

# 무선센서네트워크에서 신뢰성있는 데이터수집을 위한 고장감내형 데이터 병합 기법

정회원 백 장 운\*, 남 영 진\*\*\*, 정 승 완\*\*, 종신회원 서 대 화\*\*o

## Fault Tolerant Data Aggregation for Reliable Data Gathering in Wireless Sensor Networks

Jang Woon Baek\*, Young Jin Nam\*\*\*, Seung-Wan Jung\*\* *Regular Members,*  
Dae-Wha Seo\*\*o *Lifelong Member*

### 요 약

본 논문에서는 에너지 효율적이고 신뢰성있는 데이터수집을 제공하는 고장감내형 데이터병합을 제안한다. 기존 데이터병합 기법은 패킷 손실에 대응하지 못하거나 대응 하더라도 에너지 소모가 매우 크다. 고장감내형 데이터병합은 적응적 타임아웃 데이터병합 기법에 트랙 토폴로지를 이용한 캐싱 및 재전송 기법을 적용하여 중요 이벤트에 대해 신뢰성있는 데이터수집을 제공한다. 고장감내형 데이터병합은 이벤트 가능성이 없는 정상시에는 기존의 트리 기반의 단일경로 데이터수집을 함으로써 에너지 소모를 줄인다. 하지만 이벤트 가능성이 감지되면 트랙 토폴로지를 이용한 패킷 손실 감지 및 재전송을 통해 데이터병합 결과의 정확도를 높인다. 실험 결과에서 고장감내형 데이터병합은 평균 소모 에너지 측면에서 TAG에 비해 약 8% 에너지 소모가 감소하였고, 이벤트 발생 가능성이 있을 경우 데이터 정확도 측면에서 TAG에 비해 41%정도 우수한 성능을 보였다. 그리고 평균 소모 에너지 측면에서 PERLA에 비해 약 53% 정도 에너지 소모가 감소하였으며, 이벤트 발생 가능성이 있을 경우 데이터 정확도 측면에서는 성능 저하가 거의 없었다.

**Key Words** : Wireless Sensor Networks, Data Aggregation, Fault-tolerant

### ABSTRACT

This paper proposes a fault-tolerant data aggregation which provides energy efficient and reliable data collection in wireless sensor networks. The traditional aggregation scheme does not provide the countermeasure to packet loss or the countermeasure scheme requires a large amount of energy. The proposed scheme applies caching and re-transmission based on the track topology to the adaptive timeout scheduling. The proposed scheme uses a single-path routing based on the traditional tree topology at normal, which reduces the dissipated energy in sensor nodes without any countermeasure against packet loss. The proposed scheme, however, retransmits the lost packet using track topology under event occurrences in order to fulfill more accurate data aggregation. Extensive simulation work under various workloads has revealed that the proposed scheme decrease by 8% in terms of the dissipated energy and enhances data accuracy 41% when the potential of event occurrence exists as compared with TAG data aggregation. And the proposed scheme decrease by 53% in terms of the dissipated energy and shows a similar performance in data accuracy when the potential of event occurrence exists as compared with PERLA data aggregation.

※ 본 연구는 2010년도 경북대학교 BK21사업과 교육과학기술부와 한국산업진흥원의 지역혁신인력양성사업 및 지식경제부 및 정보통신 산업진흥원의 대학IT연구센터 지원사업의 연구결과로 수행되었습니다.(NIPA-2010-C1090-1031-0009)

\* 한국전자통신연구원 대경권연구센터 자동치SW플랫폼연구팀 (jwbaek@etri.re.kr)

\*\* 경북대학교 IT대학 전자공학부 모바일컴퓨팅 & 임베디드시스템 연구실 ({tmdrod, dwseo}@ee.knu.ac.kr)(°:교신저자),

\*\*\* 대구대학교 컴퓨터·IT공학부 (yjn@daegu.ac.kr)

논문번호 : KICS2010-05-199, 접수일자 : 2010년 5월 6일, 최종논문접수일자 : 2010년 8월 19일

## I. 서 론

무선센서네트워크에서 데이터수집 시 센서노드의 에너지 소모를 줄이기 위해 인-네트워크 데이터병합이 많이 사용된다<sup>[1]</sup>. 인-네트워크 데이터병합은 주변 노드로부터 수신한 메시지를 병합합수를 이용한 로컬 프로세싱을 통해 병합 결과만 베이스 스테이션으로 전송한다. 따라서 인-네트워크 데이터병합은 전송메시지를 줄임으로써 센서노드의 에너지 소모를 줄이고, 네트워크 트래픽을 감소시킬 수 있다. 하지만 무선센서네트워크는 링크 어러들과 노드 고장확률이 높으므로 패킷 손실이 발생할 확률이 높다<sup>[2]</sup>. 트리 토폴로지의 무선센서네트워크가 인-네트워크 데이터병합을 사용할 때, 병합결과를 포함한 패킷 손실은 서브트리 전체의 패킷 손실을 의미한다.

기존의 TAG(Tiny AGgregation)<sup>[3]</sup>, TiNA(Temporal coherency-aware in-Network Aggregation)<sup>[4]</sup> 등의 인-네트워크 병합 기법은 고장으로 인한 패킷 손실에 대한 대응 방법을 제공하지 않는다. 또한, 이러한 인-네트워크 데이터병합은 중복 데이터에 민감하기 때문에 신뢰성있는 데이터수집을 위해 기존의 다중 경로 기법, 재전송 기법을 적용하는 것이 불가능하다. 해쉬기반 고장감내형 데이터병합 기법은 중복에 민감한 병합 함수 문제를 해결함으로써 높은 수준의 신뢰성있는 데이터수집을 제공한다<sup>[5]</sup>. 하지만 높은 처리량과 비교적 대규모의 저장 공간이 요구되고, 데이터 메시지에 포함되어야 하는 부가적인 비트 벡터로 인해 에너지와 대역폭 소모가 증가한다. 캐싱 및 재전송을 이용한 경로복구기법은 재전송 단계가 복잡하여 센서노드의 깨어있는 시간이 길어짐에 따라 에너지 소모 및 지연이 증가하는 단점이 있다<sup>[6-7]</sup>. 또한, 이러한 기법들은 정상시의 데이터와 이벤트 데이터의 구분 없이 동일한 데이터 전송 기법을 사용한다. 하지만 정상시에는 인접 지역에 위치한 센서노드로부터 수집되는 센싱 값이 비슷하기 때문에 패킷 손실이 발생하더라도 데이터병합 결과에 미치는 영향은 미비하다. 따라서 정상시의 수집 데이터에 대해서는 패킷 손실에 대한 대응을 제공하지 않더라도 시스템의 성능 저하가 거의 없다. 본 논문에서는 이러한 특성을 고려하여 이벤트 발생 가능성이 있을 경우에만 손실된 패킷에 대한 재전송을 제공한다.

인-네트워크 데이터병합에서 센서노드는 데이터수집기간 동안에만 프로세싱 및 통신을 수행하고, 그 밖의 시간에는 에너지 소모가 거의 없는 슬립모드로 동작한다. 데이터수집 기간 동안 센서노드는 병합동기를

맞추기 위해 자식노드로부터 메시지를 기다려야 하므로 이 기간 동안 많은 에너지를 소모한다<sup>[8]</sup>. 이때 병합동기를 맞추기 위해 자식노드로의 메시지를 기다리는 시간을 타임아웃이라고 한다. 타임아웃이 클수록 자식노드의 메시지가 병합에 포함될 확률이 크므로 정확도는 높아지지만 에너지 소모량이 증가한다. 따라서 정확도를 유지하면서 에너지 효율을 높일 수 있는 타임아웃을 사용하는 것이 중요하다. 본 논문에서는 데이터병합 동기화를 위해 적응적 타임아웃 스케줄링 기법을 사용한다<sup>[9]</sup>. 적응적 타임아웃 스케줄링 기법은 센서네트워크에서 수집되는 데이터의 특성에 따라 데이터병합 동기화를 위한 타임아웃과 데이터수집 주기를 동적으로 조절한다. 이벤트 가능성이 없는 정상시에는 타임아웃을 작게 하고 데이터수집 주기를 길게 하여 에너지 소모를 줄인다. 하지만 이벤트 가능성이 감지되면 타임아웃을 크게 하여 데이터병합 결과의 정확도를 높이고 데이터수집 주기를 짧게 하여 보다 신속하게 이벤트를 보고한다.

본 논문에서는 에너지 효율적이고 신뢰성있는 데이터수집을 제공하기위해 적응적 타임아웃 데이터병합 기법과 트렉 토폴로지를 이용한 캐싱 및 재전송 기법을 적용한 고장감내형 데이터병합을 제안한다. 고장감내형 데이터병합은 이벤트 가능성이 없는 정상시에는 기존의 트리 기반의 단일경로 데이터수집을 함으로써 에너지 소모를 줄이고, 이벤트 가능성이 감지되면 트렉 토폴로지를 이용한 패킷 손실 감지 및 재전송을 통해 데이터병합 정확도를 높인다.

본 논문의 구성은 다음과 같다. 2장에서는 무선센서네트워크에서 신뢰성을 보완하는 데이터병합 기법에 대해 살펴본다. 3장에서는 제안하는 고장감내형 데이터병합 기법에 대하여 자세히 설명한다. 4장에서 제안하는 고장감내형 데이터병합 기법의 성능을 분석하고, 마지막으로 5장에서 결론을 맺고 향후 과제를 제시한다.

## II. 배 경

다중 경로 라우팅 알고리즘은 단일 경로 라우팅 알고리즘에 비해 이론 및 실험에서 노드 고장에 훨씬 견고함을 보여주었다<sup>[2]</sup>. 다중 경로 라우팅 기법에서는 다수의 데이터 복사본이 여러 개의 경로를 통해 전송되므로, 다중 경로 중의 일부 경로에 노드 고장이 발생하더라도 메시지 전달은 영향을 받지 않는다. 하지만, 다중경로 라우팅은 데이터병합을 사용하기 어렵다. 다중경로 라우팅을 사용할 경우, 동일한 센싱 값

이 중복되어 데이터병합에 포함될 수 있다. 따라서 AVG, SUM, COUNT와 같은 중복에 민감한 병합 함수는 중복된 데이터로 인해 잘못된 결과 값을 낼 수 있다.

해쉬기반 기법은 무선센서네트워크에서 중복에 민감한 데이터병합의 문제를 해결하면서 신뢰성있는 데이터 전송을 제공하기 위해 개발되었다[5]. 해쉬기반 기법의 핵심은 중복에 민감한 병합함수를 중복에 독립적인 함수로 변환하는 것이다. 예를 들어, 중복에 민감한 병합 함수 중에 하나인 COUNT 함수는 bitwise OR 연산으로 변환할 수 있다. 즉,  $x \text{ OR } y$  는  $x \text{ OR } (y \text{ OR } y \text{ OR } y \dots)$ 와 동일한 결과를 가진다. Synopsis Diffusion은 다중경로라우팅과 해쉬기반 기법을 사용하여 신뢰성있는 데이터수집을 제공한다. 하지만 해쉬기반 기법은 다음과 같은 문제점을 가지고 있다. 첫째, 해쉬기반 기법은 높은 처리량과 비교적 대규모의 저장 공간이 요구된다. 둘째, 각 중복에 민감한 병합 함수에 대응하는 새로운 알고리즘이 고안되어야 한다. 셋째, COUNT와 같은 병합 함수의 경우에 상대적으로 큰 비트 벡터가 각 데이터 메시지에 포함되어야 하므로 추가적인 에너지와 대역폭을 소비한다. 넷째, 쿼리에 참여하는 노드의 수가 작아지면 정확한 병합 결과가 전달될 확률은 급격히 떨어진다.

경로 복구 라우팅 알고리즘은 단일 경로를 통해 메시지를 전송하고 노드 고장이 있을 경우에 다른 경로를 찾아 메시지를 전송한다<sup>[6]</sup>. 그림 1은 로컬 노드 기반의 경로 복구 알고리즘인 PERLA (Power Efficient Routing with Limited Latency)의 데이터수집 과정을 보여준다<sup>[7]</sup>. PERLA는 암묵적 응답 (implicit acknowledgement)을 이용하여 노드 고장을 감지한다. 라우팅 트리의 부모노드 고장이 감지되면, 소스 노드로 고장 알림 메시지를 보내지 않고 이웃 노드들 중에서 새로운 부모노드를 선택하고 로컬 캐시에 저장된 데이터를 새로운 부모노드로 바로 재전송한다. 그림 2는 데

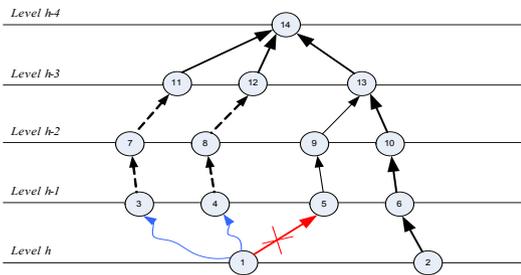


그림 1. PERLA의 데이터수집 시나리오

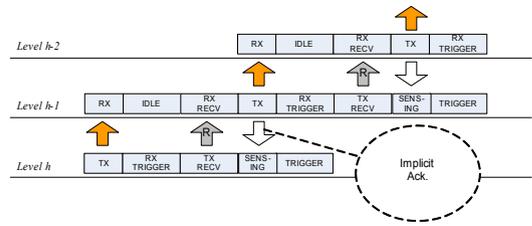


그림 2. PERLA의 메시지 전송 단계.

이터병합시 패킷의 송수신, 에러 감지 및 재전송 단계를 보여준다. PERLA는 경로 복구시간을 단축시키고 부가적인 메시지 송수신으로 인해 발생하는 에너지 소모를 줄인다. 하지만 재전송 단계가 복잡하여 센서 노드의 깨어있는 시간이 길어짐에 따라, 에너지 소모 및 지연이 증가한다.

### III. 제안 알고리즘

고장감내형 데이터병합은 트래크 토폴로지를 이용한 캐싱 및 재전송 기법을 적용하여 중요 이벤트에 대해 신뢰성있는 데이터수집을 제공한다. 중요 이벤트에 대한 신뢰성은 실제 모든 센서노드에서 수집된 데이터의 병합 값과 센서네트워크를 통해 수집된 병합 값의 동일한 정도를 의미한다. 트래크 토폴로지는 그림 3과 같고, 베이스 스테이션은 트래크  $T_0$ 에 속하고, 베이스 스테이션으로부터  $n$  홉 떨어진 노드는 트래크  $T_n$ 에 속한다. 트래크 토폴로지에서 각 센서노드는 자식노드 리스트, 부모노드 리스트, 그리고 이웃노드 리스트를 유지한다. 그리고 각 노드는 주부모노드 하나와 다수의 백업-부모노드를 가질 수 있다. 주부모노드는 라우팅 트리를 형성하고, 에러가 발생하지 않을 경우에 패킷은 주부모노드를 통해 베이스 스테이션으로 전송된다. 주부모노드와의 링크에 에러가 발생하더라도 무선 전송매체의 브로드캐스팅 특성에 의해 백업-부모노드가 성공적으로 전송된 패킷을 수신할 수도 있다.

고장감내형 데이터병합을 지원하기 위한 트래크 토폴

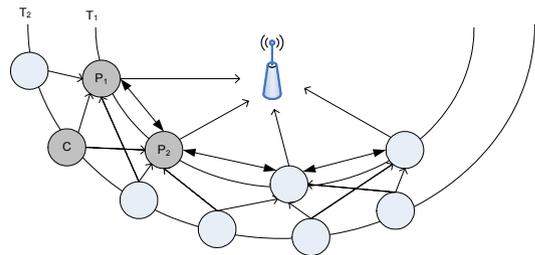


그림 3. 트래크 토폴로지

로지의 생성과정은 다음과 같다. 먼저, 베이스 스테이션에서 토폴로지 생성을 위한 요청 메시지를 브로드캐스팅 한다. 요청 메시지는 센서노드의 아이디와 홉수를 포함하고 있다. 요청 메시지를 수신한 센서노드는 자신의 홉수가 수신메시지의 홉수보다 작거나 같지 않으면 수신 메시지의 홉수를 하나 증가시킨 값을 자신의 홉수로 할당한다. 자신보다 홉수가 작은 노드는 부모노드 리스트에 추가하고, 홉수가 같은 노드는 이웃노드 리스트에 추가한다. 그리고 부모노드 중에서 연결성이 좋은 노드를 자신의 주-부모노드로 선택하고 응답 메시지를 주-부모노드로 전송한다. 응답 메시지를 수신한 주-부모노드는 응답 메시지를 전송한 노드를 자식노드 리스트에 추가한다. 그리고 주-부모노드는 이웃노드의 자식노드로 부터 응답 메시지를 엿들을 수 있고, 응답 메시지를 전송한 노드를 이웃 자식노드 리스트에 추가한다. 이 과정 후에 센서노드는 자신의 아이디와 홉수를 삽입하여 트랙 토폴로지 요청 메시지를 다시 브로드캐스팅 한다. 트랙 토폴로지 요청 메시지는 모든 센서노드가 자신의 홉수를 할당 받고 부모노드를 선택할 때까지 이러한 방식으로 계속 브로드캐스팅 된다. 베이스 스테이션은 주기적으로 요청 메시지를 브로드캐스팅 하여 트랙 토폴로지의 연결을 유지 한다. 또한, 네트워크 토폴로지 생성 기간 동안 센서노드는 자식노드의 아이디와 비트 벡터 내의 비트 포지션을 브로드캐스팅 하고, 이를 수신한 각 센서노드가 이웃노드의 비트 벡터 내부의 자식노드의 비트 포지션을 알 수 있게 한다.

고장감내형 데이터병합은 이벤트 가능성이 없는 정상시에는 기존의 트리 기반의 단일경로 데이터수집을 함으로써 에너지 소모를 줄인다. 하지만 이벤트 가능성이 감지되면 트랙 토폴로지를 이용한 패킷 손실 감지 및 재전송을 통해 데이터병합 결과의 정확도를 높인다. 고장감내형 데이터병합은 적응적 타임아웃을 통해 에너지 효율을 높이고, 응용 애플리케이션이 요구하는 실시간성을 제공한다. 알고리즘 1은 제안하는 고장감내형 데이터병합의 동작 알고리즘을 보여준다. 제안 고장감내형 데이터병합은 센서노드 ( $node_i$ )상에서 동작한다. 센서노드( $node_i$ )는 적응적 타임아웃 스케줄링에 따라  $a\%$ 의 자식노드 메시지 수신을 보장하도록 타임아웃 ( $Timeout_i = SHD_a$ )을 초기화하고, 로컬 센싱 값으로부터 이벤트 가능성 ( $PL_i$ )을 파악한다. 센서노드의 로컬 센싱 값이 이벤트 가능성이 없으면 적응적 타임아웃 스케줄링에 정의된 타임아웃( $SHD_a$ ) 동안 자식노드의 메시지 ( $data_{child(i)}$ )를 수신한다. 이때, 수신하는  $data_{child(i)}$ 는  $node_i$ 를 주-부모노드(primary

parent)로 선택한 자식노드( $node_{p\_child(i)}$ )의 데이터인  $data_{p\_child(i)}$ 와  $node_i$ 를 백업-부모노드(backup parent)로 선택한 자식노드( $node_{b\_child(i)}$ )의 데이터인  $data_{b\_child(i)}$ 를 포함한다. 센서노드는 타임아웃( $SHD_a$ ) 이내에 모든  $node_{p\_child(i)}$ 로부터 메시지를 수신하거나 타임아웃( $SHD_a$ ) 동안 수신한 메시지가 이벤트 가능성이 없으면 수신한 데이터 ( $data_{p\_child(i)}$ )와 로컬데이터 ( $data_i$ )를 병합하여 바로 주-부모노드( $node_{p\_par(i)}$ )로 병합데이터( $data_i$ )를 전송한다. 그리고 다음 데이터수집까지 데이터수집주기( $T_0$ ) 동안 슬립 한다. 하지만 센서노드의 로컬 센싱 값 ( $data_i$ )이 이벤트 가능성을 내재하고 있거나 자식노드 ( $node_{p\_child(i)}$ )로부터 이벤트 메시지를 받으면 주-부모노드 ( $node_{p\_par(i)}$ )로 즉시 이벤트를 보고한다. 그리고 이벤트에 대해 보다 정확한 병합 값을 얻기 위해 타임아웃 값을 적응적 타임아웃 스케줄링에 따라 최대값( $Timeout_i = SHD_a + MRG_i$ )으로 증가시킨다. 타임아웃의 최대값은 정상시의 타임아웃 값  $SHD_a$  에 타임아웃 여유분 ( $MRG_i$ )만큼 타임아웃을 증가시킨 것이다.  $SHD_a$ 과 홉간지연의 최대값

**Algorithm 1: Fault-tolerant data aggregation**

```

for each  $node_i$  in the network do
hopCount $_i$  = hopCount $_{par(i)}$  + 1;
MRG $_i$  = (N-hopCount $_i$ ) *  $\xi$ ;
Timeout $_i$  = SHD $_a$ ;
start TIMER with  $T_{cur}=0$ ;
PL $_i$  ← potential( $data_i$ );
if (PL $_i$ !=0)/*There are no events.*/
while  $T_{cur} \leq Timeout_i$  do
if(received the event with PL $_{child(i)}$ ) goto L1;
elseif(received every  $data_{p\_child(i)}$ ) goto L2;
else receive  $data_{child(i)}$  from  $node_{child(i)}$ ;
repeat
else /*Event potential exists.*/
L1: inform  $node_{par(i)}$  of events with PL $_i$  or PL $_{child(i)}$ ;
Timeout $_i$  = SHD $_a$  + MRG $_i$ ;
while  $T_{cur} \leq Timeout_i$  do
receive  $data_{child(i)}$  and  $data_{b\_child(child(i))}$  from
 $node_{child(i)}$ ;
repeat
PL $_i$  ← max(PL $_i$ , PL $_{child(i)}$ );
endif
L2:  $data_i$  ← aggregate( $data_{p\_child(i)}$ ,  $data_i$ );
send  $data_i$  to  $node_{par(i)}$ ;
if (PL $_i$ !=0) /*Potential of events exists.*/
then listen  $data_{ngb(i)}$  from  $node_{ngb(i)}$ 
and detect the missing  $data_{p\_child(ngb(i))}$ ;
if(missing!) then retransmit the  $data_{b\_child(i)}$  to
 $node_{par(i)}$ ;
// $data_{b\_child(i)} = data_{p\_child(ngb(i))}$ 
endif
sleep during  $T_k - Timeout_i$ ; //k=PL $_i$ 

```

( $SHD_{max}$ )의 차가  $\delta$ , 노드의 홉수가  $hopCount_i$ 이고 라우팅 트리의 깊이가  $N$ 일 때, 센서노드의 타임아웃 여유분 ( $MRG_i$ )은  $(N-hopCount_i) * \delta$ 이다. 타임아웃 최대 값 동안 센서노드는 자식노드로부터 병합데이터와 재전송되는 패킷을 수신한다. 타임아웃이 만료되면 데이터를 병합하여 주부모노드( $node_{p\_par(i)}$ )로 전송하고, 동시에 이웃노드가 전송하는 패킷을 엿듣는다. 만약 이웃노드가 전송한 병합데이터 ( $data_{ngb(i)}$ )에 포함되어야 하는 자식노드의 데이터 패킷( $data_{p\_child(ngb(i))}$ )이 손실되었다면 그 데이터 패킷( $data_{b\_child(i)}$ )을 주 부모노드( $node_{p\_par(i)}$ )로 재전송한다. 그리고 적응적 타임아웃 스케줄링에 의해 무선센서네트워크의 이벤트 발생 가능성( $PL_i$ )에 따라 데이터수집 주기를 조절한다. 이벤트 발생 가능성( $PL_i$ )이  $k$ 단계에 속하면 센서노드는 정상시의 수집주기( $T_0$ )를 이벤트가능성  $k$ 단계의 수집주기( $T_k$ )로 줄이고, 다음 데이터수집을 시작할 때까지 수정된 주기( $T_k$ )동안 슬립한다.

고장감내형 데이터병합은 정상시의 데이터수집 때 센서노드의 에너지 소모를 줄이기 위해 패킷 손실에 대한 재전송을 하지 않는다. 그림 4는 정상시에 데이터를 수집할 때, 패킷 전달 과정을 보여준다. 이때, 무선센서네트워크는 기존 트리 토폴로지에서 단일 경로를 통해 메시지를 전송하는 것과 동일하다. 이 경우에 그림 4에서와 같이 링크 혹은 노드 고장이 발생하면 패킷 손실이 발생하여 데이터병합 결과의 정확도가 떨어질 수 있다. 하지만 정상시에 수집되는 데이터는 센싱 값의 변화가 작고 이웃노드에서 수집한 값과 유사한 값을 가지므로 일부 자식노드의 데이터가 병합에 포함되지 않더라도 전체 자식노드의 데이터를 병합한 결과와 병합결과 값의 차이는 거의 없다. 따라서 본 논문에서는 이러한 특성을 이용하여 정상시에는 링크 및 노드 고장에 상관없이 대응 없이 데이터를 수집한다. 그림 5는 센서노드의 라우팅 트리 레벨에 따

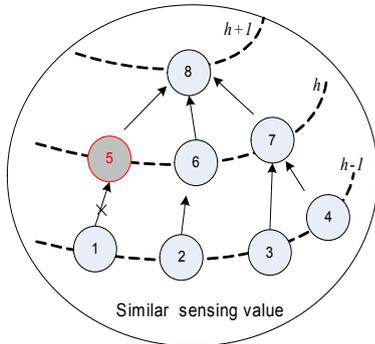


그림 4. 정상시의 데이터수집 시나리오

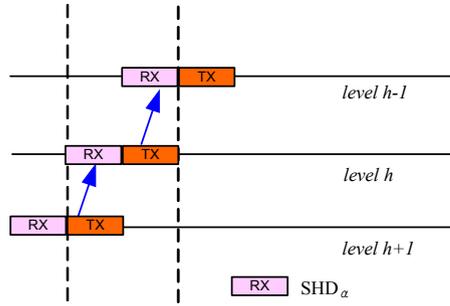


그림 5. 정상시의 메시지 전송 단계

른 송수신 상태 변화를 보여준다. 센서노드는 타임아웃 스케줄링에 따라 메시지 송수신 기간에만 깨어있고 그 외의 시간에는 슬립상태를 유지한다. 센서노드는 자식노드의 메시지 전송시점부터 타임아웃 기간 동안 자식노드의 메시지를 기다리고, 수신한 메시지를 병합하여 부모노드로 메시지를 전송한다.

이벤트 발생 가능성이 있을 경우에 고장감내형 데이터병합은 이벤트 정보를 즉시 부모노드로 전송하고 정확한 데이터를 병합하기 위해서 손실된 패킷에 대해 재전송을 수행한다. 이벤트 발생가능성이 높을 때에 센서노드에서 수집되는 데이터는 이전 주기에 수집된 데이터에 비해 시간적 변화량이 크고, 인접 노드들에서 수집된 데이터와 차이가 크다. 따라서 이벤트에 대해 보다 정확한 병합 결과를 얻을 수 있도록 패킷 손실을 가능한 줄일 수 있어야 한다. 그림 6은 이벤트 가능성이 있을 때, 제안하는 데이터병합 기법에서의 패킷 전달 과정을 보여준다. 여기서 무선센서네트워크는 트래크 토폴로지를 가지고 각 노드는 주변노드(자식노드, 부모노드, 이웃노드)의 패킷 전송을 엿들을 수 있다. 무선센서네트워크에서 센서노드는 무선채널을 공유하기 때문에 자신을 목적으로 하지 않는 패킷을 수신할 수 있다. 본 논문에서는 두 노드간의 패킷 손실이 발생하였을 때, 하나이상의 다른 센서노드가 그 손실 패킷을 정확하게 수신할 수 있다고 가정한다. 이러한 가정은 다중 경로 및 경로 복구 라우팅 알고리즘에서 사용되고 있으며, 센서노드의 라디오 수신 범위 내에 한 개 이상의 센서노드가 존재하면 만족된다. 본 논문에서 센서노드는 데이터 메시지에 첨부된 비트 벡터를 이용하여 링크 에러로 인한 자식 노드의 패킷 손실을 감지한다. 비트 벡터는 자식노드의 데이터가 정확하게 수신되어 병합에 포함되었는가를 나타낸다. 네트워크 토폴로지 생성 기간 동안 센서노드는 자식노드의 ID와 비트 벡터 내의 비트 포지션을 브로드캐스팅 하고, 이를 수신한 각 센서노드는 이웃

노드의 비트 벡터 내부의 자식노드의 비트 포지션을 알 수 있다. 비트 벡터는 각 자식노드 당 2개의 비트를 포함한다. 첫 번째 비트는 주-부모노드와 링크에러 유무를 표시하는 e-bit (error-bit)이다. 주-부모노드가 자식노드로부터 패킷을 수신하지 못하면 e-bit를 1로 설정한다. 두 번째 비트는 센서노드가 에러를 정정하거나 에러 정정을 돕고 있는 상태를 표시를 표시하는 r-bit (recovery-bit)이다. 백업-부모노드가 링크 에러가 발생한 자식노드의 패킷을 재전송할 때 r-bit를 1로 설정한다.

그림 6에서 노드 1의 주-부모노드는 노드 5이고 백업-부모노드는 노드 6이다. 주-부모노드는 메시지를 수신하여 데이터병합을 수행한 후 부모노드로 전송하는 노드이고, 백업-부모노드는 주-부모노드의 메시지 통신 에러를 대비하여 자식노드의 메시지를 수신하여 임시로 저장하는 노드이다. 그림 6에서 노드 5는 링크 에러로 인해 노드 1에서 전송한 패킷을 수신하지 못한다. 노드 5는 병합메시지에 노드 1의 메시지를 수신하지 못했음을 비트 벡터내의 노드 1의 비트 포지션의 e-bit를 1로 표시하여 부모노드로 병합 메시지를 전송한다. 노드 6은 노드 5가 전송한 데이터 패킷을 엿듣고, 노드 1의 비트 포지션의 e-bit가 1인 것을 보고 노드 1의 패킷이 병합에 포함되지 않은 것을 감지한다. 그리고 노드 6은 비트벡터내의 노드 1의 비트 포지션의 r-bit를 1로 설정하고, 노드 5의 데이터병합에 포함되지 않은 노드 1의 데이터 패킷을 주-부모노드인 노드 8로 전송한다. 노드 8은 자식노드 (노드 5, 6, 7)로부터 수신한 데이터와 재전송된 노드 1의 데이터 그리고 자신의 로컬 센싱 값을 병합하여 주-부모노드로 전송한다. 따라서 링크 에러로 인한 데이터 손실을 막을 수 있다. 그림 7는 이벤트 발생 가능성이 있을 때의 라우팅 트리 레벨에 따른 송수신 상태 변화를

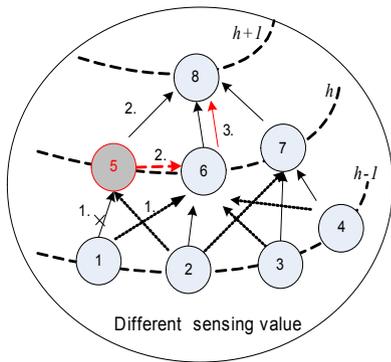


그림 6. 이벤트시의 데이터수집 시나리오

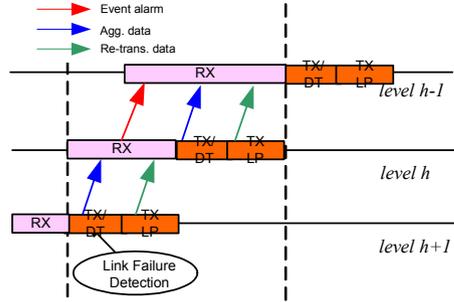


그림 7. 이벤트시의 메시지 전송 단계.

보여준다. 센서노드는 타임아웃 스케줄링에 따라 메시지 송수신 기간에만 깨어있고 그 외의 시간에는 슬립 상태를 유지한다. 센서노드는 자식노드의 메시지 전송 시점부터 타임아웃 기간 동안 자식노드의 메시지를 기다리고, 수신한 메시지를 병합하여 부모노드로 메시지를 전송한다. 고장감내형 데이터병합은 적응적 타임아웃 스케줄링에 따라서 이벤트 발생 가능성이 있을 경우에는 자식노드의 메시지 수신기간인 타임아웃을 최대값으로 유지한다. 타임아웃 기간 동안 센서노드는 자식노드의 병합데이터와 링크 에러로 인해 재전송된 패킷을 수신한다. 그리고 타임아웃이 만료되면 자식노드의 데이터와 재전송된 데이터를 병합하여 부모노드로 전송한다. 그와 동시에 이웃노드의 전송 메시지를 수신하여 패킷 손실을 감지한다.

#### IV. 성능 평가

NS-2 네트워크 시뮬레이터<sup>[10]</sup>를 이용하여 제안하는 고장감내형 데이터병합과 TAG 데이터병합, PERLA 데이터병합의 성능을 비교하였다.

##### 4.1 시뮬레이션 환경

표 1은 실험에 사용된 시뮬레이션 파라미터를 보여준다. 본 논문에서는 최신 센서노드의 한 종류인 MG2455를 기준으로 센서노드의 파라미터를 설정하였다. 센서노드는 PHY/MAC프로토콜로 IEEE 802.15.4를 사용하였다. 주파수 대역은 2.4GHz이고, 이때의 데이터 전송속도는 250kbps이다. 실험에서 무선센서네트워크는 트리 토폴로지를 가지고, 센서노드의 송수신 전파범위는 15m로 설정하였다. 송신 전력은 43.2mW, 수신 전력은 33mW, 그리고 슬립모드의 전력 소모는 0.002mW로 설정하였다. 전송 패킷의 사이즈는 70bytes 이다.

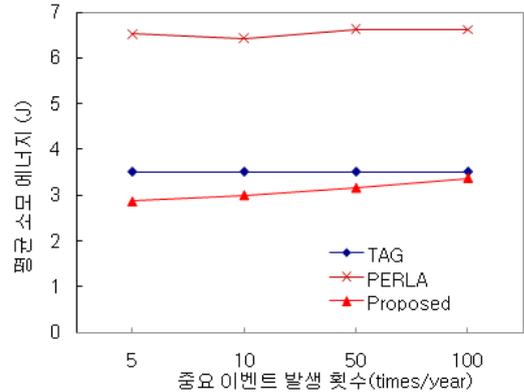
표 1. 시뮬레이션 파라미터

파라미터	값
PHY/MAC 프로토콜	IEEE 802.15.4
라디오 전송 범위	15m
전송 속도	250kbps
송신 전력	43.2mW
수신 전력	33mW
슬립 전력	0.002mW
패킷 사이즈	70bytes

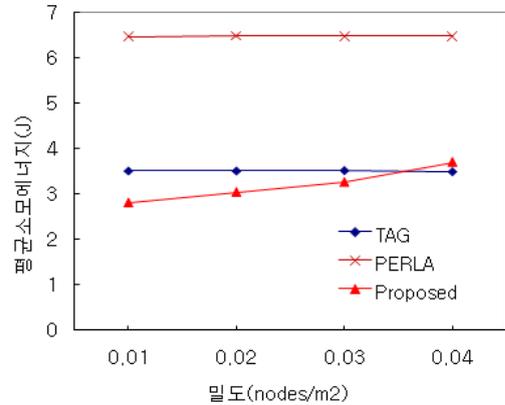
본 논문에서는 산불을 감지하는 무선센서네트워크를 대상으로 실험하였다. 산불 감지 애플리케이션은 주로 습도와 온도 센싱 값을 주기적으로 베이스 스테이션에 전송하여 사용자에게 현재 환경 상태를 보여 주고, 습도와 온도 센싱 값으로부터 FFMC(Fine Fuel Moisture Code) 값을 산출하여 산불발생 가능성을 알려준다. 본 논문에서는 간단히 산불 발생 가능성의 척도인 FFMC 값을 수집한다고 가정한다. 센서노드는 병합합수 AVG, MAX를 이용하여 주기적으로 FFMC의 평균값과 최대값을 전송한다. 센서노드는 60초마다 주기적으로 수집된 데이터를 베이스 스테이션으로 전송한다. 정상시의 산불 가능성이 없을 때 센서노드는 FFMC 값이 70에서 76사이의 값을 부모노드로 전송한다. 하지만 이벤트 가능성이 있는 기간 동안에는 76에서 100사이의 FFMC 값을 베이스 스테이션으로 전송한다. 제한하는 적응적 타임아웃 데이터병합을 적용한 센서노드는 이벤트 가능성이 있는 76에서 100사이의 FFMC 값을 수집할 경우에 동적으로 타임아웃 값과 데이터수집 주기(60초 이하)를 조절한다. 반면에, TAG와 캐스캐이딩 타임아웃 병합은 FFMC 값에 상관없이 데이터수집 주기를 60초로 일정하게 유지한다. 그리고 시뮬레이션에서 FFMC 값이 85 이상일 때 이벤트가 발생한 것으로 간주한다. 총 시뮬레이션 기간은  $10^5$ 초로 설정하였다.

#### 4.2 에너지 소모 비교

그림 8의 (가)에서, 고장감내형 데이터병합 기법은 이벤트의 발생 횟수가 증가할수록 에너지 소모량이 증가함을 알 수 있다. 이벤트가 발생하면 제안 기법에서 타임아웃을 늘리고 패킷 재전송을 수행한다. 그리고 이벤트 가능성이 증가할수록 데이터수집 빈도를 증가시킨다. 따라서 중요이벤트가 자주 발생할수록 에너지 소모량이 증가한다. 하지만 실제 무선센서네트워크 환경에서 중요이벤트는 자주 발생하지 않기 때문에 제안 알고리즘은 TAG나 PERLA 기법에 비해 에



(가) 이벤트 발생 횟수에 따른 에너지 소모량 (밀도 =  $0.02nodes/m^2$ , 트리 깊이 = 6, 링크 에러률 = 10%)



(나) 밀도에 따른 에너지 소모량 (이벤트 발생 횟수 = 10times/year, 트리 깊이 = 6, 링크 에러률 10%)

그림 8. 각 데이터병합 기법의 평균소모에너지 비교 TAG 데이터병합, PERLA 데이터병합, 고장감내형 데이터병합

너지 효율적이다. 특히, PERLA는 이벤트에 상관없이 동일한 상대적으로 긴 wake-up시간을 가지고, 패킷 재전송을 수행하므로 에너지 소비가 크다. 제안 기법은 PERLA에 비해 재전송 단계가 간단하고 센서노드가 깨어있는 시간이 PERLA의 반 정도 밖에 되지 않는다. 이벤트 발생 횟수가 5일 때 제안 기법이 PERLA에 비해 에너지 소모량이 53%정도 작음을 알 수 있다.

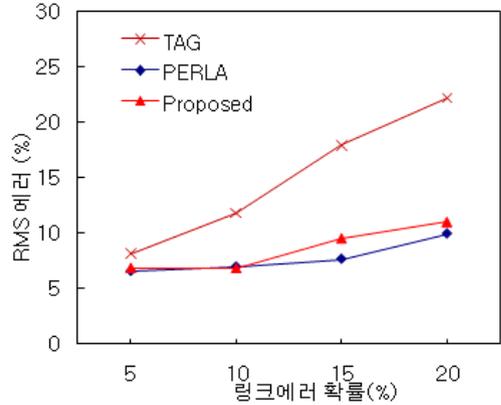
그림 8의 (나)는 센서노드의 밀도(density)를 변경하면서 TAG, PERLA 그리고 제안 기법의 평균 에너지 소모량을 비교한 것이다.  $200*200$ 으로 고정된 지역에 센서노드의 수를 증가시키면 센서노드의 밀도가 증가한다. 제안기법은 밀도가 증가할수록 에너지 소모량이 증가하고, 밀도가  $0.03nodes/m^2$ 에서  $0.04nodes/m^2$  사이에서 고장감내형 데이터병합이 TAG보다 에너지

소모가 더 큰 것을 볼 수 있다. 제안기법은 밀도가 높을수록 채널을 공유하는 센서노드가 증가하여 정상시의 타임아웃 값이 증가하고 TAG의 타임아웃 값에 수렴한다. 따라서 밀도가 어느 시점까지 증가하면 정상시의 데이터수집 시 제안기법은 TAG의 에너지 소모량과 거의 같다. 전체 시뮬레이션 기간 중 이벤트 발생 빈도는 매우 적은편이지만, 이벤트 가능성이 있을 때 고장감내형 데이터병합을 적용한 데이터수집은 TAG 데이터병합 보다 2배 이상의 에너지를 소모하기 때문에 제안기법의 에너지 소모가 TAG 보다 크다. TAG와 PERLA는 밀도에 상관없이 고정된 타임아웃을 사용하므로 에너지 소모량에 거의 변화가 없다.

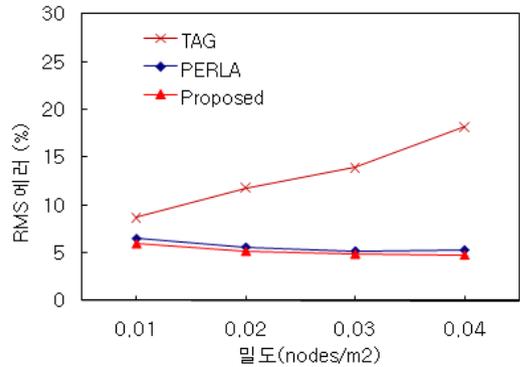
4.3 중요 이벤트의 평균 전달률 비교

그림 9는 이벤트 발생 시 각 데이터병합 기법에 따른 RMS 에러를 비교한 것이다. 이벤트 발생시에는 각 센서노드에서 수집되는 데이터는 이전 수집데이터와 차이가 크고, 이벤트 발생지점으로부터의 거리에 따라 센싱 값의 차이가 크므로 패킷 손실로 인한 병합데이터의 RMS 에러가 크다. 그림 11의 (가)는 이벤트 발생시의 무선센서네트워크에서 링크에러 확률을 변경하면서 TAG, PERLA 그리고 제안 고장감내형 데이터병합의 병합 데이터의 정확도를 비교한 것이다. PERLA와 고장감내형 데이터병합이 링크에러에 대한 패킷 재전송을 수행함으로써, 링크 에러가 있음에도 RMS 에러 값을 작게 유지한다. 반면에 TAG 경우에는 링크 에러율이 증가함에 따라 RMS 에러가 급격하게 증가하는 것을 볼 수 있다. 무선센서네트워크의 링크 에러율이 10%일 때, 고장감내형 데이터병합은 TAG보다 대략 41%정도 데이터 정확도 측면에서 성능 향상을 보였다.

그림 9의 (나)는 이벤트 발생시에 네트워크 밀도를 변경하면서 TAG, PERLA, 그리고 고장감내형의 병합 데이터의 정확도를 비교한 것이다. TAG 데이터병합에서는 밀도가 증가할수록 RMS 에러가 점차 증가함을 볼 수 있다. 밀도가 증가할수록 패킷의 충돌가능성이 증가하여 타임아웃 이내에 도착하지 못하는 패킷이 자주 발생한다. 반면에, PERLA와 고장감내형 데이터병합은 TAG에 비해 센서노드가 깨어 있는 시간이 2배 이상 더 길기 때문에 자식노드의 메시지를 수신할 확률이 증가한다. 또한, 밀도가 증가함에 따라 이웃 노드의 수가 증가하게 되고, 백업-부모노드가 손실 패킷을 수신할 확률이 높아진다. 따라서 PERLA와 고장감내형 데이터병합은 밀도가 증가함에 따라 메시지 수신 및 패킷 손실 시에 재전송확률이 높아지므로



(가) 링크 에러율에 따른 RMS 에러 (이벤트 발생 횟수 = 10times/year, 밀도 = 0.02 nodes/m<sup>2</sup>)



(나) 밀도에 따른 RMS 에러 (이벤트 발생 횟수 = 10times/year, 링크 에러율 = 10%, 트리 깊이 = 6)

그림 9. 각 데이터병합 기법의 이벤트시의 병합 데이터 정확도 비교: TAG 데이터병합, PERLA 데이터병합, 고장감내형 데이터병합

RMS 에러가 더 감소하는 현상을 보인다.

V. 결론 및 향후 연구과제

고장감내형 데이터병합은 이벤트 가능성이 없는 정상시에는 기존의 트리 기반의 단일경로를 통해 데이터를 수집하고, 이벤트 가능성이 감지되면 트랙 토폴로지를 이용한 패킷 손실 감지 및 재전송 수행하면서 데이터를 수집 한다. 시뮬레이션을 통해 제안하는 고장감내형 데이터병합 기법과 기존 TAG, PERLA 데이터병합 기법의 성능을 비교하였다. 실험 결과에서 고장감내형 데이터병합은 평균 소모 에너지 측면에서 TAG에 비해 약 8%, PERLA에 비해 약 53% 정도 에너지 소모를 줄였으며, 데이터 정확도 측면에서 TAG

에 비해 41%정도 우수한 성능을 보였다. 따라서 제안하는 고장감내형 데이터병합은 링크 에러에 대한 신뢰성을 제공하면서 높은 에너지 효율을 제공하는 것을 알 수 있다.

향후 연구에서는 실제 센서네트워크 환경에 적용하여 제안하는 고장감내형 데이터병합 기법의 성능을 테스트하고 성능을 분석하여 응용분야에 실제로 적용될 수 있도록 연구를 지속할 예정이다.

### 참 고 문 헌

[1] S. Madden, R. Szewczyk, M. Franklin and D. Cullera, "Supporting Aggregate Queries Over Ad-Hoc Wireless Sensor Networks," *Proceedings of 4th IEEE Workshop on Mobile Computing and Systems Applications*, pp. 49-58, 2002.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, 38, pp.393- 422, Mar. 2002.

[3] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, "TAG: a Tiny Aggregation Services for Ad-Hoc Sensor Networks," *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation*, 36, pp.131-146, Dec. 2002.

[4] M. Sharaf, J. Beaver, A. Labrinidis and P. Chrysanthis, "Balancing Energy Efficiency and Quality of Aggregate Data in Sensor Networks," *The VLDB Journal*, 13, pp.384-403, Dec. 2004.

[5] S. Nath, P. Gibbons, S. Seshan, and Z. Anderson, "Synopsis diffusion for robust aggregation in sensor networks," *ACM Transactions on Sensor Networks*, 4, Mar. 2008.

[6] D. Tian and N. Georganas, "Energy Efficient Routing with Guaranteed Delivery in Wireless Sensor Networks," *Proceedings of the IEEE Wireless Communications and Networking Conference*, 3, pp.1923-1929, Mar. 2003.

[7] D. Messina, M. Ortolani, and G. Re, "Achieving Robustness through Caching and Retransmissions in IEEE 802.15.4-based WSNS," *Proceedings of Sixteenth International Conference on Computer Communications and*

*Networks*, pp.1117-1122, 2007.

[8] I. Solis and K. Obraczka, "In-Network Aggregation Trade-offs for Data Collection in Wireless Sensor Networks," *International Journal of Sensor Networks*, 1, pp.200-212, Sep. 2006.

[9] J. Baek, Y. Nam, S. Jung, and D. Seo, "An Adaptive Timeout Aggregation for Periodic Data Gathering in Wireless Sensor Networks," *Proceedings on the 4th International Conference on Ubiquitous Information Technologies & Applications*, pp.1-6, Dec. 2009.

[10] VINT, "The Network Simulator NS-2," <http://www.isi.edu/nsnam>, Nov. 2005.

백 장 운 (Jang Woon Baek)

정회원



2002년 2월 경북대학교 전자전기공학부  
2004년 2월 경북대학교 전자공학과 석사  
2009년 8월 경북대학교 전자공학과 박사  
2009년 8월~현재 한국전자통신연구원 자동차SW플랫폼연구팀

<관심분야> 센서네트워크, 모바일 컴퓨팅, 임베디드 시스템

남 영 진 (Young Jin Nam)

정회원



1992년 2월 경북대학교 전자공학과  
1994년 2월 포항공과대학교 전자전기공학과 석사  
2004년 3월 포항공과대학교 컴퓨터공학과 박사  
1994년~1998년 한국전자통신연구원 컴퓨터연구단

2004년~현재 대구대학교 컴퓨터IT공학부 조교수  
<관심분야> 네트워크 스토리지, 임베디드 SW, 무선 네트워크

정 승 완 (Seung-Wan Jung)

정회원



2006년 2월 경북대학교 전자  
전기컴퓨터학부  
2008년 2월 경북대학교 전자  
전기컴퓨터학부 석사  
2008년 3월~현재 경북대학교  
전자전기컴퓨터학부 박사과정  
<관심분야> 임베디드 SW, 파  
일 시스템, 센서 네트워크

서 대 화 (Dae-Wha Seo)

중신회원



1981년 2월 경북대학교 전자  
공학과  
1983년 2월 한국과학기술원 전  
산학과 석사  
1993년 2월 한국과학기술원 전  
산학과 박사  
1983년~1995년 한국전자통신  
연구원 시스템S/W연구실  
2004년~현재 경북대학교 임베디드소프트웨어연구  
센터장  
1998년~현재 경북대학교 전자전기컴퓨터학부 교수  
<관심분야> 임베디드 SW, 병렬처리, 분산운영체제