Research Article

Energy-Efficient Unequal Chain Length Clustering for Wireless Sensor Networks in Smart Cities

Mohammad Baniata and Jiman Hong

School of Computer Science & Engineering, Soongsil University, Seoul, Republic of Korea

Correspondence should be addressed to Jiman Hong; gmaneos@gmail.com

Received 25 April 2017; Accepted 1 August 2017; Published 7 September 2017

Academic Editor: Syed Hassan Ahmed

Copyright © 2017 Mohammad Baniata and Jiman Hong. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The recent advances in sensing and communication technologies such as wireless sensor networks (WSN) have enabled low-priced distributed monitoring systems that are the foundation of smart cities. These advances are also helping to monitor smart cities and making our living environments workable. However, sensor nodes are constrained in energy supply if they have no constant power supply. Moreover, communication links can be easily failed because of unequal node energy depletion. The energy constraints and link failures affect the performance and quality of the sensor network. Therefore, designing a routing protocol that minimizes energy consumption and maximizes the network lifetime should be considered in the design of the routing protocol for WSN. In this paper, we propose an Energy-Efficient Unequal Chain Length Clustering (EEUCLC) protocol which has a suboptimal multihop routing algorithm to reduce the burden on the cluster head and a probability-based cluster head selection algorithm to prolong the network lifetime. Simulation results show that the EEUCLC mechanism enhanced the energy balance and prolonged the network lifetime compared to other related protocols.

1. Introduction

As cities come to be more and more crowded, public safety, efficient transport, preserving energy, lessening expenses, and improving the quality of life are becoming indispensable. Therefore, the smart city term is gaining popularity among scientific researchers. Chen [1] defined smart cities as “taking the advantage of communications and sensor capabilities sewn into the cities infrastructures to optimize electrical, transportation, and other logistical operations supporting daily life, thereby improving the quality of life for everyone.”

With recent advances in microelectromechanical systems technology and the low price of manufacturing very small low powered sensing devices and deploying them on a large scale, smart cities are becoming technically and economically possible. Wireless sensor networks (WSNs) are equipped with computing, sensing, and wireless communication capabilities. Moreover, due to their small size and low cost, sensors are normally deployed in different environments, where they can collaborate among themselves to form a special sensor network. Since WSN may consist of tens, hundreds, or thousands of sensors, it helps to achieve fault tolerant,

reliable, and accurate sensor network covering a wider region [2].

Given these capabilities, WSN have recently stood out as a platform on which to deploy different kinds of important smart city applications [3, 4], such as power systems [5, 6], transportation and roadside applications [7, 8], healthcare [9, 10], and air pollution monitoring [11]. Figure 1 presents the various kinds of application using WSN deployment in smart cities.

However, wireless sensor nodes have limitations in their capabilities, such as small memory, limited energy, and low processing power [12, 13]. Sensors consume energy in sensing, data processing, and communicating without a permanent power supply along with the difficulty of replacing them. Communication is often the most energy-intensive. As a result, it is generally accepted that a WSN expires when its battery runs out, which affects its performance, quality, and lifetime.

Therefore, the lifetime of a network in critical applications such as power systems is decisive because the ability to save human lives as long as possible is very urgent. Due to the strict energy constraints, prolonging the lifetime of the sensor

FIGURE 1: Various applications of WSN in a smart city.

networks and reducing energy consumption are the most critical problem that must be considered when designing a WSN for a smart city.

If each and every sensor starts to communicate their data directly to the base station, the energy of the sensor nodes will be consumed quickly due to the long distances to the base station. In addition, congestion and collisions will consume more energy due to data retransmission. Clustering mechanism for the WSN is the approach used for solving these concerns. In clustering mechanism, nodes within a cluster send their data over shorter distances to their respective cluster heads, which aggregate sensed data from the surrounding general nodes into small sets of meaningful information and pass it on to the base station. Furthermore, aggregating and fusing data at the cluster head within a cluster helps to enucleate data redundancy.

Being a cluster head node means that it consumes more energy than general nodes, but this burden can be alleviated by selecting a few sensor nodes as a cluster head and rotating this role among the cluster members. The cluster heads can send data directly or use multihop to send data to the base station. Many protocols have been proposed to improve their energy consumption and the network lifetime, such as the Low-Energy Adaptive Clustering Hierarchy (LEACH) [14], Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [15], Hybrid Energy-Efficient Distributed Clustering (HEED) [16], Energy-Efficient Unequal Clustering (EEUC) [17], and the Fast Local Clustering Service (FLOC) [18]. However, the unsolved issue of energy consumption and hotspot [19] in the previous clustering protocols for WSN still exists.

In order to overcome several hotspot issues, we present and evaluate the EEUCLC, Energy-Efficient Unequal Chain Length Clustering protocol for the systemic gathering of data from the sensing environments where each node for every round has an information packet that should be transmitted to the base station. EEUCLC is a distributed competitive protocol where cluster heads are selected by localized competition. EEUCLC groups the networks into several unequal intracluster chain lengths and finds suboptimal multihop routes over the cluster heads to propagate the aggregation data to the base station. In EEUCLC protocol, the node with the highest residual energy and shortest distance to base station has the best chance to become the cluster head. Therefore

the nodes transmission radius decreases as its distance to base station decreases, which leads to creating smaller cluster sizes with shorter sensors chain length than those farther away from the base station. Hence, the accelerated consumed energy due to intracluster data processing and long distances to respective cluster heads will be reduced. EEUCLC also considers distance and residual energy to find suboptimal paths between cluster head nodes to the base station and uses these paths randomly for every data transmission occasion.

2. Related Works

Many clustering protocols [14–16, 18] and unequal clustering protocols [20–24] have been suggested to minimize the energy consumption of WSNs.

LEACH presented in [14] is a well-known clustering protocol for wireless sensor networks. LEACH is a distributed cluster formation without centralized control where each cluster head gathers data from all of the nodes to reduce global communication. LEACH rotates the cluster head role among all the nodes to distribute the energy load evenly. Also, an improved scheme of LEACH was proposed, named LEACH-C [25]. In LEACH-C, a centralized algorithm at the base station makes cluster formation.

In [15], Lindsey and Raghavendra proposed PEGASIS. PEGASIS is an enhancement of the LEACH protocol and makes a communication chain using a TSP (Traveling Sales Person) heuristic. Each node only communicates with two close neighbors along the communication chain. Only one node is chosen to send the data to the base station instead of multiple nodes. All nodes in the chain will take turns sending to the base station.

HEED presented in [16] is a multihop clustering protocol. It extends LEACH by considering range limits of the wireless communication and intracluster communication cost. It is considered as a distributed cluster formation where each node makes a decision based on local information. The probability for each node to become a tentative cluster head depends on its residual energy, and all the tentative heads are competing for becoming the final cluster heads. The final cluster heads are selected according to the intracluster communication cost.

In [17], Li et al. proposed EEUC, which is an Energy-Efficient Unequal Clustering protocol. It divides the nodes

into clusters of unequal size. The clusters closer to the base station have smaller sizes than those farther away from the base station. Thus cluster heads closer to the base station can preserve energy for the intercluster data forwarding.

FLOC [18] is a distributed clustering technique that uses the solid-disc property to minimize overlap between multiple clusters. A node communicates reliably with others within its unite distance defined as the inner-band (I-band) range and communicates unreliably (only a percentage of messages go through) with nodes within $m \leq 2$ distance defined as the outer band (O-band) range. FLOC is fast and scalable and achieves clustering in $O(1)$ time.

In [20], Ever et al. proposed UHEED which is an unequal clustering algorithm that tries to reduce energy consumption and avoid the early death of cluster heads closer to the base station by adjusting the completion radius of the HEED protocol based on the equation being used in EEUC.

In [21], Ren et al. proposed the Unequal Clustering Scheme based on LEACH for better energy balancing and extending network lifetime. The energy ratio and competition distance are used as two elements to join the cluster head selection. In this scheme, nodes can adjust their transmission power based on receiving a distance matrix from the base station.

EADUC [22], Energy-Aware Distributed Unequal Clustering, protocol is a distributed unequal clustering technique that supports both homogeneous and heterogeneous networks. The clustering formation is divided into three sub-phases: neighbor node information collection phase, cluster head competition phase, and cluster formation phase. Cluster heads with higher residual energy and being further away from the base station will have a larger cluster size.

UCA [23], location-based unequal clustering algorithm, is another unequal clustering algorithm where each cluster head has a different cluster size based on its distance to the base station. LUCA forms a cluster which has a smaller cluster size as the distance to the base station decreases.

In [24], Yang and Zhang proposed EB-UCP, Energy-Balancing Unequal Clustering Protocol, where the closer nodes to the base station have a higher probability of being selected as cluster heads. In EB-UCP, the network is split into layers, and the nodes in each layer have their own probability of cluster head selection.

3. Problem Statement

3.1. Accelerated Energy Consumption Problem. In a large-scale network, the relationship between network size and cluster size is proportional where the number of nodes in the cluster increases as the network size increases. It means that cluster heads bear a heavy burden and consume a considerable amount of energy. Using the single hop forwarding model within the cluster, where each node sends its data directly to their respective cluster head, leads to an increase in the number of transmitted messages to the cluster head from nodes within the cluster.

Congestion and collisions can also take place, which in turn will consume more energy due to data retransmission. Other problems arise if the cluster head is far away from the

nodes. In this case, some nodes will require a large amount of transmission power to convey their data; thus the energy of the nodes will be drained quickly.

Using multihop communication models between several cluster heads to forward the aggregated data to the base station is considered as a good solution for cluster heads that may not be able to communicate directly with the base station. However, using this method might be considered as another factor causing high energy consumption by the cluster head.

It happens in two different ways. First, if a cluster head is nearest to the base station it will act as a router for other nodes and is burdened with heavy relay traffic. Second, the best single route between a cluster head node to the base station will be found based on cluster head residual energy and the energy of the transmitter. This route is used for every communication. This will drain the energy of the cluster heads along this path. Therefore, the fundamental goal is to overcome these hot spot problems.

The protocol considers the periodic selection of cluster heads based on the nodes residual energy and distance to the base station. Furthermore, we present unequal chain clustering and a suboptimal multihop routing protocol to alleviate the burden on the cluster heads and prolong the lifetime of a sensor network.

3.2. Network and Radio Models. We consider a WSN of hundreds or thousands of N energy-constrained sensor nodes, which are uniformly deployed over a sensing environment. We denote node i by s_i and the identical sensor node set $S = \{s_1, s_2, \dots, s_n\}$, where $1 \leq i \leq |S|$ and $|S| = N$. In this work, our model sensor network is based on the following assumptions:

- (i) The base station is fixed and situated at far distance from the sensor nodes.
- (ii) Links between the nodes are symmetric.
- (iii) Nodes in the network are homogeneous.
- (iv) Energy is constrained.
- (v) The network is location aware, that is, equipped with a GPS capable antenna.
- (vi) The nodes are able to utilize power control when transmitting data.
- (vii) The nodes are not mobile.

We adapt the same radio model described in Heinzelman et al. [26]. In this radio model, a radio dissipated energy to run the transmitter and transmitter amplifier. In addition, the radio dissipated energy to run the receiver circuitry. Both the free space model (d^2 power loss) and the multipath fading (d^4 power loss) channel models are used, where the dissipation energy to run the transmitter amplifier depends on the distance between the transmitter and receiver. In this radio model, the energy dissipated to transmit 1-bit message for a distance d is computed as follows.

FIGURE 2: Flowchart of EEUCLC.

Transmitting

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

$$= \begin{cases} lE_{elec} + E\epsilon_{fs}d^2, & d < d_0, \\ lE_{elec} + E\epsilon_{mp}d^4, & d \geq d_0. \end{cases} \quad (1)$$

Receiving

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

4. Proposed Algorithm

Figure 2 shows the flow chart and Figure 3 shows a general overview of the EEUCLC protocol, respectively. The EEUCLC protocol consists of three phases: (1) cluster head selection, (2) chain formation, and (3) data collection and transmission. In Figure 3, the circles of unequal size represent clusters of unequal chain length and the routes among the cluster heads clarify the multihop forwarding model.

4.1. Cluster Head Selection Phase. When clusters are being created, several candidate cluster heads are selected to compete to become final cluster heads in the current round. Periodic reselection within the cluster is performed to balance the energy consumption of each cluster.

For candidate cluster head selection, EEUCLC makes a tradeoff between the nodes residual energy and distance to the base station. A node having low residual energy with a long distance to the base station will not be selected to compete for the final cluster head. Before a node starts carrying out EEUCLC, it sets its probability to become a candidate cluster head, CCH_{prob} , as follows:

$$CCH_{prob} = a \cdot \frac{(d_{max} - d(s_i, BS))}{(d_{max} - d_{min})} + (1 - a) \cdot \frac{E_{residual}}{E_{max}}, \quad (3)$$

where $E_{residual}$ denotes the estimated current residual energy; E_{max} is the maximum energy, which is congruous for all nodes; d_{max} and d_{min} denote the maximum and minimum distance between sensor nodes and the base station, respectively. $d(s_i, BS)$ is the distance between s_i node and the base station, and a is a constant coefficient.

FIGURE 3: An overview of the EEUCLC technique (BS = base station).

For a node to become a candidate cluster head, the value of CCH_{prob} should be larger or equal to a certain threshold G . Nodes below this certain threshold value keep sleeping until the cluster head selection phase terminates. The purpose of using CCH_{prob} is to limit the initial cluster head announcement and prevent nodes far away from the base station with low residual energy from consuming more energy due to an additional packet transmission overhead while competing to become the final cluster head.

After the candidate cluster heads have been selected, competition starts for the final cluster head phase. A candidate cluster head only becomes a final cluster head if it has greater residual energy than the other nodes in its competition radius. In order to create unequal intracluster chain lengths, we need to control the cluster size. In EEUCLC, intracluster chains closer to the base station have a shorter length than those farther away.

To achieve our objective, the cluster head competition radius should decrease as its distance to the base station decreases. To compute the cluster head competition radius, we use the same equation being used in EEUC:

$$s_i \cdot R_{comp} = 1 - c \frac{(d_{max} - d(s_i, BS))}{(d_{max} - d_{min})} R_{comp}^0, \quad (4)$$

where c is a constant between 0 and 1, and R_{comp}^0 is the initial cluster range used in remote areas from the base station.

Figure 4 shows an uneven competition radius of candidate cluster heads where candidate cluster heads S_3 and S_4 cannot both be final cluster heads because they are in the competition radius of each other, while the opposite occurs for S_1 and S_2 .

Competition for final cluster head selection messages is broadcast with R_{comp}^0 . In lines 5–29 of Algorithm 1, competition for final cluster heads starts when each candidate cluster head broadcasts a *CompeteChainHead_MSG* containing residual energy and competition radius and sets a random wait time.

If no *CompeteChainHead_MSG* is received within a random wait time, the candidate head node broadcasts

```

(1) if  $CCH_{prob} \geq G$  then
(2)    $beCandidateHead \leftarrow True$ 
(3) end if
(4) if  $beCandidateHead = True$  then
(5)    $CompeteChainHead\_Msg(ID, R_{comp}, E_{residual})$ 
(6) else
(7)   EXIT;
(8) end if
(9) On receiving a  $CompeteChainHead\_MSG$  from node  $s_j$ 
(10) if  $d(s_i, s_j) < s_j.R_{comp}$  OR  $d(s_i, s_j) < s_i.R_{comp}$  then
(11)   Add  $s_j$  to  $s_i$  cluster group
(12) end if
(13) while  $beCandidateHead = True$  do
(14)   if  $s_i.E_{residual} > s_j.E_{residual}, \forall s_j \in s_i.Clustergroup$  then
(15)      $FinalChainHead\_Msg(ID)$ 
(16)     Exit
(17)   end if
(18)   On receiving  $FinalChainHead\_MSG$  from node  $s_j$ 
(19)   if  $s_j \in s_i.Clustergroup$  then
(20)      $Resignation\_MSG(ID)$  and then EXIT
(21)   end if
(22)   On receiving a  $Resignation\_MSG$  from node  $s_j$ 
(23)   if  $s_j \in s_i.Clustergroup$  then
(24)     Remove  $s_j$  from  $s_i$  cluster group
(25)   end if
(26) end while
(27)  $ChainHead\_ADV\_MSG(ID).Send()$ 
(28) On not receiving a  $CompeteChainHead\_MSG$  from node  $s_j$ 
(29)  $ChainHead\_ADV\_MSG(ID).Send()$ 

```

ALGORITHM 1: Cluster head selection.

$ChainHead_ADV_MSG$ containing $HeadNode.id$. In the case where the candidate head node receives $CompeteChainHead_MSG$ from another candidate node, if $d(s_i, s_j) < s_j.R_{comp}$ or $d(s_i, s_j) < s_i.R_{comp}$, then they become part of one cluster group.

Each candidate cluster head in the cluster group checks the residual energy of the others. The candidate head node that has the highest residual energy sends $FinalChainHead_MSG$ to inform its neighboring candidate cluster heads. If two candidate head nodes have the same residual energy, then the candidate node with the smaller ID sends $FinalChainHead_MSG$.

If candidate head node s_j is located in the competition range of candidate head node s_i and receives $FinalChainHead_MSG$ from s_i , s_j will give up and send $Resignation_MSG$ which contains the IDs to all candidate head nodes in its competition range, as shown in Figure 5.

If candidate head node s_i receives $Resignation_MSG$ from candidate head node s_j and s_j belongs to competition range of s_i , s_i will remove s_j from its cluster group, as shown in Figure 6.

After the cluster head chain has been selected, sleeping nodes wake up and each final cluster head broadcasts a $ChainHead_ADV_MSG$ containing $HeadNode.id$. Eventually, each normal node selects the cluster to which it will belong

FIGURE 4: Competition radius among candidate cluster heads.

for this round (this decision is based on the received signal strength).

The larger the received signal strength is the closer the cluster head will be selected, thus, with minimum transmission power required for normal nodes to communicate with cluster head. This is carried out by normal nodes sending $ChainMember_MSG$ which contains ID.

4.2. Chain Construction Phase. In each cluster group, chain construction starts with the cluster head node creating a

FIGURE 5: Node s_j receives *FinalChainHead_MSG*. FIGURE 6: Node s_i receives *Resignation_MSG*.

routing table and sending it to the closest node, which then joins the chain first. The routing table contains a list of all of the nodes belonging to the cluster and the target node S_{FS} (the node at the farthest distance from the cluster head). The list of nodes is ranked based on how near they are to the respective cluster head.

During chain construction, each intermediate node forwards the request only to the neighbors that are farther away from the cluster head node than itself and have the minimum distance between each other; thus, at node s_i , the request is only sent to neighbor s_j , which satisfies $d(s_i, CH_s) \leq \min(d(s_j, CH_s))$, where $d(s_i, CH_s)$ denotes the distance between node s_i and its respective cluster head node (CH_s), and s_i and s_j are the intermediate nodes between S_{FS} and CH_s , respectively.

In lines 1–15 of Algorithm 2, when a node receives a request packet, it checks whether it is the target node or not. If it is not the target (the farthest node from respective cluster head), it sends the request to a node that has not yet joined the chain and has the smallest distance to the cluster head.

During the chain construction process, if two nodes in the same cluster group cannot reach each other, a node can change its transmission power. The chain construction process terminates when the target node has been discovered.

Figure 7 shows the chain construction steps and data transmission between the nodes along the chain. After the chain construction process terminates and suboptimal multihop routes over the cluster heads have been discovered, the cluster head gathers sensing data within the cluster where each node receives data from a neighbor. The cluster head also fuses them with its own data, forwards them to another neighbor along the chain, and so on until the cluster head receives the fused data.

EEUCLC performs data fusion at each node except node S_{FS} located at the end of chain. Figure 8 shows a flow chart of cluster formation.

4.3. Intercluster Multihop Routing. In order to minimize the overhead, we have to ensure that the time duration of the data transmission phase is longer than the time taken by

the clustering protocol to cluster the network. We utilize the advantage of the long data transmission phase to build an energy-aware intercluster multihop routing protocol by considering the residual energy of and distance to the neighboring cluster head nodes.

Since the transmission phase will be long enough to send as much data as possible to the base station, we want to ensure that there is more than one route between the cluster head nodes and the base station used for communication. Frequently using the same route between cluster head nodes leads to depleting their energy levels along the path. It leads to vast variation in the energy levels of the nodes and finally causes network partitioning. Hence, we have designed an energy-aware multihop routing protocol for the intercluster communication.

Using EEUCLC, we try to find suboptimal routes between cluster head nodes to the base station and use these routes randomly for each data transmission. In lines 1–13 of Algorithm 3, after cluster head nodes have been identified for the current round, each s_i broadcasts a message across the networks at a specific power level containing residual energy, distance to the base station, and *HeadNode.id*.

We also present a threshold Tx_max into the proposed intercluster multihop routing protocol so that if the cluster head distance to the base station is smaller than or equal to Tx_max , it directly transmits its data to the base station. When the cluster head node s_j receives broadcasted packets, it checks the distance. If a node s_j has a smaller distance to the base station than node s_i , node s_j discards the received packet. Otherwise, the node s_j selects suboptimal cluster heads to forward its data from the cluster head node candidates.

The suboptimal cluster head nodes list is defined as $s_j.SCH = \{HE_{residual}(\min(d(s_j, s_i)))\}$, where SCH is defined as a set of suboptimal neighboring cluster head nodes which have higher residual energy and require the minimum transmission power for communication, $HE_{residual}$ denotes the higher residual energy of group cluster head nodes, and $\min(d(s_j, s_i))$ denote the minimum distance between the group of head nodes.

After each cluster head has chosen relay nodes, the former randomly selects nodes for each data transmission occasion from the candidate cluster headset to relay data to the base station. Figure 9 shows the suboptimal routes construction steps and data transmission between cluster heads.

5. Protocol Analysis

This section shows the analysis of the protocol described in Algorithms 1 and 2. Since the cluster head selection process is by message passing, we first discuss its message complexity.

Lemma 1. *The message complexity of the cluster formation algorithm is $O(N)$.*

Proof. At the beginning of the cluster formation, $N \times G$ candidate cluster heads are selected and each one of them broadcasts *CompeteChainHead_MSG*. Then each of them makes decision to be cluster head or not by broadcasting *FinalChainHead_MSG* or *Resignation_MSG*. Suppose k cluster

FIGURE 7: Chain construction and data transmission between nodes: (a) the closest node to the cluster head joining the chain first, (b) the intermediate nodes joining the chain, (c) termination of chain construction, and (d) relaying data to the designated cluster head.

heads are selected, they send out k *ChainHead_ADV_MSG*, and then $(N - k)$ ordinary nodes transmit $(N - k)$ *ChainMember_MSG*. After that $N - 1$ nodes send *Chain_construction_MSG*. Thus the messages are up to $2N \times G + k + N - k + N - 1 = 2N(G + 1) - 1$ for cluster formation stage per round. Lemma 1 shows the message overhead of EEUCLC is small. $\square$

Lemma 2. *The time complexity for the cluster head selection algorithm is $O(N)$.*

Proof. EEUCLC protocol takes a processing time of at most $2N \times G$ among candidate cluster heads which declared their willingness to be cluster heads. This happens if more than one candidate cluster head is in the range of competition radius of each other. For the opposite, it takes at most $N \times G$ time. In both cases for sleeping nodes to join cluster it takes at most

$k - N$ processing time. Therefore, the total processing complexity is $O(N)$. $\square$

6. Simulation Results

We evaluated the performance of the proposed EEUCLC protocol using Matlab Simulation tools [27] to perform several extensive computing experiments. We also compared our simulation results with LEACH, improved LEACH, and EEUC, in terms of energy consumption by the cluster heads, network lifetime, and the number of nodes alive.

For our experiments, we simulated an environment where sensor nodes were uniformly deployed in the sensing field and the base station was located outside the WSNs. All sensor nodes periodically sensed events and forwarded data packets to the base station. The energy model adopted in our simulation is the same as that used in Heinzelman et al. [26].

FIGURE 8: An overview of the EEUCLC cluster formation.

The values of the simulation parameters are given in Table 1. We set a to 0.5 in (3), R_{comp}^0 to 90 m, c to 0.5 in (4), and Tx_{max} to 130 m. For each simulation scenario, the results are reported as the average value of 20 runs.

6.1. Energy Efficiency. In this section, we examine the energy efficiency of EEUCLC. Figure 10 presents the amount of energy spent by cluster heads derived from the EEUCLC, LEACH, improved LEACH, and EEUC schemes. The x -axis indicates 20 sampled rounds of simulation while the y -axis indicates the amount of energy spent by the cluster heads.

The results show that the energy consumed by cluster heads per round using EEUCLC was much smaller than other schemes. This is because cluster heads of LEACH and improved LEACH communicated their data directly to the base station, which consumed more energy, and because the

cluster heads using LEACH were not uniformly distributed throughout the network due to the random selection of cluster heads.

Furthermore, in EEUC, the energy consumption of the cluster heads is higher than EEUCLC because EEUC tried to find the best single route between cluster head nodes and the base station and sent all aggregated data along this path, which drained the energy of the cluster heads along this path. In EEUCLC, the cluster heads transmitted their data to the base station via multihop and tried to find suboptimal routes between the cluster head nodes and the base station and transmitted data through the randomly selected suboptimal routes; thus a considerable amount of energy was saved.

We also investigated the energy efficiency of the proposed protocol by examining the network lifetime. Figure 11 shows

FIGURE 9: Suboptimal multihop routing: (a) message propagation, (b) the distance from the cluster head (CH) to the base station smaller than the threshold distance, (c) suboptimal routes selection, and (d) and (e) random selection for the relaying route.

FIGURE 10: The amount of energy spent by cluster heads.

FIGURE 11: The time until the first node dies.

```

(1) On receiving ChainHead_ADV_MSG from cluster head
(2) ChainMember_Msg(ID)
(3) if cluster_head = True then
(4)   previous_node  $\leftarrow$  CH
(5)   for each node  $s_i$  in cluster do
(6)     if  $s_i \neq S_{FS}$  then
(7)       if  $d(s_i, CH_s) \leq \min(d(s_j, CH_s))$  then
(8)         previous_node.SendMessage(si)
(9)         previous_node =  $s_i$ 
(10)      end if
(11)    else
(12)      EXIT;
(13)    end if
(14)  end for
(15) end if

```

ALGORITHM 2: Chain construction phase.

```

(1)  $s_i.Broadcast\_MSG(E_{residual}, d.BS)$ 
(2) Chain_Member_Msg(ID)
(3) if  $d.BS < Tx_{max}$  then
(4)   message.send(BS)
(5) end if
(6) for each  $s_j$  do
(7)   On receiving Broadcast_MSG(Eresidual, d.BS)
       from  $s_j$ 
(8)   if  $d(s_j, BS) > d(s_i, BS)$  then
(9)      $A_r \leftarrow s_j.select(\min(d(s_i, s_j)))$ 
(10)     $A_r \leftarrow select(\max(E_{residual}, A_r))$ 
(11)     $s_j.SendNext(A_r.random())$ 
(12)  end if
(13) end for

```

ALGORITHM 3: Intercluster multihop routing.

TABLE 1: Simulation parameter.

Parameter	Value
Network coverage	(0, 0)–(200, 200) m
Basestation location	(100, 250) m
Initial energy	1 J
N	150
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/bit/m ²
ϵ_{mp}	0.0013 pJ/bit/m ⁴
d_o	87 m
Data packet size	2000 bits
E_{DA}	5 nJ/bit/signal

the time until the first node died, while Figure 12 shows the time until the last node died.

EEUCLC clearly improved not only the time until the first node died, but also the time until the last node died compared to other schemes. With a sensing field of size 200 m $\times$ 200 m, the graphs show that sensor nodes in EEUC, improved LEACH, and LEACH nodes died earlier compared to EEUCLC because their sensor nodes became thinly scattered and thus consumed more energy to convey their data over the long distance to their respective cluster.

Moreover, some cluster head nodes in improved LEACH and LEACH consumed more energy to convey their data directly over a long distance to the base station. Since cluster heads consume energy based on the number of nodes within a cluster, the cluster heads in EEUC with low energy located far away base station died earlier. This is because EEUC is an unequal clustering method, where clusters far away from base station have a larger size than clusters near the base station. EEUCLC overcame these issues by convening data to the respective cluster head using chain schema, and so nodes having low residual energy at a long distance from the base station had a low probability when competing for the final cluster head.

FIGURE 12: The time until the last node dies.

FIGURE 13: The variance of amount of energy spent by cluster heads.

Figure 13 displays the variance of energy consumption by cluster heads in 20 sampled rounds. It shows that EEUCLC balanced the energy consumption between cluster heads better than other schemes. This is because the cluster heads distribution in LEACH and improved LEACH was not controllable; thus the number of nodes within clusters varied too much, while, in EEUC, some cluster heads relayed all aggregated data received by multiple cluster heads. Hence, some cluster heads consumed more energy than others. EEUCLC avoided the hot spot problem by utilizing its suboptimal multihop routing mechanism.

Figure 14 exhibits the number of rounds until 1%, 25%, 50%, and 100% of nodes died in a $200\text{ m} \times 200\text{ m}$ field. In this field size, sensor nodes started to die at a more rate after about 20% of the nodes had died. This was due to the distances between the nodes which became larger, and nodes had to become leaders more often, causing the energy to deplete quickly. EEUCLC achieved approximately 1x the number of

FIGURE 14: Performance results for a $200\text{ m} \times 200\text{ m}$ network with initial energy 1 J/node .

rounds compared to EEUC when 1%, 20%, 50%, and 100% of nodes died for a $200\text{ m} \times 200\text{ m}$ network, while compared to improved LEACH achieving approximately 2x the number of rounds. Last, compared to LEACH, EEUCLC achieved approximately 3x the number of rounds.

7. Conclusion

In this paper, we proposed EEUCLS, a novel multiple chain construction protocol capable of data aggregation and conveying data to a base station for WSNs. The proposed protocol provides balancing the energy consumption among the cluster heads, preserving energy consumption of the nodes within the cluster, and reducing accelerated energy consumption of cluster heads. To support multihop routing in EEUCLS, we presented an unequal chain length clustering and a suboptimal multihop routing protocol to alleviate the burden on the cluster heads and prolong the lifetime of the sensor network.

In addition EEUCLS can also work in a very variable sensor networks domain, for example, integrated navigation system with various sensors (GNSS, GPS, and Speed Log) because it can secure the availability, consistency, and synchronization of the information from the various sensors. Simulation results show that our unequal clustering protocol clearly improved the network lifetime and balanced the energy consumption by the cluster heads compared to LEACH, EEUC, and improved LEACH.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This material is based upon work supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program no. 10063496,

“Integrated Navigation System Development Based on Virtual Platform.”

References

- [1] T. Chen, “Smart grids, smart cities need better networks,” *IEEE Network*, vol. 24, no. 2, pp. 2–3, 2010.
- [2] A. Manjeshwar and D. P. Agrawal, “TEEN: a routing protocol for enhanced efficiency in wireless sensor networks,” in *Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS '01)*, vol. 1, pp. 2009–2015, IEEE, San Francisco, Calif, USA, 2001.
- [3] J. Huang, Q. Duan, C.-C. Xing, and H. Wang, “Topology control for building a large-scale and energy-efficient internet of things,” *IEEE Wireless Communications*, vol. 24, no. 1, pp. 67–73, 2017.
- [4] J. Huang, Y. Meng, X. Gong, Y. Liu, and Q. Duan, “A novel deployment scheme for green internet of things,” *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 196–205, 2014.
- [5] Y. Lim, H.-M. Kim, and S. Kang, “A design of wireless sensor networks for a power quality monitoring system,” *Sensors*, vol. 10, no. 11, pp. 9712–9725, 2010.
- [6] J. Ahmad, A. S. Malik, M. F. Abdullah, N. Kamel, and L. Xia, “A novel method for vegetation encroachment monitoring of transmission lines using a single 2D camera,” *Pattern Analysis and Applications*, vol. 18, no. 2, pp. 419–440, 2015.
- [7] D. Curia and C. Volosencu, “Urban traffic control system architecture based on wireless sensor/actuator networks,” in *Proceedings of the 2nd International Conference on Manufacturing Engineering, Quality and Production Systems (MEQAPS '10)*, Constantza, Romania, 2010.
- [8] M. Ceriotti, M. Corrà, L. D’Orazio et al., “Is there light at the ends of the tunnel? Wireless sensor networks for adaptive lighting in road tunnels,” in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN '11)*, pp. 187–198, Chicago, Ill, USA, April 2011.
- [9] Y.-F. Chung and C.-H. Liu, “Design of a wireless sensor network platform for tele-homecare,” *Sensors (Switzerland)*, vol. 13, no. 12, pp. 7156–7175, 2013.
- [10] J. Ko, T. Gao, and A. Terzis, “Empirical study of a medical sensor application in an urban emergency department,” in *Proceedings of the 4th International ICST Conference on Body Area Networks (BODYNETS '09)*, Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, April 2009.
- [11] X. Mao, X. Miao, Y. He, X.-Y. Li, and Y. Liu, “CitySee: urban CO₂ monitoring with sensors,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 1611–1619, Orlando, Fla, USA, March 2012.
- [12] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [13] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, 2004.
- [14] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS '00)*, p. 10, January 2000.
- [15] S. Lindsey and C. S. Raghavendra, “PEGASIS: power-efficient gathering in sensor information systems,” in *Proceedings of the IEEE Aerospace Conference*, vol. 3, pp. 1125–1130, Big Sky, Mont, USA, March 2002.
- [16] O. Younis and S. Fahmy, “HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.
- [17] C. F. Li, M. Ye, and G. Chen, “An energy-efficient unequal clustering mechanism for wireless sensor networks,” in *Proceedings of the 2nd IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS '05)*, Washington, DC, USA, 2005.
- [18] M. Demirbas, A. Arora, V. Mittal, and V. Kulathumani, “A fault-local self-stabilizing clustering service for wireless ad hoc networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 9, pp. 912–922, 2006.
- [19] R. C. Shah and J. M. Rabaey, “Energy aware routing for low energy ad hoc sensor networks,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '02)*, vol. 1, pp. 350–355, IEEE, March 2002.
- [20] E. Ever, R. Luchmun, L. Mostarda, A. Navarra, and P. Shah, “UHEED - An Unequal Clustering Algorithm for Wireless Sensor Networks,” in *Sensornets*, Rome, Italy, February 2012.
- [21] P. Ren, J. Qian, L. Li, Z. Zhao, and X. Li, “Unequal clustering scheme based LEACH for wireless sensor networks,” in *Proceedings of the 4th International Conference on Genetic and Evolutionary Computing, ICGEC 2010*, pp. 90–93, Chn, December 2010.
- [22] J. Yu, Y. Qi, G. Wang, Q. Guo, and X. Gu, “An energy-aware distributed unequal clustering protocol for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 7, no. 1, Article ID 202145, 2011.
- [23] S. Lee, H. Choe, B. Park, Y. Song, and C.-K. Kim, “LUCA: an energy-efficient unequal clustering algorithm using location information for wireless sensor networks,” *Wireless Personal Communications*, vol. 56, no. 4, pp. 715–731, 2011.
- [24] J. Yang and D. Zhang, “An energy-balancing unequal clustering protocol for wireless sensor networks,” *Information Technology Journal*, vol. 8, no. 1, pp. 57–63, 2009.
- [25] W. B. Heinzelman, *Application specific protocol architectures for wireless networks [Ph.D. dissertation]*, Massachusetts Institute of Technology, Cambridge, Mass, USA, 2000, <https://pdfs.semanticscholar.org/3b86/506f0562c1453461509c5dc43359fe-908db5.pdf>.
- [26] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [27] The matworks matlab and simulink for technical computer: <http://www.mathworks.com/>.