

# TCP CAE: ACK기반 역방향 네트워크의 혼잡 감지기법

## TCP CAE : Improving Wireless TCP under Reverse Background Congestion through Comparative ACK-based Estimator

김 재 현\*  
Jaehyun Kim

추 현 승\*\*  
Hyunseung Choo

### 요 약

TCP의 수신자는 송수신자간 신뢰성이 보장되는 전송을 위하여 ACK를 송신자에 전달하며, 이 ACK 데이터가 정상적으로 전달되지 못할 때 TCP의 성능이 저하된다. 본 논문은 백그라운드 트래픽으로 인해 역방향 네트워크가 혼잡한 경우 TCP 송신자에서 ACK의 수신이 원활하지 못하기 때문에 전송 성능이 급격히 떨어지는 문제를 해결하는 기법을 제안한다. 본 기법은 라우터나 게이트웨이 등 별도의 하드웨어나 수신자의 도움 없이 송신자에서 역방향 혼잡을 감지하는 기법으로, 송신자는 수신자로부터 전달받는 ACK들의 수신시간 차이와 그 ACK에 실려있는 타임스탬프(timestamp)값들의 시간 차이를 각각 비교한다. 그 차이 값을 성능저하의 판단 근거로 하여 역방향 혼잡을 판단할 뿐만 아니라 역방향 혼잡 상황이 발생했을 때의 대응 방안을 제안한다. NS-2 기반 시뮬레이션을 수행한 결과, 역방향 네트워크가 혼잡하고 무선 링크의 에러율이 1%인 유무선 혼합 환경에서는 제안 기법이 Reno에 비해 20%, New Reno에 비해 150%, Westwood에 비해 450%의 성능 향상을 보인다.

### Abstract

TCP receivers deliver ACK packets to senders for reliable end-to-end transfer. When ACK packets are not transferred properly because of network congestion, the performance of TCP degrades. This paper proposes a reverse congestion warning mechanism and a congestion handling mechanism in heterogeneous networks with heavy background traffic in the backward direction. Help from TCP receivers or hardware such as routers and gateways other than the ACK packets themselves is not necessary. TCP senders compare the arrival intervals of ACK data passed from receivers and the difference in timestamp values echoed by receivers. According to the simulation results using the NS-2 network simulator, the proposed scheme shows a performance elevation of 20% greater than Reno, 150% greater than New Reno, and 450% greater than Westwood, respectively, under heterogeneous networks and that the error rate of the radio link is 1% when the backward network is congested.

☞ keywords: TCP, ACK기반, 혼잡제어

## 1. 서 론

현재는 유선 네트워크와 무선 네트워크가 함께 사용되는 혼합형 네트워크(heterogeneous network)가 널리 사용된다. TCP(Transmission Control Protocol) [1]는 성능향상 측면에서 오랜 기간 연구되어 왔음에도 불구하고, 유선 네트워크 인프라

의 지속적인 고속화 기술과 다양한 무선 네트워크 표준 및 이동단말이 등장함에 따라 TCP의 안정성과 성능 개선에 대한 연구 역시 계속되고 있다 [2]. 유무선 혼합 네트워크에서 TCP 전송의 효율성을 높이고자 TCP Westwood [3], Vegas [4], Jersey, New Jersey [5], DCR [6] 등 다양한 TCP 기법들이 제시되었다. 또한, 네트워크 인프라 환경이 달라짐에 따라 전송 지연이 긴 위성 통신 환경 및 핸드오프가 발생하는 이동통신망 상에서의 TCP 성능 향상 및 안정화에 대한 연구가 TCP Peach, Freeze TCP, DEMO Vegas [7] 등을 통해

\* 준 회 원 : 성균관대학교 일반대학원 이동통신공학과 재학 (석사) luran@skku.edu

\*\* 종신회원 : 성균관대학교 정보통신공학부 부교수 choo@ece.skku.ac.kr

[2008/02/18 투고 - 2008/02/20 심사 - 2008/03/17 심사완료]

지속적으로 수행되었다.

TCP의 헤더는 네트워크의 상태에 대한 정보를 별도로 포함하지 않기 때문에 TCP 송신자는 현재 네트워크의 상태에 대한 정보가 많이 부족하다. 따라서, 에러가 발생하지 않을 때까지 전송량을 산술적으로 늘리고, 문제가 발생하면 전송량을 반으로 줄이는 방식(Additive Increase Multiplicative Decrease/ AIMD)을 사용한다 [8], [9]. 그러나, 이와 같은 기존의 대응 방식은 다양한 전송매체, 전송속도, 에러율, 라우팅 경로 등을 갖는 복잡한 환경에서는 적절히 동작하지 않는다. 결과적으로 송신자와 수신자간 전송속도 및 처리능력의 불균형 요소는 다양하다. 무선 네트워크가 표준에 따라 전송속도, 전파 전송범위 등이 다르기 때문에 다양한 무선전송 에러의 잠재적 요소를 지니고 있다. 더불어, 유선 네트워크도 지역마다 다른 네트워크 환경 및 대역폭을 사용한다. 뿐만 아니라, 중간 라우터들의 정책에 따라 경로가 비대칭적(asymmetric)으로 설정될 수도 있기 때문에 송수신 경로 및 속도 차이가 발생할 수 있다 [10], [11].

네트워크가 고속화 될수록 고속 인프라를 활용하는 대용량의 데이터가 전송되며, 이에 따라 앞서 언급한 네트워크 환경과 송수신자간의 처리 성능의 불균형은 결국 데이터 송수신의 지연 및 TCP 전송의 성능저하를 야기한다. TCP의 송신자는 수신자에 무조건 데이터를 보내지 않고 수신자에서 ACK(Acknowledgement)를 받으면서 보내게 되므로, 특히 ACK 수신을 방해하는 역방향 네트워크의 혼잡은 성능저하의 큰 원인이다. 본 논문에서는 역방향 네트워크가 극도로 혼잡하여 ACK의 수신이 원활하지 않고 정방향 네트워크에서는 원활하게 데이터가 전송될 때, 라우터 등 하드웨어의 도움 없이 송신자 기반으로 역방향 혼잡을 판단하고 그 상황에 대처하는 기법을 제안한다. 본 기법은 NS-2 네트워크 시뮬레이터를 사용하여 성능평가를 수행한 결과, 역방향 네트워크가 심각하게 혼잡하고 무선 링크의 에러율이 1%

인 환경에서 TCP Reno에 비해 20%, New Reno에 비해 150%, Westwood에 비해 450%의 성능 향상을 보인다.

본 논문은 다음과 같이 구성된다. 2장에서 관련연구를 소개하고, 3장에서 TCP CAE 기법을 설명한다. 4장에서는 네트워크 환경 및 실험 파라미터를 소개하고 실험 결과를 분석하며, 5장은 결론으로 끝맺음 한다.

## 2. 관련 연구

유선 네트워크에서 동작하는 대표적인 TCP로 TCP Reno가 있으며, 유무선 혼합 네트워크에서 종단간 무선 전송 기법으로는 TCP Westwood, TCP Jersey, TCP New Jersey가 있다.

### 2.1 TCP Reno와 New Reno

TCP Reno는 기본적으로 Fast Retransmit과 Fast Recovery 기법을 사용한다. 원하는 데이터가 수신되지 않았을 경우, 송신자는 Duplicate ACK를 계속 보내게 되고, 이를 감지한 송신자는 RTO를 기다리지 않고, 즉시 Fast Retransmit을 수행한다. Fast Retransmit을 마친 후에는 Slow Start 단계를 거치지 않고, 곧바로 Congestion Avoidance 단계로 진입하는 Fast Recovery를 수행한다. 그러나, 이와 같은 기법을 갖고 있는 Reno는 특히 무선 네트워크 환경에서 많이 발생할 수 있는 다중 패킷 손실 인식에 대해서는 취약하게 대응한다는 단점이 있다. 이를 보완하기 위해, New Reno에서는 하나의 윈도우 데이터를 모두 처리할 때까지 Fast Retransmit 단계를 빠져나가지 않는다. 결국, 하나의 윈도우내 손실된 모든 패킷을 처리할 때까지 현재의 전송률을 유지시키기 때문에, 매 손실 건에 대해 Fast Retransmit/ Fast Recovery를 반복적으로 수행하는 Reno에 비해 성능이 좋다. 그러나, New Reno도 여러 개의 패킷 손실에 대응하는 방법은 가지고 있지만, 무선 네트워크 상에서 흔히

발생할 수 있는 패킷 손실과 지연에 대한 구분이 어렵기 때문에 낮은 성능을 보인다.

## 2.2 TCP Westwood

Westwood는 전송 에러가 많은 환경에서 고전 TCP가 사용하는 AIMD 기법을 그대로 사용할 경우, 심각한 성능저하가 발생한다는 점을 지적하였다. TCP Westwood는 자신이 전송한 데이터에 대한 ACK를 샘플링하여, 수신하는 ACK의 패턴으로부터 가용한 네트워크 대역폭을 송신자에서 하드웨어 도움없이 계산해낸다. 이 정보를 활용하여, 전송량을 무조건 반으로 줄이는 대신 실제 처리량에 비례하여 전송량을 줄인다. 이를 Additive Increase Adaptive Decrease(AIAD) 기법이라고 한다. 그러나, 종단간 처리 용량에 근거하여 전송량을 줄이는 기법이라 하더라도, ACK가 지연되거나 압축되어 전송되는 경우 가용 대역폭의 판단에 오류가 생긴다는 단점이 있다. Westwood+의 연구를 통해 가용 대역폭의 계산 방법이 개선되기도 하였으나, 패킷이 손실된 직후 낮은 응답률로 인한 큰 Bandwidth Delay Product(BDP)의 문제점을 여전히 갖고 있다 [12].

## 2.3 TCP Jersey와 New Jersey

TCP Jersey는 CW와 ABE라는 핵심요소로 구성된다. Congestion Warning(CW)은 중간 라우터의 도움을 받아, 큐의 사용량이 75% 이상이 되면 혼잡상황으로 패킷에 마크를 하여 혼잡 상황에 대한 대처를 할 수 있도록 한 기법이다. Available Bandwidth Estimation(ABE)은 전송한 데이터의 처리시간을 기준으로 현재 네트워크가 처리할 수 있는 가용능력을 산출한다. 또한, New Jersey는 Jersey에서 제안하였던 ABE의 정확함을 높인 기법이다. Jersey는 ACK를 수신할 때마다 ABE계산을 하는 반면, New Jersey는 타임스탬프값을 활용하여 ABE를 산출함으로써 ACK가 일시적으로 물

리는 상황(burstiness)에도 효과적으로 대응할 수 있다. 그러나, 직관적인 ABE 산출 기법을 통해 높은 성능을 제시하였음에도 불구하고, 중간 라우터 등 하드웨어에 의존적이라는 점 때문에 구현 시 높은 비용이 문제가 될 수 있다.

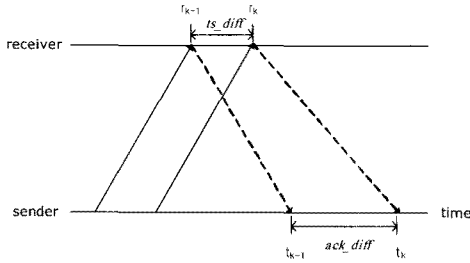
## 3. TCP Comparative ACK-based Estimator (CAE)

역방향 네트워크가 혼잡할 경우 TCP는 ACK를 원활히 수신하지 못하기 때문에 적절한 성능을 내기 어렵다. TCP CAE는 역방향 혼잡이 발생하기 전에 이를 미리 감지하고, 대응하는 기법이다.

### 3.1 역방향 네트워크 혼잡 감지 기법

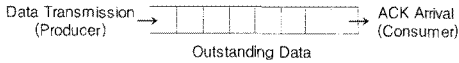
일반적으로 네트워크상에서의 지연은 전송지연, 큐잉지연, 각 노드의 처리지연으로 구성된다. 역방향 네트워크가 혼잡하면, ACK의 수신이 원활하지 못하고, 결국 TCP 전송 성능의 저하를 초래한다. 따라서, 본 논문에서는 중간 네트워크에서의 큐잉지연 및 각 노드의 처리지연은 동일하다고 가정하고 네트워크의 전송지연에만 초점을 맞추며, 특히 역방향 네트워크의 전송지연의 감지 및 대응기법에 중점을 둔다.

그림 1에서 ACK는 파선으로 표시하였으며, 네트워크가 역방향 트래픽으로 인해 혼잡해지는 상황을 나타낸다. 첫 번째 ACK 파선에 비해 두 번째 ACK 파선은 선의 기울기가 상대적으로 완만하고, 이는 역방향 네트워크가 더 혼잡해졌음을 의미한다. 송신자가 전송한 데이터가 실제적으로 수신자에 도착한 시각  $t_{k-1} \sim t_k$ 의 시간차를 기대값 변수  $ts\_diff$ 로, 수신자가 응답한 ACK를 송신자가 수신한 시각  $t_{k-1} \sim t_k$ 의 시간차를 측정값 변수  $ack\_diff$ 로 정의한다. 본 기법에서는  $ack\_diff$ 가  $ts\_diff$ 와 같거나 작을 때 네트워크 상태가 안정적이라고 판단한다.



(그림 1) 전송 패턴과 전송 지연

그림 2에서 보는 바와 같이 송신자 입장에서 수신자에게 데이터를 전송한 후 ACK를 받을 때까지 대기하는 데이터를 대기 데이터(outstanding data)라고 볼 수 있다. 이러한 대기 데이터가 존재하는 큐의 관점에서 보면, 송신자가 데이터를 전송하면 대기 데이터가 생성(enqueue)되고 수신자로부터 ACK를 받으면 대기 데이터가 소멸(dequeue)된다. 대기 데이터를 생성하는 연산을 Producer라고 정의하고, 소모하는 것을 Consumer라고 정의한다.



(그림 2) 향상된 가용대역폭 예측 알고리즘

대기 데이터가 생성되는 속도에 비해 소모되는 속도가 일정 수준 이하로 저조할 경우 역방향 네트워크의 흐름에 문제가 발생했다고 판단할 수 있다. 이는 Producer와 Consumer의 처리성능을 비교하여 확인이 가능하며, 이는 곧 네트워크 대역폭의 정의에 따라 계산할 수 있다.

$$data(bytes) = bandwidth(bytes/sec) \times time(sec) \quad (1)$$

k-1번째 ~ k번째 대기데이터를 생성할 때의 Producer의 성능  $P_p$ 와 소모할 때의 Consumer의 성능  $P_c$ 는 식 2, 3과 같다.

$$P_p(producer\ performance) = \frac{m+n}{r_k - r_{k-1}} \quad (2)$$

$$P_c(consumer\ performance) = \frac{m+n}{t_k - t_{k-1}} \quad (3)$$

(단,  $m = r_{k-1}$ 에 생성(소모)되는 데이터의 크기,  $n = r_k$ 에 생성(소모)되는 데이터의 크기)

현재 네트워크의 처리성능( $P_c$ )은 기대값( $P_p$ ) 대비 측정값( $P_c$ )으로 나타낼 수 있으며, 식 4에 따라  $ts\_diff$ 와  $ack\_diff$ 의 관계로 요약하여 표현이 가능하다.

$$P_c(estimated\ performance) = \frac{P_c}{P_p} = \frac{\frac{m+n}{t_k - t_{k-1}}}{\frac{m+n}{r_k - r_{k-1}}} \quad (4)$$

$$= \frac{r_k - r_{k-1}}{t_k - t_{k-1}} = \frac{ts\_diff}{ack\_diff}$$

이와 같이 산출한 `estimated_performance`에 대해 네트워크의 혼잡 여부를 판단하기 위해서는 기준값이 필요하기 때문에, 본 기법에서는 TCP Jersey/ New Jersey에서 인용된 값을 참고한다. TCP Jersey/ New Jersey에서는 ECN 기반의 Congestion Warning(CW) 기법을 적용하여 라우터 버퍼의 소비량이 75% 이상이 되면 혼잡상황이라고 판단한다. 이는 라우터의 네트워크 큐가 75% 이상 사용된다면 혼잡상황이 유발될 가능성이 높은 상황을 의미한다. 즉, 앞서 계산한 역방향의 처리성능값이 75% 미만이라면 역방향의 혼잡이 발생할 가능성이 높다고 판단하고 3.2장에 소개된 바와 같이 사전 대응을 한다.

### 3.2 ACK기반 역방향 네트워크 혼잡 대응 기법

일반적으로 TCP에서는 RTO 발생시 Kam 알고리즘에 따라 백오프(Backoff)가 수행된다. 결과적으로 RTO 이후에 전송되는 데이터에 대해서는 RTO 이전보다 큰 RTO값이 할당되고, 큰 RTO값은 네트워크 상황이 이전보다 더 혼잡함을 의미한다. 그러나, 본 기법에서는 최적 RTO를 얻기

위해 그림 3과 같이 역방향 네트워크의 성능 (estimated\_performance)을 계산한다. RTO가 발생하면 cwnd값이 1로 설정되어 실제 전송성능에 큰 손실을 유발할 수 있으므로 이 측정값을 통해 실제 RTO가 발생하기 이전에 사전 조치를 취함으로써 성능 손실을 예방한다.

```
// ACK packet arrived at the sender
ts_diff = rk - rk-1; // timestamp difference
ack_diff = tk - tk-1; // ACK arrival difference
rcw = false; // reverse congestion warning flag
estimated_performance = ts_diff/ack_diff;

// performance falls down under 75%
if (estimated_performance < 0.75) {
    if (rcw == false) {
        Proactive backoff;
        Rcw = true;
    }
}

// performance goes up over 75%
else {
    // refresh srtt only if reverse congestion warning alarmed
    if (rcw == true) {
        reset srtt;
        rcw = false;
    }
}
end if

update tk-1 and rk-1 with tk and rk, respectively;
```

(그림 3) 역방향 혼잡판단 및 성능 평가 메커니즘

성능 측정값이 75% 미만이라면 역방향 혼잡이 가중된 상황이고, 이 상황은 RTO의 발생 가능성이 높아진 상황을 의미한다. 따라서, 실제 RTO가 발생하기 이전에 미리 백오프를 취해줌으로써 RTO값을 타임아웃이 발생하지 않는 안정적인 값으로 설정한다.

## 4. 성능 평가

### 4.1 시뮬레이션 환경

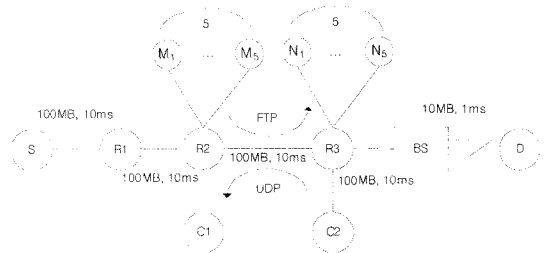
본 장에서는 NS-2 네트워크 시뮬레이터를 사용하여, TCP goodput을 측정한다. 성능 평가에 사용된 시뮬레이션 파라미터들은 표1과 같다.

(표 1) 시뮬레이션 파라미터

대역폭		패킷크기	전송지연시간		큐크기
유선	무선		유선	무선	
100MB	10MB	762byte	10ms	1ms	20 (개)

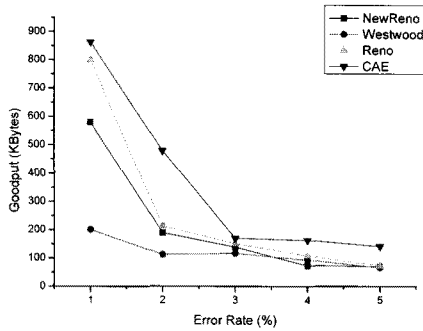
### 4.2 역방향 백그라운드 트래픽이 심한 토폴로지에서의 Goodput 측정 성능평가

본 실험에서는 TCP CAE, Reno, New Reno, Westwood의 4개 기법들이 그림 4와 같은 네트워크 토폴로지에서 동작할 때의 성능평가를 진행한다.



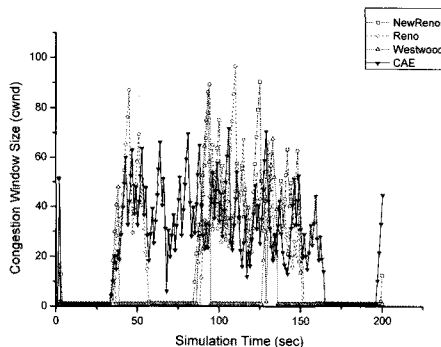
(그림 4) 백그라운드 트래픽이 존재하는 복잡한 네트워크 토폴로지

데이터 패킷은 소스 노드 S로부터 목적 노드 D로 전송되며, S로부터 BS까지는 유선으로, BS로부터 D노드는 무선으로 전송된다. 이때, M1 ~ M5노드로부터 N1 ~ N5노드로 FTP 백그라운드 트래픽이, C2 노드로부터 C1노드로 UDP CBR 백그라운드 트래픽이 존재한다. 에러율이 1%에서 5%로 변화할 때 각 기법의 goodput 변화는 그림 5와 같다. 역방향 링크가 혼잡한 상황에서는 에러율이 1%일 경우, 제안한 CAE가 New Reno보다 150%, 5%일 경우 100% 이상의 높은 성능 차이를 보인다.



(그림 5) TCP 기법간 goodput 비교

cwnd의 변화 추이는 goodput과 큰 관련이 있다. 그림 6과 같이 시뮬레이션 시간내 cwnd 크기 변화를 추적해 봄으로써 각 기법의 성능을 확인할 수 있다. Reno는 하나의 패킷 손실을 경험한 후 곧바로 Fast Retransmit 메커니즘을 수행함으로써, 현재 네트워크에서 가용한 자원을 확보하기 위한 시도를 일찍 시작한다. 그러나, New Reno나 Westwood는 이와 같은 대응 능력이 상대적으로 저조하다. 반면, CAE는 Reno만큼 높은 cwnd를 최고값으로 갖지는 못하지만, 다른 기법들이 cwnd가 1이 되는 순간에도 평균적으로 높은 cwnd 값을 유지하기 때문에 우수한 goodput을 보장해 준다.



(그림 6) TCP 기법간 cwnd 비교

## 5. 결론

본 논문에서는 역방향 네트워크가 혼잡하여 ACK의 수신에 원활하지 않을 경우의 TCP 성능이 저하되는 문제를 해결하기 위하여, 역방향 혼잡 감지 기법을 제안한다. 본 기법은 송신자가 ACK 데이터를 수신한 시간차와 수신자에서 보내는 타임스탬프값의 차를 활용하여 역방향 데이터의 처리 성능 변화를 감지하고, 성능 변화에 적절하게 대응하는 기법이다. 이를 통해, 하드웨어의 도움이 없이도 역방향의 심각한 혼잡 상황을 사전에 예측하고 대응함으로써 급격한 성능저하를 예방할 수 있다. 제안 기법은 역방향 혼잡이 극도로 심각한 상황에서 에러율 1%인 경우를 기준으로, Reno 대비 20%, New Reno 대비 150%, Westwood 대비 450%의 성능 향상을 보인다.

## Acknowledgement

본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-(C1090-0801-0046))

교신저자: 추현승.

## 참고 문헌

- [1] J. Postel, "Transmission Control Protocol," RFC 793, 1981.
- [2] Y. Tian, K. Xu, and N. Ansari, "TCP in Wireless Environments: Problems and Solutions," IEEE Radio Communications, Mar. 2005.
- [3] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links," ACM Mobicom, Jul. 2001, pp. 287-97.
- [4] L. Brakmo and L. Peterson, "TCP Vegas: End

- to End Congestion Avoidance on a Global Internet," IEEE JSAC, vol. 13, no. 8, Oct. 1995, pp. 1465-80.
- [5] K. Xu, Y. Tian, and N. Ansari, "Improving TCP performance in integrated wireless communications networks," Computer Networks, Aug. 2004.
- [6] S. Bhandarkar, A. Reddy, and N. Vaidya, "TCP-DCR : A Novel Protocol for Tolerating Wireless Channel Errors," IEEE MOBICOM, vol. 4, no. 5, Oct. 2005, pp. 517-29.
- [7] C. Y. Ho, Y. C. Chan, and Y. C. Chen, "An Efficient Mechanism of TCP-Vegas on Mobile IP Networks," Proc. IEEE INFOCOM 2006.
- [8] M. Allman, V. Paxson, and W. Stevens, "TCP Congestion Control," RFC 2581, 1999
- [9] D. Chiu, and R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks," J. Comput. Networks, vol. 17, no. 1, Jun. 1989.
- [10] H. Balakrishnan, V. Padmanabhan, and R. Kats, "The Effects of Asymmetry on TCP Performance," Proc. ACM/IEEE Mobicom, Sep. 1997, pp. 77-89.
- [11] H. Balakrishnan, V. Padmanabhan, G. Fairhurst, and M. Sooriyabandara, "The Performance Implications of Network Path Asymmetry," RFC 3449, Dec. 2002.
- [12] D. Kliazovich, F. Granelli, and D. Miorandi, "TCP Westwood+ Enhancement in High-Speed Long-Distance Networks," IEEE ICC 2006, Jun. 2006.

## ● 저자 소개 ●



### 김재현(Jaehyun Kim)

2003년 고려대학교 컴퓨터학과 졸업(학사)

2007년 ~ 현재 성균관대학교 일반대학원 이동통신공학과 재학 (석사)

관심분야 : 유/무선 TCP, 센서 네트워크

E-mail : luran@skku.edu



### 추현승(Hyunseung Choo)

1988년 성균관대학교 수학과 졸업(학사)

1990년 University of Texas 컴퓨터공학과 졸업(석사)

1996년 University of Texas 컴퓨터공학과 졸업(박사)

1997년 특허청 심사4국 컴퓨터심사담당관실(사무관)

1998년 ~ 현재 성균관대학교 정보통신공학부 부교수

2001년 ~ 현재 한국인터넷정보학회/한국시뮬레이션학회 이사

2004년 3월 ~ 2006년 8월 대통령직속 교육혁신위원회 전문위원

2004년 8월 ~ 현재 한국인터넷정보학회 논문지편집위원장

2005년 1월 ~ 현재 한국 건강보험심사평가원 전문위원

2005년 10월 ~ 현재 정보통신부ITRC 지능형HCI융합연구센터장,

정보통신공학부 컨버전스 연구소장

관심분야 : 유/무선/광 네트워크, 모바일컴퓨팅, 임베디드S/W, 그리드컴퓨팅

E-mail : choo@ece.skku.ac.kr