

WPAN 환경에서 AFH 알고리즘을 사용하는 블루투스 피코넷의 패킷 간섭과 주파수 충돌 회피를 위한 적응적 Frequency Hopping Algorithm

준회원 김 승 연*, 정회원 이 형 우*, 조 충 호**

Packet Interference of Bluetooth Piconet Using an Adaptive Frequency Hopping and Advanced Adaptive Frequency Hopping Algorithm for Frequency Collision Avoidance in WPANs

Seung-Yeon Kim* Associate Member, Hyong-Yoo Lee*, Choong-Ho Cho** Regular Members

요 약

본 논문에서는 AFH 알고리즘을 사용하는 블루투스로 이루어진 피코넷이 WLAN과 공존할 때 피코넷 패킷 간의 충돌을 다중 경로 페이딩 채널을 고려하여 수학적으로 분석하였다. AFH에 의해 줄어든 블루투스의 홉 수와 패킷 주변의 피코넷의 신호 세기에 따라 패킷의 충돌 확률에 영향을 줄 수 있음을 알 수 있다. 또한 본 논문에서는 AFH이 갖는 한계를 인식하고 이를 극복하고자 AAFH 알고리즘을 제안하였다. AAFH은 지속적인 채널 사용 여부에 따라 채널 상태의 확인 여부(scanning)를 결정하게 되는데 AFH 와 비교했을 때 collision와 waste 양에서 큰 차이를 보이지 않으면서도 채널 scanning 수가 최대 50% 감소하게 되어 블루투스의 에너지 낭비가 줄어들게 됨을 보였다.

Key Words : Bluetooth, Collision Avoidance, AFH, WPAN, Rician Fading Channel

ABSTRACT

In this paper, we present an analysis of the throughput when there are multiple piconets and WLAN sharing the ISM bands. The analysis takes channel propagation characteristics and the capture effect. We also propose an algorithm which can be used to reduce the amount of channel scanning. By using traffic prediction of the interfering WLAN, we are able to maintain a reasonable performance in terms of fraction of time channel is wasted due to collisions or unused channel. Through computer simulation, we demonstrate that the proposed algorithm achieves reduced scanning frequency.

I. 서 론

최근의 무선 네트워크 기술 발전으로 언제 어디서나 정보를 무제한적으로 끊임 없이 액세스할 수

있게 될 것으로 전망됨에 따라 애플리케이션 및 사용 방법은 극적으로 늘어나고 있다. 가장 널리 보급된 무선 기술 두 가지는 블루투스와 와이파이로 더 잘 알려진 무선 랜이다. 블루투스는 단거리의 데이

* 본 연구는 산업자원부 및 한국산업기술평가원의 성장동력기술개발사업의 연구결과로 수행되었습니다.

* 고려대학교 전자및정보공학부 B-ISDN 연구실 (kimsy8011@korea.ac.kr, hwlee@korea.ac.kr)

** 고려대학교 컴퓨터정보학과 BCN 연구실 (chcho@korea.ac.kr)

논문번호 : KICS2007-06-283, 접수일자 : 2007년 6월 27일, 최종논문접수일자 : 2007년 9월 7일

터 전송을 가능케 해주는 무선 PAN기술로서, 예컨대 컴퓨터 주변기기를 위한 USB 케이블을 대체할 수 있도록 해주고, PDA와 데이터를 동기화 하며, wireless head-set을 구현하는데 사용된다. 이에 비해 WLAN은 Ethernet 케이블 대체 기술 역할을 하여 무선 네트워킹, 인터넷 액세스 그리고 데이터, 비디오 및 VoIP를 위해 보다 장거리 상에서 보다 빠른 데이터 전송을 가능케 해준다. 블루투스와 802.11b/g WLAN 시스템들은 2.4GHz 대역에서 동작하며^[1], 두 기술 모두 가용 스펙트럼 가운데 상당 부분을 사용한다. 블루투스와 WLAN 디바이스가 매우 근접해 있으면서 무선 신호들을 동시에 송수신하려 할 때 간섭이 발생한다^[2]. 블루투스 장치들이 간섭을 감지하여 피할 수 있도록 하기 위한 방법으로서 AFH^[3]가 있다. 본 논문에서는 멀티플 슬롯 패킷을 사용하는 피코넷 클러스터의 패킷 충돌을 다중 경로 채널에서 모델링하여 AFH 알고리즘을 사용하는 블루투스로 이루어진 피코넷의 패킷 충돌 확률과 통합 처리량을 분석한다. 그리고 기존의 AFH 알고리즘이 갖는 한계를 인식하고 이를 극복하고자 가정을 통해 AAFH 알고리즘을 제안한다. AAFH은 지속적인 채널 사용 여부에 따라 채널 상태의 확인 여부(scanning)를 결정하게 되는데 AFH와 비교했을 때 collision와 waste 양에서 큰 차이를 보이지 않으면서도 채널 scanning 수가 최대 50% 감소하게 되어 블루투스의 에너지 낭비가 줄어들게 됨을 보였다.

II. 관련 연구

2.1 Adaptive Frequency Hopping 알고리즘^[3]

일반적으로 블루투스 전송 채널은 625 μs 슬롯 79개로 나눈다. 그리고 마스터의 패킷 전송은 슬롯의 홀수 번째에 이루어진다. 각각의 패킷은 최대 주파수 호핑률 (Maximum Frequency Hopping Rate) 1600hops/s로 다른 호핑 주파수로 전송된다. AFH 알고리즘 적용 블루투스의 경우 2.402-2.480GHz 주파수 범위에서 짝수 주파수와 홀수 주파수로 분류한다. 마스터가 전송할 수 있는 홀수 주파수만 본다면 분류된 주파수에서 처음 32개의 주파수를 선택한다. 그리고 선택된 주파수에서 16개의 주파수를 선택한다. 처음 32개의 주파수를 선택하고 남아있던 주파수에서도 16개를 선택하여 총 32개의 주파수를 결정한다. 선택된 32개의 주파수에는 Good 주파수

와 Bad 주파수가 있는데 이때 그림 1에서 보는 것처럼 AFH 알고리즘에 의해 Bad 주파수를 Good 주파수로 대체하게 된다. AFH 알고리즘과 원래의 블루투스 호핑 세그먼트 알고리즘 (Bluetooth Hopping Segment Algorithm) 과 차이는 선택된 32개의 채널을 Good 주파수로 채운다는 것이다. AFH 알고리즘을 적용한 블루투스를 사용할 때 Bad 주파수가 많을 경우 그 세그먼트 안에 Good 주파수 패턴의 변화는 마스터와 슬레이브 간의 LMP (Link Management Protocol) 메시지의 교환으로 이루어진다.

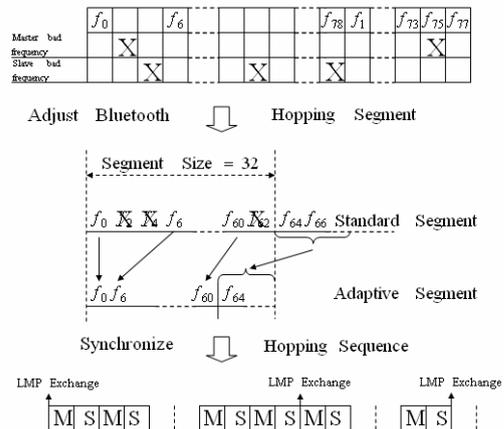


그림 1. Adaptive Frequency Hopping

2.2 다중 경로 페이딩 채널^[4,5]

다중 경로 페이딩은 그림 2와 같이 장애물 때문에 라디오 웨이브의 반사에 의해 발생하게 된다. 그래서 라디오 웨이브는 송, 수신시에 여러 경로로 전송된다. 이와 같은 페이딩의 진폭 분포는 레일리 분포로 모델링 된다. 레일리 분포는 식 (1)과 같다.

$$f_{Ray}(x) = \frac{x}{\sigma_l^2} \exp\left(-\frac{x^2}{2\sigma_l^2}\right) \quad (1)$$

x 는 신호의 진폭이고, σ_l^2 는 주변 단말기들의 평균 파워이다.

송, 수신기 사이의 거리가 짧고 주변 환경의 변화가 적을 경우 전송된 신호 파워의 많은 부분이 다중 경로이지만 line-of-sight (LOS)가 존재하게 된다. 이때 많이 사용하는 라이시안 페이딩 채널에서는 전송된 신호의 크기가 라이시안 랜덤 변수에 의해 모델링 된다. 라이시안 분포는 식 (2)와 같다.

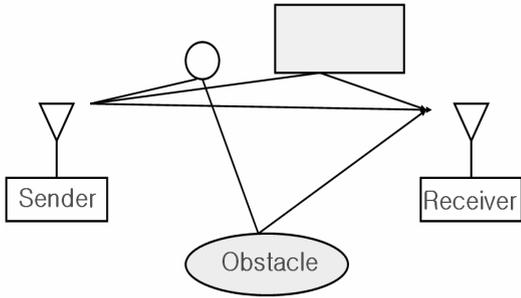


그림 2. Multi-path fading in a wireless transmission

$$f_{Rice}(x) = \frac{x}{\sigma^2} \exp\left(-\frac{(x^2 + \alpha_0^2)}{2\sigma^2}\right) I_0\left(\frac{\alpha_0 x}{\sigma^2}\right), \quad x \geq 0 \quad (2)$$

$$I_0(x) = \frac{1}{2\pi} \int_0^{2\pi} \exp(x \cos \theta) d\theta \quad (3)$$

여기서 $I_0(\cdot)$ 은 0차의 Bessel function이다. α_0^2 은 직접적인 수신 또는 신호 파워의 LOS를 나타낸다. 라이시안 페이딩 채널에서 중요한 파라미터 K 는 식 (4)와 같이 정의된다.

$$K = \frac{\alpha_0^2}{2\sigma^2} \quad (4)$$

식(4)에서 $2\sigma^2$ 는 피코넷의 파워이다. K 가 0에 접근할 경우 레일리 페이딩 채널 (Rayleigh Fading Channel)이 되고, K 가 ∞ 에 접근할 경우 Additive White Gaussian Noise (AWGN) 채널이 된다.

III. 채널을 고려한 충돌 확률

3.1 충돌 확률⁽⁶⁾

한 개의 피코넷이 갖는 주파수 홉 수는 79개이다. 그림 3에서 1슬롯을 갖는 master1(M1)이 3슬롯을 갖는 master2(M2)에 의해 충돌이 발생할 확률(P_{13})은 master1가 f1의 주파수 동안 master2의 주파수와 같을 확률 $2/79$ 이다. master1가 f2의 주파수 동안 master2의 주파수와 충돌할 확률은 $1/79$ 이다. 같은 과정으로 f4일 경우는 $1/79$, f5일 경우는 $2/79$ 이다.

반대로 master2가 master1에 의해 충돌이 발생할 확률(P_{31})은 master2가 f1'의 주파수 동안 master1의 주파수와 같을 확률은 $4/79$ 이다. master2가 f2' 주파수 동안 master1의 주파수와 같을 확률은 $2/79$ 이다. 두 경우의 평균을 구하면 $(4/79+1/79)/2=3/79$ 가 된다.

앞에서 P_{13} , P_{31} 을 구한 것과 같은 방식으로 그림

4 의 P_{35} , P_{53} 을 구하면 각각 $5/237$, $5/158$ 가 된다. 나머지 다른 슬롯을 갖는 경우의 확률을 구하면 $P_{11}=2/79$, $P_{33}=2/79$, $P_{55}=2/79$, $P_{15}=4/237$, $P_{51}= 4/79$ 가 된다.

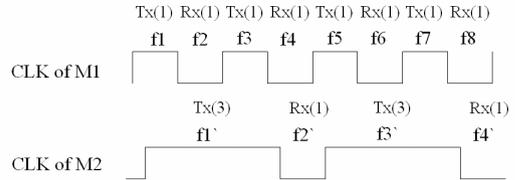


그림 3. 비동기 비대칭 피코넷의 간섭(1- and 3-slot packets)

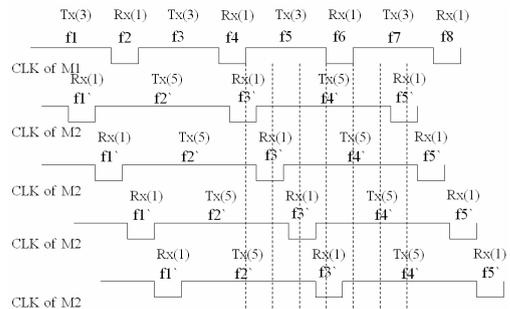


그림 4. 비동기 비대칭 피코넷의 간섭(3- and 5-slot packets)

3.2 모델링

협소한 공간 안에 다수의 피코넷이 존재할 때 각각의 피코넷이 가질 수 있는 홉의 수는 79개이다. 주변에 존재하는 피코넷이 모두 대칭(Symmetric) 동기식 (Synchronous) 일 경우 확률은 $1/79$ 가 되는데 이 확률 값은 앞장에서 구한 각각의 값처럼 비대칭(Asymmetric) 비 동기식 (Asynchronous) 의 경우 달라진다. N 개의 피코넷이 존재하고, $N-1$ 개의 주변 피코넷 중 n 개의 피코넷이 같은 주파수를 사용할 확률 $Pr[n]$ 은 식 (5)와 같다.

$$Pr[n] = \binom{N-1}{n} \cdot P^n \cdot (1-P)^{N-1-n}, \quad n = 0, 1, 2, \dots, N-1 \quad (5)$$

P 는 피코넷의 패킷 전송 방식에 따라 앞장에서 구한 확률 값으로 결정된다.

테스트 피코넷은 라이시안 채널이고, 주변 피코넷이 레일리 채널인 경우 처리량 S 는 식 (6), (7), (8), (9)을 통해 식(10)과 같다.

$$S = 1 - \left[\sum_{n=1}^{N-1} \Pr[n] F(z_0) \right] \quad (6)$$

$$F(z_0) = \Pr[P_s/P_n < z_0] \\ = \int_0^{z_0} dz \int_0^\infty f_{P_s}(zw) f_{P_n}(w) w dw \quad (7)$$

z_0 는 테스트 피코넷과 주변 피코넷과의 비율에 대한 경계(Threshold) 값으로 정의하였다.

$$f_{P_s}(P_s) = \frac{1}{\sigma^2} \exp\left[-\frac{2P_s + S_t^2}{2\sigma^2}\right] \\ \times I_0\left[\frac{\sqrt{(2P_s)S_t}}{\sigma^2}\right], P_s = (1/2)x^2 \quad (8)$$

S_t 은 테스트 피코넷의 파워이다.

$$f_{P_n}(P_n) = \frac{1}{\sigma^2} \frac{(P_n/\sigma^2)^{n-1}}{(n-1)!} \exp\left[-\frac{P_n}{\sigma^2}\right] \quad (9)$$

다음 식 $f_{P_n}(P_n) = 1/\sigma^2 \exp[-P_n/\sigma^2]$ 의 n 중 콘볼루션(convolving n time)와 Gamma distribution에 의해 식 (9)가 된다.

$$S = (1 - \sum_{n=1}^{N-1} \Pr[n] (1 - \int_0^\infty \frac{t^{n-1}}{(n-1)!} \\ \times \exp(-t) Q[\vartheta, \sqrt{(2Mt)}] dt)) \quad (10)$$

여기에서 $\vartheta \cong (2K)^{1/2}$, $M \cong z_0 \frac{\sigma^2}{\sigma^2}$ 이고 $Q(a, b)$ 는 Marcum Qfunction을 말한다⁷⁾.

테스트 피코넷과 주변 피코넷 모두가 라이시안 채널인 경우 처리량 S 는 식 (6), (7), (8), (11)을 통해서 식 (12)와 같다.

$$f_{P_n}(P_n) = \frac{1}{2\sigma^2} \left(\frac{P_n}{S_n}\right)^{(n-1)/2} \\ \times \exp\left[-\frac{P_n + S_n}{2\sigma^2}\right] I_{n-1}\left[\frac{\sqrt{P_n S_n}}{\sigma^2}\right], \\ S_n = \sum_{i=1}^n S_i^2 \quad (11)$$

S_i 는 직접 받게 된 주변 신호의 피크(peak) 값이다.

$$S = \sum_{n=1}^N \Pr[n] \left(1 - \frac{1}{\vartheta^{n-1}} \times \exp\left(-\frac{\vartheta^2}{2}\right)\right) \\ \times \int_0^\infty x^n \exp\left(-\frac{x^2}{2}\right) I_{n-1}[\vartheta \sqrt{x}] \\ \times Q\left[\alpha, \sqrt{z_0} \cdot \frac{\sigma}{\sigma} x\right] dx \quad (12)$$

$$\text{식에서 } K_d = S_t^2/2\sigma^2, K_u = S_n^2/2\sigma^2, \\ \vartheta_1 = \sqrt{2nK_n} \text{ 그리고 } \alpha = \sqrt{2K_d} \text{이다.}$$

표 1은 피코넷 클러스터의 통합 처리량을 계산하기 위해 사용하였다.

표 1. condition of calculating aggregated throughput

Slot	Total duration	Payload
1-slot	625 μ s	240 bits (30bytes)
3-slot	1875 (3 \times 625) μ s	1464 bits (183bytes)
5-slot	3125 (5 \times 625) μ s	2712 bits (339bytes)

통합 처리량(Aggregated Throughput)을 피코넷의 통합처리량 S_{th_put} 은 식 (13)와 같다.

$$S_{th_put} = N \times S \times \text{payload bits/sec} \quad (13)$$

본 논문에서는 현실성을 고려하여 비대칭 비동기 식만을 적용하였다. 각각의 피코넷은 항상 패킷 전송이 이루어지고 있다고 가정하였다. 1슬롯, 3슬롯, 5슬롯 각각 전체의 1/3로 하였다. 레일리 채널과 라이시안 채널이 존재하는 경우에는 $K = 4dB$, $\sigma^2/\sigma^2 = 2dB$ 를 적용하였다.

3.3 분석 결과

그림 5와 그림 6은 테스트 피코넷은 라이시안 채널, 주변 피코넷들은 레일리 채널일 경우, 한 클러스터의 홉 수를 바꾸면서 전체 피코넷 수가 증가할 때 패킷 충돌 확률과 통합 처리량을 나타낸다. 피코넷의 개수가 증가함에 따라 충돌 확률이 증가할 뿐 아니라 홉 수가 감소함에 따라 충돌확률 역시 증가함을 볼 수 있다. 그리고 전체 피코넷의 수가 증가

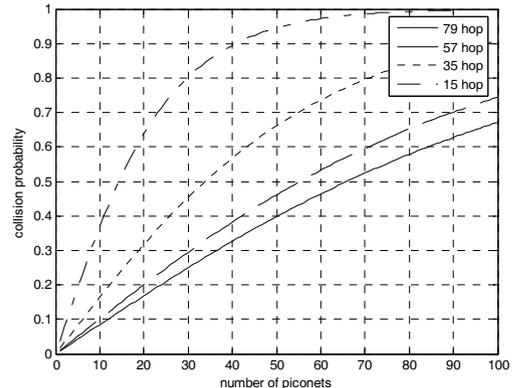


그림 5. 테스트 피코넷은 라이시안, 주변 피코넷은 레일 분포일 때 충돌 확률 ($z_0 = 4dB$)

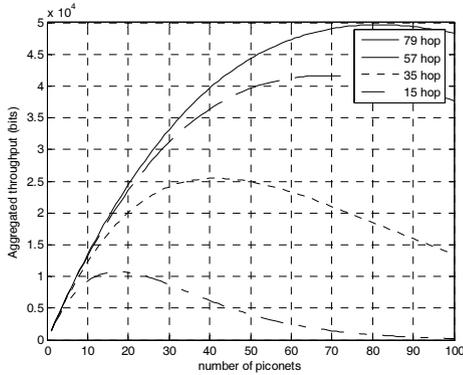


그림 6. 테스트 피코넷은 라이시안, 주변 피코넷은 레일 분포일 때 통합 처리량 ($z_0 = 4\text{dB}$)

해도 어느 이상의 피코넷 수가 존재하면 통합처리량이 떨어짐을 볼 수 있다. 채널이 고려되지 않은 상태^[8]의 결과에 비해 채널 상태에 따라 처리량이 올라감을 볼 수 있다. 또한 홉 수가 줄어들수록 최대 통합 처리량도 줄어들음을 볼 수 있다. 다시 말해 AFH 알고리즘에 의해 줄어든 홉 수를 사용하게 되면 WLAN과의 패킷 충돌은 막을 수 있으나 블루투스 피코넷간의 패킷 충돌이 발생하게 되고 처리량에 영향을 줄 수 있다.

IV. AAFH 알고리즘

4.1 용어정의

AAFH 알고리즘의 이해를 위해 용어를 정의한다. 그림 7은 용어를 나타내고 있다.

- **Scanning:** 채널을 사용하는지 여부를 확인하여 호핑 패턴을 결정하는 과정.
- **Collision:** Scanning을 통해 LAN이 사용하지 않는 것으로 호핑 패턴을 결정했으나 LAN이 사용함으로써 생기는 충돌.
- **Waste:** Scanning을 통해 LAN이 사용하지 않는 것으로 호핑 패턴을 결정했으나 LAN이 사용함으로써 생기는 낭비.
- **Estimation:** 알고리즘을 통해 다음번의 Scanning의 여부를 결정하는 것.
- **Prediction data:** Estimation을 통해 얻은 data.
- **Real data:** 실제 인터넷 데이터를 일정 시간마다 Sampling 해서 만든 data.
- **Event:** LAN traffic이 계속 존재하다 없어지는 시점, 계속 없다가 생기는 시점.
- **Without-AFH:** LAN과 블루투스간의 충돌회피를

위한 알고리즘이 적용되지 않은 상태.

- **AAFH:** Prediction data를 가지고 event가 발생하는 시점을 고려해서 AFH을 해서 얻은 결과.

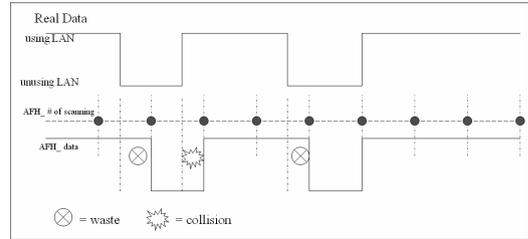


그림 7. 용어 정의

4.2 AFH 알고리즘과 AAFH 알고리즘의 차이점

AFH 알고리즘은 2장에서 설명한 것과 같이 고정된 시간 마다 채널 상태를 확인하여 채널의 사용 여부를 가지고 호핑 패턴을 결정하게 된다. 그런데 채널이 계속 사용 중이거나 계속 사용하지 않더라도 고정된 시간에 따라 채널 상태를 확인하게 되어 필요하지 않은 작업을 수행하게 된다. 블루투스의 에너지 측면에서 본다면 낭비가 되는 것이다. AAFH 알고리즘의 경우 채널의 사용 여부를 확인하여 호핑 패턴을 결정하기 전에 채널의 상태를 예측하여 scanning 여부를 결정한다. 그림 8에서보는 것과 같이 AFH 알고리즘은 채널의 상태를 계속해서 고정된 시간에 확인하게 되지만 AAFH 알고리즘의 경우 앞으로 계속 채널을 사용하거나 사용하지

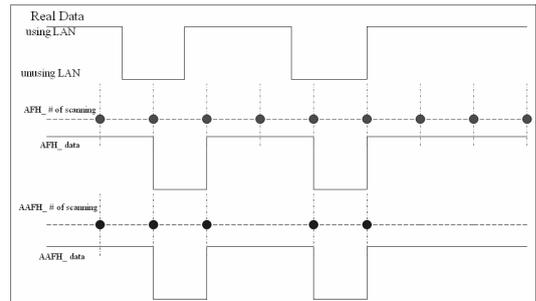


그림 8. AFH and AAFH 의Hopping 패턴 결정

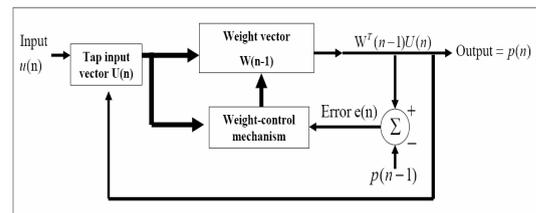


그림 9. Block diagram of the AAFH algorithm
*Tap input vector $U(n)$ 은 표 2의 $U(n)$ 와 $p(n)$ 로 결정된다.

지 않는다면 채널의 상태를 확인하지 않는다. 블루투스의 에너지 측면에서는 그만큼 에너지 낭비가 줄어들게 되는 것이다.

4.3 AAFH 알고리즘 설계

본 알고리즘은 RLS 알고리즘[9]을 참고하였다. 그림 9에서 보는 것과 같이 Output은 input vector $U(n)$ 과 weight vector $W(n)$ 의 곱으로 결정된다. Input vector $U(n)$ 는 알고리즘이 갖는 메모리 길이 (m) 만큼의 data로 만들어지는데 초기 값 $(n=1, 2, \dots, m)$ 은 real data 값을 갖게 된다. 그리고 예측이 이루어지면서 예측 값 $P(n)$ 을 다시 input vector $U(n)$ 값으로 갖게 된다. 또한 주기적으로 real data 값을 다시 input vector $U(n)$ 을 가져오게 하였다. 그래서 input vector $U(n)$ 는 주기적으로 예측 값과 real data 값을 갖게 된다. 이는 예측 값을 계속해서 예측함으로써 0으로 수렴하여 예측이 불가능해지는 현상을 막기 위해서이다. Weight vector $W(n)$ 는 weight control mechanism에 의해서 결정되는데 표 1에서 보는 것과 같이 error $e(n)$ 값과 gain vector $K(n)$ 을 가지고 inverse correlation matrix $P(n)$ 을 결정하게 된다. $P(n)$ 은 다시 input vector $U(n)$ 와 gain vector $K(n)$ 을 결정하게 된다. 표 2에 나오는 threshold는 LAN이 채널을 사용하는지의 여부를 결정하는 기준 값이다.

표 2. Summary of the AAFH Algorithm

Initialize the algorithm by setting	
$W(0) = 0,$	
$P(0) = \delta^{-1}I.$	
For each instant of time, $n=1, 2, \dots,$ compute	
$\pi(n) = P(n-1)U(n),$	
$K(n) = \frac{\pi(n)}{(\lambda + U^T(n)\pi(n-1))}$	$U(n) = \begin{cases} u(n) & : 1 \leq n \leq m, u(n) \text{ real data} \\ p(n) & : m < n \end{cases}$
$e(n) = W^T(n-1)U(n) - p(n-1),$	$m = \text{Prediction filter length}$
$W(n) = W(n-1) + K(n)e(n),$	and
$P(n) = \frac{P(n-1) - K(n)U^T(n)P(n-1)}{\lambda}$	$p(n) = u(n); n = \text{a multiple of } 5$
Prediction	$p(n+1) = u(n+1); \text{ if } u(n) < \text{threshold},$
$p(n) = W(n-1)^T U(n).$	$n = \text{a multiple of } 5$

4.4 성능 분석 결과

분석을 위한 시뮬레이션 환경은 다음과 같다.

- WLAN, 블루투스 피코넷 각각 1개이다.
- 시뮬레이션 data는 실제 측정된 LAN traffic^[10]을 사용하였다.
- 블루투스의 호핑 패턴을 고려하지 않음으로 결과 분석을 쉽게 하였다.
- LAN이 사용하는 대역을 블루투스도 계속 사용하고 있다.

- 알고리즘 메모리(m)은 5로 input 값을 5개 까지 저장하게 된다.
- AAFH 알고리즘의 parameter 값은 $\delta=100, \lambda=0.9995$ 이다.
- 실험 총 data는 200sec 동안 74000(bytes/2.7ms)로 되어 있다.

분석은 실제 측정된 LAN data를 0.0027 sec 에서 10sec로 sampling interval을 조절하여 만든 Real data를 가지고 예측을 하여 sampling interval(sec)이 증가할 때 전체 data(74000개)에서 collision 발생량(%), waste 발생량(%), scanning 수의 관점에서 AFH 알고리즘과 성능을 비교하였다.

그림 10과 그림 11은 sampling interval이 증가할 때 발생하는 collision과 waste를 보여준다. Without-AFH의 경우 블루투스가 LAN이 채널을 사용하지 않는다고 판단하여 호핑 패턴을 결정 한 후 통신을 하기 때문에 sampling interval에 상관없이 32.1%의 collision이 발생하게 되고, waste는 발생하지 않았다. AFH 알고리즘의 경우 sampling interval이 작아질수록 collision과 waste가 줄어든다. Sampling interval이 0.0027sec가 되면 data 간

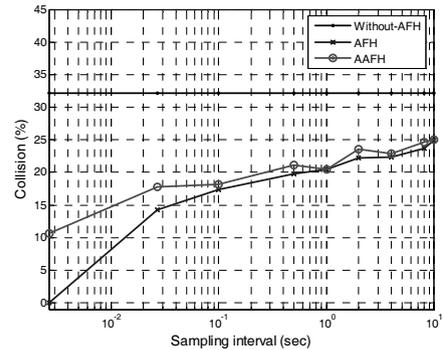


그림 10. sampling interval에 따른 collision

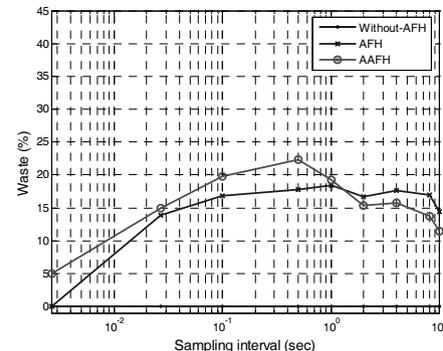


그림 11. sampling interval에 따른 waste

interval이 매우 작기 때문에 collision과 waste가 0이 된다. Sampling interval이 작아서 채널의 변화에 그 만큼 민감하게 반응할 수 있기 때문이다. Sampling interval이 증가할수록 collision과 waste가 증가하는 경향이 있으나 비선형적이다. 그림 12는 AFH 알고리즘의 sampling interval에 따른 collision과 waste의 변화를 보여주는데 interval이 증가할수록 collision과 waste의 증가가 왜 비선형적인지 보여준다. 그림 12에서 보는 것과 같이 sampling interval이 2sec에서 4sec로 증가하면서 waste은 증가하는데 4sec에서 8sec로 증가하면서 2sec 보다는 증가하나 4sec에 비해 감소하게 된다. 하지만 collision의 경우는 계속 증가하게 된다. 그래서 전체적으로는 collision, waste 모두 증가하는 경향을 보이지만 비선형적이다. AAFH 알고리즘의 경우 sampling interval이 증가하면서 collision과 waste가 증가하는 경향을 보이고 있는데 AFH 알고리즘과 마찬가지로 비선형적으로 증가한다. Sampling interval이 작을 때는 AFH 알고리즘과 큰 차이를 보이지만 sampling interval이 증가하면서 그 차이는 줄어든다. 하지만 AFH 알고리즘에 비해 collision과 waste가 증가하여 성능이 떨어짐을 알 수 있다.

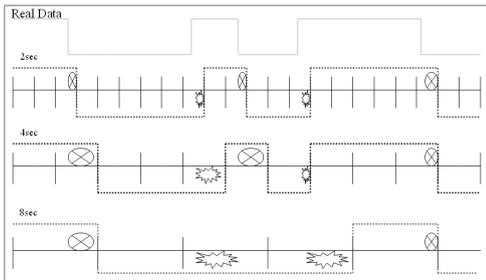


그림 12. AFH의 sampling interval 증가에 대한 collision과 waste

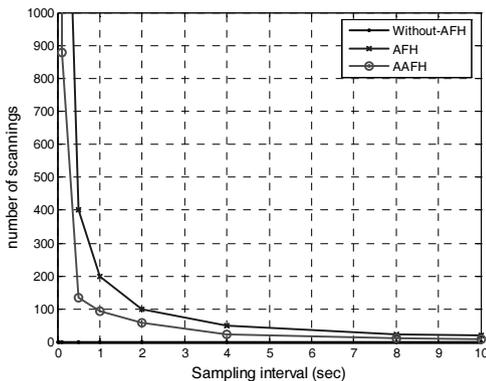


그림 13. sampling interval에 따른 scanning

그림 13은 sampling interval에 따른 블루투스의 scanning 수를 나타내고 있다. Without-AFH의 경우 채널 상태를 확인하지 않기 때문에 scanning 수가 0으로 나타난다. AFH 알고리즘의 경우 채널 scanning time이 고정되어 있기 때문에 sampling interval에 따라 scanning 수도 알 수 있다. 시뮬레이션 총 시간이 200sec로 sampling interval이 10sec 이면 scanning 수는 20번, 4sec 이면 50번, 1sec 이면 100번으로 sampling interval이 감소함에 따라 scanning 수는 증가하게 된다. 시뮬레이션에서는 최소 sampling interval로 0.0027 sec을 사용해서 74000번의 scanning 수를 갖는다. AAFH 알고리즘의 경우도 sampling interval이 증가함에 따라 scanning 수가 감소함을 볼 수 있다. 예측을 통해 scanning 여부를 결정하기 때문에 AFH 알고리즘에 비해서 scanning 수가 줄어들었음을 볼 수 있다. sampling interval이 0.1sec일 때는 3230, 0.5sec일 때는 878번의 scanning 수로 AFH 알고리즘에 비해 최대 50%가 감소한 것을 볼 수 있다. 고정된 scanning time을 갖는 AFH 알고리즘의 경우 LAN traffic이 계속 존재 하여 channel이 계속 busy 상태로 유지되더라도 scanning을 하게 되는데 AAFH 알고리즘은 예측을 통해 scanning을 결정하기 때문에 채널이 idle할 때 scanning을 함으로써 scanning 수가 줄게 됨을 알 수 있다.

V. 결론

본 논문에서는 AFH 알고리즘을 사용하는 블루투스로 이루어진 피코넷이 여러 개가 존재하고 WLAN과 공존하게 될 때 피코넷의 홉 수의 변화와 다중 경로 페이딩 채널을 고려하여 피코넷 패킷 간의 충돌과 통합 처리량을 분석하였다. 다수의 피코넷이 패킷을 전송할 때 동일 주파수를 사용하여 충돌이 발생하더라도 주변의 피코넷의 신호 세기에 따라 패킷의 충돌 확률에 영향을 주게 된다. AFH 알고리즘은 고정된 시간 마다 채널 상태를 확인하여 호핑 패턴을 결정하기 때문에 채널이 계속 사용중이거나 계속 사용하지 않더라도 고정된 시간에 따라 채널 상태를 확인하게 되어 필요하지 않은 작업을 수행하게 된다. 블루투스의 에너지측면에서 낭비가 된다. AAFH 알고리즘의 경우 앞으로 계속 채널을 사용하거나 사용하지 않는다면 채널의 상태를 확인하지 않는다. AFH 알고리즘과 성능을 비교했을 때 collision과 waste 양에서 큰 차이를 보이지

않았지만 scanning 수에서는 최대 50% 줄어들게 됨을 알 수 있다. 블루투스의 에너지 측면에서 낭비가 줄어들게 되는 것이다. 낭비가 줄어든 에너지로 블루투스가 예측할 수 있는 time interval을 줄인다면 채널의 변화에 그만큼 민감하게 반응할 수 있고 WLAN이 사용하는 채널을 사용할 수 있으므로 피코넷 간의 패킷 충돌도 줄일 수 있을 것이다.

참 고 문 헌

- [1] N. GOLMIE, R.E. VAN DYCK, A. SOLTANIAN, A. TONNERRE, O. REBALA, "Interference Evaluation of Bluetooth and IEEE 802.11b Systems", *Wireless Networks*, vol. 9, pp 201-211, 2003.
- [2] A.K. Arumugam, A. Doufexi, A.R. Nix, P.N. Fletcher, "An Investigation of Coexistence of 802.11g WLAN and High Data Rate Bluetooth Enable Consumer Electronic Devices in Indoor Home and Office Environments", *IEEE Transactions on Consumer Electronics*, Vol. 49, No. 3, August 2003.
- [3] Nada Golmie, Nicolas Chevrollier, Oliver Rebal, "Bluetooth and WLAN Coexistence: challenges and solutions", *IEEE Wireless Communication*, December 2003.
- [4] R. Prasad, and C.-Y. Liu, "Throughput analysis of some mobile packet radio protocols in Rician fading channels", *IEE Proceedings*, Vol. 139, No. 3, June 1992.
- [5] S.BENEDETTO, E.BIGLIERI AND V. CASTELLANI, "*digital transmission theory*", *Prentice- Hall*.
- [6] Kshirasagar Naik, David S. L. Wei, Yu T. Su and Norio Shrotori, "Analysis of packet interference and aggregated throughput in a cluster of Bluetooth piconets under different traffic conditions", *IEEE Journal on Selected Areas in Communications*, vol. 23, No. 6, June 2005.
- [7] PROAKS, J. G, *Digital communications*, 2nd Ed, Mc Graw-Hill Book Company, New York, 1989.
- [8] 김승연, 조충호, 이형우, "Adaptive Frequency

Hopping을 사용하는 블루투스 피코넷의 패킷 간섭과 통합처리량 분석", *한국정보처리학회*, 2006. 5월, pp 1159-1162.

- [9] SIMON HAYKIN, *ADAPTIVE FILTER THEORY*, Prentice-Hall, 4TH Ed, 2002, pp 436-465.
- [10] Web site:<http://ita.ee.lbl.gov/html/contrib/BC.html>.

김 승 연 (Seung-Yeon Kim)

준회원



2005년 2월 고려대학교 전자 및 정보공학부 졸업
2007년 2월 고려대학교 전자정보공학과 석사
2007년 3월~현재 고려대학교 전자정보공학과 박사과정
<관심분야> 통신망 설계 및 성능 분석, MAC 프로토콜

이 형 우 (Hyong-Yoo Lee)

정회원



1979년 University of British Columbia Electrical Engineering (학사)
1983년 University of Waterloo, Electrical Engineering (박사)
1983~1991년 Carleton University, System and Computer Engineering 조교수

1992년~1995년 University of Waterloo, Electrical and Computer Engineering 조교수
1995~현재 고려대학교 전자 및 정보공학부 교수
<관심분야> 통신망 설계 및 성능 분석, 트래픽 제어, MAC 프로토콜

조 충 호 (Choong-Ho Cho)

정회원



1981년 2월 고려대학교 산업공학과 졸업
1993년 2월 고려대학교 산업공학과 석사
1986년 프랑스 INSA de Lyon 전산학과(석사)
1989년 프랑스 INSA de Lyon 전산학과(박사)

1990년~1994년 순천향대학교 전산통계학과 조교수
1994년~현재 고려대학교 전산학과 교수
<관심분야> 통신망 트래픽 관리기술, 무선통신 시스템, 멀티미디어통신, 인터넷 비즈니스