

## 무선랜 성능향상을 위한 AP의 동적 전송속도 할당 기법

박병창<sup>0 1</sup>, 우희경<sup>2</sup>, 김종권<sup>1</sup>

<sup>1</sup> 서울대학교 전기·컴퓨터공학부

{bcpark<sup>0</sup>, ckim}@popeye.snu.ac.kr

<sup>2</sup> 상명대학교 소프트웨어학부

woohk@smu.ac.kr

### Dynamic Rate Adaptation of Centralized AP for WLAN Performance Enhancement

Byoungchang Park<sup>0 1</sup>, Heekyoung Woo<sup>2</sup> and Chongkwon Kim<sup>1</sup>

<sup>1</sup> School of Electrical Engineering & Computer Science, Seoul National University

<sup>2</sup> College of Computer Software and Media Technology, Sangmyung University

#### 요 약

최근 유비쿼터스 네트워크 환경을 구축하기 위해 많은 무선 통신 기술들이 등장하였다. 그 중 무선랜 기술은 쉬운 설치와 저렴한 비용 덕에 폭발적인 보급이 이루어 졌다. IEEE 802.11 표준을 기반으로 만들어진 무선랜은 여러 개의 전송속도를 지원한다. 대표적으로 802.11b는 1, 2, 5.5, 11Mbps의 4가지 전송속도를 지원한다. 낮은 전송속도로 전송을 하면 패킷 전송시간은 증가하지만 비트 에러율은 감소된다. 각각의 전송속도에 따라 다른 패킷 전송 성공률을 갖게 된다. 한 무선랜 AP에 연결된 노드들은 각자 자신의 처리율을 최대화할 수 있는 전송속도를 결정하여 사용하게 된다. 이러한 상황은 전체 시스템 성능에는 문제를 발생시킬 수 있다.

본 논문에서는 각자의 전송속도를 각자가 결정하는 것이 전체 시스템 성능에 문제를 발생시키는 경우를 보이고, 그것을 해결하기 위해 AP가 각 노드의 전송속도를 결정하는 기법을 제안한다. 나아가 무선랜에서의 문제점인 Performance Anomaly 현상을 해결하고, [3]에서 언급된 Inefficient Equilibria 현상도 막아 본다.

#### 1. 서론

무선랜은 여러 개의 전송속도를 지원한다. 802.11b는 1, 2, 5.5, 11Mbps의 4가지 전송속도를, 802.11g는 6, 9, 12, 18, 24, 36, 48, 54Mbps의 8가지 전송속도를 갖는다. 높은 전송속도는 한 심볼에 더 많은 비트를 실게 하는 효율적인 변조기법에 의해 만들어 진다. 하지만 더 효율적인 변조기법은 인코딩된 데이터의 정확도를 보장하는 능력이 떨어진다. 무선 환경에서는 전송받은 신호에 경로 감쇠, 페이딩, 간섭 등을 고려하여 Signal-to-Noise Ratio(SNR)를 구하게 된다. SNR이 전송받은 패킷의 비트 에러율을 결정하게 된다. SNR이 낮을수록 변조기법이 받은 정보를 복호화하기 어려워진다. 높은 전송속도는 더 조밀한 변조기법을 사용하기 때문에 데이터 전송속도가 높을수록 비트 에러율이 높아진다. 그렇기 때문에 무선랜에서 적절한 전송속도를 결정해 주는 것이 중요한 이슈이다. 적절한 전송속도를 결정하여 비트 에러율을 낮추고 처리율이 최대로 해주는 것이 가장 좋다.

현재까지 여러 전송속도 선택 기법이 제안되었다. 요즘 무선랜카드에서 가장 널리 사용되고 있는 기법은 Auto Rate Fallback(ARF)[4]이다. ARF는 이전 전송 패킷들의 성공과 실패를 기반으로 전송속도를 결정하는 기법이다. 구현하기는 쉬운 반면 채널에 빠르게 적응 못할 뿐 아니라 충돌과 채널 상황을 구별을 못하여 문제를 발생시킨다. 그래서 등장한 것이 Receiver-Based Auto

Rate(RBAR)[5]이다. 수신자가 RTS를 통해 채널을 측정하여 적절한 전송속도를 CTS에 넣어서 전송을 해준다. 데이터는 그 전송속도에 맞춰 전송을 하게 된다. 채널 상황을 빠르고 정확하게 반영하는 반면 표준 패킷 헤더에 수정을 가하여야 하는 문제를 발생시킨다.

무선랜에서 위와 같은 방법들로 전송속도를 결정하는 것은 여러 가지 문제가 발생한다. 문제의 원인은 전송속도를 전송하는 노드 각자가 결정하는 데에 있다. 전송속도를 각자 결정하게 되면 전체 처리율이 떨어지고 빠른 속도를 갖는 노드에 피해를 입을 수도 있게 된다[2]. 빠른 속도의 노드와 느린 속도의 노드가 공존하게 되면 모든 노드의 처리율이 가장 낮은 노드의 처리율과 동일하게 되는 Performance Anomaly 현상을 발생시킨다.

전송속도를 각자 결정하는 것의 또 다른 문제는 [3]에서 언급한 Inefficient Equilibria 현상이다. 각 노드마다 각자 전송속도를 결정하게 되는데, 위의 현상이 발생하는 구간에서는 노드가 자신의 전송속도를 자신에게 알맞은 빠른 속도보다 더 안정적인 느린 속도를 결정하게 된다. 그렇게 되면 [2]에서 언급한 Performance Anomaly 현상이 더 심각하게 나타남을 알 수 있다.

본 논문에서는 위의 현상을 막기 위해, 무선랜 서비스를 받는 노드의 전송속도를 조절해 주는 역할을 AP에 맡기는 기법을 제안한다. AP가 각 서비스 노드로부터 수집한 정보를 바탕으로 각 노드의 전송속도를 결정하여 주고 결정해 준 전송속도 외의 전송속도로 전송을 하는 노드가 있으면 서비스를 해주지 않게 하는 것이 핵심 아이디어이다.

2장에서 위에서 언급한 문제점이 발생하는 것을 보이

고, 그것을 해결하기 위해 AP의 동적 전송속도 할당기법을 제안하겠다. 3장에서는 ns-2를 이용한 시뮬레이션 결과를 보이도록 하겠다. 마지막 4장에서는 결론과 함께 향후 연구 계획을 정리하도록 하겠다.

2. 무선랜 성능향상을 위한 AP의 동적 전송속도 할당 기법

2.1. 각 노드별로 결정한 속도의 문제점

노드의 속도 별 성공확률을 계산하여 서론에서 언급한 전체 성능을 떨어뜨릴 수 있는 경우를 표로 만들어 보았다.

표 1. 노드 2개가 서비스 받는 경우

	노드 1의 성공확률	노드 2의 성공확률
11Mbps	0.4 (11*0.4 = 4.4)	1 (11*1 = 11)
5.5Mbps	0.9 (5.5*0.9 = 4.95)	1 (5.5*1 = 5.5)

위의 표 1을 통해 각자의 시간 사용량(기회가 같으므로, 두 노드가 각각 하나의 패킷을 전송하는 시간의 합분의 각 노드가 하나를 전송하는 시간)을 고려해서 각 노드의 처리율을 계산해 보면 아래의 표 2와 같다. 예를 들어 노드 1은 5.5Mbps를 노드 2는 11Mbps를 선택했다고 하면 처리율은 계산은 아래와 같다.

노드 1: 5.5Mbps \* 0.9 \* 2/3 = 3.3Mbps  
 노드 2: 11Mbps \* 1 \* 1/3 = 3.7Mbps

표 2. 표 1에 따른 처리율 계산표

노드1 \ 노드2	11Mbps	5.5Mbps
	11Mbps	노드1 + 노드2 =2.2 + 5.5=7.7
5.5Mbps	노드1 + 노드2 =3.3 + 3.67=6.97	노드1 + 노드2 =2.48 + 2.75=5.23

표 1을 보면 노드 1은 5.5Mbps를, 노드 2는 11Mbps를 선택하는 것이 자신에게 유리하다 판단을 하여 그렇게 선택을 하게 된다. 하지만 표 2를 보게 되면 둘 다 11Mbps를 선택하는 것이 전체 처리율면에서는 더 좋은 것을 볼 수 있다.

2개의 노드가 아니라 4개의 노드를 통해서도 그러한 예를 들 수 있다.

표 3. 노드 4개가 서비스 받는 경우

	노드 1의 성공확률	노드 2의 성공확률	노드 3의 성공확률	노드4의 성공확률
11Mbps	0.2	1	1	1
5.5Mbps	0.7	1	1	1

표 3에서 노드 1, 2, 3, 4의 속도가 각각 (11, 11, 11, 11)인 경우는 처리율이 각각 (0.55, 2.75, 2.75, 2.75)가 되어 전체 처리율은 0.55+2.75+2.75+2.75=8.80이 되게 된다. 속도가 각각 (5.5, 11, 11, 11)인 경우는 속도가 각각 (1.54, 2.2, 2.2, 2.2)가 되어 전체 처리율은 1.54+2.2+2.2+2.2=8.14가 된다. 결국 노드 1은 5.5Mbps를 선택하는 것보다 11Mbps를 선택하는 것이 전체 처리율을 증가 시키게 된다.

2.2. AP의 동적 전송속도 할당 기법 (DRACA): Dynamic Rate Adaptation of Centralized AP

2.1.에서 확인한 결과, 각자가 선택한 전송 속도가 전체적으로 최적이지 아닌 경우를 발생시키게 된다. 이러한 문제점을 해결하고 나아가 Performance Anomaly와 Inefficient Equilibria를 해결할 수 있는 방법을 제안해 보도록 하겠다.

위의 문제점을 해결하기 위해서는 AP에서 2.1.에서 고려해본 값들을 모두 알고 있어야 한다. 그러기 위해서는 AP에서 테이블을 유지해야 한다. AP는 노드로부터 받는 패킷으로부터 아래의 표4와 같은 테이블을 유지하게 한다. 802.11b를 예시로 사용하였다.

표 4. AP에서 유지하는 노드마다의 전송 성공률 테이블

	BPSK(1M)	QPSK(2M)	CCK(5.5M)	CCK(11M)
노드 1	1	1	1	0.6
노드 n	0.9	0.2	0	0

위의 테이블을 바탕으로 2.1.에서 고려했던 식을 바탕으로 각 노드의 전송속도를 결정하게 된다.

전송속도 정보는 ACK 등의 AP가 전송하는 데이터에 함께 넣어서 전송을 하여 각 노드는 자신의 전송속도를 정하게 된다.

AP는 위의 테이블과 함께 각 노드의 결정된 전송속도 테이블을 갖고 있어 테이블과 다르게 전송을 하는 노드가 발생을 하면 그 노드에는 서비스를 해주지 않게 된다.

위의 방법대로 해주게 되면 Performance Anomaly 현상과 Inefficient Equilibria 현상을 막을 수 있게 된다.

3. 성능평가

ns-2 시뮬레이터를 이용하여 시뮬레이션을 하였다. 우선 802.11b의 네 가지 변조기법에 따른 변화를 적용을 해주었다. SNR 값에 따른 비트 에러율은 Intersil 칩셋 [6]의 정보를 이용하여 구하였다.

시뮬레이션 환경은 그림 1과 같다. AP가 존재하고 AP 가까이 좋은 채널 환경을 갖는 노드가 존재한다. 다른 노드는 AP와 같은 위치에 있다가 거리가 멀어지면서 UDP 패킷을 전송하게 된다. 2개의 전송노드는 보낼 패킷이 계속해서 발생한다.

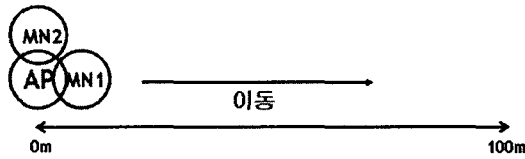


그림 1. 시뮬레이션 환경

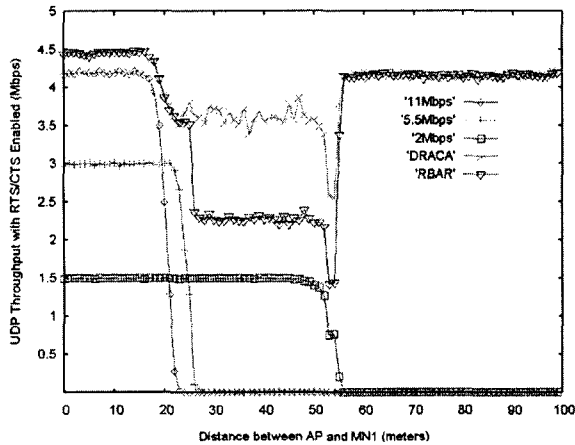


그림 2. 시뮬레이션 결과

간단한 시뮬레이션을 위해 RTS/CTS를 사용한다. RTS를 통해 쉽게 테이블을 만들 수 있으며, RTS/CTS를 쓰는 RBAR과 더 쉽게 비교를 할 수 있다. RTS를 받아서 테이블의 전송성공률을 업데이트 해주고 CTS를 통해 정해진 전송속도를 노드들에게 알려 준다.

RBAR과 비교한 결과는 그림 2에 주어진다. 11, 5.5, 2Mbps의 그래프는 AP와 한 개의 노드가 낼 수 있는 거리에 따른 처리율을 그려본 것이다. 제안한 기법은 영문 앞 글자를 따서 DRACA라고 해주었다. 그림 2에서 보듯이 DRACA가 RBAR보다 높은 성능을 내는 구간이 존재함을 알 수 있다.

#### 4. 결론 및 향후 연구계획

이 논문에서는 지금까지 제안된 무선랜의 전송속도 결정 기법들이 각자의 최적 전송속도를 결정해 주기 때문에 전체 시스템 성능을 떨어뜨릴 수 있다는 것을 보였다. 이 문제를 해결하기 위해 AP의 동적 전송속도 할당 기법을 제안했다. 현재 자신에게 유리하다고 생각되는 전송속도보다 빠른 전송속도를 결정할 수 있게 되어 Performance Anomaly 현상을 경감시킬 수 있고, 나아가 AP가 선택해 준 전송속도를 쓰지 않는 노드는 서비스를 해주지 않게 하여 Inefficient Equilibria 현상을 막을 수 있게 된다.

제안한 기법이 한 AP에 여러 노드가 전송을 경쟁할 시에 좀 더 나은 성능을 나타낼 수 있음을 시뮬레이션을 통해 보였다.

앞으로 좀 더 정밀한 전체 처리율을 분석하여, 노드간

의 형평성을 고려하면서 전체 처리율을 증가시키는 방법을 생각해 볼 예정이다. 또한 더욱 다양한 상황에서 시뮬레이션을 진행하여 동적으로 전송속도를 잘 할당하는 것을 보이도록 하겠다.

#### 5. 참고문헌

- [1] IEEE 802.11 WG, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard, 1999
- [2] Martin Heusse, Franck Rousseau, Gilles Berger-Sabbatel, Andrzej Duda, "Performance Anomaly of 802.11b", Infocom 2003
- [3] Godfrey Tan and John Guttag, "The 802.11 MAC Protocol Leads to Inefficient Equilibria", Infocom 2005
- [4] A. Kamerman and L. Monteban, "WaveLAN-II: A High-performance wireless LAN for the unlicensed band", Bell Lab Technical Journal, pages 118-133, Summer 1997
- [5] Gavin Holland, Nitin H. Vaidya, and Paramvir Bahl, "A rate-adaptive MAC protocol for multi-hop wireless networks", In Mobile Computing and Networking, pages 236-251, 2001.
- [6] Intersil, HFA3861B: Direct Sequence Spread Spectrum Baseband Processor 2000.
- [7] ns-2. The Network Simulator, < <http://www.isi.edu/nsnam/ns/> >