

무선 센서 네트워크의 링크 품질을 모니터링하는 응용 구현

노태호^o, 정광수

광운대학교 전자공학부 컴퓨터통신연구실

thno^o@adams.kw.ac.kr, kchung@kw.ac.kr

An Application Implementation Monitoring the Link Quality of Wireless Sensor Networks

Taeho Roh^o, Kwangsue Chung

School of Electronics Engineering, Kwangwoon Univ.

Abstract

센서 네트워크는 주로 물리적인 공간의 모니터링이나 위치 추적과 같은 주변의 정보를 얻고자 하는 환경에 사용되며, 이러한 정보는 비대칭적이고 비신뢰적인 무선 링크로 인해 불필요한 재전송을 요구하고 많은 손실이 발생한다. 이 때문에 신뢰적이고 에너지 효율적인 링크를 선택하기 위해 RSSI(Received Signal Strength Indicator), LQI(Link Quality Indicator)를 이용하여 무선 링크에 대한 품질을 추정하는 기법이 필요하다. 본 논문에서는 그 일차적인 단계로써 단일 홉 무선 센서 네트워크에서 MICAZ에 구현된 RSSI, LQI 값을 이용하여 노드간 무선 링크 품질을 모니터링하는 응용을 구현하였다. 구현 시나리오는 무선 링크의 비대칭성을 고려하기 위해, 순방향 링크 품질의 경우 센서 플랫폼이 BS로부터 수신한 요구 메시지의 RSSI, LQI 값을 응답 메시지에 캡슐화하여 BS로 전송하도록 하였고, 역방향 링크 품질의 경우 BS가 센서 플랫폼으로부터 수신한 응답 메시지의 RSSI, LQI 값을 기반으로 하였다. 또한 BS로 취합된 이러한 두 링크 품질을 PC상에 시각적으로 표시하기 위해 자바 기반의 링크 품질을 모니터링하는 응용을 구현하였다. 간단한 실험을 통해 RSSI, LQI로 얻은 무선 링크의 품질과 특성을 확인하였다.

1. 서론

최근 정보 통신 기술의 비약적인 발전은 기존의 단순 계산기로서의 컴퓨터가 아닌 초소형의 정보 단말로의 컴퓨터로 진화하여 더욱 더 인간의 생활에 밀접한 영향을 주고 있다. 이러한 기술의 진보는 유비쿼터스 컴퓨팅이라는 새로운 정보 통신 혁명을 야기하게 되었고, 그로인해 센서 네트워크의 필요성이 제기되고 있다.

센서 네트워크란 여러 개의 저-전력 센서 노드가 무선 통신을 통해 서로 협력하여 특정 작업을 수행하는 새로운 컴퓨팅 분야이다. 센서 네트워크는 주로 물리적인 공간의 모니터링이나 위치 추적과 같은 주변의 정보를 얻고자 하는 환경에 사용되며, 이러한 정보는 무선 링크를 통해 다른 이웃 노드로 전달된다.

이러한 무선 링크는 멀티-경로, 페이딩, 간섭으로 인해 불안정성과 비예측성을 갖으며, 이전의 무선 센서 네트워크를 위한 연구는 링크의 대칭성을 가정하였다. 하지만 현재 무선 링크에 대한 실험적인 연구들은 그러한 비현실적인 가정은 링크의 현실성을 반영하지 못한다는 성능 결과와 함께 무선 링크의 비대칭성을 실험적으로 증명하였다[1, 2]. 이러한 성향을 갖는 무선 링크로 정보를 전송하는 것은 많은 패킷 손실을 야기하며 불필요한 재전송으로 인해 신뢰성과 에너지 효율성을 저하시킨다.

특히, 이러한 점은 무선 멀티 홉 환경에서 다음 홉을 결정하는 라우팅 계층에 매우 중요하게 작용한다. 그 이유는 동적으로 변하는 무선 링크로 인해 에너지 효율적이고 신뢰적인 링크를 선택하는 것이 상당히 어렵기 때문이다. 따라서 이러한 링크를 선택하기 위해 라우팅 계층은 하위 계층의 하드웨어 혹은 소프트웨어적으로 처리된 정보를 이용하여 무선 링크에 대한 품질을 추정하는 기법이 반드시 필요하다.

무선 링크의 품질을 추정하는 기법중의 한 가지는 링크 품질을 명시적으로 나타내는 지표(indicators)를 사용하는 것이다. 현재 MICAZ[3], Telos[4]와 같은 센서 플랫폼들은 그 지표로써 RSSI, LQI를 제공한다. 특히 MICAZ의 경우, RSSI는 CC2420[5] 레디오 칩상에 하드웨어적으로 구현되었고, LQI는 MAC 계층의 소프트웨어 차원으로 구현되었다. 현재 이러한 지표들을 이용하여 무선 센서 네트워크를 위한 연구가 활발하게 진행되고 있다[6, 7].

본 논문에서는 그 일차적인 단계로써 단일 홉 무선 센서 네트워크에서 MICAZ에 구현된 RSSI, LQI 값을 이용하여 노드간 무선 링크 품질을 모니터링하는 응용을 구현하였다. 구현된 응용은 MICAZ의 RSSI, LQI 값을 구하는 모듈과 시리얼 인터페이스를 통해 그 값을 BS와 연결된 PC에 표시하는 부분으로 구성된다. 또한 구현 시나리오는 무선 링크의 비대칭성을 고려하기 위해, 순방향 링크 품질의 경우 센서 플랫폼이 BS로부터 수신한 요구 메시지의 RSSI, LQI 값을 응답 메시지에 캡슐화하여 BS로 전송하도록 하였고, 역방향 링크 품질의 경우 BS가

* 본 연구는 한국과학재단 특정기초연구[R01-2005-0000-10934-0(2006)]의 지원에 의해 수행되었음.

센서 플랫폼으로부터 수신한 응답 메시지의 RSSI, LQI 값을 기반으로 하였다. BS로 취합된 이러한 두 링크 품질을 PC상에 시각적으로 표시하기 위해 자바 기반의 링크 품질을 모니터링하는 응용을 구현하였다.

본 논문의 2장에서는 CC2420에서 링크 품질을 나타내는 지표(Link Quality Indicators)에 대해 기술하였고, 3장에서는 무선 링크 품질을 모니터링하는 응용에 대한 구현 내용을 기술하였다. 4장에서는 간단한 실험을 통해 RSSI, LQI로 얻은 무선 링크의 품질과 특성을 확인하였으며 5장에서는 결론 및 향후 과제에 대해 기술하였다.

2. 링크 품질을 나타내는 지표

본 절에서는 CC2420의 데이터시트에 기술된 내용을 기반으로 링크를 분류하고 선택하기 위해 사용되는 MICAZ의 RSSI, LQI에 대해서 살펴보고자 한다.

그림 2.1은 수신된 신호의 복조화가 순차적으로 수행되는 블록 다이어그램을 간단하게 나타낸 것이다 [5]. RSSI는 RSSI Generator에서 필터링된 신호를 입력받아 구할 수 있으며, LQI는 수신된 신호 중에서 데이터 필터링을 통해 실제 MAC 데이터와 미리 정의된 칩 시퀀스를 맵핑시켜 그 상관관계를 기반으로 계산된다.

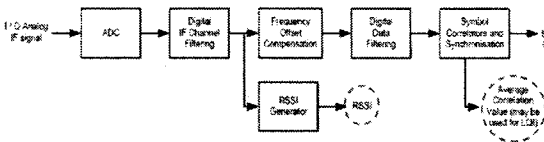


그림 2.1 복조화 블록 다이어그램

RSSI는 채널의 신호 레벨을 추정하기 위해 사용되며, 8 심볼 기간(128us)내에 수신된 신호를 평균하여 계산된다. 그 계산된 값은 8비트의 RSSI_VAL 레지스터에 저장되어 수신된 신호 파워 레벨(P)을 구하기 위해 사용된다. P는 식 (1)과 같으며 데시벨미터로 환산된다. RSSI_OFFSET은 실험적으로 얻은 -45를 사용한다.

$$P = \text{RSSI_VAL} + \text{RSSI_OFFSET} \text{ [dBm]} \quad (1)$$

LQI는 수신된 데이터의 품질과 세기의 특징을 나타내는 지표이다. 이 기법은 각 수신된 패킷의 8 심볼에 대한 에러율을 기반으로 수행된다. 즉, 각 수신된 패킷과 미리 맵핑된 칩 시퀀스와의 상관관계를 살펴본 후, 50에서 110까지의 범위에서 평균 상관 값(CORR)을 만들어 낸다 [8]. 실제 사용자가 확인 가능한 LQI 값은 발생된 CORR을 MAC 소프트웨어에 의해 0에서 255까지의 정수 범위로 변환함으로써 얻는다. 최소 0과 최대 255값은 신호 품질의 가장 낮고 높음을 의미하게 된다.

3. 응용 구현 내용

본 절에서는 단일 홉 무선 센서 네트워크에서 MICAZ에 구현된 RSSI, LQI 값을 이용하여 노드간 무선 링크 품질을 모니터링하는 응용에 대해서 살펴보고자 한다. 구현된 응용은 MICAZ의 RSSI, LQI 값을 구하는 모듈과 시리얼 인터페이스를 통해 그 값을 BS와 연결된 PC에 표시하는

부분으로 구성된다.

3.1 구현 시나리오

구현 시나리오는 무선 링크의 비대칭성을 고려하여 순방향, 역방향 링크의 품질을 확인하는 것이다. 순방향 링크 품질은 수신된 BS의 요구 메시지에 대한 RSSI, LQI 값이 되고, 역방향 링크 품질은 그 요구 메시지에 대응하여 발생하는 센서 플랫폼의 응답 메시지에 대한 RSSI, LQI 값이 된다. 이러한 두 링크에 대한 품질은 센서 플랫폼이 BS로 전송하는 응답 메시지를 기반으로 응용 레벨에서 확인할 수 있도록 하였다. 그림 3.1은 구현 레벨에서 무선 링크의 비대칭성을 고려하기 위해 링크에 대한 정의를 나타낸 것이다. 또한 그림 3.2는 이러한 구현 시나리오에 대한 전체 시스템을 나타낸 것이다.

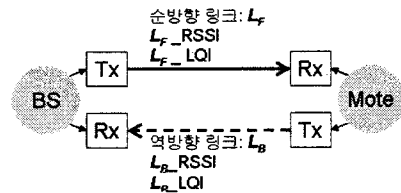


그림 3.1 링크에 대한 정의

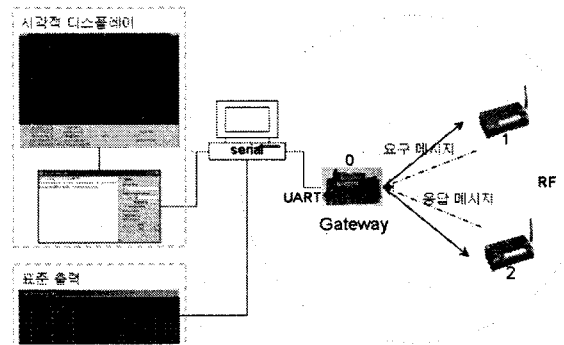


그림 3.2 전체 시스템

그림 3.2에서, 초기에 BS의 요구 메시지는 주기적으로 브로드캐스팅되고, 센서 플랫폼은 그 수신된 요구 메시지에 대한 RSSI, LQI 값을 응답 메시지에 캡슐화하여 브로드캐스팅한다. 응답 메시지를 받은 BS는 수신된 응답 메시지에 대한 RSSI, LQI 값을 구하고, 캡슐화된 RSSI, LQI 값을 디-캡슐화함으로써 순방향, 역방향에 대한 링크 품질을 표준 출력이나 시각적 디스플레이로 확인할 수 있다.

3.2 메시지 구조

MICAZ간의 통신에 사용되는 메시지 포맷은 응용 구현 환경으로 인해 무선 센서 네트워크에서 사실상의 표준으로 자리매김한 TinyOS-1.1 [9]의 액티브 메시지 포맷 (TOS_Msg)을 사용하였다. 그림 3.3은 구현에 사용된 요구 메시지와 응답 메시지의 구조를 나타낸 그림이다. 주기적으로 전송되는 요구 메시지와 그에 따른 응답 메시지는 이 메시지의 'data' 필드를 사용한다.

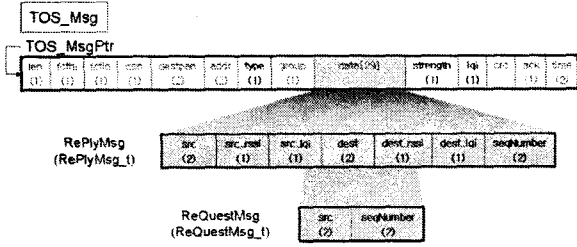


그림 3.3 요구 및 응답 메시지 구조

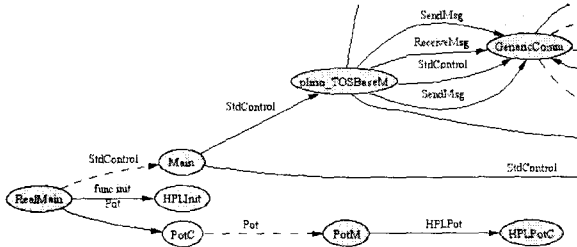
또한 요구 메시지와 응답 메시지를 구별하기 위한 타입은 각각 'AM_REQMSG=7', 'AM_REPMMSG=8'로 정의하였다.

3.3 nesC로 구현된 통신 모듈

MICAZ에 적재된 nesC[10]로 구현된 통신 모듈은 (가) BS의 주기적인 요구 메시지 전송, (나) 수신된 요구 메시지에 따른 응답 메시지 전송, (다) MICAZ의 RSSI, LQI 값을 얻기 위한 기능, (라) BS로 취합된 순방향과 역방향 링크 품질을 UART로 전송하는 기능을 구현되었다. 크게 BS에 적재될 게이트웨이 역할을 하는 phno_TOSBaseM과 센서 플랫폼에 적재될 phno_TestM으로 나눌 수 있다.

phno_TOSBaseM:

이 모듈은 게이트웨이 기능을 하는 BS에 적재되며, (가), (다), (라) 기능을 포함하고 있다. 그림 3.4는 이러한 기능에 대한 구현을 나타낸 것이다. phno_TOSBaseM은 Timer 컴포넌트를 추가하여 요구 메시지를 2초마다 주기적으로 전송하고, 기존의 TOSBase와 달리 패킷에 대한 송수신은 GenericComm 컴포넌트를 사용하여 처리하였다.



```

task void phno_RequestMsgSendTask() {
    seq = seq + 2; //per 2sec
    phno_greQmsg->Lf_id = TOS_LOCAL_ADDRESS;
    phno_greQmsg->seqNumber = seq;
    call RequestMsg.send(TOS_BCAST_ADDR,
        sizeof(ReQuestMsg_t), phno_gTxMsgPtr);
}
event TOS_MsgPtr RePlyMsg.receive(TOS_MsgPtr Msg) {
    if (_replymsg->Lf_id == TOS_LOCAL_ADDRESS)
        post phno_ReplyMsgRcvdTask();
}
    
```

그림 3.4 phno_TOSBaseM

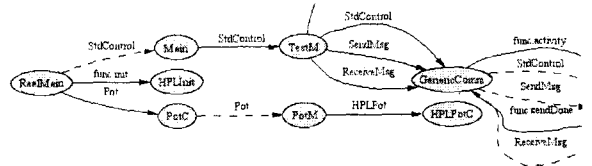
```

task void phno_ReplyMsgRcvdTask() {
    _replymsg=(RePlyMsg_t*)phno_gRxMsgPtr->data;
    phno_grePmsg->Lf_id = _replymsg->Lf_id;
    phno_grePmsg->Lf_rssi = _replymsg->Lf_rssi;
    phno_grePmsg->Lf_lqi = _replymsg->Lf_lqi;
    phno_grePmsg->Lb_id = _replymsg->Lb_id;
    phno_grePmsg->Lb_rssi = phno_gRxMsgPtr->strength;
    phno_grePmsg->Lb_lqi = phno_gRxMsgPtr->lqi;
    phno_grePmsg->seqNumber
        =_replymsg->seqNumber;
    call UARTSend.send(TOS_UART_ADDR,
        sizeof(RePlyMsg_t), phno_gRxMsgPtr);
}
    
```

그림 3.4 phno_TOSBaseM

phno_TestM:

이 모듈은 센서 플랫폼에 적재되며, (나), (다) 기능을 포함하고 있다. 그림 3.5는 이러한 기능에 대한 구현을 나타낸 것이다. phno_TestM은 위에서 설명한 phno_TOSBaseM과의 패킷 송수신을 위해 동일하게 GenericComm 컴포넌트를 사용한다. 또한 이 모듈은 요구 메시지의 수신 이벤트를 받아 순방향 링크에 대한 RSSI, LQI 값을 구한 후, 응답 메시지를 전송하는 함수에서 이러한 값들을 응답 메시지에 캡슐화하여 브로드캐스팅한다.



```

task void RequestMsgRcvdTask() {
    _replymsg = (RePlyMsg_t*)phno_gMsgPtr->data;
    phno_grePmsg->Lf_id = phno_grePmsg->Lf_id;
    phno_grePmsg->Lf_rssi = phno_gMsgPtr->strength;
    phno_grePmsg->Lf_lqi = phno_gMsgPtr->lqi;
    call RePlyMsg.send(TOS_BCAST_ADDR,
        sizeof(RePlyMsg_t), phno_gMsgPtr);
}

//request 메시지를 받으면 RequestMsgRcvdTask() 호출
event TOS_MsgPtr RequestMsg.receive(TOS_MsgPtr Msg){
    _requestmsg = (ReQuestMsg_t*)Msg->data;
    if(_requestmsg->Lf_id != TOS_LOCAL_ADDRESS) {
        phno_grePmsg->Lf_id = _requestmsg->Lf_id;
        phno_grePmsg->seqNumber
            =_requestmsg->seqNumber;
        post RequestMsgRcvdTask();
    }
}
    
```

그림 3.5 phno_TestM

3.4 RSSI, LQI 모니터링 응용

BS는 센서 플랫폼에서 전송된 응답 메시지에 캡슐화된 순방향 링크의 RSSI, LQI 값과 그 응답 메시지에 대한 역방향 RSSI, LQI 값을 하나의 데이터 패킷에 담아 UART로 전송한다. 그림 3.6은 순방향 RSSI, LQI 그리고 역방향 RSSI, LQI 값과 정의한 메시지 구조를 잘 보여준

다. 또한 시각적인 디스플레이를 위해 자바 기반의 링크 품질을 모니터링하는 응용을 구현하여 수신된 값들에 대한 결과를 확인하였다.

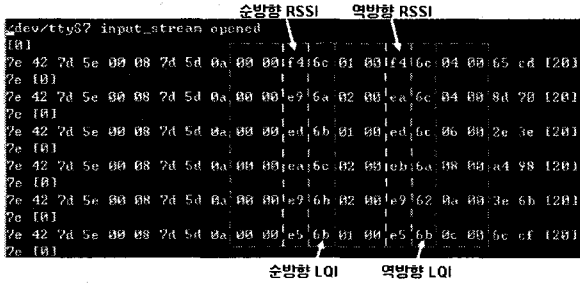


그림 3.6 수신된 데이터 패킷

4. 실험

본 논문에서 구현된 응용을 이용하여 얻은 RSSI, LQI 그리고 무선 링크 특성을 확인하기 위해 실내와 실외에 간단한 테스트 환경을 구축하여 실험을 수행하였다. 실험 환경은 센서 플랫폼으로 MICAZ을 사용하였으며, BS 노드에 0번 그리고 각 센서 노드에 1, 2번을 할당하였다. 자바 기반의 링크 품질을 모니터링하는 응용을 이용하여 변화된 값을 확인하였다.

그림 4.1은 실외에서 1번 센서 노드와 BS간의 링크에 대한 RSSI, LQI를 나타낸 것이다. 또한 그림 4.2는 연구실에서 2번 센서 노드와 BS간의 링크에 대한 RSSI, LQI를 나타낸 것이다. 두 결과를 통해 무선 링크의 비대칭성을 확인하였으며, 실외보다 실내에서 상대적으로 링크 품질에 대한 변화가 있음을 확인하였다. 이러한 변화는 멀티-경로, 장애물, 간섭, 페이딩 때문이다.

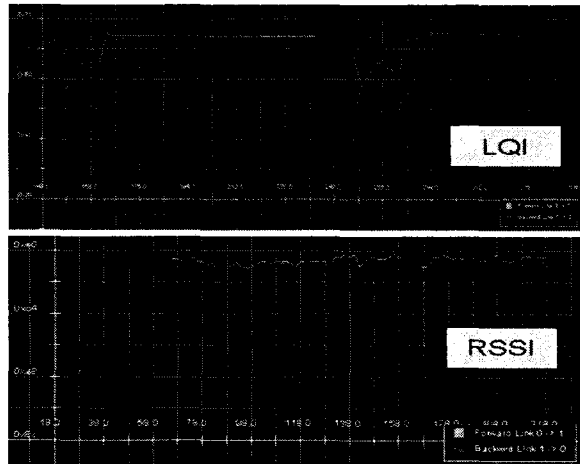


그림 4.1 실외에서의 LQI, RSSI

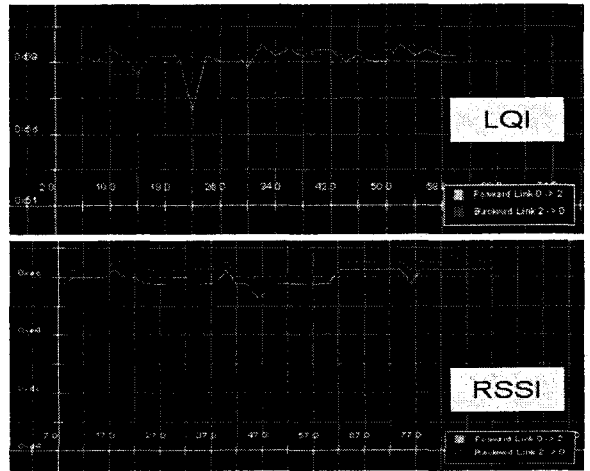


그림 4.2 실내에서의 LQI, RSSI

그림 4.3은 실내에서 1, 2번 센서 노드와 BS간의 링크에 대한 RSSI, LQI를 나타낸 것이다. 그림 4.1과 동일하게 링크 품질에 대한 변화가 있음을 확인할 수 있다. 물론 결과를 통해 무선 링크의 비대칭성을 확인하였으며, LQI에 비해 RSSI가 링크 품질을 나타내는 변화의 폭이 적음을 확인할 수 있었다. 이렇게 변화 폭의 차이를 보이는 이유는 링크 품질을 구하는 방식의 차이 때문이다.

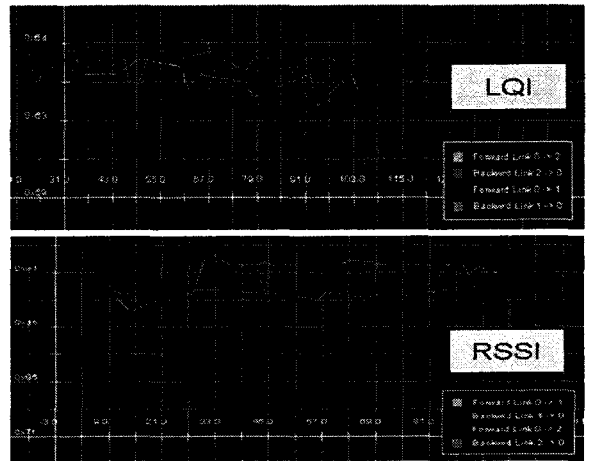


그림 4.3 실내에서의 LQI, RSSI

5. 결론 및 향후과제

무선 링크는 예측할 수 없으며, 환경에 매우 민감하게 반응한다. 이렇게 신뢰할 수 없는 링크를 통해 데이터를 전송하는 것은 그렇게 생산적이지 못하다. 특히, 무선 멀티 홉 환경에서 다음 홉을 결정하는 라우팅 계층은 에너지 효율적이고 신뢰적인 링크를 선택하는 것이 매우 어렵게 된다. 이러한 링크를 선택하기 위해 라우팅 계층은 하위 계층의 하드웨어 혹은 소프트웨어적으로 처리된 정보를 이용하여 무선 링크에 대한 품질을 추정하는 기법이 반드시 필요하다. 무선 링크의 품질을 추정하는 한

가지 방법은 링크 품질을 명시적으로 나타내는 지표를 사용하는 것이다.

본 논문에서는 그 일차적인 단계로써 단일 홉 무선 센서 네트워크에서 MICAZ에 구현된 RSSI, LQI 값을 이용하여 노드간 무선 링크 품질을 모니터링하는 응용을 구현하였다. 구현 시나리오는 무선 링크의 비대칭성을 고려하기 위해, 순방향 링크 품질의 경우 센서 플랫폼이 BS로부터 수신한 요구 메시지의 RSSI, LQI 값을 응답 메시지에 캡슐화하여 BS로 전송하도록 하였고, 역방향 링크 품질의 경우 BS가 센서 플랫폼으로부터 수신한 응답 메시지의 RSSI, LQI 값을 기반으로 하였다. BS로 취합된 이러한 두 링크 품질을 PC상에 시각적으로 표시하기 위해 자바 기반의 링크 품질을 모니터링하는 응용을 구현하였다. 간단한 실험 결과를 통해 무선 링크의 비대칭성을 명확히 확인할 수 있었고, 실외보다 실내에서 링크 품질에 대한 변화가 있음을 알 수 있었다.

향후 과제로는 좀 더 다양한 테스트 환경을 구축하여 링크 품질을 나타내는 지표에 대한 연구를 지속할 것이다. 또한 링크를 추정하는 다른 기법에 대한 연구를 병행하여 라우팅 계층과의 상호 연동을 심도있게 검토할 것이다.

참고 문헌

- [1] A. Woo, T. Tong and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," In Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys), 2003.
- [2] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," In Proceedings of the First ACM Conference on Embedded Network Sensor Systems, 2003.
- [3] Crossbow Technology Inc.
<http://www.crossbow.com>
- [4] J. Polastre, R. Szewczyk and D. E. Culler, "Telos: enabling ultra-lowpower wireless research," In IPSN, 2005.
- [5] ChipCon Inc. <http://www.chipcom.com>
- [6] TinyOS. MultiHopLQI. <http://www.tinyos.net/>
[/tinyos-1.x/tos/lib/MultiHopLQI](http://www.tinyos.net/tinyos-1.x/tos/lib/MultiHopLQI), 2004.
- [7] G. Tolle and D. Culler, "Design of an application cooperative management system for wireless sensor networks," In Proceedings of Second European Workshop on Wireless Sensor Networks (EWSN), 2005.
- [8] IEEE Std 802.15.4™-2003
- [9] <http://www.tinyos.net>
- [10] D. Gay, P. Levis, R. Behren, M. Welsh, E. Brewer and D. Culler. "The nesC Language: A Holistic Approach to Networked Embedded Systems." In ACM SIGPLAN Conference on Programming Language Design and Implementation, 2003.