

# WPANs에서 센서노드의 위치 기반 비컨 스케줄링 기법

조영준<sup>o</sup> 안세영 남흥우 안순신

고려대학교 전자전기공학과

yjjo@dsys.korea.ac.kr, syahn@korea.ac.kr, hwnam@dsys.korea.ac.kr,

sunshin@dsys.korea.ac.kr

## Location-based beacon transmission scheduling in WPANs

Youngjun Jo<sup>o</sup>, Sea-young Ahn, Heungwoo Nam and Sunshin An

Department of Electronics Engineering, Korea University

### 요 약

센서 네트워크는 여러 서비스 분야로 그 영역을 확장하여 발전하고 있다. 에너지 측면을 고려해야 하는 센서 네트워크의 위치 인식 기술 분야 역시 많은 발전을 하여 상용화 된 RF 칩을 선보이고 있다. 현재 RF를 이용한 센서 노드를 이용한 위치 인식은 정확도 측면에 취약성을 가지고 있지만 기술이 발전함에 따라 극복 될 것이다. 본 논문은 코디네이터 노드가 위치 인식 능력을 가지고 있다고 가정하고 그 정보를 이용하여 비컨 스케줄링 하여 센서 네트워크의 효율을 극대화 시키는 것이다.

### 1. 서 론

802.11 [1]의 에너지 사용은 802.15.4의 에너지 사용보다 관대하여 hidden node problem, exposed node problem을 해결하기 위해 CTS와 RTS 기법을 사용한다. 하지만 802.15.4는 에너지 측면을 고려하였기 때문에 최대한 간단한 구조로 디자인되어 있다. 802.15.4를 사용하는 Zigbee는 비컨모드에서 offset 값을 이용하여 2hop까지 비컨 프레임간 충돌을 피해 비컨 프레임의 Hidden node problem을 해결하고자 하였다. 하지만 노드의 빈번한 이동성을 지원한다면 Zigbee의 비컨 스케줄링은 hidden node problem과 exposed node problem [2]에 취약해지고 그로 인해 비컨 프레임들 간의 충돌이 발생하고 재전송으로 인해 에너지 소모가 많아 질 것이다.

본 논문이 제안하는 비컨 지역 분할 알고리즘은 hidden node problem과 exposed node problem을 해결하기 위한 알고리즘이다. 기본 개념은 그림 1과 같다.

노드 A가 임의의 이웃노드와 통신하기 위해서는 같은 시간 내에 그 목적지가 되는 임의의 노드가 도넛1에 있는 노드들로부터 호출을 받아서는 안 된다.(hidden node problem) 또한 같은 시간에 도넛 1의 임의의 노드들 간의 통신 또는 도넛 1의 임의의 노드와 도넛2의 임의의 노드 간의 통신으로 간섭이 생겨 충돌이 생겨서는 안 된다.(Exposed node problem)

따라서 노드 A노드가 충돌 없이 원1 영역의 이웃노드들과 통신을 하기 위해서는 최소한 노드 A로부터 반경 3D인 원의 영역을 관리해야 한다.

모든 노드들이 이동성을 갖고 있기 때문에 이상적으로 그림1처럼 관리하기는 어렵다.

### 2. 위치 기반의 비컨 스케줄링 알고리즘 위한 가정

센서 노드의 지리정보를 이용한 비컨 스케줄링을 하기 위해 다음과 같은 가정을 한다.

1. 코디네이터 중 비컨 전송을 하기 위한 노드는 자신의 위치 정보를 알고 있다.
2. 모든 노드는 이상적인 등방성 안테나를 사용하며 최대 전송거리는(D) 동일하다.
3. 장비의 종류에는 코디네이터, 디바이스로 구분된다. 코디네이터의 위치정보를 인식할 수 있는 능력을 가진 장비로서 팬 코디네이터와 비컨 전송 코디네이터 그리고 비컨 전송 능력을 갖춘 코디네이터로 구분된다. 팬 코디네이터(PANC: PAN coordinator)는 망 전체를 관리하는 능력을 갖춘 노드이고 비컨 전송 코디네이터(BTC: Beacon transmission coordinaotr)는 지역 영역 안에서 비컨을 전송하는 노드이다.

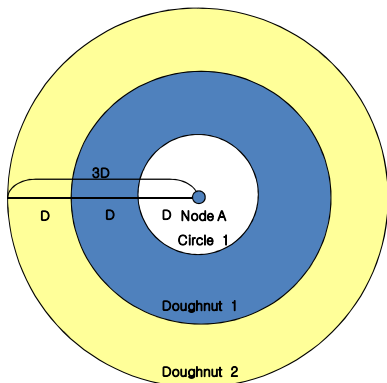


그림 1 hidden/exposed node problem 해결하기 위한 영역

마지막으로 비컨 능력을 갖춘 노드(BTCC: Beacon transmission capability coordinator)는 비컨 전송을 하지 않지만 지역 영역에 비컨 전송 코디네이터가 이동하거나 문제가 생겼을 시 비컨 전송을 할 수 있는 노드이다.

디바이스는 코디네이터로부터 비컨을 전송받고 센싱한 데이터를 코디네이터에게 전송하는 일반적인 센싱 디바이스이다.

4. Zigbee [3-5] 와 802.15.4 [6-8]기반의 비컨 스케줄링 기법이다.

본 논문은 위치 기반 비컨 스케줄링을 위하여 첫째 비컨 지역 분할 알고리즘과 둘째 비컨 허가(획득) 알고리즘 나누어 설명한다.

### 3. 비컨지역 분할 알고리즘

본 논문은 반지름이 3D인 영역을 관리하기 위하여 다음과 같이 지형의 위치를 분할 한다.

다음은 비컨 지역 분할 알고리즘을 나타낸다.

**Step 1:** 최대전송거리(D) 설정.

가정에 의해 모든 노드의 최대 전송 범위(D)는 동일하다. D는 실험을 통해 얻어지거나 제조사의 데이터 시트에 의해 제공되는 주어진 정보이다.

노드의 계산능력의 한계로 이 단계에서 최대전송거리(D)의 측정 단위 및 범위를 제한한다. 본 논문에서 측정단위는 미터(m) 단위, 측정범위는 소수 2자리(버림)을 기본으로 한다.

**Step 2:** 최대전송거리(D) 대각선으로 하는 정사각형 한 변의 길이(G)를 계산.

D가 설정이 되면 D를 대각선으로 하는 정사각형의 한 변의 길이(G)를 계산한다. G의 측정 단위 및 범위를 이 단계에서 제한한다. G 역시 측정단위는 미터(m) 단위, 측정범위는 소수 2자리(버림)을 기본으로 한다. Step 1,2에서 버림을 하는 이유는 G를 한 변으로 하는 정사각형 내에 임의의 두 점을 선택할 때 두 노드가 이웃이 되기 위해서이다. G는 수식1을 통해 얻어진다.

$$G = \frac{\sqrt{2}}{2} \cdot D \quad (1)$$

노드의 계산의 능력의 한계와 계산 효율을 위해  $\sqrt{2} \cong 1.41$  로 즉 소수 2자리에서 버림 한다.

**Step 3:** 반지름이 3D인 원을 포함 할 수 있는 최소 정사각을 그리드(G) 분할.

3D를 반경으로 하는 원을 포함하는 정사각형은 한 변의 길이가 6D이다. 한 변의 길이 6D인 정사각형을 한 변의 길이가 G인 정사각형으로 분할 한다. 만약 6D와 G가 정확히 나누어 떨어지지 않는다면 몫을 올림한다. 올림한 몫(n)은 식2와 같다.

$$n = \left\lceil \frac{6D}{G} \right\rceil \quad (2)$$

3D를 반경으로 하는 원을 포함하고 hidden node problem과 Exposed node problem를 해결하기 위하여 관리되어야 하는 최소 정사각형의 한 변의 길이는  $n \cdot G$ 가 된다. n는 커버해야 하는 영역을 G정사각형으로 분할 할 때 한 변에서 생기는 분할 개수를 의미한다. 따라서  $n \cdot G$  정사각형에 포함되는 G 정사각형의 개수는  $n^2$ 가 된다.

**Step 4:** 분할된 영역에 index를 설정.

$n^2$ 의 정사각형들은 비컨 충돌을 방지하기 위하여 분할된 영역으로 비컨 충돌방지 영역(BCA-Z:Beacon collision avoidance Zone)이라고 칭한다. 본 단계에서 비컨 충돌방지 영역(BCA-Z)에 index를 부여한다. Index 부여 방법에 따라 전송지연과 같은 성능이 향상될 수 있다. 그러나 index 부여 방법은 본 논문의 영역을 벗어 나고 우리는 순차적으로 증가하는 인덱스를 기본으로 한다.

그림2는 비컨지역 분할 알고리즘을 보여 준다.

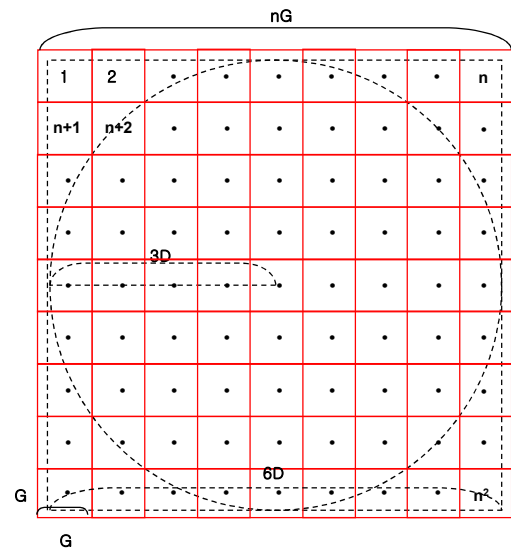


그림 2 비컨지역 분할 알고리즘

### 3.1 비컨지역 분할 알고리즘 적용을 위한 superframe 구조

모든 노드는 active 상태와 inactive 상태로 나뉜다. Active 상태에는 fully active 상태와 half active 상태로 나뉜다. Fully active 상태는 송수신(송신과 수신)이 가능한 상태이다. Half active 상태는 수신만 가능한 상태이다. Inactive 상태는 fully inactive 상태와 half inactive 상태로 나뉜다. Fully inactive 상태는 에너지 소모를 줄이기 위한 fully sleep 상태이다. Half inactive 상태는 에너지 소모와 전송 충돌을 위한 half sleep 상태이다. Superframe 구조 802.15.4와 같다. 그림 3은 위치기반 충돌 회피 superframe 구조를 보여 준다.

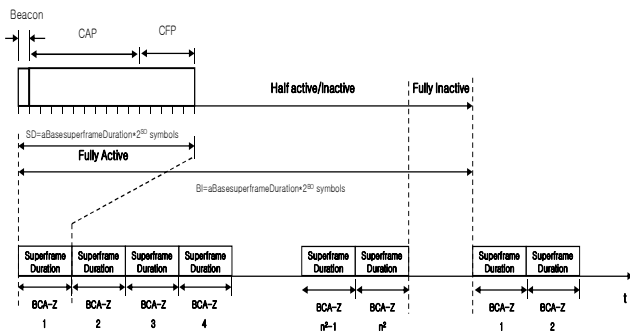


그림 3 비컨지역 분할 알고리즘 적용을 위한 superframe 구조

식 1과 2에 의해서 n을 유추하면 식 3과 같다.

$$n \cong 9 \cong \left\lceil \frac{6}{0.7} \right\rceil \quad (3)$$

그림4는 n이 9이고 BCA-Z 210이 fully active 시간에 각 BCA-Z의 상태를 보여 준다.

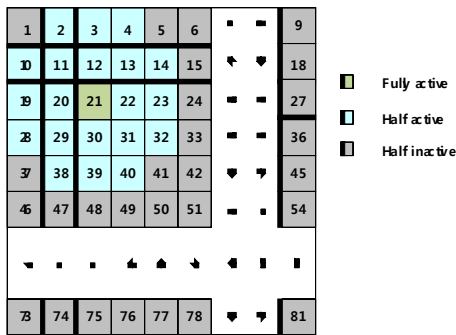


그림 4 BCA-Z 210이 fully active 시간에 다른 BCA-Z의 상태 BCA-Z 에서 비컨전송 노드의 위치가 정확하게 결정된 것이 아니기 때문에 좌우상하 방향으로 2개의 BCA-Z 그리고 대각선 방향으로 1개의 BCA-Z 영역이 Half active 상태이어야 한다. 그 이유는 BCA-Z의 정사각형 한 변의 길이가 G이고 대각선의 길이가 D이기 때문이다.

위치기반 충돌 회피 superframe 구조가 갖는 장점은 비컨 전송 노드의 이동성을 보장하고 비컨 충돌을 회피할 뿐만 아니라 데이터 전송의 hidden node problem과 exposed node problem을 최소화 할 수 있다.

### 3.2 노드의 상태 천이 알고리즘

현재 시간에서 다음과 같은 알고리즘에 의해 상태 천이를 한다.

**Step1:** 현 시점에서 Fully active 인덱스 (X)를 계산  
 센서 네트워크를 구성할 때 SD(Superframe duration)와 BI(Beacon Interval)이 결정되었기 때문에 PAN 코디네이터에 의한 동기화부터 현 시점의 Fully active 인덱스를 계산할 수 있다.

**Step2:** 상태 천이 계산

자신의 속해있는 BCA-Z 인덱스(Y)를 통해 X와 비교하여 자신의 상태를 파악한다.

```

If Y=X
    then fully inactive
Else if Y={ (X-2n-1,X-2n, X-2n+1), (X-n-2,X-n-1,X-n,X-n+1, X-n+2), (X-2,X-1,X+1, X+2), (X+n-2,X+n-1,X+n,X+n+1, X+n+2), (X+2n-1,X+2n, X+2n+1) }
    then half active
Else if 0 < Y < n^2
    then half inactive
Else //Y<0 or n^2<Y
    Fully inactive
    
```

n은 G정사각형으로 한 변을 분할하는 정사각형의 개수이다.

### 4. 비컨 전송 코디네이터 허가 알고리즘

비컨 전송 능력을 갖고 있는 코디네이터가 망을 형성하여 PAN 코디네이터가 되거나 비컨 전송 코디네이터가 되는 알고리즘을 보여준다.

**Step 1:** 정해진 시간 채널 스캔 및 신호 감지.

```

If No signal
    then 비컨 관리 테이블 생성, 채널 설정 및 PAN 코디네이터로 망 형성
Else
    기존 PAN 가입. 비컨 관리 테이블 생성, step2
    
```

**Step 2:** 비컨 전송 코디네이터 동작.

```

If 자신이 비컨 전송 능력을 가진 노드
    then 비컨 전송 능력을 가진 노드로 참여
Else
    디바이스 노드로 참여
    
```

**Step 3:** 비컨 관리 테이블 유지/관리 (자신의 속한 BCA-Z 인덱스 (Y))와 수신된 비컨 전송 노드의 BCA-Z 인덱스 정보들 비교.

```

If 같은 BCA-Z 인덱스가 존재
    then 잠재된 비컨 전송 노드로서 참여, step 5.
Else
    비컨 전송 노드가 되기 위한 허가요청 메시지 이웃노드에 전송, Step4
    
```

**Step 4:** 비컨 전송 허가 프로토콜.

```

If 이웃노드들에게 일정시간 Nack이 없음
    then 비컨을 전송하여 이웃노드들에게 비컨 전송을 전송하여 노드임을 알림, Step 5.
Else
    비컨을 전송하지 않고 비컨 전송 능력을 가진 노드로 참여. 일정시간 후 Step3.
    
```

**Step 5:** 비컨 전송 수락 프로토콜.

```

If 비컨 허가 요청 노드의 BCA-Z 인덱스와 비컨 관리 테이블에
    
```

```

BCA-Z 인덱스가 존재하지 않음
then 아무 메시지도 발생시키지 않음. Step 3.
Else
    비컨 허가 거절 Nack 전송. Step 3
    
```

비컨 관리 테이블은 node id, BCA-Z 인덱스, Fully active 시간 정보를 갖고 있다.

### 5. 성능 평가

802.15.4 와 제안된 Location-based Beacon Transmission Scheduling (LBTS) 알고리즘에 대해 에너지 소모, 지연, 성공률을 비교한다. NS-2를 이용하여 성능을 측정하고 다음과 같은 시나리오를 통하여 성능을 평가한다.

#### 5.1 시뮬레이션 시나리오 및 환경

그림5은 시뮬레이션 시나리오를 보여준다.

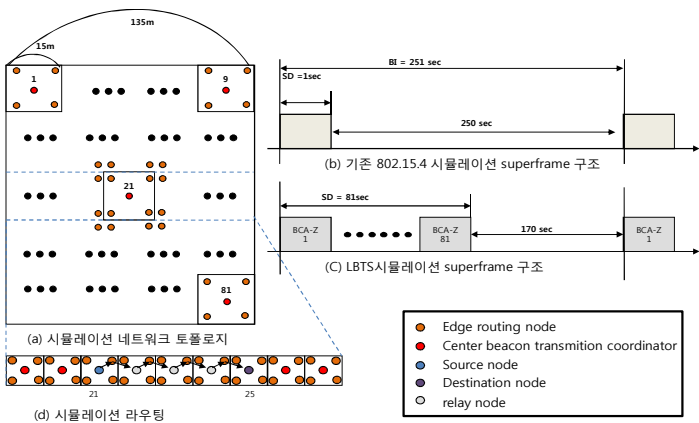


그림 5 시뮬레이션 시나리오

그림5의 (a)는 시뮬레이션 네트워크 토폴로지를 보여주고 있다. 시뮬레이션 네트워크는 BCA-Z 인덱스 1의 중앙에 있는 node가 PAN Coordinator(PANC)로 Zigbee 트리구조로 네트워크를 형성하였다. RF의 최대 전송거리는 21.3m로 설정하였다. (현재 센서 노드의 RF 모듈은 5-55m를 기본으로 하며 그 이상의 전송범위를 갖는 모듈도 있다.)

식 1에 의해 D를 대각선으로 하는 정사각형의 한 변의 길이(G)는  $15m (21.3 m * 0.7 = 14.91 m)$ 로 소수 둘째 자리 올림이다. 식 2에 의해 커버해야 하는 영역을 G정사각형으로 분할 할 때 한 변에서 생기는 분할 개수 n은  $9( (6 * 21.3m)/15m = 8.52, 올림)$  이다.

따라서  $135 x 135 m^2$ 의 정사각형 영역을 커버한다. BCA-Z의 총 개수  $n^2$ 은 81이다.

모든 노드는 중계(router) 역할과 비컨을 전송할 수 있는 능력을 갖는 노드이고 각 BCA-Z의 중앙에는 비컨노드가 위치하며 BCA-Z의 4개의 모서리에

중계역할을 하는 노드 4개가 배치된다. 성능 측정 시나리오에서는 edge effect가 없는 BCA-Z 21에 대하여 LBTS와 기존 알고리즘(802.15.4 and Zigbee) 성능을 비교한다. 총 노드의 개수는 405개(BCA-Z 하나당 5개의 노드,  $5*81$ ) 이고 이중 비컨 노드는 81개가 존재한다. 비컨 노드는 이동성을 갖고 BCA-Z 21의 비컨노드를 제외한 80개의 비컨 노드가 BCA-Z 21로 유입(이동) 할 수 있다. 우리는 이때의 BCA-Z 21에서의 에너지 소모, 지연, 성공률을 측정한다. 그림 5의 (b)와 (c)는 superframe 구조를 나타내며 (b)는 802.15.4의 superframe 구조로 Beacon interval(BI)는 4min 11sec(251.65 sec) 이고 Slot Duration(SD)은 1 sec (0.98 sec) 이다.

802.15.4에서는 SO(Superframe Order)와 BO(Beacon Order)를 조절하여 BI와 SD를 15.36ms에서 4min 11sec까지 영역을 갖는다. ( $0 \leq SO \leq BO$ ) 계산을 위하여 symbol 단위를 16us로 계산하면  $SD=15.36 * 2^{SO} ms$ ,  $BI= 15.36 * 2^{BO} ms$ 가 된다.  $SO= 6$ ,  $BO= 14$ 로 설정하였다.

또 비컨이 충돌이 생겼을 경우 random backoff를 1 sec으로 고정하였다.

(c)는 LBTS의 superframe 구조로 BI와 SI는 251 sec과 1sec으로 즉 fully active 구간은 1sec이고 Half active와 Half inactive 구간은 80sec 그리고 fully inactive구간은 170sec 이다.

라우팅 프로토콜과는 관계없이 MAC layer에서 발생하는 지연 및 성공률을 측정하기 위하여 라우팅 경로를 그림5 (d)의 고정 경로 일 때를 측정한다.

표 1. 시뮬레이션 환경

시뮬레이션 설정 요소	설정 값
최대 전송거리(D)	21.3m
BCA-Z의 한 변의 길이(G)	15m
커버 영역의 한 변을 G로 분할 할 때 개수(n)	9
커버 영역의 넓이	$135 x 135 m^2$
BCA-Z의 총 인덱스 개수( $n^2$ )	81
BO(Beacon Order)	14
SO(Superframe Order)	6
BI (Beacon Interval)	4min 11sec
SD (Slot Duration)	1 sec
802.15.4 random backoff	1 sec
노드의 총 개수(비컨 노드 및 중계 노드)	405
fully active 구간	1sec
Half active와 Half inactive 구간	80sec
fully inactive구간	170sec

### 5.2 성능 분석

### 5.2.1 에너지 소모

BCA-Z 21에 비컨 전송 노드가 1개에서 80개까지 추가될 때 임의의 edge routing node의 wake up time을 측정하였다. 비컨 전송 노드들은 BI 구간에서 한번의 비컨을 전송하며 측정은 센서 네트워크가 형성되는 초기 단계 이후 안정화 단계에서의 측정하기 위하여 시뮬레이션 시작 후 10분 이후부터 360분 wake up time을 측정하였다. Wake up time 은 비컨 전송노드가 비컨을 보내어 edge routing node를 깨우는 시간이며 이는 기존 알고리즘에서 active 구간을 의미하며 LBTS에서의 fully active 구간과 half active 구간을 의미한다.

기존의 알고리즘은 BCA-Z 21에 81개의 비컨 노드들을 BI 구간내 162 sec에서 on-off하게 하여 89 sec의 inactive 구간을 확보하였다. 한 BI 구간(4min 11sec)로 Wake up time을 나타내어 보여준다.

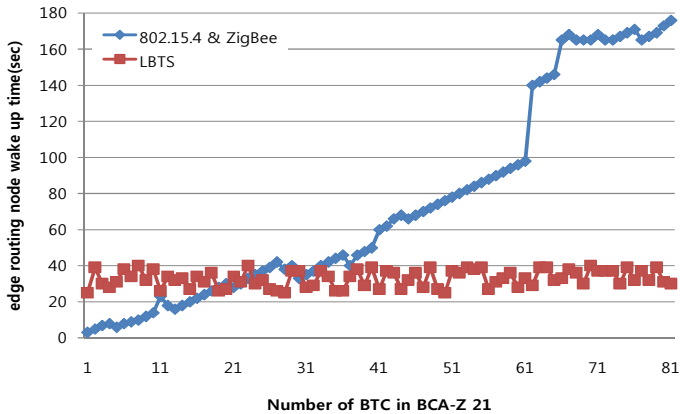


그림 6 BTC 개수 당 edge routing node의 wake up time

BTC-Z 21에서 비컨 전송 노드(BTC)가 21개까지는 기존알고리즘(802.15.4 and Zigbee) LBTS에 비해 에너지 측면에서 효율적이지만 21개에서 42개 사이에서는 유사한 성능을 보이고 42개 이상일 때 LBTS의 성능이 좋다. 특히 61개 이상일 때 비컨 노드의 충돌로 edge routing node가 계속 wake up 상태에 있음을 알 수 있다. BTC 노드가 증가할 때 Wake up 시간의 평균을 비교해 보면 기존 알고리즘은 74 sec 이고 LBTS는 33 sec을 갖는다.

BCA-Z 21에서 BTC 개수당 충돌확률은 그림 7과 같으며 43개에서 기존의 알고리즘은 높은 충돌 확률을 갖는다. LBTS는 0.2% 이내의 충돌 확률을 갖는다.

전체적인 에너지 소모 성능을 측정하기 위하여 초기부터 200초 간 BCA-Z 21영역에서 발생한 비컨프레임의 개수를 확인하였다. BCA-Z 21 영역에 비컨 전송 노드를 10, 20, 40, 60개 었을 때의 비컨프레임 개수를 확인하였으며 LBTS는 비컨 전송 코디네이터 허가 알고리즘이 적용된 것이다. 기존 알고리즘과 LBTS의 비컨 전송 평균 횟수를 비교하면

BTC가 10, 20, 40, 60에서 LBTS가 4.7, 11.4, 22.6, 59.5의 비컨 전송을 덜 하게 된다. (그림 8)

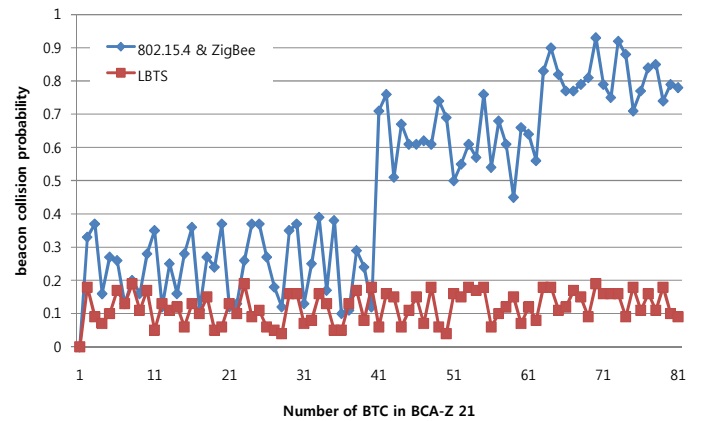


그림 7 BTC 개수 당 충돌 확률

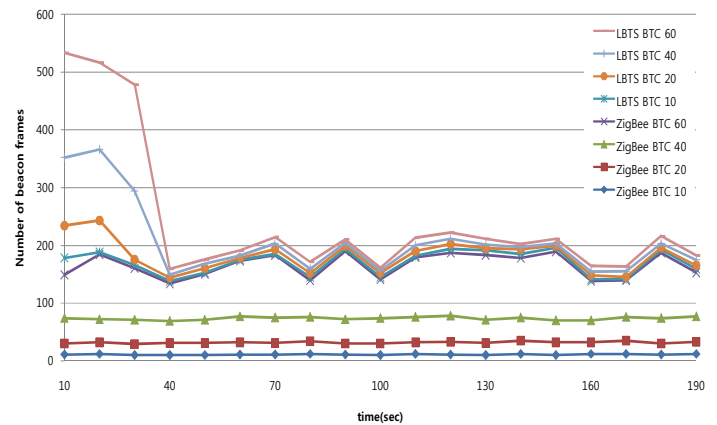


그림 8 BCA-Z 21에서 시간에 따른 비컨 프레임 개수

### 5.2.2 지연

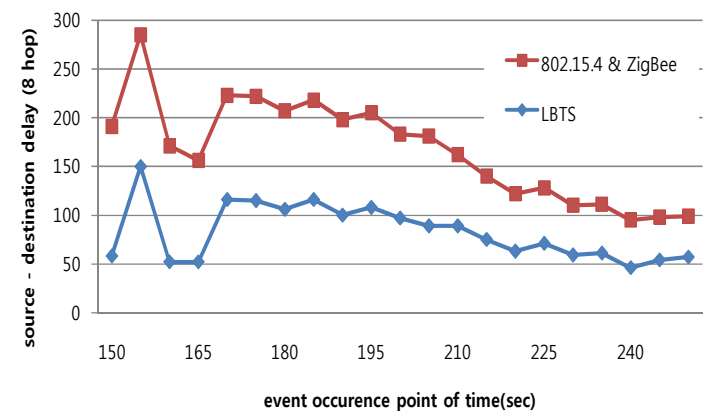


그림 9 소스와 목적지간의 전송지연 (8hop)

소스와 목적지 간의 지연을 측정하면 그림 5 (d) 처럼 8hop의 고정된 라우팅 갖는 시나리오로 측정한다.

BCA-Z 21~25에 16개의 비컨 전송 노드를 배치한다. 소스와 목적지간의 전송 지연은 평균적으로 0.2배 더 많은 지연을 LBTS가 갖는다. 하지만 기존 방법은 지연시간을 예측 할 수 없으며 이벤트가 많이 발생하는 환경에서는 LBTS 보다 더 좋지 않는 성능을 보여 준다. (그림 9)

### 5.2.3 성공률

그림 5에 (d)의 경로(BCA-Z 21~25)에 BTC가 10, 20, 40, 60, 80를 배치하고 소스와 목적지까지 1000개의 패킷을 전송하여 성공적으로 수신한 패킷 갯수를 성공률로 나타낸다.

BCA-Z에 BTC 노드가 많아 질수록 기존 알고리즘에서는 많은 충돌을 갖게 된다. BTC의 개수가 10, 20, 40, 60, 80 일 때 기존알고리즘과 LBTS의 성공 확률을 보면 0.024, 0.09, 0.341, 0.543, 0.632로 BTC 노드의 개수가 10개, 20개의 성공률은 기존 알고리즘과 LBTS가 거의 비슷하지만 BTC 노드가 60개와 80개일 때 LBTS가 약 54% 이상의 좋은 성공률을 갖는다.(그림 10)

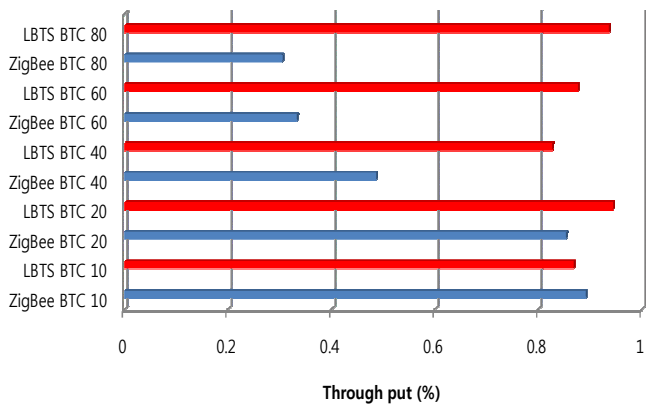


그림 10 성공률

## 6. 결론

LBTS 알고리즘은 에너지 측면에서 802.15.4와 ZigBee 알고리즘 보다 성능이 우수하고 지연 측면에서 성능이 떨어지지만 이는 성공률 측면에서 보다 우수하다는 것을 수학적 해석으로 증명하였다. 예로 그림 5와 같은 시뮬레이션 시나리오에서 에너지 측면에서 가장 큰 부분을 차지하는 전송부분을 보면 을 보면 기존의 알고리즘에 비해 LBTS는 54%의 비컨 전송을 절감하여 에너지 효율을 증대시킨다. 뿐만 아니라 Wake up time 과 충돌 측면에서도 LBTS가 더 좋은 효율을 보였다. 하지만 지연부분에 있어서 기존의 알고리즘의 비해 LBTS는 0.2배의 지연을 갖지만 이것은 기존 알고리즘이 충돌이 없다는 가정에서의 결과이며 실제로는 그 이하가 될 것이다. 또 LBTS는 지연 시간 예측이 가능하여 라우팅

경로가 확정이 되면 그 지연시간을 예측 할 수 있으며 기존 알고리즘은 예측하기가 어렵다. 마지막으로 성공률은 LBTS가 기존 알고리즘에 비해 0.09% 높은 성공률을 갖는다.

현재 위치 인식은 많은 연구가 진행되고 있으며 여러 응용 어플리케이션 중심으로 이루어 지고 있다. 우리는 이러한 위치 인식 기술을 근간으로 그 정보를 MAC layer에 적용하여 비컨 컨트롤을 적용하였다. 위치 인식 기술을 어플리케이션을 벗어나 하위 layer 적용하면 더 많은 에너지와 성공률의 증대에 따른 보다 많은 서비스를 창출을 기대할 수 있다.

### ACKNOWLEDGMENT

“본 연구는 지식경제부 및 정보통신산업진흥원의 대학IT 연구센터 지원사업의 연구결과로 수행되었음” (NIPA-2010-C1090-1021-0001)

### 참고문헌

- [1] IEEE Standard 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.
- [2] P Jarn, MACA-A new channel access method for packet radio. In Proc. ARRL/CRRL Amateur Radio Ninth Computer Networking Conf., ARRL, 1990, pp. 134-140
- [3] Zigbee Alliance, <http://www.zigbee.org>, 2007.
- [4] Ran Peng, Sun Mao-heng and Zou You-min "ZigBee Routing Selection Strategy Based on Data Services and Energy-balanced ZigBee Routing" 2006. APSCC '06. IEEE Asia-Pacific Conference on Dec. 2006 Page(s):400 ~ 404
- [5] S. C. Ergen, ZigBee/IEEE 802.15.4 Summary, <http://www.eecs.berkeley.edu/csinem/academic/publications/zigbee.pdf>
- [6] Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPAN), IEEE Standard 802.15.4TM, Oct. 2003.
- [7] A. Koubaa et al., "Collision-Free Beacon Scheduling Mechanisms for IEEE 802.15.4/Zigbee Cluster-Tree Wireless Sensor Networks," IPP-HURRAY Technical Report, HURRAY-TR-061104, 2006.
- [8] M. Attia, A. Koubaa, and M. Alves, "Beacon scheduling in cluster-tree IEEE 802.15.4/ZigBee Wireless Sensor Networks," IPP-HURRAY Technical Report, HURRAY-TR-060604, 2006.