
전송 부하를 분산하는 무선 센서 네트워크 구축을 위한 TinyOS 기반 LMPR 구현

Implementation of LMPR on TinyOS for Wireless Sensor Network

오용택, 김동혁, 정국상, 최덕재
전남대학교 전산학과

Yongtaek Oh(yotzz@iat.chonnam.ac.kr), Pung-Hyeok Kim(dcwork@iat.chonnam.ac.kr),
Kugsang Jeong(handeum@iat.chonnam.ac.kr), Deokjai Choi(dchoi@chonnam.ac.kr)

요약

무선 센서 네트워크의 노드에서 측정된 데이터는 노드의 전송 거리의 제한 때문에 멀리 홉을 통해 베이스 노드에 전송된다. 또한 노드의 에너지가 한정되어 있기 때문에 무선 센서 네트워크의 수명을 연장하기 위해서는 각 노드의 에너지 소모를 가능한 균일하게 하여야 한다. Level based MultiPath Routing(LMPR)은 베이스 노드로 부터의 거리에 해당하는 노드의 레벨을 기반으로 무선 센서 네트워크를 자가 구축하고 데이터 처리 및 전송 부하를 각 센서 노드에 분산시키는 무선 센서 네트워크 라우팅 프로토콜이다. 본 논문에서는 TinyOS 기반으로 LMPR를 구현하고 실험을 통해 무선 센서 네트워크상에서 LMPR의 성능을 측정하였다. 실험 결과 LMPR이 최소 비용 방식으로 선택된 단일 경로로 데이터를 전송하는 프로토콜보다 데이터 처리 및 전송 부하를 약 4.6배 분산시켰다. LMPR을 사용하여 데이터 처리 및 전송 부하를 각 노드에 분산시켜 TinyOS를 기반으로 구성된 무선 센서 네트워크의 수명을 연장시킬 수 있을 것으로 기대한다.

■ 중심어 : 무선 센서 네트워크 라우팅 프로토콜 | TinyOS | 전송 부하 분산

Abstract

In Wireless Sensor Network(WSN) a sensor node transfers sensing data to the base-node through multi-hop because of the limited transmission range. Also because of the limited energy of the sensor node, the sensor nodes are required to consume their energy evenly to prolong the lifetime of the network.

LMPR is a routing protocol for WSN. LMPR configures the network autonomously based on level which is the depth from the base-node, and distributes the transmission and computation load of the network to each sensor node. This paper implements LMPR on TinyOS and experiments on the performance of LMPR in WSN. As the result, the average of the received rate of LMPR is 91.39% and LMPR distributes the load of the transmission and computation about 4.6 times compare to the shortest cost routing protocol.

We expect LMPR evenly distributes the transmission and computation load of the network to each node, and the lifetime of the network will be longer than it used to be.

■ keyword : Routing Protocol for Wireless Sensor Network | TinyOS | Transmission Load Balancing

접수번호 : #081120-004
접수일자 : 2008년 11월 20일

심사완료일 : 2008년 12월 07일
교신저자 : 오용택, e-mail : yotzz@iat.chonnam.ac.kr

1. 서론

무선 센서 네트워크는 군사 작전지역, 사람의 접근이 어려운 숲이나 강 같은 자연 환경, 또는 다리나 건물 등에 구성되어 각 응용에서 요구하는 다양한 데이터를 수집하여 특정 지역을 모니터링 하는데 활발히 이용되고 있다.

무선 센서 네트워크는 일반적으로 데이터를 수집하는 센서 노드와 센서 노드들로부터 데이터를 받아 사용자가 접근할 수 있는 장치에 전달하는 베이스 노드로 구성된다. 특정 지역의 데이터를 수집하는 무선 센서 네트워크 응용의 경우 센서 노드들이 설치된 후 센서 노드들의 움직임이 적고 베이스 노드는 사용자가 접근할 수 있는 일정한 곳에 고정되어 있는 경우가 많다. 무선 센서 네트워크를 구성하는 노드는 데이터를 수집하는 센서, 수집한 데이터를 처리하는 연산장치, 데이터를 저장하는 저장장치, 라디오를 매체로 데이터를 전송하고 수신하는 송·수신 장치로 구성되고 각 센서 노드들은 동일한 능력을 갖는다.

무선 센서 네트워크를 구성하는 노드의 전송 거리가 제한되어 있기 때문에 베이스 노드로부터 멀리 떨어진 센서 노드가 측정한 데이터를 베이스 노드에 전송하기 위해서는 멀티 홉을 거쳐 데이터를 전송해야 한다. 또한 각 센서 노드의 제한된 에너지 때문에 효율적으로 에너지를 소모하여 전체 센서 네트워크의 수명을 연장시킬 수 있는 라우팅 프로토콜이 필요하게 되었으며 이에 대한 연구가 많이 있어왔다[1][2][4][5].

본 논문에서는 무선 센서 네트워크 전송 부하 분산 라우팅 프로토콜인 LMPR을 TinyOS 기반으로 구현하여 무선 센서 네트워크를 구성하고 실험을 통해 무선 센서 네트워크상에서 LMPR의 성능을 측정 하였다. LMPR을 통해 TinyOS를 기반으로 구성된 센서 네트워크에서 각 노드의 데이터 처리 및 전송 부하를 분산시켜 네트워크 토폴로지에 따라 특정 노드가 네트워크로부터 빨리 단절 되는 것을 방지하여 네트워크 수명을 연장할 수 있을 것으로 기대된다.

서론에 이어 2장에서는 관련 연구에 대해서 살펴보고 3장에서는 TinyOS 기반 LMPR 구현에 대해서 살펴본

다. 이어서 4장에서는 실험 및 결과에 대해서 기술하고 마지막으로 5장에서 결론을 맺는다.

II. 관련 연구

1. 기존의 무선 센서 네트워크 라우팅 프로토콜 문제점

센서 노드는 지속적으로 에너지를 공급 받지 못하기 때문에 센서 노드의 한정된 에너지를 효율적으로 소모하는 라우팅 기법에 관한 많은 연구가 있어왔다. LEACH [1]은 무선 센서 네트워크의 전송 부하를 균일하게 하여 전체 센서 네트워크의 수명을 연장시켰으며 SPIN [2]은 센서 노드들 사이에서 메시지를 주고받아 중복된 데이터 전송을 줄이고 신뢰성을 향상시켰다. 하지만 LEACH의 경우 클러스터 헤더로 선택된 노드들이 베이스 노드와 통신 할 수 있어야 한다는 제약이 있으며 전송 손실이 많은 무선 통신의 특성을 고려할 때 SPIN과 같이 데이터 전송 전에 메시지를 주고받는 방식은 이 메시지가 손실될 경우 데이터를 전송 할 수 없다는 제약이 있다.

MultiHopLQ는 TinyOS[3]에서 제공하는 멀티 홉 라우팅 프로토콜이다. MultiHopLQ에서 수신 라디오 세기가 클수록 비용이 작고 각 센서 노드는 이 수신 라디오 세기를 측정하여 최소 비용 방식으로 라우팅 경로를 결정한다. 하지만 최소 비용 방식으로 라우팅 경로를 결정할 경우 무선 센서 네트워크를 구성하는 센서 노드의 위치에 따라 특정 노드에 데이터 처리 및 전송 부하가 집중되어 특정 노드의 에너지가 모두 소모될 경우 무선 센서 네트워크로부터 분리 될 수 있으며 이는 무선 센서 네트워크의 수명을 단축시키는 원인이 된다.

2. Level based MultiPath Routing

Level based MultiPath Routing(LMPR)[4]은 무선 센서 네트워크를 자가 구축 하여 데이터 처리 및 전송 부하를 무선 센서 네트워크를 구성하는 모든 노드에 분산시켜 멀티 홉을 통해 전송하고 손실이 많은 라디오의 특성을 감안하여 데이터 전송 시 필요한 메시지를

최소화한 무선 센서 네트워크 라우팅 프로토콜이다.

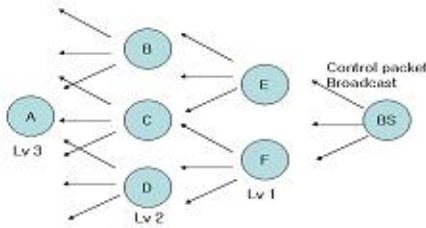


그림 1. Control Message 브로드캐스트

2.1 자가 네트워크 구성

LMPR은 무선 센서 네트워크에서 노드의 위상을 나타내는 레벨을 사용하여 네트워크를 자가 구성한다. 레벨은 통신 범위를 기준으로 베이스 노드로부터 노드가 떨어진 정도를 나타낸다. 예를 들어 베이스 노드의 레벨이 0일 때 베이스 노드와 직접 통신할 수 있는 노드들의 레벨은 1이 된다.

Control Message는 LMPR 무선 센서 네트워크를 구성하는데 사용된다. 베이스 노드는 자신의 레벨이 저장된 Control Message를 주기적으로 브로드캐스트한다. 새로운 Control Message를 수신한 노드는 자신의 레벨을 Control Message의 레벨에 1을 더한 값으로 수정하고 Control Message를 보낸 노드를 Parent Table에 추가한 후 Control Message에 자신의 레벨을 저장하여 다시 브로드캐스트한다. [그림 1]은 Control Message를 사용하여 무선 센서 네트워크를 자가 구성하는 과정을 보여준다.

무선 센서 네트워크에서는 통신 매체로 라디오를 사용하기 때문에 노드에서 브로드캐스트한 Control Message는 자신의 부모 노드를 포함한 모든 이웃 노드들에게 전달된다. 이때 자신의 자식 노드를 제외한 이웃 노드가 다시 Control Message를 브로드캐스트할 경우 Control Message는 루프에 빠지게 된다. 이를 해결하기 위해 LMPR에서는 부모 노드로부터 Control Message를 받았을 경우에만 Control Message를 다시 브로드캐스트한다. 또한 [그림 1]의 C 노드와 같이 부모 노드 E, F로부터 Control Message를 수신할 경우 Control Message의 수를 최소화하기 위해 중복 전송하지 않는다.

LMPR의 네트워크 구성 방법은 PEQ (Periodic, Event-Driven and Query-Based Protocol)[5]의 방법과 유사하지만 LMPR에서는 베이스 노드에서 최신 정보를 의미하는 Control Message의 번호를 증가하면서 Control Message를 주기적으로 브로드캐스트하는 것이 다르다. LMPR은 Control Message 주기를 조절하여 다양한 응용에 적용될 수 있다. 예를 들어 노드의 움직임이 많은 무선 센서 네트워크 응용의 경우 Control Message 주기를 짧게 하여 노드의 움직임을 무선 센서 네트워크 구성에 반영할 수 있다.

2.2 데이터 전송 및 Broken link 탐지

LMPR은 전송할 데이터 발생하면 무선 센서 네트워크를 구성할 때 작성된 각 노드의 Parent Table을 참조하여 각 노드의 부모 노드들 중 하나를 라운드 로빈 방식을 사용하여 next-hop으로 결정한다. [그림 2]와 같이 LMPR은 데이터 전송 시 마다 next-hop을 변경하여 전송함으로써 데이터 전송 및 처리 부하를 무선 센서 네트워크의 각 노드에 분산시킨다.

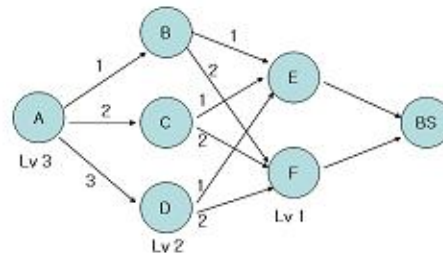


그림 2. Data Message 유니캐스트

LMPR은 부모 노드와 자식 노드 사이의 전송 에러를 카운트하여 에러 횟수가 한계 값을 초과하면 해당 부모 노드를 Parent Table에서 삭제하여 해당 부모 노드로 데이터를 전송하지 않게 함으로써 데이터 전송 에러를 감소시켰다.

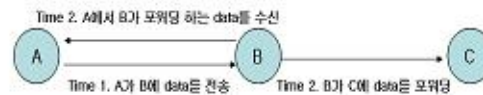


그림 3. Broken link 탐지 과정

[그림 3]은 자식 노드에서 부모 노드로 전송한 데이터의 에러 여부를 판별하는 과정을 보여준다. LMPR에서는 전송 매체로 라디오를 사용하기 때문에 자식 노드 A가 부모 노드인 B로 데이터를 전송하고 노드 B가 자신의 부모인 C로 데이터를 포워딩 할 때 B에서 보낸 데이터는 노드 A와 C에서 동시에 수신된다. 이런 라디오의 특성을 이용하여 만약 노드 A에서 노드 B로 자신의 데이터를 송신 한 후 정해진 시간 내에 노드 B가 노드 C로 포워딩한 데이터를 수신하지 못한다면 노드 A는 노드 B로 전송한 데이터가 손실 되었다고 판단하고 노드 B에 대한 전송 에러를 증가시킨다.

III. TinyOS 기반 LMPR 구현

본 논문에서는 LMPR을 TinyOS 기반으로 구현하였다. TinyOS는 임베디드 무선 센서 네트워크를 위해 고안된 운영체제로써 무선 센서 네트워크에 관련된 응용이나 알고리즘의 시뮬레이션 또는 구현에 많이 사용되고 있다. TinyOS로 구현된 응용이나 알고리즘은 Mca나 Telos 계열의 모트에 포팅되어 동작한다.

또한 LMPR은 TinyOS에서 사용되는 프로그래밍 언어인 NesC[6]로 구현되어 TelosB[7] 모트에 포팅되었다. TinyOS 응용은 Component들로 구성된다. Component의 종류에는 Configuration과 Module이 있으며 Configuration은 응용을 구성하는 Component들의 연결 관계를 나타내며 Module은 특정 작업을 구현한 것이다.

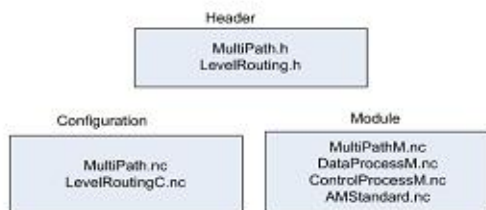


그림 4. LMPR 소스 파일

[그림 4]는 주기 마다 온도를 측정하여 LMPR을 라우팅 프로토콜로 사용하여 베이스 노드에 전송하는

MultiPath 응용과 LMPR 라우팅 프로토콜을 구현한 파일 구성이다.

MultiPath.nc와 MultiPathM.nc는 응용에 관련된 파일로써 MultiPath.nc는 모트에 장착된 센서 및 라디오 송-수신 장치인 CC2420, 전원 관련 장치 등에 관련된 Component들의 연결 관계가 나타나 있으며 MultiPathM.nc는 주기마다 온도를 측정하여 next-hop에 전송 하는 Module을 구현하였다.

LevelRoutingC.nc, DataProcessM.nc, ControlProcessM.nc, AMStandard.nc는 LMPR 라우팅 프로토콜을 구현한 파일로써 LevelRoutingC.nc는 DataProcessM.nc와 ControlProcessM.nc 모듈에서 사용하는 Component의 연결이 나타나 있다. 또한 ControlProcessM.nc는 Control Message를 송-수신하여 무선 센서 네트워크를 구성하는 Module이 구현되어 있으며 DataProcessM.nc는 LMPR 라우팅 프로토콜을 사용하여 next-hop을 선택하여 각 센서 노드에서 측정된 온도 값을 전송하거나 다른 센서 노드에서 측정된 온도 값을 포워딩 하는 Module이 구현되어 있다. AMStandard.nc는 MAC 계층에서 데이터를 송-수신하는 모듈이 구현되어 있다.

1. LMPR 메시지 및 Parent Table

일반적으로 무선 센서 네트워크에서는 라디오를 전송 매체로 사용하기 때문에 전파 간섭 등 여러 요인으로 인하여 메시지 손실률이 높다. AODV[8]나 SPIN과 같은 프로토콜은 3 hand-shake 방식을 사용하여 네트워크를 구성하고 데이터를 전송한다. 이렇게 네트워크 구성 및 데이터 전송에 필요한 메시지가 많을 수록 이 메시지가 손실될 확률이 증가한다. 따라서 LMPR에서는 메시지 손실 확률을 줄이기 위해 Control Message와 Data Message만을 사용하여 네트워크를 구성하고 데이터를 전송한다. Data Message는 [그림 4]의 MultiPath.h에 선언되어 있으며 Control Message와 Parent Table는 [그림 4]의 LevelRouting.h에 선언되어 있다.

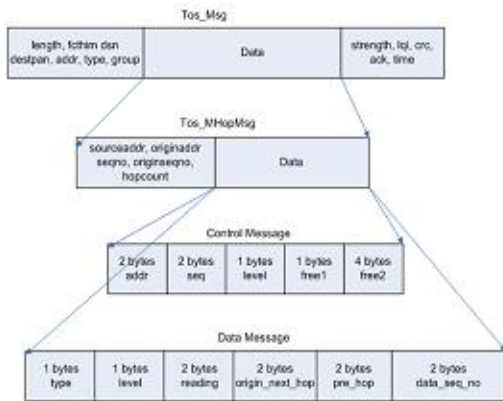


그림 5. TinyOS & LMPR Message

[그림 5]는 TinyOS에서 사용하는 메시지 구조를 보여준다. LMPR은 기본적으로 TinyOS 메시지 구조를 사용하며 Control Message와 Data Message를 추가하였다.

Control Message는 LMPR 네트워크를 구성하는데 사용된다. 베이스 노드는 Control Message를 주기적으로 생성하여 브로드캐스트 한다. Control Message를 받은 센서 노드는 Control Message를 수정한 후 다시 브로드캐스트 한다. Control Message는 [그림 6]과 같이 구성되며 각 필드의 의미는 다음과 같다.

- addr

Control Message를 브로드캐스트 하는 센서 노드의 주소

- seq

베이스 노드에서 보낸 Control Message의 순서 (베이스 노드에서만 증가)

- level

Control Message를 브로드캐스트 하는 노드의 Level



그림 6. Control Message 구조

Data Message는 센서 노드에서 측정된 데이터나 포

워딩할 데이터를 전송 할때 쓰인다. Data Message는 [그림 7]과 같이 구성되어 있으며 각 필드의 의미는 다음과 같다.

- type

data 메시지의 type

- level

센싱 정보를 보낸 노드의 level

- reading

조도 센서에서 측정한 조도 값

- origin_next_hop

데이터를 센싱한 노드의 next hop ID

- pre_hop

Data Message를 보낸 자식 노드 ID

- data_seq_no

센싱 데이터 순서



그림 7. Data Message 구조

Parent Table은 노드의 제한된 메모리 용량을 고려하여 유지되는 부모 노드의 정보를 최소화 하였으며 새로운 Control Message를 받거나 센서 노드의 레벨이 변할 때 마다 업데이트 된다.

또한 동적 메모리 할당이 불가능한 TinyOS의 특성 때문에 Parent Table의 최대 크기를 미리 정의 하였다. Parent Table의 구성요소는 다음과 같다.

- addr

각 부모의 주소

- err

각 부모의 주소로 전송 시 에러 횟수

• flag

부모의 주소로 전송 시 전송 성공 여부

2. Control Message 프로세스

Control Message 프로세스는 무선 센서 네트워크를 구성한다. Control Message 프로세스는 베이스 노드와 센서 노드가 각각 다르게 동작한다. 베이스 노드에서는 주기마다 Control Message를 생성하여 브로드캐스트 하며 센서 노드는 Control Message를 수신하여 부모 노드로부터 온 것인지 확인 후 Parent Table을 작성하고 Control Message를 다시 브로드캐스트 한다.

그림 8은 Control Message 프로세스의 함수 호출도이며 각 함수는 다음과 같이 동작한다.

• 베이스 노드

- **ControlProcessM:Timer.start(random * 1024)**
임의의 시간동안 타이머를 동작
- **ControlProcessM:Timer.fired()**
타이머의 임의의 시간이 지나면 호출되는 이벤트 함수
- **ControlProcessM:SendRouteTask()**
Control Message를 작성하여 MAC 계층으로 전송
- **AMStandard:SendMsg.Send(id)(Broadcast)**
Control Message를 브로드캐스트 함
- **ntdProcessM:Timer.start(BEACON_PERIOD)**
Control Message 주기 만큼 타이머를 동작

• 센서 노드

- **ControlProcessM:ReceiveMsg.receive(pMsg, id)**
MAC 계층으로부터 Control Message를 수신하고 부모 노드로부터 온 것인지 검사 후 Parent Table에 추가 함
- **ControlProcessM:RandomTimer.start(random)**
임의의 시간동안 타이머를 동작
- **ControlProcessM:RandomTimer.fired()**
타이머의 임의의 시간이 지나면 호출되는 이벤트 함수

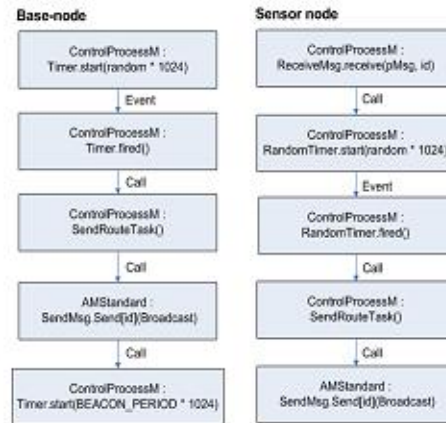


그림 8. Control Message 프로세스 함수 호출도

3. Data Message 프로세스

3.1 Sensing Data 프로세스

Sensing Data 프로세스는 MultiPath 응용에서 온도를 주기적으로 측정하여 LMPR 라우팅 기법을 이용하여 next-hop으로 전송하고 동시에 next-hop에 대한 전송 에러를 검사한다. [그림 9는 Sensing Data 프로세스의 함수 호출도이며 주요 함수 및 변수는 다음과 같다.

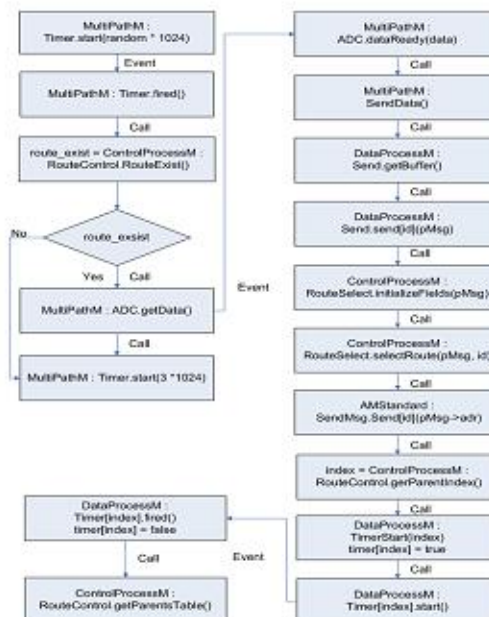


그림 9. Sensing Data 프로세스 함수 호출도

- **ControlProcessMRouteControl.RouteExist()**
현재 Parent Table에 부모 노드가 존재하는지 검사
- **route-exist**
현재 Parent Table에 부모 노드가 존재하면 참
- **MultiPathMADC.getData()**
센서로부터 데이터를 얻는 함수
- **MultiPathMADC.dataReady(data)**
센서로부터 데이터가 준비되면 호출되는 이벤트 함수
- **MultiPathMSendData()**
측정한 온도 값이 삽입된 Data Message를 작성하여 네트워크 계층으로 전송하는 함수
- **DataProcessMSend.send(id)(pMsg)**
응용으로부터 수신한 Data Message를 MAC 계층으로 전송하는 함수
- **RouteSelect.selectRoute(pMsg, id)**
Parent Table에서 next-hop을 선택 함
- **ControlProcessMRouteControl.getParentIndex()**
현재 next-hop의 Parent Table에서의 index를 반환함
- **Timer[index].start()**
현재 next-hop에서 포워딩하는 Data Message를 시간을 나타내는 타이머
- **timer[index]**
현재 next-hop에 해당하는 Timer의 동작 여부를 나타내는 지표
- **Timer[index].fired()**
Timer[index] 가 만료되면 호출되는 이벤트 함수로써 Parent Table에서 현재 next-hop의 flag 필드 값을 검사하여 flag가 0이면 현재 next-hop에 대한 에러 값을 증가시킴
- **ControlProcessMRouteControl.getParentTable()**
현재 Parent Table의 포인터를 반환 함

3.2 Forwarding Data 프로세스

TinyOS에서는 TOS_Msg의 addr 값과 메시지를 수신한 센서 노드의 ID와 비교하여 그 값이 같으면 메시지를 수신한다. 하지만 LMPF에서는 전송 에러를 검사

하기 위해 센서 노드에서 포워딩한 Data Message를 next-hop과 이전에 Data Message를 전송한 pre-hop에서 동시에 수신해야 한다. 따라서 메시지를 수신한 센서 노드에서 TOS_Msg의 addr 필드 값과 Data Message의 pre_hop 필드 값을 자신의 ID와 비교하여 둘 중 하나의 값이 같을 경우 메시지를 수신하도록 구현하였다.

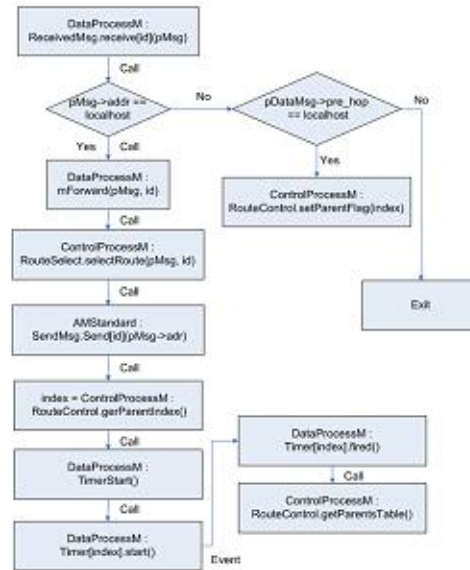


그림 10. Forwarding Data 프로세스 함수 호출도

[그림 10]은 Forwarding Data 프로세스의 함수 호출 도이며 next-hop과 pre-hop에서 Data Message를 수신 하였을 경우 각각 다르게 처리하는 것을 보여준다. 주 요 함수 및 변수의 의미는 다음과 같다.

- **DataProcessMReceiveMsg.receive(id)(pMsg)**
MAC 계층에서 Data Message를 수신하면 호출 되는 이벤트 함수
- **DataProcessMmForward(pMsg, id)**
수신한 Data Message가 자식 노드로부터 온 것이 면 Parent Table에서 next-hop을 선택하여 포워딩 하는 함수
- **ControlProcessMRouteControl.setParentFlag(index)**
pre-hop에서 정해진 시간 내에 Data Message를 수

신하면 현재 next-hop의 Parent Table의 flag 필드 값을 1로 설정하는 함수

IV. 실험 및 성능 평가

1. 실험 방법

MultiHopLQI와 LMPR을 라우팅 프로토콜로 사용하여 매 3초마다 온도를 측정하여 베이스 노드에 100회 전송하는 Surge와 MultiPath 응용을 TelosB 모뎀에 각각 포팅 한 후 장애물이 없는 환경에서 [그림 11]과 같이 배치하였다. 또한 각 노드에서 측정된 온도 값과 관계 식 1의 SN값과 TN값을 베이스 노드로 전송하여 식 (1)의 각 노드 수신율과 TN의 분산을 산출 하였다.

$$\begin{aligned} & \text{신성 데이터 전송 횟수} && (1) \\ & \text{포워딩 데이터 전송 횟수} \\ & \text{에서 수신한 신성 데이터 횟수} \\ & \times && \text{는 센서 노드} \\ & \text{수신율} && \end{aligned}$$

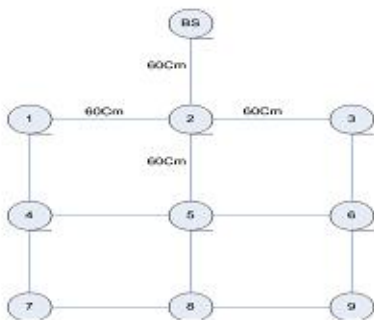


그림 11. 센서 노드 배치

TN값이 센서 노드에서 측정된 데이터 전송 횟수와 데이터 포워딩 횟수를 의미하기 때문에 각 노드 TN의 분산 값은 무선 센서 네트워크를 구성하는 센서 노드의 전송 부하를 의미한다. 즉 특정 노드에 전송 부하가 집중 될 경우 분산 값이 크며 전체 센서 노드에 전송 부하가 분산 될 경우 분산 값이 작다. 이렇게 각 센서 노드 TN의 분산 값을 이용하여 무선 센서 네트워크에서 MultiHopLQI와 LMPR 라우팅 프로토콜에 따른 전송 부하 정도를 정량화하여 비교하였다.

2. 결과 및 분석

[그림 12]와 [그림 13]은 MultiHopLQI 라우팅 프로토콜로 사용한 Surge 응용과 LMPR을 라우팅 프로토콜로 사용한 MultiPath 응용의 동작과 각 센서 노드의 수신율 및 전송 횟수의 분산을 각각 보여준다. 이 모니터링 프로그램은 Surge 응용 모니터링 프로그램을 MultiPath 응용도 모니터링 할 수 있도록 수정 된 것이다. 이 모니터링 프로그램의 상단부에는 센서 네트워크의 토폴로지를 나타내며 하단부 왼쪽 그래프는 각 센서 노드에서 측정하여 전송한 온도값의 수신율을 나타내며 오른쪽 그래프는 각 센서 노드에서 측정된 온도값을 나타낸다.



그림 12. Surge 응용 모니터링 실행 화면

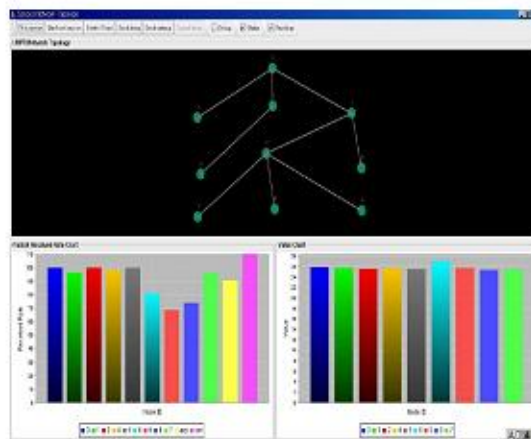


그림 13. MultiPath 응용 모니터링 실행 화면

MultiHopLQI 라우팅 프로토콜의 경우 최소 비용 결정 방식에 의해 경로를 결정하기 때문에 일정한 경로로 각 노드에서 측정된 데이터를 전송하였으며 LMPR 라우팅 프로토콜의 경우 각 노드에서 측정된 데이터를 전송 할 때마다 Parent Table에서 라운드 로빈 방식으로 next-hop을 결정하기 때문에 경로가 변경 되었다.

표 1. Surge 응용 전송 횟수 분산

Surge Transmission											
	ID 1	ID 2	ID 3	ID 4	ID 5	ID 6	ID 7	ID 8	ID 9	평균	분산
1회	505	100	397	223	343	100	100	100	162	221.22	19628.00
2회	421	100	456	197	366	103	100	100	133	224.00	23992.00
3회	280	100	520	100	341	103	100	100	100	193.78	20860.00
4회	718	125	172	407	318	104	100	100	100	238.33	39795.33
5회	678	100	370	251	340	100	100	100	100	237.67	35201.78
6회	542	100	300	327	373	100	100	100	100	226.89	24957.21
7회	733	100	112	492	394	100	100	100	154	252.78	47337.73
8회	797	175	245	414	373	100	100	100	100	267.56	48231.80
9회	733	100	179	431	342	100	100	100	100	242.78	43509.51
10회	589	100	196	421	334	100	100	100	108	226.89	29989.32
최대값	797	175	520	492	394	104	100	100	162	267.56	48231.80
최소값	280	100	112	100	318	100	100	100	100	193.78	19628.00
평균	600.7	110.1	294.5	324.3	352.4	101	100	100	115.7	223.19	33953.17

표 2. MultiPath 응용 전송 횟수 분산

MultiPath Transmission											
	ID 1	ID 2	ID 3	ID 4	ID 5	ID 6	ID 7	ID 8	ID 9	평균	분산
1회	289	335	100	150	247	283	100	100	100	179.22	6908.40
2회	154	157	100	101	323	207	103	100	100	149.44	5038.91
3회	169	160	100	100	312	196	100	100	100	147.33	4488.69
4회	189	164	100	105	274	215	102	100	100	147.56	3544.02
5회	172	167	100	100	344	191	100	100	100	151.56	5760.91
6회	270	535	105	232	294	102	100	196	249	231.44	16570.25
7회	189	182	435	205	218	114	100	102	100	182.78	10039.84
8회	197	171	453	201	236	118	100	100	100	186.22	11241.28
9회	238	196	344	201	164	108	100	139	100	176.67	5642.00
10회	199	230	204	195	289	100	101	115	117	172.36	2875.54
최대값	270	535	463	232	344	283	103	196	249	231	16570.25
최소값	154	157	100	100	164	102	100	100	100	147	2875.54
평균	196.11	229.67	204.11	155	268	186	100.99	115.22	116.56	172.36	7216.90

[표 1][표 2]는 Surge와 MultiPath 응용이 포팅된 각 노드의 TN(식1)을 베이스 노드에서 10회 반복 측정하여 분산 값을 산출한 결과이다. MultiHopLQI 라우팅 프로토콜을 사용한 Surge 응용의 경우 분산 최대값이 48,231.80이며 최소값이 19,628 그리고 평균이 33,053.17로 산출 되었다. LMPR 라우팅 프로토콜을 사용한 MultiPath 응용의 경우 분산 최대값이 16,570.25이며 최

소값이 2,875.54 그리고 평균이 7,216.90으로 산출 되었다. 실험 결과 LMPR 라우팅 프로토콜이 MultiHopLQI 라우팅 프로토콜에 비해 무선 센서 네트워크에서 약 4.6배 정도 데이터 처리 및 전송 부하를 각 노드에 분산 시켰다.

표 3. Surge 응용 수신율

Surge Received Rate										
	ID 1	ID 2	ID 3	ID 4	ID 5	ID 6	ID 7	ID 8	ID 9	평균
1회	95	94	95	96	93	76	97	91	89	91.78
2회	91	90	92	91	92	71	90	93	92	89.11
3회	91	85	99	94	90	90	83	82	85	88.78
4회	94	85	95	87	84	88	61	82	85	84.56
5회	91	90	90	92	91	86	82	82	81	87.22
6회	90	80	98	96	97	61	91	67	89	85.44
7회	94	85	85	87	86	81	99	100	99	90.67
8회	98	95	99	96	96	85	94	93	92	94.22
9회	94	86	84	87	85	78	82	80	82	84.22
10회	93	90	95	86	90	83	93	83	83	88.44
최대값	98	95	99	96	97	90	99	100	99	94.22
최소값	90	80	84	86	84	61	61	67	81	84.22
평균	93.1	88	93.2	91.2	90.4	79.9	87.2	85.3	87.7	88.44

표 4. MultiPath 응용 수신율

MultiPath Received Rate										
	ID 1	ID 2	ID 3	ID 4	ID 5	ID 6	ID 7	ID 8	ID 9	평균
1회	96	100	90	98	98	98	86	96	85	91.89
2회	100	99	95	99	96	100	89	94	82	94.09
3회	100	99	93	93	70	100	78	67	75	86.11
4회	100	100	97	97	95	100	86	90	81	94.00
5회	100	100	99	100	97	100	87	96	87	96.22
6회	100	100	99	91	100	94	92	75	85	90.67
7회	97	100	100	99	99	76	96	84	85	90.67
8회	99	100	100	100	99	74	96	93	71	92.44
9회	100	99	98	99	93	78	83	58	57	85.00
10회	99	100	97	97	94	91	88	84	72	91.32
최대값	100	100	100	100	100	100	96	96	87	96
최소값	96	99	90	91	70	74	78	58	57	85
평균	99.11	99.67	95.78	97.33	94.11	91.11	88.11	83.67	72.00	91.32

[표 3][표 4]는 Surge와 MultiPath 응용이 포팅된 각 센서 노드의 수신율을 베이스 노드에서 10회 반복 측정 한 결과를 보여 준다. Surge의 경우 전체 노드의 최대 값이 94.22%, 최소값이 84.22% 그리고 평균이 88.44%로 측정 되었다. MultiPath 응용의 경우 최대값이 96%,

최소값이 85%, 그리고 평균이 91.32%로 측정 되었다. Surge의 경우 무선 센서 네트워크를 구성 할 때 경로가 결정 되지 않아도 측정된 데이터를 전송하기 때문에 데이터 손실이 발생이 발생하여 MultiPath보다 수신율이 2.88% 낮게 측정 되었다.

V. 결 론

본 논문에서는 무선 센서 네트워크의 각 노드에서 다중 경로를 설정하여 그 중 하나를 선택하여 멀티 홉을 통해 측정된 데이터를 베이스 노드에 전송하는 Level based MultiPath Routing(LMPR)을 TinyOS 기반으로 구현하였다. LMPR을 구현하여 실험한 결과 최소 비용 방식으로 결정된 단일 경로로 데이터를 전송하는 라우팅 프로토콜에 비해 약 4.6배 정도 데이터 처리 및 전송 부하를 분산시켰다.

하지만 현재 LMPR은 센서 노드로부터 데이터를 수집할 뿐 센서 노드를 통제하는 명령을 센서 노드에 보낼 수는 없다. 따라서 향후 연구에서 명령을 데이터 처리 및 전송 부하나 에너지 소모 면에서 효율적으로 센서 노드에 전송할 수 있도록 LMPR을 개량 할 것이다.

LMPR을 통해 TinyOS 기반으로 구성된 센서 네트워크에서 각 노드의 데이터 처리 및 전송 부하를 균일하게 분산시켜 네트워크 토폴로지에 따라 특정 노드가 네트워크로부터 단절 되는 것을 방지 하여 무선 센서 네트워크의 수명을 연장 할 수 있을 것으로 기대한다.

참 고 문 헌

- [1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.
- [2] J. Kulik, W. Heinzelman, and H. Balakrishnan,

"Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks," Wireless Networks, Vol.8, pp.169-185, 2002.

- [3] <http://www.tinyos.net>
- [4] Y. Oh, P. Kim, N. Bao, and D. Choi, "A Load balancing Level based Multi-Path Routing Protocol for Wireless Sensor Network," International Conference on Asian Language Processing and Information Technology, 2006.
- [5] A. Boukerche, R. Pazzi, and R. Araujo, "A Fast and Reliable Protocol for Wireless Sensor Networks in Critical Conditions Monitoring Applications," MSWIM04, 2004.
- [6] <http://www.tinyos.net/tinyos-1.x/doc/nesc/ref.pdf>
- [7] <http://www.moteiv.com/products-tmotesky.php>
- [8] C. Perkins, E. B. Foyer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, 2003.

저 자 소 개

오 용 택(Yongtaek Oh)

준회원



- 2005년 2월 : 전남대학교 전산학과 졸업 (이학사)
- 2005년 3월 ~ 현재 : 전남대학교 전산학과 대학원 석사과정
- <관심분야> : 무선 센서 네트워크 라우팅 프로토콜, 임베디드 시스템, 네트워크

김 풍 혁(Pung-Hyeok Kim)

준회원



- 2005년 2월 : 전남대학교 전산학과 졸업 (이학사)
- 2005년 3월 ~ 현재 : 전남대학교 전산학과 대학원 석사 과정
- <관심분야> : 임베디드 시스템, 센서 네트워크, 멀티미디어 전송

정 국 상(Kugsang Jeong)

정회원



- 1999년 2월 : 전남대학교 전산학과 (이학사)
- 2001년 2월 : 전남대학교 전산학과 대학원(이학석사)
- 2001년 ~ 현재 : 전남대학교 전산학과 대학원 박사 과정

<관심분야> : 유비쿼터스 미들웨어 시스템, 센서 네트워크, 네트워크

최 덕 재(Deokjai Choi)

정회원



- 1982년 2월 : 서울대학교 전자공학과 (공학사)
- 1984년 2월 : 한국과학기술원 전산학과 (공학석사)
- 1995년 2월 : Univ. of Missouri-Kansas City (공학박사)

•1995년 ~ 현재 : 전남대학교 전자컴퓨터공학부 교수

<관심분야> : 네트워크, 무선 센서 네트워크