

# CARA: IEEE 802.11 무선랜에서 충돌을 인지한 적응적 전송속도 조절기법

준회원 김 종 석\*, 정회원 김 성 관\*, 종신회원 최 성 현\*\*

## CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs

Jongseok Kim\* *Associate Member*, Seongkwan Kim\* *Regular Member*,  
Sunghyun Choi\*\* *Lifelong Member*

### 요 약

IEEE 802.11 무선랜(Wireless LAN)은 다수의 전송속도(transmission rate)를 지원하고 있으며, 채널 상태에 따라 전송속도를 적응적으로 조절함으로써 시스템 성능을 최대화할 수 있다. 지금까지 많은 종류의 전송속도 조절기법들이 제안되어 왔으나 최근까지 대부분의 상용제품에 구현된 기법은 ARF(Automatic Rate Fallback)라는 간단한 open-loop 전송속도 조절기법이다. 이러한 open-loop 전송속도 조절기법의 가장 큰 문제점은 데이터의 충돌현상(collision effect)을 고려하지 않는다는 것이며, 이로 인해 충돌에 의한 다수의 전송실패가 발생할 경우 시스템 성능이 급격히 낮아진다는 문제점이 있다. 이 논문에서는 CARA(Collision-Aware Rate Adaptation)라는 새로운 전송속도 조절기법을 제안하고 있다. CARA는 송신 단말이 채널에러에 의해 발생한 데이터 전송실패를 충돌에 의해 발생한 것과 구분할 수 있도록 적응적으로 CCA(Clear Channel Assessment)와 RTS/CTS(Request-to-Send/Clear-to-Send) 전송을 사용하는 기법이다. 따라서 기존의 open-loop 전송속도 조절기법과 비교할 때, CARA는 보다 정확하게 현재 채널상태에 적합한 전송속도를 선택할 수 있게 된다. 시뮬레이션을 통한 많은 실험결과들로부터 CARA는 채널상태에 관계없이 다른 기법들보다 월등히 높은 성능을 나타낸다는 것을 알 수 있다.

**Key Words** : Wireless LAN, collision, channel error, rate adaptation

### ABSTRACT

Today's IEEE 802.11 WLANs(Wireless LANs) provide multiple transmission rates so that different rates can be exploited in an adaptive manner depending on the underlying channel condition in order to maximize the system performance. Many rate adaptation schemes have been proposed so far while most(if not all) of the commercial devices implement a simple open-loop rate adaptation scheme(i.e., without feedback from the receiver), called ARF(Automatic Rate Fallback) due to its simplicity. A key problem with such open-loop rate adaptation schemes is that they do not consider the collision effect, and hence, malfunction severely when many transmission failures are due to collisions. In this paper, we propose a novel rate-adaptation scheme, called CARA(Collision-Aware Rate Adaptation). The key idea of CARA is that the transmitter station combines adaptively the Request-to-Send/Clear-to-Send(RTS/CTS) exchange with the Clear Channel Assessment(CCA) functionality to differentiate frame

\* 서울대학교 전기공학부 멀티미디어 무선통신망 연구실 (jjskim, skim)@mwml.snu.ac.kr

\*\* 서울대학교 전기공학부 멀티미디어 무선통신망 연구실 (schoi@snu.ac.kr)

논문번호 : KICS2005-11-451, 접수일자 : 2005년 11월 4일

collisions from frame transmission failures cause by channel errors. Therefore, compared with other open-loop rate adaptation schemes, CARA is more likely to make the correct rate adaptation decisions. Through extensive simulation runs, we evaluate our proposed scheme to show that our scheme yields significantly higher throughput performance than the existing schemes in both static and time-varying fading channel environments.

## I. 서론

지난 10여 년 동안, IEEE 802.11 무선랜은 광대역 무선 네트워크를 위한 주도적인 기술로 성장해왔다. 802.11 표준은 매체접근제어(MAC) 계층과 물리(PHY) 계층에 대한 세부사항<sup>[1]</sup>을 정의하고 있다. 802.11의 채널접근 방법 중 우선적으로 사용되는 DCF(Distributed Coordination Function)는 경쟁에 기반을 둔 채널접근 방법으로서 이는 Carrier Sense Multiple Access/Collision Avoidance(CSMA/CA) 기법을 적용하고 있다. 부가적으로 사용되는 PCF(Point Coordination Function)는 네트워크 통제 단말의 제어에 의해 통신하는 것에 기반을 둔 방법으로서 표준에 정의되어 있으나 대부분의 802.11 제품군에는 거의 사용되고 있지 않다. 다양한 종류의 변조기법 및 채널 부호화 기법을 적용함으로써 802.11 PHY는 다수의 전송속도를 지원한다. 예를 들어, 802.11 표준은 1과 2 Mbit/s(Mbps)로 동작하는 저속의 PHY에 대해 정의하고 있으며, 아래의 세 가지 고속 PHY가 보조 표준에 정의되어 있다. (1) 802.11b PHY<sup>[2]</sup>는 2.4 GHz 대역에서 11 Mbps 까지 4개의 전송속도를 지원하고, (2) 802.11a PHY<sup>[3]</sup>는 5 GHz 대역에서 54 Mbps까지 8개의 전송속도를 지원한다. 그리고 (3) 802.11g PHY<sup>[4]</sup>는 2.4 GHz 대역에서 54 Mbps까지 12개의 전송속도를 지원한다.

802.11 표준에는 다수의 전송속도를 효율적으로 활용하기 위한 어떠한 알고리즘이나 프로토콜이 정의되어 있지 않으므로 지금까지 다양한 종류의 전송속도 조절기법들<sup>[11-13, 15, 18-21]</sup>이 제안되었다. 전송속도 조절기법의 효율성은 무선 채널의 변화에 얼마나 신속하게 적응할 수 있는냐에 달려있다. 또한, 802.11 DCF는 경쟁에 의한 채널 접근이라는 특성으로 인해 다수의 사용자가 존재하는 환경에서 데이터의 충돌은 부득이 하게 발생하게 되는데, 이러한 환경에서 무선랜 단말이 어떻게 충돌을 감지하며 또한 어떠한 방법으로 충돌을 적절하게 조절할 수 있는가에 전송속도 조절기법의 효율성이 달려있다.

그렇지만 대부분의 open-loop 전송속도 조절기법

들은 충돌을 고려하지 않고 있기 때문에 이로 인해 다수의 전송실패가 발생할 경우 채널의 상태가 나빠진 것으로 판단하여 전송속도를 낮춤으로써 전체적으로 시스템 성능이 낮아지는 문제점이 있다. 예를 들어, 현재 가장 널리 사용되고 있는 ARF<sup>[15]</sup>는 그림 [10]의 실험결과처럼 연속적인 데이터 충돌이 발생할 경우 전송속도를 과도하게 낮추게 되므로 다수의 사용자가 경쟁하는 환경에서는 제대로 동작하지 않는 현상을 나타낸다. 반면 RBAR<sup>[13]</sup>와 OAR<sup>[18]</sup>과 같은 closed-loop 전송속도 조절기법에서는 송신 단말과 수신 단말의 상호 작용으로 인해 충돌현상은 거의 나타나지 않는다.

위에서 설명한 내용을 기반으로, 이 논문에서는 CARA(Collision-Aware Rate Adaptation)라는 새로운 전송속도 조절기법을 제안한다. CARA의 핵심은 송신 단말이 채널에러에 의해 발생한 데이터 전송실패를 충돌에 의해 발생한 것과 구분할 수 있도록 적응적으로 Clear Channel Assessment(CCA)와 Request-to-Send/Clear-to-Send(RTS/CTS) 전송을 사용하는 기법이다. 따라서 다른 open-loop 전송속도 조절기법과 비교할 때, CARA는 보다 정확하게 현재 채널상태에 적합한 전송속도를 선택할 수 있게 된다. 또한 CARA는 802.11 프로토콜의 주요 기능인 RTS/CTS와 CCA를 이용하는 기법으로서 802.11 표준에 부합하며 802.11 표준을 따르는 어떠한 장치에도 용이하게 구현될 수 있다는 장점이 있다.

이 논문의 구성은 다음과 같다. 관련 연구는 II장에 기술되어 있으며 III장에서는 802.11 표준의 일반적인 내용과 ARF에 대해 간략하게 설명할 것이다. IV장에서는 CARA의 세부적인 내용을 기술하고 V장에서는 시뮬레이션을 통해 성능을 분석하였으며, VI장에서 향후 연구계획과 함께 결론을 맺는다.

## II. 관련 연구

지금까지 802.11 무선랜의 전송속도 조절에 대해서는 많은 연구결과가 있다. 전송속도 조절을 위해 송신 단말은 수신 단말로부터 피드백을 받거나 또는 받지 않는 방법을 사용할 수 있다. 피드백으로

전송되는 정보는 신호 대 간섭/잡음 비율(SINR) 또는 수신 단말이 판단한 전송 단말의 전송속도 등이 될 수 있다. 수신 단말로부터 전송되는 피드백 정보를 사용하느냐의 여부에 따라 전송속도 조절기법은 closed-loop과 open-loop의 두 가지 접근방식으로 구분된다.

closed-loop 접근방식<sup>[13, 18]</sup>에서는, 수신 단말이 채널 상태에 적합한 전송속도를 결정하고 이 정보를 변형된 RTS/CTS 전송을 통해 송신 단말에 전송하며, 송신 단말은 수신된 정보를 이용하여 전송속도를 조절한다. 이러한 형태의 전송속도 조절기법에서는 송신 단말과 수신 단말의 상호작용을 통해 프레임의 충돌현상이 사전에 예방될 수 있다. 그러나 이러한 피드백을 제공하기 위해서는 CTS 프레임의 구조가 정보를 전송하기에 적절한 형태로 변형되어야하며 이러한 프레임 구조의 변형은 802.11 표준을 따르지 않는다는 문제점이 있다. 또한 RTS/CTS를 데이터 프레임 전송 전에 항상 전송하는 것은 무선자원을 상당히 낭비하는 것으로서, hidden 단말이 존재하지 않는 경우에는 시스템 성능이 낮아지는 문제점이 있다. 인프라스트럭처에 기반을 둔 무선랜에서는 위와 같은 이유로 인해 RTS/CTS 전송은 거의 사용되지 않고 있으나, hidden 단말이 존재하는 다중 홉의 애드 혹 네트워크에서는 항상 사용되는 것이 바람직하다.

open-loop 접근방식에서는 송신 단말이 Acknowledgement(Ack) 정보를 이용하여 단독으로 전송속도를 조절한다. 802.11 표준에서 수신 단말은 성공적으로 데이터 프레임을 수신하였을 경우 Ack 프레임을 전송한다. 따라서 송신 단말은 Ack 프레임을 수신한 이후에 데이터 프레임이 성공적으로 전송되었다고 판단할 수 있게 된다. 반면에 Ack 프레임이 정상적으로 수신되지 않았거나 또는 전혀 수신되지 않았을 경우 데이터 프레임 전송이 실패하였다고 추정한다. open-loop 접근방식은 송신 단말과 수신 단말 사이에 어떠한 상호작용이 필요 없는 간단한 방법으로 표준에 부합한다.

open-loop 접근방식은 다음의 두 가지 방식으로 나누어 볼 수 있다. 첫 번째 방식은 송신 단말이 채널상태를 예측하여 전송속도를 조절하는 것으로서, 수신된 Ack 프레임을 이용하여 송신 단말과 수신 단말 간의 무선 채널 환경을 예측하는 방법<sup>[12, 19, 20]</sup>이다. 이러한 종류의 방식은 closed-loop 기법들과 유사한 성능을 나타내지만 구현하기가 어렵다는 문제점이 있다. 두 번째 방식은 전송속도를 조절할

때 Ack 프레임의 수신 여부만을 이용하는 방법으로서 구현이 매우 간단하기 때문에 대부분의 802.11 무선랜 상용 제품들이 이 방식에 속하는 방법 중의 하나인 ARF를 적용하고 있다.

[21]에는 전송속도 조절기법을 고안할 때의 두 가지 기본적인 고려사항이 나타나 있다. 첫째, 언제 전송속도를 올리고 둘째, 언제 전송속도를 낮출 것인가에 대한 것이다. 전송속도 조절기법의 효율성은 무선 채널의 변화에 얼마나 빠르게 대응할 수 있느냐에 달려 있다. [11], [12]에서 제안된 기법들은 첫 번째 고려사항을 기술하고 있다. 즉, 송신 단말이 시간에 따라 변화하는 무선채널의 환경에 따라 적응적으로 전송속도를 높이는 것으로, ARF의 성능을 향상시킨 방식들이다. 이 논문에서는 두 번째 고려사항인 언제 전송속도를 낮출 것인가만을 고려하고 있다. 우리가 알고 있는 바로는, 대부분의 현존하는 open-loop 전송속도 조절기법들은 네트워크에 프레임 충돌을 야기하는 다수의 경쟁 단말들이 동작하고 있을 경우 시스템 성능이 상당히 낮아진다는 문제점이 있다. 즉, 이러한 기법들은 프레임 충돌에 의해 발생하는 전송실패를 채널에러에 의해 발생하는 것과 구분할 수 없으므로 전송속도를 과도하게 낮추는 결과를 초래하게 된다.

### III. IEEE 802.11 기본 특성 및 ARF

#### 3.1 IEEE 802.11 CSMA/CA

IEEE 802.11 표준은 매체접근제어 방법으로 우선적으로 사용되는 DCF와 부수적인 PCF<sup>[11]</sup> 등의 두 가지 방식을 정의하고 있다. 현재 대부분의 802.11 무선랜 장치들은 구현의 용이성으로 인해 DCF만을 적용하고 있다. DCF는 그림 1과 같이 CSMA/CA에 기반하고 있다. 즉, 프레임 전송이 준비된 단말은 채널의 상태를 확인하며, 만약 채널이 사용 중이라면 단말은 현재 채널을 사용하고 있는 단말의 프레임 전송이 끝날 때까지 기다린다. 이는 DCF가 CSMA를 한다는 것을 의미한다. 이후 채널

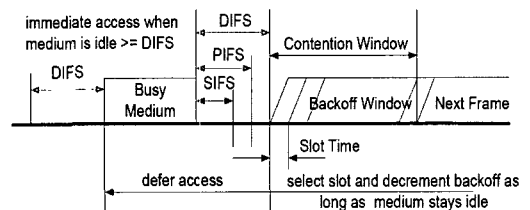


그림 1. IEEE 802.11 DCF의 채널접근 과정

이 사용 중이 아닐 경우 단말은 바로 전송을 시작하는 대신 충돌 확률을 낮추기 위해 임의의 백오프(Backoff) 시간을 선택한다. 이는 DCF가 충돌 회피(CA) 프로토콜이라는 것을 의미한다.

임의의 백오프를 사용하더라도 전송된 프레임은 백오프가 동시에 끝난 두개 혹은 그 이상의 단말이 있을 경우 여전히 충돌할 가능성이 있다. 그러한 프레임 충돌들은 DCF의 경쟁 특성상 완전히 제거될 수 없으며 이러한 문제는 경쟁 단말의 수가 증가할수록 더욱 심각해진다. 또한 충돌과는 별도로 프레임 전송실패는 채널에러에 의해서도 발생할 수 있다.

PHY가 채널이 사용 중인지 또는 그렇지 않은지를 판단하는 방법을 CCA라고 한다. 안테나 종단에서 측정된 에너지 레벨이 일정 임계값(threshold) 이상일 경우 PHY에 의해 CCA Busy가 선언된다. 이러한 CCA는 CSMA/CA 프로토콜의 핵심 부분이다. 이 논문에서는 단말이 전송한 프레임과 다른 단말이 전송한 프레임간의 충돌을 감지하기 위해 CCA를 사용하는 것을 제안하고 있다.

### 3.2 IEEE 802.11에서의 RTS/CTS 전송

네트워크에 hidden 단말이 존재하는 경우, CSMA/CA의 성능은 크게 낮아질 수 있다. hidden 단말로 부터 전송되는 프레임과의 충돌로 인한 자원(Bandwidth)의 손실은 RTS와 CTS 같은 작은 제어 프레임들을 데이터 프레임 전송 전에 미리 전송함으로써 감소될 수 있고, 이렇게 함으로써 hidden 단말 문제가 개선될 수 있다. 이것이 RTS/CTS 전송의 근본적인 목적이다. RTS/CTS 전송 과정은 그림 2에 자세하게 나타나 있다. 성공적인 RTS/CTS 전송 이후에 무선 채널은 송신 단말의 데이터 전송을 위해 예약된다. 802.11 표준에 따르면, RTS 프레임을 사용할지의 여부는 전적으로 송신 단말이 결정한다. 즉, RTS는 단말이 전송할 데이터 프레임의 크기가 RTS 전송 임계값(RTS Threshold)보다 크거나 또는 같은 경우에 전송되게 된다. 그러나 AP(Access

Point)가 설치되어 있는 인프라스트럭처에 기반을 둔 무선랜에서 동작하고 있는 대부분의 802.11 장치들은 RTS 전송 임계값이 가장 큰 값(2347 바이트)으로 설정되어 있기 때문에 원천적으로 RTS/CTS 사용이 제한되어 있다. 따라서 RTS/CTS 프레임은 일반적인 무선랜에서는 거의 사용되지 않고 있다.

또한 RTS/CTS는 많은 전송들이 충돌에 의해서 실패하는 경쟁이 심한 무선랜 환경에서 유용하다고 알려져 있으며 이러한 이점은 데이터 프레임 크기가 클 경우<sup>[7]</sup>에 더욱 효과적일 수 있다. [7]의 저자들은 경쟁하는 단말들의 수와 충돌 확률에 상관없이 충돌에 의해 낭비되는 부수적인 시간은 RTS/CTS 전송을 사용할 경우 무시할 정도의 양이라는 것을 보여주고 있다.

[1]에 정의되어 있는 것처럼 RTS 프레임의 전송은 RTS 전송 임계값에 의해 결정되어야 한다. 그러나 보조 표준에 의하면 별도의 목적을 위해 RTS/CTS 프레임을 사용하는 것에 대해서는 어떠한 제한사항이 없다. IEEE 802.11e 표준<sup>[5]</sup>에 따르면, RTS/CTS는 RTS 전송 임계값에 관계없이 사용될 수 있다. 예를 들어, RTS 프레임은 다수의 데이터 프레임을 연속적으로 전송하기 위한 전송기회(TXOP)라는 시간을 예약하기 위해 사용될 수 있다. 또한 IEEE 802.11g 표준을 사용하는 송신 단말은 RTS/CTS를 전송하거나, CTS 프레임의 수신 측 주소에 자신의 주소를 기입하여 전송할 수 있는데, 이러한 방법은 OFDM 방식으로 변조된, 즉 기본 전송속도를 사용하지 않는 프레임들의 전송을 보장하기 위해 채널을 예약하는 것으로서 802.11g와 802.11b 장치들의 co-existence 문제를 해결하기 위해 사용된다.

이 논문은 충돌에 의해 발생한 전송실패와 채널에러에 의해 발생한 전송실패를 구분하기 위한 수단으로 RTS/CTS를 적응적으로 사용하는 방법을 제안한다. 채널의 상태를 감지(probe)하기 위해 RTS/CTS를 사용하는 것은 엄밀한 의미에서 802.11 표준의 정의에 맞지 않으나 위에서 설명한 것처럼 802.11e와 802.11g와 같은 보조 표준에서 이미 다른 용도로 사용이 가능하게 된 것을 참고한다면 표준에 크게 위배되는 것으로 판단하기는 어렵다. 또한 RTS/CTS를 사용하는 것은 802.11 표준의 필수적인 것으로서 여기서 제안하는 기법은 현존하는 802.11 장치에 곧바로 구현될 수 있으며, 이것이 제안하고 있는 기법의 가장 큰 장점이 될 수 있다.

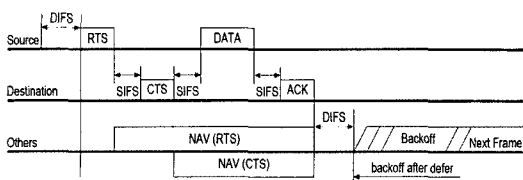


그림 2. IEEE 802.11의 RTS/CTS 전송을 통한 데이터 전송 과정

### 3.3 IEEE 802.11에서의 ARF

802.11에 가장 널리 적용되고 있는 전송속도 조절기법은 Lucent Technologies의 WaveLAN-II<sup>[15]</sup> 장치에서 최초로 고안된 ARF이다. ARF는 전송에 실패한 Ack 프레임의 수 및 타임아웃에 따라서 전송속도를 자동으로 조절한다. 만약 두개의 Ack 프레임이 연속적으로 송신 측에 정확하게 수신되지 않았을 경우 데이터 프레임의 두 번째 재전송 및 그 이후의 데이터 전송은 현재 전송속도 보다 낮은 속도로 전송되며 동시에 타이머가 작동된다. 타이머가 종료되거나 또는 성공적으로 수신된 Ack 프레임의 수가 10개가 될 경우 전송속도는 한 단계 높은 속도로 바뀌게 되고 동시에 작동중인 타이머는 취소된다. 그러나 바로 다음번 데이터 프레임에 대한 Ack 프레임이 수신되지 않을 경우 전송속도는 다시 낮아지게 되며 타이머가 재시작 된다.

ARF는 특성상 무선 채널이 빠르게 변화는 환경에서는 적응적으로 반응하지 못한다. 다시 말해서, 송신 단말은 연속적으로 수신된 성공적인 Ack 프레임의 수에 따라 전송속도를 높이며, 또한 채널에러 또는 프레임 충돌 등 전송실패의 원인을 고려하지 않고 연속적으로 (재)전송이 실패할 경우 전송속도를 낮추게 된다. 그러나 구현의 용이성으로 인해 ARF는 여전히 상용 802.11 무선랜 장치에 가장 널리 적용되고 있으며 이는 현재까지 제안된 다수의 open-loop 전송속도 조절기법([11], [21])들이 ARF에 기반을 두고 있는 이유이기도 하다.

### IV. 충돌을 인정한 전송속도 조절기법(CARA)

이 장에서는 논문에서 제안하고 있는 충돌을 인정한 전송속도 조절기법(CARA)의 세부적인 내용을 설명하고 있다. CARA의 가장 큰 특징은 송신 단말이 수신 단말로부터 어떠한 피드백을 받지 않고 단독으로 프레임 충돌과 채널에러를 구분한다는 것이다.

802.11 무선랜에서 지원하고 있는 다양한 전송속도를 효율적으로 이용하기 위해 여러 가지의 전송속도 조절기법들이 제안되어 왔으며, 이 중에서 ARF는 가장 널리 적용된 전송속도 조절기법 중의 하나이다. ARF를 포함한 대부분의 open-loop 전송속도 조절기법들은 한정된 무선 자원을 점유하기 위해 경쟁하는 다수의 사용자들이 네트워크에 존재할 때, 전송된 데이터 프레임에 대한 Ack 프레임이 수신되지 않은 원인이 프레임 충돌인지 또는 채널

에러인지를 구분하지 못하므로 적절하게 동작하지 못한다.

### 4.1 RTS Probing을 이용한 충돌 구분 방법

CARA는 프레임 충돌과 채널에러를 구분하기 위해 두 가지의 방법을 제안하고 있으며, 그 중에서 RTS Probing 방법이 우선적으로 사용된다. 이 논문에서는 RTS 프레임의 전송 에러 확률은 아주 낮다고 가정하고 있다. 왜냐하면 RTS 프레임은 크기가 상당히 작고(20 바이트), 에러 확률이 가장 낮은 전송속도(1 Mbps)로 전송되기 때문이다. 따라서 RTS 프레임 전송실패의 모든 원인은 충돌 때문이라고 가정할 수 있다. 다른 한편으로, 성공적인 RTS/CTS 전송 이후에 전송되는 데이터 프레임의 전송실패 원인은 채널에러 때문이라고 생각할 수 있다. 왜냐하면 성공적인 RTS/CTS 전송은 이후 데이터 프레임이 경쟁 없이 전송될 수 있도록 무선 채널을 점유하는 상태가 되므로 데이터 전송 시 충돌발생을 사전에 예방하기 때문이다. 그러므로 모든 데이터 프레임 전송 전에 RTS/CTS를 전송하면서 ARF를 적용하면 데이터 충돌은 발생하지 않을 것이며 결과적으로 데이터 프레임 충돌에 의한 전송실패를 채널에러에 의해 발생한 전송실패로 오인하는 것은 일어나지 않을 것이다. 따라서 불필요하게 전송속도를 낮추는 것을 피할 수 있게 된다. 이러한 방법을 적용할 때의 문제점은 RTS/CTS 전송에 따른 오버헤드가 발생한다는 것이며 이로 인해 무선 자원을 낭비하는 결과를 초래한다는 것이다. 사실상, 대부분의 802.11 상용제품에서 RTS/CTS 기능은 사용하지 않도록 설정되어 있다. 위에서 설명한 문제점을 해결하기 위해 이 논문에서는 RTS/CTS를 모든 데이터 프레임 전송 전에 사용하는 대신, 데이터 프레임 전송이 실패하였을 경우에만 RTS/CTS를 사용하는 RTS Probing을 제안한다.

1) 상태전이 도형: RTS Probing의 세부적 진행 과정은 그림 3에 나와 있는 송신 단말의 상태전이 도형 및 관련된 기호와 매개변수가 표현된 표 1에 자세하게 설명되어 있다. 그림 3에 표현된 네 가지의 상태는 다음과 같다.

- *Initial State*: RTS Probing의 시작 상태
- *Wait for MPDU*: 새로운 데이터 프레임이 상위계층으로부터 도달하거나 또는 현재 프레임의 전송실패로 재전송을 하는 상태

- DATA Tx: 단말이 데이터 전송을 종료하고 Ack 프레임 수신을 기다리는 상태
- RTS Tx: 단말이 RTS 전송을 종료하고 CTS 프레임 수신을 기다리는 상태

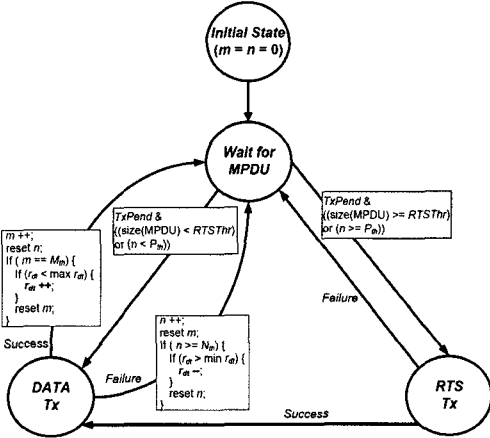


그림 3. RTS Probing의 상태전이 도형

표 1. RTS Probing에서 사용된 용어의 정의

Notations	Comments
$m$	consecutive success count
$n$	consecutive failure count
$M_{th}$	consecutive success threshold
$N_{th}$	consecutive failure threshold
$TxPend$	status: a data frame is pending
$R_{dt}$	array of transmission rates 802.11a = {6, 12, 18, 24, 36, 48, 54 Mbps}* 802.11b = {1, 2, 5.5, 11 Mbps}
$r_{dt}$	transmission rate: an element of $R_{dt}$
++	increase transmission rate to the next higher one
--	decrease transmission rate to the next lower one
$P_{th}$	probe activation threshold
$RTSThr$	frame size-based RTS Threshold as defined in the standard

\* The 9 Mbps rate is excluded as it is shown useless in [19].

그림 3 및 표 1에 나와 있는 것과 같이 연속적인 실패 횟수인  $n$ 은 서로 다른 목적을 가진 두 가지의 임계값, 즉 채널상태 확인을 시작하는 임계값 ( $P_{th}$ )과 연속적 실패 횟수 임계값( $N_{th}$ )과 비교된다.  $n$ 의 값이  $P_{th}$ 와 같아지게 되면 다음에 시작되는 데이터의 재전송 전에 먼저 RTS/CTS를 전송한다. 반면,  $n$ 의 값이  $N_{th}$ 와 같아지게 되면 바로 이후의 데이터 재전송은 한 단계 낮은 전송속도를 이용하여 시도된다.  $P_{th}$ 와  $N_{th}$ 의 설정값에 따라 RTS Probing은 다르게 동작한다. 이 논문에서 제안한 기법에서의  $P_{th}$ 와  $N_{th}$ 의 값은 각각 1과 2이다. 몇 개의 예를 살펴보면 다음과 같다.

•  $P_{th} = 0$ : 이 경우는 모든 데이터 (재)전송 전에

RTS/CTS 프레임이 먼저 전송되며, RTS/CTS 전송이 성공하였을 경우에 데이터 프레임이 (재)전송된다. 데이터 전송실패 횟수( $n$ )가  $N_{th}$ 와 같아질 경우 전송속도는 한 단계 낮아진다.

•  $P_{th} >= 1, N_{th} = 1$ : 이 경우는 모든 데이터 프레임이 RTS/CTS 없이 전송이 되며, 한 번의 데이터 전송실패가 발생하면 전송속도가 바로 낮아지게 된다. 즉,  $P_{th} >= N_{th}$ 일 경우 RTS/CTS 전송은 일어나지 않는다.

•  $P_{th} >= 2, N_{th} = 2$ (ARF와 동일): 이 경우, 두 번의 연속적인 전송실패가 발생하게 되면 RTS/CTS 전송이 없이 전송속도가 낮아지게 된다. 여기에서  $M_{th} = 10$ 일 경우 ARF와 동일하다.

•  $P_{th} = 1, N_{th} = 2$ (CARA의 기본 설정값): 이 경우는 최초의 데이터 프레임 전송은 RTS/CTS 없이 전송되며 만약 데이터 전송이 실패한다면 데이터의 첫 번째 재전송 전에 RTS/CTS가 전송되고, 이후 재전송이 실패한다면 전송속도가 낮아진다.

만약 무선 채널의 상태가 갑자기 나빠지게 되어 RTS와 데이터 전송이 모두 실패한다면 송신 단말은 (Wait for MPDU) → (RTS Tx) → (Wait for MPDU) 상태를 계속 반복할 것이다. 그러나 채널의 상태가 좋지 않은 상황에서는 데이터 프레임이 RTS/CTS 보다 채널에러에 더욱 취약하기 때문에 성공적으로 전송될 수 없을 것이므로 위와 같은 loop를 반복하는 상황은 거의 발생하지 않을 것이다. 이때 무선 채널의 상태가 다시 좋아지게 된다면 RTS 프레임이 성공적으로 전송될 것이고 이후 데이터 프레임의 전송이 재개될 것이다.

또 하나의 임계값인 연속적 성공 횟수 임계값 ( $M_{th}$ )은 송신 단말이 전송속도를 증가시키기 위해 필요로 하는 데이터 프레임 전송의 연속적 성공 횟수를 의미한다. 이 논문에서는 언제 전송속도를 낮출 것인가에 초점을 두고 있으므로  $M_{th}$ 의 값은 ARF와 동일한  $M_{th} = 10$ 으로 설정하였다. 우리는 향후 이 논문에서 제안하고 있는 기법의 확장을 위해 [11], [12]에서 연구된 것과 같이 시간에 따라 변하는 채널 환경에서 어떻게  $M_{th}$ 를 적응적으로 적용할 것인가에 대해 연구할 계획이다.

2) 예제: 그림 4는 RTS Probing과 ARF의 전송 속도 조절과정을 비교하고 있다. 두 개의 802.11b

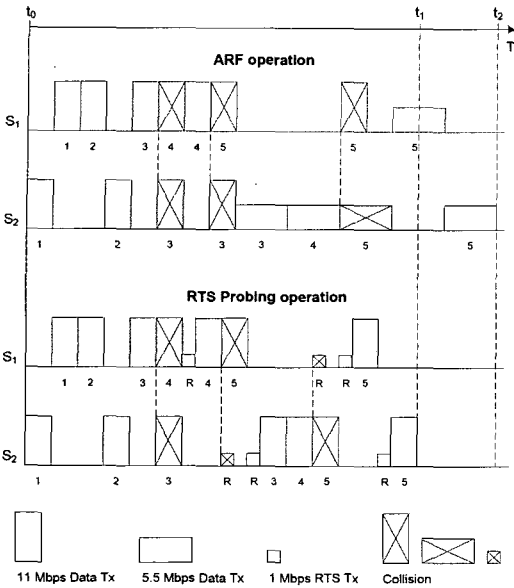


그림 4. 802.11b PHY의 최고 전송속도인 11 Mbps를 사용할 수 있는 채널환경에서 시간에 따른 ARF와 RTS Probing의 작동 과정

단말인 S<sub>1</sub>과 S<sub>2</sub>가 무선 채널을 이용하여 동일한 크기의 데이터 프레임들을 전송하기 위해 경쟁하고 있으며, 이때 채널의 상태는 가장 높은 전송속도인 11 Mbps를 사용할 수 있는 상태이다. 성공적인 전송은 그림에서 빈 사각형으로 표시되고 있으며, 대각선이 표시된 사각형은 프레임 충돌을 의미한다. 전송속도 조절과정을 단순화시켜 표현하기 위해 모든 프레임 전송간격(Inter-frame spaces), 백오프 시간, CTS/Ack 전송 과정은 생략하였다.

ARF를 사용하는 S<sub>2</sub>는 네 번의 전송 시도 후 전송속도를 5.5 Mbps로 낮추는 반면, RTS Probing을 사용하는 S<sub>2</sub>는 네 번의 데이터 전송 시도 이후에도 여전히 11 Mbps의 높은 전송속도를 유지하고 있다. 또한, RTS Probing에서는 다섯 개의 데이터 전송이 완료되는 시점이 ARF를 사용하였을 때 보다 빠르다는 것을 알 수 있다. 즉  $t_1 < t_2$ 인데 이는 RTS Probing을 사용하는 CARA를 적용할 경우 더 많은 데이터 프레임이 전송될 수 있으며, 이로 인해 채널을 더욱 효과적으로 사용할 수 있다는 것을 의미한다.

4.2 CCA Detection을 이용한 충돌 구분 방법  
 채널에러와 충돌을 구분하는 두 번째 방법은 RTS Probing의 보조 방식으로 사용되는 CCA Detection이다.

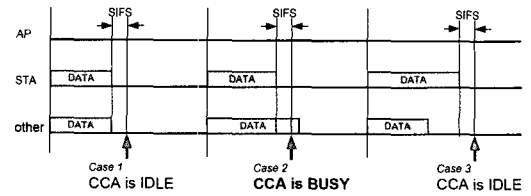


그림 5. 충돌발생의 세 가지 경우, 두 번째 경우 충돌은 CCA Detection으로 감지될 수 있다.

Hidden 단말이 없는 상황에서 발생 가능한 세 가지의 충돌예가 그림 5에 나타나있다. 그림에서 “STA”는 우리가 관심을 가지고 있는 단말이고 “other”는 STA와 동시에 프레임 전송을 시작하는 다른 단말을 의미한다. 사실상, 이 그림은 다수의 프레임 충돌이 나타나는 경우를 의미할 수도 있으며 그러한 경우에 “other”은 다수의 충돌을 일으키고 있는 프레임 중에서 가장 긴 전송시간을 가진 프레임을 전송하는 단말이 된다. 충돌은 그림 5에 나타나있는 것처럼 세 가지로 분류될 수 있다. 첫 번째는 충돌하는 프레임들이 동일한 전송시간을 가진 경우이고, 두 번째와 세 번째는 다른 전송시간을 가진 경우를 의미한다.

일반적으로 802.11에서의 CCA는 단말이 주어진 시간에 채널이 사용 중인가의 여부를 확인하기 위해 사용되는 기능이다. 예를 들어, CCA busy의 경우 단말은 자신의 백오프를 멈추게 되며, CCA idle 이면 단말은 자신의 백오프를 재시작하거나 또는 백오프 카운트가 0이 되었을 경우 바로 데이터 전송을 시작할 수 있다. 여기에서 제안하고 있는 CCA Detection 방법은 다음과 같이 동작한다. 무선 단말은 데이터 전송을 종료한 후 SIFS 시간 이후에 CCA를 이용하여 무선 채널이 사용 중인지 판단한다. 또한 이 시점은 데이터를 전송한 무선 단말이 Ack 프레임 수신을 기다리고 있는 상태이다. 그런데 예상하고 있던 Ack 프레임이 수신되지 않으면서 동시에 채널이 busy한 상태로 확인이 될 경우 단말은 그림 5의 Case 2에 있는 것처럼 전송된 데이터 프레임이 충돌을 일으켰다고 판단하게 된다. 그러나 Case 1과 Case 3에서는 CCA Detection은 적용될 수 없다.

1 Mbps에서 동작하고 있는 단말은 데이터 프레임 크기에 상관없이 Case 1 또는 Case 2 보다 Case 3을 더 많이 겪게 될 것이다. 왜냐하면 프레임 크기가 동일할 경우 낮은 전송속도로 동작하고 있는 단말의 전송시간이 높은 전송속도로 동작하고 있는 다른 단말들의 전송시간 보다 길기 때문이다.

따라서 CCA Detection은 송신 단말이 1 Mbps로 동작할 경우 쉽게 발생하지 않을 것이다. 그러나 1 Mbps는 가장 낮은 전송속도로서 더 이상 전송속도를 낮추는 것이 필요치 않으므로 이것이 CCA Detection의 장점이 될 수 있다.

### V. 성능 평가

이 장에서는 802.11 DCF를 사용하고 802.11b PHY와 시간에 따라 변하는 무선 채널 모델을 적용한 ns-2 simulator<sup>[23]</sup>을 이용하여 CARA의 성능을 분석한다.

#### 5.1 시뮬레이션 환경 설정

이 논문에서는 애드 혹(Ad Hoc) 모드가 실험된 한 가지를 제외하고 모두 인프라스트럭처(Infrastucture)의 802.11b 시스템을 이용하여 실험하였다. 모든 단말들은 20 dBm의 파워로 통신하고 있으며 한 위치에 고정되어 있다. FER(Frame Error Rate)을 추정하기 위해 Inersil<sup>[22]</sup>에서 제공된 BER(Bit Error Rate) 대 SNR(Signal-to-Noise Ratio) 곡선을 사용하였다. 실험환경의 배경 소음은 -96 dBm으로 설정하였다. 또한 실내 환경을 실험하기 위해 경로 손실지수(Path Loss Exponent)를 4로 설정한 경로 손실 모델을 사용하였다.

또한 송신 측과 수신 측 사이의 채널 상태가 시간에 따라 변하는 다중경로 감쇠 효과(Multi-path Fading Effect)를 고려하였다. 이를 위해 Ricean fading 모델<sup>[16]</sup>을 이용하여 시간에 따라 변하는 무선 채널 상태를 실험하였다.

$$p(r) = \frac{r}{a^2} e^{\left(-\frac{r}{2a^2} + K\right)} I_0(2Kr) \quad (1)$$

Ricean 분포는 식 (1)과 같다. 여기에서  $K$ 는 수신된 신호의 가시도를 나타내는 분포 변수,  $a^2$ 은 잡음의 분산,  $r$ 은 수신된 파워, 그리고  $I_0(\cdot)$ 는 Bessel 함수<sup>[17]</sup>를 의미한다. 채널 상태가 변하는 빈도는 주위 환경의 변화 속도 및 단말의 이동속도에 따라 다르다. 채널의 변화는 단말이 움직이지 않더라도 주위의 환경이 변하는 것에 의해서도 일어날 수 있다. 논문에서 사용한 주위 환경의 변화 속도는 2.5 m/s로 설정하였으며 이는 20 Hz의 Doppler spread와 동일한 값이다.

논문에서는 시뮬레이션을 통해 다음의 네 가지 방식을 비교하고 있다: (1) 고정된 전송속도  $x$

Mbps ( $x = 1, 2, 5.5, 11$ )를 사용하는 단일 전송속도 기법 (Rx); (2) ARF; (3) RTS/CTS를 항상 사용하는 ARF(RTS/CTS); 그리고 (4) 제안하고 있는 전송속도 조절기법, CARA-1(RTS Probing만 사용)과 CARA-2(RTS Probing과 CCA Detection을 모두 사용). 실험 기법들의 성능 비교를 위해 시스템 처리량의 합(aggregate system throughput, Mbps)을 사용하였다. VI장에서 설명된 것과 같이, ARF와 CARA의 연속적 성공 횟수 임계값( $M_{th}$ )은 10으로 설정하였고, 연속적 실패 횟수 임계값( $N_{th}$ )은 2로 설정하였다. CARA에서 채널상태 확인을 시작하는 임계값( $P_{th}$ )은 1로 설정하였다. RTS/CTS 프레임은 항상 가장 낮은 전송속도인 1 Mbps로 전송된다.

우리는 다양한 네트워크 토폴로지, 데이터 프레임 크기 및 채널 모델을 적용하여 실험하였다. 모든 단말들은 항상 전송할 데이터가 있으며(전송을 위한 큐가 항상 채워져 있음), 모든 데이터 프레임은 작은 크기로 나누어지지 않고 본래의 크기로 전송된다. 또한 LLC/IP/UDP를 상위 계층 프로토콜로 사용하였다. 매체접근제어 계층에서의 데이터 크기는 별도로 언급하지 않는 한 1500 바이트로 설정하였다.

#### 5.2 일대일 전송 토폴로지에서의 실험 결과

먼저 가장 단순한 일대일 전송 네트워크 환경에서 실험결과이다. 이 실험에서 송신 단말은 수신 단말과  $r(30 \leq r \leq 80)$  미터 떨어져 있으며 연속적으로 데이터를 전송한다. 무선 채널 모델은 AWGN (Additive White Gaussian Noise)을 고려하였다. 실험 결과는 그림 6과 같으며 원이 있는 굵은 실선이 CARA-1을 나타낸다. 일대일 전송 환경에서 CARA-1과 CARA-2의 성능은 거의 동일하기 때문에 여기

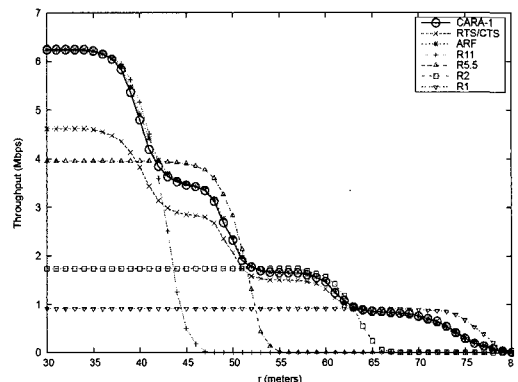


그림 6. 일대일 전송 토폴로지에서의 거리에 따른 성능 비교



에서 CARA-2의 성능은 실험하지 않았다. 즉, CARA-2에서 사용하는 CCA Detection은 프레임 충돌이 발생하지 않는 상황에서는 효과가 나타나지 않기 때문이다.

일반적으로 모든 실험 기법들에서 시스템의 성능은 거리가 멀어질수록 낮아진다. R1은 모든 실험 기법들 중에서 채널에러의 영향을 적게 받는 기법이다. 이 기법을 사용하는 단말은 모든 프레임을 가장 낮은 전송속도인 1 Mbps로 전송하며 따라서  $r$ 이 작을 때 가장 낮은 시스템 성능을 나타내고 있다. 또한 1 Mbps에 사용된 변조 및 채널 코딩의 강력한 에러 정정 특성으로 인해 송신 단말과 수신 단말이 서로 멀리 떨어져 있어도 통신이 성공할 수 있는 가능성이 커진다. 반면, R11은 IEEE 802.11b 표준의 전송속도 중에서 가장 높은 성능을 나타내는 방식으로 모든 데이터를 가장 높은 전송속도인 11 Mbps로 전송함으로써 송신 단말과 수신 단말의 거리가 가까울 때 가용한 자원(bandwidth)을 효율적으로 사용할 수 있다. 그러나 11 Mbps의 낮은 에러 정정 특성으로 인해  $r$ 이 증가할수록 성능은 급격하게 떨어진다. 그림에서 보는바와 같이 송신 단말과 수신 단말의 거리가 47 미터 이상으로 멀어지게 되면 모든 데이터 전송 시도는 실패하며 시스템 성능은 0이 된다. 다른 단일 전송속도 기법들은 R1과 R11의 중간 특성을 나타내고 있다.

일대일 전송과 같이 단말들 간의 경쟁이 없는 네트워크 환경에서는 단말 간의 거리에 따라 가장 좋은 성능을 나타내는 전송속도가 존재한다. 실험결과로는 CARA-1이 거리에 따라서 가장 좋은 성능을 나타내는 전송속도 기법들의 성능과 근접하게 동작하는 것을 나타내고 있다. 또한 CARA-1은 적응적인 RTS/CTS 프레임 전송으로 ARF보다는 약간 낮지만 거의 유사한 성능을 나타내고 있다. RTS/CTS 기법은 모든 데이터 프레임 전송 전에 RTS/CTS를 먼저 전송한다. 이로 인해 발생하는 오버헤드 때문에 CARA-1과 ARF보다 항상 낮은 성능을 나타내고 있다.

### 5.3 경쟁 단말의 수가 변하는 스타 토폴로지에서의 실험 결과

여기에서는 충돌 현상이 시스템 성능에 미치는 영향을 알아보기 위해 경쟁하는 단말의 수가 변하는 상황을 고려한 스타 토폴로지 네트워크를 실험하였다. 이 실험에서는 여러 개의 경쟁 단말이 AP로부터 반경 10 미터의 동심원에 균등하게 분포하

고 있으며 모든 단말은 고정되어 있다. 채널 모델은 앞 실험에서 적용한 AWGN을 고려하였다. AP로부터 각 단말이 10 미터 이격되어 있다는 것은 그림 6에서 보는 바와 같이 모든 단말이 항상 11 Mbps를 사용하여 데이터를 전송할 수 있다는 것을 의미한다.

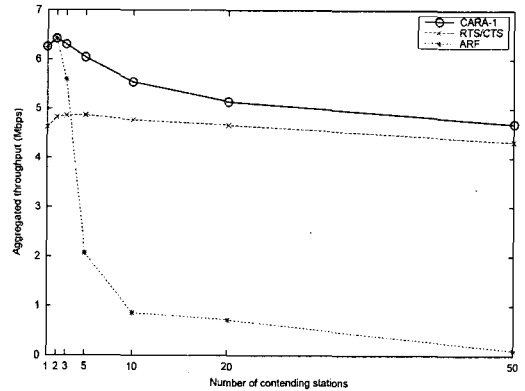


그림 7. 스타 토폴로지에서 경쟁 단말의 수에 따른 성능 비교

실험 결과는 그림 7에 나타나 있다. 그림에서 볼 수 있듯이 RTS/CTS와 ARF는 CARA-1보다 항상 낮은 성능을 나타내고 있다. 특히, ARF는 네트워크에 소수의 단말이 있는 경우에도 성능이 심각하게 낮아지는 것을 보여주고 있다. 예를 들어, 단말의 수가 2에서 5와 10으로 증가할 때 ARF의 성능은 6 Mbps로부터 2 Mbps와 1 Mbps 이하로 떨어지고 있다. 그러나 RTS/CTS의 성능은 ARF처럼 낮지 않다. 즉, 단말의 수가 증가하는 상황에서도 잘 동작하고 있다. 그렇지만 RTS/CTS 기법은 데이터 프레임을 전송하기 전에 항상 RTS/CTS를 전송하기 때문에 오버헤드로 인한 자원의 낭비가 크다. 이러한 이유로 RTS/CTS의 성능이 CARA-1보다 떨어지게 된다.

ARF의 성능이 좋지 않은 데에는 두 가지 이유가 있다. 첫째, ARF로 동작하는 단말은 충돌과 채널에러를 구분할 수 없기 때문에 채널 상태가 좋더라도 충돌에 의해 데이터 전송이 실패하면 과도하게 전송속도를 낮추려는 특성이 있다. 둘째, 모든 경쟁 단말들은 서로 독립적으로 전송속도를 조절한다. 이로 인해 모든 단말의 데이터 전송속도가 달라질 수 있다. 이러한 전송속도의 다양성을 performance anomaly라고 하는데 이것은 [14]의 저자들에 의해 실험적으로 발견되었다. 즉, 802.11 DCF는 모든 경쟁 단말들에게 동일한 전송 기회를 제공

하는 것으로 설계되었기 때문에 높은 전송속도를 사용하는 단말의 성능은 항상 네트워크에서 가장 낮은 전송속도를 사용할 때 보다 낮아지게 된다는 것이다.

결론적으로 이 실험에서 CARA-1은 ARF 보다 약 11.5배 정도 높은 성능을 나타내고 있다.

#### 5.4 데이터 크기와 단말의 위치가 임의로 선택 되는 라인 토폴로지에서의 실험 결과

이 실험에서는 단말의 수가 변하는 환경에서 CARA-1과 CARA-2를 ARF와 비교하였다. CARA-2는 앞에서 설명한 것과 같이 RTS Probing과 CCA Detection을 모두 사용한다. CCA Detection은 경쟁 단말들의 데이터 전송시간이 서로 다르며 네트워크에 hidden 단말이 존재하지 않을 경우에 효과가 있다. 이러한 실험환경을 구현하기 위해 여기에서는 각각의 경쟁 단말이 전송하는 데이터 프레임의 크기를 임의로 선택하도록 설정하였다. 또한 네트워크 토폴로지는 AP가 직선상의 한쪽 끝에 위치하고 있고 다른 단말들은 직선상의 임의의 지점에 위치하도록 구성된 라인 토폴로지를 고려하였다. 무선 단말과 AP와의 최대 거리는 hidden 단말의 문제를 해소하기 위해 70 미터로 하였다. 즉, 그림 6에 나와 있는 것과 같이 두 개의 서로 다른 단말은 최대 70 미터 이내로 이격되어 있어야만 통신이 가능하기 때문이다. 이 실험에서도 무선 채널 모델은 AWGN이 적용되었다.

실험 결과는 그림 8에 나타나 있다. 그림에서 각각의 점은 50회를 실험한 결과의 평균값을 나타낸다. 그림에 나타나있듯이 단말의 수가 증가하면 CARA-2와 CARA-1의 성능차이는 커지게 된다. 즉, 네트워크에 다수의 경쟁 단말들이 있는 경우

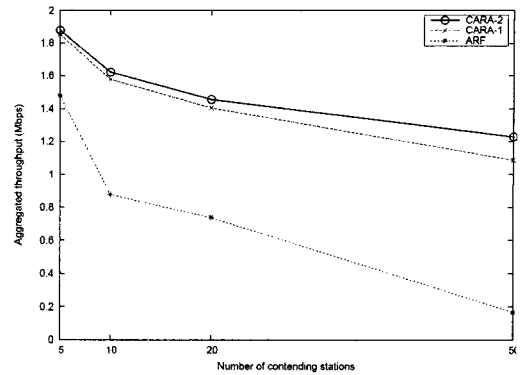
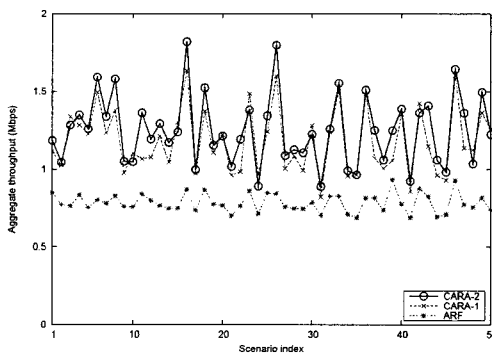


그림 8. 데이터 크기와 단말의 위치가 임의로 선택되는 라인 토폴로지에서의 경쟁 단말의 수에 따른 성능 비교

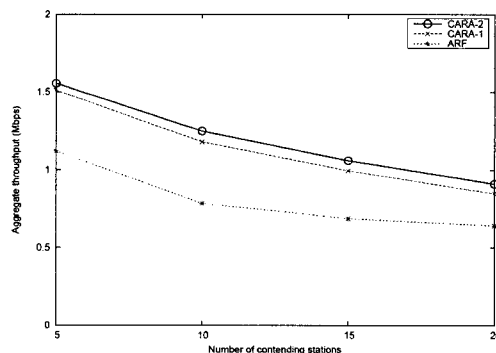
충돌하는 데이터 프레임들의 전송시간이 서로 다르기 때문에 CARA-2의 성능이 좋아지는 것이다. 결론적으로 이러한 환경에서는 CCA Detection이 충돌과 채널에러를 구분하는데 더욱 효과적이라는 것을 보여주고 있다. 반면 ARF의 성능은 CARA-1과 CARA-2보다 항상 낮게 나타나고 있다.

#### 5.5 랜덤 토폴로지에서의 실험 결과

여기에서는 임의로 형성된 네트워크 토폴로지, 즉 모든 송신 단말들과 그것들의 수신 단말들이 AP를 중심으로 하는 반경 40미터 동심원 내부의 임의의 지점에 위치하고 있는 네트워크 환경에서 실험하였다. 모든 단말들의 데이터 크기도 임의로 설정되었다. 이 실험에서 무선 채널 환경은 실내에서의 채널 감쇠 환경(indoor fading channel environment)<sup>[8]</sup>을 묘사하기 위해 Ricean K 변수를 3 dB로 하는 Ricean 감쇠 채널<sup>[16]</sup>을 고려하였다. 네트워크에 10개의 단말들이 통신하는 환경을 고려하여 총 50회의 실험을 진행하였으며 그 결과는 그림 9의



(a) 50 different scenarios when 10 stations contend



(b) Averaged results with various number of contending stations

그림 9. 랜덤 토폴로지 네트워크에서의 실험 결과

(a)에 나타나 있다. 또한 그림 9의 (b)는 경쟁 단말의 수가 변하는 상황에서의 실험결과가 나타나 있으며 각각의 실험결과는 임의로 선택된 50개의 토폴로지에서 실험한 결과를 평균한 값이다.

이 그림으로부터 두 가지 결과를 알 수 있다. 첫째, CARA-1과 CARA-2는 ARF 보다 월등히 우수한 시스템 성능을 나타내고 있으며, 경쟁 단말의 수에 상관없이 CARA-2는 CARA-1보다 평균적으로 우수한 성능을 나타내었다. 둘째, AWGN 채널 모델을 사용한 V.4절의 결과와 다르게 Ricean 채널 모델을 사용하였을 때에는 CARA-2가 CARA-1보다 항상 좋은 성능을 나타내는 것은 아니라는 것이다. 그림 9의 (a)에서 보듯이 총 50번의 실험 중 CARA-1이 CARA-2보다 더 좋은 성능을 나타낸 경우가 총 8회 있다. 이러한 이유는 다음과 같이 설명될 수 있다. CARA-2에 의해 성공적으로 CCA detection이 수행된 후 갑자기 채널의 상태가 나빠졌을 경우 송신 단말은 데이터 프레임의 재전송하는데 또다시 실패할 것이다. 그러나 이 경우 연속적인 실패 횟수를 나타내는  $n$ 은 성공적인 CCA detection으로 인해 증가되지 않을 것이며 결과적으로 전송속도를 낮추는 과정이 지연될 것이다. 이로 인

해 더 많은 데이터 전송실패가 발생하게 될 것이며 결국 시스템 성능이 낮아지는 결과를 초래한다.

### 5.6 시간에 따른 전송속도 조절 과정

여기에서는 시간에 따른 세 가지 실험 기법들의 전송속도 조절 과정을 비교하였다. 실험환경은 V.3 절과 유사하다. 즉, 네트워크는 스타 토폴로지이며 5개의 단말들이 AP로부터 반경 40 미터의 동심원 상에 균등하게 분포하고 있고 AWGN 채널 모델을 고려하였다. 그림 6에 나타나 있는 것처럼 각각의 전송 단말들은 이 거리에서 11 Mbps 또는 5.5 Mbps를 이용하여 전송할 것이다.

1.6초 동안 5개의 단말 중 1개 단말의 전송속도 조절 과정이 ARF, CARA-1과 CARA-2의 순으로 각각 그림 10(a), 10(b), 그리고 10(c)에 나타나 있다. 가로축은 RTS 전송이 포함되지 않은 데이터 프레임의 전송 시도 횟수를 나타낸다. 총 1.6초 동안의 실험시간 동안 모든 실험기법들의 데이터 프레임 전송 시도 횟수는 그림에서와 같이 모두 다르다. 이것은 적용된 실험기법 별로 서로 다른 전송 시도를 한다는 것을 의미하며, 이는 각각의 실험기법에서 사용한 전송속도가 모두 다르다는 것에 기

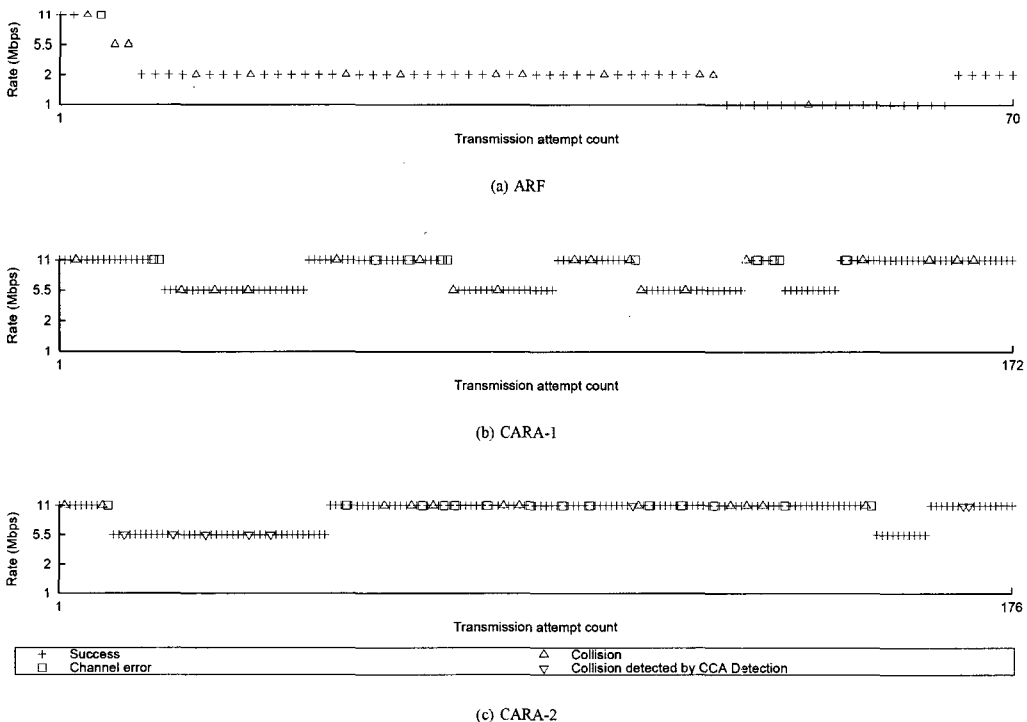


그림 10. ARF와 제안하고 있는 기법(CARA-1과 CARA-2)의 전송속도 조절 과정 비교

인한다. 이 그림에서 사용된 기호는 다음과 같다.

- 열십자: 성공적인 데이터 프레임 전송
- 사각형: 채널에러에 의해 발생한 데이터 전송 실패
- 삼각형: 데이터 프레임 충돌
- 역삼각형: CCA Detection에 의해 감지된 데이터 프레임 충돌

그림에서는 생략되었지만 RTS Probing을 사용하는 CARA-1과 CARA-2의 결과 중 모든 삼각형 기호, 즉 데이터 프레임 충돌에 의한 전송실패 다음에는 적어도 한번의 RTS 전송 시도가 있다.

그림으로부터 세 가지의 결과를 알 수 있다. 첫째, ARF는 충돌과 채널에러를 구분하지 못하며 따라서 과도하게 전송속도를 낮춘다. 예를 들어, ARF는 두 번의 연속적인 데이터 전송실패가 발생하면 전송속도를 5.5 Mbps에서 2 Mbps로, 2 Mbps에서 1 Mbps로 낮춘다. 즉, ARF는 그림에서처럼 대부분의 시간동안 낮은 전송속도인 1 또는 2 Mbps에서 동작한다. 둘째, CARA-1과 CARA-2는 프레임 충돌이 발생할지라도 무선 채널 상태의 변화에 적응적으로 동작한다. CARA-1과 CARA-2는 그림에서와 같이 모두 11 Mbps 또는 5.5 Mbps에서 동작한다. 마지막으로 CARA-2는 CARA-1보다 더 많은 데이터 전송 시도를 하고 있다. 이로 인해 CARA-2는 더 많은 데이터 전송을 성공할 수 있으며 결과적으로 시스템 성능이 가장 우수한 전송속도 조절 기법이 된다. 예를 들어, 그림 10에서의 총 실험 시간은 30초이며(여기에서는 일부본인 1.6초 구간만 표시하였다) 이 시간동안의 총 프레임 전송 시도 횟수, 성공적인 데이터 전송 횟수, 그리고 시스템 성능을 비교한 것이 표 2에 나와 있다. 다시 말해서 CARA-2는 CCA Detection으로 인해 CARA-1보다 더 좋은 성능을 나타낼 수 있는 것이다.

표 2. 30초 간 실험한 세 가지 기법의 결과 비교

	ARF	CARA-1	CARA-2
# of tx attempts	1344	3092	3246
# of tx successes	1094	2518	2643
Throughput (Mbps)	1.58	3.37	3.49

### 5.7 요약

여기에서는 앞에서 설명한 실험 결과를 근거로 논문에서 제안하고 있는 CARA의 특징을 요약하였다.

RTS Probing은 충돌과 채널에러를 구분하는데 매우 효과적이다. 이의 근거로 CARA의 성능이 ARF의 성능보다 우수하게 나타난 실험 결과를 들 수 있다.

또한, CCA Detection을 사용하는 CARA-2는 경쟁 단말들의 데이터 전송 시간이 다를 경우 CARA-1보다 높은 시스템 성능을 나타낸다.

현재 대부분의 상용 802.11 무선랜 장치에 적용되고 있는 ARF는 네트워크에 다수의 사용자가 통신하고 있을 경우 시스템 성능이 심각하게 낮아진다. 따라서, 802.11에서 지원되는 다양한 전송속도를 효율적으로 사용하면서 시스템 성능을 높이기 위해서는, CARA와 같이 데이터 충돌과 채널에러를 구분할 수 있도록 설계된 전송속도 조절기법을 적용하는 것이 중요하다는 결론을 내린다.

## VI. 결론 및 향후 연구계획

이 논문에서는 802.11 무선랜을 위한 새로운 전송속도 조절기법인 CARA가 제안되었다. CARA는 기본적으로 상용 802.11 무선랜 장치에 적용되는 전송속도 조절기법 중에서 가장 많이 알려져 있고, 또 널리 구현되어 있는 ARF의 문제점인 다수의 단말이 통신하고 있는 네트워크에서의 성능저하 문제를 해결할 수 있다. CARA의 핵심은 전송 단말이 채널에러에 의해 발생하는 데이터 전송실패로부터 데이터 충돌을 구분해 내기 위해 RTS/CTS 전송과 CCA 기능을 적응적으로 사용하는 것이다. 그러므로 ARF와 비교할 때, CARA는 채널상태에 따라 보다 정확하게 전송속도를 조절 할 수 있다. 또한, CARA는 802.11 표준을 그대로 적용하므로 802.11 무선랜 장치에 쉽게 구현될 수 있다.

CARA의 성능은 네트워크 토폴로지, 데이터 프레임 크기, 그리고 무선 채널 모델 등 다양한 환경에서의 실험을 통해 평가되었다. CARA는 다수의 전송 단말이 존재하는 환경에서 ARF보다 월등히 우수한 성능을 나타내는 것이 실험결과로서 나타났으며 이러한 성능차이는 단말의 수가 증가할수록 더욱 확연히 드러난다.

향후 우리는 CARA의 성능을 더욱 높이기 위해 다음과 같은 연구를 진행할 계획이다.

- 1) 이 논문에서 제안된 CARA는 언제 전송속도를 낮출 것인가에 초점을 두고 있다. 그러나 연속적인 성공 횟수 임계값( $M_{th}$ )과 관련된 '언제 전송속

도를 높일 것인가'에 대한 문제도 매우 중요하다. 이러한 문제는 시간에 따라 변하는 채널 환경에서 전송속도 조절 알고리즘의 성능에 아주 결정적인 요소가 된다. 따라서 우리는 최근에 제안되었던, 적응적으로 전송속도를 높이는 방법을 고려한 변형된 ARF 기법<sup>[21]</sup>을 CARA와 결합하여 전송속도 조절 기법을 보다 완벽하게 구현할 계획이다.

2) 또한 채널상태를 확인하는 임계값( $P_{th}$ )과 연속적인 실패 횟수 임계값( $N_{th}$ ) 등과 같은 시스템 매개변수를 최적화할 계획이다. 이 논문에서는 각각 1과 2의 기본값을 사용하였지만 이러한 매개변수를 최적화한다면 시스템 성능은 보다 좋아질 것이다. 이와 관련된 연구는 더 많은 전송속도를 지원하고 있는 802.11a와 802.11g와 같은 고속의 PHY에서 진행할 계획이다.

3) 마지막으로, 제안된 기법의 성능을 입증하기 위해 실제 무선랜 장치에 CARA를 구현할 계획이다. 이것은 공개 소스인 MADWIFI<sup>[24]</sup>를 이용하여 매체접근제어 계층의 일부를 변형시킴으로써 구현 가능할 것이라고 예상된다.

### 부 록

이 논문에서 사용된 약어 및 두문자어는 표 3에 알파벳순으로 요약되어 있다.

표 3. 약어 및 두문자어 목록

Ack	Acknowledgement
AP	Access Point
ARF	Automatic Rate Fallback
BER	Bit Error Rate
CARA	Collision-Aware Rate Adaptation
CCA	Clear Channel Assessment
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
CTS	Clear-to-Send
DCF	Distributed Coordination Function
DIFS	Distributed Inter-Frame Space
FER	Frame Error Rate
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LLC	Logical Link Control
MAC	Medium Access Control
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
NAV	Network Allocation Vector
OAR	Opportunistic Auto Rate
OFDM	Orthogonal Frequency Division Multiplexing
PCF	Point Coordination Function
PIFS	PCF Inter-Frame Space
PHY	Physical Layer
RBAR	Receiver-Based Auto Rate
RTS	Request-to-Send
SIFS	Short Inter-Frame Space
SINR	Signal-to-Interference/Noise Ratio
SNR	Signal-to-Noise Ratio
TXOP	Transmission Opportunity
UDP	User Datagram Protocol
WLAN	Wireless Local-Area Network

### 참 고 문 헌

- [1] IEEE 802.11, Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer (PHY) specifications, IEEE Std 802.11-1999, Aug. 1999.
- [2] IEEE 802.11b, Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications: Higher-speed Physical Layer Extension in the 2.4 GHz Band, Supplement to IEEE 802.11 Standard, Sep. 1999.
- [3] IEEE 802.11a, Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications: High-speed Physical Layer in the 5 GHz Band, Supplement to IEEE 802.11 Standard, Sep. 1999.
- [4] IEEE 802.11g, Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications: Further Higher Data Rate Extension in the 2.4 GHz Band, Supplement to IEEE 802.11 Standard, Jun. 2003.
- [5] IEEE 802.11e/D13.0, Part 11: Wireless LAN Medium Access Control(MAC) and Physical layer(PHY) specifications: Medium Access Control(MAC) Enhancements for Quality of Service(QoS), Draft Supplement to IEEE 802.11 Standard, Draft 13.0, Jan. 2005.
- [6] IEEE 802.11k/D2.0, Part 11: Wireless LAN Medium Access Control(MAC) and Physical layer(PHY) specifications: Radio Resource Measurement, Draft Supplement to IEEE 802.11 Standard, Draft 2.0, Feb. 2005.
- [7] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Area in Communications*, vol. 18, no. 3, March 2000.
- [8] M. Carroll, T. A. Wysocki, "Fading Characteristics for Indoor Wireless Channels at 5GHz Unlicensed Bands," in Proc. *Sympo TIC'03*, pp 102-105, Bratislava, Slovakia, Oct. 2003.
- [9] Sunghyun Choi and Javier del Prado, "802.11g CP: A Solution for IEEE 802.11g

- and 802.11b Inter-Working,” in Proc. *IEEE VTC'03-Spring*, Jeju, Korea, April 2003.
- [10] Sunwoong Choi, Kihong Park, and Chongkwon Kim, “On the Performance Characteristics of WLANs: Revisited,” in Proc. *ACM SIGMETRICS '05*, Jun. 2005.
- [11] P. Chevillat, J. Jelitto, A. Noll Barreto, and H. L. Truong, “A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs,” in Proc. *IEEE ICC'03*, Anchorage, AK, pp. 1141-1145, May 2003.
- [12] Javier del Prado Pavon and Sunghyun Choi, “Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement,” in Proc. *IEEE ICC '03*, Anchorage, AK, May 2003.
- [13] Gavin Holland, Nitin Vaidya, and Paramvir Bahl, “A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks,” in Proc. *ACM MobiCom '01*, Rome, Italy, Jul. 2001.
- [14] M. Heusse, F. Rousseu, G. Berger-Sabbatel, and A. Duda, “Performance Anomaly of 802.11b,” in Proc. *IEEE INFOCOM '03*, vol. 2, pp. 836-843, Mar. 2003.
- [15] Ad Kamerman and Leo Monteban, “WaveLAN-II: a high-performance Wireless LAN for the Unlicensed Band,” *Bell Labs Technical Journal*, vol.2, no.3, pp.118-133, Aug. 1997.
- [16] Ratish J. Punnoose, Pavel V. Nikitin, and Daniel D. Stancil, “Efficient Simulation of Ricean Fading within a Packet Simulator,” in Proc. *IEEE VTC '00-Fall*, Sept. 2000.
- [17] T. S. Rappaport, *Wireless Communications: Principle and Practice*, Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [18] B. Sadeghi, V. Kanodia, A. Sabharwal and E. Knightly, “Opportunistic Media Access for Multirate Ad Hoc Networks,” in Proc. *ACM Mobi-Com '02*, Atlanta, Georgia, Jul. 2002.
- [19] Daji Qiao and Sunghyun Choi, “Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation,” in Proc. *IEEE ICC '01*, Jun. 2001, pp. 1995-2000.
- [20] Daji Qiao, Sunghyun Choi, and Kang G. Shin, “Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs,” *IEEE Trans. On Mobile Computing(TMC)*, vol. 1, no. 4, pp. 278-292, Oct.-Dec. 2002.
- [21] Daji Qiao and Sunghyun Choi, “Fast-Responsive Link Adaptation for IEEE 802.11 WLANs,” in Proc. *IEEE ICC '05*, Seoul, Korea, May 2005.
- [22] Intersil, “HFA3861B; Direct Sequence Spread Spectrum Baseband Processor,” January 2000.
- [23] “The Network Simulator-ns-2,” <http://www.isi.edu/nsnam/ns/>, online link.
- [24] MADWIFI, <http://sourceforge.net/projects/madwifi/>, online link. Wireless LAN via Link Adaptation,” in Proc. *IEEE ICC'01*, Jun. 2001, pp. 1995-2000.

김 종 석 (Jongseok Kim)

준회원



1996년 2월 육군사관학교 전자계산학과 졸업  
2004년 3월~현재 서울대학교 전기공학부 석사과정  
<관심분야> 무선통신망

김 성 관 (Seongkwan Kim)

정회원



2000년 2월 고려대학교 전자공학(학사)  
2002년 2월 고려대학교 전자컴퓨터 공학과(석사)  
2004년 3월~현재 서울대학교 전기공학부 박사과정  
<관심분야> 무선통신망

최 성 현 (Sunghyun Choi)

종신회원



1992년 2월 한국과학기술원 전기전자공학(학사)  
1994년 2월 한국과학기술원 전기전자공학(석사)  
1999년 9월 미시간대학교 전기컴퓨터공학과(박사)  
1999년 9월~2002년 8월 미국

필립스 연구소 선임연구원

2002년 9월~현재 서울대학교 전기공학부 조교수  
<관심분야> 무선통신망, 이동통신망