

# 무선 애드혹 네트워크에서의 확률적 이웃 탐색 기법

송 태 원\*, 박 현 희\*, 백 상 현<sup>o</sup>

## Probabilistic Neighbor Discovery Algorithm in Wireless Ad Hoc Networks

Taewon Song\*, Hyunhee Park\*, Sangheon Pack<sup>o</sup>

### 요 약

무선 애드혹 네트워크에서 이웃 탐색 과정은 네트워크를 초기화하는데 먼저 수행되어야 하고, 라우팅 알고리즘이나 토폴로지 컨트롤, 그리고 MAC 계층 설계를 위해서도 반드시 필요한 과정이므로 효율적인 분산적 이웃 탐색 방법 설계가 필수적이다. 본 논문에서는 확률적 이웃 탐색 기법 (PND: Probabilistic neighbor discovery)을 제안한다. 제안한 기법에서는 MIMD (Multiplicative-increase, multiplicative-decrease) 정책을 통해 광고 메시지의 전송 확률을 제어함으로써 이웃 탐색에 소요되는 시간을 줄이는 것이 가능하다. 더불어 임의의 기기가 광고 메시지를 전송한 경우, 그 메시지가 성공적으로 전송되었는지 여부를 알 수 있는 충돌 감지 기법 (CD: Collision detection)을 도입함으로써 확률적 이웃 탐색 기법의 성능을 더 높일 수 있다. 시뮬레이션 결과는 제안 기법이 모든 이웃을 탐색하기까지 소요되는 시간을 15.6%~57%까지 감소시킬 수 있음을 보여준다.

**Key Words** : Collision detection, distributed MAC, neighbor discovery, wireless ad hoc networks

### ABSTRACT

In wireless ad hoc networks, neighbor discovery is essential in the network initialization and the design of routing, topology control, and medium access control algorithms. Therefore, efficient neighbor discovery algorithms should be devised for self-organization in wireless ad hoc networks. In this paper, we propose a probabilistic neighbor discovery (PND) algorithm, which aims at reducing the neighbor discovery time by adjusting the transmission probability of advertisement messages through the multiplicative-increase/multiplicative-decrease (MIMD) policy. To further improve PND, we consider the collision detection (CD) capability in which a device can distinguish between successful reception and collision of advertisement messages. Simulation results show that the transmission probabilities of PND and PND with CD converge on the optimal value quickly although the number of devices is unknown. As a result, PND and PND with CD can reduce the neighbor discovery time by 15.6% to 57.0% compared with the ALOHA-like neighbor discovery algorithm.

\* 본 연구는 미래창조과학부 및 정보통신산업진흥원의 대학 ICT연구센터 지원사업 (NIPA-2014-H0301-14-1015) 및 정보통신기술연구원 진흥센터의 정보통신 및 방송연구개발사업 (2014-044-006-004)의 일환으로 수행되었음

♦ First Author : Korea University School of Electrical Engineering, crazytb@korea.ac.kr, 학생회원

o Corresponding Author : Korea University School of Electrical Engineering, shpack@korea.ac.kr, 정회원

\* INRIA French Institute for Research in Computer Science and Control, hyunhee.park@inria.fr, 정회원

논문번호 : KICS2014-06-226, Received June 4, 2014; Revised August 12, 2014; Accepted September 3, 2014

## I. 서론

이웃 탐색은 무선 네트워크를 초기화하거나, 주위 기기들에게 이웃 기기에 대한 정보를 알려주기 위해 반드시 필요한 과정이다. 또한, 라우팅 알고리즘을 설계하거나 토폴로지 컨트롤 방법을 설계하는 데 있어서 반드시 이루어져야 하는 과정이다. 하지만 무선 애드혹 네트워크와 같이 분산적인 환경이라면 이웃 기기들에 대한 정보를 공유하기 힘들기 때문에 효율적인 이웃 탐색 방법은 반드시 필요하다.

가장 기본적인 이웃 탐색 방법으로는 매체 제어 프로토콜로 잘 알려진 ALOHA 기법이 널리 사용된다. 정해진 확률  $p$ 로 광고 메시지를 전송하고 그렇지 않으면 주위의 다른 광고 메시지를 수신할 수 있도록 광고 메시지 송신을 시도하지 않는 것이다. 이러한 방법에서는 확률  $p$ 를 정하는 것이 매우 중요한데,  $p$ 가 과도하게 크다면 전송된 광고 메시지가 많아져 충돌을 유발하게 되고,  $p$ 가 작다면 채널이 비어있는 시간이 많아져 채널이 낭비될 수 있기 때문이다. 확률  $p$ 는 네트워크에 존재하는 기기의 수의 역수일 때 최대의 처리율이 발생된다는 것이 알려져 있다<sup>[1]</sup>.

그 동안 많은 이웃 탐색 방법이 연구되어 왔다<sup>[2-4]</sup>. 상기 방법들은 모두 사전에 모든 기기들이 네트워크에 존재하는 기기의 수를 알고 있다고 가정하고 있다. 하지만 무선 애드혹 네트워크에서와 같은 분산적인 환경에서는 네트워크에 존재하는 기기의 수를 미리 알기 힘들다. 이웃 기기에 대한 정보를 알기 힘들기 때문에 [5]에서는 60GHz 지향성 안테나를 사용한 환경에서 2.4GHz 대역을 통해 이웃 탐색을 수행하는 기법을 제안하였으나, 또 다른 통신 대역이 필요하다는 단점을 지니고 있다. [6]는 지향성 안테나를 활용한 이웃 탐색 기법을 제시하였다. 그러나 주어진 시간 안에  $k$ 개의 이웃, 즉  $k$ -connectivity만을 보장할 수 있는 제한된 알고리즘을 제시하였다. [7]에서는 빠른 이동성 환경에서의 이웃 탐색 기법을 제안하였으나 네트워크에 존재하는 기기의 수에 대한 고려를 하지 않았다. [8]에서는 지향성 안테나를 사용한 경우에서의 이웃 탐지 과정을 분석하였으나 수학적인 모델링만이 제시되었다. 이러한 단점을 극복하기 위해서 네트워크의 기기의 수를 알 수 있다는 전제를 완화하고, 광고 메시지 전송 확률을 점차 감소시키는 방법을 통해 기기의 수를 알지 못해도 작동할 수 있는 기법<sup>[9]</sup>이 제안되었으나, 광고 메시지의 전송 성공 여부에 적응적으로 대처하지 못한다는 단점이 있다.

본 논문에서는 확률적 이웃 탐색 기법 (PND:

Probabilistic neighbor discovery)을 제안한다. 본 기법은 MIMD (Multiplicative-increase, multiplicative-decrease) 정책을 통해 광고 메시지의 전송 확률을 제어함으로써 이웃 탐색에 소요되는 시간을 줄이는 것을 목표로 하고 있다. 각 기기는 임의의 확률로 자신의 ID와 자신의 전송 확률을 포함한 광고 메시지를 전송한다. 그리고 광고 메시지 전송의 결과, 즉 송신 시도, 수신성공, 수신실패, 그리고 채널이 사용되지 않은 상황에 따라 적응적으로 전송 확률을 조정한다. 그러므로 본 기법에서는 네트워크의 기기 수를 알지 못해도 시간이 흐름에 따라 최적의 전송 확률로 수렴해 가며, 이웃 탐색에 소요되는 시간을 줄일 수 있다. 더불어 임의의 기기가 광고 메시지를 전송한 경우, 그 메시지가 성공적으로 전송되었는지 여부를 알 수 있는 충돌 감지 기법 (CD: Collision detection)<sup>[10], [11]</sup>,<sup>[12]</sup>을 도입함으로써 확률적 이웃 탐색 기법의 성능을 더 높일 수 있다. 또한 모든 기기가 각자의 전송 범위 내에 존재하지 않는 멀티홉 환경에서도 PND 기법을 통해 이웃 탐색을 시행할 수 있다. 시뮬레이션을 통해 기존의 ALOHA-like neighbor discovery (AND) 기법<sup>[2]</sup>과 성능 비교를 수행하였으며 그 결과 이웃 탐지 과정에 소요되는 시간이 15.6% 감소함을 확인하였다. 또한 충돌 감지 기능을 적용했을 경우 최대 57.0% 까지 이웃 탐지 시간을 줄일 수 있음을 확인하였다.

2장에서는 PND 기법의 시스템 모델을 서술하였다. PND 알고리즘은 3장에서 서술하였고 충돌 감지 기법에 대한 분석은 4장에서 서술하였다. 5장과 6장에서는 시뮬레이션 결과와 그에 따른 결론을 서술하였다.

## II. 시스템 모델

본 기법에서는 slotted-ALOHA 프로토콜을 기반으로 한 매체 접근 제어 프로토콜을 가정한다. Slotted-ALOHA 프로토콜에서는 시간이 슬롯으로 나뉘어져 있고 각 시간 슬롯의 시작점에서만 전송을 시도할 수 있다. 또한 slotted-ALOHA 프로토콜은 단방향 통신만을 지원한다고 가정하였다. 즉, 모든 기기는 메시지를 송신하고 있는 중이라면 어떠한 메시지도 수신할 수 없게 된다. 만약 둘 이상의 메시지가 동일한 시간 슬롯에서 수신한다면 충돌이 발생하여 메시지를 성공적으로 수신할 수 없게 된다. 일반적으로, 메시지를 송신한 기기는 해당 시간 슬롯에서 충돌이 발생했는지 여부를 알 수 없으나, 물리 계층 정보를 통하거나 또 다른 라디오 채널을 통한 기법을 통해 충돌을 감지할 수도 있다<sup>[8]</sup>. 이러한 충돌 감지 기법을 통해 이웃

탐색에 소요되는 시간을 줄일 수 있고, 본 논문에서는 PND 기법에 충돌 감지 기법을 적용했을 때와 적용하지 않았을 때 모두 고려하였다. 네트워크에 존재하는 기기들은 고정된 위치를 지니고 있으며 각자 고유한 ID를 가지고 있다고 가정한다. 광고 메시지를 전송할 경우 그 메시지에는 기기의 고유 ID와 전송 확률이 포함되어 그 메시지를 수신한 다른 기기들은 어떠한 확률을 통해 전송된 메시지인지 파악할 수 있고, 어떤 기기가 전송하였는지 알 수 있다. 이러한 정보를 통해 광고 메시지를 수신한 기기들은 각자의 전송 확률을 갱신한다.

그림 1은 상기 명시한 slotted-ALOHA를 이용한 이웃 탐색 기법의 예시를 나타낸다.  $D$ 는 네트워크 내에 존재하는 기기의 집합을 의미한다.  $D_T(t)$ 는 시간 슬롯  $t$ 에서 광고 메시지 전송을 시도하는 기기의 집합이고,  $D_w(t)$ 는 시간 슬롯  $t$ 에서 광고 메시지 수신을 위해 대기하는 기기의 집합을 나타낸다. 그림 1에서 검은 상자는 네트워크의 기기를 의미하며 색칠된 부분은 광고 메시지 전송을 의미한다. 즉, 기기 1과 기기 4는 첫 번째 시간 슬롯에서 광고 메시지 전송을 시도하였다. 반면에 두 번째 시간 슬롯에서는 어떤 기기도 광고 메시지 전송을 시도하지 않았다. 본 예제에서는  $D_T(1)=\{1,4\}$ ,  $D_w(1)=\{2,3\}$ ,  $D_T(2)=\emptyset$ ,  $D_w(2)=\{1,2,3,4\}$ 가 된다.

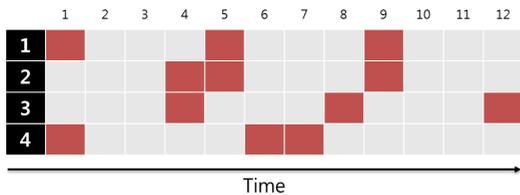


그림 1. Slotted ALOHA 방식의 이웃 탐색 기법 예시  
Fig. 1. Example of slotted ALOHA-like neighbor discovery

### III. 확률적 이웃 탐색 기법

본 장에서는 먼저 PND 기법의 동작 과정을 서술하며, 이후에 PND 기법의 동작 예시를 서술한다.

#### 3.1 PND 동작 과정

PND 기법은 분산적인 환경에서 동작된다. 즉, 모든 기기들은 각자의 전송 확률을 분산적으로 연산하여 그에 따라 광고 메시지를 slotted-ALOHA 프로토콜을 통해 브로드캐스팅한다.  $p_{t,i}$ 를 기기  $i$ 가 시간 슬롯  $t$ 에서 광고 메시지 전송을 시도할 확률이라고 정의

한다. 기기  $i$ 는  $1-p_{t,i}$ 의 확률로는 다른 기기로부터의 광고 메시지 수신을 위해 대기하게 된다. 기기  $i$ 는 다음 시간 슬롯  $t+1$ 에서의 전송 확률을 MIMD 정책으로 제어한다. 즉, 시간 슬롯  $t+1$ 에서의 전송 확률  $p_{t+1,i}$ 는  $p_{t,i}$ 에 정해진 상수를 곱하거나 나누는 값의 꼴로 나타나게 된다.  $p_{t+1,i}$ 를 결정하는 경우는 네 가지로 나뉘는데, 1) 기기  $i$ 가 시간 슬롯  $t$ 에 광고 메시지를 전송한 경우, 2) 기기  $i$ 가 시간 슬롯  $t$ 에 광고 메시지를 전송하지 않고, 다른 기기의 광고 메시지를 성공적으로 수신한 경우, 3) 기기  $i$ 가 시간 슬롯  $t$ 에 광고 메시지를 전송하지 않고, 다른 기기로부터 둘 이상의 메시지를 수신하게 되어 충돌이 발생한 경우, 4) 어떤 기기도 전송 시도를 하지 않아 채널이 비어있는 경우이다.

1) 경우 1: 기기  $i$ 가 시간 슬롯  $t$ 에 광고 메시지를 전송한 경우, 기기  $i$ 는 다음 시간 슬롯인  $t+1$ 에 동일한 확률로 광고 메시지를 전송 시도 한다. 기기  $i$ 가 전송을 시도했다면, 해당 시간 슬롯에서 기기  $i$ 가 얻을 수 있는 정보는 없으므로 다음 시간 슬롯인  $t+1$ 에서는 동일한 확률로 전송할 수 밖에 없다. 즉, 기기  $i$ 의 시간 슬롯  $t+1$ 에서의 전송 확률  $p_{t+1,i}$ 는 아래와 같다.

$$p_{t+1,i}=p_{t,i} \text{ for all } i \text{ in } D_T(t) \quad (1)$$

2) 경우 2: 기기  $i$ 가 시간 슬롯  $t$ 에 광고 메시지를 전송하지 않고 대기하면서, 다른 기기  $j$ 만이 시간 슬롯  $t$ 에  $p_{t,j}$ 의 확률로 광고 메시지 전송을 시도한 경우이다 (즉,  $|D_T(t)|=1$ ). 이 경우 다음 시간 슬롯인  $t+1$ 에 기기  $j$ 로부터 수신한 광고 메시지에 포함된 확률 (즉,  $p_{t,j}$ )로 광고 메시지를 전송 시도 한다. 즉, 기기  $i$ 의 시간 슬롯  $t+1$ 에서의 전송 확률  $p_{t+1,i}$ 는 아래와 같다.

$$p_{t+1,i}=p_{t,j} \text{ for all } i \text{ in } D_w(t) \quad (2)$$

3) 경우 3: 둘 이상의 기기가 동일한 시간 슬롯에서 광고 메시지 전송을 시도한 경우 충돌이 발생하게 된다 (즉,  $|D_T(t)|\geq 2$ ). 이러한 경우  $D_w(t)$ 에 포함된 기기, 즉 광고 메시지를 전송 시도하지 않은 기기는 충돌된 메시지를 수신하게 된다. 이러한 경우 수신을 시도한 기기  $i$ 는 다음 시간 슬롯인  $t+1$ 에는 변수  $c_{coll}$ 만큼 낮은 확률로 광고 메시지를 전송 시도한다. 둘 이상의 광고 메시지를 받은 경우, 네트워크에 존재하는 기기들의 전송 확률이 최적값에 비해 높게 형성되어 있다고 간주할 수 있으므로 둘 이상의 광고 메시지를 받은 기기는 MIMD 정책에 따라 다음 시간 슬롯  $t+1$ 에서의 전송 확률을 줄이게 된다. 즉, 기기  $i$ 의 시간 슬롯

t+1에서의 전송 확률  $p_{t+1,i}$ 는 아래와 같다.

$$p_{t+1,i} = p_{t,i} / c_{coll} \text{ for all } i \text{ in } Dw(t), c_{coll} > 1 \quad (3)$$

4) 경우 4: 어떤 기기도 광고 메시지 전송을 시도하지 않은 경우, 채널은 사용되지 않고 비어있게 된다 (즉,  $|D_T(t)|=0$ ). 기기 i는 어떤 기기도 광고 메시지를 전송하지 않았다는 것을 알 수 있고, 수신을 시도한 모든 기기들은 다음 시간 슬롯 t+1에서 변수  $c_{idle}$ 을 곱한 확률로 광고 메시지를 전송 시도한다. 어떤 기기도 광고 메시지 전송을 시도하지 않은 경우는 네트워크에 존재하는 기기들의 전송 확률이 낮게 형성되어 있다고 간주할 수 있으므로, 기기 i는 시간 슬롯 t+1에서 MIMD 정책에 따라 전송 확률을 높이게 된다. 즉, 기기 i의 시간 슬롯 t+1에서의 전송 확률  $p_{t+1,i}$ 는 아래와 같다.

$$p_{t+1,i} = p_{t,i} * c_{idle} \text{ for all } i \text{ in } Dw(t), c_{idle} > 1 \quad (4)$$

### 3.2 PND 동작 예제

표 1에서 4개의 기기가 PND 기법을 기반으로 이웃 탐색을 수행하는 과정을 나타내었다. tx라고 기재된 열이 회색으로 칠해진 경우는 해당 기기가 광고 메시지 전송을 시도했다는 의미이고, 칠해지지 않은 경우는 전송을 시도하지 않고 주위 기기의 메시지 수신을 시도했다는 의미이다.  $p_{0,i}$ 는 기기 i의 최초 전송 확률을 의미한다. 최초 전송 확률은 기기 1, 2, 3, 4가 각각 0.4, 0.3, 0.2, 0.1로 지정하였다. 본 확률값은 동작 예제를 설명하기 위해 임의로 설정하였다. 변수  $c_{coll}$ 과  $c_{idle}$ 은 각각 임의로 1.5로 지정하였다.

첫 번째 시간 슬롯에서는 오직 기기 3만이 광고 메시지 전송을 시도하였다. 기기 3이 전송한 광고 메시지는 기기의 ID와 기기 3의 전송 확률, 즉 0.2라는 값이 포함되어 있으므로 이를 수신한 기기 1, 2, 4는 다음 시간 슬롯에서의 전송 확률을 0.2로 조정한다.

두 번째 시간 슬롯에서 어떤 기기도 광고 메시지를 전송하지 못했다. 그러므로 모든 기기는  $0.2 * c_{idle}$ , 즉 0.3으로 전송 확률을 조정한다.

표 1. PND 동작 예제  
Table 1. An example of PND

Dev	$p_{0,i}$	1st timeslot		2nd timeslot		3rd timeslot		4th timeslot	
		tx	$p_{1,i}$	tx	$p_{2,i}$	tx	$p_{3,i}$	tx	$p_{4,i}$
1	0.4		0.2		0.3		0.2		0.3
2	0.3		0.2		0.3		0.3		0.3
3	0.2		0.2		0.3		0.3		0.3
4	0.1		0.2		0.3		0.2		0.3

세 번째 시간 슬롯에서는 기기 2과 기기 3이 전송을 시도하게 되어 충돌이 발생하였다. 이러한 경우 기기 2과 기기 3은 충돌이 발생했다는 사실을 알 수 없으므로 다음 시간 슬롯에서는 같은 확률로 광고 메시지 전송을 시도하게 되나, 기기 1과 기기 4는 둘 이상의 메시지가 송신되어 충돌이 발생했다는 것을 알 수 있으므로  $0.3 / c_{coll}$ , 즉 0.2로 전송 확률을 조정한다.

네 번째 시간 슬롯에서는 기기 2만이 광고 메시지 전송을 시도하였다. 전송을 시도한 기기 2는 전송 확률을 바꾸지 않으며, 이를 성공적으로 수신한 기기 1, 3, 4는 자신의 광고 메시지 전송 확률을 기기 2의 전송 확률이었던 0.3으로 변경한다.

## IV. 충돌 방지 기법이 도입된 확률적 이웃 탐색 기법

PND 기법에서는 어떤 기기가 광고 메시지를 전송하게 되는 경우 그 전송이 성공적으로 송신되었는지 여부를 알 수 있는 방법이 없다. 그렇기 때문에 그 기기가 성공적으로 광고 메시지를 전송하였다고 하더라도 기기는 지속적으로 이웃 탐색 과정에 참여할 수 밖에 없다.

반면, 충돌 방지 기법이 도입된다면 광고 메시지를 성공적으로 전송한 기기는 이 사실을 알 수 있으므로 더 이상 이웃 탐색 과정에 참여하지 않아도 되므로 결과적으로 이웃 탐색에 소요되는 시간이 감소하게 된다. 표 2는 충돌 방지 기법이 도입된 확률적 이웃 탐색 기법의 예를 나타낸다. 표 1의 가정과 동일하게 기기 1, 2, 3, 4의 최초 전송 확률은 각각 0.4, 0.3, 0.2, 0.1로 지정하였고  $c_{coll}$ 과  $c_{idle}$ 은 반복 실험을 통해 가장 좋은 성능을 보여주는 값인 1.5로 지정하였다.

첫 번째 시간 슬롯에서는 오직 기기 3만이 광고 메시지 전송을 시도하였다. 기본 PND와 마찬가지로 광고 메시지는 기기의 ID와 기기 3의 전송 확률, 즉 0.2라는 값이 포함되어 있으므로 이를 수신한 기기 1,

표 2. 충돌 방지 기능이 적용된 PND 동작 예제  
Table 2. An example of PND with collision detection

Dev	$p_{0,i}$	1st timeslot		2nd timeslot		3rd timeslot		4th timeslot	
		tx	$p_{1,i}$	tx	$p_{2,i}$	tx	$p_{3,i}$	tx	$p_{4,i}$
1	0.4		0.2		0.13		0.2		0.2
2	0.3		0.2		0.13		0.2		-
3	0.2		-		-		-		-
4	0.1		0.2		0.13		0.2		0.2

2, 4는 다음 시간 슬롯에서의 전송 확률을 0.2로 조정한다. 기기 3은 충돌 방지 기능으로 인해 자신의 전송이 성공적으로 이루어졌다는 것을 알 수 있고, 이후로 광고 메시지 전송 시도를 하지 않는다는 점이 기본 PND와 다른 점이다.

두 번째 시간 슬롯에서는 기기 1과 기기 2가 전송을 시도하게 되어 충돌이 발생하였다. 충돌 방지 기능의 적용으로 인해 기기 1과 기기 2는 자신들의 광고 메시지가 성공적으로 전송되지 못했음을 파악할 수 있고, 기기 1, 2, 그리고 기기 4는  $0.2/c_{coll}$ , 즉 0.13으로 자신들의 전송 확률을 낮춘다.

세 번째 시간 슬롯에서는 어떤 기기도 광고 메시지 전송을 시도하지 않았다. 이 경우에는 기본 PND와 마찬가지로 각 기기들은  $0.13 * c_{idle}$ , 즉 0.2로 전송 확률을 조정한다.

네 번째 시간 슬롯에서 기기 2만이 광고 메시지 전송을 시도하였다. 기기 1과 기기 4는 자신의 광고 메시지 전송 확률을 기기 2의 확률이었던 0.2로 변경하고 기기 2는 이후의 이웃 탐색 과정에 참여하지 않는다.

### V. 시뮬레이션 결과

기본 PND와 충돌 감지 기능이 적용된 PND의 성능을 분석하기 위해 MATLAB 프로그램을 사용하였다. 모든 이웃 탐색 과정은 모든 기기가 성공적으로 광고 메시지를 전송하였을 경우 종료되며, 신뢰성 있는 시뮬레이션 결과 도출을 위해 모든 결과값은 1000번씩 시행된 결과의 평균값을 의미한다. 최초의 전송 확률은 0과 0.5 사이의 임의의 값을 설정하도록 하였다.

제안하는 PND는 두 가지 기법과의 비교를 통해 분석되었다. 첫 번째 비교 기법은 균등확률기법(EP: Equal probability)이다. 균등확률기법에서는 각 기기가 자신의 전송 범위에 존재하는 기기의 수를 모두 알고 있다고 가정하고, 전송 확률은 slotted-ALOHA의 최적 값으로 알려진  $1/N$ , 즉 존재하는 기기의 수의 역수의 확률을 유지한다.

두 번째 비교 기법은 AND (ALOHA-like neighbor discovery)<sup>[2]</sup> 기법이다. AND 기법에서 이웃 탐색은 단계별로 진행되며 각 단계는 하나 이상의 시간 슬롯으로 구성되어 있다. 기기들은 각자의 전송 확률을 다음 단계로 진행할 때 까지 기하급수적으로 감소시킨다. 단계 j에 있는 기기는  $1/2^j$ 의 확률로 광고 메시지를 전송하며 단계 j는  $2^j e^{(ln2+c)}$  시간 슬롯만큼 지속된다. c는 근사 오류이며, 0으로 설정하였다.

#### 5.1 최초 전송 확률에 대한 변화

그림 2는 최초 전송 확률의 변화에 따른 이웃 탐색 소요 시간을 나타낸 그림이다. 초기 전송 확률을 0과 0.5 사이의 임의의 값을 설정하게 한 경우와,  $1/N$ , 즉 기기 수의 역수로 지정하게 한 경우를 비교하였다. 두 경우에 대해 상당한 차이가 나타나는 않은 모습을 관찰할 수 있는데, 이것은 초기 전송 확률이 어떻게 부여되던 몇 번의 이웃 탐색이 이루어지면서 최적값에 빠르게 수렴하기 때문이다.

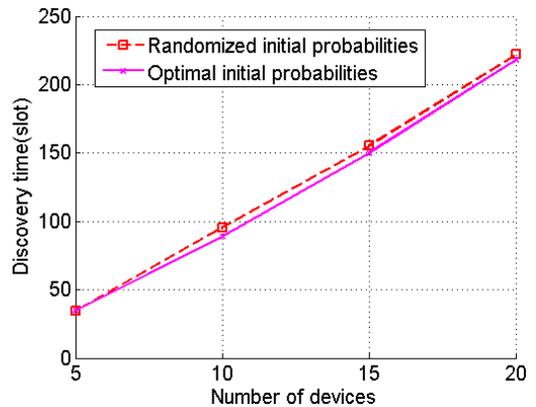


그림 2. 임의의 초기 전송 확률과 지정된 초기 전송 확률에 대한 이웃 탐색 소요 시간

Fig. 2. Neighbor discovery time when randomized initial transmission probabilities and optimal initial transmission probabilities

#### 5.2 전송 확률의 수렴

PND 기법은 모든 기기로부터의 광고 메시지가 성공적으로 전송될 때까지 수행된다. 그림 3은 PND 기법에서 20개의 기기가 각자 연산한 광고 메시지 전송 확률의 변화를 나타낸 그림이다. 그림 3(a)는 기본 PND의 경우이고 그림 3(b)는 충돌 방지 기능이 적용된 PND의 경우이다. 그림 3(b)에서 최적값을 나타내는 푸른 점선이 이차함수의 형태를 띠는 것을 관찰할 수 있는데, 이는 광고 메시지를 성공적으로 전송한 기기는 더 이상 이웃 탐색 과정에 참여하지 않기 때문이다.

그림 3(a)에서는 다양하던 전송 확률들이 점점 충돌 확률을 줄여가다가 하나의 값으로 수렴한 후 최적 값인 0.03, 즉 네트워크에 존재하는 기기의 수 (즉, 30개)의 역수로 수렴해 가는 것을 관찰할 수 있다. 그림 3(b)에서 역시 점점 증가하는 최적값을 향해 수렴해 가는 전송 확률을 관찰할 수 있다. 본 결과를 통해 PND 기법이 최초의 확률을 무작위로 부여하더라도 시간 슬롯이 경과함에 따라 효과적으로 최적값에 수

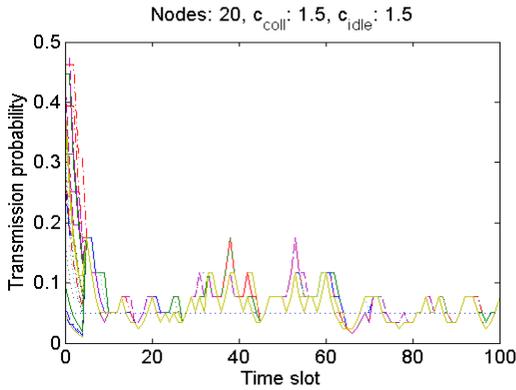


그림 3. (a) 시간의 흐름에 따른 PND 기법의 전송 확률 추이  
Fig. 3. (a) Transmission probabilities when PND is adopted

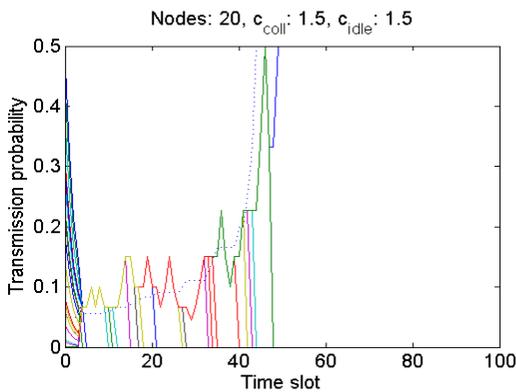


그림 3. (b) 시간의 흐름에 따른 충돌 감지 기능이 적용된 PND 기법의 전송 확률 추이  
Fig. 3. (b) Transmission probabilities when PND with CD is adopted

렴하는 것을 볼 수 있고, 결과적으로 네트워크에 존재하는 기기의 수를 알지 못하고, 최초의 전송 확률을 무작위로 부여하더라도 최적 전송 확률값에 수렴하는 것을 볼 수 있다.

### 5.3 기존 이웃 탐색 기법과의 성능 비교

그림 4에서는 충돌 방지 기능을 적용하지 않은 경우에서, PND 기법과 비교기법인 AND 기법, 그리고 EP 기법을 비교하였다. PND(X,Y)는, X는  $c_{coll}$ , Y는  $c_{idle}$ 을 의미한다. PND(1.5, 1.5)는 AND 기법 대비 네트워크의 기기 수가 10개와 40개일 때 각각 3.1%와 15.6%의 소요시간 감소를 보였다. PND(1.5, 1.5)는 또한 EP 기법 대비 기기 수가 40개일 때 9.1%만의 성능 하락을 보였다.

PND(1.1, 1.5)와 PND(1.5, 2)의 경우는 PND(1.5,

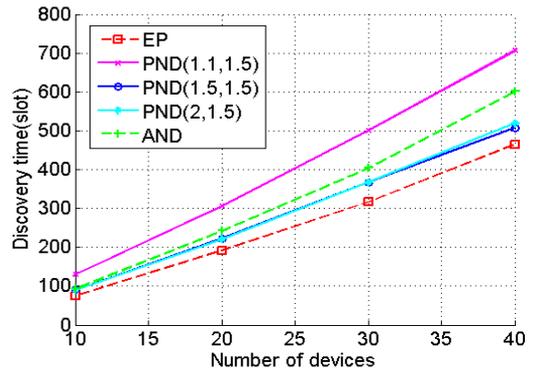


그림 4. (a) 가변적인  $c_{coll}$ 에 대한 이웃 탐색 소요 시간  
Fig. 4. (a) Neighbor discovery time when  $c_{coll}$  is varying for PND

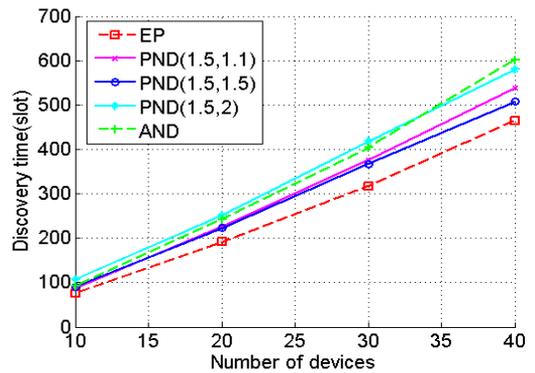


그림 4. (b) 가변적인  $c_{idle}$ 에 대한 이웃 탐색 소요 시간  
Fig. 4. (b) Neighbor discovery time when  $c_{idle}$  is varying for PND

1.5)의 소요 시간보다 더 오래 걸리는 것을 볼 수 있는데, 이는 적절한  $c_{coll}$ 값과  $c_{idle}$ 값의 설정이 성능에 큰 영향을 끼친다는 것을 나타낸다.

그림 5에서는 충돌 감지 기능이 적용된 경우의 소요 시간을 나타낸다. PND(1.5, 1.5)는 AND 기법 대비 기기 수가 10개와 40개일 때 각각 45.7%와 57.0%의 소요 시간 감소를 보였는데, 충돌 방지 기능을 PND 기법에 적용했을 때, 적용하지 않는 PND 기법 대비 기기 수가 10개와 40개일 때 각각 69.3%와 76.5%의 성능 향상을 보이는 것으로 보아 충돌 방지 기능이 PND 기법의 성능 향상에 큰 영향을 끼치는 것을 알 수 있다.

### 5.4 멀티홉 환경에서의 성능 비교

그림 6에서는 다양한 멀티홉 환경에서 PND 기법의 성능이 어떤 추이를 보이는지 나타내었다. 가능한 멀티홉 환경 중에서 대표적으로 Linear, Dumbbell, 그

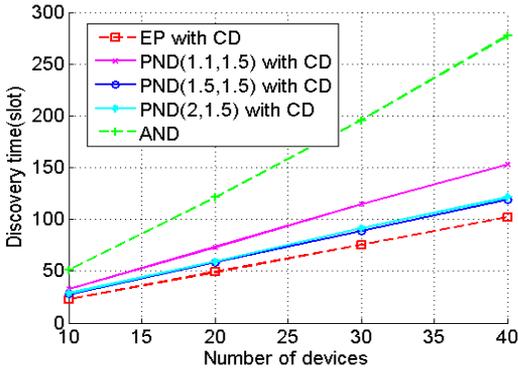


그림 5. (a). 충돌 감지 기법이 적용된 환경에서의 가변적인 ccoll에 대한 이웃 탐색 소요 시간  
Fig. 5 (a). Neighbor discovery time when ccoll is varying for PND with CD

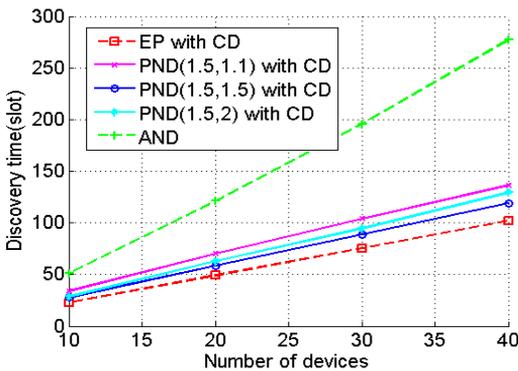


그림 5. (b). 충돌 감지 기법이 적용된 환경에서의 가변적인 cidle에 대한 이웃 탐색 소요 시간  
Fig. 5 (b). Neighbor discovery time when cidle is varying for PND with CD

리고 Fullmesh 구조를 선택하여 시뮬레이션을 수행하였다. Linear 구조에서는 기기가 일렬로 나열되어 있으며 임의의 기기의 전송범위는 양 옆의 기기, 즉 두 기기까지 도달할 수 있는 환경을 의미한다. Dumbbell 구조에서는 중심되는 두 기기가 서로 연결되어 있고 나머지 기기들은 절반씩 상기 언급한 두 기기에 연결되어 마치 덤벨형상으로 하고 있는 환경을 의미한다. Fullmesh 구조에서는 모든 노드가 서로의 전송 범위 내에 존재하는 환경을 의미한다. 모든 환경에 대해 존재하는 기기의 수는 10개,  $c_{coll}$ 과  $c_{idle}$ 은 모두 1.5로 지정하였다.

Linear 구조에서는 EP 기법, 즉 기기 개수의 역수의 값으로 광고 메시지를 전송하는 기법보다도 탐색 시간이 40.3% 감소되었음을 알 수 있는데, 모든 노드가 일렬로 나열된 환경이라면 실질적으로는 자신과 양 옆에 존재하는 기기 둘을 포함하여 1/3의 확률로

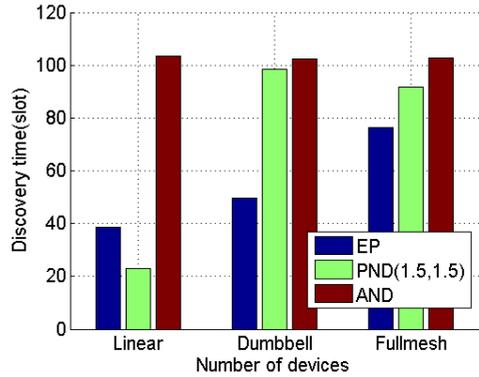


그림 6. 다양한 멀티홉 환경에서의 이웃 탐색 소요 시간  
Fig. 6. Neighbor discovery time for multihop environments

광고메시지를 전송하는 것이 최적값이나 EP 기법에서는 1/N, 즉 0.1의 확률로 광고메시지를 전송하게 되었다. 반면에 PND 기법을 사용하면 광고 메시지 전달 결과에 따라 분산적이고 자율적으로 전송 확률을 조정하기 때문에 최종 소요 시간이 감소하게 된다. 반면 Dumbbell 구조에서는 EP 기법 대비 98.5% 소요 시간이 증가하였는데, 이는 중심되는 두 기기는 광고 메시지를 자주 전송해야 함에도 불구하고 광고 메시지 전송 확률을 빠르게 확보하지 못하기 때문으로 해석된다. 그러나 Dumbbell 구조에서도 비교 기법인 AND의 성능은 상회하는 것을 알 수 있다.

## VI. 결 론

우리는 본 논문에서 확률적 이웃 탐색 기법, 즉 PND 기법을 제안하였다. PND 기법은 이웃 탐색 과정 시, 광고 전송 결과에 따라 다음 시간의 광고 메시지 전송 확률을 분산적이고 자율적으로 조정함으로써 네트워크에 존재하는 기기의 수를 알지 못하는 상황에서도 효과적으로 이웃 탐색 과정을 수행할 수 있다. MIMD 정책을 통해 최적의 광고 메시지 전송 확률로 빠르게 접근함으로써 최적 소요시간에 근접한 성능을 보여준다. 비교 기법 대비 15.6~57%의 성능 향상을 볼 수 있었다. 더불어 충돌 감지 기능을 적용하면 적용하지 않았을 때 대비 70%의 소요 시간 감소 효과를 보임을 보였다. 또한 하드웨어적인 추가나 수정 없이도 멀티홉 환경과 같은 다양한 환경에 사용될 수 있음을 보였다.

변수  $c_{coll}$ 과  $c_{idle}$ 의 값은 본 PND 기법의 성능에 큰 영향을 주는 것을 알 수 있었다. 본 논문에서는 최적의 변수 값에 대한 고려는 하지 않았으나, 향후 최적

의 변수 값을 도출해낼 수 있는 연구를 수행하여 보다 정밀한 기법을 제안할 수 있도록 할 예정이다.

### References

[1] L. G. Roberts, "ALOHA packet system with and without slots and capture," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28-42, Apr. 1975.

[2] S. Vasudevan, D. Towsley, D. Goeckel, and R. Khalili. "Neighbor discovery in wireless networks and the coupon collector's Problem," in *Proc. Mobicom*, pp. 181-192, Beijing, China, Sept. 2009.

[3] X. An, R. V. Prasad, and I. Niemegeers, "Neighbor discovery in 60 GHz wireless personal area networks," in *Proc. IEEE WoWMoM*, pp. 1-8, Montreal, Canada, Jun. 2010.

[4] L. You, Z. Yuan, P. Yang, and G. Chen, "ALOHA-like neighbor discovery for low-duty-cycled wireless sensor networks," in *IEEE WCNC*, pp. 749-754, Cancun, Mexico, Mar. 2011.

[5] H. Park, Y. Kim, and C.-H. Kang, "Cooperative neighbor discovery in 60GHz ad-hoc networks using directional antennas," in *Proc. KICS ICC*, pp. 1165-1166. Jeju Island, Korea, Jun. 2011.

[6] W. Kim and D. Cho, "Directional antenna based neighbor discovery for connectivity of wireless networks," in *Proc. KICS ICC*, pp. 1412-1413, Jun. 2010.

[7] J. H. Song, K. Kang, and Y.-J. Cho, "Fast neighbor discovery for construction of WPAN in extreme mobile network," in *Proc. KICS ICC*, pp. 720-723, Nov. 2007.

[8] M. Kim and W. Lee, "Analysis of neighbor discovery process with directional antenna for IEEE 802.15.3c," *J. KICS*, vol. 37, no. 1, pp. 9-14, Jan. 2012.

[9] S. Vasudevan, M. Adler, D. Goeckel, and D. Towsley, "Efficient algorithms for neighbor discovery in wireless networks," *IEEE/ACM Trans. Networking*, vol. 21, no. 1, pp. 69-83,

Feb. 2013.

[10] F. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II. the hidden terminal problem in carrier sense multiple-access and the busy-tone solution," *IEEE Trans. Commun.*, vol. 23, no. 12, pp. 1417-1433, Dec. 1975.

[11] J. Peng, L. Cheng, and B. Sikdar, "A wireless MAC protocol with collision detection," *IEEE Trans. Mobile Computing*, vol. 6, no. 12, pp. 1357-1369, Dec. 2007.

[12] S. Sen, R. Roy Choudhury, and S. Nelakuditi, "CSMA/CN: Carrier sense multiple access with collision notification," *IEEE/ACM Trans. Networking*, vol. 20, no. 2, pp. 544-556, Apr. 2012.

#### 송 태 원 (Taewon Song)



2010년 2월 : 고려대학교 전기 전자전파공학과 졸업  
 2010년 3월~현재 : 고려대학교 전기전자공학과 석박사 통합 과정  
 <관심분야> 차세대 무선랜, 지향성 안테나

#### 박 현 희 (Hyunhee Park)



2002년 2월 : 한국항공대학교 통신정보공학과 졸업  
 2011년 8월 : 고려대학교 전자컴퓨터공학과 박사 졸업  
 2014년 8월~현재 French Institute for Research in Computer Science and Control Postdoctoral Fellow  
 <관심분야> 무선 랜 네트워크, 지향성 안테나, 통신 표준화, WiFi alliance

백 상 현 (Sangheon Pack)



2000년 2월 : 서울대학교 컴퓨터공학부 졸업

2005년 2월 : 서울대학교 전기컴퓨터공학부 박사 졸업

2007년 3월~현재 : 고려대학교 전기전자전파공학부 부교수  
<관심분야> 미래인터넷, 무선 이동 네트워크