

# 중앙 집중식 불균등 체인 클러스터링을 위한 스케줄링 모델

(Scheduling Model for Centralized Unequal Chain Clustering)

지현호\*, 모하마드 바니아타\*\*, 홍지만\*\*\*

(Hyunho Ji, Mohammad Baniata, Jiman Hong)

## 요약

수많은 디바이스들이 무선 네트워크를 통해 연결 되고 있고, 이러한 연결을 효율적으로하기 위한 연구들이 진행되고 있다. 많은 연구에서 효율적인 디바이스 관리를 위해 클러스터링을 사용하고 있지만 클러스터의 특정 노드에 부하가 집중되는 경우가 많아 전체 네트워크가 불안정해질 수 있다. 이러한 문제를 해결하기 위해 본 논문에서는 센서 노드의 효율적인 관리를 위해 중앙 집중식 불균등 체인 클러스터 스케줄링 모델을 제안한다. 클러스터의 구성을 위해 클러스터 헤드 범위와 기지국까지의 거리를 기반으로 하고, 기지국의 위치가 동일하지 않은 동심 체인 클러스터링을 구축하기 위해 주벡터 투사 기법을 사용한다. 데이터의 전송은 다중 무선 액세스 인터페이스인 MIMO(Multiple-Input Multiple-Output)를 활용한다. 실험을 통해 클러스터 헤드의 에너지 소비를 줄이고 네트워크 수명이 향상됨을 보인다.

■ 중심어 : 동심 체인 클러스터링, MIMO, 주벡터 투사 기법

## Abstract

As numerous devices are connected through a wireless network, there exist many studies conducted to efficiently connect the devices. While earlier studies often use clustering for efficient device management, there is a load-intensive cluster node which may lead the entire network to be unstable. In order to solve this problem, we propose a scheduling model for centralized unequal chain clustering for efficient management of sensor nodes. For the cluster configuration, this study is based on the cluster head range and the distance to the base station(BS). The main vector projection technique is used to construct clustering with concentricity where the positions of the base stations are not the same. We utilize a multiple radio access interface, multiple-input multiple-output (MIMO), for data transmission. Experiments show that cluster head energy consumption is reduced and network lifetime is improved.

■ keywords : Centralized Chain Clustering, MIMO, Principal Vector Projection Approach

## I. 서론

시스코(CISCO)에 따르면 2020년까지 500억 개가 넘는 기기가 서로 연결될 것으로 전망된다[1]. 다양한 분야의 다목적 기기 및 센서들이 대부분 무선 네트워크를 통해 연결된다. 무선 센서 네트워크에서 센서 노드는 배터리 용량과 전송 범위가 제한되기 때문에 많은 노드에서 생산되는 데이터를 효율적으로 처리하고 노드의 수명을 연장하기 위한 방법이 필요하다. 일반적으로 이러한 센서 노드들은 네트워크의 낮은 에너지 소비와 균일한 부하 분산을 보장하기 위해 네트워크를 개체 그룹인 클러스터로 분류하여 계층적으로 구성한다[2]. 대규모 네트워크에서 클러스터 헤드는 클러스터 멤버 노드와 비교하여 더 많은 에

너지를 소비한다. 이는 멤버 노드로부터 데이터를 수신 및 집계하고 멀리 있는 기지국에 전송하기 때문이다. 이러한 문제들을 해결하기 위해 클러스터 스케줄링이 필요하다. 클러스터 스케줄링은 네트워크 센서의 수명을 연장하고 시스템 에너지 소비를 줄이기 위해 클러스터 멤버 간의 작업을 분산한다.

요즘에는 ZigBee, 블루투스, Wi-Fi, LTE를 비롯한 많은 통신 인터페이스가 센서 네트워크에서 사용 가능하다[3-5]. 각 통신 인터페이스는 주파수, 데이터 전송 속도, 전송 범위 및 에너지 소비와 같은 개별 특징이 있다. MIMO의 활용은 링크의 총 전송 속도를 증가시키고 독립적인 페이딩 경로를 통해 시그널 버전을 교환함으로써 통신 인터페이스의 다양성을 얻는다. 그러나 데이터 전송을 위해 적절한 통신 인터페이스를 효과적으로

\* 비회원, 송실대학교 컴퓨터학과

\*\* 비회원, 송실대학교 컴퓨터학과

\*\*\* 종신회원, 송실대학교 컴퓨터학부

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

선택하는 것에 어려움이 있다. 센서 네트워크에서 전송 비용이 중요하기 때문에 통신 인터페이스를 선택할 때 애플리케이션 요구에 맞는 네트워크 전송 속도와 노드 사이의 거리가 고려되어야 한다. 그러나 센서 네트워크를 위한 에너지 절약 클러스터 스케줄링에 대한 연구가 부족한 실정이다. 무선 센서 네트워크 분야에서 에너지 소비 및 네트워크 수명을 향상시키기 위해 많은 클러스터 스케줄링 모델이 제안되고 있지만 에너지 소비, 핫스팟(Hot Spot), 에너지 구멍(Energy Hole) [6-7]문제는 여전히 존재한다. 핫스팟 이슈를 극복하기 위해 본 논문에서는 매 라운드마다 각 노드가 정보 패킷을 전송하고 데이터를 체계적으로 수집하는 에너지 균형을 위한 중앙 집중식 불균등 체인 클러스터링을 위한 스케줄링 모델을 제안한다.

2장에서 클러스터 스케줄링 모델에 대한 관련 연구를 소개하고, 3장에서 제안하는 스케줄링 모델에 대해 설명한다. 4장에서 제안한 스케줄링 모델의 실험 결과를 보이고, 5장에서 결론을 맺는다.

## II. 관련 연구

네트워크 자원 문제를 해결하기 위해 많은 불균등 클러스터 스케줄링 모델 [8-11]이 제안되었다. EEUC(Energy Efficient Unequal Clustering protocol) [8]는 네트워크의 노드를 크기가 다른 클러스터로 분할한다. 클러스터의 크기는 기지국까지의 거리에 비례하도록 설정된다. 기지국에 가까운 클러스터의 크기는 멀리 떨어져있는 클러스터의 크기보다 작다. EEUCLC(Energy Efficient Unequal Chain Length Clustering) [9]는 분산 경쟁 프로토콜이며, 네트워크는 몇 개의 동일하지 않은 길이의 클러스터 내부 체인으로 그룹화된다. EEUCLC는 클러스터 헤드를 통해 차선의 다중 홉 경로를 찾아서 집계된 데이터를 기지국에 전달한다. EBCAG(Eradig-Balancing Unequal Clustering Approach for Gradient Based routing) [10]는 노드를 크기가 다른 클러스터로 분할하고 홉 수를 사용하여 클러스터 헤드 분포를 제어한다. 각 노드는 기지국에 대한 최소 홉 카운트로 정의되는 경도 값(gradient value)을 유지한다. LEACH(Lower Energy Adaptive Clustering Hierarchy)[11]는 계층적인 라우팅 프로토콜로, 모든 센서 노드가 직접 통신 할 수 있고 많은 라운드로 실행될 수 있다는 가설에 기반한다. 각 라운드는 클러스터가 구성 될 때 설정 단계부터 시작되며 그 다음 데이터가 일반 노드에서 클러스터로 전송 될 때 정상 상태 단계가 시작된다. 데이터를 클러스터 헤드로 전송 한 다음 기지국으로 전송한다.

## III. 에너지 균형 스케줄링 모델

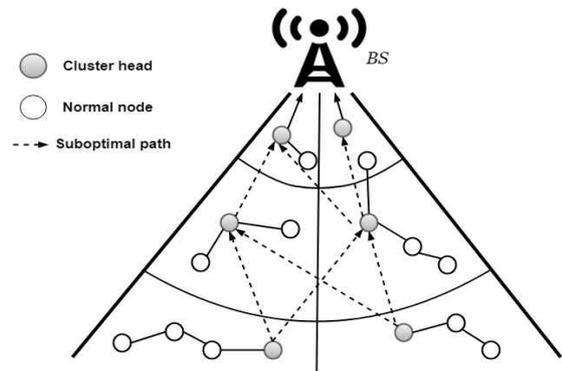


그림 1. 동심 체인 클러스터 구성

제안한 스케줄링 모델은 LEACH를 기반으로 클러스터 헤드가 기지국에 의해 선택되는 중앙 집중식 불균등 체인 클러스터링 모델이다. 잔류 에너지가 가장 높고 기지국과의 거리가 가장 짧은 노드가 클러스터 헤드가 될 확률이 높다. 네트워크를 몇 개의 불균등 동심 트리 체인으로 나누고, 경로 및 각 노드들이 에너지 소비를 고려하여 클러스터 헤드가 차선의 멀티 홉 경로를 찾아 스케줄링한다. 체인 토폴로지를 구축하기 위해 기지국의 위치를 고려한 주벡터 투사 기법을 사용한다. 노드의 데이터 전송 거리는 기지국까지의 거리에 따라 감소하고, 이는 기지국에 가까운 클러스터의 크기가 멀리 떨어져 있는 것보다 작은 것을 의미한다.

동심 클러스터링은 센서 네트워크가 여러 클러스터로 그룹화 될 때 클러스터의 모양이 동심 체인임을 말한다. 제안하는 스케줄링 모델은 클러스터 헤드 선택, 체인 형성, 데이터 수집 및 전송의 3 단계로 구성된다. 그림 1은 크기가 다른 원들이 다른 동심 체인 클러스터를 나타내고, 다중 홉 전달 모델을 사용하는 것을 보여준다.

### 1. 네트워크 모델

$M \times M$  정사각형 필드에  $N$ 개의 센서가 균일하게 배치되어 있다.  $i$ 번째 센서는  $s_i$ 로 나타내고 해당 센서 노드의 집합은  $S = \{s_1, s_2, \dots, s_n\}$ ,  $1 \leq i \leq |S|$ ,  $|S| = N$ 이다. 네트워크 모델을 단순화하기 위해 다음과 같은 가정을 한다.

- (1) 모든 센서는 MIMO가 가능하다.
- (2) 네트워크의 노드는 계산, 통신 성능 등이 다르고 고유 식별자가 있다.
- (3) 에너지는 제한적이다.
- (4) 모든 센서 노드는 배치된 위치를 모른다.
- (5) 노드는 전력 제어를 할 수 있다.
- (6) 각 센서는 데이터 전송을 위한 통신 인터페이스를 선택할 수 있다.

본 논문에서는 W. B. Heinzelman이 제안한 무선 모델을 사

용하였다[10]. 해당 모델에는 무선으로 데이터를 송수신 할 때 필요한 에너지가 있다. 거리가  $d$  일 때 1 비트 메시지를 전송하기 위해 사용되는 에너지는 수식 1, 2와 같다.

$$E_{TX}(l, d) = E_{TX-elec}(l) + E_{TX-amp}(l, d) \\ = \begin{cases} lE_{elec} + E\epsilon_{fs}d^2, & d < d_0 \\ lE_{elec} + E\epsilon_{amp}d^4, & d \leq d_0 \end{cases} \quad (1)$$

$$E_{RX} = R_{RX-elec}(1) = lE_{elec} \quad (2)$$

## 2. 클러스터 헤드 선택

기지국에서 모든 센서 노드에 특정 메시지를 브로드캐스팅한 후, 수신된 신호의 강도를 이용하여 각 노드는 기지국과의 거리를 구한다. 메시지를 받은 노드는 경쟁 범위를 계산하고 기지국에 응답 메시지(예상 잔여 에너지, 조정된 경쟁 범위, 노드 ID)를 보낸다. 알고리즘 1과 같이 경쟁 범위 내 잔여 에너지가 높은 노드를 클러스터 헤드로 선택하고, 이를 이용하여 불균등 체인 클러스터를 만든다. 제안하는 모델에서는 에너지가 많은 노드가 더 많은 작업을 하게하여 전체적인 수명을 동일하게 한다. 따라서 기지국에서 멀고 잔여 에너지가 많은 노드가 더 긴 체인을 갖는다. 이를 위해 잔여 에너지가 높은 노드는 기지국으로부터 멀어질수록 경쟁 범위가 넓어져야 한다. 노드의 경쟁 범위를 계산하는 식은 수식 3과 같다.

### 알고리즘 1. 클러스터 헤드 선택

---

```

1: BS.Broadcast_MSG(Hello)
2: On receiving a Broadcast_MSG(Hello) from BS
3: for each node  $s_j$  do
4:    $s_j$ .Computes( $D_c$ )
5:    $s_j$ .Report_MSG(ID,  $D_c$ ,  $E_r$  esidual)
6: end for
7: On receiving a Report_MSG from node  $s_j$ 
8: if  $d(s_i, s_j) < s_j.D_c$  OR  $d(s_i, s_j) < s_i.D_c$  then
9:   Add  $s_j$  to  $s_i$  ClusterGroup
10: end if
11: if  $s_i.E_r$  esidual  $> s_j.E_r$  esidual,  $\forall s_j \in s_i$  ClusterGroup then
12:    $CH \leftarrow BS.select(s_i)$ 
13: else
14:    $CH \leftarrow BS.select(s_j)$ 
15: end if
16: for each cluster group k do
17:   BS.SendMatrix(ClusterGroup_ID, CH_ID,  $d_{ij}$ )
18: end for

```

---

$$D_c = [1 - \alpha \frac{(d_{max} - d(s_i, BS))}{(d_{max} - d_{min})} - \beta(1 - \frac{E_{residual}}{E_{max}})] D_c^0 \quad (3)$$

이 때  $d(s_i, BS)$ 는 노드  $s_i$ 로부터  $BS$ 까지의 거리이고,  $\alpha$ 와  $\beta$ 는 0과 1 사이의 값을 갖는 가중치이며,  $D_c^0$ 는 경쟁 범위의 최댓값이다.  $E_{residual}$ 는 노드  $s_i$ 의 추정 잔류 에너지이다.  $E_{max}$ 는 초기 에너지로 모든 노드에서 동일하다.  $d_{max}$ 와  $d_{min}$ 은 센서 노드와 기지국간의 최대 및 최소 거리를 나타낸다. 기지국이 센서 노드로부터 응답 메시지를 수신한 후 기지국은 후보 노드 중 클러스터 헤드를 결정한다. 그 후 기지국은 클러스터 헤드의 행렬을 저장하고 네트워크에 브로드캐스팅 한다. 행렬에는 특정 클러스터에 속한 모든 노드의 목록과 클러스터 헤드의 ID, 클러스터간의 거리가 들어있다.

$$\begin{bmatrix} h.d_{11} & \dots & h.d_{1n} \\ \vdots & h.d_{ij} & \vdots \\ h.d_{n1} & \dots & h.d_{nn} \end{bmatrix}$$

$i$ 는 노드 ID를 나타내고  $d_{ij}$ 는 노드  $i$ 에서  $j$ 까지의 거리를 나타내며,  $h$ 는 클러스터 헤드의 ID를 나타낸다. 행렬을 수신 받으면 각 노드는 클러스터 헤드와 다른 노드 사이의 거리를 알 수 있고, 이를 이용하여 송신 전력을 조절하여 에너지 소비를 줄일 수 있다.

## 3. 체인 구현

기지국은 무선 센서 네트워크의 구성 요소이며 기지국의 위치가 클러스터 체인 또는 네트워크 토폴로지 형태에 영향을 미친다. 이는 노드의 에너지 소모에 영향을 미치고 이로 인해 네트워크의 성능, 품질, 수명에 영향을 미친다. 따라서 본 논문은 기지국의 위치에 맞는 적절한 체인 토폴로지를 알고리즘 2를 이용하여 만든다. 불규칙한 클러스터 체인 토폴로지를 피하기 위해 투사 기법을 사용하여 체인을 만든다. 각 클러스터의 모든 노드는 주벡터를 투사하고 벡터 값에 따라 클러스터 노드를 정렬한다. 기본 벡터는 수식 4와 같이 정의된다.

$$\begin{cases} (x_M, 0) & \text{if } x_a = 0 \\ (y_M, 0) & \text{if } y_a = 0 \\ (-1, \frac{x_M}{y_M}) & \text{otherwise} \end{cases} \quad (4)$$

여기서  $(x_M, y_M)$ 은 네트워크 필드의 중심(M) 위치를 나타내고,  $x_a$ 와  $y_a$ 는  $\vec{a}$ 이며 기지국과 중심(M)을 연결하는 데 사용된 벡터이다.  $\vec{a}$ 는 수식 5와 같이 정의된다.

$$\vec{a} = (x_B - x_M, y_B - y_M) \quad (5)$$

$(x_B, y_B)$ 는 기지국의 위치를 나타낸다. 기지국은 네트워크

## 알고리즘 2. 체인 형성 방법

---

```

1: On receiving a Report_MSG from node  $s_j$ 
2:  $BS.Calculate(\vec{a})$ 
3:  $BS.Calculate(\vec{v}) \perp (\vec{a})$ 
4: for each node  $s_j \in ClusterGroup\ k$  do
5:    $BS.Calculate(v_j)$ 
6:    $BS.Calculate P_j(s_j)$ 
7:    $S_{P_j} \leftarrow BS.Sort(P_j(s_j))$ 
8: end for
9: for each ClusterGroup  $k$  do
10:   $BS.SendMatrixs(S_{P_j})$ 
11: end for
12:  $previous\_node \leftarrow lowest(P_j(s_j))$ 
13: On receiving Matrixs from BS
14: for each node  $s_i \in ClusterGroup\ k$  do
15:  if  $s_j, \max(P_j(s_j))$  then
16:   if  $P_j(s_j) \leq \min(P_i(s_i))$  then
17:     $previous\_node.SendMessage(P_j(s_j))$ 
18:     $previous\_node = (P_j(s_j))$ 
19:  end if
20: else
21:    $Exit$ 
22: end if
23: end for

```

---

필드에 있는 모든 노드로부터 응답 메시지를 받은 후, 수신 강도를 이용하여 각 노드의 위치를 추정한다. 이를 이용하여 어떤 노드를 클러스터 헤드로 선정할지와 체인을 어떤 모양으로 형성할 지 계산한다. 첫 번째 기지국은 기지국과 네트워크 필드의 중심점을 연결하는 공간 벡터  $\vec{a}$ 와  $\vec{a}$ 의 수직 벡터  $\vec{v}$ 를 계산한다. 그 후 기지국은 클러스터 그룹의 각 노드와 네트워크 필드의 중심점을 연결하는 벡터  $s_j$ 를 계산한다. 그 다음 기지국은 아래 공식을 사용하여 벡터  $\vec{v}$ 에 대한 벡터  $s_j$ 를 반영한다. 계산이 끝나면 기지국은 행렬을 생성하고 이를 브로드캐스팅 한다. 행렬에는 투사 길이 값이 증가하는 방식으로 정렬된 노드가 있다. 이를 통해 각 노드는 체인 토폴로지를 구성한다.

$$P_j(S_j) = \frac{\vec{v} \cdot \vec{s}_j}{|\vec{v}|} \quad (6)$$

기지국은 그림 2와 같이 클러스터 3의 노드  $s_4$ 를 계산하고, 노드  $s_4$ 를  $\vec{v}$ 로 투사한 값을 나타내는  $P_j(s_j)$ 를 구한다. 위 과정은 동일한 클러스터의 나머지 노드에도 적용 된다. 그 후 클러스터 3의 모든 노드는 투사 값이 증가하는 방식으로 정렬되고 체인 토폴로지를 구성한다.

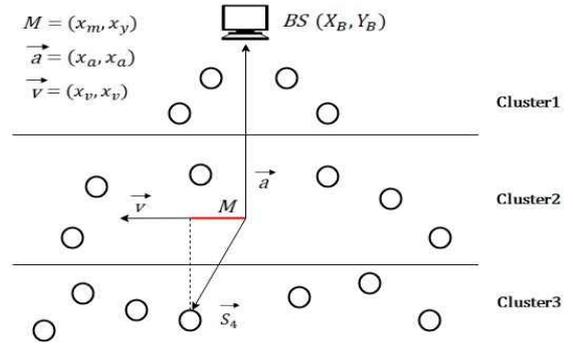


그림 2. 클러스터 체인 구성의 프로젝션 계산

## 4. 클러스터 간 다중 홉 라우팅

클러스터링 작업 시간인  $T_{\omega}$ 는 클러스터 스케줄링 모델이 네트워크를 클러스터링하는데 소요된 시간이다. 데이터 전송 시간인  $T_{dt}$ 는  $T_{\omega}$  종료 시간과 다음  $T_{\omega}$  시작 시간의 차이이다. 오버헤드를 최소화하기 위해서  $T_{dt} \geq T_{\omega}$ 인지 확인해야한다. 본 논문에서 클러스터 헤드 노드와 기지국 사이에 차선의 경로가 있음을 고려한다. 최적 경로를 자주 사용하면 해당 경로에 있는 노드의 에너지 소모가 늘어난다. 이는 네트워크 파티셔닝을 유발한다. 따라서 제안하는 스케줄링 모델은 알고리즘 3과 같이 확률 기반 멀티 홉 경로 설정을 한다. 클러스터 헤드 노드와 기지국 사이에 여러 경로를 찾고 그중 사용할 경로는 확률적으로 선택된다. 그림 3은 차선의 멀티 홉 라우팅을 보인다.

## 알고리즘 3. 클러스터 간 다중 홉 라우팅

---

```

1:  $Cost \leftarrow zero$ 
2:  $BS.Broadcast\_MSG(Cost)$ 
3: if  $d_{BS} \leq Tx_{max}$  then
4:   $Message.Send(BS)$ 
5: end if
6: for each  $s_j$  do
7:  On receiving  $Broadcast\_MSG(Cost)$  from  $s_i$ 
8:   $s_j.Compute(TotalCost)$ 
9:  if  $TotalCost = high$  then
10:    $s_j.Discard\ s_i.Broadcast\_MSG$ 
11:  else
12:    $RT \leftarrow s_j.Add\ s_i$ 
13:   for each node  $s_j \in RT$  do
14:     $s_j.Assign(probability)$ 
15:   end for
16:    $s_j.Compute(average)$ 
17:    $Cost \leftarrow s_j.add(average)$ 
18:  end if
19:   $s_j.NextHopBroadcast\_MSG(Cost)$ 
20: end for

```

---

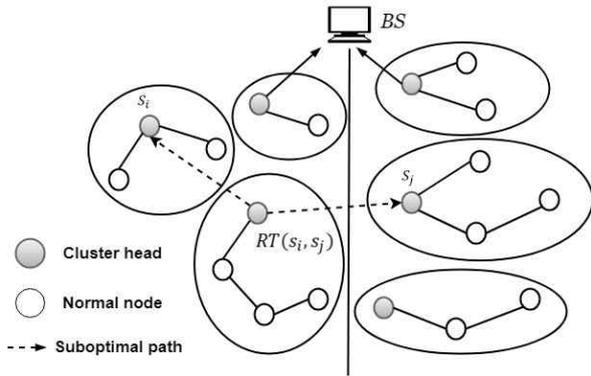


그림 3. 차적 멀티 홉 라우팅의 도시화  
본 논문은 J. H. Chang이 제안한 간단한 에너지 행렬을 사용한다[13]. 행렬은 경로를 사용하는 비용 및 경로에 따른 클러스터 헤드 노드의 잔여 에너지 정보를 포함한다.

$$Cost_{ij} = e_{ij}^c E_{residual}^2 \quad (7)$$

$Cost_{ij}$ 는 클러스터 헤드 노드  $i$ 와  $j$  사이의 비용을 나타내며,  $e_{ij}$ 는 링크에서 송신 및 수신하는 데 사용되는 에너지이다.  $E_{residual}$ 은 클러스터 헤드 노드에서 예상되는 현재 잔여 에너지를 나타내며,  $c_1$ 과  $c_2$ 는 가중 요소이다. 클러스터 헤드 노드가 식별되고 모든 노드가 현재 라운드에 속한 클러스터 그룹을 알고 있으면 기지국이 라우팅 테이블을 생성한 다음 대상 헤드 노드 방향으로 브로드캐스팅한다. 기지국은 라우팅 테이블에서 비용 필드가 0인 값을 찾고 브로드캐스팅하기 전에 클러스터 헤드 ID 목록을 추가한다.

제안하는 스케줄링 모델에 한계값  $Tx\_max$ 를 지정하여 헤드 노드와 기지국까지의 거리가  $Tx\_max$  값보다 작거나 같으면 기지국에 직접 데이터를 전송한다. 클러스터 헤드 노드  $s_j$ 가 브로드캐스팅된 라우팅 테이블을 수신하면 요청을 보낸 중간 클러스터 헤드 노드  $s_i$ 에 대한 에너지를 계산하고 수식 8과 같이 경로의 총 비용에 추가한다.

$$Tcost_{s_i, s_j} = Cost_{s_i, s_j} + e_{ij}^c E_{residual}^2 \quad (8)$$

클러스터 헤드 노드  $s_j$ 는 가장 높은 비용의 경로를 버리고 라우팅 테이블에 해당 경로를 추가하지 않는다. 저비용의 경로를 갖는 클러스터 헤드 노드  $s_i$ 만이  $s_j$ 의 라우팅 테이블  $RT_j$ 에 추가된다. 정상 노드는 수신 패킷을 확인하고 자신의 ID를 포함하지 않으면 폐기한다.

$$RT_j = \{i | Tcost_{s_i, s_j} \leq \alpha \cdot \min_k(Tcost_{s_k, s_j})\} \quad (9)$$

클러스터 헤드 노드  $s_j$ 는 확률이 비용에 반비례하는 라우팅 테이블  $RT_j$ 에서 각 인접 클러스터 헤드  $s_i$ 에 대한 확률을 할당

한다.

$$P_{s_i, s_j} = \frac{1}{Tcost_{s_i, s_j} \sum_{k \in RT_j} \frac{1}{Tcost_{s_k, s_j}}} \quad (10)$$

그 후, 클러스터 헤드 노드  $s_j$ 는 라우팅 테이블의 인접 클러스터 헤드 노드를 통해 기지국까지의 평균 비용을 계산한다.

$$Tcost(s_j) = \sum_{i \in RT_j} Tcost_{s_i, s_j} P_{s_i, s_j} \quad (11)$$

마지막으로, 현재 클러스터 헤드 노드의 평균 비용이 비용 필드에 추가되어 최종 목표 클러스터 헤드가 발견 될 때까지 다음 대상 클러스터 헤드 노드로 보낸다.

#### IV. 실험 결과

제안하는 스케줄링 모델을 평가하기 위해 Matlab Simulation tools [14]을 사용하여 실험을 수행한다. 실험을 위해 센서 노드를 균일하게 배치하고 기지국이 무선 센서 네트워크 외부에 위치하는 환경을 설정한다. 실험에서 센서는 다양한 인터페이스 중 선호도에 따라 선택될 수 있다고 가정한다. 표 1은 실험 파라미터의 값을 보여준다. 실험에서  $\alpha$ ,  $\beta$ 를 각각 0.5로 설정하고  $c_1$ 은 1,  $c_2$ 는 50으로 설정한다.  $D_c^0$ 는 0에서 200으로 값을 변화시키며 실험한다.

표 1. 실험의 평가를 위한 파라미터 값

Parameter	Value
Network coverage	(0,0)-(200,200)m
Base station location	(100,250)m
N	150
$E_{elec}$	50 nJ/bit
$\epsilon_{fs}$	10 pJ/bit/m <sup>2</sup>
$\epsilon_{amp}$	0.0013 pJ/bit/m <sup>4</sup>
packet size	1000 bits
$E_{DA}$	5 nJ/bit/signal
Initialenergy	1J
Interfaces	Wiri,Bluetooth,Zigbee,LTE
$d_0$	40m
BroadcastPacketSite	25 bytes

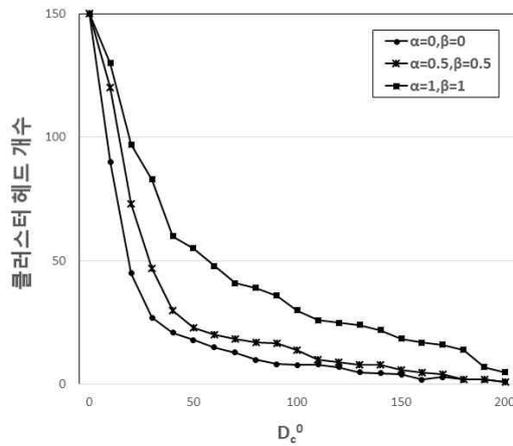


그림 4.  $D_c^0$  대비 클러스터 헤드 수

1. 클러스터 헤드 분산

실험을 통해  $\alpha, \beta$ 의 값과 지정된  $D_c^0$  사이의 관계에 따른 클러스터 헤드 수를 분석한다. 그림 4는 제안하는 스케줄링 모델이 선택한 클러스터 헤드의 평균 개수를 나타낸다. 실험 결과  $\alpha$ 와  $\beta$ 의 값이 클 때 클러스터 헤드의 수가 많다. 이는  $\alpha$ 와  $\beta$ 가 증가하면 경쟁 반경이 감소하고, 더 많은 클러스터가 생성되기 때문이다. 또한, 클러스터 헤드의 수는 반경의 증가에 따라 감소하는데, 이는 생성된 클러스터 헤드의 수가  $D_c^0, \alpha$  및  $\beta$ 에 의해 결정됨을 나타낸다.

$\alpha = 0.5, \beta = 0.5, D_c^0 = 50$ 으로 설정하고 제안하는 스케줄링 모델을 실행 시 그림 5의 클러스터 수의 분포가 나타난다. 제안

하는 스케줄링 모델의 클러스터 수가 UN-LEACH의 클러스터 수보다 일정하다는 것을 나타낸다. UN-LEACH의 분포 일정하지 않은 이유는 무작위 방법을 사용하여 클러스터 헤드를 생성했기 때문이다.

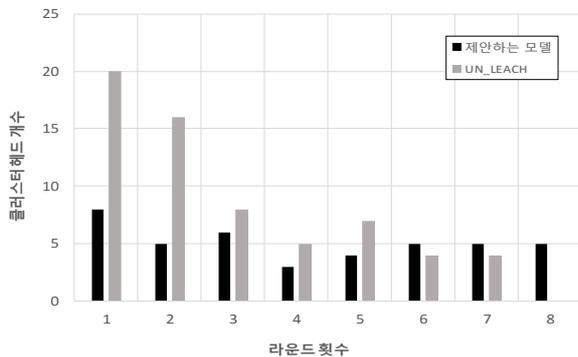


그림 5. 라운드대비 클러스터 헤드 수

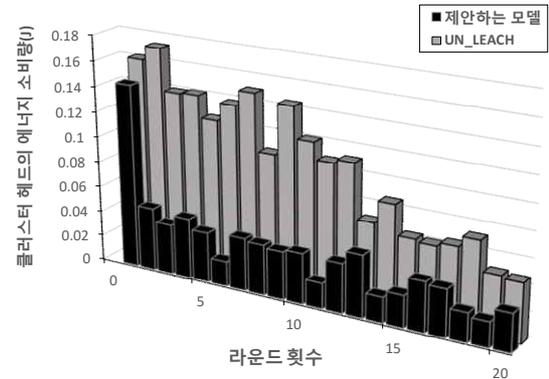


그림 6. 라운드 숫자 대비 클러스터 헤드 결정 에너지 소비

2. 에너지 효율

이 절에서는 제안하는 스케줄링 모델의 클러스터 헤드의 에너지소비량과 네트워크의 수명을 통해 에너지 효율을 분석한다. 그림 6은 제안하는 스케줄링 모델과 UN-LEACH에서 클러스터 헤드가 소비한 에너지 비용을 나타낸다. 제안하는 스케줄링 모델을 사용하는 것이 UN-LEACH를 사용하는 것보다 라운드 당 클러스터 헤드의 에너지 소모량이 적은 것을 볼 수 있다. 이는 UN-LEACH의 클러스터 헤드가 데이터를 곧바로 기지국으로 전달하여 더 많은 에너지를 소비했기 때문이다. 또한 클러스터 헤드가 임의로 선택되었기 때문에 네트워크 전체에 균일하게 분산되지 않았다. 제안하는 스케줄링 모델에서 클러스터 헤드는 차선의 경로를 찾아 멀티 홉을 통해 기지국으로 데이터를 전송한다. 또한 무작위로 선택된 경로를 통해 데이터를 전송

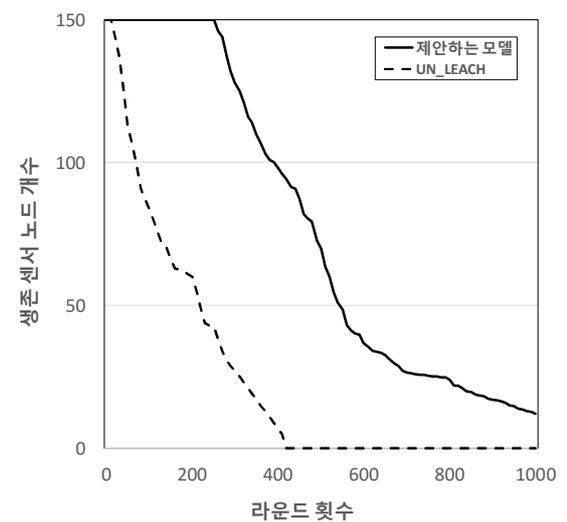


그림 7. 첫 번째 노드의 서비스 가능 라운드 횟수 하므로 상당한 양의 에너지소모를 줄일 수 있다.

그림 7은 첫 번째 노드가 수명을 다할 때까지의 시간을 보여

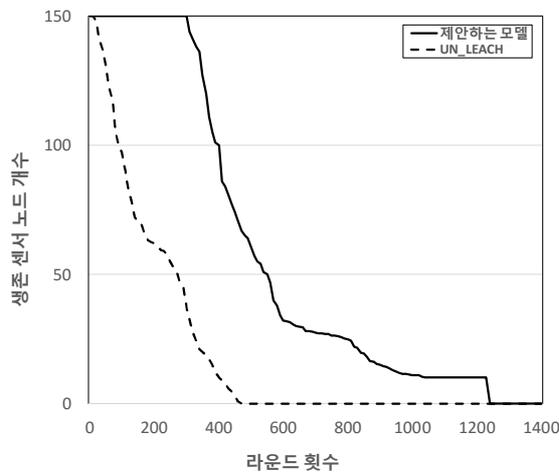


그림 8. 마지막 노드의 서비스 가능 라운드 횟수

주며 그림 8은 마지막 노드가 수명을 다할 때까지의 시간을 보여준다. 제한하는 스케줄링 모델은 UN\_LEACH와 비교하여 첫 번째 노드의 수명뿐만 아니라 마지막 노드의 수명도 향상시킨다. UN\_LEACH의 센서 노드는 드물게 분산되어 데이터를 장거리로 전송하기 위해 더 많은 에너지를 소비하기 때문에 제한하는 스케줄링 모델에 비해 수명이 짧은 것을 볼 수 있다. 제한하는 스케줄링 모델은 체인 스키마를 사용하고 클러스터 헤드 선정 시 거리가 멀고 잔여에너지가 낮은 노드를 선택하지 않는 스케줄링 정책으로 위 문제를 해결했다.

## V. 결론

본 논문에서는 센서 디바이스를 불균등 동심 체인 클러스터의 계층 구조로 구성하여 네트워크 수명을 유지하고 핫스팟 문제를 해결한다. 또한, 네트워크 수명을 연장하는 새로운 중앙 집중식 스케줄링 모델을 소개했다. 제한한 스케줄링 모델은 유용한 체인 토폴로지 생성, 클러스터 헤드 간의 에너지 균형, 클러스터 내 노드의 에너지 절약의 세 가지 기능을 제공한다. MATLAB 시뮬레이션 툴을 이용한 시뮬레이션 결과, 제한한 스케줄링 모델이 UN\_LEACH와 비교하여 클러스터 헤드의 에너지 소비를 줄이고 네트워크 수명을 향상시켰음을 보였다.

## REFERENCES

[1] D. Evans, "The Internet of Everything: How More Relevant and Valuable Connections Will Change the World," Cisco IBSG, pp.1-9, 2012.  
 [2] P. M. Wightman and M. A. Labrador, "Topology

Maintenance: Extending the Lifetime of Wireless Sensor Networks," IEEE Latin America Transactions, Vol. 8, no. 4, pp. 469-475, Oct. 2010.

- [3] 최민호, 김영상, "Zigbee와 GPS를 이용한 해녀 사고예방 스마트 모니터링 시스템 개발," *스마트미디어저널*, 제5권, 제3호, 1-7쪽, 2016년 9월
- [4] 이효승, 오재철, "도어 안전고리 스마트 제어시스템 설계 및 구현," *스마트미디어저널*, 제6권, 제1호, 62-67쪽, 2017년 3월
- [5] 조익현, 황태규, 김대호, 홍지만, "스마트 기기와 Wi-Fi 인프라를 이용한 실내 측위 시스템," *스마트미디어저널*, 제4권, 제2호, 68-76쪽, 2015년 6월
- [6] R. C. Shah and J. M. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," 2002 IEEE Wireless Communications and Networking Conference Record. WCNC 2002, Orlando, Florida, USA, Vol. 1. pp. 350-355, Mar. 2002
- [7] X. Wu, G. Chen and S. K. Das, "On the Energy Hole Problem of Nonuniform Node Distribution in Wireless Sensor Networks," 2006 IEEE International Conference on Mobile Ad Hoc and Sensor Systems, Vancouver, BC, Canada, pp. 180-187, Jan. 2006
- [8] C. Li, M. Ye, G. Chen and J. Wu, "An Energy-Efficient Unequal Clustering Mechanism for Wireless Sensor Networks." IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, Washington, DC, USA, pp. 597 - 604, Nov. 2005
- [9] M. Baniata and J. Hong, "Wireless Communications and Mobile Computing," Vol. 2017, Article ID 5790161, 12 pages, 2017
- [10] T. Liu, Q. Li and P. Liang, "An Energy-Balancing Clustering Approach for Gradient-Based Routing in Wireless Sensor Networks," *Computer Communications*, Vol. 35, No. 17, pp. 2150-2161, Oct. 2012
- [11] P. Ren, J. Qian, L. Li, Z. Zhao and X. Li, "Unequal Clustering Scheme Based LEACH for Wireless Sensor Networks," 2010 Fourth International Conference on Genetic and Evolutionary Computing, Shenzhen, China, pp. 90-93, Feb. 2010
- [12] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," in IEEE Transactions on Wireless Communications, Vol. 1, No. 4, pp. 660-670, Oct. 2002

- [13] J. H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks," Proceedings IEEE INFOCOM 2000, Conference on Computer Communications, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Tel Aviv, Israel, pp. 22-31, Vol. 1, Aug. 2000
- [14] Inc MathWorks, "MATLAB: The Language of Technical Computing. Desktop Tools and Development Environment," *MathWorks*, Ver. 7, Vol. 9, 2005

---

저 자 소 개

---



**지현호**

2013년 숭실대학교 컴퓨터학부 학사 졸업.  
 2013년~현재 숭실대학교 컴퓨터학과 석박통합 재학.  
 2018년~현재 전자부품연구원 에너지 ICT융합연구소 연구원.

<주관심분야 : 시스템 소프트웨어, 운영체제>



**모하매드 바니아타**

2016년 University of Jordan 석사 졸업.  
 2018년~현재 숭실대학교 컴퓨터학과 박사 재학.

<주관심분야 : 시스템 소프트웨어, 운영체제>



**홍지만(중신회원)**

2003년 서울대학교 컴퓨터공학과 박사 졸업  
 2004년~2007년 광운대학교 컴퓨터공학과 교수.  
 2007년~현재 숭실대학교 컴퓨터학부 교수

<주관심분야 : 시스템 소프트웨어, 운영체제>