

---

# 무선 랜 성능 향상을 위한 링크 계층 트래픽 제어 알고리즘

최 선 응\*

Link Layer Traffic Control Algorithm to Improve the Performance of WLANs

Sun-Woong Choi\*

---

이 논문은 2007년도 국민대학교 교내연구비를 지원 받아 수행된 연구임. 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 지원 사업의 연구결과로 수행되었음 (IITA-2008-C1090-0801-0019)

---

## 요 약

오늘날 인프라 무선 랜은 많은 사용자들이 사용하는 중요한 인터넷 접속 기술이다. 무선 랜의 성능은 포화상태에서 시스템의 처리율이 저하되고, 각 노드의 처리율은 시간에 따라 심하게 변화하는 문제점이 있다. 본 논문에서는 링크 계층에서 DCF 시스템의 입력 로드를 조정하여 이러한 문제를 해결하는 트래픽 제어 방법을 제안한다. 트래픽 제어기를 사용하여 DCF로 유입되는 시스템 로드를 적절하게 조정하면 기존 MAC 프로토콜을 수정 없이도 시스템 처리율을 향상시키고 노드의 처리율 변화도 줄일 수 있는 것을 보인다. 또한, 시스템의 상황에 맞는 최적의 시스템 로드를 찾는 동적 제어 알고리즘을 제안하고, ns-2 시뮬레이션을 통하여 성능 향상을 보였다.

## ABSTRACT

Wide-spread deployment of infrastructure WLANs has made Wi-Fi an integral part of today's Internet access technology. WLANs suffer from degraded system throughput and each node's throughput fluctuates significantly in the saturation regime. In this paper, we propose a link layer traffic control mechanism which controls the offered load of DCF system. It is shown that the link layer traffic controller can improve DCF system throughput and reduce nodes' throughput fluctuation with properly controlled offered load. We propose a dynamic traffic control algorithm which can find an optimal offered load and show its performance improvement with ns-2 simulation.

## 키워드

무선 랜, DCF 처리율, 트래픽 제어

## I. 서 론

캠퍼스 망, 엔터프라이즈 망, 상용 핫스팟(hotspot) 등에서 널리 사용되는 IEEE 802.11 무선 랜 [1]은 유무선 망이 통합되어 가고 있는 오늘날, 인터넷의 중요한 구성요소가 되었다. 1970년대의 ALOHA [2]와 이더넷 [3]의 경

우와 마찬가지로, 1990년대 후반 등장한 무선 랜의 특성을 이해하기 위한 많은 연구들이 진행되었다. 무선 랜에 관한 대표적인 연구들로는 IEEE 802.11 CSMA/CA의 성능 분석 [4], [5], 공평성을 제공하는 스케줄링 방법 [6], [7], 실제 동작 중인 무선 랜의 트래픽 측정 및 분석 [8], [9], [10], AP 채널 할당 문제 [11], [12], [13], [14] 등이 있

다. 1980년대 CSMA/CD를 사용하는 이더넷의 성능에 관한 많은 논쟁이 있었지만 [15], 그러한 논쟁은 스위칭 기술의 개발과 더불어 자취를 감추었다. 그에 비해, 무선 랜에서는 스위치를 통하여 간섭(interference)이나 충돌(contention)을 지역화하는 것이 불가능하다. 이와 같이 유선망과는 다른 무선 랜의 특성으로 인하여, 무선 랜의 성능은 채널 접근 시에 발생할 수 있는 프레임 간의 충돌에 크게 영향을 받게 된다.

본 논문에서는 오늘날 대부분의 무선 랜이 동작하는 핫스팟 환경에서의 성능에 초점을 맞춘다. 이러한 환경에서는 충돌(contention)에 기반을 둔 다중 접속 방법의 성능이 주요한 관심사가 된다. [4][5]에서는 IEEE 802.11 DCF의 성능을 모델링하였다. 시스템 입력 로드(offered load)가 증가하여 포화 상태에 진입하면, 시스템의 MAC 처리율이 크게 저하되는 것을 보이고 있다. 또한, 포화 상태에서는 각 노드가 자신의 데이터를 전송하기 위하여 이웃 노드와 치열하게 경쟁을 하여야 하고, 경쟁이 치열한 만큼 여러 노드가 동시에 데이터를 전송하여 충돌이 일어날 확률이 높다. 따라서 포화 상태에서는 각 노드의 처리율이 시간에 따라 크게 변하게 되고, 이는 안정적인 QoS를 제공하는 것을 어렵게 한다.

본 논문에서는 포화 상태에서 MAC의 성능이 저하되는 것을 막기 위하여 링크 계층에서 트래픽 제어를 수행하는 방법을 제안한다. DCF 시스템의 입력 로드를 제한하여 DCF 시스템이 포화 상태에서 동작하는 것을 막는다. 이를 통하여 시스템 처리율이 향상되고, 그와 더불어 노드의 처리율 변화가 줄어드는 효과가 있다. [16]에서는 MAC 계층의 충돌 정보를 이용하여 MAC 계층으로 유입되는 트래픽을 제한하였으나, 본 논문에서는 MAC 계층의 도움 없이 시스템의 상황에 따라 최적의 시스템 로드를 찾아 갈 수 있는 동적 제어 알고리즘을 제안한다. MAC 계층을 수정할 필요 없기, 상위의 링크 계층에서 동작하는 알고리즘이기 때문에, 기존에 사용하는 WLAN 장치를 그대로 사용할 수 있는 장점이 있다. 제안한 알고리즘의 성능 특성을 ns-2 시뮬레이션을 통하여 확인하였다.

이후의 구성은 다음과 같다. 다음 장에서는 시뮬레이션 환경과 실험 환경에 대하여 기술한다. III 장에서는 IEEE 802.11 DCF의 처리율 특성에 대하여 논하고, IV 장에서는 공평성과 안정성 측면에서 DCF의 성능에 대하여 논한다. 마지막으로, 결과에 대한 토의로 결론을 맺는다.

## II. 시스템 모델

### 2.1. 인프라 무선 랜 환경

본 논문에서는 무선 호스트를 서비스하는 IEEE 802.11 인프라 무선 랜 접속 망(access network)으로 구성된 유무선 통합 IP 네트워크를 고려한다. 그림 1은 그러한 네트워크 시스템의 구성 요소를 보여 준다. 특히, 본 논문에서는 무선 접속 망의 특성에 초점을 맞춘다. 네트워크 시스템은 아령 모양의 네트워크로 모델링하였다. 무선 호스트들이 n 개가 있고, 각 무선 호스트는 AP를 통하여 인터넷에 접근할 수 있다. AP는 라우터와 연결되어 있고, 다시 라우터는 n 개의 유선 호스트와 연결되어 있다.

### 2.2. 시뮬레이션 환경

그림 1과 같은 아령 형태의 유무선 무선 랜 토폴로지를 실험하기 위해서 ns-2.27 [17]을 사용한다. 구름 형태의 IP 네트워크는 하나의 라우터로 대체되었다. 유선 링크의 대역폭은 100 Mb/s로, 라우터와 AP 사이 링크의 지연은 1 msec, 유선 호스트와 라우터 사이 링크의 지연은 10 msec로 지정하였다. IEEE 802.11 데이터 프레임의 전송 속도는 11 Mb/s로 지정하였고, ACK 프레임의 전송 속도는 2 Mb/s로 지정하였다. 기본적으로  $10^{-6}$ 의 BER (Bit Error Rate)을 가정하였다.

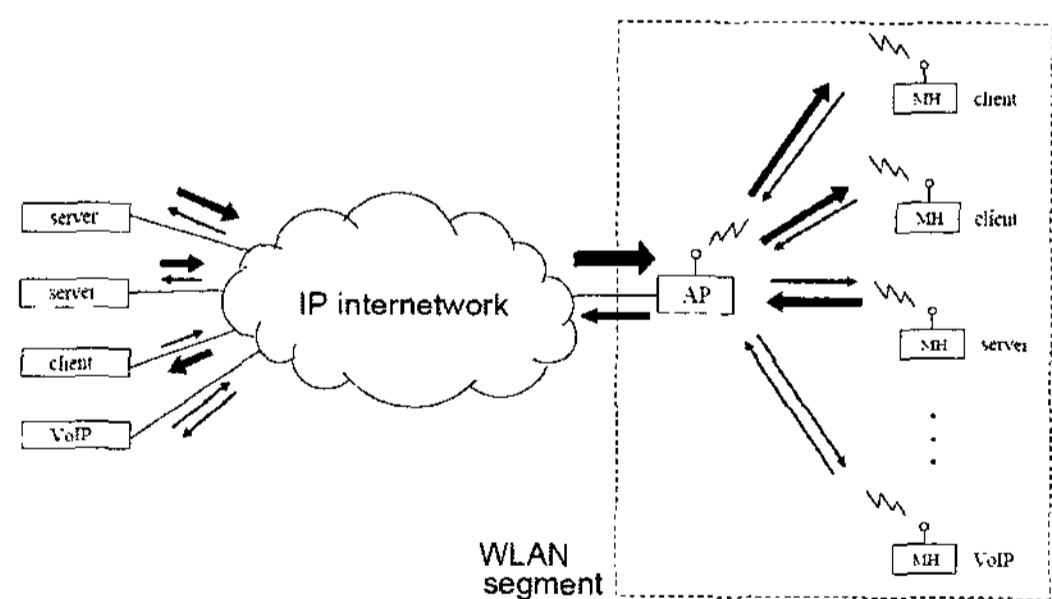


그림 1. IEEE 802.11 인프라 무선 랜을 포함하는 유무선 통합 IP 네트워크  
Fig. 1 Wireless/wireline IP Internetnetwork with IEEE 802.11 infrastructure WLAN segment

그 밖의 IEEE 802.11b 관련 파라미터 값은 그림 2(a)에 요약되어 있다. 이 값들에서 하나의 무선 노드가 얻을 수 있는 최대의 성능을 계산할 수 있다. 프레임의 데이터 크기를 L이라고 하고, 백오프(backoff) 구간의 길이를  $T_{BO}$

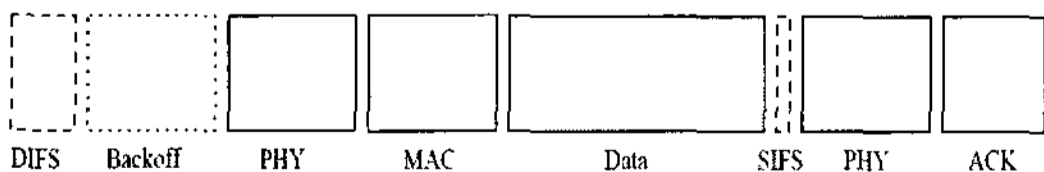
라고 하면, 하나의 프레임 전송을 마치기 위하여 필요한 시간은  $50 + T_{BO} + 192 + (224+8L)/11 + 10 + 192 + 112/2$  ( $\mu$ s)이 된다. 그림 2(b)는 DCF 프레임과 ACK의 전송 과정을 보여 준다. 만일 백오프 구간을 무시한다면,  $L = 1000$  bytes의 프레임을 전송하는데 약 1.25 msec가 걸리는 것을 알 수 있다. 따라서 최대 성능을 6.4 Mb/s임을 알 수 있고, 이러한 이론적 최대값은 프레임의 크기에 따라 변하게 된다.

무선 노드는 전송할 데이터가 발생하면, 임의의 시간 동안 백오프를 수행하여 이웃 노드와 동시에 전송하는 확률을 줄인다. 충돌이 없다면, 0~31 슬롯에서 임의의 값을 선택하기 때문에 평균적으로 310 $\mu$ s를 기다리게 된다. 이와 같이 백오프를 통하여 낭비되는 시간을 고려한다면 이론적으로 하나의 노드가 얻게 되는 성능은 5.136 Mb/s이다.

2~4개의 무선 노드가 있는 경우에는, 둘 이상 노드가 동시에 프레임 전송을 시도하여 충돌로 인한 무선 채널 자원 낭비가 있을 수 있음에도 불구하고, 백오프 카운터 값을 각 노드가 동시에 줄여 나갈 수 있는 이득이 더 커서 전체 시스템 성능이 더 높아진다.

slot time	20 $\mu$ s
SIFS	10 $\mu$ s
DIFS	50 $\mu$ s
CWmin	31
CWmax	1023
physical preamble	144 bits
physical header	48 bits
ACK frame	112 bits
MAC header + CRC	224 bits

(a)



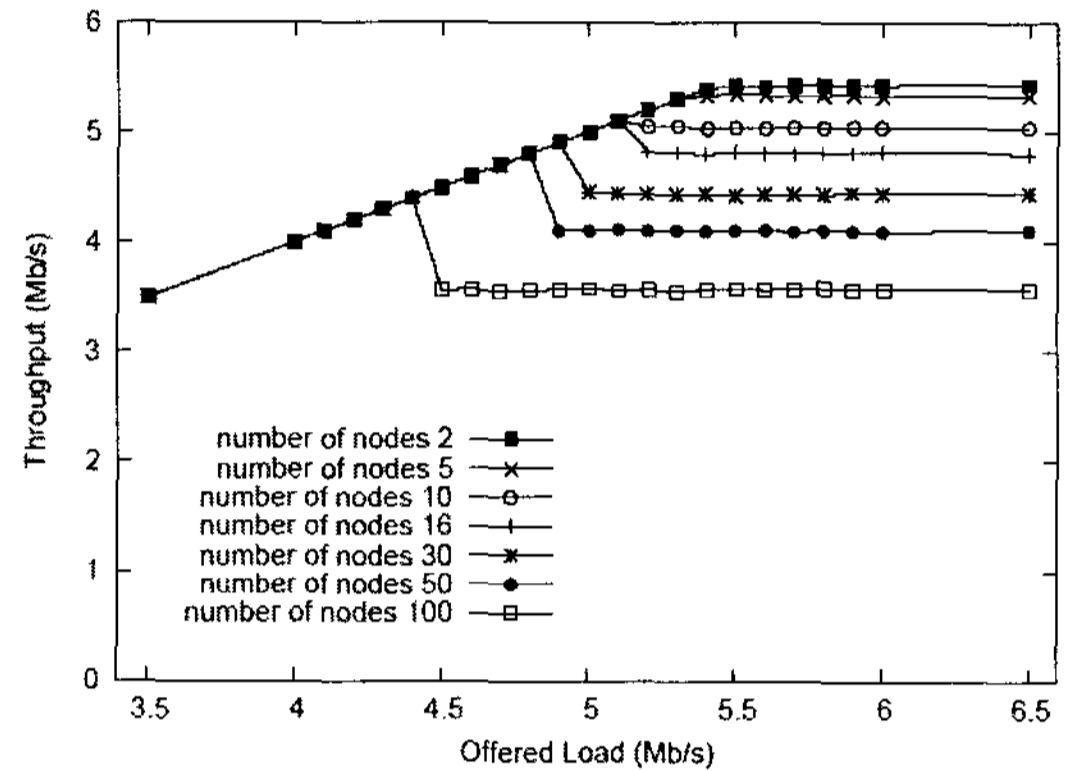
(b)

그림 2. (a) IEEE 802.11b 파라미터 값. (b) DCF 프레임과 ACK 전송 스케줄

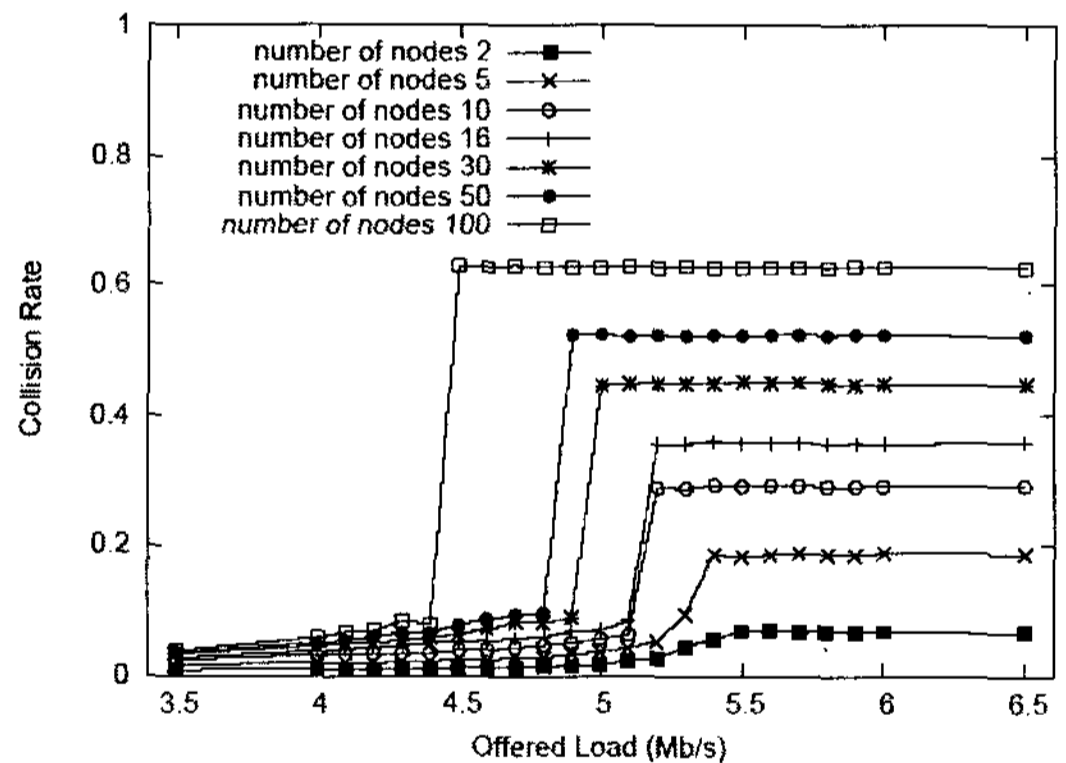
Fig. 2 (a) IEEE 802.11b parameter specification (b) DCF frame transmission and acknowledgment schedule.

### III. DCF의 성능

먼저, DCF의 처리율에 관하여 논의를 시작한다. AP를 중심으로 무선 노드들이 반지름 10m의 원주 위에 일정한 간격으로 대칭적으로 위치하는 경우를 가정한다.



(a)



(b)

그림 3. (a) 2~100개의 무선 노드에 대한 시스템 입력 로드에서 IEEE 802.11 DCF의 시스템 처리율. (b) DCF의 충돌 확률

Fig. 3 (a) Simulated IEEE 802.11 DCF throughput as a function of offered load for two to 100 wireless stations (b) Corresponding DCF collision rate.

그림 3(a)는 2~100 개의 무선 노드가 존재할 때, 시스템 입력 로드에서 DCF의 성능이 어떻게 변하는지를 보여 준다. 시스템 입력 로드는 일정한 속도로 생성되는 CBR (constant bit rate) 트래픽으로 구성되었다. 단, 특정 동기화 현상을 방지하기 위하여 패킷을 생성하는 시간

간격에 10%의 범위에서 확률적인 변화를 주었다. 하나의 시뮬레이션은 200초 동안 지속되고, 그 시간은 10개의 무선 노드가 포화 상태에서 약 18만 개의 프레임 전송할 수 있는 정도의 길이이다. 앞으로 제시될 실제 실험 결과들도 200초 동안 수행하여 얻은 값이다.

먼저, 노드가 2개나 5개 노드일 때의 시스템 처리율이 노드가 하나일 때의 처리율인 5.117 Mb/s 보다 높은 것을 볼 수 있다. 그리고 시스템 입력 로드가 증가함에 따라 시스템 처리율은 포화 상태에 도달할 때까지 선형으로 증가하고 있다. 포화 상태에 도달하는 순간 시스템의 처리율은 증가를 멈추고, 오히려 감소하는 모습을 보이고 있다. 그림 3(b)에서는 이와 같이 포화 상태에 도달하는 순간 프레임의 충돌 확률이 급격히 증가하는 것을 확인할 수 있다.

포화 상태에서 동작하는 DCF 시스템의 특성을 좀 더 살펴보자. 포화 상태에서는 각 노드가 자신의 데이터를 전송하기 위하여 이웃 노드와 치열하게 경쟁을 하여야 하고, 경쟁이 치열한 만큼 여러 노드가 동시에 데이터를 전송하여 충돌이 일어날 확률이 높다. 따라서 포화 상태에서는 각 노드의 처리율이 시간에 따라 크게 변하게 된다. 그림 4는 16개의 노드가 존재하는 시스템에서, 한 노드의 처리율이 시간에 따라 변화하는 모습을 보이고 있다. 시스템 입력 로드가 아직 포화 영역에 진입하기 전인 5.1Mbps인 경우에는 안정적인 노드 처리율을 볼 수 있다. 그에 비하여 포화 영역에 진입하게 만드는 입력 로드인 5.2Mbps 경우에는 노드의 처리율이 심하게 변동하는

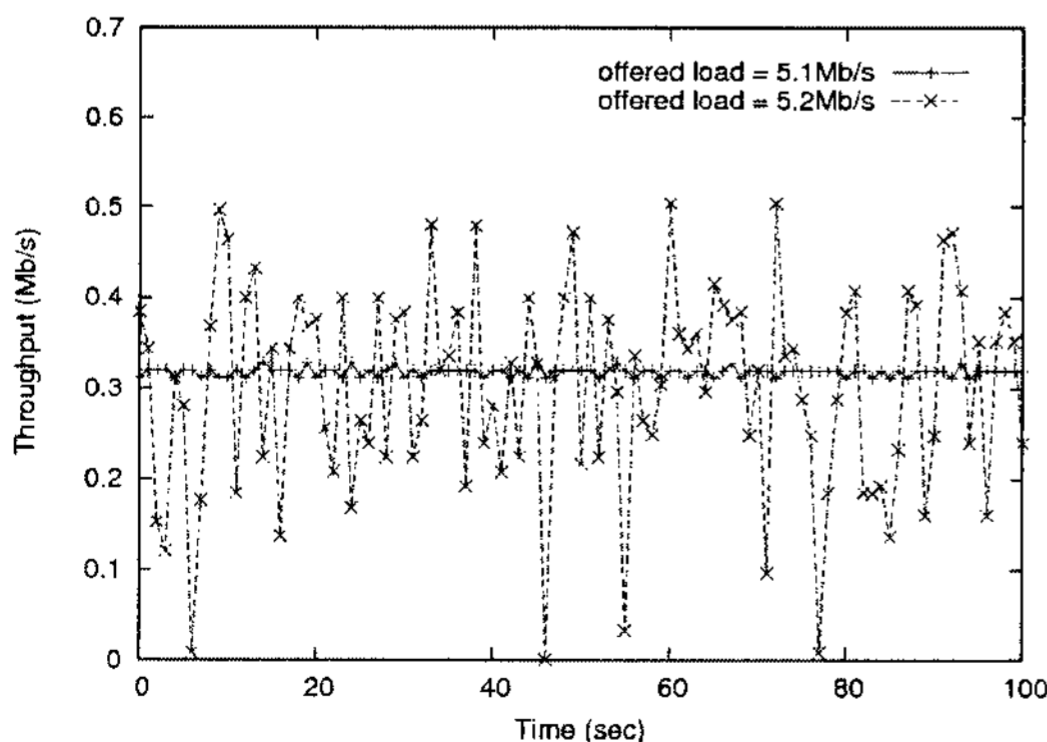


그림 4. 경계 시스템 입력 로드 전(5.1Mbps)과 후(5.2Mbps)의 시간에 따른 처리율 변화  
Fig. 4 Throughput variation over time before (5.1 Mbps) and after (5.2 Mbps) critical offered load

것을 볼 수 있다. 이와 같이 노드의 처리율이 불안정한 모습으로 변화한다면 해당 노드에게 안정적인 QoS를 제공하는 것은 어려운 문제가 될 수밖에 없다.

#### IV. 트래픽 제어

앞에서 우리는 포화 상태에서 동작하는 DCF 시스템의 경우에, 시스템 처리율이 저하되고, 각 노드의 처리율이 시간에 따라 심하게 변화하는 것을 보았다. 우리는 링크 계층에서 802.11 MAC 계층으로 전달하는 트래픽의 속도를 제어함으로써, 이와 같은 문제를 해결하는 방안을 모색하고자 한다. MAC 계층을 수정하지 않고 상위 링크 계층만 수정하기 때문에, 기존의 WLAN 제품들을 그대로 사용할 수 있는 장점이 있다.

##### 4.1. 정적 제어 (Static control)

최적의 시스템 입력 로드를 자동적으로 찾아 나가는 동적 제어 알고리즘을 고려하기에 앞서, 시스템의 입력 로드를 임의의 정해진 값으로 강제하는 정적 제어를 사용하여 트래픽 속도를 제어하는 것의 효과에 대하여 알아보자.

그림 5는 트래픽 속도 제어기의 위치를 보여준다. 트래픽 속도 제어기는 링크 계층과 MAC 계층 사이의 버퍼에 위치한다. 버퍼에서 MAC으로 내보내는 트래픽의 양을 일정한 수준으로 제어하여, DCF 시스템으로 유입되는 로드를 제한한다. 상위 계층에서  $\lambda_{original}$ 의 속도로 유

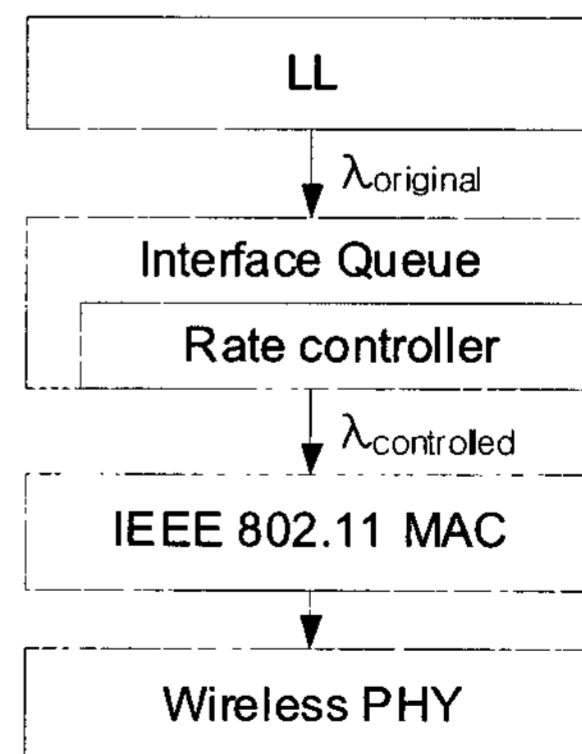


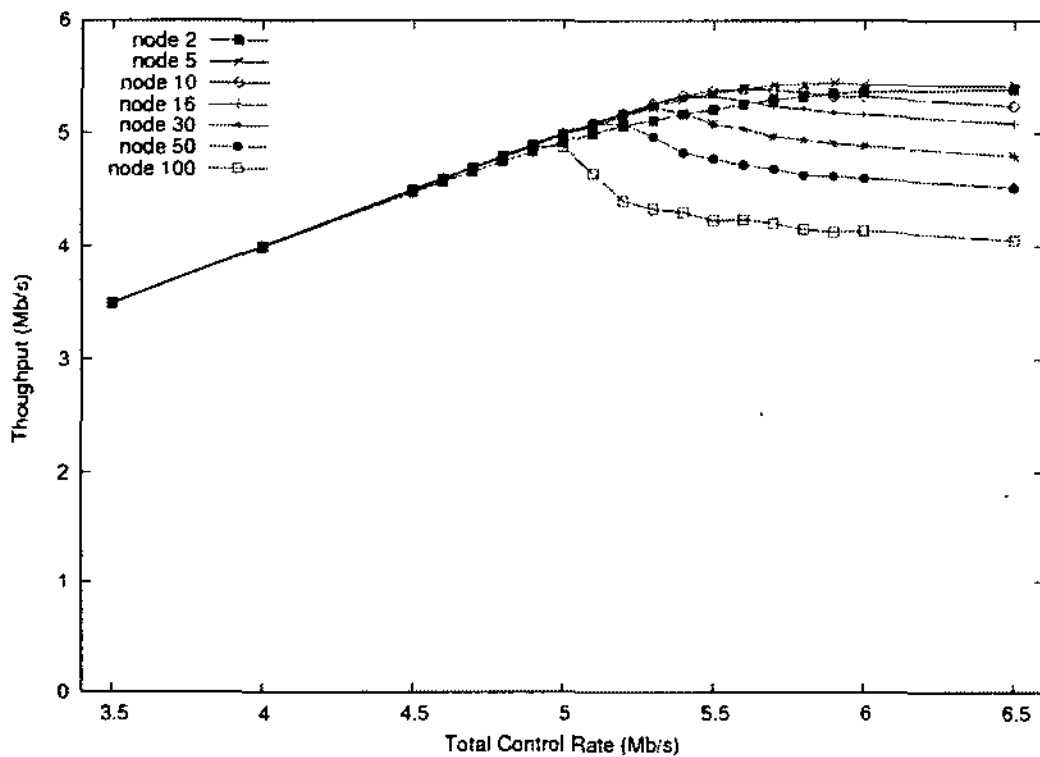
그림 5. 프로토콜 스택에서 트래픽 속도 제어기의 위치  
Fig. 5 The position of traffic rate controller in the protocol stack

입되는 트래픽을, IEEE 802.11 MAC으로 전달할 때에는  $\lambda_{controlled}$  이하의 속도로 제한한다. 정적 제어기란  $\lambda_{controlled}$  값이 일정한 값으로 계속 유지되는 제어기를 말하고, 동적 제어기란 상황에 따른 최적의 설정값을 찾기 위하여  $\lambda_{controlled}$  값이 계속해서 변하는 제어기를 말한다.

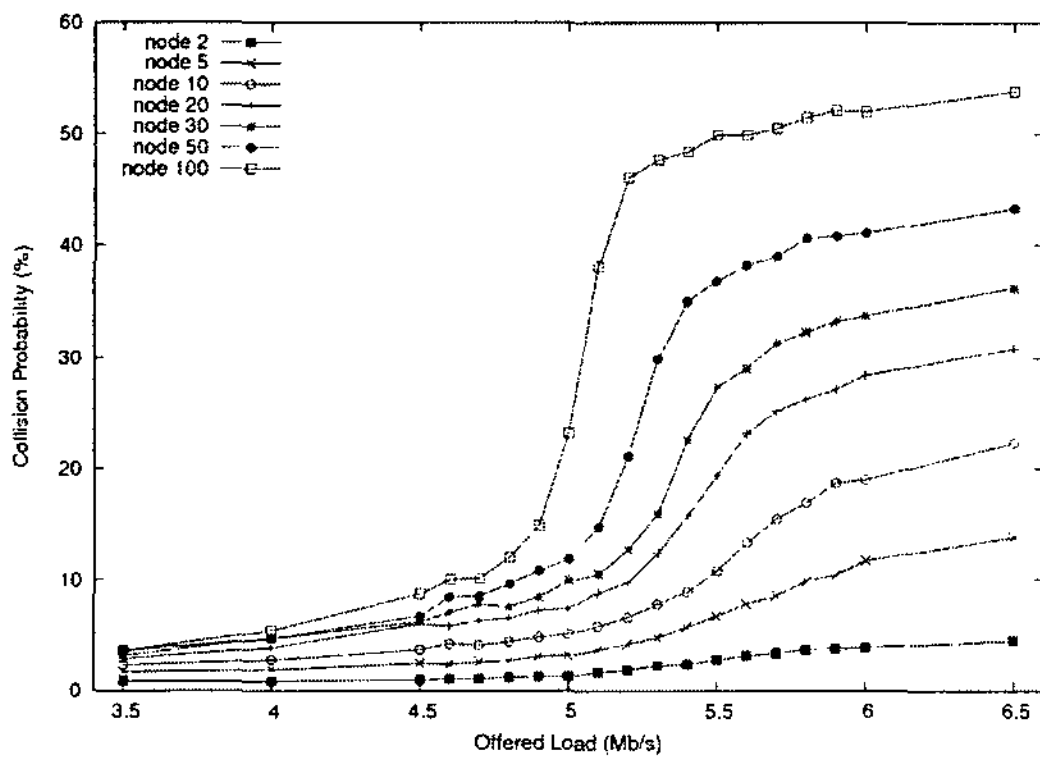
그림 6은 그림 3과 동일한 시뮬레이션 환경에서 정적 트래픽 속도 제어기를 사용했을 때의 DCF 시스템 성능을 보여 준다. 응용 계층에서 6.5Mb/s의 속도로 트래픽을 발생시키는 포화 상태에서, 트래픽 속도 제어기를 사용하여 MAC으로 유입되는 트래픽을 일정한 수준으로 제어하여 DCF 시스템 로드를 제한한다. X축은 각 노드의 트래픽 속도 제어가 제한하는 트래픽 양의 총합을 의미한다. 예를 들어, 노드의 개수가 10개일 때 X축의 좌표값이 5Mb/s라면, 각 노드의 트래픽 속도 제어기는 MAC으

로 유입되는 트래픽 양을 500kb/s로 제한한다. MAC으로 유입되는 트래픽의 속도를 적절한 속도로 제어하면, 시스템의 성능을 최대로 할 수 있는 것을 볼 수 있다.

한 가지 더 특기할 사항은, 트래픽 속도 제어기를 사용할 때 아무런 제어를 하지 않는 경우에 비하여 시스템의 최대 성능이 증가하는 것이다. 50개의 노드가 존재할 때, 트래픽 속도 제어를 하지 않은 경우에 DCF의 최대 성능은 4.80Mb/s(그림 3(a) 참조) 인데 비해, 트래픽 속도를 제어한 그림 6(a)에서는 최대 성능이 5.09Mb/s로 6% 정도 높아진 것을 볼 수 있다. 이것은 트래픽 속도 제어기에서 MAC으로 진입하는 패킷 사이의 시간 간격을 일정 시간 이상으로 강제하기 때문에, MAC 계층에서의 패킷 전송 충돌을 줄일 수 있기 때문이다. 패킷 충돌 확률이 감소한 것을 그림 6(b)에서 확인할 수 있다.



(a)



(b)

그림 6. (a) 정적 제어 시의 시스템 처리율. (b) 정적 제어 시의 충돌 확률

Fig. 6 (a) Throughput profile under static rate control (b) Corresponding DCF collision rate.

#### 4.2. 동적 제어 (Dynamic control)

다음으로 최적의 시스템 입력 로드를 자동적으로 찾아 가는 동적 제어 알고리즘을 고려하자. 그림 6(a)에서 정적 제어기를 사용한 경우에 제어 속도와 DCF 시스템 처리율 사이의 관계가 돔 (dome) 모양을 보이고 있으므로, 점근적으로 안정적이고 효율적인 PI 제어기(Proportional-Integral controller)를 만드는 것이 어렵지 않다는 것을 알 수 있다.

알고리즘 1 제어 목표 속도를 갱신하는 알고리즘

Algorithm 1. Target control rate update algorithm

Input  $\alpha, \epsilon$  := 상수값

Input  $\lambda_{achieved}$  := 최근 제어 목표 속도를 갱신한 이후에 실제 전송한 트래픽의 평균 속도

Input  $\lambda'_{achieved}$  := 이전 갱신 구간의  $\lambda_{achieved}$  값

Input/Output  $\lambda_{target}$  := 제어 목표 속도

Input/Output  $\Delta\lambda$  := 제어 목표 속도의 증가량

1: function update()

2:  $\Delta\lambda_{achieved} = \lambda_{achieved} - \lambda'_{achieved}$

3: if ( $\Delta\lambda_{achieved} / \Delta\lambda > 0$  and  $\lambda_{achieved} > \alpha \lambda_{target}$ ) then

4:      $\Delta\lambda = \epsilon$

5:     else

6:      $\Delta\lambda = -\epsilon$

7:     end if

8:      $\lambda_{target} += \Delta\lambda$

9:      $\lambda'_{achieved} = \lambda_{achieved}$

10: end function

직관적으로 현재의 목표 제어 속도가 최고의 DCF 시스템 처리율을 보이는 제어 속도보다 작을 때에는 목표 제어 속도를 높이고, 반대로 최고의 처리율을 보이는 제어 속도보다 클 때에는 목표 제어 속도를 낮추어야 한다. 이와 같이 자동으로 최적의 목표 제어 속도를 찾아 가는 동적 제어기를 구현하기 위하여 알고리즘 1을 제안한다.

알고리즘 1은 현재의 제어 목표 속도를 최적의 제어 목표 속도로 수렴시키는 것을 목적으로 한다. 이 알고리즘을 주기적으로 실행함으로써, 제어 목표 속도를 갱신한다. 이 제어 알고리즘의 핵심은 현재의 목표 제어 속도가 최적의 목표 제어 속도보다 작은 상태인지 아니면 큰 상태인지를 정확하게 판단하는 것에 있다. 이 판단은 3번째 줄에서 수행되는데, 기본적으로는 제어 속도와 DCF 시스템 처리율 사이의 관계가 돔 (dome) 모양인 것을 활용한다. 현재의 목표 제어 속도가 최적의 목표 제어 속도보다 작은 상태에서는 기울기가 양수이다. 즉, 목표 제어 속도를 높이면 DCF 시스템 처리율이 증가하고, 목표 제어 속도를 낮추면 DCF 시스템 처리율도 그에 따라 낮아질 것이다. 이것을 식으로 나타내면,  $\Delta\lambda_{achieved} / \Delta\lambda > 0$ 와 같다. 단, 목표 제어 속도가 너무 클 경우에는 DCF 시스템이 포화 상태로 들어가게 되어 각 노드의 처리율이 돔 모양의 그래프와는 달리 불안정한 특성을 가질 수 있다. 이러한 경우를 구별하기 위하여  $\lambda_{achieved} > \alpha \lambda_{target}$ 와 같은 예외 처리 검사가 추가되었다. 일단 판단이 내려지면, 목표 제어 속도를 갱신하는 일은 간단하다. 만일 작은 상태라면 목표 제어 속도를  $\epsilon$ 만큼 증가시키고(4번째 줄), 반대의 경우에는  $\epsilon$ 만큼 감소시킨다(6번째 줄).

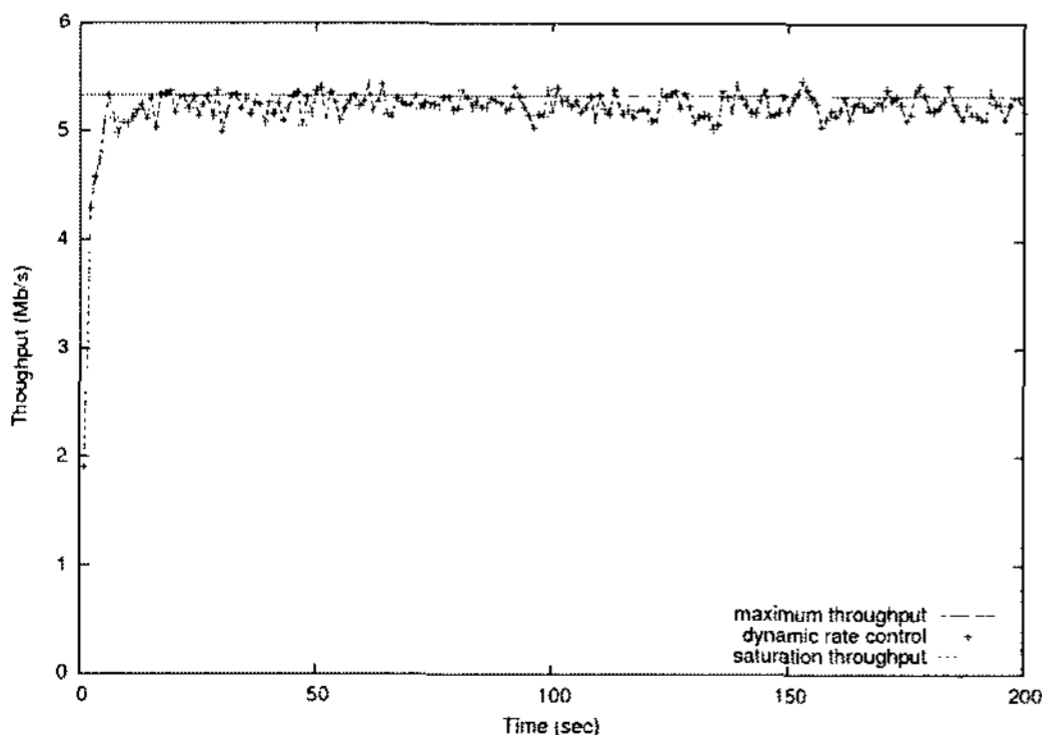


그림 7. 동적 제어 시의 시스템 처리율 변화  
Fig. 7 Throughput evolution under dynamic rate control

그림 7은 트래픽 동적 제어 기능을 갖는 16개의 노드가 달성하는 DCF 시스템 처리율을 시간에 따라 나타낸 그림이다.  $\alpha$ 와  $\epsilon$ 는 각각 0.9, 10kb/s를 사용하였고, 목표 제어 속도는 0.5초 마다 갱신하였다. 트래픽 입력은 제어를 하지 않는 일반 DCF 시스템의 경우에는 포화 상태가 될 만큼 충분히 크다. 동적 제어기를 사용하였을 때, 제어를 하지 않는 경우에 비하여 시스템 처리율의 향상을 볼 수 있다. 초기 수렴 시간을 거친 후에는 시스템 처리율이 최대값 부근에서 안정적으로 동작하고 있다.

제어를 통하여 얻어지는 이득은 시스템 처리율의 향상뿐만 아니라, DCF 시스템이 동작하는 영역을 포화 상태에 진입하기 이전 영역으로 옮김으로써 해서 각 노드의 처리율 변화를 줄이는 효과도 있다. 그림 8은 제어하지 않았을 때, 정적 제어를 할 때, 동적 제어를 할 때의 노드의 처리율이 변화하는 정도를 비교한 것이다. 처리율이 변화하는 정도는 처리율의 표준편차를 구하여 나타내었다.

우선 트래픽 제어하는 경우의 노드의 처리율 변화가 제어를 하지 않는 경우에 비하여 작은 것을 볼 수 있다. 특히, 정적 제어를 하는 경우에는 현저히 줄어드는 것을 볼 수 있다. 그에 비하여 동적 제어는 많이 부족하는데, 이것은 최적의 제어 속도를 찾는 과정에서 포화 상태에 진입했다가 다시 이전 상태로 되돌아오는 과정을 반복하기 때문이다. 하지만, 정적 제어를 하기 위해서는 최적의 제어 속도를 알고 있어야 한다. 이는 동적으로 상황이 변하는 시스템에서는 사용하기 어려운 방식이다.

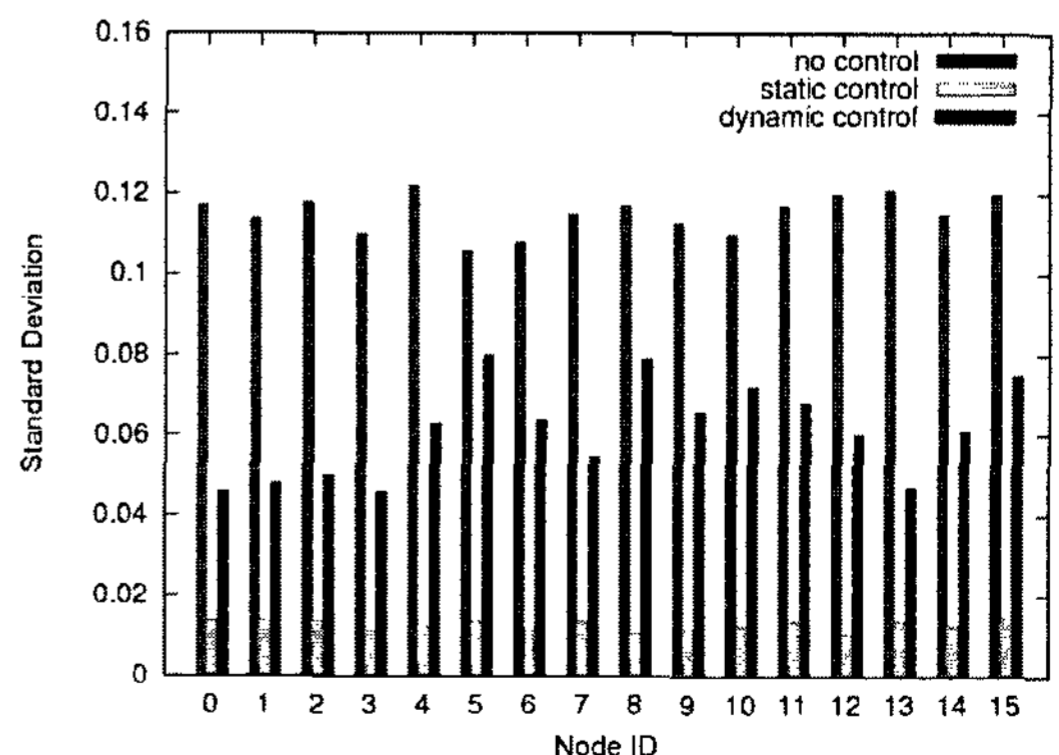


그림 8. 노드 처리율이 변하는 정도 비교: 무 제어, 정적 제어, 동적 제어  
Fig. 8 Throughput variance: no control, static control, and dynamic control

## V. 결 론

본 논문에서는 시스템 로드에서 IEEE 802.11 DCF의 처리율과 처리율의 변화에 대하여 고찰하고, 링크 계층에서 DCF의 입력 로드를 조정하는 트래픽 제어기를 제안하였다. DCF 시스템은 포화 상태에서는 시스템 처리율이 저하되고, 노드의 처리율이 불안정하게 변화하는 특성을 갖고 있다. 트래픽 제어기를 사용하여 링크 계층에서 DCF로 유입되는 시스템 로드를 적절하게 조정하면 시스템 처리율을 향상시키고, 노드의 처리율 변화를 줄일 수 있는 것을 보였다. 그리고, 최적의 시스템 로드를 찾을 수 있는 동적 제어 알고리즘을 제안하였다. 본 논문에서 제안한 트래픽 속도 제어 방법은 향후 무선 랜 연구의 기초로 활용될 수 있을 것이다.

## 참고문헌

- [ 1 ] IEEE 802.11 part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, Aug. 1999.
- [ 2 ] N. Abramson, "The Aloha system - another alternative for computer communications," Proc. of Fall Joint Comput. Conf. AFIPS Conf., pages 281 - 285, 1970.
- [ 3 ] R. Metcalfe and D. Boggs, "Ethernet: distributed packet switching for local computer networks," Communications of the ACM, vol. 19, no. 7, pp. 395 - 404, 1976.
- [ 4 ] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE Journal on Selected Areas in Communications, vol. 18, no. 3, pp. 535 - 547, 2000.
- [ 5 ] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 protocol: Design and performance evaluation of an adaptive backoff mechanism," IEEE Journal on Selected Areas in Communications, vol. 18, no. 9, pp. 1774 - 1786, 2000.
- [ 6 ] S. Lu, T. Nandagopal, and V. Bharghavan, "A wireless fair service algorithm for packet cellular networks," Proc. of ACM MobiCom '98, pp. 10 - 20, 1998.
- [ 7 ] N. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN," Proc. of ACM MobiCom '00, pp. 167 - 178, 2000.
- [ 8 ] A. Balachandran, G. Voelker, P. Bahl, and P. Rangan, "Characterizing user behavior and network performance in a public wireless LAN," Proc. of ACM SIGMETRICS '02, pp. 195 - 205, 2002.
- [ 9 ] D. Kotz and K. Essien, "Analysis of a campus-wide wireless network," Proc. of ACM MobiCom '02, pp. 107 - 118, 2002.
- [ 10 ] D. Tang and M. Baker, "Analysis of a local-area wireless network," Proc. of ACM MobiCom '00, pp. 1 - 10, 2000.
- [ 11 ] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh, "A Client-driven Approach for Channel Management in Wireless LANs," Proc. of Infocom '06, pp. 1 - 12, 2006.
- [ 12 ] B. Kauffmann, François Baccelli, Augustin Chaintreau, Vivek Mhatre, Konstantina Papagiannaki, and Christophe Diot, "Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks," Proc. of Infocom '07, pp. 1451 - 1459, 2007.
- [ 13 ] A. Mishra, V. Shrivastava, D. Agrarwal, S. Bangerjee, and S. Ganguly, "Distributed Channel Management in Uncoordinated Wireless Environments," Proc. of Mobicom '06, pp. 170 - 181, 2006.
- [ 14 ] E. Rozner, Y. Mehta, A. Akella, and L. Qiu, "Traffic-Aware Channel Assignment in Enterprise Wireless LANs," Proc. of ICNP '07, pp. 133 - 143, 2007.
- [ 15 ] D. Boggs, J. Mogul, and C. Kent, "Measured capacity of an Ethernet: myths and reality," Proc. of ACM SIGCOMM '88, pp. 222 - 234, 1988.
- [ 16 ] H. Kim and J. Hou, "Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs," Proc. of ACM MobiCom '03, pp. 190 - 204, 2003.
- [ 17 ] The network simulator - ns-2. <http://www.isi.edu/nsnam/ns/>.

## 저자소개



최 선 웅(Sun-Woong Choi)

2005년 서울대학교 전기, 컴퓨터  
공학 (공학박사)

2005년 9월 ~ 2007년 2월 삼성전자  
정보통신총괄 책임연구원

2007년 3월 ~ 현재 국민대학교 전자공학부 전임강사

※관심분야: 무선 네트워크, 네트워크 자원관리, 시스템  
성능 평가