의 혼잡손실 탐지정확도는 100%).

M 그룹은 현실성을 반영한 네트워크 환경에서 각 LDA 기법의 탐지 정확도를 비교 평가하기 위한 시나리오들로 구성된다. 따라서 M 그룹의 각 시나리오에는 혼잡손실과 무선손실이 모두 공존하며, 패킷 손실률은 최저 4%에서 최고 8% 사이이다. 그리고 혼잡손실과 무선손실간의 비율은 대략 5:5, 8:2, 2:8 이다.

2. 평가 기준

LDA 기법의 가장 중요한 평가 기준은 얼마나 정확하게 혼잡손실과 무선손실을 탐지하는가에 있다. 왜냐하면 LDA가 혼잡손실과 무선손실을 정확하게 구별하면, 패킷손실의 원인에 따라 적절한 혼잡 제어가 가능하기 때문이다. 특히, 무선손실인 경우에 TCP가 불필요하게 전송률을 절반으로 감소하는 것을 방지할 수가 있다. 본 실험에서는 각 LDA 기법에 대해서 다

표 1. 네트워크 파라미터
Table 1. Network Parameters

Bandwidth	2Mbps
Application	FTP/Generic
Transport Protocol	TCP Reno
Queuing Policy	DropTail
Link Protocol	IEEE 802.11b
Maximum segment size	1024Bytes
Packet size	1024Bytes

음의 세 가지 탐지정확도(A_w, A_c, A_t)를 측정하였으며, 수식은 다음과 같다.

$$A_w = \frac{D_w}{N_w} \times 100 \dots \dots \dots (7)$$

식 (7)에서 A_w는 무선손실에 대한 탐지정확도이며, D_w는 무선 오류로 인하여 손실된 패킷들 중에서 LDA 기법이 정확하게 탐지한 무선손실의 개수를 나타내고, N_w는 무선 오류로 인해서 발생한 무선손실의 개수를 나타낸다. 따라서 A_w는 0%에서 100%사이의 값을 가진다.

$$A_c = \frac{D_c}{N_c} \times 100 \dots \dots \dots (8)$$

식 (8)에서 A_c는 혼잡손실에 대한 탐지정확도이며, D_c는 혼잡으로 손실된 패킷들 중에서 LDA 기법이 정확하게 탐지한 혼잡손실의 개수를 나타내고, N_c는 혼잡으로 인해서 발생한 혼잡손실의 개수를 나타낸다.

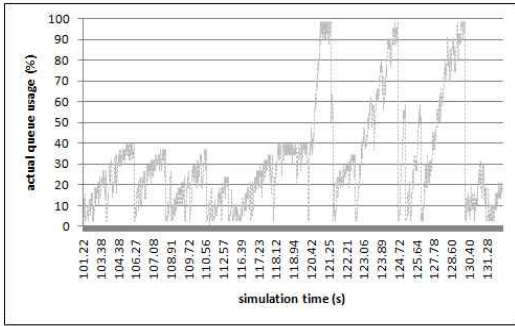
$$A_t = \frac{A_w + A_c}{2} \dots \dots \dots (9)$$

식 (9)에서 A_t는 A_w와 A_c에 대한 평균값으로 각 LDA를 대표하는 정확도이다. 예를 들어, 네트워크상의 모든 패킷손실을 혼잡손실로 간주하는 TCP인 경우에 A_w는 0%, A_c는 100%, 그리고 A_t는 50%이다.

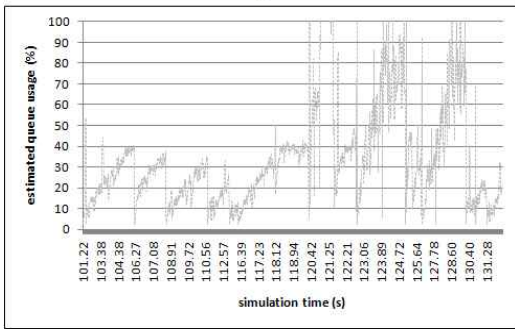
3. 실험 결과 및 분석

3.1 탐지정확도 평가

탐지정확도를 평가하기 이전에, 먼저 본 기법이 추정한 큐 사용률이 실제 큐 사용률에 얼마나 근접하는가를 간단히 살펴 보았다. 그림 4의 (a)와 (b)는 어느 한 시나리오의 특정 시간대에 실제 큐 사용률과 본 기법이 추정한 큐 사용률(estimated_queue)을 나타낸다. 그림에서 보듯이, 본 기법이 추정한 큐 사용률은 실제 큐 사용률에 매우 근접함으로 알 수 있다. 따라서 본 기법이 추정한 큐 사용률(estimated_queue)은 혼잡손실과 무선손실을 식별하는데 매우 적합함을 알 수 있다.

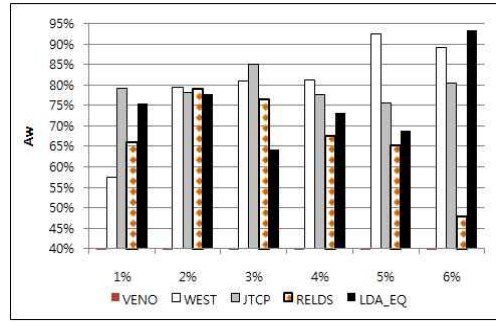


(a)

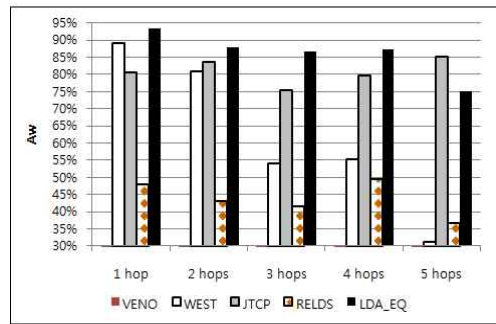


(b)

그림 4. 실제 큐와 추정된 큐의 관계
Fig. 4. the actual queue usage and the estimated queue usage



(a)



(b)

그림 5. W 그룹에서의 Aw 비교
Fig. 5. Aw in W group

그림 5는 W 그룹에서의 실험 결과를 보이며, 그림 (a)는 W 그룹에서 패킷 손실률 증가에 따른 각 LDA 기법들의 무선 손실 탐지정확도(Aw)를 보인다. 기존의 LDA 기법들 중에서 Veno는 40%이하의 매우 낮은 정확도를 가지는 반면, 본 기법인 LDA_EQ를 비롯하여, West, JTCP는 매우 높은 탐지정확도를 가진다. 또한 LDA_EQ와 West의 정확도는 패킷 손실률이 증가할수록 탐지정확도가 증가하는 경향이 있다.

그림 5의 (b)는 홉 수 증가에 따른 각 LDA 기법의 Aw를 보인다. 이 그래프는 홉 수가 증가할 때, 모든 LDA 기법들의 Aw가 감소함을 보인다. 특히 West와 RELDS의 정확도가 홉 수 증가에 따라 다소 크게 감소하는 경향이 있다. 본 기법도 홉 수에 다소 영향을 받지만, 여전히 가장 높은 탐지정확도를 가지며, 다른 기법들에 비해서 다소 덜 영향을 받음을 알 수 있다.

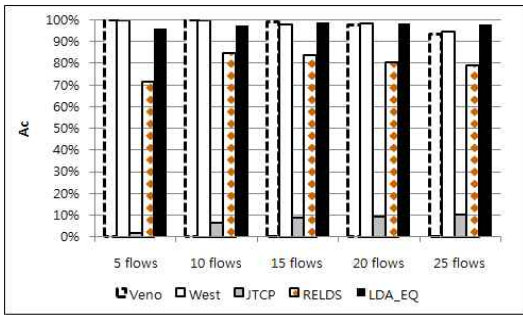
그림 6은 C 그룹에서 패킷 손실률, 그리고 홉 수에 따른 각 LDA 기법의 혼잡손실 탐지정확도(Ac)를 측정하여 비교한 결과를 보인다. 그림 6의 (a)는 혼잡으로 인한 패킷손실이 증가할 때 Ac의 변화를 나타낸다. 그림에서와 같이, Veno와 본 기법인 LDA_EQ가 가장 높은 정확도를 가지고, JTCP가 가장 낮은 정확도를 가진다. 다시 말해서, 기존의 기법들 중에서

JTCP가 가장 크게 Ac를 희생시키는 반면에, 본 기법은 Ac를 가장 적게 희생시키는 것을 알 수 있다.

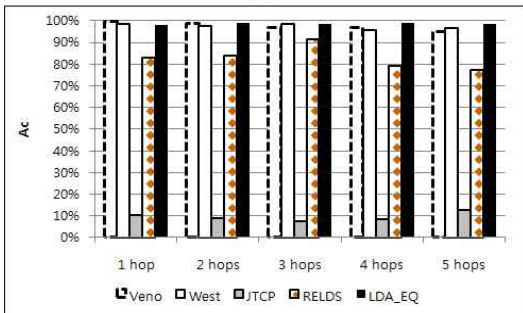
그림 6의 (b)는 홉 수 증가에 따른 Ac의 변화를 보인다. 이 그래프에서 RELDS의 정확도는 홉 수의 증가에 따라 감소하는 경향을 보이고, 특히, JTCP의 Ac가 크게 감소함을 알 수 있다 (TCP Reno의 Ac는 100%). 반면에 본 기법의 Ac는 홉 수 증가에도 여전히 높은 정확도를 가짐을 볼 수 있다.

그림 5와 그림 6을 통해서, JTCP와 RELDS의 Aw가 상당히 높은 반면에 Ac의 희생이 매우 큼을 알 수 있다. 또한 Veno와 West의 Ac는 상당히 높은 반면에 Aw는 상대적으로 낮다. 그러나 제시된 기법인 LDA_EQ는 매우 높은 Aw를 가질 뿐만 아니라 Ac의 희생도 매우 작음을 알 수 있다.

그림 7은 M 그룹에서 측정된 각 LDA기법의 탐지정확도를 보인다. 그림 7의 (a)는 홉 수 증가에 따른 At의 변화를 나타낸 것으로, 홉 수가 증가할 때 거의 모든 LDA기법의 정확도가 감소하는 것을 알 수 있다. 본 기법 LDA_EQ도 홉 수가 증가할수록 탐지정확도가 감소하지만, LDA 기법들 중에서 여전히 가장 높은 정확도를 가지며, 홉 수에 다소 덜 영향을 받음을 알 수 있다.



(a)

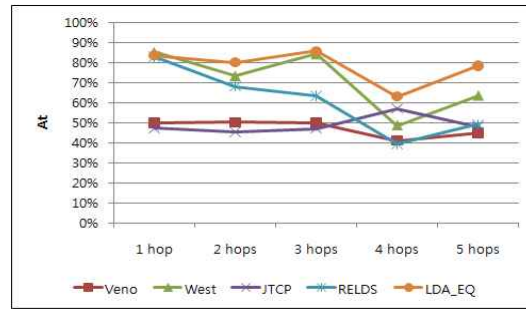


(b)

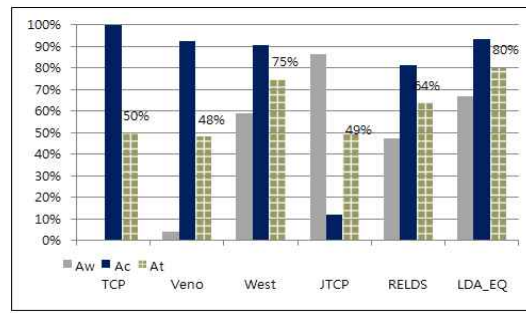
그림 6. C 그룹에서의 Ac 비교
Fig. 6. Ac in C group

그림 7의 (b)는 M 그룹의 모든 시나리오에서 관찰된 정확도들의 평균값을 나타낸 것이다. 그림에서 최 좌측은 전형적인 TCP의 탐지정확도를 나타낸 것으로, 혼잡손실에 대한 탐지정확도가 100%, 무선손실에 대한 탐지정확도는 0%, 평균 정확도(A_t)는 50%이다. 그림에서 Veneno, JTCP의 A_t는 TCP의 A_t(50%)보다 낮다. 이것은 이들 기법들이 무선손실 탐지정확도(A_w)를 조금만 개선한 반면에, 혼잡손실 탐지정확도(A_c)를 다소 크게 희생시켰기 때문이다. 즉, Veneno인 경우에, 혼잡손실에 탐지정확도(A_c)가 TCP의 100%보다 낮은 90%에 이르고, JTCP인 경우에는 A_c가 20%이다. 반면에 West, RELDS, LDA_EQ의 평균 정확도(A_t)는 TCP의 50%보다 크다. 이것은 이 기법들이 무선손실 탐지정확도를 크게 개선함과 동시에, 혼잡손실 탐지정확도를 크게 희생시키지 않았기 때문이다.

그림 7의 (b)를 살펴보면, 각 LDA기법의 혼잡손실 탐지정확도와 무선손실 탐지정확도 사이에 trade-off 관계가 존재함을 알 수 있다. 즉 무선손실 탐지정확도가 높으면, 혼잡손실 탐지정확도가 낮고, 혼잡손실 탐지정확도가 높으면 무선손실 탐지정확도가 낮다. 예를 들어, Veneno의 혼잡손실 탐지정확도는 매우 높지만, 무선손실 탐지정확도는 매우 낮고, JTCP의 무선손실 탐지정확도는 매우 높으나, 혼잡손실 탐지정확도는 가장 낮다. 반면 본 기법인 LDA_EQ은 두 개의 탐지정확도



(a)



(b)

그림 7. M 그룹에서의 A 비교
Fig. 7. A in M group

(A_w 와 A_c)가 모두 높으며, 결과적으로 평균 정확도(A_t)가 80%로서 기존 기법들 중에서 가장 높다.

3.2 성능평가

본 실험에서는 각각의 LDA 기법을 TCP Reno에 적용했을 때 보이는 goodput을 측정하여 비교/평가하였다. 이를 위해서 TCP Reno의 fast recovery 알고리즘을 간단히 수정하였다. 즉 패킷손실을 알리는 세 개의 연속적인 중복 ACK를 받았을 때, 혼잡손실로 판단된 경우에는 기존의 fast recovery를 그대로 적용하고, 무선손실로 판단된 경우에는 손실된 패킷만 재전송하고 전송률을 절반으로 줄이지 않도록 하였다.

그림 8의 (a)는 W 그룹에서 무선오류로 인한 패킷손실이 6%일 때 측정된 goodput을 보이고 있다. TCP Reno를 기준으로 Veneno와 JTCP는 비슷한 성능을 보이고, RELDS와 West는 다소 성능이 향상됨을 알 수 있다. 또한 기존의 기법들과 비교해서 본 논문에서 제안된 기법(LDA_EQ)이 TCP Reno의 성능을 가장 많이 향상시켰음을 알 수 있다.

그림 8의 (b)는 혼잡손실과 무선손실이 공존하는 M 그룹에서 측정된 각 LDA기법의 goodput을 보인다. 해당 시나리오에서 발생한 패킷손실은 8%정도이며, 혼잡손실과 무선손실의 비율은 대략 5:5이다. 그림에서와 같이, M 그룹에서도 본 기법

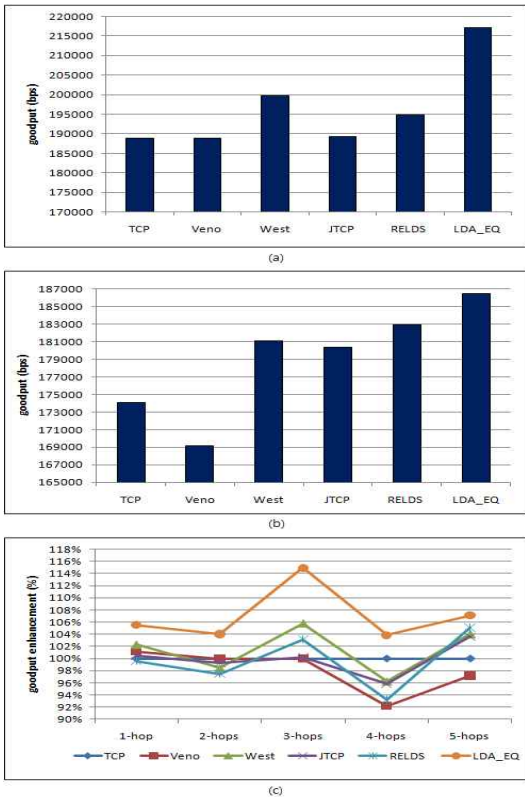


그림 8. 성능향상 비교
Fig. 8. Goodput Comparison

의 성능이 가장 높음을 알 수 있다.

마지막으로 그림 8의 (c)에서 홉 수 증가에 따른 각 기법의 성능향상을 비교하였다. TCP Reno의 성능을 100%로 설정했을 때, 홉 수가 증가할수록 기존 기법들의 성능이 감소하는 것을 알 수 있다. 반면에 본 기법은 홉 수에 적게 영향을 받고, 기존 기법들 보다 TCP의 성능을 높게 향상시킴을 알 수 있다.

V. 결론

본 논문에서는 네트워크 혼잡과 상호 관련성이 높은 TCP 송수신자간의 큐 사용률을 추정하고, 추정된 큐 사용률을 이용하여 혼잡손실과 무손실을 구별하는 송신자 기반의 패킷손실 구별기법을 제안하였다. QualNet을 이용한 기존 기법과의 비교실험에서, 본 기법이 가장 높은 탐지정확도를 가질 뿐만 아니라, TCP의 성능을 가장 높게 향상시킴을 보였다.

제안된 본 기법은 오직 TCP 송신자 측에서만 동작하므로, TCP의 시맨틱을 그대로 유지하면서 현재의 네트워크에 손쉽게

적용할 수 있는 이점을 가지고 있다.

참고문헌

- [1] J. Postel, Transmission control protocol, RFC 793, Sept. 1981. [TCP].
- [2] Song Cen, Pamela C. Cosman, and Geoffrey M. Voelker, "End-to-end differentiation of congestion and wireless losses," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 703-717, Oct. 2003.
- [3] Li-Ping Tung, Wei-Kuan Shih, Te-Chung Cho, Yeali S. Sun, and Meng Chang Chen, "TCP Throughput Enhancement over Wireless Mesh Networks," *IEEE Communications Magazine*, vol. 45, no. 11, pp. 64-70, Nov. 2007.
- [4] Zhenghua Fu, Petros Zerfos, Haiyun Luo, Songwu Lu, Lixia Zhang, and Mario Gerla, "The impact of multihop wireless channel on TCP throughput and loss," *Proc. of 22th Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 1744-1753, USA, Mar. 2003.
- [5] Cheng Peng Fu, and Soung C. Liew, "TCP Veno: TCP enhancement for transmission over wireless access networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 2, pp. 216-228, Feb. 2003.
- [6] Guang Yang, Ren Wang, Mario Gerla, and M.Y. Sanadidi, "TCPW bulk repeat," *Computer Communications*, vol. 28, no. 5, pp. 507-518, Mar. 2005.
- [7] Eric Hsiao-Kuang Wu, and Mei-Zhen Chen, "JTCP: jitter-based TCP for heterogeneous wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, pp. 757-766, May 2004.
- [8] C.-H. Lim, and J.-W. Jang, "Robust end-to-end loss differentiation scheme for transport control protocol over wired/wireless networks," *IET Communications*, vol. 2, no. 2, pp. 284-291, Feb. 2008.
- [9] <http://www.scalable-networks.com/index.php>
- [10] Lawrence S. Brakmo, and Larry L. Peterson, "TCP Vegas: End to end congestion avoidance on a global Internet," *IEEE J. Select. Areas. Commun.*, vol. 13, pp. 1465-1480, IEEE, Oct. 1995.
- [11] Vern Paxson, "End-to-End Internet Packet Dynamics,"

IEEE/ACM Transactions on Networking, vol. 7,
no. 3, pp. 277-292, June 1999.

저 자 소 개



박 미 영

1999 : 동서대학교 공학사
2001 : 경상대학교 공학석사
2005 : 경상대학교 공학박사
2006 : Iowa State University 포닥
2007 : 전남대학교 유비쿼터스정보가
전사업단 포닥
2008-현재 : 부산대학교 컴퓨터공학
과 연구교수
관심분야 : 네트워크 프로토콜, 고성
능 컴퓨팅

Email: openmp@korea.com



정 상 화

1985 : 서울대학교 공학사
1988 : Iowa State Univ. 공학석사
1993 : Univ. of Southern California
공학박사
1993-1994 : Univ. of Central Florida
컴퓨터공학과 조교수
1994-현재 : 부산대학교 컴퓨터공학과
교수
2002-2003 : Oregan State Univ. 컴
퓨터공학과 초빙교수
관심분야 : 클러스터 시스템, TOE,
RDMA, RFID, 무선랜
메쉬 네트워크

Email: shchung@pusan.ac.kr



이 윤 성

2006 : 부산대학교 공학사
2008 : 부산대학교 공학석사
2008-현재 : 부산대학교 컴퓨터공학과
박사과정
관심분야 : RFID 무선랜 메쉬 네트워크
네트워크 프로토콜

Email: nzellys@pusan.ac.kr