ECDC: An energy and coverage-aware distributed clustering protocol for wireless sensor networks

Xin Gu, Jiguo Yu, Dongxiao Yu, Guanghui Wang, Yuhua Lv

School of Computer Science, Qufu Normal University, Rizhao, Shandong 276826, China
Key Laboratory for Intelligent Control Technique of Shandong Province, Rizhao, Shandong 276826, China
Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong, China
School of Mathematics and System Science, Shandong University, Jinan, Shandong 250100, China

Abstract

Clustering for wireless sensor networks (WSNs) is an effective scheme in utilizing sensor nodes energy and extending the network lifetime, while coverage preservation is one of the most essential issues to guarantee the quality of service (QoS). However, the coverage problem has not been well understood so far. For mission-critical applications of networks, it is crucial to consider coverage requirements when we select cluster heads and routing nodes for the clustering topology. In this paper, we propose the ECDC (Energy and Coverage-aware Distributed Clustering Protocol), an integrated protocol involving both energy and coverage, which is different from previous clustering protocols. For different practical applications, we design corresponding coverage importance metrics and introduce them into the clustering algorithm. Theoretical analysis and simulation results show that our protocol is effective in improving network coverage performance, reducing nodes energy dissipation and extending the network lifetime.

1. Introduction

A wireless sensor network consists of plentiful low-power sensor nodes capable of sensing, processing and communicating. These sensor nodes observe the phenomenon of the monitored field, and send the measured data to the base station (BS). As sensor networks have limited and non-rechargeable energy resources, energy efficiency is a very critical issue in designing the network topology, which affects the lifetime of sensor networks greatly. Thus, it is a central concern how to minimize energy dissipation and maximize network lifetime when designing protocols for WSNs.

In recent years, clustering has been proved to be an important factor of decreasing the energy dissipation and extending the lifetime of WSNs. In clustering schemes, sensor nodes are grouped into clusters. In each cluster, a node is selected as the leader named as the cluster head, while other nodes are called cluster members [1]. Each cluster member measures physical variables related to its environment and then sends them to the cluster head. When the data from all cluster members is collected, cluster heads aggregate the data and send it back to the BS. During this process, cluster heads takes heavier traffic load. Consequently, the energy dissipation of cluster heads is much higher than that of cluster members. Therefore, to extend the lifetime of WSNs, it is critical to select the optimal cluster heads and routers for the clustering topology.

Furthermore, in order to meet the quality of service (QoS), many surveillance applications (main tasks of which are target monitoring, information acquisition and effective transmission) require the ability of provide information on the spot from...
each part of the monitored area. So it attracts great attention to design coverage-preservation protocols, whose performance is usually measured by the coverage ratio of a network. Due to the fact that the energy resource of WSNs is limited, a key problem of these algorithms is how to utilize the energy of sensor nodes efficiently while achieving better coverage performance.

Though both clustering topology construction and coverage-preservation protocols have been extensively studied separately, only few protocols considered them in a joint way. Most of existing clustering algorithms for the selection of cluster heads care about various dimensions: residual energy, deployment of nodes, node degree or communication cost between nodes, and so on. However, most of these approaches are only for reducing or balancing the node energy dissipation, while how to cover the surveillance area effectively is out of the scope. These algorithms are unable to persistently provide effective coverage for the monitored targets. In contrast, in this paper, we propose ECDC, an integrated protocol involving both energy efficiency and coverage importance metrics. In our algorithm, different coverage importance metrics are designed for different practical applications. We select cluster heads based on the relative residual energy and the coverage importance metrics of nodes. The inter-cluster communication adopts multi-hop forwarding mechanism. When choosing the forwarding node, the cluster head considers both the residual energy and the coverage importance of nodes. Compared with previous protocols, our algorithm can construct a better clustering topology through less control information which has lower energy dissipation and is shown a better coverage performance.

The rest of the paper is organized as follows. The related work of this field is introduced in Section 2. Section 3 exhibits the network model and Section 4 proposes the coverage importance metrics \( CI_p \) and \( CI_a \) for the point coverage problem and area coverage problem, respectively. Section 5 presents the ECDC in details. Section 6 analyzes the properties of the algorithm. In Section 7 we detail our simulation efforts and the analysis of results. Finally, Section 8 concludes the paper.

2. Relate work

Previously, much work has been done on improving energy efficiency and coverage-preservation protocols. However, there are only a few clustering protocols considering both energy and coverage in a joint way. In this section, we list and analyze some typical clustering protocols, and some other clustering protocols involving coverage preservation.

LEACH [1] is a typical clustering protocol proposed for periodical data gathering applications in WSNs. In LEACH, each node elects itself as a cluster head with a probability. Cluster heads are responsible for receiving and aggregating data from cluster members, and then send the aggregated data back to the BS by single-hop communication. In order to balance the energy consumption, the role of cluster head is periodically rotated among all the nodes. LEACH is simple and does not require large communication overhead. However, because LEACH elects cluster heads without considering the residual energy of nodes, the performance of LEACH in heterogeneous networks is not very well.

Energy-efficient distributed clustering algorithm HEED was proposed in [2]. It selects the optimal set of cluster heads based on the remaining energy of nodes and their communication costs. At the beginning of the clustering phase, nodes with higher residual energy have higher probability to be selected as tentative cluster heads. Once a node becomes a tentative cluster head, it broadcasts a status message to all sensor nodes within its cluster competition radius. Other nodes that hear from multiple tentative cluster head choose their cluster heads based on the communication cost of the tentative cluster heads.

A distributed energy saving clustering algorithm called BPEC was proposed in [3]. BPEC selects cluster heads by a primary probability which is defined as the ratio between the average residual energy of neighbor nodes and the residual energy of the node itself, and a subsidiary probability that is the degree of the node.

EADDEEG [4] is a novel distributed clustering algorithm, which elects cluster heads based on the ratio between the average residual energy of neighbor nodes and the residual energy of the node itself. By doing this, EADDEEG achieves a good cluster heads distribution and prolongs the network lifetime.

EECS [5], EEUC [6], EDUC [7] and EADUC [8] are energy efficient clustering schemes by dynamic sizing of clusters based on the distance from cluster heads to the BS. Cluster heads closer to the BS require more energy for data forwarding. These protocols use uneven competition ranges to construct clusters of uneven sizes. Clusters closer to the BS have smaller sizes than those farther away from it, thus cluster heads closer to the BS can preserve some energy for the inter-cluster data forwarding. Consequently, the distribution of energy throughout the network is improved and network lifetime is extended.

In [9], Yu et al. proposed a cluster-based routing protocol for wireless sensor networks with nonuniform node distribution, whose cores are an energy-aware clustering algorithm and a cluster-based routing algorithm. The protocol can achieve the balance of energy dissipation among nodes and prolong the network lifetime.

However, focuses of these algorithms are to reduce or balance the energy consumption of nodes such that the network lifetime can be extended, but the coverage requirement of practical applications is not guaranteed. For example, when some important nodes for network coverage are elected to be cluster heads or routers, their energy will be consumed quickly. As a result, the “coverage holes” of the network will soon appear along with the death of these important nodes, which causes the inaccuracy of surveillance data and makes the algorithms unsuitable for mission-critical applications.

To satisfy the coverage demands in some practical applications, a coverage-preserving routing protocol LEACH-Coverage-U was proposed in [10]. Different from previous protocols, LEACH-Coverage-U protocol firstly calculates the overlap sensing areas of all sensor nodes, and then selects cluster heads from nodes deployed in a highly overlapped area. The simulation
results show that the LEACH-Coverage-U protocol could prolong the network lifetime better than existing protocols. Noh et al. proposed a novel coverage-preserving scheme (BCoPS) in [11]. This scheme is based on cluster formation that allows the BS to maintain the network with consideration of network coverage.

In [12], Soro and Heinzelman proposed the cluster-based network construction algorithms CPCP based on a set of coverage-aware cost metrics. Authors explored different coverage-aware cost metrics that favor nodes deployed in densely populated network areas as better candidates for cluster heads, active sensor nodes and routing nodes. However, it requires the location of nodes and these cost metrics, which causes a large computational burden on sensor nodes. In [13], Wang et al. presented the coverage-aware clustering protocol CACP for randomly deployed networks simplifying the cost metric for selection of cluster heads and active nodes. But in CACP, each cluster head consumes much more energy, since they directly transmit the aggregated data to the BS, which leads to the quick death of cluster heads.

Multi-hop transmission is generally more efficient in reducing energy consumption than the single-hop transmission [14]. Thus, the multi-hop transmission, i.e., a cluster head sends the aggregated data from its members to the sink via multiple intermediate cluster heads, has generally been considered as an efficient energy-saving approach for large scale sensor networks, and has been widely used in inter-cluster communication in previous research [13,15–17]. However, most multi-hop transmission approaches are based on a tree topology. The traffic flow passing through the sensors may be unbalanced, and some nodes which are important for the coverage may die prematurely.

In order to deal with the problems incurred in above algorithms, we propose ECDC (Energy and Coverage-aware Distributed Clustering Protocol) in this paper. In our algorithm, corresponding coverage importance cost metrics CI aiming at different application coverage problems are introduced. Only nodes with higher energy and smaller coverage importance are reliable to be selected as cluster heads and routers.

3. Network model

In the network, $N$ sensor nodes are distributed in an $M \times M$ square field, each of which has a unique identity $id$. All nodes and the BS are stationary after deployment. Nodes are energy heterogeneous. Power control are assumed to vary the amount of transmit power. The elected cluster heads can communicate with the BS directly. The BS is out of the sensor field, and it has enough energy. The location of the BS is assumed to be known by each node.

The expended radio energy for transmitting $l$-bit data is as follows.

$$E_{tx}(l, d) = \begin{cases} l \times E_{elec} + l \times \epsilon_{fs} \times d^2, & d < d_0 \\ l \times E_{elec} + l \times \epsilon_{mp} \times d^4, & d \geq d_0 \end{cases}$$

where $d$ is the transmission distance. $E_{elec}$, $\epsilon_{fs}$ and $\epsilon_{mp}$ are parameters of the transmission/reception circuit. According to the distance between the transmitter and receiver, free space($\epsilon_{fs}$) or multi-path fading($\epsilon_{mp}$) channel model is used.

The expended radio energy for receiving an $l$-bit data is as follows.

$$E_{rx}(l) = l \times E_{elec}.$$ 

Additionally, we assume that the energy for data aggregation is $E_{com}$, and the energy for data sensing is $E_{sen}$.

4. Coverage importance metrics

DARP [18] is the first routing protocol designed to avoid routing of data through areas sparsely covered by sensor nodes. The idea behind this approach is that, nodes deployed in sparsely areas are less selected as routing nodes such that they can collect data for longer periods of time. To explore the benefit of this approach in cluster-based sensor networks, based on the existing coverage problems (point coverage problems and area coverage problems) in practical applications, we introduce two corresponding coverage importance metrics for them. Details about these two problems and metrics are described as follows.

4.1. Point coverage

In the point coverage problem, the monitoring targets of network are a series of discrete interest points of surveillance area. Fig. 1 shows a case of point coverage. The black points denote the interest points for monitoring, and the red points represent the sensor nodes performing monitoring tasks. The blue circle expresses the sensing region of each sensor node. The sensing radius is set to $r_s$, and the communication radius is set to $r_c$. $N$ is the number of all nodes in the network.

**Definition 1.** If the Euclidean distance between the node $s_i$ and the interest point $p_j$ is not greater than $r_s$, then $p_j$ is covered by $s_i$. If the Euclidean distance of the node $s_i$ and $s_j$ is not greater than $r_c$, then $s_j$ is the neighbor node of $s_i$.

$C(s_i)$ is defined as the set of interest points covered by $s_i$. As shown in Fig. 1, the coverage set of node $s_i$ is as follows.
As stated above, multiple sensor nodes may cover the same interest points. The set of interest nodes that are covered by $s_i$ and its neighbors is defined as $O(s_i)$. As shown in Fig. 1, the coverage intersection of $s_i$ and its neighbors is:

$$O(s_1) = C(s_1) \cap (C(s_2) \cup C(s_3)) = \{p_3, p_4, p_5, p_6\}.$$

By this analogy, the intersection of $C(s_i)$ and sets of interest points covered by all neighbor nodes of $s_i$ is:

$$O(s_i) = C(s_i) \cap (C(s_1) \cup C(s_2) \cup \cdots \cup C(s_{i-1}) \cup C(s_{i+1}) \cdots \cup C(s_n)).$$

We define the coverage importance metric of $s_i$ in point coverage problems as follows.

**Definition 2.** The point coverage importance metric of $s_i$ is

$$Cl_p(s_i) = \frac{|C(s_i)| - |O(s_i)|}{|\bigcup_{j=1}^{n} C(s_j)|}.$$  (3)

where $|C(s_i)| - |O(s_i)|$ is the number of the interest points only covered by $s_i$, $|\bigcup_{j=1}^{n} C(s_j)|$ is the number of interest points covered by $s_i$ and all neighbor nodes of $s_i$. Thus, the larger the ratio $Cl_p$ is, the greater the coverage importance of $s_i$ is.

### 4.2. Area coverage

The area coverage problem, in which the monitoring target of network is a surveillance region, requires that every point in the region should be covered by at least one node. Fig. 2 shows a case of area coverage, where the nodes are distributed randomly in the rectangular surveillance area and the sensing range of each sensor node is a circle. $r_c$ is the communication radius and $r_s$ is the sensing radius.

When exploring the point coverage problem as above, it is simple to obtain the number of interest points only covered by certain sensor node based on the local information. But in the area coverage problem, it is not practical to acquire the exact
size of the region covered by certain node. Therefore, we propose a simplified method here to measure the area coverage importance of nodes. A sensor node only need to know its local information, i.e., the information of neighbor nodes within its communication range. The fewer the nodes around this node, the greater the coverage importance of the node is.

**Definition 3.** In the network, the set of 1-hop neighbor nodes of \( s_i \) is denoted as \( NB(s_i) \), which satisfies \( NB(s_i) = \{ s_j | d(s_i, s_j) \leq r_c \} \). The number of neighbor nodes within the communication radius \( r_c \) of \( s_i \) is called the degree of \( s_i \), denoted as \( |ND(s_i)| \).

**Definition 4.** The total number of \( k \)-hop neighbor nodes of \( s_i \) is called the \( k \)-hop node degree of \( s_i \), denoted by \( |ND_k(s_i)| \).

To avoid affecting the coverage performance of ECDC and the complexity of the algorithm, we use the \( |ND_k(s_i)| \) with \( k = 2 \), i.e., the 2-hop node degree of \( s_i \), denoted by \( |ND_2(s_i)| \).

The more densely nodes deployed in certain region of network, the larger the \( |ND_2(s_i)| \) of nodes is in that region, and the more overlapped area among nodes is. In other words, the node redundancy is higher. If one node of this region has been dead, other nodes would quickly repair the coverage hole. The overall coverage performance of network is just affected slightly. So the coverage importance of nodes in the region is lower, and vice versa. To depict this, we define the coverage importance metric of node \( s_i \) as follows.

**Definition 5.** The area coverage importance metric of node \( s_i \) is:

\[
CI_a(s_i) = \frac{1}{|ND_2(s_i)|}.
\]

(4)

### 5. ECDC protocol

In this section, we introduce an energy and coverage-aware distributed clustering protocol for area coverage and point coverage in WSNs.

Our clustering process is similar to LEACH [1] proposed previously. The whole operation is divided into rounds each of which contains a cluster set-up section and a data transmission section. In the cluster set-up section, ECDC is run to form a clustering topology based on the local information that combines the relative residual energy and coverage importance of nodes. In the data transmission section, cluster members first collect local data from the environment, and send it to cluster heads. Then cluster heads aggregate the data from their cluster members and then send back to the next hop nodes on the constructed routing tree. To reduce the overhead of the algorithm and prolong the lifetime of the network, the transmission phase should be longer than the set-up phase.

States of nodes are listed in Table 1 and the description of control messages is given in Table 2.

#### 5.1. Cluster set-up section

The whole process is divided into three phases: the information collection phase, whose duration is \( T_1 \); the cluster head competition phase, whose duration is \( T_2 \); the cluster formation phase, whose duration is \( T_3 \). Since the coverage importance

<p>| Table 1 |</p>
<table>
<thead>
<tr>
<th>States of nodes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>Head</td>
</tr>
<tr>
<td>Member</td>
</tr>
<tr>
<td>Plain</td>
</tr>
</tbody>
</table>

<p>| Table 2 |</p>
<table>
<thead>
<tr>
<th>Description of control messages.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>Node_Msg</td>
</tr>
<tr>
<td>1_Node_Msg</td>
</tr>
<tr>
<td>2_Node_Msg</td>
</tr>
<tr>
<td>Head_Msg</td>
</tr>
<tr>
<td>Join_Msg</td>
</tr>
<tr>
<td>Schedule_Msg</td>
</tr>
<tr>
<td>Route_Msg</td>
</tr>
</tbody>
</table>
metrics we defined for different coverage problems are embedded into our algorithms in the same way, to easy of description, two coverage importance metrics, CI_p and CI_a, are called by a joint name CI in the subsequent protocol description.

At the beginning of the information collection phase, in the point coverage application, all nodes are in state Plain. Each node broadcasts a Node_Msg, which contains the node id, id of interest points covered by it and its residual energy, within its transmission radius r_c. With the messages Node_Msgs received from other nodes within its transmission range, each node then calculates its coverage importance CI_p according to formula (3) and the average residual energy of its neighbors according to the following formula (5) based on its neighbor information.

\[
E_{ave}(s_i) = \frac{1}{|\text{ND}(s_i)|} \sum_{j=1}^{\text{ND}(s_i)} E_{\text{cur}}(s_j),
\]

where |ND(s_i)| denotes the number of neighbor nodes of s_i, and E_{\text{cur}}(s_j) denotes the residual energy of the jth neighbor of s_i.

In the area coverage application, each Plain node broadcasts a 1_Node_Msg that contains the node id within its transmission radius r_c. After receiving all 1_Node_Msgs from its 1-hop neighbors, each node will calculate the number of its 1-hop neighbors, and then broadcasts a 2_Node_Msg, which contains the node id, its residual energy and its 1-hop neighbor number, within its transmission radius r_c. After that, based on the 2-hop neighbors information, each node will calculate its coverage importance CI_a according to formula (4) and the residual energy of its 2-hop neighbors according to the following formula (6).

\[
E_{ave}(s_i) = \frac{1}{|\text{ND}_2(s_i)|} \sum_{j=1}^{\text{ND}_2(s_i)} E_{\text{cur}}(s_j),
\]

where |ND_2(s_i)| denotes the number of neighbor nodes of s_i, and E_{\text{cur}}(s_j) denotes the residual energy of the jth neighbor of s_i.

Node i collects its local information and computes its t_i

If node i has received Head_Msgs when t_i expires

Yes

No

Broadcasts a Head_Msg within R_c

Prepares to receive Join_Msgs

Sends a Join_Msg to the nearest cluster head

Creates a TDMA schedule and broadcasts it in a Schedule_Msg

Prepares to receive a Schedule_Msg

Data transmission phase

Send a Head_Msg within R_c

Prepares to receive Join_Msgs

Sends a Join_Msg to the nearest cluster head

Creates a TDMA schedule and broadcasts it in a Schedule_Msg

Prepares to receive a Schedule_Msg

Data transmission phase

Fig. 3. Cluster set-up algorithm.
where \(|ND_2(s_i)|\) denotes the number of neighbor nodes within two hops of \(s_i\) and \(E_{cur}(s_j)\) denotes the residual energy of the \(j\)th neighbor of \(s_i\).

At the end of the information collection phase, each node calculates its waiting time for broadcasting the Head_Msg message according to the following formula.

\[
t_i = \begin{cases} 
\alpha \frac{E_{cur}(s_i)}{E_{ave}(s_i)} + (1 - \alpha)C1 \quad & E_{cur}(s_i) \geq E_{ave}(s_i) \\
\frac{T_2 V_r}{2} \quad & E_{cur}(s_i) < E_{ave}(s_i)
\end{cases}
\]

(7)

where \(E_{cur}(s_i)\) is the current residual energy of node \(s_i\); \(E_{ave}(s_i)\) is the average residual energy of neighbor nodes of \(s_i\); \(V_r\) is a real value randomly distributed in \([0.9,1]\), which is introduced to reduce the probability that two nodes send Head_Msgs at the same time; \(\alpha\) is a random value in \((0,1)\), whose value can be determined according to different application requirements. Since the waiting time \(t_i\) is determined by both the relative residual energy and the coverage importance metric \(C1\) of nodes, \(\alpha\) is used to determine which factor is more important in choosing the cluster head.

When \(T_1\) has expired, ECDC begins the cluster head competition phase, whose duration is \(T_2\). In this phase, if node \(s_i\) receives no Head_Msg before \(t_i\) expires, it broadcasts the Head_Msg within the cluster radius \(R\), to claim that it will be a cluster head. Otherwise, it gives up the competition. In summary, after the cluster head competition phase, nodes with relatively higher residual energy and smaller coverage importance will be chosen as cluster heads. Thus we can obtain the cluster heads with relatively higher residual energy and lower \(C1\).

After \(T_2\) expires, the last phase, the cluster formation phase, whose duration is \(T_3\), will start. In this phase, each plain node chooses the nearest cluster head and sends the message Join_Msg which contains the \(id\) and residual energy of this node. According to the received Join_Msgs, each cluster head creates a schedule list included in the message Schedule_Msg for its cluster members. The Schedule_Msg tells the cluster members when they can transmit their data to their cluster head, such that they can alter their state to asleep in other time interval to reduce the energy consumption.

The following pseudo code give the details of this section.

```
begin (cluster set-up phase)
state \leftarrow Plain
Broadcast the Node_Msg

while (\(T_1\) has not expired) do
  Receive Node_Msgs
  Update neighborhood table NT[]
end

\(t_i\) \leftarrow waiting time for competing cluster head

while (\(T_2\) has not expired) do
  if CurrentTime < \(t_i\) do
    if receive a Head_Msg from a neighbor NT[j] do
      state \leftarrow Plain
      NT[j].state \leftarrow Head
    else
      Continue
  end
  else if state = Plain do
    state \leftarrow Head
    Broadcast the Head_Msg
  end
end

while (\(T_3\) has not expired) do
  if state = Plain\&\& has not sent Join_Msg do
    Send the Join_Msg to the nearest cluster head
  else if state = Head do
    Receive Join_Msgs from its neighbor Plain nodes
  end
end

if state = Head do
  Broadcast the Schedule_Msg
end
```

end
5.2. Data transmission section

This section is divided into two phases: intra-cluster communication and inter-cluster communication. During the intra-cluster communication process, cluster members sense and collect local data from the environment, and send the collected data to cluster heads. For simplification, cluster members communicate with cluster heads directly. While in the inter-cluster communication phase, a routing tree on the elected cluster head set is constructed. The multi-hop communication from cluster heads to the BS will further reduce and balance the energy consumption.

Several nodes need to be selected as child nodes of the BS from all cluster heads, and communicate with the BS directly. Therefore, each cluster head determines whether to be selected as the child node of the BS depending on its distance to the BS according to a threshold Euclidean distance $DIST_{TH}$. If the distance from cluster head $s_i$ to the BS is less than $DIST_{TH}$, $s_i$ communicates with the BS directly, and sets the BS as its next hop. Otherwise, it communicates with the BS through a multi-hop routing.

An routing tree will be constructed on the elected cluster head set as follows. At the beginning, each cluster head broadcasts a $Route_{Msg}$ message within the transmission radius $R_r$ with the following values: the id, the residual energy, the $CI$ and the distance to the BS of itself. To ensure the connectivity of all cluster heads, we set the radius $R_r = 2R_c$. Each cluster head chooses its next hop according to the received $Route_{Msg}$s. If the distance from cluster head $s_i$ to the BS is less than $DIST_{TH}$, $s_i$ chooses the BS as its next hop. Otherwise, it chooses the neighbor cluster head that has higher residual energy and smaller $CI$ and is not further away from the BS than it as its next hop. We give the formula of "relay" as follows when cluster head $s_i$ chooses cluster head $s_j$ as its next hop, where $\beta$ is also a random value in $(0,1)$ whose value can be determined according to different application requirements. Since the cost of relay is determined by both the residual energy and the coverage importance metric $CI$ of nodes, $\beta$ is used to determine which factor is more important in choosing the next router. $E_{cur}(s_j)$ in the formula denotes the residual energy of $s_j$, and $E_{max}(s_j)$ is the initial energy of $s_j$.

$$relay(s_i, s_j) = \beta \frac{E_{cur}(s_j)}{E_{max}(s_j)} + (1 - \beta)CI.$$ (8)

The following pseudo codes give the details of this section.

```
begin (Data transmission phase)
  Broadcast the $Route_{Msg}$
  if disttoBS < $DIST_{TH}$ do
    nexthop = BS
  else
    while ($T_4$ has not expired) do
      Receive $Route_{Msg}$s
      Compute the value of relay
      Update cluster head neighborhood table CHNT[]
    end while
  end if
  if $s_j$ has the max value of relay in CHNT[] & & $s_j$ has a smaller disttoBS in CHNT[] do
    Update MR[]
  end if
end
```

6. Protocol analysis

By analogizing the ECDC, we can obtain some properties of it as follows.

**Theorem 1.** There is at most one cluster head within each cluster radius $R_c$.

**Proof.** As we state previously, formula (7) ensures that different nodes have different waiting time. We assume that node $s_i$ has a shorter waiting timer than others and broadcasts the $Head_{Msg}$ within cluster radius $R_c$. Thus, all nodes within this range will give up the competition and become cluster members. Therefore, there is no more than one cluster head within each cluster radius $R_c$. \[\square\]

**Theorem 2.** The cluster head set generated by the ECDC algorithm is a dominating set.
Proof. According to Theorem 1, there is no more than one cluster head within a cluster, so the cluster head set must be an independent set. After the execution of ECDC, each node in the network is either a cluster head or a cluster member of some cluster. Any plain node adding to the set will destroy its independence. Thus, the cluster head set is the maximal independent set. Since the maximal independent set is also a dominating set, the cluster head set generated by ECDC is a dominating set of the WSN. □

**Theorem 3.** The distribution of cluster heads generated by ECDC is reasonable and the cluster size is uniform.

**Proof.** Since the cluster head competition is based on the relative residual energy and the coverage importance, which may lead to such a situation: nodes in the densely populated network areas will be more likely to be elected as cluster heads, while less cluster heads are generated in the sparse areas. In this case, cluster members in the sparse area will consume more energy in sending data to their cluster heads, and cluster heads in these region sending data to other distant cluster heads will also consume more energy. Thus, we set the same cluster competitive radius \( R_c \) to restrict the size of each cluster.

Any plain node received the \( \text{Head} \_\text{Msg} \) will be covered by the cluster head. For the plain node that has not received the \( \text{Head} \_\text{Msg} \), it is forced to be the final cluster head. Thus, the CH generated by the algorithm can cover all nodes in the network.

According to Theorem 1, there will be only one cluster head within the cluster radius of any cluster head, as a result of which cluster heads generated by ECDC are distributed reasonably and the cluster size is uniform. □

**Theorem 4.** The overhead complexity of control messages in the network is \( O(N) \), and the time complexity is \( O(1) \).

**Proof.** In clustering process of every round, in area coverage application, each node elected to be the cluster head would send a \( 1\_\text{Node} \_\text{Msg} \) message, a \( 2\_\text{Node} \_\text{Msg} \) message, a \( \text{Head} \_\text{Msg} \) message, a \( \text{Schedual} \_\text{Msg} \) message and a \( \text{Route} \_\text{Msg} \) message. Each cluster member node would send a \( 1\_\text{Node} \_\text{Msg} \) message, a \( 2\_\text{Node} \_\text{Msg} \) message and a \( \text{Join} \_\text{Msg} \) message. Suppose the number of generated cluster heads is \( k \). The total number of control messages in the entire network is \( N + 2(N - k) + 2k + k = 3N + 2k \), which is a linear function of \( N \) (the number of nodes in the network). In point coverage application, each node elected to be the cluster head would send a \( \text{Node} \_\text{Msg} \) message, a \( \text{Head} \_\text{Msg} \) message, a \( \text{Schedual} \_\text{Msg} \) message and a \( \text{Route} \_\text{Msg} \) message. Each cluster member node would send a \( \text{Node} \_\text{Msg} \) message and a \( \text{Join} \_\text{Msg} \) message. The total number of control messages in the entire network is \( N + (N - k) + k + k = 2N + 2k \), which is also a linear function of \( N \). Thus, the complexity of control message of ECDC is \( O(N) \).

ECDC adopts a distributed clustering strategy. Thus, the time complexity of the entire network is equal to that of a single node \( O(1) \). In other words, the time complexity is constant and independent of the network size. □

7. Simulations

Simulations were performed in NS2 platform aiming at different network coverage application problems.

7.1. Area coverage

For the practical network required for area coverage, two kinds of simulation scenarios are chosen, as shown in Fig. 4.

**Scenario 1** 100 nodes are randomly deployed over a 200 m × 200 m field.

**Scenario 2** 100 nodes are non-uniformly deployed over a 200 m × 200 m field.

The parameters of simulations are listed in Table 3.

![Fig. 4. Simulation scenarios.](image-url)
7.1.1. Cluster head distribution

In these scenarios, we firstly set $R_c$ from 10 m to 200 m, $a = 0.5$ and $b = 0.5$, and then run the protocol. Fig. 5 exhibits the number of cluster heads generated in each scenario, which is gradually decreasing along with the increase of $R_c$. Meanwhile, two curves coincide roughly which means that the number of clusters generated in two scenarios are approximately equal. The reason is that $R_c$ controls the coverage range of cluster head. Consequently, clusters have uniform cluster sizes and the distribution of cluster heads is also controlled. In the case of the nonuniform scenario, since cluster heads in dense areas would serve clusters with larger sizes than that in sparse areas, if we have no restriction on the competition radius of nodes, energy consumption of nodes in sparse region will be much higher for the reason that they have to send data to cluster heads far away from them.

To show that the related parameters we set for ECDC are feasible and effective in two kinds of scenarios, we divide experiments into different groups. Parameter $a$ are set from 0.1 to 0.9, and $b$ is also set from 0.1 to 0.9, $R_c = 100$ m. We then run ECDC in two scenarios. The variation of network lifetime along with the variation $a$ and $b$ is shown in Fig. 6. Results show that network lifetime is the optimal when we set $a = 0.2$, $b = 0.2$.

We set the competition radius $R_c$ to be 100 m. As shown in Fig. 5, the cluster number is 5, so we set the number of cluster head to be 5. To be fair, we only set $a = 0.5$, and $b = 0.5$ which is a general case of ECDC. Then LEACH, HEED, EADC and ECDC are run in the same scenario. Experimental results of 50 rounds are selected randomly. As shown in Fig. 7, the number of cluster heads varies widely in LEACH for the reason that nodes in LEACH elect themselves as cluster heads by generating random numbers as a result of which the number of cluster heads is not controlled. While in HEED, EADC and ECDC, the competition radius of them ensure that there is one and only one cluster head within any cluster radius $R_c$ and well distribution of cluster heads over the network is achieved in most of rounds. Compared with HEED and EADC, ECDC can obtain a better distribution of cluster heads number in nonuniform networks, since it considered nodes density when selecting cluster heads and constructing the inter-cluster routing tree.

---

**Table 3**

Parameters of the simulation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor field</td>
<td>$200$ m $\times$ $200$ m</td>
</tr>
<tr>
<td>BS location</td>
<td>$(250, 100)$</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>100</td>
</tr>
<tr>
<td>Initial energy of nodes</td>
<td>$1$–$3$ J</td>
</tr>
<tr>
<td>Data packet size</td>
<td>500 bytes</td>
</tr>
<tr>
<td>Packet header size</td>
<td>25 bytes</td>
</tr>
<tr>
<td>Control message size</td>
<td>25 bytes</td>
</tr>
<tr>
<td>$r_c$</td>
<td>$20$ m</td>
</tr>
<tr>
<td>$R_c$</td>
<td>$10$–$200$ m</td>
</tr>
<tr>
<td>$E_{elec}$</td>
<td>$50$ nJ/bit</td>
</tr>
<tr>
<td>$e_p$</td>
<td>$10$ pJ/(bit m$^2$)</td>
</tr>
<tr>
<td>$e_{mp}$</td>
<td>$0.0013$ pJ/(bit m$^4$)</td>
</tr>
<tr>
<td>$E_{com}$</td>
<td>$5$ nJ/(bit signal)</td>
</tr>
<tr>
<td>$E_{sen}$</td>
<td>$0.5$ nJ/(bit signal)</td>
</tr>
</tbody>
</table>

---

![Image](image.png)  

**Fig. 5.** Cluster heads generated in two scenarios.
7.1.2. Network lifetime

Here we define the network lifetime as Percentage Node Alive (PNA) [19] which defines the network lifetime as the time when 70% of nodes are still alive. We run LEACH, HEED, EADC and ECDC in two scenarios. Since ECDC considers the coverage importance of nodes besides the relative residual energy, nodes with relatively lower energy and higher coverage importance would not be selected as cluster heads or routers. As shown in Fig. 8, the improvement of ECDC in lifetime is higher in nonuniform scenario, with improvements of 58%, 28% and 5% compared with LEACH, HEED and EADC, respectively. In scenario 1, the improvements are 48%, 23% and 11%, respectively.
7.2. Point coverage

For the practical network required for point coverage, 100 interest nodes are firstly deployed randomly over a 200 m \times 200 m field. Then we also set two scenarios to monitor these interest nodes as shown in Fig. 9, where the red flags denote sensor nodes, and the green flags are interest nodes.

![Network lifetime](image)

**Fig. 8.** Network lifetime.

![Simulation scenarios](image)

**Fig. 9.** Simulation scenarios.

| Table 4 |
| Parameters of the simulation. |
| Parameter | Value |
| Sensor field | 200 m \times 200 m |
| BS location | (250, 100) |
| Number of interest nodes | 100 |
| Number of nodes | 100 |
| Initial energy of nodes | 1–3 J |
| Data packet size | 500 bytes |
| Packet header size | 25 bytes |
| Control message size | 25 bytes |
| \( r_c \) | 20 m |
| \( R_c \) | 10–200 m |
| \( E_{\text{elec}} \) | 50 nJ/bit |
| \( \epsilon_{fs} \) | 10 pJ/(bit m²) |
| \( \epsilon_{mp} \) | 0.0013 pJ/(bit m⁴) |
| \( E_{\text{com}} \) | 5 nJ/(bit signal) |
| \( E_{\text{sen}} \) | 0.5 nJ/(bit signal) |
**Scenario 1** 100 nodes are randomly deployed over a 200 m × 200 m field.

**Scenario 2** 100 nodes are non-uniformly deployed over a 200 m × 200 m field.

The parameters of simulations are listed in Table 4.

### 7.2.1. Coverage performance

To show that the related parameters we set for ECDC are feasible and effective in two kinds of scenarios, in this section, we also divide experiments into different groups. Parameter \( \alpha \) are set from 0.1 to 0.9, and \( \beta \) is also set from 0.1 to 0.9, \( R_c = 100 \) m. Then we run ECDC in two scenarios. The variation of network lifetime along with the variation \( \alpha \) and \( \beta \) is shown in Fig. 10. Results show that network lifetime is the optimal when we set \( \alpha = 0.6, \beta = 0.6 \).

To be fair, we only set \( \alpha = 0.3, \beta = 0.3 \), a general case of ECDC, then run the LEACH, HEED, EADC and the ECDC in two scenarios. The related data is summarized to show their coverage performance, i.e., the number of rounds they can maintain when the coverage of the network are 100%, 90%, 80% and 70%, respectively. Fig. 11 shows that the number of interest points that are not covered by the network is gradually increasing with the rounds, which means that the coverage ratio of network is decreasing. In addition, the coverage performance of ECDC in different scenarios is similar, which means that ECDC is suitable for either uniformly deployed network or non-uniformly deployed network. We can also obtain the results from Fig. 11 that ECDC can provide a higher coverage ratio, and maintain better coverage performance for a longer period under the required coverage ratio. The simulation results in the two scenarios show the advantage of ECDC on the coverage performance compared with other protocols, which validates the necessity of introducing the coverage importance metric into the clustering algorithm.

### 7.2.2. Network lifetime

Since the coverage importance of nodes is taken into account when selecting cluster heads and routers and the competition radius of the cluster head is controlled, the network lifetimes of ECDC in two scenarios are similar, as shown in the left

---

Fig. 10. network lifetime under different \( \alpha \) and \( \beta \).

Fig. 11. Coverage performance of different protocols.
part of Fig. 12. Thus, we only compare the network lifetime of our protocol with that of others in scenario 1. As shown in Fig. 12, a slight difference exists among the number of interest points covered by the network at the beginning, the same as the alive nodes. However, after about 80 rounds, the number of interest points covered by the network running ECDC is greater than others due to the dramatic increase of dead nodes in their networks. Since cluster heads selected by ECDC have lower $c_l$ values, even if the cluster head has been dead after this round, there will be other nodes to cover these interest nodes within its cover range in the following round. Thus, along with the postponing appearance of coverage hole, ECDC can improve the network coverage performance, so as to extend the life time significantly.

8. Conclusions

Clustering for WSNs is an effective scheme for fully utilizing sensor nodes’ energy and effectively extending the network lifetime, while coverage preservation is one of the most essential issues to guarantee the QoS of networks. However, in previous literatures, these two issues have not been extensively studied in a joint way. In this paper, we propose an energy efficient and coverage-aware distributed clustering protocol named ECDC.

In our algorithm, for different application coverage problems, corresponding coverage importance cost metrics $c_l$ are introduced. Specifically, in the point coverage problem, the point coverage importance of a node is determined by the number of interest points only covered by the node. The more the interest points only covered by a node, the larger the point coverage importance $c_{lp}$ of the node is. In the area coverage problem, the area coverage importance of a node is determined by the number of its neighbors. The fewer the nodes around a node, the greater the area coverage importance $c_{la}$ of the node is. Then only nodes with higher energy and smaller coverage importance are reliable to be selected as cluster heads and routers. The idea behind our approach is that, nodes deployed in sparsely areas or that can cover fewer interest points alone are less selected as routing nodes such that they can collect data for longer period of time.

Through theoretical analysis and massive simulation experiments, it is shown that compared with previous protocols, in a fully distributed manner and using less control information, our algorithm can construct a reasonable clustering network topology with lower energy consumption and better coverage performance. The proposed energy and coverage-aware distributed clustering protocol is energy efficient and can effectively prolong the network lifetime.

Acknowledgments

This work was partially supported by the NSF of China for Contract (61373027, 11101243), NSF of Shandong Province for Contract (ZR2012FM023, ZR2012FQ011), STPU of Shandong Province for Contract (J10LG09, J12LN06), PFMYS of Shandong Province for Contract (BS2009DX024, BS2010DX013).

References


Xin Gu received the BS degree in computer science and technology from Qufu Normal University, Shandong, China, in 2010. She is currently a postgraduate in the School of Computer Science, Qufu Normal University. Her main research interest is wireless sensor networks.

Jiguo Yu received the PhD degree in operations research and control theory from Shandong University, Shandong, China, in 2004. He is currently a professor in the School of Computer Science, Qufu Normal University, Shandong, China. His main research interests include wireless networks, distributed algorithms, and graph theory.

Dongxiaoyu Yu is currently a PhD student in Department of Computer Science, The University of Hong Kong. His research interests include wireless networks, distributed algorithms, algorithmic mechanism design and graph theory.

Guanghui Wang received the Ph.D. degree in computer and information science, at Paris sud University, France, and the Ph.D degree in Mathematics, at Shandong University, China in 2007. He is now an associate professor in School of Mathematics, Shandong University, Shandong, China. His research interests include wireless network, graph theory, combinatorics and algorithms.

Yuhua Ly received the BS degree in computer science and technology from Qufu Normal University, Shandong, China, in 2010. He is currently a postgraduate in the School of Computer Science, Qufu Normal University. His main research interest is wireless sensor networks.