

무선 센서 네트워크를 위한 효율적인 클러스터링 기법

(An Efficient Clustering Mechanism for WSN)

이진우*, 모하매드 바니아타*, 홍지만**

(Jinwoo Lee, Baniata Mohammad, Jiman Hong)

요약

무선 센서 네트워크에서는 원격의 열악한 환경에 센서 노드들이 배치된다. 이러한 네트워크에서 센서 노드의 전원이 모두 소모되면, 사람의 생명을 구할 수도 있는 센서 네트워크의 품질 및 성능 저하와 함께 센서 노드들은 쓸모가 없게 된다. 따라서 에너지 소비 개선 및 센서 네트워크의 수명 연장에 관련된 수 많은 클러스터링 프로토콜들이 제안되었으나 대부분의 기존 연구들의 기법들은 클러스터 헤드의 오버헤드 상당히 크다. 센서 노드의 전력 제한과 클러스터 헤드의 오버헤드 문제 때문에 각 노드의 에너지 소비를 최소화하며 네트워크 수명을 최대화하는 라우팅 프로토콜을 설계하는 것은 중요하다. 따라서 본 논문에서는 클러스터 헤드의 부담을 줄여주며 에너지 소비를 최소화하는 라우팅 알고리즘과 네트워크 수명을 최대화할 수 있는 알고리즘을 적용한 효율적인 클러스터링 기법을 제안한다. 시뮬레이션 결과는 제안한 클러스터링 기법이 에너지 균형을 향상시켰으며, 유사한 역할을 하는 기법들과 비교하였을 때 네트워크 수명이 연장됨을 보여준다.

■ 중심어 : 라우팅 프로토콜, 네트워크 수명, 클러스터링

Abstract

In wireless sensor networks, sensor nodes are deployed in a remote, harsh environment. When the power of the sensor node is consumed in such a network, the sensor nodes become useless together with the deterioration of the quality and performance of the sensor network which may save human life. Although many clustering protocols have been proposed to improve the energy consumption and extend the life of the sensor network, most of the previous studies have shown that the overhead of the cluster head is quite large. It is important to design a routing protocol that minimizes the energy consumption of each node and maximizes the network lifetime because of the power limitations of the sensor nodes and the overhead of the cluster heads. Therefore, in this paper, we propose an efficient clustering scheme that reduces the burden of cluster heads, minimizes energy consumption, and uses algorithms that maximize network lifetime. Simulation results show that the proposed clustering scheme improves the energy balance and prolongs the network life when compared with similar techniques.

■ keywords : Routing Protocol, Network lifetime, Clustering

I. 서론

최근 전자 및 무선 통신 기술 분야의 발전으로 인해 비용이 적게 들고 저전력이며 여러 기능을 가진 센서 노드를 기반으로 한 대규모의 무선 센서 네트워크가 구축되고 있다 [1]. 이러한 가능성을 고려하여 무선 센서 네트워크는 여러 분야에서 응용되어 사용되고 있는 추세이다 [2-6].

센서 노드들은 그것들끼리 네트워크를 구축하고, 환경 감지, 모니터링 및 센서에서 수집한 데이터를 저장하고 가공하는 일

을 협동해서 수행한다. 그러나, 센서들은 주로 용량의 제한이 있는 배터리로써 에너지를 사용하며 그것들이 환경에 배치된 이후에는 다시 배터리를 교체하기 어렵다. 또한, 센서 노드의 수명은 무선 센서 네트워크의 수명과 직결되므로, 각 노드의 에너지 자원은 센서의 수명이 최대가 되도록 관리되어야 한다. 그러므로, 무선 센서 네트워크에서 토폴로지를 구성할 때 에너지 효율은 우선적으로 고려해야 하는 사항이다 [7]. 낮은 에너지 소비량과 네트워크 상의 동일한 부하 분산을 위해서 센서 노드들은 네트워크를 그룹으로 나눈 개체인 클러스터를 통하여 계층적으로 구성 될 수 있다 [8]. 각 클러스터는 클러스터 헤드와 일반

* 학생회원, 숭실대학교 컴퓨터학과

** 정회원, 숭실대학교 컴퓨터학부

이 논문은 2016년도 정부(산업통상자원부)의 재원으로 한국산업기술평가관리원-산업기술혁신사업의 지원을 받아 수행된 연구임(No. 10063496)

접수일자 : 2017년 11월 07일

게재확정일 : 2017년 12월 31일

수정일자 : 2017년 12월 28일

교신저자 : 홍지만 e-mail : jiman@ssu.ac.kr

노드로 구성되며, 클러스터 내부 및 다른 클러스터 간의 통신은 단일 홉과 다중 홉 통신으로 통합 될 수 있다.

최근에도 LEACH[9] 나 EEUC[10] 와 같은 네트워크 수명 및 에너지 소모를 개선하기 위한 많은 프로토콜들이 제안되고 있으나, 이전에 제안되었던 프로토콜과 동일하게 여전히 에너지 소모와 에너지 홀 문제들이 남아있는 상황이다. 이러한 문제점들을 극복하기 위해서 본 연구에서는 센싱되는 장소에서 발생하는 데이터를 수집하는 방법에 대한 효율적인 클러스터링 기법을 제안하고 실험하였다. 제안한 기법은 네트워크를 클러스터로 분해하고 클러스터 헤드를 통해서 차선의 다중 홉 경로를 찾고 기지국으로 수집된 데이터를 전송한다. 여기서 선택되는 센서 노드는 남아있는 노드들 중 잔여 에너지가 높으며 기지국까지의 거리가 가장 가까운 노드가 된다. 그러므로, 기지국까지의 거리가 가까워질수록 노드의 반경이 줄어들기 때문에 클러스터 크기가 감소하게 되며 결국 클러스터에서 수행하는 데이터 처리 및 클러스터 헤드에 대한 에너지 소비가 감소하게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 소개하고, 3장에서는 문제에 대한 정의를 한다. 4장에서는 제안한 기법을 설명하고 5장에서는 실험 결과를 보인 다음 6장에서 결론을 맺는다.

II. 관련 연구

무선 센서 네트워크의 에너지 소비를 최소화하기 위해 많은 클러스터링 프로토콜이 제안되었다[9-10].

LEACH는 무선 센서 네트워크를 위한 클러스터링 프로토콜로 알려져 있다. LEACH는 중앙 집중식 제어가 아닌 분산형 클러스터를 구성하며, 각 클러스터의 헤드가 모든 노드의 데이터를 수집하여 글로벌 통신을 줄임으로써 에너지를 균등하고 적절하게 분배한다[9].

Ever는 EEUC[10] 기법에서 사용 된 방정식에 기반하여 HEED 프로토콜의 완료 반경을 조정하였다. 그것을 통해 에너지 소비를 줄이면서 기지국 근처의 클러스터 헤드가 빨리 죽는 것을 피하게 하는 UHEED라는 불균등 클러스터링 알고리즘을 제안하였다[11].

Ren은 에너지 소비 균형과 네트워크 수명을 늘리는 LEACH 기법을 기반으로 한 불균등 클러스터링 기법을 제안하였다[12]. 이 기법에서 클러스터 헤드 선택을 위해 사용되는 요소는 에너지 비율과 경쟁 거리이며 각각의 노드들은 기지국으로부터 떨어진 거리를 통해서 송신 전력을 조정할 수 있다.

Yang은 기지국에서 가까운 노드들이 클러스터 헤드로 선택될 확률이 높은 Energy-Balancing Unequal Clustering Protocol(EB-UCP)라는 프로토콜을 제안하였다[13]. 이 프로토콜에서는 네트워크가 층으로 나뉘어지며, 각 층에 존재하는 노드

들은 각각 클러스터 헤드로 선택될 확률을 지니고 있다.

III. 문제 정의

1. 클러스터 헤드 에너지 소비 문제

대규모 네트워크에서 네트워크 크기와 클러스터 크기 간의 관계는 네트워크 크기가 커질수록 클러스터 노드 수가 증가하는 만큼 비례하는 관계에 있다. 이것은 클러스터 헤드가 많은 부담을 한다는 것과 동시에 상당한 양의 에너지를 소비한다는 것을 뜻하는 것과 동일하다. 예를 들어, 클러스터 내부에서 단일 홉 모델을 사용하면 각 노드가 해당 클러스터 헤드에 직접 데이터를 전송하므로 클러스터 내의 노드에서 클러스터 헤드로 전송되는 메시지 수가 증가하게 되는데, 이 상황에서 혼잡과 충돌이 발생하게 되면 데이터 재전송으로 인해서 더 많은 에너지를 소비하게 될 수 있다.

수집된 데이터를 기지국까지 전달하기 위해서 여러 클러스터 헤드 간의 다중 홉 통신 모델을 사용하는 것은 기지국과 직접 통신 할 수 없는 클러스터 헤드에 적합한 방법이 될 것 이다. 그러나, 이 방법을 사용하는 것 또한 클러스터 헤드가 많은 에너지를 소비할 수 있다. 두 가지 상황을 고려해 볼 수 있는데, 하나의 상황으로는 클러스터 헤드가 기지국에 가장 가까운 상황이라면 이 클러스터 헤드는 다른 노드의 라우터 역할을 하게 되며 많은 양의 트래픽이 발생하므로 그만큼의 에너지 소비가 비례해서 발생할 것이다. 다른 상황으로는 클러스터 헤드의 잔여 에너지와 보내는 노드의 잔여 에너지를 기반으로 클러스터 헤드와 기지국 간의 최적 경로가 찾아진 상황인데, 이 경로를 통해서 모든 통신이 발생하게되므로 결국 이 경로에 있는 클러스터 헤드의 에너지가 통신이 발생하는 만큼 소모 될 것이다. 결국 근본적인 원인을 해결하기 위해서는 핫 스팟 문제를 극복해야한다.

2. 센서 네트워크

무선 센서 네트워크를 수백 혹은 수천개의 에너지 제한이 있고, 감지 환경에 균일하게 배치된 N개의 센서 노드들이며 센서 노드 집합은 $S = \{s_1, s_2, s_3, \dots, s_n\}$, $|S| = N$ 로 가정한다. 또한, 이 연구에서의 센서 네트워크는 다음과 같은 가정을 기반으로 한다.

- 기지국의 위치는 고정되어있고, 센서 노드로부터 먼 거리에 있다.
- 네트워크 상의 노드는 동질적이며, 노드 간의 링크는 대칭적이다.
- 노드들은 데이터를 전송 할 때 전력 제어를 할 수 있다.
- 노드의 에너지는 제한되어있으며, 이동이 불가능하다.

- 네트워크는 각 노드의 위치 인식이 가능하다.

또한, 라디오 모델은 Heinzelman의 모델과 동일한 것을 사용하였다. 이 모델에서 라디오는 송신기 및 송신기 증폭기, 수신기 회로를 작동시키기 위하여 에너지를 분산시킨다. 자유 공간 모델(d^2 만큼의 전력 손실)과 다중 경로 페이딩 채널 모델(d^4 만큼의 전력 손실)이 사용되며 송신기 증폭기를 구동시키는데 소비되는 에너지는 송신기와 수신기 간의 거리에 따라 달라진다. 이 모델에서 거리에 대해 1 비트 메시지를 전송하기 위해서 소비되는 에너지는 다음과 같은 수식으로 계산된다.

보내는 경우는 다음과 같다.

$$E_{TX}(1, d) = E_{TX-elec}(1) + E_{TX-amp}(1, d)$$

$$= \begin{cases} lE_{elec} + E\epsilon_{fs} d^2, & d < d_0 \\ lE_{elec} + E\epsilon_{mp} d^4, & d \geq d_0 \end{cases}$$

받는 경우는 다음과 같다.

$$E_{RX} = E_{RX-elec}(1) = lE_{elec}$$

IV. 효율적인 클러스터링 기법

본 연구에서 제안하는 효율적인 클러스터링 기법은 1) 클러스터 헤드 선택, 2) 체인 형성, 3) 데이터 선택 및 전송의 3단계로 구성된다.

1. 클러스터 헤드 선택

클러스터가 생성될 때, 여러 클러스터 헤드 후보군이 선택되는데 이들은 현재 라운드의 최종 클러스터 헤드가 되기 위해서 경쟁을 한다. 각각의 클러스터의 에너지 소비량의 균형을 맞추기 위해서 클러스터 사이의 주기적인 재선택 과정이 수행된다. 후보군이 될 클러스터 선택을 위해서, 제안된 알고리즘은 노드 잔여 에너지와 기지국까지의 거리에 초점을 맞춘다. 기지국과의 거리가 멀고, 잔여 에너지가 낮은 노드는 최종 클러스터 헤드의 후보군에 선택되지 않는다.

■ 후보군 클러스터 노드

□ 센서 노드

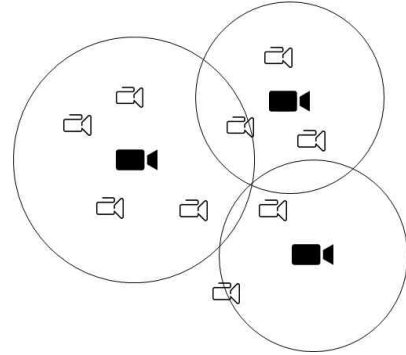


그림 1. 클러스터 헤드간의 경쟁

[그림 1]은 클러스터 헤드 경쟁을 나타낸 것이다. 후보군에 해당하는 클러스터 헤드 선택이 끝나면, 최종 클러스터 헤드를 놓고 후보군의 클러스터들이 경쟁을 시작한다. 후보군에 있는 클러스터들에서는 경쟁 거리에 있는 다른 노드보다 잔여 에너지가 더 많은 클러스터가 클러스터 헤드로 선택된다. 이 때, 동일하지 않은 클러스터 내부의 체인 길이를 생성하려면 클러스터 크기를 제어해야한다. 제안된 알고리즘에서는 기지국에 가까운 클러스터 내 체인이 멀리 있는 클러스터 내 체인보다 더 짧은 길이를 가지게 된다.

알고리즘 1 클러스터 헤드 선택

```

1  if CandidateCluster = True then
2      CompeteClusterHeadMsg(ID, E_remain, D_comp)
3  else
4      TERMINATE;
5  end if
6  When received CompeteClusterHeadMsg from node s_j
7  if distance(s_i, s_j) < s_j · D_comp OR distance(s_i, s_j) < s_i · D_comp then
8      add s_j into cluster group
9  end if
10 while CandidateCluster = True do
11     if remainE(s_i) > remainE(s_j) and s_j ∈ cluster group then
12         FinalClusterHeadMsg(ID)
13         TERMINATE;
14     end if
15     When received FinalClusterHeadMsg from node s_j
16     if s_j ∈ cluster group then
17         ResignationMsg(ID) and then TERMINATE
18     end if
19     When received ResignationMsg from node s_j
20     if s_j ∈ cluster group then
21         Remove s_j from cluster group
22     end if
23 end while
24 send ClusterChainHeadMsg(ID)
25 Not received CompeteClusterHeadMsg from node s_j
26 send ClusterChainHeadMsg(ID)

```

[알고리즘 1]은 클러스터가 여러 개 주어졌을 때 이 중에서 최종적으로 클러스터 헤드를 선택하는 알고리즘을 나타낸 것이다. *CandidateCluster*는 클러스터 후보인지 여부를 판별하는 변수이며, 이 값이 *True*라면 *CompeteClusterHeadMsg*를 통해서 경쟁에 들어가게 된다. 메시지를 보내는데 사용되는 인수는 노드 식별자인 ID, 현재 노드의 잔여 에너지인 E_{remain} , 그리고 기지국으로부터 원격 지점까지의 클러스터 범위를 나타내는 D_{comp} 이다.

[알고리즘 1]의 6-9줄은 노드 s_j 로부터 *CompeteClusterHeadMsg*를 받게 되는 경우를 나타내고 있으며 노드 s_j 의 클러스터 그룹 추가여부를 결정한다. 이미 클러스터 그룹에 있는 s_i 노드로부터 새롭게 추가할 노드 s_j 까지의 거리가 각각의 노드에서 기지국으로 가는 거리보다 짧은 경우에 노드 s_j 를 클러스터 그룹에 추가한다. 여기서 $distance(s_i, s_j)$ 는 노드 s_i 와 노드 s_j 간의 거리를 나타낸다.

[알고리즘 1]의 10-23줄은 최종 클러스터 헤드의 후보들에 대해서 반복적으로 수행된다. 클러스터 그룹에 존재하는 임의의 노드 s_j 에 대해서 잔여 에너지가 가장 많이 남은 노드를 최종 클러스터 헤드로 선정하게 된다. $remainE(s_j)$ 는 s_j 노드의 잔여 에너지를 나타낸다. 만약, 에너지가 동일한 노드가 2개 이상 선정된 경우에는 낮은 ID를 가진 노드를 선정한다. 선정된 노드는 이웃한 클러스터 헤드 노드들에게 *FinalClusterHeadMsg* 메시지를 전송한다. 그리고 이 메시지를 받은 클러스터 그룹에

알고리즘 2 체인 형성	
1	On received <i>ClusterChainHeadMsg</i> from cluster head
2	<i>ClusterChainMemberMsg</i> (ID)
3	if <i>ClusterHead</i> = <i>True</i> then
4	<i>PreviousNode</i> ← <i>CHNode</i>
5	for each node s_i in cluster do
6	if $s_i \neq S_{FS}$ then
7	if $distance(s_i, CHNode_s) \leq \min(distance(s_j, CHNode_s))$ then
8	<i>PreviousNode.SendMsg</i> (s_i)
9	<i>PreviousNode</i> = s_i
10	end if
11	else
12	TERMINATE;
13	end if
14	end for
15	end if

존재하며 이웃한 클러스터 헤드 노드들은 선정되지 못했으므로 *ResignationMsg* 메시지를 전송하게된다. 마지막으로, 메시지를 전송한 노드들은 클러스터 그룹에서 제거된다.

최종 클러스터 헤드가 선정되면, 그 노드의 ID와 함께 *ClusterChainHeadMsg* 메시지를 브로드캐스팅한다.

2. 체인 형성

각각의 클러스터 그룹에서 클러스터 헤드 노드가 라우팅 테이블을 만들고 그것을 가장 가까운 노드에게 전송함으로써 체인 형성이 이루어진다. 라우팅 테이블은 클러스터에 속하는 모든 노드들의 목록과 클러스터 헤드 노드로부터 가장 먼 노드인 타겟 노드인 S_{FS} 를 포함하게 된다. 노드의 목록은 클러스터 헤드로부터 각각 얼마나 가까운지에 따라서 순서가 매겨진다.

체인 형성과정은 다음과 같이 진행된다. 중간 노드는 자신보다 클러스터 헤드 노드에서 멀리 떨어져 있고, 자신과 가까이있는 이웃 노드에만 요청을 수행한다. 그러므로 노드 s_i 에서는 $distance(s_i, CHNode_s) \leq \min(distance(s_j, CHNode_s))$ 의 조건을 만족하는 이웃 노드인 s_j 에게만 요청이 전달된다. $distance(s_i, CHNode_s)$ 는 노드 s_i 와 클러스터 헤드인 $CHNode_s$ 까지의 거리를 나타내며, 노드 s_i 와 노드 s_j 는 S_{FS} 와 $CHNode_s$ 사이의 중간 노드이다.

[알고리즘 2]의 1-15줄은 노드가 요청 패킷을 수신하면 노드가 타겟 노드인지 여부를 확인한다. 만약, 타겟노드가 아니라면 체인에 아직 들어오지 않았고 클러스터 헤드로부터 가장 가까운 거리에 있는 노드에게 요청을 보낸다. 노드 체인 형성 과정은 타겟 노드를 발견할 때까지 수행을 반복한다.

알고리즘 3 클러스터 간 다중 홉 라우팅	
1	s_i . <i>BroadcastMsg</i> (E_{remain} , $distance(BS)$)
2	<i>ChainMemberMsg</i> (ID)
3	if $distance(BS) < Thrs_max$ then
4	<i>message.send</i> (BS)
5	end if
6	for each s_j do
7	On receiving <i>BroadcastMsg</i> (E_{remain} , $distance(BS)$) from s_i
8	if $distance(s_j, BS) > distance(s_i, BS)$ then
9	$A_r \leftarrow s_j.select(\min(distance(s_i, s_j)))$
10	$A_r \leftarrow select(\max(E_{remain}, A_r))$
11	$s_j.SendForward(A_r, rand())$
12	end if
13	end for

3. 데이터 선택 및 전송

제한한 클러스터링 기법은 긴 데이터 전송 단계를 이용하여 이웃한 클러스터 헤드 노드의 잔여 에너지 및 거리를 고려하여 클러스터 간 다중 홉 라우팅 프로토콜을 구축한다.

전송 단계는 많은 데이터를 기지국에 전송할 수 있을 만큼 충분히 길기 때문에 클러스터 헤드 노드와 통신에 사용되는 기지국 사이에는 하나 이상의 경로가 있어야 한다. 그러나 클러스터 헤드 노드간에 동일한 경로를 자주사용하게 되면 그 경로에 해당되는 노드들의 에너지 소모가 잦게 된다. 이는 노드의 에너지

레벨에 변화를 가져올 수 있으며 네트워크 파티셔닝을 유발할 수 있다. 그러므로 클러스터 간의 통신을 위한 에너지 인식 다중 홉 라우팅 프로토콜을 설계하였다.

제안한 클러스터링 기법을 통해 클러스터 헤드 노드와 기지국 사이에 최적에 가까운 경로들을 찾고, 이 경로들을 무작위로 선택하여 데이터 전송에 사용할 것이다. [알고리즘 3]의 1-13줄은 클러스터 헤드 노드가 확정되고 나면 각각 노드 s_i 는 잔여 에너지와 기지국까지의 거리, 그리고 클러스터 헤드 노드의 ID를 네트워크에 브로드캐스팅한다.

또한, 클러스터간 다중 홉 라우팅 프로토콜에 임계값인 $Thrs_max$ 를 사용하였다. 만약 클러스터 헤드로부터 기지국까지의 거리가 임계값인 $Thrs_max$ 보다 같거나 작다면, 기지국으로 데이터를 바로 송신한다. 클러스터 헤드 노드인 s_j 가 브로드캐스팅된 패킷을 받는 경우, 거리를 확인한다. 만약 노드 s_j 가 노드 s_i 보다 기지국에 가까운 경우에는 받은 패킷을 버린다. 그렇지 않다면 노드 s_j 는 받은 패킷에 있는 클러스터 헤드에게 자신의 데이터를 송신한다.

V. 실험 및 결과

제안한 클러스터링 기법의 성능을 평가하기 위해서 Matlab 시뮬레이션 툴을 사용하였으며, 클러스터 헤드, 네트워크 수명 및 노드 수에 따른 에너지 소비 측면을 LEACH, UQ-LEACH 및 EEUC와 비교하였다.

실험을 위한 시뮬레이션 환경은 다음과 같이 구성하였다. 센서 노드가 감지 영역에 균일하게 배치되고 기지국이 무선 센서 네트워크 외부에 위치하도록 구성하였다. 모든 센서 노드는 주기적으로 감지된 이벤트와 전달된 데이터 패킷을 기지국에 전송하게 된다. 시뮬레이션에서 채택하여 사용한 에너지 모델은 Heinzelman이 사용한 것과 동일한 모델을 사용하였다[14].

시뮬레이션에 사용된 파라미터 값은 [표 1]과 같다. $Thrs_max$ 는 130m로 지정하였으며 각 시뮬레이션은 20회 반복하여 수행하였다.

표 1. 시뮬레이션 파라미터

파라미터	값
네트워크 효율 범위	(0,0) ~ (200,200)m
기지국 위치	(100,250)m
센서노드 개수	150
초기 에너지	1 J
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/bit/m ²

ϵ_{mp}	0.0013 pJ/bit/m ⁴
d_0	87m
패킷 크기	2000 bits

[그림 2]는 라운드 별 클러스터 헤드의 에너지 소모량을 나타낸 것이다. 실험결과는 제안한 클러스터링 기법을 사용하였을 때 라운드 당 클러스터 헤드의 에너지 소비량이 다른 기법들보다 현저히 낮았음을 보여주었다. 이는 LEACH, 그리고 개선된 LEACH는 클러스터 헤드가 직접 기지국에게 데이터를 전송하므로 에너지를 더 소비하기 때문이었다. 더하여, EEUC 또한 제안한 클러스터링 기법보다 에너지 소비량이 높았는데, 이는 EEUC가 클러스터 헤드와 기지국 사이의 최적의 단일 라우팅 경로를 찾고 수집된 모든 데이터를 이 경로로 보내므로 이 과정에서 에너지 소모가 극심하기 때문이었다.

제안한 클러스터링 기법에서 클러스터 헤드는 각각의 데이터를 다중 홉을 통해서 기지국에 전송하였고, 클러스터 헤드와 기지국 사이의 유사 최적 경로들을 찾았으며, 그 경로들 중 임의로 선택된 경로를 통해서 데이터를 전송하였고, 이는 상당한 에너지를 절약하는 결과를 보였다.

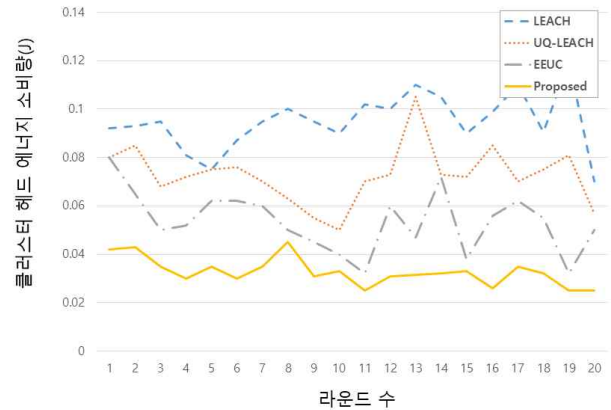


그림 2. 라운드 별 클러스터 헤드의 에너지 소비량

또한, 네트워크 수명을 검사하여 제안한 클러스터링 기법의 에너지 효율을 조사하였다. [그림 3]은 첫 노드가 죽는 시간을 보여주고, [그림 4]는 마지막 노드가 죽을 때까지의 시간을 보여주고 있다. 제안한 기법은 첫 노드가 죽을 때까지의 시간뿐만 아니라, 마지막 노드가 죽을 때까지의 시간을 고려하였을 때 다른 기법들에 비해 상당한 개선이 이루어졌음을 보여주고 있다. 200m x 200m의 감지 환경에 흩뿌려진 센서들에 대해서 EEUC, UQ-LEACH, 그리고 LEACH는 제안한 기법에 비해 노드가 죽는시간이 빨랐는데, 이는 각각의 클러스터끼리 장거리에서 데이터를 주고 받는데 많은 에너지를 소모하기 때문이었다.

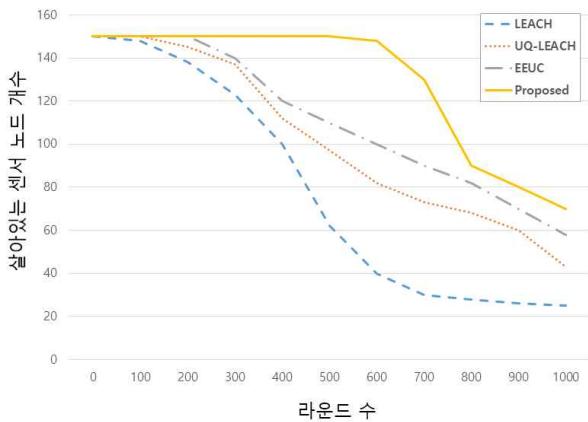


그림 3. 첫 노드가 죽는데 까지 걸리는 시간

또한, UQ-LEACH 및 LEACH의 일부 클러스터 헤드 노드는 데이터를 장거리에서 직접 기지국으로 전달하기 위해서 더 많은 에너지를 소비했다. 그리고 EEUC기법은 클러스터 헤드의 에너지 소모량이 해당 클러스터 내부에 있는 노드 수에 비례하는 경향이 있으므로 기지국에서 멀리 떨어져있고 잔여 에너지가 낮은 클러스터 헤드가 더 일찍 죽는 경향을 보였다. 이는 EEUC가 불균등 클러스터링 기법을 사용하므로 기지국으로부터 멀리 떨어진 클러스터의 경우에는 기지국으로 가까운 클러스터보다 큰 크기를 가지기 때문이다. 제안한 기법은 체인을 형성하여 각 클러스터 헤드의 데이터를 수집함으로써 이러한 문제들을 극복하였다.

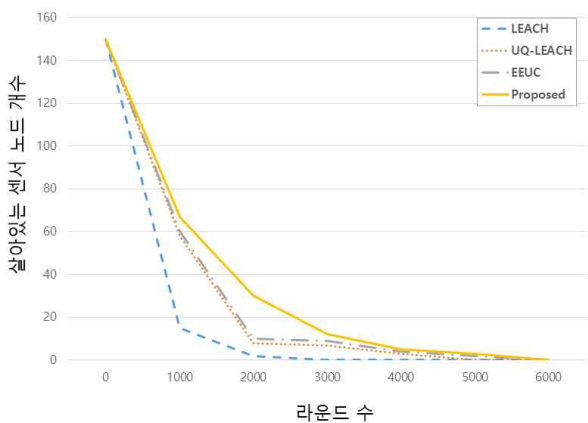


그림 4. 마지막 노드가 죽는데 까지 걸리는 시간

[그림 5]는 20번 반복수행한 각 라운드 별 클러스터 헤드의 에너지 소비변화를 나타낸 것이다. 제안한 기법은 다른 기법과 비교해 보았을 때, 클러스터 헤드 간의 균형적인 에너지 소비를 보였다. LEACH와 UQ-LEACH의 클러스터 헤드 분포를 제

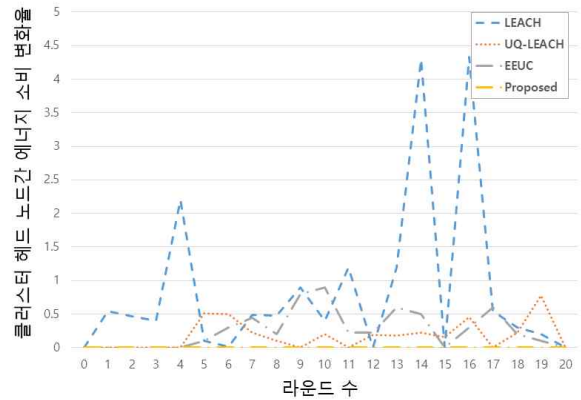


그림 5. 클러스터 헤드에 의해 소모된 에너지의 변화율

어 할 수 없으므로 노드가 많은 클러스터가 존재 할 수 있고 EEUC 기법에서는 일부 클러스터 헤드가 여러 클러스터 헤드에서 받은 데이터를 그대로 연계했기 때문이다. 그러므로 몇몇 클러스터 헤드는 다른 헤드들 보다 에너지를 더 소모하였다. 또한, 제안한 기법은 주로 사용되는 경로만 사용되는 핫 스팟 문제를 유사 최적 경로 검색을 통해서 다중 홉 라우팅 알고리즘을 적용하여 해결하였다.

VI. 결 론

본 논문에서는 데이터 집합과 데이터를 기지국으로 전송할 수 있는 효율적인 클러스터링 방법을 제안하였다. 제안된 방법은 클러스터 헤드간의 에너지 소비를 균형있게 유지하며, 클러스터 내부 노드의 에너지 소비를 보존하고, 클러스터 헤드의 에너지 소비량을 줄였다. 또한, 클러스터 헤드의 부담을 줄이고 무선 센서 네트워크의 수명을 연장할 수 있는 다중 홉 라우팅 프로토콜을 제시하였다.

본 논문에서 제안한 방법은 다양한 무선 센서 네트워크 환경에 적용이 가능하다. 예를 들어, GNSS, GPS, Speed Log와 같은 다양한 센서를 사용하는 통합 항해 시스템(Integrated Navigation System)에 적용하게 된다면 다양한 센서의 정보에 대해서 가용성, 일관성 및 동기화가 보장되므로 유용하게 사용할 수 있다. 실험 결과는 제안한 클러스터링 방법이 네트워크 수명을 명확하게 개선했고, 에너지 소비의 관점에서 기존의 프로토콜인 LEACH, EEUC등과 비교하였을 때 더 균형있는 결과를 보여주었다.

REFERENCES

[1] Akyildiz, Ian F and Su, Weilian and

- Sankarasubramaniam, Yogesh and Cayirci, Erdal, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393-422, 2002
- [2] Huang, Jun and Duan, Qiang and Xing, Cong-Cong and Wang, Honggang, "Topology Control for Building a Large-Scale and Energy-Efficient Internet of Things," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 67-73, 2017
- [3] Huang, Jun and Meng, Yu and Gong, Xuehong and Liu, Yanbing and Duan, Qiang, "A novel deployment scheme for green internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 196-205, 2014
- [4] Lim, Yujin and Kim, Hak-Man and Kang, Sanggil, "A design of wireless sensor networks for a power quality monitoring system," *Sensors*, vol. 10, no. 11, pp. 9712-9725, 2010
- [5] Ahmad, Junaid and Malik, Aamir Saeed and Abdullah, Mohd Faris and Kamel, Nidal and Xia, Likun, "A novel method for vegetation encroachment monitoring of transmission lines using a single 2D camera," *Pattern Analysis & Applications*, vol.18, no. 2, pp. 419-440, 2015
- [6] 최민호, 김영상, "Zigbee와 GPS를 이용한 해너 사고예방 스마트 모니터링 시스템 개발," *스마트미디어 어저널*, vol. 5, no. 3, pp. 1-7, 2016
- [7] Younis, Ossama and Krunz, Marwan and Ramasubramanian, Srinivasan, "Node clustering in wireless sensor networks: Recent developments and deployment challenges," *IEEE network*, vol. 20, no. 3, pp. 20-25, 2006
- [8] Wightman, Pedro M and Labrador, Miguel A, "Topology maintenance: Extending the lifetime of wireless sensor networks," *IEEE Latin America Transactions*, vol. 8, no. 4, pp. 469-475, 2010
- [9] Heinzelman, Wendi Rabiner and Chandrakasan, Anantha and Balakrishnan, Hari, "Energy-efficient communication protocol for wireless microsensor networks," *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pp 10-pp, 2000
- [10] LiCF, YeM and ChenGH, WuJ, "An energy-efficient unequal clustering mechanism for wireless sensor networks," *Proceedings of the 2nd IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS2005), Washing ton, DC, 2005*
- [11] Ever, Enver and Luchmun, Ravi and Mostarda, Leonardo and Navarra, Alfredo and Shah, Purav, "UHEED-an unequal clustering algorithm for wireless sensor networks," 2012
- [12] Ren, Peng and Qian, Jiansheng and Li, Leida and Zhao, Zhikai and Li, Xiaobin, "Unequal clustering scheme based leach for wireless sensor networks," *Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on*, pp. 90-93, 2010
- [13] Yang, J and Zhang, D, "An energy-balancing unequal clustering protocol for wireless sensor networks," *Information Technology Journal*, vol. 8, no. 1, pp. 57-63, 2009
- [14] Heinzelman, Wendi B and Chandrakasan, Anantha P and Balakrishnan, Hari, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660-670, 2002

저 자 소 개



이진우(학생회원)

2017년 숭실대학교 컴퓨터학부 학사 졸업.
2017년 숭실대학교 컴퓨터학과 석사 재학.

<주관심분야 : 시스템 소프트웨어, 운영체제>



모하메드 바니아타(학생회원)

2016년 University of Jordan 석사 졸업.
2016년 숭실대학교 컴퓨터학과 박사 재학.

<주관심분야 : 시스템 소프트웨어, 운영체제>



홍지만(정회원)

1999년 서울대학교 컴퓨터학과 박사 졸업.
2004년 ~ 2007년 광운대 컴퓨터학과 교수.
2007년 ~ 숭실대학교 컴퓨터학부 교수.

<주관심분야 : 시스템 소프트웨어, 운영체제>