

---

# Ad Hoc 네트워크에서 Cooperative MAC 프로토콜에 관한 연구

장재신\*

A study on a cooperative MAC protocol at Ad Hoc networks

Jaeshin Jang\*

---

이 논문은 2008년도 인제연구장학재단 교수연구년 지원에 의한 연구결과임

---

## 요약

Ad Hoc 네트워크는 이동통신망이나 AP (access point)와 통신하는 무선 랜과는 달리 유사한 특징을 가진 모든 단말들이 무선 채널을 통해 직접 통신하는 좀 더 진화된 통신망이라고 할 수 있다. Ad Hoc 네트워크에서 시스템 성능을 높이려는 다양한 연구가 진행되고 있는데 그 중의 하나가 Cooperative 통신이라고 할 수 있다. 본 연구에서는 Cooperative 통신을 실현할 수 있는 CO-MAC 프로토콜을 제안하고, 수학식을 사용하여 성능평가를 수행하였다. 도출된 성능평가 결과를 통해 IEEE 802.11 표준인 DCF 기법, 그리고 기존 연구에서 제안된 rDCF 기법과 성능비교를 통해 프레임 전송오류 확률이 증가할 때 시스템 처리량이 많이 개선됨을 확인할 수 있었다.

## ABSTRACT

Being different from traditional mobile communication networks or wireless LAN with access points, ad hoc network can provide direct communications between homogeneous stations. Thus, it can be considered as a much advanced network. Somehow, many researches have been done to enhance system performance of ad hoc networks up to now, one of which is cooperative communications. In this paper, a cooperative MAC protocol, called CO-MAC protocol, is proposed and its performance is evaluated with a mathematical approach. Numerical results show that this scheme provides more enhanced system throughput especially when frame transmission error probability is high than DCF scheme, which is the key protocol in IEEE 802.11 standard, and rDCF scheme.

## 키워드

Ad Hoc 네트워크, Cooperative 통신, MAC 프로토콜, DCF, 도움노드

---

\* 인제대학교 정보통신공학과

접수일자 2009. 02. 11

심사완료일자 2009. 03. 13

## I. 서 론

Ad Hoc 네트워크는 이동통신망이나 AP (access point)와 통신하는 무선 랜과는 달리 유사한 특징을 가진 모든 단말들이 무선 채널을 통해 직접 통신하는 좀 더 진화된 통신망이라고 할 수 있다[1]. 그러나 Ad Hoc 네트워크를 포함한 모든 무선 통신망은 유선 통신망과 비교할 때 대역폭 제한이라는 치명적인 한계점을 안고 있다. 그래서 무선 통신망에서 해결해야 하는 중요한 과제 중의 하나는 시스템 성능을 증가시키는 것이며, 이를 달성하기 위해 데이터 링크 계층의 프레임 전송속도를 증가시키는 것이 중요하다. 프레임 전송속도를 높이기 위해서는 수신측에서의 SNR 값을 향상시키거나 수신 성능을 향상시켜야 한다. 현재 물리계층 및 데이터 링크 계층으로 널리 사용하고 있는 표준이 IEEE 802.11 무선 랜이다. 이것은 CSMA/CA 프로토콜에 기반을 둔 DCF 기법이 hidden node 문제나 exposed node 문제를 원만하게 해결하면서 단일 흡에서 매우 우수한 성능을 제공하고 있기 때문이다.

WiBro 시스템, 무선 랜 시스템 등을 포함하여 최근 무선통신망에서는 시스템의 성능을 향상시키기 위하여 링크 어댑테이션 (Link adaptation) 기능을 사용한다. 이 기능을 실현하기 위해 물리계층에서는 다양한 변복조/채널코딩 기법을 제공하며, 채널상태에 따라서 적절한 채널 변복조/채널코딩 기법을 선택하여 사용하도록 하고 있다. 예를 들면, 채널상태가 양호할 때는 64QAM 변복조 기법을 사용하여 시스템 성능을 향상시키고, 채널상태가 불량할 때는 QPSK 변복조 기법을 사용하여 프레임 전송오류에 잘 견디도록 한다. 간단한 계산으로도 64QAM 변복조 기법을 사용하여 전송하면 QPSK 변복조 기법을 사용하여 전송하는 것보다 동일한 대역을 사용하여 전송할 경우보다 3배 더 많은 정보를 보낼 수 있다.

한편 무선통신망에서 링크 어댑테이션 기법에 기반을 두고 시스템 성능을 더욱 향상시키기 위해 최근에 Cooperation 통신 기법을 연구하고 있다. Cooperative 통신을 설명하기 위해 그림 1과 같이 송신노드( $N_s$ ), 목적지노드( $N_d$ ), 도움노드(Helper node,  $N_r$ )로 이루어진 간

단한 무선통신망을 고려하자.  $N_s$ 에서  $N_d$ 로 직접 데이터를 전송할 경우에는 열악한 채널상태 때문에 2 Mbps로 전송할 수 있다. 한편, 송신노드와 목적지노드 중간에 위치한 노드가 도움노드 역할을 수행하며, 송신노드와 도움노드, 그리고 도움노드와 목적지노드 사이는 11 Mbps로 전송할 수 있다고 하자.

도움노드를 통해서 송신노드와 목적지노드가 통신을 한다고 하면, 평균전송속도는  $\frac{11 \cdot 11}{11 + 11} = 5.5 \text{ Mbps}$ 가 되며, 이 수치는 송신노드와 목적지노드가 직접 통신할 때의 2 Mbps보다 더 큼을 알 수 있다. 즉, 채널상태가 우수한 도움노드의 도움으로 채널 전송속도를 향상시키고, 패킷 전송시간을 감소시킴으로써 시스템 처리량을 향상시키는 접근방법을 Cooperative 통신이라고 한다.

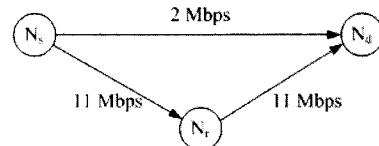


그림 1. Cooperative 통신 예  
Fig. 1 Example of cooperative communications

본 논문에서는 이러한 Cooperative 통신의 연구 추세에 발맞춰 무선통신망의 성능을 향상시키기 위해, 데이터 링크계층의 Cooperative MAC 프로토콜을 제안하고, 수학적 접근방법을 사용하여 제안된 프로토콜의 성능을 평가하고자 한다. 본 논문의 II장에서는 Cooperative MAC 프로토콜 관련 기존 연구동향을 기술하고, 본 논문에서 제안한 CO-MAC 프로토콜 기법을 III장에서 설명한 뒤, IV장에서 수학적 접근방법으로 제안된 기법을 분석하고, 수치계산 결과를 V장에서 제시한 뒤 VI장에서 끝맺는다.

## II. 관련 연구동향

Ad Hoc 네트워크에서의 Cooperative 통신용 MAC 프로토콜에 대해 최근의 연구들은 대부분 현재 널리 사용

되고 있는 IEEE 802.11 무선 랜 프로토콜에 근거하고 있다[2]. 또한 모바일 WiMAX로 알려진 IEEE 802.16e와 최대 54 Mbps 속도를 제공하는 IEEE 802.11a 또는 IEEE 802.11g 규격에서는 채널의 품질에 따라 상이한 변복조 기법과 채널코딩 기법을 사용하여 시스템 성능을 향상시키는 링크 어댑테이션 기법을 사용하고 있다. 이와 관련하여 참고문헌[3]에서는 수신노드에서 채널 품질을 측정한 뒤, 송신노드에게 전송하면 송신노드가 이 값을 기반으로 적절한 전송속도를 사용하여 데이터를 전송할 수 있도록 하는 RBAR (receiver-based auto rate) 기법을 제안하였다.

한편 참고문헌[4]-[10]들은 현재 상용 중인 무선 랜 프로토콜에 기반을 두고, Cooperative 통신을 지원하기 위한 MAC 프로토콜에 대한 연구결과들을 제시하였다. 먼저 참고문헌[4]에서는 IEEE 802.11 무선 랜의 PCF 기법을 사용하여 Cooperative 통신을 수행하는 rPCF 기법을 제안하였다. 이 기법에서는 모든 단말이 인접한 노드에서 전송한 모든 프레임을 검사하여 인접 노드들과의 채널상태 정보(예를 들면 수신 SNR 값 등)를 기록한 뒤, AP에게 통보한다. 각 단말에서 전송한 인접 노드들 간의 채널상태 정보를 수신한 AP는 Poll 프레임을 사용하여 각 단말들에게 전송기회를 부여하며, 송신 노드와 릴레이 노드, 릴레이 노드와 목적지 노드 사이의 전송속도 정보를 Poll 프레임에 같이 실어서 단말에게 전송한다. 하지만 rPCF 기법이 갖고 있는 문제는 PCF 기법이 무선 랜 시장에서 현재 거의 사용되지 않고 있으며, AP가 관리하고 있는 모든 노드들 간의 채널 품질 정보를 관리해야 하므로 너무 복잡하다는 단점을 갖고 있다. 또한, 기존 PCF 기법에서는 노드들 간에 직접 통신할 수 없는데 rPCF에서는 노드들이 직접 통신할 수 있다고 가정하고 있어서 또 다른 문제점을 야기할 수 있다.

참고문헌[5]에서는 CMAC 프로토콜을 제안하였으며, 오류제어기법과 결합한 FCMAC 프로토콜도 제안하였다. 이 연구결과는 현재 IEEE 802.11 무선 랜 표준을 적게 변경하면서 Cooperative 통신을 지원하는 MAC 프로토콜을 제안했다는 점에서 눈에 띠지만 도움노드를 선정하는 방법에 대해서 전혀 언급하지 않았다. 참고문헌 [6],[7]은 Cooperative 통신을 지원하기 위한 새로운 ARQ 기법을 제안하였다.

ARQ 기법의 경우, 송신 노드에서 전송한 프레임을 수신 노드가 전송오류 때문에 제대로 수신하지 못하였을 경우, 수신 노드가 NACK 제어프레임을 전송하여 재전송을 요청한다. 참고문헌[6]에서 제안한 ARQ 기법에서는 중간에서 NACK 프레임을 가로챈 이웃노드가 송신노드 대신 오류가 발생한 데이터 프레임을 재전송하여 응답시간을 줄이는 것이 요점이다.

이 연구는 Cooperative 통신보다는 Cooperative 재전송에 관한 내용이며 매우 단순하지만, 수학식을 사용하여 성능평가를 깔끔하게 수행한 점이 주목할 만하다. 참고문헌[7]에서도 ARQ 기법을 사용하여 Cooperative 재전송하는 방법을 연구했는데, 재전송을 수행하는 주체와 방법에 따라 3가지 변형 ARQ기법을 제안하고 시뮬레이션을 통해 성능평가를 수행하였다. 참고문헌[8]에서는 DCF 기법을 기반으로 Cooperative 통신을 수행할 수 있는 rDCF 프로토콜을 제안하고 성능평가를 수행하였다.

이 연구에서는 각 노드들이 Cooperative 통신에 사용할 도움노드를 결정하는 방법에 대해서 언급한 것이 눈에 띈다. 먼저 모든 노드들은 다른 노드들이 전송한 모든 프레임들을 검사하여 송신노드와 도움노드 사이의 무선채널에 대한 수신 SNR 값(또는 이에 대응하는 전송속도 값)을 계산한다. 이 값을 willing 리스트에 저장한 뒤, 주기적으로 인접 노드들에게 willing 리스트를 전송하여 서로 공유한다.

인접 노드로부터 willing 리스트를 수신한 노드들은 relay 테이블에 해당 정보를 저장한다. 그 후, 송신노드로부터 목적지노드에게 전송할 패킷이 발생하면 송신노드에서 목적지노드로 직접 전송하는 것보다 이중 흡전송을 통해 더 효율적으로 전송할 수 있는 도움노드가 있는지 relay 테이블을 검색하여 확인한다.

만약 적절한 도움노드가 있을 경우에는 목적지노드 대신 도움노드에게 RRTS1 (Relay RTS1) 프레임을 전송하여 Cooperative 통신을 수행한다. 이 연구에서는 도움노드를 설정하는 방법 그리고 Cooperative 통신과 다중 전송속도를 구현하기 위해 기존 DCF에서 수정되어야 하는 부분들을 잘 언급하고 있지만, 변경된 제어프레임의 구조에 대한 설명이 없어서 성능평가에서 RRTS, RCTS 프레임이 전송해야 하는 추가정보들을 ( $N_s$ ,  $N_r$ ,  $N_d$  사이의 전송속도 정보 등) 모두 고려했

는지 확인할 수 없다. 참고문헌[9]는 다이버시티 이득을 얻기 위해 두 개의 도움노드를 사용하는 환경으로 rDCF 기법을 확장한 CODE라는 MAC 프로토콜을 제안하였다.

CODE 프로토콜은 도움노드를 설정하는 방법과 제어프레임을 송수신하는 과정은 rDCF와 매우 유사하다. 그러나 송신노드가 CRTS1 (Cooperative RTS1) 프레임을 전송한 후 두 개의 도움노드가 연속적으로 CRTS2 와 CRTS3를 각각 전송하는 것이 개선된 됨이다. 하지만 CODE 프로토콜이 rDCF를 확장한 것이므로 성능비교를 수행할 때 rDCF와 비교했어야 하는데 기존 RBAR[3] 기법과 성능비교를 수행한 것은 적절하지 못했다고 생각된다. 참고문헌[10]에서는 CoopMAC 프로토콜을 제안하였는데, 해당 프로토콜에서 도입한 제어프레임과 제어프레임을 전송하는 일부 절차는 다르지만, 개략적인 동작은 rDCF와 유사하다. 또한 도움노드를 선정하는 기준을 수식으로 표시한 점은 매우 주목할 만하다.

마지막으로 참고문헌[11]은 기존 DCF 프로토콜을 확률을 사용하여 수학적으로 성능분석을 수행하였다. 이 논문에서는 모든 단말이 항상 전송할 패킷을 가지고 있는 포화상태를 가정하였다. 그래서 이 논문에서 구한 시스템 처리량을 시스템 최대 처리량, 또는 포화상태에서의 시스템 처리량이라고도 한다. 참고문헌[8]-[10]에서 사용한 수학적인 성능평가는 모두 참고문헌[11]의 결과를 기반으로 수행되었다.

이제까지 Cooperative MAC 프로토콜 관련 연구동향을 분석한 결과 다음과 같은 두 가지 문제점을 발견하였다. 첫 째는 성능비교에서 대부분의 프로토콜들이 초기 기법인 RBAR 기법이나 IEEE 802.11 무선 랜과 성능비교를 수행하였고, 최근에 연구된 Cooperative MAC 프로토콜과는 성능분석 비교를 수행하지 않았다. 이것은 아마도 성능평가를 수행한 시스템 환경이 서로 상이하기 때문에 직접 비교할 수 없었던 것으로 판단된다. 두 번째는 수학적 접근방법이나 모의실험을 통해서 시스템 성능평가를 수행할 때 채널에서는 전송오류가 발생하지 않는 환경을 가정했다는 것이다.

### III. CO-MAC 프로토콜

본 논문에서 제안한 CO-MAC (Cooperative MAC) 프로토콜은 그림 2에 나타낸 바와 같이 크게 두 단계 전송으로 이루어져 있다. 1단계는 CRTS (Cooperative RTS) 와 ARTS (Ack RTS), 그리고 CCTS (Cooperate CTS) 제어프레임을 주고받는 과정이며, IEEE 802.11 무선 랜의 CSMA/CA 기법을 사용하여 CRTS를 전송하고, 이 단계에서 도움노드를 선정한다. 따라서 성능평가 시에는 사전에 송신노드가 사용하고자 하는 도움노드에 대한 정보를 모두 가지고 있다고 가정하였으며, 도움노드를 결정하는 방법은 참고문헌[8][9]와 동일한 방법을 사용한다. 즉, 모든 노드들은 willing 리스트를 관리하여 인접한 노드와의 채널상태 정보 또는 가능한 전송속도를 파악하고 있다.

그리고 모든 노드들은 주기적으로 willing 리스트를 인접노드들에게 전송하여 채널정보를 공유한다. 인접노드로부터 willing 리스트를 수신한 각 노드는 보유하고 있는 relay 테이블에 도움노드ID와 가능한 전송속도 정보 등을 저장한다.

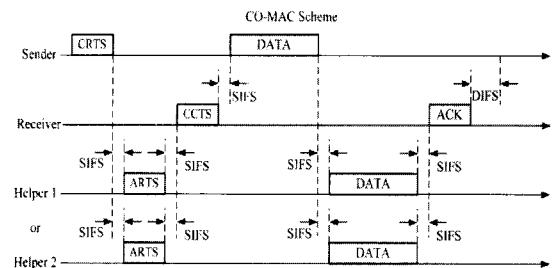


그림 2. CO-MAC 프로토콜의 프레임 전달과정  
Fig. 2 Frame exchange of CO-MAC protocol

2단계는 송신노드가 선정된 도움노드를 통해서 DATA 프레임을 전송하고 ACK 프레임을 목적지노드로부터 직접 수신하는 과정이다. CO-MAC 프로토콜은 기본적으로 DCF 기법에 기반을 두고 설계되었으며, 그림 3, 4, 5는 CO-MAC 프로토콜을 구현하기 위해서 송신노드, 목적지 노드 그리고 도움노드에서 수행하는 동작절차를 나타낸다.

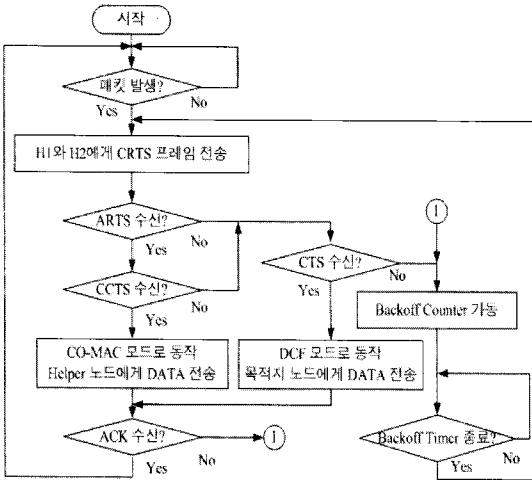


그림 3. 송신노드의 동작 절차

Fig. 3 Working procedure of sender nodes

먼저 송신노드에서 수행하는 절차를 그림 3에 나타내었다. 상위계층으로부터 패킷이 도착하면 송신노드는 무선채널로 목적지노드에게 데이터 프레임을 전송하기 위해 채널경쟁을 시작한다. 채널경쟁은 CO-MAC 프로토콜에서 새롭게 설계한 CRTS (Cooperative RTS) 프레임을 사용한다. 이 제어프레임은 rDCF의 RRTS와 유사하지만 프레임 헤더에 목적지노드, 도움노드 H1과 H2의 MAC 주소를 모두 포함시키는 것이 다르며, CRTS 프레임 헤더에 송신노드가 관리하고 있는 Relay 테이블을 참조하여 해당 도움노드와의 가능한 전송속도들도 같이 포함 시킨다 (예:  $R_{sh}$ ,  $R_{hd}$ ,  $R_{sd}$ ). 여기에서  $R_{sh}$ 는 송신노드(S)와 도움노드(H) 사이의 채널전송속도를 의미한다. 참고문헌[9]에서 제안한 CRTS1도 CO-MAC 프로토콜에서 제안한 CRTS와 유사한 헤더 구조를 가지고 있을 것으로 생각되나 참고문헌[9]에서 프레임구조에 대한 설명이 없어서 확인할 길은 없다.

한편 CRTS 프레임을 전송한 뒤, SIFS 시간 후에 도움노드가 전송한 ARTS 프레임을 수신하면 SIFS 이후에 CCTS 프레임을 기다린다. 만약 SIFS 이후에 CCTS 프레임을 수신하면 도움노드에게 DATA를 전송하고, SIFS 후에 ACK 프레임을 기다린다. ACK 프레임을 성공적으로 수신하면 전송이 완료되며, 만일 지정된 시간 내에 ACK 프레임을 수신하지 못하면 Backoff 카운터를 구동하여 CRTS 프레임 재전송을 시도한다.

이때, Backoff 카운터를 사용하는 절차는 기존 IEEE 802.11과 동일하며, DATA 프레임의 헤더에는 목적지 노드의 MAC 주소도 같이 포함되어 있어야 한다. 만일 지정된 시간 내에 CCTS 프레임을 수신하지 못하였거나 CCTS 프레임을 수신하였을 경우에는 기존 DCF로 모드로 동작하며, CCTS 프레임도 수신하지 못하였을 경우에는 앞에서 기술한 바와 같이 Backoff 카운터를 구동하여 재전송을 실시한다.

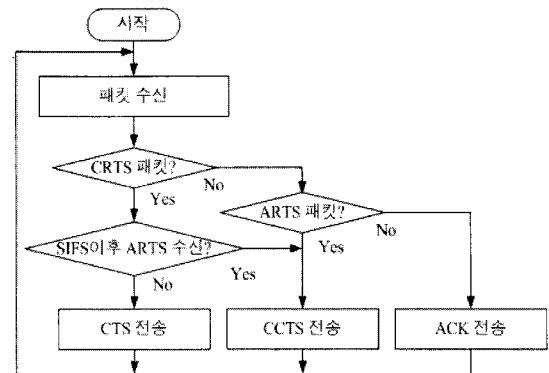


그림 4. 목적지노드의 동작 절차

Fig. 4 Working procedure of destination nodes

목적지노드에서는 (그림 4 참조) 송신노드로부터 CRTS 프레임을 수신하면 SIFS 이후에 ARTS 프레임을 기다린다. 지정된 시간 내에 ARTS 프레임을 수신할 경우에는 CCTS 프레임을 송신 노드에게 전송하며, 지정된 시간 내에 ARTS 프레임을 수신하지 못할 경우에는 CCTS 프레임을 전송한다. 도움노드로부터 DATA 프레임을 수신하였을 경우에는 ACK 프레임을 송신 노드에게 전송하여 해당 DATA 프레임을 잘 수신하였음을 알린다. 한편 도움노드가 전송한 ARTS 프레임 내에 포함되어 있는 채널정보와 도움노드와 목적지노드 사이의 무선채널에서 측정한 수신 SNR 값을 기반으로 전송 가능한 전송속도를 계산하여 이 정보들을 CCTS 프레임에 포함시킨다.

마지막으로 Cooperative 통신을 수행하기 위해 도움노드에서 수행하는 동작을 그림 5에 나타내었다. CRTS 프레임 헤더에 있는 두 개의 도움노드 ID 중에서 앞부분에 위치한 ID의 도움노드가 높은 우선순위를 갖는다. 여기에서 설명의 편의를 위해 도움노드 H1의 우선순위가

높다고 가정하자. H1노드는 CRTS 프레임을 수신하면 Cooperative 통신에 참여할 수 있는지를 체크하고, 참여 할 수 있다고 판단되면 SIFS 시간 이후 목적지 노드에게 ARTS 프레임을 전송한다. Cooperative 통신에 참여할 수 있는지 여부는 송신노드에서 전송한 CRTS 프레임의 수신 SNR 값을 기반으로 CRTS 프레임에 포함되어 있는 전송속도  $R_{sh1}$ 을 제공할 수 있는지 판단하는 것이다. 만일 참여할 수 없다고 판단되면 Idle 상태를 유지한다. 한편 송신노드로부터 CRTS 프레임을 수신한 도움노드 H2는 H1이 Cooperative 통신에 참여하는지를 먼저 체크 한다. 체크하는 방법은 CRTS 프레임을 수신한 후 SIFS 시간 후에 H1이 ARTS 프레임을 전송하는지 여부를 확인하는 것이다. 만일 SIFS 시간 이후에 아무런 프레임이 전송되지 않으면 H2 노드가 ARTS 프레임을 전송하면서 Cooperative 통신에 참여한다. 물론 H2 노드도 CRTS 프레임에 기록된  $R_{sh2}$  값 이상으로 데이터를 전송할 수 있어야 한다. 한편 송신 노드로부터 DATA 프레임을 수신하였을 경우에는 목적지노드에게 DATA 프레임을 전송한다.

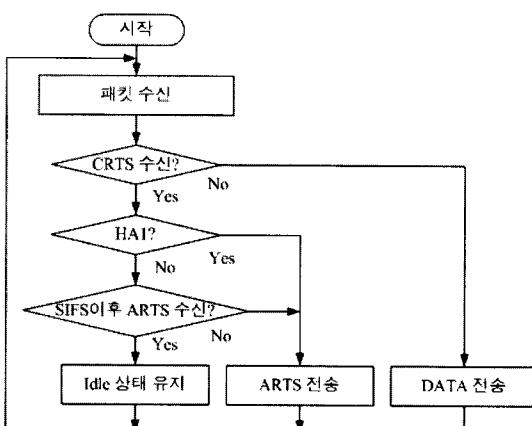


그림 5. 도움노드의 동작 절차  
Fig. 5 Working procedure of helper nodes

#### IV. 성능평가

성능평가는 참고문헌[11]에서 Bianchi가 도입한 확률 기반의 수학적 기법을 그대로 사용하였다. 이 기법은 포화상태에서 시스템의 최대 성능을 구하기 위해 사용된

해석적 기법이며, 포화상태란 모든 단말의 전송 베피에 전송할 패킷이 항상 존재한다고 가정한 상태를 의미한다. 이 기법은 Cooperative 통신을 지원하는 MAC 프로토콜을 연구하고 성능 평가했던 최근 연구들의 대부분에서 채용한 기법이다. 그리고 본 논문에서는 송신노드가 채널경쟁을 수행하기 위해 전송하는 CRTS 프레임에서만 전송오류가 발생한다고 가정하였다.

성능평가에 앞서 다음과 같은 몇 가지 변수를 정의 한다.

- $p_c$  : CRTS 프레임 전송 시 프레임 충돌 확률
- $p_e$  : CRTS 프레임 전송 시 프레임 전송 오류 확률
- $\tau$  : 한 개의 노드에서 CRTS 프레임 전송 확률
- $p_t (= p_e^2 + p_c)$  : CRTS 프레임 전송 실패 확률

CRTS 프레임 전송 실패는 CRTS 프레임 충돌에 의해 발생하거나 프레임 전송오류에 의해서 발생할 수도 있으며, 본 논문에서는 두 사건이 독립이라고 가정하였다. 그리고 프레임이 충돌하면서 SNR 값이 나빠서 프레임 전송오류가 동시에 발생할 확률은 매우 작다고 판단되어 무시하였다. 그러므로 rDCF, CODE, 또는 CoopMAC 프로토콜에서는 CRTS 프레임 전송 실패 확률을  $p_t' (= p_e + p_c)$ 로 표시할 수 있지만, CO-MAC 프로토콜에서는 CRTS 프레임 전송오류가 H1 노드와 H2 노드에서 모두 전송오류가 발생하는 경우에만 발생하며, 송신노드와 H1 및 H2 노드로의 전송링크가 상호 독립이라고 가정하면  $p_t (= p_e^2 + p_c)$ 로 나타낼 수 있다. 그리고 성능평가의 편의를 위해, 두 도움노드에서 겪는 통신환경은 모두 동일하다고 가정하였다. 그러면 참고문헌[11]에서 유도한 수식결과에 따라서 CO-MAC 프로토콜에서의 CRTS 프레임 전송확률은 다음 식으로부터 구할 수 있다.

$$2\left\{2\left((1-\tau)^{n-1} - p_e^2\right) - 1\right\} - \left\{2\left((1-\tau)^{n-1} - p_e^2\right) - 1\right\}(W+1)\tau - \left\{1 - (1-\tau)^{n-1} + p_e^2\right\} W \left\{1 - 2^m \left(1 - (1-\tau)^{n-1} + p_e^2\right)^m\right\}\tau = 0 \quad (1)$$

여기에서  $n$ 은 송신노드로부터 송신도달거리 내에 있는 노드들의 총 수를 의미하며,  $W$ 는 CWmin을 의

미하고,  $m$ 은 최대 재전송 횟수 (retryLimit)를 의미한다. 한 개 이상 노드가 무선 채널 상에 RTS 프레임을 전송할 확률을  $P_{tr}$ 이라고 하면 다음과 같이 구할 수 있다.

$$P_{tr} = 1 - (1 - \tau)^n \quad (2)$$

한편 채널에 RTS 프레임을 전송했다는 조건 하에서 RTS 프레임이 충돌 없이 성공적으로 전송될 확률  $P_s$ 는 다음과 같이 구할 수 있다.

$$P_s = \frac{n\tau(1 - \tau)^{n-1}}{P_{tr}} (1 - p_e^2) \quad (3)$$

마지막으로 RTS 프레임을 전송하였을 경우 성공적으로 전송되었을 때의 슬롯시간  $T_s^{CO}$ , 그리고 충돌이나 전송오류 등으로 인해 실패했을 때의 슬롯시간  $T_c^{CO}$ 를 다음과 같이 나타낼 수 있다.

$$\begin{aligned} T_s^{CO} &= CRTS + ARTS + CCTS \\ &+ DATA(L, R_{sh}) + DATA(L, R_{hd}) + ACK \\ &+ 5SIFS + 6\delta + DIFS \end{aligned} \quad (4)$$

$$T_c^{CO} = CRTS + DIFS + \delta \quad (5)$$

식 (4)에서  $DATA(L, R_{sh})$ 는 크기가  $L$ 인 DATA 프레임을  $R_{sh}$  속도로 전송할 때 소요되는 프레임 전송시간을 의미하며, 제어프레임(예를 들면 RTS)은 표 1에 기술한 크기의 제어프레임을 기본 속도 (Basic Rate)로 전송할 때 소요되는 전송시간을 의미한다. 그리고  $\delta$ 는 전파지연시간으로 성능평가에서는 무시하였다. 그러므로 CO-MAC 프로토콜의 시스템 처리량 (throughput) 성능을 다음 식으로 나타낼 수 있다.

$$S = \frac{P_s P_{tr} E[P]}{(1 - P_{tr})\sigma + P_s P_{tr} T_s^{CO} + P_{tr}(1 - P_s) T_c^{CO}} \quad (6)$$

위 식에서  $E[P]$ 는  $DATA(L, R_{sh}) + DATA(L, R_{hd})$

이며,  $\sigma$ 는 빈 슬롯시간의 크기를 나타낸다.

## V. 성능평가 결과

성능평가 결과를 도출하기 위해 본 논문에서 사용한 시스템 파라미터를 표 1에 나타내었다. 그리고 송신노드, 목적지노드, 그리고 2개의 도움노드로 구성된 망을 고려하였다. 송신노드와 목적지노드는 기본 전송속도 (Basic Rate)로 프레임을 전송하며, 송신노드와 도움노드, 그리고 도움노드와 목적지노드 간에는 11 Mbps로 전송한다고 가정하였다. 성능평가 척도로 사용한 시스템 처리량은 식 (6)에서 정의한 값에 CO-MAC과 rDCF의 경우는 11 Mbps를 곱한 값을 사용하였으며, DCF의 경우는 송신노드와 목적지노드가 직접 통신하기 때문에 기본 속도인 1 Mbps를 곱한 값을 사용하였다.

표 1. 시스템 파라미터  
Table. 1 System parameters

시스템 변수	값	시스템 변수	값
CRTS 크기	352 bits	SIFS	10 $\mu$ s
ARTS 크기	352 bits	DIFS	50 $\mu$ s
CCTS 크기	304 bits	CWmin	31 slots
ACK 크기	304 bits	CWmax	1023 slots
DATA 크기	1024 bytes	retryLimit (m)	6
Slot 시간( $\sigma$ )	20 $\mu$ s	$R_{sh}, R_{hd}$	11 Mbps
RTS/CTS	160/112	Basic Rate	1 Mbps

그림 6과 7에는 본 논문에서 제안한 CO-MAC 프로토콜의 성능평가 결과를 나타내었다. 먼저 그림 6에는 다양한 프레임 전송오류 환경에서 노드 수의 변화에 따른 시스템 처리 성능을 나타내었다. 결과로부터 망 내에 위치한 노드 수가 증가함에 따라 시스템 처리량이 감소함을 알 수 있는데, 이것은 본 성능평가에서는 모든 단말이 포화상태에 있다고 가정하였기 때문에 단말 수가 증가하면 충돌확률이 증가하기 때문이다. 통상적으로 Cooperative 통신을 사용할 경우, 노드 수가 증가함에 따라 어느 지점까지는 시스템 성능이 증가한다는 연구결과가 있다[10].

이러한 결과는 도움노드와 송신노드 또는 목적지 노드 사이의 거리와 전송속도가 시간에 따라서 가변인 동적인 상황에서 발생하는데, 노드의 수가 증가하면 송신노드와 목적지노드 사이에 더 좋은 채널조건을 갖는 도움노드가 생길 확률이 증가하기 때문이다. 하지만 본 논문에서는 수학식을 사용한 성능평가 한계 때문에 도움노드와 송신노드 또는 목적지노드와의 환경을 최대조건인 11 Mbps로 항상 고정하였기 때문에 노드 수 증가에 따라서 시스템 성능이 일부 증가하는 현상을 확인할 수 없다. 그리고 노드 개수가 45개 이상일 때 시스템 처리량이 거의 감소하지 않음을 볼 수 있다. 이것은 참고문헌[11]에서 사용한 시스템 모델의 결함에 기인한 것으로, 해당 모델에서는 전송노드가 프레임 전송을 성공할 때까지 계속 재전송하는 모델을 채택하고 있다.

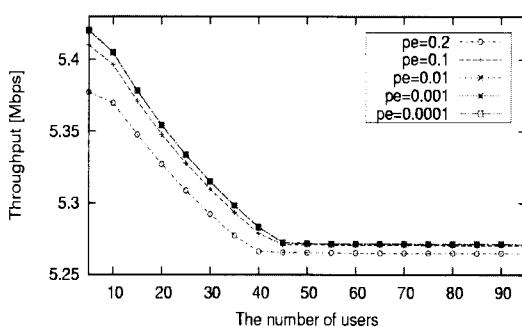


그림 6. 단말 개수에 따른 시스템 처리 성능  
Fig. 6 System throughput over the number of nodes

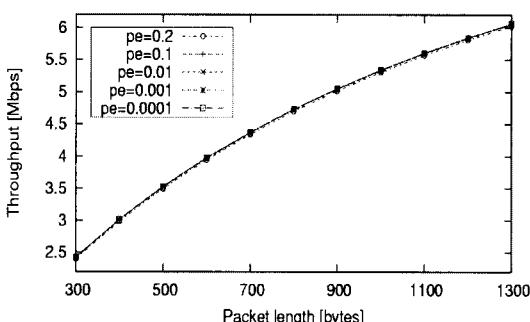


그림 7. 프레임 크기 변화에 따른 시스템 처리 성능  
Fig. 7 System throughput over frame size

그림 7은 노드의 수를 10으로 고정한 상태에서 DATA 프레임의 크기 변화에 따른 시스템 처리량을 나타낸다. DATA 프레임의 크기가 증가하면 전체적으로 제어 프레임이 차지하는 오버헤더 비율이 줄어들기 때문에 시스템 처리량이 증가한다.

한편 프레임 전송오류 값 변화에 대해서는 시스템 처리량 변화가 거의 없는 것처럼 보이지만, 이것은 그림 6과 비교할 때 시스템 처리량의 변화폭이 프레임 전송오류에 따른 시스템 처리량 변화폭보다 훨씬 큰 이유에 기인한 것이다. 프레임 전송오류가 증가함에 따라 그림 6과 같이 시스템 처리량이 일부 감소함을 확인 할 수 있었다.

그림 8은 참고문헌[8]에서 제안한 rDCF 기법과 본 논문에서 제안한 CO-MAC 기법의 성능평가 결과를 비교한 것이다. 프레임 전송오류가 감소하면 CO-MAC 기법과 rDCF 기법의 성능차이는 줄어들지만 프레임 전송오류가 클 경우에는 성능차이가 큼을 알 수 있다. 왜냐하면 프레임 전송오류가 작을 때에는 CO-MAC 기법에 따르면, 송신노드가 처음에 선택한 도움노드 H1을 통해서 대부분 통신하게 되며, 이렇게 되면 rDCF 기법과 유사하기 때문이다.

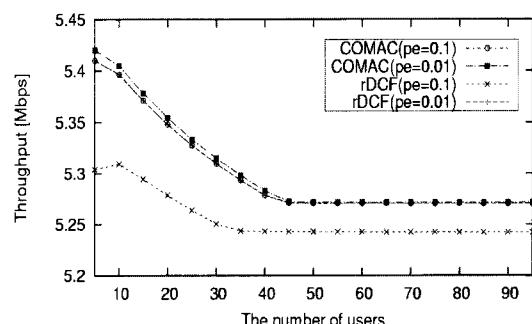


그림 8. 단말 개수 변화에 따른 성능비교  
Fig. 8 Throughput comparison over the number of nodes

그림 9는 DATA 프레임 크기의 변화에 따른 시스템 처리량을 DCF[11] 및 rDCF[8] 기법과 비교한 것이다. 이 결과에서 DCF가 상대적으로 성능이 매우 낮은 것은 송신노드와 목적지노드 간의 직접경로가 상대적으로 멀어서 프레임 전송속도가 매우 낮은 1 Mbps 전송 환경을

고려했기 때문이며, 직접 경로의 채널 환경이 개선되면 시스템 처리량 차이가 그림 9에 나타낸 것보다 줄어든다. 그리고 그림 9에서 시스템 처리량의 변화폭이 매우 크기 때문에 그림 8에서 확인했던 rDCF와 CO-MAC 기법 간의 시스템 처리량 차이가 상대적으로 작은 것처럼 보인다.

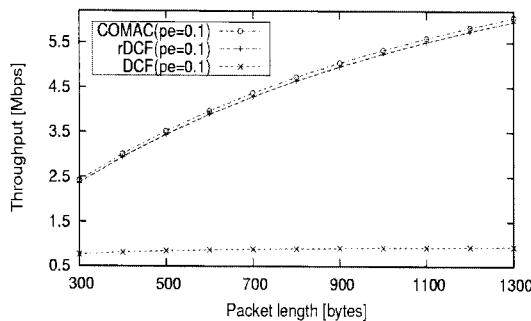


그림 9. 프레임 크기 변화에 따른 성능비교  
Fig. 9 Throughput comparison over frame size

## VI. 결론

본 논문에서는 Cooperative 통신에 사용할 수 있는 MAC 프로토콜을 제안하고 수학적인 접근방법을 사용하여 성능평가를 수행하였다. 성능평가 척도로는 시스템 처리량을 사용하였으며, IEEE 802.11 규격의 핵심인 DCF 기법, 그리고 기존 연구에서 Cooperative 통신용 MAC 프로토콜로 제안된 rDCF 기법과 성능비교를 수행하였다. Cooperative 통신을 사용할 경우 DCF 기법에 비해서는 성능이 많이 개선되며, 프레임 전송오류가 큰 환경에서는 rDCF 기법에 비해서도 성능이 개선됨을 확인할 수 있었다.

차후 연구로는 본 연구에서 제안한 프로토콜을 컴퓨터 모의실험을 통해 본 연구에서 얻지 못했던 좀 더 다양한 환경에서의 성능평가 결과를 도출할 계획이다.

## 참고문헌

- [1] IETF MANET Working Group. Available from: <<http://www.ietf.org/html.charters/manet-charter.html>>.
- [2] IEEE std 802.11-1997, part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, June 1997.
- [3] G. Holland, N. Vaidya, and P. Bahl, "A rate-adaptive MAC protocol for multi-hop wireless networks", in Proc. of ACM/IEEE MOBICOM'2001, Italy, June 2001.
- [4] H. Zhu and G. Cao, "On improving the performance of IEEE 802.11 with relay-enabled PCF," Mobile networks and applications, vol. 9, pp. 423~434, 2004.
- [5] Sai Shankar N, Chun-Ting Chou, and Monisha Ghosh, "Cooperative communication MAC (CMAC) - A new MAC protocol for next generation wireless LANs," in Proc. of IEEE Int. Conf. on Wireless Networks, Communications and Mobile Computing, Hawaii, June 2005
- [6] M. Dianati, X. Ling, K. Naik, and X. Shen, "A node-cooperative ARQ scheme for wireless Ad Hoc Networks," IEEE Trans. on Vehi. Tech., Vol 55, No. 3, pp. 1032~1044, May 2006.
- [7] G. Yu, Z. Zhang, and P. Qiu, "Cooperative ARQ in wireless networks: Protocol description and performance analysis," IEEE 2006.
- [8] H. Zhu and G. Cao, "rDCF: A relay-enabled medium access control protocol for wireless Ad Hoc Networks," IEEE Trans. on Mobile Computing, Vol. 5, No. 9, pp. 1201~1214, September 2006.
- [9] K. Tan, Z. Wan, H. Zhu and J. Andrian, "CODE: Cooperative medium access for multirate wireless Ad Hoc network," in Proc. of IEEE SECON'2007, 2007.
- [10] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, "CoopMAC: A cooperative MAC for wireless LANs," IEEE J. of Select. Areas, Vol. 25, No. 2, pp. 340 ~ 353, 2007.
- [11] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. of Select. Areas, pp. 537~547, March 2000.

### 저자소개



장 재신 (Jaeshin Jang)

1990년 2월 동아대학교  
전자공학과 학사  
1992년 2월 한국과학기술원  
전기및전자공학과 석사  
1998년 2월 한국과학기술원 전기및전자공학과 박사  
1997년 7월 ~ 2002년 2월 (주)삼성전자  
네트워크사업부 책임연구원  
2008년 8월 ~ 2009년 7월 Iowa 주립대 방문연구원  
2002년 3월 ~ 현재 인제대학교 공과대학  
정보통신공학과 조교수  
※관심분야: 이동통신망, 무선랜, 무선인터넷,  
센서네트워크, QoS 스케줄러