

수명 연장 기능의 무선 센서 네트워크용 EIBS 알고리즘

배 시 규*

An EIBS Algorithm for Wireless Sensor Network with Life Time Prolongation

Shi-Kyu Bae *

요 약

무선 센서 네트워크에서도 다른 네트워크와 마찬가지로 시간 관계를 유지하는 것이 중요하므로, 무선 센서 네트워크의 특성을 고려한 시각 동기화 프로토콜인 IBS(Indirect-Broadcast Synchronization)가 2012년에 개발되었다. IBS는 클러스터 트리 형태에서 동작하므로 일반 노드보다 더 많은 에너지를 소모하는 클러스터 헤더에 의해서 주로 네트워크 전체 수명이 좌우될 수 있다. 이 논문에서는 클러스터 트리를 부분적으로 재구성함으로써 전체 에너지 소모를 줄이고 네트워크 수명을 연장시키는 향상된 IBS(EIBS)를 개발하고자 한다. 다른 트리 구성 접근 방식과 비교할 때, 이 논문에서 사용된 방법은 알고리즘이 간단할 뿐만 아니라 전력 소모나 네트워크 수명 면에서 더 효율적이다.

▶ Keywords : 시각 동기화, 무선센서네트워크, 전력 소모, 트리 재구성

Abstract

Since Time synchronization is also critical in Wireless Sensor Networks (WSN) like other networks, a time synchronization protocol for WSN called IBS(Indirect-Broadcast Synchronization) has been already proposed in 2012. As IBS operates in cluster tree topology, network lifetime may be mainly shortened by cluster head node(s), which usually consumes more power than cluster member (i.e. non-cluster head) nodes. In this paper, I propose enhanced version of IBS (called EIBS) which saves overall energy and prolongs network lifetime by re-constructing partial cluster tree locally. Compared with other tree construction approaches, this tree reconstruction algorithm is not only simpler, but also more efficient in the light of overall power consumption and network lifetime.

▶ Keywords : Power Consumption, Time Synchronization, Tree-Reconstruction, Wireless Sensor Network

•제1저자 : 배시규 •교신저자 : 배시규

•투고일 : 2014. 7. 9, 심사일 : 2014. 7. 27, 게재확정일 : 2014. 8. 11.

* 동양대학교 컴퓨터공학과(Dept. of Computer Engineering, DongYang University)

I. 서론

무선센서네트워크(WSN : Wireless Sensor Networks)의 여러 분야에서 다양한 연구가 이루어져오고 있으며 그 중에 시각 동기화에 관한 연구가 있다. 시각 동기화에 관한 연구는 최근에 새로 시작된 것이 아니라 예전의 인터넷이 개발되었던 초기부터 계속 되어왔으나, 무선 네트워크 환경, 특히 무선 센서 네트워크 분야에서는 새로운 요구사항과 고려 사항이 필요하여 다시 관심을 끌게 되었다.

대표적으로 인터넷과 같은 유선 네트워크에서 동작하는 NTP(Network Time Protocol)[1]가 일찍이 개발되었다. WSN에서의 각 센서 노드는 서로 협업을 하기 위하여 공통 클럭을 유지하는 것이 중요하며, 이를 이용하는 응용에서는 네트워크 전체 혹은 각 센서 노드간에 시각 동기화가 유지되는 것을 전제로 한다. 무선 센서 네트워크를 설계할 때 공통적으로 중요한 사항이 전력 소모를 최소화하는 것이며 시각 동기화 프로토콜의 설계에도 마찬가지로 전력 소모를 최소화하는 것이 필요하다[2].

최근에 WSN에서 동작하는 새로운 시각 동기화 프로토콜인 IBS(Indirect-Broadcast Synchronization)[3]이 개발되었는데, 통신 부하를 줄여 에너지 효율을 높이며, 동기화 정확도와 네트워크 확장성도 우수하다. 이 프로토콜은 멀티홉에서 클러스터 단위로 글로벌 시각을 전달할 수 있다.

계층 구조를 갖는 네트워크가 비계층구조의 네트워크보다 확장성이 우수함이 알려져 있다[4]. 또한 동적인 네트워크에서 확장성, 부하, 견고성, 전력 소모 등의 효율성을 위해 클러스터 구조기반의 라우팅 프로토콜이 개발되어 왔다[5]. 이들 여러 클러스터링 방안들은 서로 다른 특성과 디자인 목표를 가지고 개발되고 있다.

이 논문에서는 클러스터 트리를 부분적으로 재구성함으로써 전체 에너지 소모를 줄이고 네트워크 수명을 연장시키는 기능을 추가한 IBS의 향상된 버전인 EIBS(Enhanced Indirect-Broadcast Synchronization)를 제안한다. 여기서 제안한 클러스터 트리의 부분적인 재구성 기능은 다른 클러스터 재구성 방식과 비교할 때 간단하고, 전력 소모와 네트워크 수명에서 더 우수하다.

논문의 구성은 다음과 같다. 2장에서 기존의 무선센서네트워크용 시각 동기화 프로토콜과 클러스터링 방안을 검토한 후, 부분 클러스터 재구성 기능을 3장에서 제안한다. 4장에서 제안한 부분 클러스터 재구성 기능을 IBS에 적용하여 성능을 분석하며, 마지막으로 5장에서 결론을 다룬다.

II. 관련 연구

지금까지 개발된 여러 시각 동기화 프로토콜들은 송신자와 수신자간의 메시지를 주고받는 형태에 따라 송신-수신형과 수신-수신형으로 나뉜다.

송신-수신형 동기화는 한 노드가 송신자가 되어 데이터를 전송하며, 송신자를 제외한 다른 노드들은 수신자가 되어 보내어진 데이터를 수신하며, TPSN(Timing-sync Protocol for Sensor Networks)[6]과 FTSP(Flooding Time Synchronization Protocol)[7]이 여기에 속하고, 글로벌 참조 클럭을 전체 네트워크에 전달할 수 있다. 이와 대조적으로 수신-수신형 동기화에서는 참조 노드가 특정한 패킷을 한 번 전송하면, 이 후에는 전송한 노드를 제외한 다른 노드들이 서로 협력하여 동기화를 이루는 방식을 갖는다. 이 종류에 해당하는 대표적인 프로토콜이 RBS(Reference Broadcast Synchronization) [8]로서, 한 노드(송신자)가 다른 수신 노드들이 동시에 수신하도록 비이컨 메시지를 방송하면, 송신자를 제외한 다른 수신자들이 이 비이컨 메시지를 수신한 각각의 로컬 클럭 값을 서로 교환하여 상호간의 클럭 오프셋을 기록한다. 이 RBS 프로토콜은 가까운 거리에 위치한 여러 노드간에 상대적인 클럭 오프셋을 비교적 정확하게 파악할 수 있지만, 특정 노드의 클럭(혹은 글로벌 클럭)으로 네트워크 전체를 동기화할 수는 없다.

지금까지 개발된 이러한 여러 시각 동기화 프로토콜들은 노드간에 특별한 구조를 갖지 않는 FTSP와 같은 형태가 있는가 하면, 대부분은 네트워크에서 트리나 계층 클러스터와 같은 구조를 가지고 동작한다[2]. 전형적으로 트리 구조를 사용하는 방식으로는 TPSN과 RBS가 있다. 또한 이들 이후에 개발되고 있는 시각 동기화 프로토콜들은 클러스터나 클러스터 트리 구조를 사용하여 에너지 효율을 높이기 위한 노력을 계속하고 있다. IBS[3]와 TSRT(Tree Structured Referencing Time Synchronization)[9], Adaptive Time Synchronization for Homogeneous WSNs[10], Clustered Time Synchronization Algorithm for Wireless Sensor Networks[11] 등은 클러스터 구조를 기본으로 동작한다.

이 논문에서는 클러스터 트리를 부분적으로 재구성함으로써 전체 에너지 소모를 줄이고 네트워크 수명을 연장시키는 향상된 IBS를 개발하고자 한다. 다른 트리 구성 접근 방식과 비교할 때, 이 논문에서 사용된 방법은 알고리즘이 간단할 뿐만 아니라 전력 소모나 네트워크 수명 면에서 더 효율적이다.

III. EIBS 알고리즘

3.1 IBS의 네트워크 모델

이 논문에서 제안하는 클러스터 재구성 알고리즘은 IBS에 적용이 되었으므로, 우선 IBS의 기본 동작에 대하여 알아본다.

IBS 프로토콜(3)은 TPSN과 유사하게 클러스터 트리 구조에서 동작하며, 크게 두 개의 과정, 즉, 준비 과정과 동기화 과정으로 이루어진다. 준비 과정에서는 클러스터링 알고리즘을 사용하여 클러스터 트리를 생성하며, 그 다음 동기화 과정에서 생성된 클러스터 트리에서 각 노드들을 동기화시킨다. 클러스터를 생성하는 알고리즘은 어떤 것이라도 사용하여 클러스터를 만들 수 있으며, IBS에서는 TPSN에서 사용된 방법과 같은 간단한 클러스터링 알고리즘을 적용하였다. 그림 1은 IBS에서 사용하는 클러스터 트리의 예를 나타내고 있다.

동기화 과정에서는 생성된 클러스터 트리에서 루트 노드에서 시작하여 하위 레벨 노드로 순서대로 동기화를 실시하여 전체 네트워크를 루트 노드의 글로벌 클럭으로 동기화시킨다.

IBS에서는 노드의 역할을 루트, 보조, 리더, 일반 노드로 구분하며, 간단히 노드의 형태는 리더(LN : Leader node)와 일반 노드(NN : Normal node)로 구분한다.

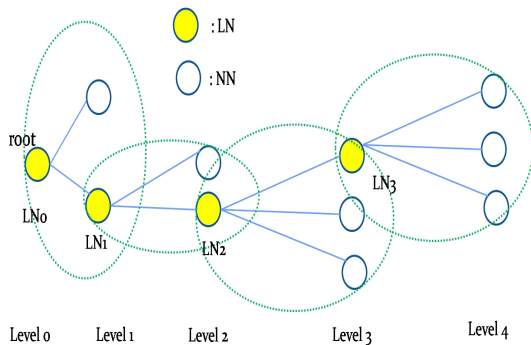


그림 1 IBS에서 사용하는 클러스터 트리의 예
Fig. 1 Cluster tree used in IBS

LN은 클러스터 내의 일반 노드를 동기화하는 기능을 하며, NN은 LN가 주도하는 클러스터의 구성원으로서 LN보다 상대적으로 간단한 동기화 동작을 한다. 그러므로 LN은 NN보다 더 많은 전력을 소모하며, 반면에 NN은 LN보다 전력 소모가 적어 전체 네트워크의 수명은 LN에 의해서 결정이 된다.

그림 2는 IBS의 동기화 동작을 나타내고 있다. 각 레벨에서 LN은 보조 역할을 하는 상위의 LN과 협력하여 글로벌 클럭 정보를 얻은 다음 클러스터 내부의 일반 노드에 이 정보를 제공하여 동기화를 시킨다.

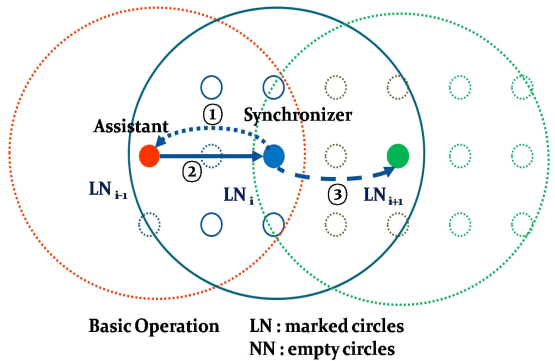


그림 2 IBS의 동기화 동작에서 LN의 역할
Fig. 2 LN's role in basic operation of IBS.

3.2 EIBS의 네트워크 모델

EIBS에서는 IBS의 동작 단계에 한 단계가 더 추가된다. 그림 3에 나타낸 것처럼, 트리 구성 단계와 동기화 단계는 IBS의 준비 과정과 동기화 과정과 거의 동일하다. 여기에 추가로 부분 재구성 단계가 추가되었는데 각 단계의 동작을 설명하면 다음과 같다. 처음 트리 구성 단계에서 트리 생성 알고리즘을 사용해서 트리를 만들어 동기화 단계에서 사용을 한다(그림에서 ①로 표시). 시간이 경과함에 따라 일부 노드가 사망하여 트리 유지가 어려운 경우(②), 부분 재구성 단계에서 트리의 일부를 새로운 LN로 재구성하며(③), 이 과정이 비효율적이거나 불가능하면 처음 단계로 돌아가서 전체 트리를 재구성(④)한다.

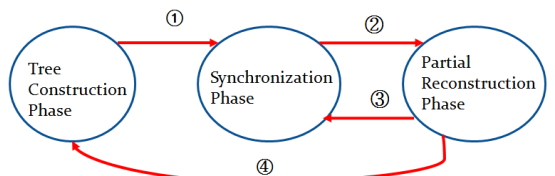


그림 3 EIBS에서 동작 단계
Fig. 3 Operation Phases of EIBS

3.3 클러스터 트리 재구성 알고리즘

LN의 에너지 잔량이 충분하지 않아 에너지 홀이 생길 수 있으므로 일정한 수준의 전력이 남아있을 때 다른 노드로 LN을 교체함으로써 전체 클러스터 트리를 유지하게 된다.

클러스터 트리를 G라고 하고, 각 노드의 집합을 V, 두 노드간의 연결 링크를 E라고 할 때, $G = (V, E)$ 로 나타내기로 하자.

노드 i의 이웃 노드 집합 N는 노드 i로부터 통신 반경 내에 있어 데이터 수신이 가능한 노드들의 집합으로 다음과 같이 정의할 수 있다.

$$N(i) = \{ j \in V : i \neq j \text{ and } (i,j) \in E \} \quad (1)$$

IBS의 트리에서 LN 중 하나인 LN_i의 이웃 노드 중에는 LN_i의 부모 노드(LN_{i-1})와 자식 노드(LN_{i+1})가 포함되어 있다. 즉,

$$N(LN_i) = \{ LN_{i-1}, LN_{i+1}, j, k, \dots \} \quad (2)$$

LN_i의 이웃노드 k의 이웃 노드 집합이 LN_i의 부모 노드(LN_{i-1})와 자식 노드(LN_{i+1})를 포함한다면, 노드 k가 LN_i를 대체하더라도 전체 트리의 근간은 변하지 않는다. 즉,

$$N(k) = \{ LN_{i-1}, LN_{i+1}, m, n, \dots \} \quad (3)$$

그림 4에서 LN₂의 통신 반경 내에 있는 노드 중 노드 k의 통신 반경에는 LN₁과 LN₃이 포함된다. 따라서 노드 k로 LN이 변경되더라도 LN₂와 노드 k의 통신 반경 밖에 있는 NN들은 트리 구성에서 변화되지 않는다.

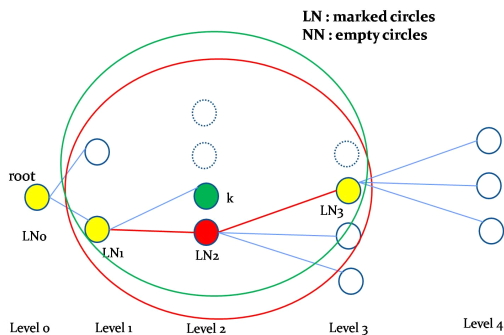


그림 4 클러스터 트리 재구성의 개념
Fig. 4 Concept of Cluster Tree Reconstruction

3.3.1 LN 교체

LN을 교체하는 과정은 다음과 같다.

클러스터 트리에서 LN는 에너지 잔량이 문턱값 이하로 떨어지면, 주위에 있는 에너지 잔량이 충분한 다른 NN중에서 새로운 LN을 찾아 교체한다. 이 때 새 LN이 될 후보 노드는 다음의 조건을 만족하여야 한다.

- ① 현재의 LN보다 에너지 잔량이 충분하여 교체 후의 수명이 교체전보다 더 길어야 한다.
- ② 후보 노드는 특정한 영역 내에 위치하여야 하며, 그림 5에 나타난 것처럼 현재 LN의 부모 노드와 자식 노드의 중첩된 영역 내에 존재해야 한다. 그 이유는 LN의 교체 후에도 현재의 클러스터를 제외한 다른 나머지 클러스터 트리의 구조는 변하지 않아야 하기 때문이다.

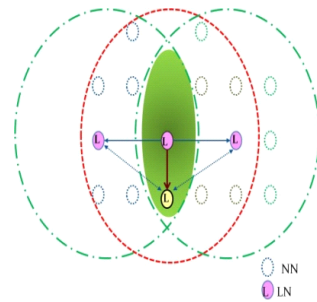


그림 5 새 후보 LN의 위치한 영역
Fig. 5. Area for new LN candidates.

새로운 LN을 찾아 교체하는 과정을 순서대로 설명하면 다음과 같다.

- (a) 현재 LN은 새로운 LN을 찾기 위하여 자신과 부모, 자식의 ID와 함께 "HELP" 메시지를 발송한다(그림 6-(a) step-a 참조).
- (b) 현재 LN의 부모와 자식 노드는 현재 LN으로부터 "HELP" 메시지를 받고 이에 대한 응답으로 "AGREE" 메시지를 가족의 ID와 함께 보낸다. 가족 ID는 가족 관계를 나타내기 위한 것으로, 현재 LN의 부모 노드는 자신과 자식의 ID를, 현재 LN의 자식 노드는 부모와 자신의 ID를 순서대로 보냄으로써, 현재 LN에서 자신과의 관계를 알려준다. "AGREE" 메시지는 일반노드들이 자신이 새로운 LN 후보가 되는 지 여부를 파악하는데 사용된다. 다시 말하면, 현재 LN와 현재 LN의 부모 및 자식 LN으로부터의 메시지를 모두 받을 수 있다면, 자신이 다음 LN의 후보가 될 자격이 있음을 알 수 있다(그림 6-(b))

- step-b참조).
- (c) 다음 LN 후보 노드들은 자신의 남은 에너지가 현재 LN보다 많은 지 확인한다. 만일 남은 에너지가 충분하다면 현 LN에서 "REQUEST"를 보내어 새로운 LN이 되기를 요청한다(그림 6-(c) step-c 참조).
 - (d) 현재 LN은 일정한 기간 동안 후보 노드로부터 요청을 받은 다음, 에너지가 가장 풍부한 노드를 새로운 LN으로 결정한다. 새로운 LN이 결정되었음을 알리기 위해 "NOMINATE" 메시지를 발송한다(그림 6-(d) step-d 참조).
 - (e) 새로운 LN은 일정한 시간 후부터 새로운 LN으로서 임기를 시작함을 알리기 위해 "INAUGURATE" 메시지를 발송한다(그림 7-(a) step-e 참조).
 - (f) 새로 선출된 LN을 제외한 모든 일반노드는 새로운 LN과의 관계에 대한 정보를 갱신한다. 여기에는 4가지의 경우가 발생한다.

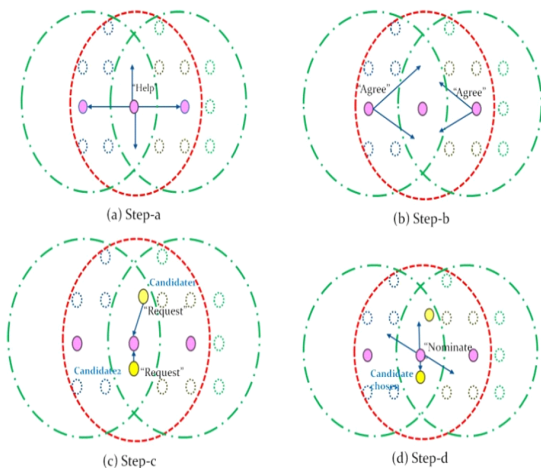


그림 6 새 LN를 선택하는 절차
Fig. 6 Procedure for new LN selection.

① 새로 선출된 LN으로의 변경

NN이 과거의 LN과 현재의 LN으로부터 모두 메시지를 수신할 수 있다면, 현재 노드의 LN 정보를 현재의 새로 선출된 LN의 ID로 변경한다(그림 7-(b)의 왼쪽 그림 참조).

② 다른 클러스터로 이동

NN이 과거의 LN으로부터 새로운 LN을 선출했다는 메시지를 받은 후, 일정 시간이 지나도 새로운 LN으로부터 "INAUGURATE"를 받지 못했다면, 새로운 LN의 통신 영역을 벗어난 것이므로 다른 LN의 클러스터로 재가입하는 절차를 밟는다.

를 밟는다.

②-1 만일 NN이 현재의 LN으로부터 "INAUGURATE" 메시지를 받지 못했지만, 전에 과거의 LN의 부모 노드로부터 "AGREE" 메시지를 받은 적이 있다면, 이 노드를 자신의 새로운 LN으로 등록한다. 그림 7-(b)의 오른쪽 그림에서 예를 들면, 파랗게 칠해진 그림에 속한 노드 번호 1과 노드 번호 2가 여기에 속한다.

②-2 만일 NN이 현재의 LN으로부터 "INAUGURATE" 메시지를 받지 못했지만, 전에 과거의 LN의 자식 노드로부터 "AGREE" 메시지를 받은 적이 있다면, 이 노드를 자신의 새로운 LN으로 등록한다. 그림 7-(b)의 오른쪽 그림에서 초록색으로 칠해진 그림에 속한 노드 번호 4와 5가 여기에 속한다.

③ NN이 과거 LN의 부모나 자식 노드 어떤 것으로부터도 "AGREE" 메시지를 받지 못했다면, 다른 LN이 관할하는 클러스터에 가입하여야 한다. 예를 들면, 그림 7-(b)의 오른쪽 그림에서 빨강색 칠한 부분의 노드 3이 여기에 해당한다.

그림 8에 제안 알고리즘을 의사 코드로 나타내었다.

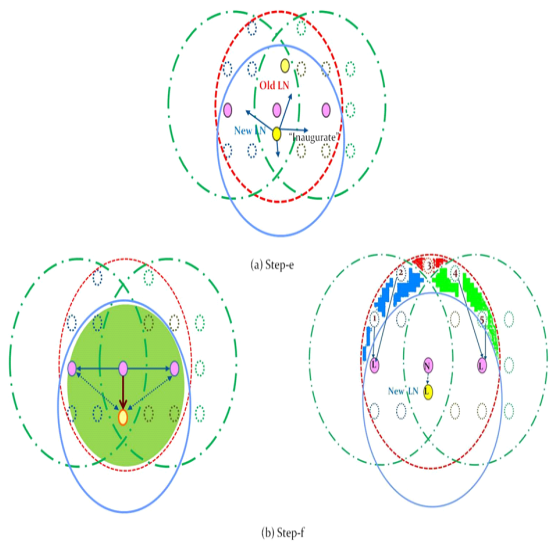


그림 7 새로운 LN에 대한 일반 노드의 재가입
Fig. 7 Rejoining of NNs according to the new LN.

3.3.2 제안 알고리즘의 메시지 복잡도 분석

EIBS에서 사용된 단순 클러스터링 알고리즘은 TPSN에서 사용된 방법과 동일하게 루트 노드로부터 패킷을 플러딩시켜 각 노드당 1회의 방송을 실행함으로써 트리가 생성된다. 제안한 알고리즘에서는 이미 생성된 클러스터 트리에서 특정한 LN을 다른 일반 노드로 교체함으로써 부분적으로 클러스터를 변경하지만 전체 네트워크의 트리 구성은 변화시키지 않는다. 이런 동

작에 필요한 메시지는 최소 6개이다. 즉, "HELLO", 2개의 "AGREE", 몇 개의 "REQUEST", "NOMINATE", "INAUGURATE" 등의 메시지가 소요된다.

부분적인 트리 재구성에서 k개의 LN이 교체된다면 전체적으로 약 (6 * k)개의 메시지 전송이 필요하다. 전체 노드 수를 N이라고 할 때, 클러스터 트리 전체를 다시 생성하는데 N개의 메시지 전송이 필요하므로, 부분 트리 재구성이 전체 트리 재생성보다 효율적이기 위해서는 다음의 조건이 만족되어야 한다. 즉,

$$k < N/6 \tag{4}$$

L을 LN의 수, LNR(리더 노드 비율 : Leader-Node Ratio)을 전체 노드 수에 대한 L의 비율이라고 정의하면,

$$\begin{aligned} LNR &= L / N, \\ 0 < LNR < 1 \end{aligned} \tag{5}$$

의 값을 갖게 된다.

식(5)를 사용하여 식(4)을 다시 작성하면,

$$k < L / (LNR * 6) \tag{6}$$

이 된다.

이 식의 의미는 LNR=0.15 (즉, 15%)이하이면 식 (4)를 만족하므로 부분적인 트리 재구성이 전체 트리 구성보다 유리하다는 것이다. 이와 대조적으로 만일, LNR=0.2 (즉, 20%)이상이면 식 (4)를 만족하지 않으므로 부분적인 트리 재구성보다는 전체 트리를 재구성하는 것이 유리하다는 의미이다.

대표적인 클러스터링 알고리즘인 LEACH[12]에서 클러스터링에 필요한 메시지 수를 분석해보면 다음과 같다. 각 노드는 미리 정해진 확률에 의해 클러스터 헤드 또는 클러스터 멤버로 동작할 지 역할을 결정한 후, 클러스터 헤드로 결정된 노드는 이웃 노드들에게 광고(Advertisement) 메시지를 발송한다. 클러스터 멤버 노드는 클러스터 헤드로부터 광고 메시지를 받으면 그 신호의 세기에 따라 하나의 클러스터를 결정하고 클러스터 헤드에게 참가(Joint-request) 메시지를 보낸다. 최종적으로 클러스터 헤드는 멤버들의 요청 메시지를 수합하여 멤버들을 결정하고 TDMA 스케줄을 작성하여 모든 멤버에게 발송한다. 클러스터 헤드의 수를 CH, 전체 노드 수 N이라고 하자. LEACH에서 라운드 당 클러스터링에

```

Reconstruction Algorithm
Input : "HELP" packet from LN
Output : LN change and rejoin
.....
If( parent or child LN of current LN) // LN
  send "AGREE" packet
else // normal nodes
  wait "AGREE" packets from both
  If(receive both "AGREE"s) // LN candidate
    participate at election or not
    after considering residual energy.
    If(selected as new LN by current LN)
      notify new LN chosen (send "INAUGURATE ")
    else // not a new LN
      Determine its LN and Join
      Switch(condition) {
        case 1 : No change ; break;
        case 2 : Change to new elected LN ; break;
        case 3 : Change to the parent of
                  the previous LN; break;
        case 4 : Change to the child
                  of the previous LN; break;
        case 5 : Find new LN and Join; break;
      }
  
```

그림 8 제안 알고리즘의 의사 코드
Fig. 8 Pseudo code of the proposed algorithm.

필요한 메시지 복잡도는

$$M_{LEACH} = (2 * CH) + (N-CH) \tag{7}$$

IBS와 비교하기 위해, CH=L, 전체 노드 수에 대한 클러스터 헤드의 비율을 LNR= CH / N 이라고 하자.

식 (4)를 다시 작성하면,

$$\begin{aligned} M_{LEACH} &= 2 * LNR * N + N (1 - LNR) \\ &= N (1 + LNR) > N \end{aligned} \tag{8}$$

가 되어, 메시지 복잡도는 N보다 크다.

IBS에서는 처음 준비 단계에서 트리를 한 번만 구성하지만, LEACH에서는 일정 시간마다 식 (7)과 같은 라운드가 반복되어 이루어져 메시지 복잡도는 라운드 수에 비례하여 더 커진다.

IV. 실험

알고리즘의 동작을 검증하기 위하여 NS2(Network Simulator)-2.35[13]을 사용하여 몇 가지 시나리오의 네트

워크 토폴로지에서(예, 3행 * 6열 형태) IBS의 동작을 실시하였다. 공통되는 시뮬레이션 환경은 노드간 거리 100m, 전송 영역 반경 550m이고, 노드에서의 전력 변수로는 송신전력(0.660W), 수신전력(0.395W), 대기전력(0.035W)으로 설정하였다. 이들 전력값에 따라 실제 수명 시간이 변할 수 있으나, 제안 알고리즘 적용 전후의 수명을 비교하는 데에는 어떤 값으로 실험하여도 상관이 없다.

4.1 노드별 전력소모

IBS에서는 LN이 중심이 되어 동작하므로 NN보다 더 많은 전력을 소모한다. 각 노드형태에 따른 전력 소모를 살펴보기 위하여 그림 9와 같은 토폴로지에서 IBS를 여러 라운드동안 동작시켰다. 그림에서 노드 1은 루트 노드이고, 노드 4와 노드 10은 LN인 3홉 네트워크이다. 노드 0은 IBS에서 루트노드의 Assist Node로 동작하며, 나머지 노드는 모두 NN이다.

초기의 각 노드의 에너지는 1 (J)로 설정하여, 각 라운드를 진행하면서 남은 에너지량을 측정한 결과는 그림 10에 나타내었다.

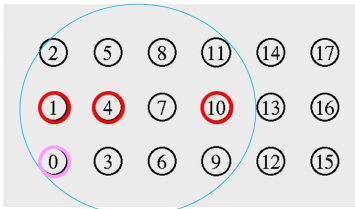
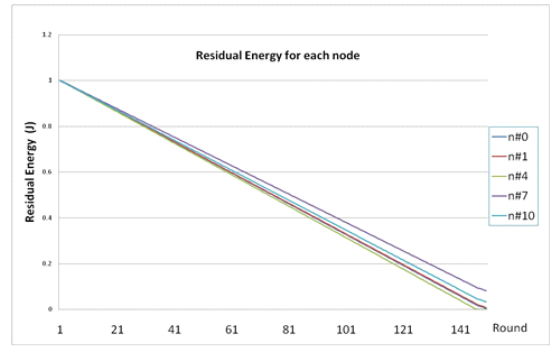


그림 9 노드별 전력 소모 테스트
Fig. 9 Power consumption tests.

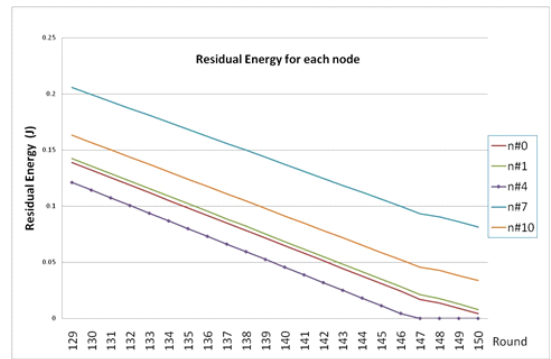
실험 결과에서는 LN중 하나인 노드 4에서의 전력 소모가 가장 많고, NN인 노드 17에서 전력 소모가 가장 적었다. LN들 사이에서 전력 소모 차이가 난 것은 노드의 위치에 따라 이웃 노드의 수가 다르기 때문이다. 즉, 노드 4의 통신 반경내에는 LN인 노드1과 노드 10 그리고 Assist Node인 노드 0이 모두 포함되어 있어 송신전력뿐 아니라 수신 전력이 가장 많이 소모된다.

그림 10의 (b)는 각 노드별로 사망 시점을 정확히 측정하기 위해 그림 10의 (a)에서 129~150라운드 부분을 확대한 것이다. 노드 4가 147라운드에서 가장 먼저 에너지가 소진되었고, 다음 노드 0이 150 라운드, 노드 1이 151 라운드만에 모든 에너지가 소진되었다. 노드 4가 사망하면, 전체 네트워크에 대한 동기화 동작은 이루어지지 않는다. 이는 루트 노드에서 동기화를 시작하더라도 다음 LN노드인 노드 4와 노

드 10을 통해서 연속적으로 전달이 되지 않기 때문이다. 그림에서 보듯이 노드 1과 노드 0이 사망하고 나면 다음 LN인 노드 10을 포함하여 다른 NN에서는 데이터 송수신이 이루어지지 않으므로 대기 전력을 제외한 더 이상의 전력 소모는 없다.



(a)



(b)

그림 10 각 노드에 남은 에너지 량
Fig. 10 Residual energy at each node.

그림 11은 100라운드에서 노드에서 노드5로 LN를 변경한 후 동기화 라운드를 계속하며, LN 변경전과 후의 각 노드의 수명을 비교, 측정한 것이다. 그림에서 n#0, n#1, n#4, n#7, n#10은 LN의 변경없이 끝까지 실험한 값이고, n#0', n#1', n#4', n#7', n#10'은 100라운드에서 LN을 변경한 후 각 노드의 잔여 에너지를 측정하여 비교한 것이다.

노드 4가 LN으로 사용되었을 때의 사망시점이 147라운드 이었던 것과 비교할 때, LN 변경 후에는 전력 소모가 줄어들어 150라운드에서도 0.011431(J)의 잔여 에너지를 가지고 있다. 반면에 노드 7은 처음에 NN로 동작하였으므로 노드 4보다 전력 소모가 별로 없다가, LN으로 변경된 후에는

이전보다 전력 소모가 증가하였으나 150라운드에서도 0.045605(J)로 다른 노드보다 충분히 남아있다. 그리고 LN의 변경 후에는 노드 4보다 노드 0이 먼저 에너지가 고갈되어 150라운드 후에 사망하였다. 노드 0은 IBS에서 Assist Node로서 루트 노드와 첫 번째 LN에 글로벌 클럭을 제공하는 역할을 하며 LN와 대등한 전력을 소모한다. 최초 노드의 사망을 전체 네트워크의 수명시간으로 정의한다면, LN 변경 전의 수명이 147라운드이고, LN 변경 후에는 전체 네트워크의 수명이 150으로 연장되었다.

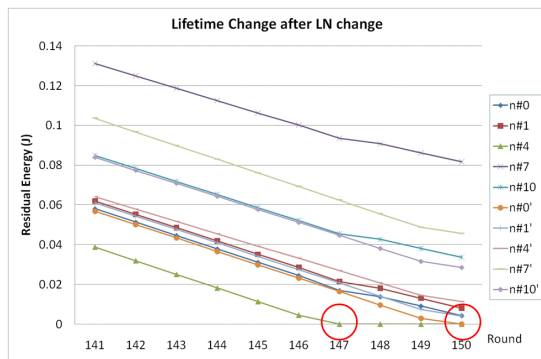


그림 11 각 노드의 사망 시각
Fig. 11 Dead time for each node.

VI. 결 론

WSN용 시각 동기화 프로토콜인 IBS가 2012년에 이미 개발되었다. 이 프로토콜은 클러스터 트리 구조에서 동작하므로 전력 소모가 일반노드보다 리더노드에서 많이 발생하여 네트워크 수명은 리더노드에 의해 결정된다. 이 논문에서는 클러스터 트리의 일부만을 재구성하여 전력 소모를 줄이고 네트워크 수명을 연장시킴으로써 IBS의 동작을 개선한 EIBS를 제안하였다. 전체 클러스터 트리를 재구성하는 방안과 비교할 때, 부분 클러스터 트리 재구성 방안은 간단할 뿐 아니라 전체 소모 전력과 네트워크 수명 연장 관점에서 더 효율적이다.

제안 방안을 기존에 제안되었던 IBS 프로토콜에만 적용을 하였으나, IBS 뿐 아니라 WSN에서 동작하는 클러스터 트리 기반의 다른 응용에서도 적용이 가능할 것으로 본다. 향후에는 이 알고리즘을 라우팅이나 데이터 수집(Aggregation) 프로토콜에 적용할 수 있도록 확장할 계획이다.

참고문헌

- [1] D. Mill, *Computer Network Time Synchronization: the Network Time Protocol on Earth and in Space*, Second Edition, CRC Press, 2011.
- [2] S. Bae, "A Survey on Time Synchronization Protocols for Wireless Sensor Networks", *Journal of The Korea Society of Computer and Information*, Vol. 19, No.6, June 2014.
- [3] S. Bae, "Time Synchronization by Tree-based Indirect-Broadcasting for Wireless Sensor Networks", *Journal of Korean Institute of Information Scientists and Engineers*, Vol. 39, No. 6, Dec. 2012.
- [4] P. Sharma and P. Mandal, "Reconstruction of Aggregation Tree in spite of Faulty Nodes in Wireless Sensor Networks", *6th IEEE International Conference on Wireless Communication and Sensor Networks (WCSN'10)*, Allahabad, India, Dec. 2010.
- [5] X. Liu, "A Survey on Clustering Routing Protocols in Wireless Sensor Networks", *Sensors*, 2012.
- [6] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-Synch Protocol for Sensor Networks, in *Proceeding of ACM Sensys*, USA, 2003.
- [7] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol", in *Proceeding of ACM SenSys'04*, 2004.
- [8] J Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts", in *Proceeding of ACM OSDI*, 2002.
- [9] S. Rahamatkar and A. Agarwal, "An Approach towards Lightweight, Reference Based, Tree Structured Time Synchronization in WSN", in *Proceeding of CCSIT 2011*, pp. 189-198, 2011.
- [10] S. Chauhan and L. K. Awasthi, "Adaptive Time Synchronization for Homogeneous WSNs", *International Journal of Radio Frequency Identification and Wireless Sensor Networks*, 2011.

- [11] A. Zurani and B. N. Mahajan, "Clustered Time Synchronization Algorithm for Wireless Sensor Networks", *International Journal of Recent Technology and Engineering (IJRTE)*, Vol.1, Jun 2012.
- [12] W. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks", *IEEE Trans. on Wireless Communications*, Vol. 1, No. 4, Oct. 2002.
- [13] "The Network Simulator ns-2: Documentation", <http://www.isi.edu/nsnam/ns/ns-documentation.html>, Nov. 2011.

저 자 소 개



배 시 규

1986: 경북대학교 전자공학과 공학사.

1998: 경북대학교

컴퓨터공학과 공학박사.

2011: 미국 Iowa주립대 방문교수

현 재: 동양대학교

컴퓨터정보전학과 교수

관심분야: 컴퓨터네트워크

Email : skbae@dyu.ac.kr