

무선 센서네트워크에서의 신뢰성 있는 비동기적 이미지 전송 프로토콜 설계 및 구현

선주호[○], 권영완, 박총명, 이현길, 정인범
강원대학교 컴퓨터정보통신공학과

{ jhseon[○], ywkwon, cmpark, hglee, ibjung }@snslab.kangwon.ac.kr

Design and Implementation of Reliable Asynchronous Image Transfer Protocol In Wireless Sensor Network

Juho Seon[○], Youngwan Kwon, Chongmyung Park, Heonguil Lee, Inbum Jung
Dept. of Computer, Information and Telecommunication Engineering,
Kangwon National University

요 약

무선 센서네트워크는 다양한 환경에서 감시와 정보수집 역할을 수행하는 시스템이다. 최근에는 하드웨어의 발달로 무선 센서네트워크를 이용하여 멀티미디어 데이터를 수집하기 위한 시도가 활발히 이루어지고 있다. 본 논문에서는 센서네트워크에서의 신뢰성 있는 비동기적 이미지 전송 프로토콜인 RAIT를 제안한다. RAIT는 작은 크기의 메모리를 갖는 센서노드의 특징을 고려하여 전송 중 발생하는 Overflow의 방지를 위하여 Token-Bucket 방식으로의 네트워크 트래픽 관리를 통해 이미지 전송의 신뢰성을 높인다. 또한 시뮬레이션 결과를 통해 RAIT의 이미지 전송 신뢰성을 확인하였다.

1. 서 론

무선 센서네트워크는 인간 중심의 유비쿼터스 패러다임의 확대로 전 세계적으로 활발하게 연구되어지는 기술이다. 최근에는 CMOS 카메라 및 센서노드의 하드웨어 성능 향상으로 무선 센서네트워크 망을 이용하여 멀티미디어 데이터를 전송하기 위한 WMSN(Wireless Multimedia Sensor Network) 분야가 주목받고 있다[1]. WMSN을 활용한 시스템은 사람이 접근하기 어려운 지역의 멀티미디어 데이터 수집이 가능하여 다양한 응용분야에 활용될 수 있다.

WMSN을 이용하기 위해서는 센서노드의 멀티 홉 네트워크를 활용하여 비디오, 오디오 등의 멀티미디어 데이터를 원활하게 전송할 수 있는 메커니즘이 보장되어야 한다. 하지만 좁은 네트워크 대역폭을 이용하여 수 킬로 바이트에 이르는 멀티미디어 데이터를 신뢰성 있게 전송하기 위해서는 노드장애, 네트워크 어러와 같은 상황에

대한 내성이 필요하다[2]. 그리고 센서노드가 갖고 있는 적은 메모리는 대용량의 멀티미디어 데이터를 전송하는데 한계가 있다. 멀티 홉 통신의 특성으로 소수의 센서노드에게 데이터가 집중되는 성향이 나타나는데, 이로 인해 작은 크기의 수신 Queue를 갖는 센서노드는 여러 루트로부터 수신하는 많은 양의 멀티미디어 데이터를 저장하는데 문제가 발생하기도 한다. 일반적인 수신 Queue는 Overflow를 방지하기 위해 수신한 패킷을 Drop하는 방법을 사용한다. 이러한 동작으로 멀티 홉 통신을 이용하는 무선 센서네트워크에서 홉이 늘어날수록 에러율이 점점 높아지게 된다. 또한 작은 크기의 배터리를 갖고, 외부적인 충격에 파손될 가능성이 높기 때문에 라우터는 이러한 네트워크 장애에 대한 내성을 지녀야 한다.

본 논문에서는 무선 센서네트워크에서 신뢰성 있는 이미지를 전송하기 위한 프로토콜인 RAIT(Reliable Asynchronous Image Transfer protocol)를 제안한다. RAIT는 Token-Bucket 방식을 이용한 메커니즘을 이용해 이미지 데이터의 흐름제어를 담당한다. 이러한 구조

** 본 연구는 산업자원부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임 **

는 부모노드의 Queue Overflow를 막아주어 작은 메모리 크기로 인해 발생하게 되는 이미지 패킷의 손실을 없애 이미지 전송의 신뢰성을 유지하여 준다. 또한 Table-Driven 방식의 멀티 홉 라우터 사용으로 노드의 손실에 대한 내성을 지닌다.

본 논문의 구성은 다음과 같다. 2장에서는 흐름제어 기법 중 하나인 Token-Bucket 방식에 대해 알아본다. 3장에서는 RAIT를 이루고 있는 각 Layer의 구성에 대해, 4장에서는 RAIT의 동작방식을 알아본다. 5장에서는 시뮬레이터를 이용하여 RAIT의 성능을 평가하며, 6장의 결론 및 향후연구로써 끝을 맺는다.

2. 관련연구

무선 네트워크에서 데이터 전송의 신뢰성을 보장하기 위한 많은 연구가 진행 되었지만, 이를 그대로 무선 센서네트워크에 적용시키는 것은 적합하지 않다[3]. 무선 통신에서 데이터 신뢰성을 유지하기 위한 기법으로 알려져 있는 PSFQ[4]와 RMST[5]는 NAK(Negative Acknowledgement)와 ARQ(Automatic Repeat Request)를 사용하여 재전송으로 인해 일어날 수 있는 오버헤드를 최소로 하기 위한 기법이다. 기존 무선 네트워크에서는 링크 에러가 발생하면 ARQ방식으로 재전송하여도 오버헤드가 크게 발생하지 않았지만, 에너지의 제약이 큰 무선 센서네트워크에서의 패킷 재전송은 큰 에너지 소모를 발생시키므로 적합하지 않다. 게다가 잦은 재전송은 작은 크기의 수신 큐를 갖는 메모리 크기의 한계를 가져와 정상적인 데이터 전송이 어렵게 만든다.

FEC(Forward Error Correction)는 에러정정코드를 사용하여 데이터 에러를 복구한다. 이미지 전송을 위한 FEC 기반의 UEP(Unequal Error Protection)는 무선통신에서의 에러를 줄이고 효과적인 전송을 가능하게 한다 [5-7]. 패킷의 손실이 감지되면 FEC 알고리즘은 중요 데이터를 복구하기 위해 FEC code를 적용한다. 하지만 FEC를 사용하면 패킷 전체가 손실되는 경우에 대한 복구가 어려우며, 멀티 홉을 이용한 무선 네트워크에서의 에너지 소비와 수신 이미지 품질에 관한 검증이 충분하지 않은 문제가 있다.

이미지 데이터의 높은 신뢰성을 유지하기 위한 다른 한 가지 방법은 Multi-path 방식의 전송이다[2]. Multi-path 방식은 두 개의 센서노드에 동일한 데이터를 전송함으로써 통신 에러나 노드 장애로 인해 발생하는 손실로 인한 재전송을 최소화한다. 이러한 분할 스키마는 링크 채널 저하 및 노드에러 발생에서 일정 수준 이상의 이미지 품질을 유지해준다.

3. RAIT 구성

Cross Layer 설계는 기존 계층 기반의 설계를 복잡하게 만들지만, 제한적인 리소스를 효율적으로 사용하는 무선 네트워크 환경에서는 적절한 기법으로 고려된다 [10]. RAIT는 Multihop Layer와 Queue Layer를 RAIT에서 통제하도록 하는 Cross Layer 구조로 설계되었다. 그리고 이미지 전송에서 중요하게 고려되는 네트워크 트래픽으로 인한 이미지 손실은Token-Bucket 기법을 사용함으로써 해결한다.

3.1 RAIT Layer

이미지 프로토콜 구현에서 가장 중요하게 고려되어야 하는 사항은 이미지 전송의 신뢰성 유지이다. 일반적인 센서노드는 낮은 사양의 하드웨어를 사용하기 때문에 작은 크기의 수신 큐를 갖게 된다. 그렇기 때문에 대용량의 멀티미디어 데이터를 전송하게 되면, 수신 측에서는 Queue Overflow가 발생할 가능성이 높다. 이러한 문제의 해결을 위해 RAIT에서는 Token-Bucket 기법을 이용한다. Token-Bucket에서의 Token은 하나의 바이트 혹은 패킷을 의미하며, 데이터 전송의 흐름제어를 위한 최소의 단위이다. RAIT Layer에서는 하나의 패킷을 Token으로 정의하고 네트워크 트래픽을 고려하여, 통신이 가능한 경우에만 이미지를 전송하는 메커니즘으로 동작한다. 네트워크 Congestion을 판단하기 위한 기준은 부모노드로부터 할당 받은 Token 수이며, 자식노드는 부모노드의 Token이 남아있지 않으면 데이터 전송을 잠시 중지한다. 이러한 동작을 위해 RAIT Layer는 Multihop Layer,

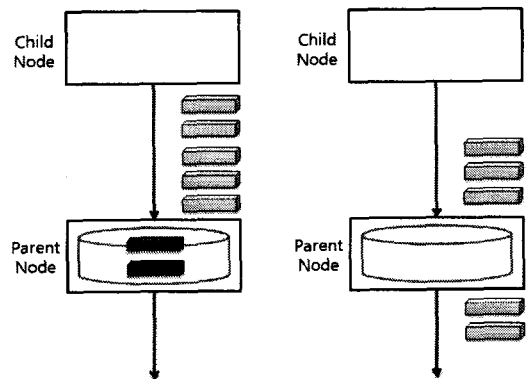


그림 3. Token-Bucket 동작방식.

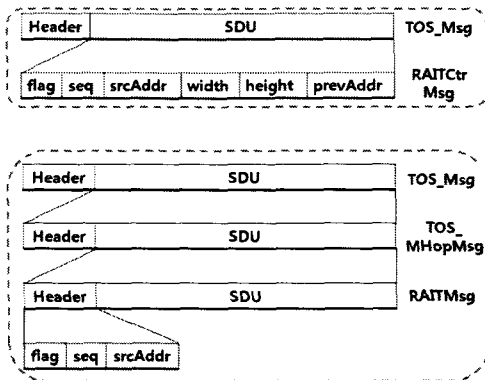


그림 4. RAIT에서의 패킷 매핑.

Queue Layer를 제어 할 수 있는 Cross Layer로 설계되었다. RAIT Layer는 이미지 전송을 위해 필요한 Control Message를 부모 노드에 전송하고, 부모노드의 Token을 얻어온다. 자식노드는 현재의 전송가능 패킷 수를 그림 1과 같이 Token의 수로 제한하며, 한 패킷을 전송할 때마다 할당받은 Token 수를 줄여 나간다. 할당받은 Token이 남아있지 않으면 전송을 일시 중지하고 부모노드에게 Token을 요청한다. 만약 부모노드에 이용 가능한 Token이 없는데 다른 노드로부터 패킷을 수신하면 Queue Layer는 Queue Overflow를 방지하기 위해 해당 데이터를 Drop한다. 하지만 자식 노드가 부모노드의 Token을 고려한다면 Queue Overflow로 인해 발생하는 패킷 손실은 없을 것이다.

RAIT에서는 그림 2와 같은 형식의 컨트롤 메시지와 이미지 전송 메시지를 사용한다. 메시지 타입은 flag를 이용하여 구분하며, 이미지의 순서를 나타내는 seq, 이미지 센서의 노드 번호를 나타내는 srcAddr는 공통적으로 사용한다.

3.2 Multihop Layer

RAIT에서는 Table-Driven 방식의 Multihop Layer를 사용한다. 라우팅이 시작되면 센서노드는 라우팅 메시지를 브로드캐스트 방식으로 주변 노드에게 전송하고, 타 노드로부터 라우팅 메시지를 수신한 노드는 이를 기반으로 라우팅 테이블을 갱신한다. Multihop Layer의 Estimator는 주기적으로 라우팅 테이블로부터 가장 적당한 노드 하나를 선택하여 부모노드로 선택한다. 부모노드를 선택하기 위한 Metric은 Hop-Count와 Link-Quality로써, 라우팅 테이블에 센서노드 번호와 함께 저장한다. 이러한 구조의 Multihop Layer는 부모노드를 선택하기 적절한 라우팅 테이블을 갖지 못한 경우 이를 구성하는데 많은

시간이 걸리는 편이지만, 라우팅 테이블 구성이 완료되면 안정적인 패킷 전송이 가능한 장점이 있다.

3.3 Queue Layer

노드와 링크의 조합인 무선 센서네트워크에서는 데이터의 손실을 막기 위해 트래픽 제어 기법을 필요로 한다. 트래픽 제어는 송신 노드에서 목적 노드로 데이터가 전송될 때 목적 노드 쪽에서 데이터를 너무 빠르게 혹은 너무 느리게 처리하지 않도록 전송량을 조절한다. 혼잡 제어는 2개 이상의 노드로부터 데이터를 전송할 때 데이터 충돌을 피하거나 해결하기 위한 방법이다. RAIT는 트래픽 조절과 혼잡 제어를 위해 Drop-Tail(FIFO) 기반의 Queueing 기법을 사용한다. Drop-Tail 기반의 Queueing 기법은 수신 큐가 찼을 때 수신되는 패킷을 Drop하는 방식을 사용하는데, 이미지 데이터는 손실이 생기면 정상적인 디코딩이 어렵기 때문에, Queue Layer에서의 Drop으로 발생하는 패킷 손실이 발생하지 않게 한다. RAIT는 부모노드 Queue Layer의 Token 크기를 고려하여 데이터를 전송하기 때문에 다수의 데이터 수신으로 인해 패킷이 Drop되는 문제가 발생하지 않는다.

4. 동작방식

동작방식은 다음과 같이 이미지 데이터 송신부, 이미지 데이터 수신부, 포워딩 이미지 처리부로 분류된다.

4.1 이미지 데이터 송신

RAIT는 이미지 데이터를 송신하기 위해 그림 3과 같이 Token을 얻어오며, 그림 4와 같이 이미지를 송신하기

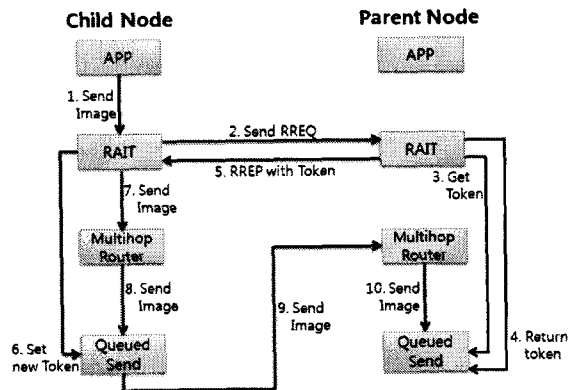


그림 5. RAIT에서의 Token-Bucket 알고리즘.

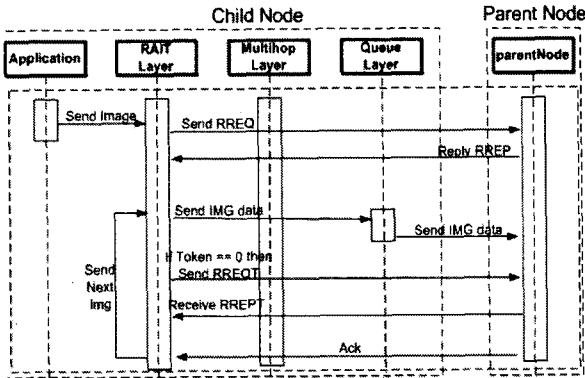


그림 4. 이미지 전송에 대한 시퀀스 다이어그램.

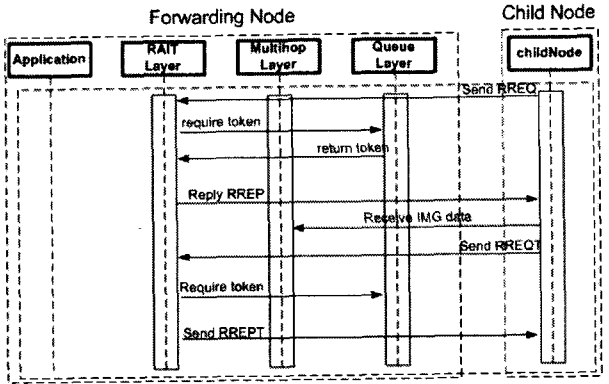


그림 5. 이미지 수신에 대한 시퀀스 다이어그램.

위한 시퀀스 다이어그램을 갖는다. 먼저 Application은 이미지 샘플링 데이터를 배열 형태로 RAIT로 넘겨준다. RAIT에서는 부모노드에 이미지 전송 요청을 위한 REQ_INIT 메시지를 전송하고 부모노드의 컨트롤 메시지를 기다린다. 부모노드는 이미지 전송 혹은 타 노드로부터 포워딩 중이 아니라면 Token을 포함한 REP_INIT 메시지를 자식노드에게 전송하여 이미지 전송을 허용한다. REP_INIT 메시지를 자식노드가 정상적으로 수신하였으면 이미지 노드는 Multihop Layer에 현재 상태를 알려주고 이미지 전송 중 다른 부모노드를 선택하지 못하도록 라우팅 경로를 고정한다. 이는 이미지 데이터가 동일한 라우팅 경로로 전송되어야 이미지 전송 신뢰성을 보장받을 수 있기 때문이며, 해당 이미지의 전송이 완료되면 라우팅 경로가 변경될 수 있도록 설정을 되돌린다. 부모노드가 할당된 Token이 0보다 크면 이미지 전송 가능 상태로 판단하고 데이터 전송을 시작한다.

이벤트 기반 운영체제의 특징을 갖는 TinyOS에서는 환경 수집 데이터 전송에 주기적인 타이머 이벤트를 사용하는 것이 일반적이다. 하지만 이미지 전송 시 타이머 이벤트 사용하는 방법은 적절치 못하다. 이미지 전송은 대용량의 데이터를 가능한 빠른 속도로 보내야 하는 것이 관건이다. 하지만 주기적인 타이머 이벤트를 사용하면 하나의 패킷을 전송하는데 어느 정도의 시간이 걸릴지 예측하기 매우 어렵다. 무선 환경은 변화가 매우 심하며 매체를 공유하여 노드 간에 경쟁이 심하고 경쟁하는 노드의 수가 증가할수록 패킷을 전송하는 시간도 증가한다. 너무 적은 타이머 주기를 사용하면 Split-phase 문제로 인해 전송명령이 실패한다. 왜냐하면 무선 센서 네트워크에서 사용되는 TinyOS가 특정 명령을 Split-phase 방식으로 실행하고, 특정 이벤트를 실행하

는 동안 다른 종류의 이벤트가 실행되면 두 번째로 실행하는 이벤트의 동작은 강제 종료시키기 때문이다. Split-phase 현상을 방지하고 전송속도를 향상시키기 위해서는 전송완료 이벤트에서 전송 함수를 호출하여 연속적으로 데이터를 전송하는 방식이 효율적이다[9].

이미지 데이터의 전송 중 Token이 부족하면 이미지 데이터의 전송을 정지하고 부모노드에게 Token을 요청하기 위해 REQ_TOKEN 메시지를 전송한다. 부모노드는 REP_TOKEN 메시지를 통해 Token을 돌려주고, 자식노드가 0보다 큰 Token을 얻었으면 이미지 데이터 전송을 재개하고 그렇지 않으면 임의의 시간을 주기로 REQ_TOKEN 메시지 전송을 재시도 한다. 이미지 전송이 완료되면 REQ_END 메시지를 부모노드에게 전달하여 이미지 전송을 종료한다.

4.2 이미지 데이터 수신

RAIT를 이용하여 이미지를 수신하기 위한 시퀀스 다이어그램은 그림 5와 같다. 특정 노드가 REQ_INIT 메시지를 수신하면 자신의 현재 상태를 확인하고, 이미지 전송이거나 혹은 타 노드로부터 포워딩 중이 아니면 자식노드에게 이미지 수신이 가능함을 알린다. 이를 위한 REP_INIT 메시지에는 Queue Layer에서 얻어온 Token을 포함한다. 그 다음 Multihop Layer에 이미지 포워딩 상태를 알리며, Multihop Layer는 현재 설정된 자식노드에서 송신한 데이터만 수신한다. 만약 이미지 수신 중 자식노드로부터 Token 요청을 위한 REQ_TOKEN 메시지가 도착하면 Queue Layer의 Token 크기를 얻어와 자식노드에게 반환한다. 자식노드에게서 REQ_END 메시지가 도착하면 이미지 전송이 끝났음을 의미하기 때문에 Multihop Layer를 재설정하고 이미지 수신을 종료한다.

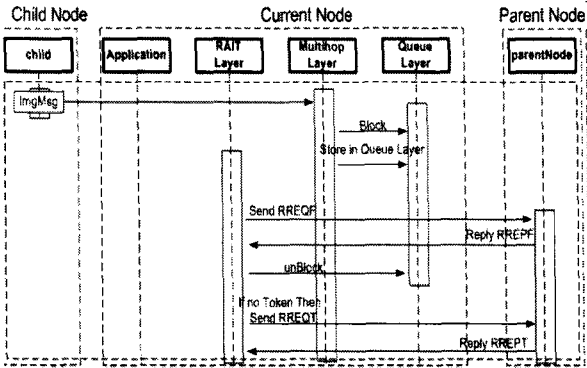


그림 8. 이미지 포워딩에 대한 시퀀스 다이어그램.

4.3 이미지 데이터 포워딩

RAIT를 이용한 이미지 포워딩을 위한 시퀀스 다이어그램은 그림 6과 같다. 자식노드에서 이미지 데이터가 도착하면 현재 노드는 이미지를 부모노드에게 포워딩하기 위한 특별한 동작을 수행해야 한다.

먼저 Multihop Layer에서는 수신한 이미지 데이터의 시퀀스 헤더를 확인하여 첫 번째 이미지이면 Queue Layer를 Blocking하여 전송 지연(Transmission delay)시킨다. Queue Layer는 수신 및 동시에 전송을 위한 수신 큐 역할을 한다. Queue Layer에 대기하고 있는 패킷은 Downstream될 이웃 노드가 enable 될 때까지 대기한다. 자식노드는 현재노드 Queue의 크기를 고려하여 데이터를 전송하기 때문에 수신한 이미지 데이터가 Drop으로 인한 손실은 발생하지 않는다. 그 다음은 RAIT에 포워딩 초기화 명령을 전달하고 부모노드에게 REQ_FORWARD 메시지를 전송하여 포워딩을 요청한다. 부모노드가 REP_FORWARD 메시지를 전달하여 이미지 포워딩이 가능함을 알리고, Token을 할당하면 RAIT는 Queue를 Enable하여 부모노드에게 이미지 데이터 포워딩을 시작한다. 이때 Queue에서는 REP_FORWARD 메시지로부터 수신한 Token을 전달하여 실제 데이터 전송 시 Token을 체크하도록 Queue의 전송으로 인해 부모노드로 전송한 데이터가 Drop되지 않도록 방지한다. Token이 부족하면 Queue를 Blocking하여 포워딩을 중지하고 RAIT를 통해 부모노드의 Token을 얻어오도록 한다. 이미지에서의 토큰 관리를 Queue에서 하는 이유는 Queue의 sendDone에서 Token을 관리하면 이벤트 기반 언어의 특징으로 일반 명령보다 이벤트 명령이 먼저 수행되기 때문에 이로 인해 발생할 수 있는 오동작을 방지하기 위해서이다.

자식노드의 이미지 전송이 끝나 RAIT에서 REQ_END

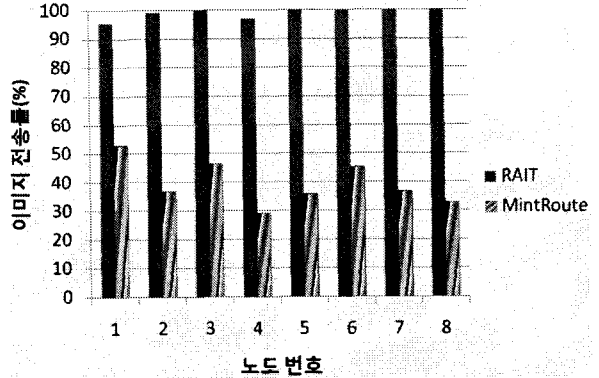


그림 9. 시뮬레이션에서의 이미지 전송률.

메시지를 수신하면 Multihop Layer와 Queue에 이미지 전송 종료 상태를 알린다. Queue에서는 이미지 데이터의 시퀀스를 확인하고 데이터의 포워딩이 완료되었으면 RAIT에 REQ_END 메시지를 전달하고, 이를 확인한 RAIT는 부모노드에 REQ_END 메시지를 전송함으로써 이미지 포워딩이 끝났음을 확인한다.

5. 성능평가

본 논문에서 제안하는 RAIT의 성능을 알아보기 위해 TinyOS에서 제공하는 Table-Driven 방식의 멀티 홉 컴포넌트인 MintRoute 컴포넌트와의 이미지 전송률 차이를 알아보았다. 이를 위해 시뮬레이터인 TOSSIM과 TinyViz, 스크립트 언어인 Tython을 이용하였다. Tython은 TOSSIM에서 사용되는 Java와 Python으로 구성된 스크립트 언어이며 TOSSIM과 연동하여 센서노드를 제어하거나, 센서 어플리케이션의 데이터 값을 얻어오는 등 다양한 기능을 제공한다.

실험은 100개의 센서노드를 Grid하게 배치하고, 임의로 지정한 8개의 이미지 노드에서 싱크까지 전송되는 이미지에 대한 전송률을 알아보았다. 각 이미지는 36개의 패킷으로 구성되며, 이미지의 완전한 데이터가 수신 완료되어야 이미지 전송을 성공하였다고 가정한다.

RAIT와 멀티 홉 컴포넌트를 이용한 이미지 전송 결과는 그림 7과 같다. RAIT를 이용한 이미지는 100% 가까운 전송률을 가졌지만, MintRoute 컴포넌트를 이용한 전송률은 약 50% 밖에 안 되는 것을 볼 수 있다. 이러한 결과가 나온 이유는 다음과 같다. 첫 번째 원인은 여러 이미지 노드가 전송한 데이터가 싱크노드에 도착하는 도중에 유실되기 때문에 일어난다. 그 이유는 센서노드가 부모노드의 데이터 수용량을 고려하지 않기 때문인데, 본 실험에서는 여러 개의 노드에서 선택한 하나의

부모노드에게 동시에 데이터를 전송하여 발생하였다. 특히 싱크노드로부터 가까운 흡에 존재하는 센서노드에게서 이러한 현상이 발생하였다. 이러한 문제로 인한 네트워크 트래픽은 Queue Overflow로 인한 데이터 손실을 가중시킨다. 두 번째 이유는 이미지 전송 중 라우팅 경로가 임의로 바뀌기 때문이다. 이미지 데이터는 순차적으로 싱크 노드에 전송되어야 하는데 루트가 변하면 최종 목적지인 싱크노드에 도착하지 못하고 패킷 손실이 발생할 가능성이 높아진다.

6. 결론 및 향후연구

최근 무선 센서네트워크에서 멀티미디어 데이터를 이용하기 위한 시도가 이루어지고 있다. RAIT는 무선 센서 네트워크 환경에서 이미지를 신뢰성 있게 전송하기 위한 프로토콜이다. Token-Bucket 방식으로 동작하는 RAIT를 통해 네트워크 트래픽으로 발생하는 이미지 손실을 최소화하였다. 이로써 무선 센서네트워크의 멀티 흡 통신을 통한 전송시 이미지 신뢰성을 보장받을 수 있었다.

향후 시뮬레이션이 아닌 실제 이미지 센서를 사용하여 영상 전송을 이용한 모니터링 시스템을 개발할 계획이다. 그리고 Table-Driven 방식의 멀티 흡 라우팅 대신 AODV 프로토콜을 적용하여 모바일 싱크에서의 적용 방안에 대한 연구를 진행할 예정이다.

참고문헌

- [1] I. F. Akyildiz, T. Melodia, K. R. Chowdhury, "A survey on wireless multimedia sensor networks", *Computer Networks*, 51:921-960, 2007.
- [2] H. Wu, A. A. Abouzeid, "Error resilient image transport in wireless sensor networks", *Computer Networks*, 50:2873-2887, 2007.
- [3] C.Y. Wan, A.T. Cambell, L. Krishnamurthy, "PSFQ: a reliable transport protocol for wireless sensor networks", *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, ACM Press, 2002, pp. 1-11.
- [4] F. Stann, J. Heidemann, "RMST: reliable data transport in sensor networks", *1st IEEE International Workshop on Sensor Net Protocols and Applications(SNPA)*, Anchorage, Alaska, USA, 2003.
- [5] P.G. Sherwood, K. Zenger, "Error protection for progressive image transmission over memoryless and fading channels", *IEEE Transactions on Communications* 46(12)(1998) 1555-1559.
- [6] A. Mohr, E. Riskin, R. Ladner, "Unequal loss protection: graceful degradation of image quality

over packet erasure channel through forward error correction", *IEEE Journal on Selected Areas in Communications* 18(6)(2000) 819-828.

- [7] K. W. Lee, R. Puri, T. Kim, K. Ramchandran, V. Bhargh-avan, "An integrated source and congestion control frame-work video streaming in the Internet", *Proceedings on Infocom*, Tel-Aviv, Israel, 2000.
- [8] H. Wu, A. A. Abouzeid, "Error Robust Image Transport in Wireless Sensor Networks", *Proceeding of 5th workshop on Applications and Services in Wireless Networks(ASWN 2005)*
- [9] S. Shakkottai, T. S. Rappaport, C. Karlsson, "Cross-layer Design for Wireless Networks", *IEEE Communications manazine*, October, 2003
- [10] 선주호, 박충명, 이좌형, 김윤, 정인범, "무선 센서 네트워크에서의 데이터 전송속도 향상 기법", 2006 한국정보처리학회 추계학술발표대회 논문집 제13권 제2호