

# 6LoWPAN의 계층적 라우팅 기법에서 이웃 노드 리스트를 이용한 Short-cut 라우팅 알고리즘

## Short-cut Routing Algorithm Using Neighbor Node List in Hierarchical Routing for 6LoWPAN

강문경\*, 진교홍\*

Moon-Kyoung Kang\*, Kyo-Hong Jin\*

### 요 약

6LoWPAN에서 사용되는 계층적 라우팅 프로토콜인 HiLow는 적은 메모리 사용과 낮은 에너지 소모율을 보이지만 최적의 경로를 제공하지 못하는 단점이 있다. 본 논문에서는 HiLow 라우팅 알고리즘의 성능을 개선하기 위해서 계층적 라우팅 경로정보 뿐만 아니라 소스 노드 주위의 이웃 노드의 정보를 활용하여 데이터가 전달될 최적의 다음 홉(Next hop)을 결정하는 알고리즘을 제안하였다. 제안된 알고리즘은 컴퓨터 시뮬레이션을 통해 성능을 분석하였으며, 그 결과 홉 수, 전송되는 메시지 수 측면에서 많은 개선이 있음을 알 수 있었다.

### Abstract

The HiLow, a hierarchical routing protocol for 6LoWPAN, needs smaller memory allocation and energy than other routing protocols. However, the HiLow has a weak point that does not provide the optimized routing path. In this paper, we propose the algorithm that can find the optimum next hop using neighboring nodes around the source node as well as hierarchical routing path information. We evaluate the performance of proposed algorithm using the computer simulation, and we can acquire results that decreases a number of hop count and transmitted message.

Key words : 6LoWPAN, IPv6, Hierarchical routing, HiLow, Neighbor List

### I. 서 론

무선 센서네트워크는 유비쿼터스 컴퓨팅 분야의 핵심 기술 중의 하나로써 사물 및 환경 정보의 수집이 요구되는 지역에 설치되어 다수의 센서 노드들이 스스로 네트워크 구축과 사용자에게 정보 전달을 목적으로 하는 네트워크이다.

무선 센서네트워크는 Application-Specific한 특징

을 가지므로 다음과 같은 요구사항을 가질 수 있다. 먼저, 센서 노드들은 배터리 충전 및 교체가 불가능하기 때문에 최대한 에너지 소모를 줄여야만 한다. 또한 센서 노드에서 생성된 데이터가 기존의 IP 네트워크를 통해 사용자에게 전달될 수 있도록 외부 네트워크와의 연동이 필요하다. 마지막으로, 많은 수의 센서 노드들에게 주소 할당이 가능하고 센서 노드들을 직접 조작하는 것이 제한적이기 때문에 자

\* 국립장원대학교 메카트로닉스공학부 전자공학전공(Dept. of Electronics Engineering, Changwon National University)

· 제1저자 (First Author) : 강문경

· 접수일자 : 2007년 5월 29일

동 주소 할당 등의 기능이 제공되어야만 한다[1,2].

6LoWPAN(IPv6 over Low Power Wireless Personal Area Network)은 위의 요구사항들을 모두 수용할 수 있는 기술로써 PHY, MAC 계층은 IEEE 802.15.4를 채택하였으며 외부 IP 네트워크와의 연동 및 주소 할당을 위해서 IPv6(Internet Protocol version 6)를 사용한다. 또한, IEEE 802.15.4와 IPv6사이에 Adaptation 계층을 두어 IPv6 헤더의 압축과 해제, 분할과 재조립, 주소의 압축과 매핑 및 라우팅 등의 역할을 수행하도록 하고 있다[1,2].

6LoWPAN에서 사용하기 위해 제안된 라우팅 프로토콜 중에서 HiLow는 16비트 Short Address를 계층적으로 할당하여 라우팅 테이블 없이 라우팅이 가능하게 하는 계층적(Hierarchical) 라우팅 프로토콜이다[1]. 따라서 센서 노드는 지정된 계층적인 경로만을 통해서 데이터를 전달하기 때문에 노드 오류나 링크 에러가 발생한 경우에 목적지로 데이터를 전달할 수 없으며 Addressing 과정에서 결정되는 계층적 경로 때문에 직접 전송이 가능함에도 불구하고 계층적으로 다수의 노드를 거쳐 데이터를 전송하는 문제점이 발생할 수 있다.

본 논문에서는 이러한 HiLow의 문제점을 보완하기 위해서 계층적 라우팅 정보와 소스 노드 주위의 이웃 노드들의 정보를 담은 Neighbor List(NL)를 활용하여 데이터가 전달될 최적의 다음 홉을 결정하는 알고리즘을 제안하였다.

본 논문의 구성은 다음과 같다. II장에서는 센서 네트워크에서 사용되는 라우팅 프로토콜들과 6LoWPAN, 그리고 대표적인 계층적 라우팅 프로토콜인 HiLow에 대해 간단히 살펴보고, III장에서는 HiLow의 문제점을 정의한다. IV장에서는 제안된 알고리즘을 소개하고, V장에서는 제안된 알고리즘의 성능을 분석하고 평가하였다. 마지막으로 VI장에서 논문의 결론 및 향후 연구 계획을 기술하였다.

## II. 관련 연구

### 2-1 6LoWPAN

6LoWPAN은 IEEE 802.15.4 MAC/PHY의 상위 계층으로 IP 및 TCP/UDP 등의 기존 인터넷에서 사용하는 통신 프로토콜을 이용하는 환경을 말한다. IEEE802.15.4 MAC/PHY를 사용하기 때문에 패킷 전송에 있어서 여러 제약을 갖게 된다. 6LoWPAN의 특징은 다음과 같다[3].

- 1) 물리 계층의 패킷 크기가 최고 127 바이트이고, 상위 MAC 계층의 패킷 크기는 102 바이트이기 때문에 작은 패킷 크기를 가져야 한다.
- 2) MAC 계층에서 16비트 길이의 짧은 주소 방식과 64비트 길이의 확장 주소 방식을 지원한다.
- 3) 250kbps 이하의 적은 대역폭을 가진다.
- 4) 스타 토폴로지와 메쉬 토폴로지를 지원한다.
- 5) 비교적 저가형 장비와 관련이 있다.
- 6) 기기들의 위치는 일반적으로 정해지지 않는다. 또한, 때때로 기기들에 대한 접근성이 떨어질 수 있다.
- 7) LoWPAN 내의 기기들은 다양한 이유로 신뢰성을 보장받기 힘들다.

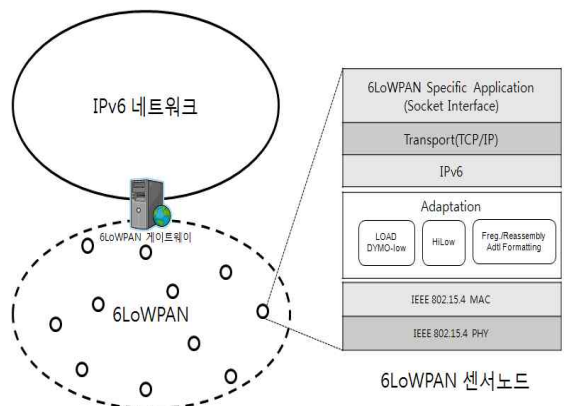


그림 1. 6LoWPAN의 센서 노드 구조 및 6LoWPAN과 외부 IPv6 네트워크의 연결 구조

Fig. 1. Structure of 6LoWPAN Sensor Node and Connection Between 6LoWPAN and IPv6 Network

6LoWPAN은 IPv6 프로토콜을 지원하는 네트워크

이다. 따라서, 네트워크 내에서 각 센서 노드는 IPv6 주소를 부여 받게 되므로 외부에 있는 IP 네트워크 내의 호스트가 6LoWPAN 내의 센서 노드를 제어할 수 있다.

그림 1에 제시된 바와 같이 6LoWPAN과 외부 IPv6의 네트워크 연결은 게이트웨이를 통해서 이루어진다. 6LoWPAN 센서 노드의 프로토콜 구조는 그림 1과 같다. 새롭게 추가된 어댑테이션 계층은 IPv6 헤더의 압축/해제, 분할과 재조립, 주소의 압축과 매핑 등 상위 계층인 IPv6와 하위 계층인 IEEE 802.15.4를 절충해 주는 역할을 한다.

현재 6LoWPAN의 기술 정의에 대한 표준이 RFC 4919로 정의 되었고, 6LoWPAN의 메시지 형식에 대한 인터넷 드래프트가 IESG를 통과한 상태이다. 이외에도 라우팅, 연동, 이웃 탐색 등에 대한 연구가 활발하게 진행 중이다.

2-2 HiLow

6LoWPAN 노드가 네트워크에 접속하고자 할 때, HiLow는 현재 6LoWPAN 네트워크가 존재하는지 확인하기 위해서 스캔 과정을 수행한다. 만약 네트워크가 존재한다면, 연결을 요청하는 메시지를 전송한다. 이 메시지를 수신한 디바이스는 부모 노드가 되며 16비트 Short Address를 생성한 후, 메시지에 포함시켜 전송한다. 연결을 요청한 디바이스는 메시지를 수신하고 부모 노드가 할당한 16비트 Short Address를 자신의 Address로 설정 한다. 그리고 자신은 부모 노드의 자식 노드가 된다. 16비트 Short Address를 할당하기 위해서는 수식(1)을 이용한다[4].

$$AC_n = MC \times A_p + n \quad (1)$$

$AC_n$  : 현재 할당된 자식의 수 (n-1)개일 때 다음 할당되는 자식 노드의 주소

$A_p$  : 주소를 할당하는 부모 노드의 주소

$MC$  : Max Children

그림 2은 16비트 Short Address를 할당하는 개념을 나타낸 것이다. 그러나 만약 주변에 6LoWPAN이 존재하지 않으면 노드는 자신의 Address를 0으로 하

고 새로운 6LoWPAN을 구성하기 위해 스스로 코디네이터가 된다. 주소 할당 과정에서서는 라우터(또는 코디네이터)가 자식 노드와 부모 노드의 정보를 가진 Neighbor Table을 반드시 유지해야 한다.

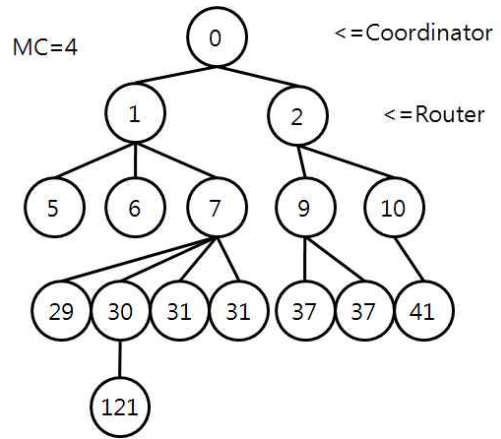


그림 2. 16비트 Short Address 할당 예  
Fig. 2. Assignment of Short Address

수식(2)는 수식(1)의 재귀적 공식이다. 이를 이용하면 부모 노드의 주소들을 구할 수 있다.

$$A_p = (A_c - 1) / MC \quad (2)$$

$A_c$  : 부모 노드의 주소를 알고자 하는 노드의 주소

```
function Find_NextHop()
    if C is belong to SA
        next_hop = Find_Ancessor_D(DC+1, D)
    else if C is belong to SD
        next_hop = Find_Ancessor_D(DC-1, C)
    else
        next_hop = Find_Ancessor_D(DC-1, C)
    end if
    return next_hop
end function

function Find_Ancessor_D(D,k)
    ::find an ancestor of a descendant of node k
    depth D end function
```

그림 3. HiLow 라우팅 알고리즘  
Fig. 3. HiLow Routing Algorithm

- D : The destination node address
- C : The current node address
- SA : The set of ancestors of D
- SD : The set of descendants of D
- DD : The depth of D
- DC : The depth of C

그림 3는 HiLow의 라우팅 알고리즘의 Pseudo Code를 보여준다. 메시지를 수신한 노드가 SA이면, 메시지의 목적지 주소 필드 값이 D가 되어 다음 홉은 D의 DC+1 깊이의 노드가 된다. 반면, 현재 노드가 SD이면, 메시지의 목적지 주소 필드 값이 C가 되어 C의 DC-1 깊이의 노드가 다음 홉이 된다. 그 외의 경우는 현재 노드가 SA인 경우와 같은 방법으로 다음 홉을 설정한다.

### III. HiLow 프로토콜의 문제점

#### 3-1 최적 경로

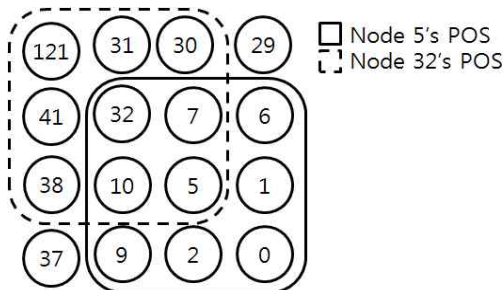


그림 4. 그림 2의 각 노드들 실제 위치와 노드 5와 32의 POS

Fig. 4. Position of Fig.2's Nodes and Each POS of Nodes 5 and 32

그림 4는 그림 2 노드들의 실제 위치를 보여준다. 그림 4에서는 노드 121과 노드 41이 1홉 거리에 있어 직접 전송이 가능하지만 그림 2의 계층적 라우팅을 사용하여 데이터를 전송하는 경우, 노드 30, 7, 1, 0, 2, 9를 거쳐서 목적지 41로 도착하게 된다. 계층적 라우팅을 사용하면 데이터의 전송이 비효율적인 라우팅 경로를 통해 발생할 수 있다는 단점이 있다 [5,6].

#### 3-2 연결 실패 또는 링크 에러

부모 또는 자식 노드로 경로가 이미 결정된 계층적 라우팅의 경우, 노드 에러가 발생하면 연결 실패 또는 링크 에러가 발생한다. 이런 경우 HiLow는 POS(Personal Operating Space) 안에서 자신의 부모 노드를 찾는 메시지를 보내게 된다. 부모 노드를 찾으면 다시 연결을 설정하고, 찾지 못하면 POS 안에서 새로운 부모 노드로부터 16비트 Short Address를 다시 할당받게 된다. 여기서 문제는 다시 주소를 할당 받게 되어 네트워크 안에 존재하는 모든 노드들의 NL를 수정해야하는 오버헤드가 생긴다.

### IV. 제안된 알고리즘

본 논문에서 제안한 알고리즘은 On-demand 라우팅보다 적은 오버헤드와 Hierarchical 라우팅보다 최적의 경로를 설정할 수 있도록 이웃 노드의 정보를 추가하여 제한적 환경의 6LoWPAN에 좀 더 적합하도록 도모하였다. IETF 6LoWPAN WG에 internet-draft로 제안된 계층적 프로토콜 HiLow를 기반으로 설계하였다.

#### 4-1 이웃 리스트(Neighbor List, NL)

각 노드의 POS안에 존재하는 다른 노드들의 정보를 Neighbor List로 구성한다. 그러기 위해서 그리 5와 같이 Neighbor Table의 Relationship 필드에 Neighbor를 추가하여, 부모노드와 자식노드뿐만 아니라 이웃노드도 인식할 수 있도록 한다.

전체 네트워크상의 각 노드들이 어드레싱 작업을 수행할 때 이웃 리스트는 자동적으로 구성 된다. 이때, 사용되는 메시지는 IEEE 802.15.4 MAC계층의 Association Request MSG와 Association Response MSG이다. 다른 노드들이 주고받는 이 메시지들을 overhearing하여 이웃 리스트를 구성하게 된다.

하지만, 이들 메시지를 받지 못해, NL에 등록하지 못하는 노드가 발생한다. 이러한 문제를 해결하기 위해서는 노드들이 제일 처음 데이터를 전송할 때, 64비트 주소와 16비트 주소 동시에 사용한다면 노드

들은 이미 64비트 주소에 대한 정보가 있기 때문에 매핑 작업을 통해 16비트 주소를 NL에 등록 가능하게 된다.

(16bits)	(16bits)	(64bits)	(2bits)	(2bits)
PANId	16bit Addr	64bit Addr	Device Type	Relationship

<b>b1 b0</b>	: Description	<b>b1 b0</b>	: Description
00	: Coordinator	00	: Parent
01	: Router	01	: Child
10	: End Device	10	: Neighbor
11	: Reserved	10-11	: Reserved

그림 5 이웃 테이블의 관계 필드의 값  
Fig. 5. Value of The Relationship Subfield In Neighbor Table

이런 방법으로 그림 4에서 5 노드의 NL에는 0, 1, 2, 6, 4, 9, 10, 32 노드들이 노드 32의 NL에도 5, 7, 10, 30, 31, 38, 41, 121 노드들이 존재해야한다.

4-2 메시지 형식

General MAC Frame	
Frame Control Field(2octets)	
Frame Type Value	
<b>b2 b1 b0</b>	: Description
000	: Beacon
001	: Data
010	: Acknowledgment
011	: MAC command
100	: Redirect_ACK
101-111	: Reserved

그림 6. 프레임 타입의 보조 필드의 값  
Fig. 6. Value of The Frame Type Subfield

그림 6은 IEEE 802.15.4 MAC 계층의 프레임 형식을 나타낸다. 제안하는 라우팅 알고리즘에서는 프레임 형식에 Redirect\_ACK라는 프레임 타입이 새롭게 추가된다. 기능은 현재 노드에서 전송하려는 패킷이 만약 Redirect\_ACK 안의 소스 주소와 시퀀스번호가

같은 경우 다른 노드에서 이미 라우팅을 하였으니 현재 노드에서는 라우팅을 중단하라는 의미이다. Redirect\_ACK 프레임은 그림 7.(a) ACK 프레임에 2 옥텟의 소스 주소가 추가된 형태이고 그림 7.(b)와 같다.

<b>Octets : 2</b>	<b>1</b>	<b>2</b>
Frame control	Sequence number	FCS
MHR		MFR

(a)

<b>Octets : 2</b>	<b>1</b>	<b>2</b>	<b>2</b>
Frame control	Sequence number	Source Address	FCS
MHR			MFR

그림 7. (a) 확인 프레임 형식, (b) 리다이렉트-확인 프레임 형식

Fig. 7. (a) Acknowledgment Frame Format, (b) Redirect-acknowledgment frame format

4-3 라우팅 알고리즘

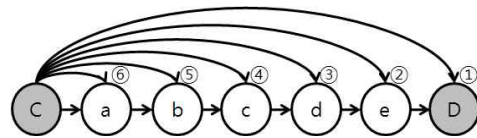


그림 8. 알고리즘의 개념도  
Fig. 8. Diagram of Algorithm

그림 8은 현재 노드 C에서 목적지 노드 D까지의 계층적 경로를 나타낸다. 만약 다음 홉이 a가 아니라 D, e, d 등으로 바로 전송하게 된다면 계층적 라우팅 보다 전체 홉 수가 줄어든다. 그림 8에서의 숫자는 C에서 D까지의 전체 홉 수이자 우선순위이다.

```

function Find_NextHop()
  if (D is belong to current node's NL)
    send Redirect_ACK
    next_hop = D
  else
    next_hop = Find_NearestNode()
  end if
  return next_hop
end function
    
```

```

function Find_NearestNode()
    loop(i=0; i<DC-Dc;i++)
        F = Find_Ancestors_D(DC-i, C)
        if (F is belong to current node's NL)
            nearest_node = F
        end if
    end loop
    loop(j=0;j<DD-Dc;j++)
        F = Find_Ancestors_D(DD-j, D)
        if (F is belong to current node's NL)
            nearest_node = F
            break
        end if
    end loop
    return nearest_node
end function

function Find_Ancestors_D(D, k)
    ::find an ancestor of a descendant of node k depth D
end function
    
```

그림 9. 제안된 라우팅 알고리즘  
Fig. 9. Suggested Routing Algorithm

- C : Current node address
- D : Destination node address
- F : Family node address
- NL : Neighbor List
- DC : Depth of current node
- DD : Depth of Destination node
- Dc : Depth of center node
- i, j : 임의의 변수

제안된 알고리즘은 어드레싱 과정에서 Neighbor List가 미리 구성된다. 제일 처음 목적지 노드 D가 현재 노드의 NL에 있는지 확인하고, 있으면 다른 노드에게 Redirect\_ACK를 보내 다른 노드에서 중복으로 라우팅 하는 것을 방지한다. 그리고 다음 홉을 D로 설정한다. 현재 노드의 NL에 D가 존재하지 않은 경우, Find\_NearestNode()을 이용하여 계층적 경로에서 현재 노드의 NL에서 존재하고 우선순위가 높은 노드를 찾는다.

알고리즘의 동작은 그림 10을 통하여 설명한다.

소스 노드 S는 16비트 주소 67로 목적지 노드 D는 16비트 주소 20으로 할당 받은 노드이다. 먼저 기존의 HiLow의 라우팅 결과 67 - 16 - 3 - 0 - 4 - 20으로 총 5홉 거리의 경로가 지정된다. 실제 최단 거리가 2홉이므로 HiLow의 경로는 최적의 경로라고 할 수 없다.

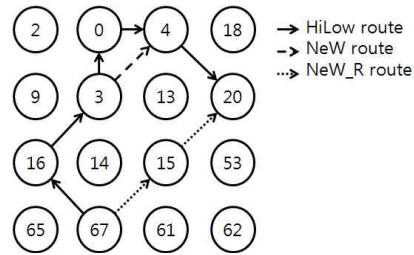


그림 10. 알고리즘 동작 예  
Fig. 10. The Example of Algorithm Operating

현재 노드 C의 NL에 계층적 경로 상의 노드들이 있는지 검사하는 제안하는 알고리즘은 현재 노드가 3인 경우 NL에서 0보다 우선순위가 높은 4를 발견하고 4로 라우팅 하여 1 홉을 줄인다.

Redirect\_ACK를 이용하는 경우는 67에서 16으로 보내는 패킷을 15가 overhearing하여 15의 NL에 20이 존재한다는 의미로 Redirect\_ACK를 보낸다. 주위노드에서 더 이상 20을 가는 경로를 찾지 못하도록 하고 20으로 라우팅 한다. 이렇게 하면 계층적 경로에 얽매이지 않고 새로운 최선의 경로 67 - 15 - 20을 찾게 되어 전체 3 홉을 줄인다.

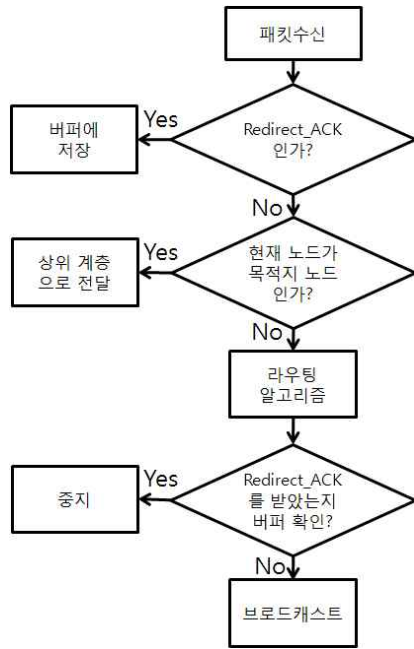


그림 11. 수신 패킷 처리 과정

Fig. 11. Time Chart of The Received Packet Process

그림 11은 제안하는 라우팅 알고리즘의 전반적인 순서를 알아본다. 먼저, 패킷수신을 받게 되면 프레임 형식이 Redirect\_ACK인지 확인한다.

Redirect\_ACK인 경우 버퍼에 저장 해두고, 아닌 경우는 다음 항목으로 넘어간다. 목적지 노드가 현재 노드인가는 현재노드에게 오는 패킷인지를 물어 보는

것을 의미한다. 그렇다면, 현재 노드의 상위계층으로 보내진다. 그렇지 않은 경우 전송해야 할 다음 홉을 제안하는 라우팅 알고리즘을 이용하여 찾는다.

전송 전에 필히 버퍼에 보내려는 패킷에 대한 Redirect\_ACK 유무를 먼저 검토하고 없으면 브로드캐스트한다. 있다면, 전송을 중지하고 패킷 수신 대기 상태가 된다.

4-4 경로 유지

3-2절에서 언급했던 문제점이 제안된 알고리즘에

서도 여전히 발생하지만, 제안된 라우팅 알고리즘에서는 이웃노드를 이용하여 좀 더 융통성 있게 경로를 선택하기 때문에 새로운 부모를 찾는 경우가 줄어들게 된다. 그래서 네트워크 내에 존재하는 모

든 노드의 NL을 수정해야 하는 이벤트가 줄어들 것으로 예상된다.

V. 시뮬레이션 및 성능 분석

이 절에서는 제안한 라우팅 알고리즘의 성능을 컴퓨터 시뮬레이션을 통하여 기존의 HiLow와 비교 분석하였다. 시뮬레이션은 C언어를 이용하였으며 시뮬레이션의 모델은 다음과 같다.

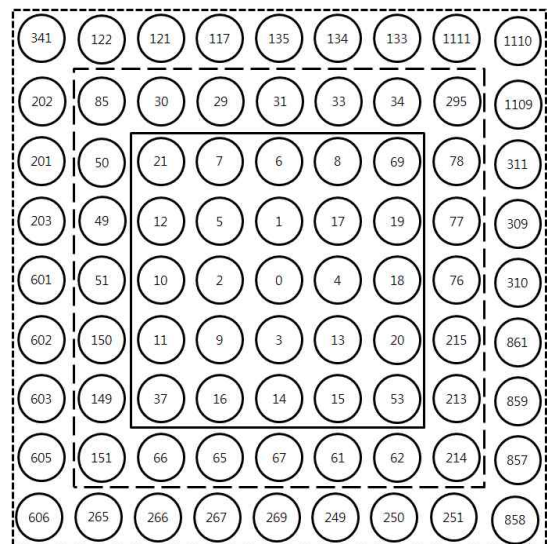


그림 12. 시뮬레이션 모델(5×5,7×7,9×9)

Fig.12. Simulation Model(5×5,7×7,9×9)

그림 12에서 볼 수 있듯이 각 노드들의 위치는 행과 열로 그리드 형태를 유지하고 코디네이터의 위치는 네트워크의 중앙이며, 주소는 0을 가진다. POS 반경은 대각선 노드까지라고 가정한다. 그림 13은 9×9 네트워크에서 각 노드의 계층적 구조를 나타낸 것이다. 어드레싱은 임의적으로 HiLow를 기반으로 행하였다. 다음 그림은 동적으로 할당받은 16비트 주소에 대한 계층적 구조를 나타낸다.

시뮬레이션의 시나리오는 노드들이 소스 노드와 목적지 노드가 될 모든 경우에 홉 수와 각 노드의 메시지 전송 횟수를 통하여 성능 분석에 활용하였다. 뿐만 아니라 센서 네트워크는 유동성이 강하므로 코디네이터의 위치와 노드의 수에 따른 여러 형태의 결과도 고려하였다.

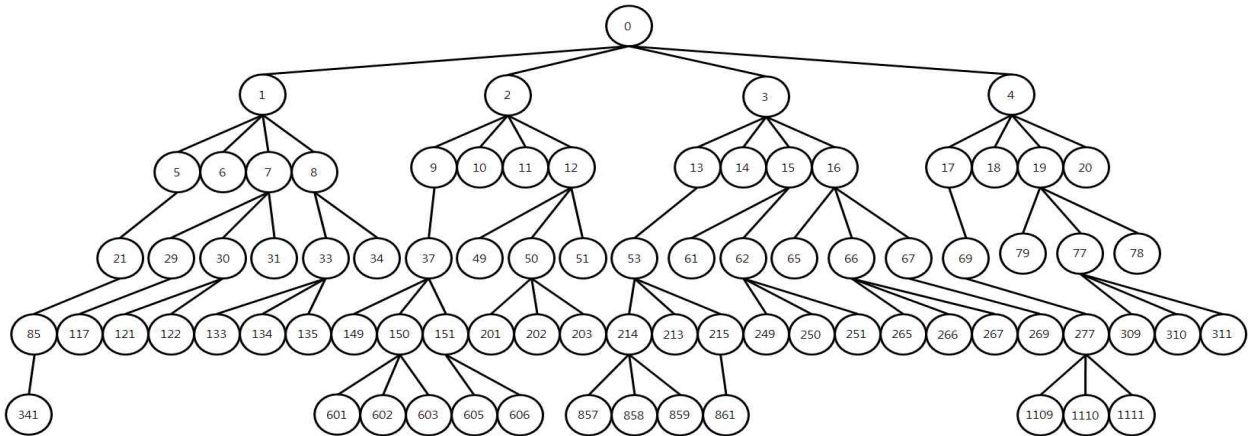


그림 13. 시뮬레이션 모델의 계층적 구조(9×9, 코디네이터의 위치:(4,4))  
 Fig. 13. Hierarchical Structure of Simulation Model(9×9,Coordinator's position is (4,4))

제안된 라우팅 알고리즘을 표에서 ‘제안 라우팅 (1)’로 표기하였고, 이 기법에서 Redirect\_ACK를 고려한 결과에 ‘제안 라우팅(2)’라고 표기하였다. 계산 결과는 소수 넷째자리에서 반올림하여 셋째자리까지 표기한다.

표 1. 실험 결과 (홉 수 비교, 7×7, (3,3))  
 Table 1. Result of The Simulation(Hop Count,7×7, (3,3))

	최단 홉수	HiLow	제안 라우팅(1)	제안 라우팅(2)
평균	3.219	4.555	3.429	3.318
분산	2.375	3.057	2.560	2.565
표준편차	1.541	1.749	1.600	1.602

표 1은 라우팅 기법에 따른 홉 수 비교이다. 비교할 대상은 HiLow의 결과와 제안하는 라우팅 알고리즘들의 결과이다. New 기법은 HiLow와 비교해 평균 1홉이 줄었다는 것을 알 수 있다. 제안 라우팅(2) 알고리즘이 이상적인 수치의 최단 홉 수에 가장 가까운 결과를 보인다.

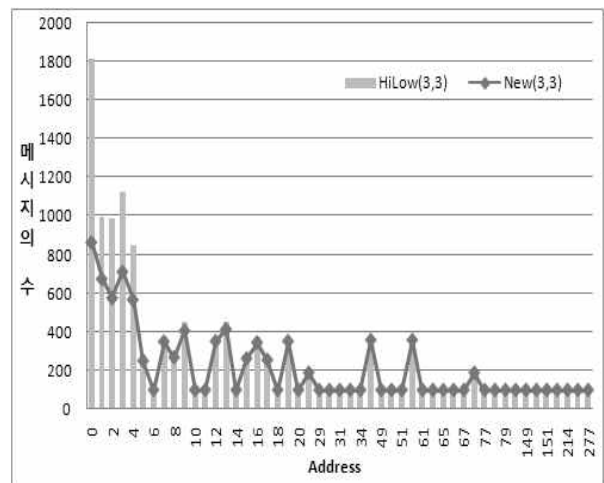


그림 14. 실험 결과 (각 노드별 전송횟수 비교, 7×7, (3,3))  
 Fig.14. Result of The Simulation Transmitted Message 7×7, (3,3)

각 노드가 메시지를 전송한 횟수를 그림 14를 통해서 HiLow와 제안하는 라우팅 알고리즘을 비교하고 있다. 0과 그 주위 노드에 집중되어 있는 양상이다. 하지만 제안하는 라우팅 알고리즘을 통해 0과 주위 노드에 부담을 덜어 주고 있다. 특히 노드 0은 약 55% 감소율을 보여주고 있다. 이로써 제안하는 라우팅 알고리즘은 각 노드들의 에너지 분배에도 효율적이라는 것을 알 수 있다.



표 3. 실험 결과(평균 홉 수 비교, 7×7, (3,3))  
 Table 3. Result of The Simulation(Average Hop Count, 7×7, (3,3))

	코디네이터 위치(좌표)	최단 홉수	HiLow	제안 라우팅 (1)	제안 라우팅 (2)
5 × 5	중앙 (2,2)	2.266	3.277	2.304	2.266
	모서리 (0,0)	2.266	3.693	2.502	2.342
	선중앙 (0,2)	2.266	3.930	2.723	2.460
7 × 7	중앙 (3,3)	3.219	4.555	3.429	3.318
	모서리 (0,0)	3.219	5.126	3.835	3.573
	선중앙 (0,3)	3.219	5.574	4.272	3.910
9 × 9	중앙 (4,4)	4.163	5.866	4.601	4.461
	모서리 (0,0)	4.163	6.548	5.163	4.876
	선중앙 (0,4)	4.163	7.271	5.846	5.454

표 3을 통해서 네트워크가 확장함에 따라 오차도 커진다는 결과를 얻을 수 있다. 또, 코디네이터의 위치에 따라 다른 결과를 확인할 수 있다. 제안하는 라우팅 알고리즘의 가장 좋은 효과를 나타내는 위치는 네트워크의 중앙에 위치할 때라는 정보를 얻을 수 있다.

VI. 결 론

본 논문에서는 계층적 라우팅에서 발생하는 문제점을 해결하고 6LoWPAN에 적합한 라우팅 알고리즘을 제안한다.

제안된 라우팅 알고리즘은 계층적 라우팅을 기반으로 하여 적은 메모리와 낮은 에너지 소모율을 가진다. 노드들의 관계에 Neighbor이라는 개념을 추가하고, 라우팅의 중복을 방지하는 Redirect\_ACK의 사용으로 계층적 라우팅의 최선의 경로가 보완되었다.

제안된 라우팅 알고리즘의 성능을 시뮬레이션을 통해 검증해 보았다. 7×7 시뮬레이션 모델에서 코디네이터의 위치가 (3,3)인 경우, 최단 홉 수와 약

41.5% 차이를 보인 HiLow 라우팅 알고리즘과 달리 제안된 알고리즘(1)과 (2)는 각각 약6.5%와 약3.1% 차이를 보이며 현저한 성능을 보였다. 또, 특정 노드에 집중되던 메시지 전송이 코디네이터와 그 주변 노드에서 현저히 줄어든 결과를 보았다. 하지만 제안된 알고리즘은 Neighbor List를 라우팅에 활용하기 때문에 알고리즘의 복잡도가 다소 증가될 수는 있다.

앞으로는 노드들의 위치를 랜덤 설정하고 하드웨어에 라우팅 알고리즘을 탑재하여 제안한 라우팅 알고리즘의 성능을 분석하고 문제점을 보완할 예정이다.

참 고 문 헌

- [1] 박준성, 임채성 외 5명, "IPv6 기반의 유비쿼터스 센서 네트워크(USN) 기술", *한국정보처리학회*, VOL. 13, NO. 04, pp. 79-87, July. 2006.
- [2] 임채성, Waleed Mansoor 외 4명, "IPv6 기반 센서 네트워크(6LoWPAN)을 위한 라우팅 프로토콜 기술", *전자공학회지*, 제33권, 제8호, pp.854-863, Aug. 2006.
- [3] N. Kushalnagar, G. Montenegro and C. Schumacher, "6LoWPAN : Overview, Assumptions, Problem Statement and Goals", *draft-ietf-6lowpan-problem -07(RFC4919)*, Feb, 2007.
- [4] K. Kim, Ed., J. Park, S. Daniel Pard, Ed. and J. Lee, "Hierarchical Routing over 6LoWPAN", *draft-daniel-6lowpan-HiLow-hierarchical-routing-00(work in progress)*, Jun, 2007.
- [5] 정부만, 김정근 외 8명, "2006년도 국내외 USN 산업동향 분석 연구", *한국정보사회진흥원*, Sep, 2006.
- [6] 유승화, 김희정, 김선호, 남동일, 유제택, 정명호, "RFID 및 USN에 IPv6 적용방안 및 활용 분야에 관한 연구", *한국전산원*, Dec, 2005.

### 강 문 경 (姜旻旻)



2007년 2월 : 창원대학교 전자공학과 (공학사)

2007년 3월 ~ 현재 : 창원대학교 전자공학과 석사과정

관심분야 : 유비쿼터스컴퓨팅, 센서네트워크

### 진 교 홍 (陳敎弘)



1991년 2월 : 부산대학교 컴퓨터공학과 (공학사)

1993년 2월 : 부산대학교 컴퓨터공학과 (공학석사)

1997년 8월 : 부산대학교 컴퓨터공학과 (공학박사)

1995 ~ 1997 : 부산대학교 컴퓨터 및 정보통신연구소 전문연구요원

1997 ~ 2000 : 국방과학연구소 선임연구원

2003 ~ 2004 : 동의대학교 멀티미디어공학과 조교수

2004 ~ 현재 : 창원대학교 전자공학과 부교수

관심분야 : 데이터통신, 센서네트워크, 유비쿼터스컴퓨팅, 홈네트워크