

애드혹 네트워크에서 협력통신을 위한 도움노드 선정방법 비교연구

장재신^{1†}

Comparison Study of Helper Node Selection Schemes of Cooperative Communications at Ad Hoc Networks

Jaeshin Jang

ABSTRACT

In this paper, a study on finding an appropriate helper node that can help increase effective frame transmission rate for cooperative communications at ad hoc networks is carried out. Those researches from reference^{[7]-[9]} use the reactive helper node selection mechanism which starts its role after exchanging RTS and CTS frames between source and destination nodes, and are implemented into our simulator for performance comparison. System throughput and average channel access delay are used for performance measures and all communicating nodes are assumed to move independently within the communication range. It is anticipated that this research result can be used as basic information for designing a new efficient helper node selection scheme.

Key words : Cooperative communications, MAC protocol, System throughput, ad hoc networks

요 약

본 논문에서는 애드혹 무선 네트워크에서 채널 전송속도를 높일 수 있는 협력통신 기법에서 도움노드를 선정하는 방법에 대한 연구를 수행하였다. 송수신 노드가 RTS/CTS 제어 프레임의 서로 교환한 이후에 도움노드를 선정하는 사후선정 기법을 채용한 참고문헌^[7-9] 연구결과를 컴퓨터 모의실험을 사용하여 성능비교를 수행하기 위해 시뮬레이터를 구현하였다. 성능평가 척도는 시스템 처리량과 채널 액세스 지연시간을 사용하였으며, 통신노드는 모두 지정된 통신영역 내에서 독립적으로 이동하는 모델을 사용하였다. 성능비교와 장단점 분석을 통해 취합된 결과는 성능이 우수한 새로운 도움노드 선정기법을 설계하는데 귀중한 자료가 될 것으로 판단된다.

주요어 : 협력통신, 도움노드, MAC 프로토콜, 시스템 처리량, 애드혹 네트워크

1. 서 론

이동통신을 포함하여 무선채널을 사용하는 무선 랜과 같은 무선통신 시스템은 제한된 주파수 자원과 열악한 무선채널 때문에 실효 전송속도와 시스템 용량을 증대시키는 것이 아주 중요한 목표이다^[1]. 이 목표를 달성하기 위해 물리계층에서는 OFDM(orthogonal frequency division multiplexing), MIMO(multiple input multiple output)

안테나 등 다양한 해결책이 제시되었으며, 이 분야에 대한 연구는 계속 진행 중이다^[2]. 한편 추가적인 하드웨어를 사용하지 않으면서 송신노드와 수신노드 사이에 존재하는 다른 통신노드들의 도움으로 실효 전송속도를 개선하려는 노력이 최근 몇 년 동안 많이 연구되었는데 대표적인 것이 협력통신(cooperative communications)이다^[3]. 무선통신에서는 채널상태가 양호하면(또는 수신 SNR 값이 크면) 높은 레벨 변복조 기법을 사용하여 데이터를 전송함으로써 전송속도를 높일 수 있다. 따라서 송신노드와 수신노드가 직접 통신하는 경우보다 중간에 위치한 임의의 통신노드가 도움노드(helper node)로서 역할을 수행하여 협력통신을 가능케 하면 송신노드와 도움노드, 그리고 도움노드와 수신노드 사이의 무선채널상태가 크게 개선

접수일(2012년 3월 18일), 심사일(1차 : 2012년 5월 4일),
게재 확정일(2012년 5월 23일)

¹⁾ 인제대학교 정보통신공학과

주 저 자 : 장재신

교신저자 : 장재신

E-mail; icjoseph@inje.ac.kr

되어 송수신 노드가 직접 통신하는 것보다 실효 전송속도를 높일 수 있다. 협력통신은 물리계층 연구에서 시작되었기 때문에 MAC 계층에서의 연구는 물리계층 연구결과물과 비교하면 크게 부족하다. 그러나 협력통신이 좋은 성능결과를 제공하기 위해서는 물리계층 뿐만 아니라 상위계층과의 협력이 절대적으로 필요함은 두말할 나위가 없다. 그러므로 성능이 우수한 협력통신용 MAC 프로토콜을 연구하는 것은 무선통신 시스템의 성능을 개선하는데 매우 중요하다.

한편 MAC 계층 관점에서 바라볼 때 협력통신을 지원하기 위해서는 기존의 무선통신에서 취했던 절차와는 달리, 협력통신을 지원하는 통신노드에서는 수신한 프레임의 목적지가 해당 통신노드가 아닐지라도 수신한 프레임으로부터 필요한 정보를 수집해야 하며, 도움이 필요할 경우에는 도움노드로서 송신노드와 수신노드 사이의 통신에 참여해야 한다. 이것은 무선 랜과 같은 기존 통신 시스템에서 임의의 A 통신노드가 어떤 프레임을 수신했을 때 해당 프레임의 목적지 주소가 A 통신노드가 아닐 때는 해당 프레임을 폐기했던 것과는 큰 차이가 있다. 또한 MAC 계층에서는 송신노드가 수신노드에게 프레임을 전송하고자 할 때 어떤 도움노드를 사용하여 통신하면 시스템 성능이 크게 개선되는지를 해결해야 한다. 즉, 적절한 도움노드를 찾는 것이 MAC 계층에서 해결해야 할 협력통신 관련 주요 안건 중의 하나이다. 해당 주제와 관련된 협력통신용 MAC 프로토콜에 대한 연구는 다소 진행되었지만 제안된 연구결과들의 성능비교에서 다른 협력통신용 MAC 프로토콜과 성능 비교한 예는 찾아보기 힘들며, 대부분 협력통신 기법을 채용하지 않은 기존 무선 랜의 MAC 프로토콜에 기반을 둔 RBAR 기법^[6]과 성능비교를 수행하였다. 그래서 효율적이며 성능이 우수한 새로운 협력통신용 MAC 프로토콜을 설계하기 위해서는 기존에 제안된 방식들의 성능 비교분석 자료가 반드시 필요하다고 판단되어 본 논문에서 해당 연구를 수행하였다.

본 연구는 MAC 계층에서 협력통신에 필요한 도움노드를 선정하는 방법에 대한 선행연구로서, 지금까지 제안되었던 몇 가지 도움노드 선정방법들의 성능을 서로 비교 분석하고 장단점을 파악하여 얻은 연구결과를 새로운 도움노드 선정방법을 찾는데 활용하고자 한다. 2장에서는 협력통신에서 도움노드 선정기법과 관련된 기존 연구동향을 기술하였고, 3장에서는 선정된 세 가지 도움노드 선정기법을 컴퓨터 모의실험에 의해 성능평가하기 위해 시뮬레이터를 구현한 과정에 대해서 기술하였다. 4장에서는 성능평가 결과를 제시하고 상호 비교분석하였으며 5장에

서 본 논문을 끝맺는다.

2. 관련 연구동향

이동통신, WiBro 시스템 등은 시스템 성능을 향상시키기 위해 물리계층에 다양한 전송속도의 물리계층 채널을 제공하며, 송수신 노드 사이의 무선채널 상태에 따라 적절한 통신 신뢰도를 갖춘 변복조 및 코딩 기법을 갖춘 물리계층 채널을 사용하여 데이터를 전송할 수 있는 링크 어댑테이션 기법을 사용한다. 무선 랜에서도 다양한 속도의 물리계층 채널을 제공하는데, 링크 어댑테이션 기능을 사용하기 위해 물리계층 채널 전송속도를 송신노드와 수신노드 중에서 누가 결정하느냐에 따라서 크게 두 가지로 나눌 수 있다. 첫 번째는 송신노드가 물리계층 전송속도를 결정하는 방법으로 ARF(auto rate fallback) 기법이 여기에 해당된다. ARF 기법은 Lucent사에서 만든 WaveLAN II 무선 랜에서 사용하는 것으로 수신노드로부터 받은 피드백 정보(ACK 프레임 수신 유무 등)를 토대로 송신노드가 신뢰성 있게 전송할 수 있는 적절한 물리계층 전송속도를 결정한다. 두 번째는 수신노드가 직접 물리계층 전송속도를 결정하는 것으로 참고문헌^[6]에서 제안한 RBAR(receiver-based auto rate) 기법이 여기에 해당된다. 이 기법은 수신노드가 RTS 제어프레임들을 수신하면서 측정한 채널상태정보(CSI: channel state information) 값을 기반으로 최적의 물리계층 채널 전송속도를 결정하여 송신노드에게 CTS 제어프레임을 통해서 알려준다.

협력통신에서 도움노드 역할을 수행하기 위해서는 해당 도움노드가 목적지인 제어프레임 뿐만 아니라 다른 노드에게 전송되는 제어프레임들도 모두 수신한 뒤 필요한 정보를 수집해야 한다. 한편 협력통신에 참여할 도움노드를 선정하는 시점에 따라 사전선정(proactive) 방법과 사후선정(reactive) 방법으로 분류할 수 있다. 도움노드 사전선정 방법은 후보 도움노드들이 주변 노드가 전송한 제어프레임을 항상 분석하여 주변노드 등과 해당 후보 도움노드사이의 채널상태정보를 측정하고, 이를 토대로 적절한 물리계층 전송속도를 파악한 결과들을 테이블에 기록 관리해야 한다. 그리고 파악한 정보들은 주기적으로 인접노드들과 교환하여 주변노드들에 대한 채널상태정보를 미리 파악하고 있어야 한다. 그래서 만일 어떤 통신노드에서 전송할 데이터가 발생하면 해당 노드가 관리하고 있는 테이블을 검색하여 해당 송신노드와 수신노드 사이에 최적의 도움노드를 먼저 선정된 뒤, RRTS(relay RTS) 프레임을 수신노드가 아닌 도움노드에게 전송하여 협력통

신을 시작한다. 사전선정 방법을 사용하는 협력통신 MAC 프로토콜로는 참고문헌^[7]에서 제안한 rDCF 기법과 참고문헌^[8]에서 제안한 CoopMAC 기법이 있다. 두 기법은 서로 유사하며, 모든 통신노드가 테이블에 해당하는 Willing 리스트를 관리한다. 그러나 사전선정 방법은 최적 도움노드를 사전에 찾아서 관리해야 하는 부담이 너무 크며, 단말의 이동성 때문에 데이터를 전송해야 하는 시점에서는 선정된 도움노드가 최적이지 아닐 수도 있다는 단점을 갖는다. 또한 각 통신노드가 관리하고 있는 테이블 내용을 주기적으로 인접노드들과 공유하기 위해서는 이와 관련된 교환절차와 패킷규격이 규정되어야 하나 이에 대한 연구가 부족하고, 주기적으로 테이블 정보를 서로 공유하는 것은 망 내에 큰 부하를 야기할 수 있다는 단점이 있다.

참고문헌^[9]^[11]에서는 송수신 노드가 서로 RTS 프레임과 CTS 프레임을 서로 교환한 이후 도움노드 선정단계에 돌입하는 사후선정 기법을 채택하고 있다. 먼저 참고문헌^[9]에서는 협력통신용 MAC 프로토콜로 CTBTMA (cross layer triple busy tone multiple access) 기법을 제안하였으며, 이 기법은 RTS와 CTS 제어프레임을 송수신노드가 서로 주고받은 후, 후보 도움노드들이 RTS/CTS 제어프레임을 수신하면서 파악한 채널상태정보를 기반으로 Utility 함수를 계산한다. Utility 함수는 실패 전송속도에 해당하며, Utility 값이 클수록 도움노드 역할을 잘 수행할 수 있기 때문에 후보 도움노드 경쟁에서 Utility 함수가 클수록 busy tone을 오랫동안 전송하게 함으로써 최종적으로는 Utility 함수가 가장 큰 도움노드가 RTH(ready to help) 프레임을 전송하여 도움노드로 선정되도록 한다.

참고문헌^[10]에서는 후보 도움노드 간의 충돌을 줄이기 위해 송수신 노드가 직접 통신하는 것보다 성능을 개선시킬 수 있다고 판단한 후보 도움노드들이 계산한 CCTR (composite cooperative transmission rate) 값을 기반으로 몇 개의 그룹으로 분류한 뒤 2단계 경쟁을 수행한다. 첫 단계 경쟁은 분류된 그룹간의 경쟁이며, 두 번째 단계 경쟁은 선정된 그룹 내의 멤버들 간의 경쟁으로, 이를 통해 후보 도움노드들 간의 경쟁을 어느 정도 해결하며, RTH 프레임 전송 시 충돌이 발생한 경우를 대비하여 K 개 미니슬롯 중 임의로 선정된 한 개의 미니슬롯을 통해 RTH 프레임을 전송함으로써 도움노드 선정경쟁에서 충돌을 크게 줄이고 있다. 또한 제안된 프로토콜 운영에 필요한 각종 파라미터들의 값들을 최적화 절차를 통해 구하도록 제안하고 있는 것이 눈에 띄나, 최적화 절차가 너무 복잡하여 실효성 문제는 검토가 필요하다.

마지막으로 참고문헌^[11]에서는 CRBAR(cooperative relay-

based auto rate) MAC 기법을 제안하였는데, 이 기법에서는 후보 도움노드들이 RTS/CTS 제어프레임을 수신하면서 측정된 채널상태정보에 따라 선정된 채널 전송속도에 기반을 둔 확률 값을 사용하여 p-persistent 백오프 기법으로 도움노드 선정경쟁을 수행한다. 이 방식은 앞의 두 가지 기법보다 간단하기는 하지만 확률 값에 기반을 둔 p-persistent 백오프 기법을 사용하면 최적의 도움노드를 항상 선정한다고 보장 할 수 없으며, 도움노드 선정기법의 성능이 이 확률 값에 크게 영향을 받기 때문에 좋은 성능을 유지하기가 어려울 것으로 판단된다.

지금까지 설명한 사후선정기법들은 도움노드선정에서 채널 전송속도에 기반을 둔 성능척도를 모두 사용하고 있으나 채널 전송속도는 그 값이 다양하지 못하기 때문에 통신영역 내에 배치된 통신노드의 밀도가 증가하면 동일한 성능척도 값을 가진 도움노드 수가 증가하므로 도움노드 경쟁에서 지속적으로 충돌이 발생할 가능성이 매우 크다. 도움노드 선정단계에서 적절한 도움노드를 선정하지 못하고 실패하는 빈도가 증가하면 도움노드를 찾기 위해 시간을 많이 투자했지만 결국 협력통신을 사용하지 못하게 되므로, 협력통신을 도입한 경우가 송수신 노드가 직접 통신하는 경우보다 성능이 더 나빠져서 낭패를 겪을 수도 있다.

3. 시스템 모델링 및 시뮬레이터 구현

본 논문에서 컴퓨터 모의실험을 통해 성능비교를 할 협력통신용 도움노드 선정기법들은 모두 송수신 노드가 RTS 프레임과 CTS 프레임을 상호 교환한 이후에 도움노드를 찾는 도움노드 사후선정 방식으로 도움노드를 찾는 단계가 차지하는 위치는 그림 1에 나타낸 바와 같다. 본 논문에서 비교한 세 가지 도움노드 사후선정 방식에서 채택한 도움노드를 선정하는 절차에 대한 설명과 그림은 참고문헌^[7]^[9]를 참고하기 바라며, 지금부터는 세 가지 도움노드 사후선정 방식을 컴퓨터 모의평가를 위해 시뮬레이터에서 어떻게 구현하였는지 측면에서 자세하게 설명하고자 한다.

먼저 참고문헌^[9]에서 제안한 CTBTMA 기법에서 도움노드를 찾는 절차를 그림 2와 같이 시뮬레이터에 구현하였다. 먼저 RTS와 CTS 제어프레임을 수신한 후보 도움노드들은 식 (1)과 식 (2)을 사용하여 Utility 함수 U 를 계산한다. 여기에서 W 는 데이터 프레임의 길이이며, γ_{sh} 는 송신노드와 도움노드사이의 채널 전송속도를 의미한다.

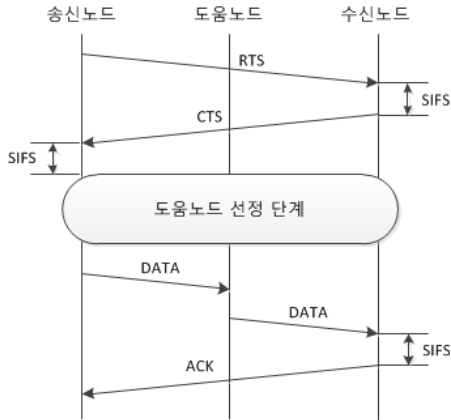


그림 1. 협력통신 일반 호 처리도

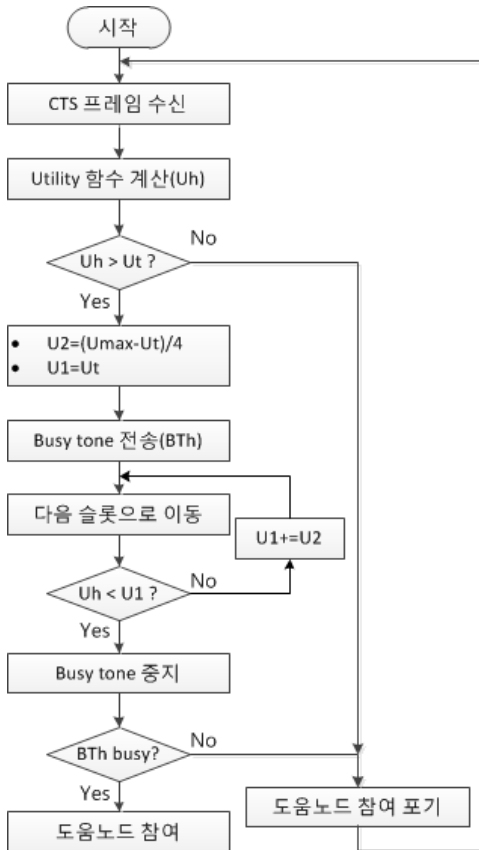


그림 2. CTBTMA 도움노드 선정절차

$$U = W / (T_o + T_p) \tag{1}$$

$$T_p = W / r_{sh} + W / r_{hd} \tag{2}$$

만약 후보 도움노드의 Utility 함수 U_h 값이 송수신 노드가 직접 통신하는 환경을 염두에 두고 송신노드가 계산한 Utility 함수 U_t 보다 크면 협력통신이 송수신 노드가 직접 통신하는 경우보다 실효 전송속도가 높은 것을 의미하므로 해당 후보 도움노드가 선정경쟁 절차에 참여한다. 선정경쟁에 참여한 후보 도움노드는 Utility 함수가 클수록 busy tone을 오랫동안 전송하며, 가장 오랫동안 busy tone을 전송한 후보 도움노드가 경쟁에서 승리하여 최종 도움노드로 선정된다. 컴퓨터 모의실험용 시뮬레이터 구현에서는 γ_{sh} 와 γ_{hd} 가 모두 최대 전송속도인 경우의 Utility 함수 U_{max} 와 U_t 의 크기 차이를 네 등분하고, 후보 도움노드들의 U_h 값에 따라 네 가지 그룹으로 분류한 뒤, 기준 Utility 함수 U_1 을 점차 증가시키면서 U_h 가 속해있는 그룹을 찾는다. 해당 그룹에 도달하면 busy tone 전송을 중지한 뒤, 채널 상에 busy tone이 전송 중인지 확인한다. 만약 busy tone이 전송 중이라면 채널상태가 더 우수한 다른 후보 도움노드가 존재함을 의미하기 때문에 도움노드 선정 경쟁을 포기하며, 만약 채널 상에 busy tone이 없으면 Utility 함수가 더 큰 후보 도움노드가 없음을 의미하므로 RTH 프레임을 전송하여 도움노드로 선정되었음을 알린다.

참고문헌^[10]에서 제안한 MAC 기법에서 도움노드를 찾는 과정을 시뮬레이터에 그림 3과 같이 구현하였다. 해당 기법에서는 EPTR (effective payload transmission rate)를 계산하여 송수신 노드가 직접 통신할 때의 실효 전송속도 C_1 과 비교하여 도움노드 경쟁에 참여할지 여부를 판단한다. 이때 EPTR은 참고문헌^[9]에서 사용한 Utility 함수와 매우 유사하다. 도움노드 경쟁에 참여한 후보 도움노드들은 식 (3)에 따라 CCTR R_h 를 계산한 뒤, 이 값의 크기에 따라 M개의 그룹으로 분류한다.

여기에서 R_{C_1} 과 R_{C_2} 는 참고문헌^[9]에서의 γ_{sh} 와 γ_{hd} 와 동일하다.

$$R_h = \frac{W}{W/R_{C_1} + W/R_{C_2}} = \frac{R_{C_1}R_{C_2}}{R_{C_1} + R_{C_2}} \tag{3}$$

다음에는 2단계 도움노드 선정경쟁을 수행하는데, 그룹 간 경쟁인 1단계 경쟁은 GI(group indication) 슬롯을 이용하며, 그룹 내에서 멤버들 간의 경쟁인 2단계 경쟁은 MI(member indication) 슬롯을 사용하여 수행한다. 경쟁

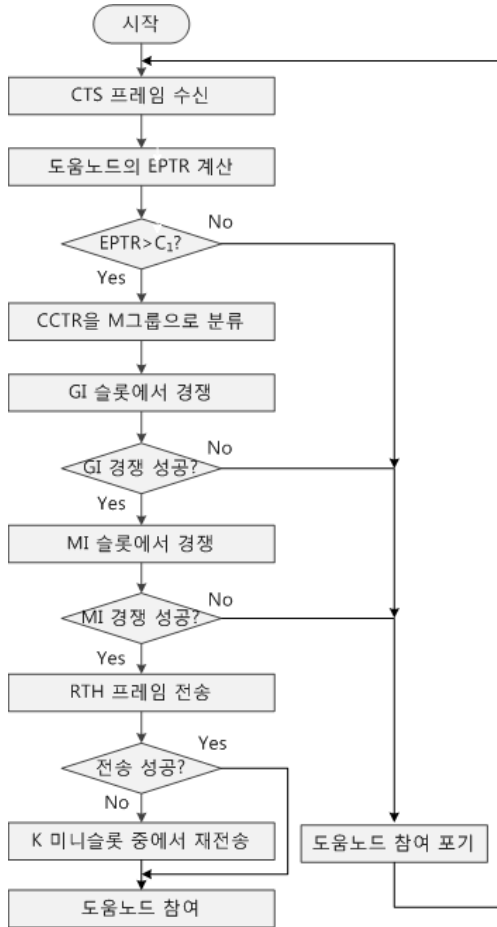


그림 3. Cross-layer MAC 도움노드 선정절차

은 CTBTMA에서 busy tone을 사용하여 경쟁하는 것과 유사하며 CCTR 값이 큰 그룹 또는 멤버가 경쟁에서 승리하는 방식으로 진행된다. 2단계 경쟁에서 승리한 후보 도움노드는 도움노드로 선정되었음을 RTH 프레임을 전송함으로써 송수신 노드에게 알린다. 그러나 2단계 경쟁까지 진출한 후보 도움노드가 2개 이상 존재할 수 있으며 이 경우에는 RTH 프레임 전송에서 충돌이 발생한다. RTH 프레임 충돌이 발생한 경우에는 K개의 미니슬롯 중 임의의 선정된 미니슬롯을 통해 RTH 프레임을 재전송하고 도움노드 선정단계를 끝낸다.

참고문헌^[11]에서 제안한 CRBAR 기법에서 도움노드를 찾는 과정을 시뮬레이터에 구현한 내용을 그림 4에 나타내었다. CRBAR 기법에서 도움노드를 찾는 절차는 p-persistent 백오프 기법에 기반을 두고 동작하고, 각 후보 도움노드에서 사용하는 확률 값 p_i 는 식 (4)으로 표시되며, δ 는 시

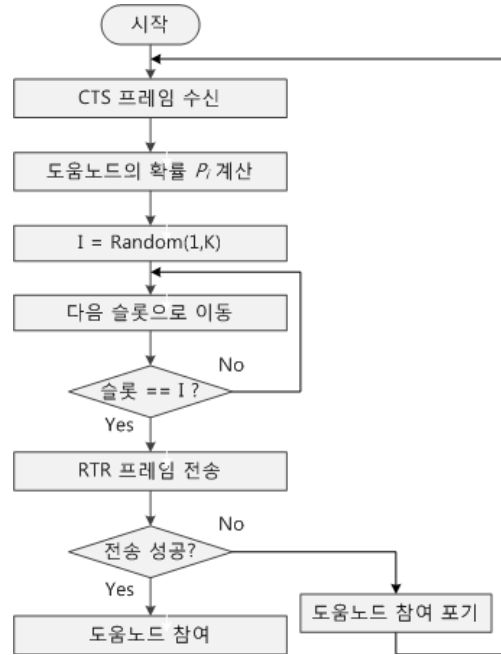


그림 4. CRBAR 도움노드 선정절차

스템 변수로 도움노드 선정 성공률과 밀접한 관련이 있다.

$$p_i = \delta / (1/r_{sh} + 1/r_{hd}) \quad (4)$$

윗 식에서 확률 값 p_i 가 해당 후보 도움노드와 송신노드 그리고 수신노드와 채널 전송속도와 밀접한 관련이 있음을 알 수 있는데, 이것은 채널상태가 좋으면 p-persistent에서 사용할 확률 값이 증가하여 도움노드로 선정될 확률을 높여주기 위함이다.

4. 성능평가 결과 분석

본 논문에서는 세 가지 도움노드 사후선정 기법의 성능비교를 위해 컴퓨터 모의실험을 수행하였으며, 모의실험 툴로는 SMPL을 사용하여 C++ 언어로 구현하였다^[12]. 성능평가 척도로는 시스템 처리량(throughput)과 채널 액세스 지연시간을 사용하였다. 시스템 처리량은 시뮬레이션 수행시간 동안 송신노드가 수신노드로 성공적으로 전송한 총 데이터 크기(bits)를 모의실험 총 수행시간으로 나눈 값을 의미하며 단위는 bps이다. 또한 채널 액세스 지연시간은 송신노드에서 데이터 전송을 위해 채널 접근 제어를 시작한 시점(백오프 카운터를 신규로 설정한 시점)으로부터 수신노드로부터 성공적으로 ACK 프레임으

수신할 때까지 소요된 평균 시간을 의미한다. 한편 본 논문에서 성능평가 척도로 사용한 시스템 처리량은 시스템 최대 처리량으로 볼 수 있다. 왜냐하면 본 연구에서 사용한 트래픽 모델은 각 송신노드의 버퍼에 전송할 데이터 패킷이 항상 존재한다고 가정한 Saturated 트래픽 모델을 사용하였고, 기존 연구에서도 최대 시스템 성능을 구할 때 자주 사용되었다. 성능평가를 수행하면서 사용한 시스템 파라미터를 표 1에 나타내었다.

성능평가에서 제어 프레임과 PLCP 헤더, 그리고 MAC 헤더는 기본 전송속도로 전송되며, 데이터 프레임의 페이로드는 두 노드 사이의 거리에 따라 표 2에 지정된 속도로 전송된다고 가정하였다. 또한 통신노드들이 초기상태에서는 100m×100m 크기의 통신영역 내에서 균일하게 배치되어 있으며, 시간이 흐름에 따라 Random waypoint 이동성 모델에 따라 모든 통신노드가 독립적으로 움직인다고 가정하였다⁶⁾. 참고문헌⁶⁾에서 사용한 이동성 모델과 차이점은 통신노드의 이동속도가 0과 Max_speed(표 2 참조) 사이에서 균일분포를 따른다고 가정한 점과, 이동 후 정지해 있는 시간도 0과 Max_pause(표 2 참조) 사이 값에서 균일분포를 따른다고 가정한 것이다. 본 성능평가에서 Max_speed 값을 비현실적으로 매우 크게 설정한 이유는 시뮬레이션 수행시간이 1500초로 실제 통신상

표 1. 시스템 파라미터

시스템변수	값	시스템변수	값
RTS 크기	160 bits	SIFS	10 μ s
CTS 크기	112 bits	DIFS	50 μ s
RTH 크기	164 bits	CWmin	32 slots
ACK 크기	112 bits	CWmax	1024 slots
DATA 크기	1024 bytes	PLCP 헤더	192 bits
슬롯 시간	20 μ s	MAC 헤더	224 bits
소요시간	1500 초	기본 속도	1 Mbps
GI(MI)	3(3)	K	4
δ	0.01, 0.02, 0.06		
전송속도	1, 2, 5.5, 11 Mbps		
CRBAR 미니슬롯 수	6		

표 2. 이동성 모델 파라미터

속도	11Mbps	5.5Mbps	2Mbps	1Mbps
거리	48.2m	67.1m	74.7m	≥ 100 m
Max_speed	300 m/sec	Max_pause	10	

황과 비교할 때 매우 짧기 때문에 해당 성능평가 시간 동안 통신노드가 지정된 통신영역 내에서 골고루 충분히 움직이도록 하기 위해서는 이동속도를 매우 빠르게 하지 않을 수 없었다.

그림 5와 그림 6은 CTBTMA 기법에서 도움노드 개수의 변화에 따른 시스템 처리량 성능변화를 나타낸 것이다. 시스템 처리량 성능 결과를 살펴보면 송신노드 수가 증가함에 따라 시스템 처리량이 감소함을 알 수 있다.

이것은 송신노드의 트래픽 모델을 Saturated 트래픽 모델을 사용하였기 때문에 송신노드가 5개 정도만 되면 이미 망에서 최대 수송할 수 있는 한계에 도달하였으며, 송신노드 수가 더 증가하면 충돌로 인한 재전송 때문에 시스템 처리량이 감소하는 것이다(그림 7 참조). 한편 도움노드 수가 증가함에 따라 시스템 처리 성능이 계속 감소하는 것은 처음에 직관적으로 이해하기 어려우나 이것은 CTBTMA 기법이 도움노드 경쟁에서 최적의 도움노드를 잘 선택하지 못하는 것에 기인한다.

즉, 최적 도움노드를 선정하기 위해 식 (1)을 사용하지만 채널 전송속도가 4개로 이루어져 있으면 후보 도움노드의 개수가 증가함에 따라 동일한 Utility 값을 갖는 후보 도움노드 개수가 덩달아 증가한다. 따라서 도움노드를

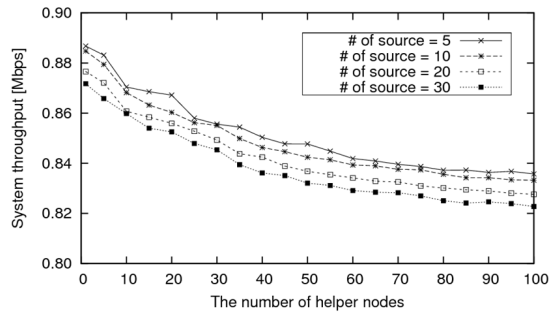


그림 5. CTBTMA 기법의 시스템 처리량 성능 변화1

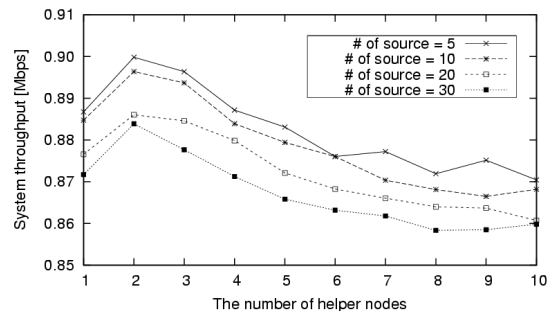


그림 6. CRBAR 기법의 시스템 처리량 성능변화2

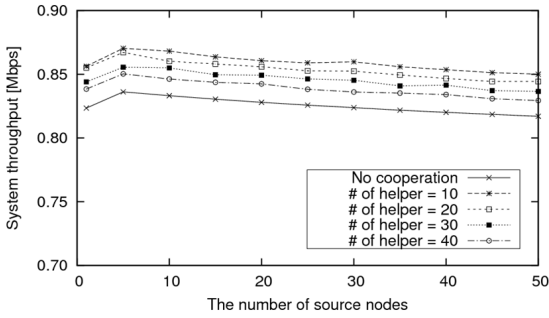


그림 7. CRBAR 기법의 시스템 처리량 성능변화3

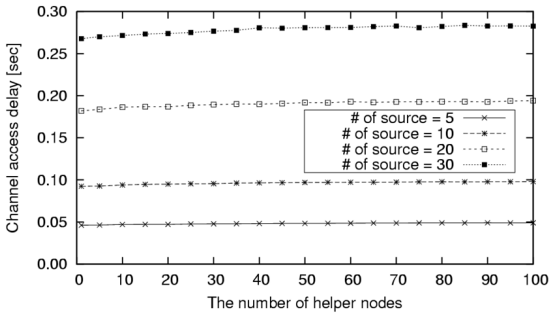


그림 8. CTBTMA 기법의 채널 액세스 지연시간 성능

선정하기 위한 선정경쟁을 시작했지만 성공하지 못하고 결국에는 송신노드와 수신노드가 직접 통신하는 단계로 회귀하는 횟수가 증가하기 때문이며 그림 6에서 이 결과를 확인할 수 있다. 그림 6에 따르면 도움노드가 약 2개일 때 이미 최댓값에 도달하며 그 이후에는 성능이 계속 떨어짐을 알 수 있다.

그림 7에서는 송신노드 개수 변화에 따른 시스템 처리량 성능 변화를 보여준다. 이 결과에 따르면 송신노드 개수가 약 5개 일 때 시스템 처리량이 최대가 되며, 그 이후에는 도움노드 개수가 증가할수록 시스템 처리량이 감소함을 알 수 있다. 하지만 본 성능평가 범위 내에서는 여전히 협력통신을 사용하지 않는 경우와 비교하여 협력통신을 사용할 경우 성능이 더 우수함도 알 수 있다.

도움노드 개수 증가에 따른 채널 액세스 지연시간 변화를 그림 8에 나타내었다. 도움노드가 증가함에 따라 채널 액세스 지연시간이 완만하게 증가하며, 송신노드의 개수가 증가함에 따라 평균 지연시간이 큰 폭으로 증가하는 것을 알 수 있는데 이것은 송신노드 개수가 증가하면 송신노드 간 충돌 확률이 증가하여 재전송에 따른 평균 지연시간이 증가하기 때문이다.

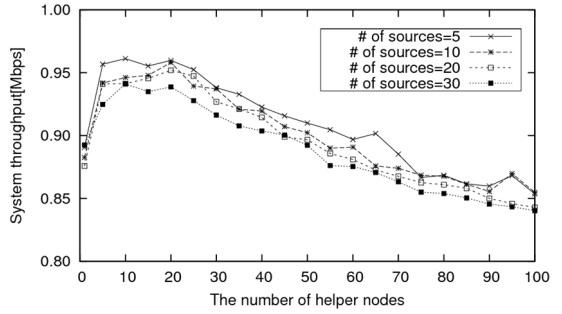


그림 9. Cross-layer MAC기법의 시스템 처리량 성능

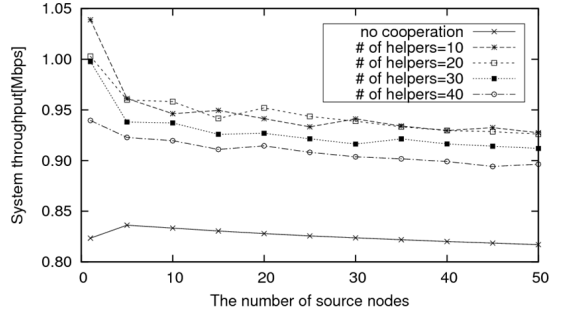


그림 10. Cross-layer MAC기법의 시스템 처리량 성능

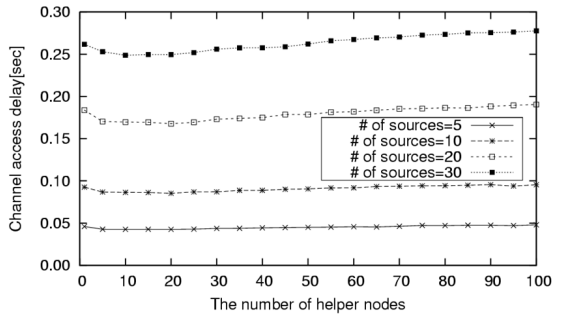


그림 11. Cross-layer MAC기법의 평균 액세스 시간

참고문헌^[10]에서 제안한 협력통신용 MAC 기법의 성능평가 결과를 그림 9 ~ 그림 11에 나타내었다. 먼저 그림 9는 도움노드 개수 변화에 따른 시스템 처리량 성능변화를 보여주는 것으로 그림 5의 CTBTMA 기법과 비교하면 시스템 처리량이 더 크고, 최적 도움노드 개수도 10개 ~ 20개 사이 값으로 최적 도움노드를 더욱 더 잘 선정하고 있음을 알려준다. 그림 10은 송신노드 개수 변화에 따른 시스템 처리량 변화를 보여주며, 이 결과에 따르면 협력통신을 사용하지 않고 송수신 노드가 직접 통신하는 경우와 비교하여 시스템 성능이 큰 폭으로 증가했음을 알

수 있다.

또한 그림 11에서는 도움노드 개수 변화에 따른 채널 액세스 지연시간의 변화를 보여준다. 도움노드 개수가 10 ~ 20개 사이에서 감소했다가 나머지 구간에서는 완만하게 증가하는데 이것은 그림 8의 결과와 유사하다.

그림 12 ~ 그림 14는 CRBAR 기법의 성능평가 결과를 보여준다. 먼저 그림 12는 도움노드 개수 변화에 따른 시스템 처리량의 변화를 나타내는데($\delta = 0.02$), 그림 5와 그림 9의 결과와 비교하면 도움노드 개수의 증가에 따라 시스템 처리량이 매우 급격하게 감소함을 알 수 있다. 이

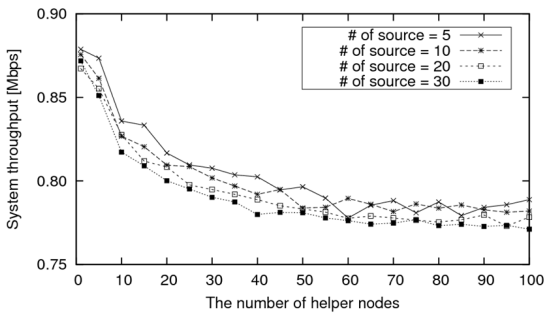


그림 12. CRBAR기법의 시스템 처리량 성능

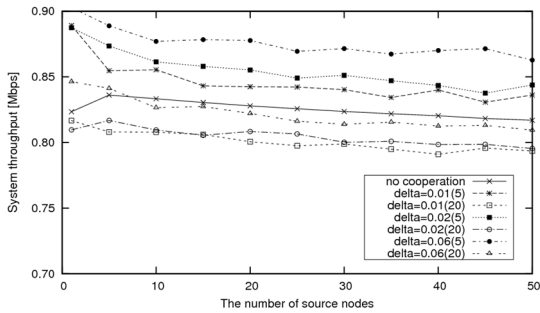


그림 13. CRBAR기법의 시스템 처리량 성능

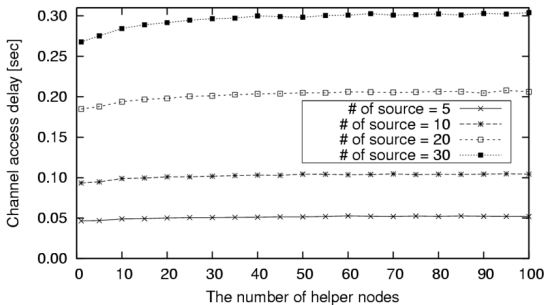


그림 14. CRBAR기법의 채널 액세스 지연시간

것은 p-persistent 백오프 기법이 안고 있는 근본적인 문제점에 기인한 것으로 판단되며 도움노드 수가 증가함에 따라 충돌 가능성이 급격하게 증가하기 때문이다. 한편 CRBAR 기법의 경우는 p-persistent 백오프에 사용하는 확률 값이 식 (4)에 따르면 파라미터 δ 와 밀접하게 관련되어 있음을 알 수 있다. 따라서 그림 13에는 파라미터 δ 값의 변화에 따른 시스템 처리량 성능을 송신노드 개수의 변화에 따라 나타내었다.

그림 13의 범례에서 괄호 안의 값은 도움노드의 개수를 나타낸다. 성능결과에 따르면 시스템 처리량 성능이 파라미터 δ 값의 크기에 밀접하게 관련되어 있음을 알 수 있다. 또한 도움노드 크기가 20개 이상이 되면 대부분의 범위 내에서 송수신 노드가 직접 통신하는 경우보다 성능이 오히려 더 떨어짐을 알 수 있다. 이것은 p-persistent 백오프 기법이 당초에 우려했던 대로 도움노드를 선정단계에서 최적 도움노드를 제대로 찾아내지 못하기 때문이다.

그림 14는 CRBAR 기법의 채널 액세스 지연시간 성능 결과를 나타내며, 앞에서 제시한 두 가지 기법과 유사하게 도움노드 개수 증가에 따라 완만하게 증가함을 알 수 있다.

지금부터는 이제까지 제시한 세 가지 도움노드 선정기법의 성능차이를 좀 더 정확하게 알아보기 위해 한 그림에 나타내었다. 먼저 그림 15에는 송신노드의 개수가 5개이고 CRBAR 기법의 경우 $\delta = 0.06$ 인 경우, 도움노드 개수의 변화에 따른 세 가지 도움노드 선정기법의 시스템 처리량 크기를 서로 비교하여 나타내었다.

결과에 따르면 참고문헌^[10]에서 제시한 Cross-layer MAC 기법이 제일 우수한 성능을 가짐을 알 수 있는데, 이것은 저자가 판단하기에 K개 미니슬롯을 사용하여 충돌해소를 한 번 더 수행했기 때문이라고 생각한다. 왜냐하면 CTBTMA 기법 설명에서 언급했듯이 본 연구에서 사용한 무선 랜 시스템은 채널 전송속도가 4가지만 존재

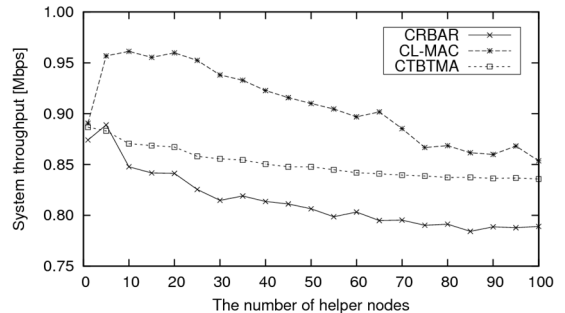


그림 15. 사후 선정기법 성능비교

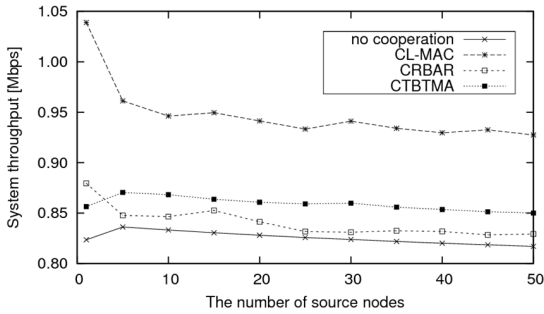


그림 16. 사후 선정기법 성능비교2

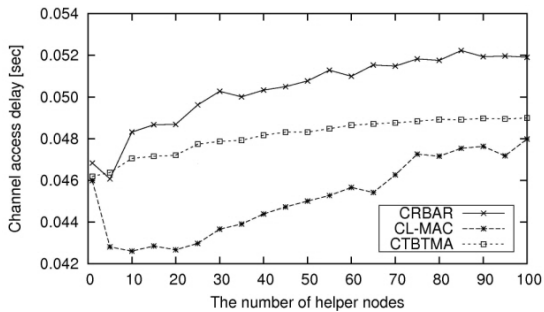


그림 17. 사후 선정기법 평균지연시간 성능비교

하기 때문에 식 (1)과 식 (2)에 의해서 도출된 값도 4종류만 존재한다. 따라서 도움노드 개수가 증가하면 동일한 Utility 값이나, EPTR 값을 갖는 후보 도움노드 개수가 증가하여 충돌확률이 증가한다.

그림 16은 도움노드의 개수가 10개인 경우, 송신노드 개수 변화에 따른 세 가지 도움노드 기법과 협력통신을 사용하지 않는 경우의 시스템 처리량 크기를 서로 비교하여 나타내었다. 성능결과에 따르면 참고문헌^[10]에서 제안한 CL-MAC 기법의 성능이 다른 방법에 비해 크게 우수하며, CRBAR 기법은 협력통신을 사용하지 않는 경우와 비교하여 성능차이가 크지 않음을 알 수 있다.

그림 17에는 송신노드 개수가 5개이고, CRBAR 기법은 $\delta=0.06$ 인 경우, 도움노드 개수 변화에 따른 도움노드를 선정하는 세 가지 기법의 채널 액세스 지연시간의 변화를 서로 비교하여 나타내었다. 이 결과는 그림 8, 11, 14의 결과에서 송신노드 개수가 5개인 경우를 발췌한 것으로, 두 결과가 서로 다르게 보이는 이유는 그림 17의 y-축(채널 액세스 지연시간)의 변화 폭이 그림 8, 11, 14의 y-축 변화 폭보다 훨씬 작아서 상대적으로 상세하게 나타내었기 때문이다. 한편 그림 17결과는 그림 15를 x-축을 기준으로 180도 회전한 것과 유사한 패턴으로 나타나는

데, 이것은 시스템 처리량과 채널 액세스 지연시간의 크기가 서로 반비례 관계가 있기 때문이다.

5. 결론

협력통신에서 적절한 도움노드를 찾아내어 선정하는 것은 매우 중요하다. 도움노드를 선정하는 기법은 시점에 따라 사전선정 기법과 사후선정 기법으로 분류된다. 본 논문에서는 사후선정 기법으로 분류되는 관련 연구 세 가지를 컴퓨터 모의실험을 통해 성능평가한 뒤 세 가지 기법의 성능결과를 서로 비교하고 장단점을 분석하였다. 성능평가 척도로는 시스템 처리량과 채널 액세스 지연시간을 사용하였으며, 트래픽 모델로는 모든 송신노드가 전송할 패킷을 항상 보유하고 있는 Saturated 모델을 사용하였다. 또한 송신노드를 포함하여 수신노드와 후보 도움노드들은 지정된 통신영역 내에서 상호 독립적으로 움직이는 Random waypoint 모델을 사용하였다. 성능평가 결과에 따르면 참고문헌^[10]에서 제안한 Cross-layer MAC 기법의 성능이 제일 우수하였는데, 이는 2단계 도움노드 선정 절차 외에 K개 미니슬롯을 사용하여 또 한 번 충돌 해소 절차를 거치도록 하여 도움노드를 가장 잘 선정했기 때문인 것으로 판단된다.

참고 문헌

1. William C. Y. Lee (1995), *Mobile cellular telecommunications*, 2nd Edition, McGraw-Hill press.
2. Wookbong Lee, Jin Sam Kwak, Bin-Cheul Ihm, and Shuangfeng Han (2012), "Multi-BS MIMO cooperation: challenges and practical solutions in 4G systems", *IEEE Wireless Commun. Magazine*, Vol. 19, No. 1, pp. 89-96.
3. A. Nosratinia, T. E. Hunter, and A. Hedayat (2004), "Cooperative communication in wireless networks", *IEEE Commun. Magazine*, Vol. 42, No. 10, pp. 74-89.
4. IEEE Std 802.11 (2007), *Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE Computer Society.
5. A. Kamerman and L. Monteban (1997), "WaveLAN II: A high-performance wireless LAN for the unlicensed band", *Bell Labs Tech. Journal*, Vol. 2, pp. 118-133.
6. G. Holland, N. Vaidya, and P. Bahl (2001), "A rate adaptive MAC protocol for multi-hop wireless networks", *Proc. of ACM/IEEE MOBICOM*, Italy, pp. 236-251.
7. H. Zhu and G. Cao (2006), "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.
8. P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar (2007), “CoopMAC: A cooperative MAC for wireless LANs”, *IEEE J. of Selected Areas on Commun.*, Vol. 25, No. 2, pp. 340-353.
 9. H. Shan, et al. (2008), “Cross-layer cooperative triple busy tone multiple access for wireless networks”, *Proc. of IEEE GLOBECOM*, New Orleans, pp. 1-5.
 10. H. Shan (2011), “Cross-layer cooperative MAC protocol in distributed wireless networks”, *IEEE Trans. on Wireless Commun.*, Vol. 10, No. 8, pp. 2603-2615.
 11. T. Guo, R. Carrasco (2009), “CRBAR: Cooperative relay based auto-rate MAC for multirate wireless networks”, *IEEE Trans. on Wireless Commun.*, Vol. 8, No. 12, pp. 5938-47.
 12. M. H. MacDougall (1992), *Simulating computer systems: Techniques and tools*, The MIT Press.



장 재 신 (icjoseph@inje.ac.kr)

1990 동아대학교 전자공학과 학사
 1992 한국과학기술원 전기및전자공학과 석사
 1998 한국과학기술원 전기및전자공학과 공학박사
 1997~2002 (주)삼성전자 정보통신총괄 네트워크사업부 책임연구원
 2008~2009 Iowa 주립대 방문연구원
 2002~현재 인제대학교 정보통신공학과 부교수

관심분야 : 이동통신 네트워크, 무선인터넷, 무선 LAN, 애드혹 네트워크