Computers and Electrical Engineering 48 (2015) 358–370

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Energy efficient data collection through hybrid unequal clustering for wireless sensor networks ☆

L. Malathi [a,*], R.K. Gnanamurthy [b], Krishnan Chandrasekaran [c]

[a] Vivekanandha College of Engineering for Women, Namakkal 637205, Tamil Nadu, India
[b] S.K.P. Engineering College, Thiruvannamalai 606611, Tamil Nadu, India
[c] Angel College of Engineering and Technology, Tirupur 641665, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 14 August 2014
Received in revised form 13 June 2015
Accepted 16 June 2015
Available online 22 July 2015

Keywords:
WSN
Network lifetime
Unequal clustering
Energy hole problem
Hybrid clustering
Compression

ABSTRACT

The existing clustering algorithms are either static or dynamic depending on the frequency of clustering. In static clustering, clusters are formed once, which reduces the clustering overhead but leads to early energy drain of a few nodes in the network. The network lifetime can be improved by dynamic clustering in which clusters are reformed after every round, which increases the clustering overhead. To optimize the parameters, including clustering overhead, network lifetime, energy hole, FND (first node dies) and LND (last node dies) in WSN, a hybrid unequal clustering with layering protocol (HUCL) is proposed. The HUCL is a hybrid of static and dynamic clustering approaches. In HUCL, the network is divided into layers and clusters of various sizes. The cluster heads are selected based on available energy, the distance to the sink and the number of neighbors. Once the cluster is formed, the same structure is maintained for a few rounds. The data are forwarded to the sink through a multi-hop layer-based communication with an in-network data compression algorithm. In comparison with the existing protocols, the HUCL balances energy and achieves a good distribution of clusters, extends the lifetime of the network and avoid the energy hole problem.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Wireless sensor networks (WSNs) are highly distributed networks of small, lightweight, battery-embedded sensor nodes. WSNs are becoming increasingly attractive for numerous application areas, such as military reconnaissance, disaster management, security surveillance, habitat monitoring, health care and industrial automation [1]. Data collection from these networks may be periodic, query based or event based. In periodic data gathering, the sensor nodes sense the environment and transmit the sensed value to the sink at regular intervals; the routing strategy plays an important role in prolonging the network lifetime in these applications. The widely suggested routing techniques are distributed among different classes, including flat, hierarchical and location-based techniques. Many energy efficient solutions have been developed in each category of routing protocols. An approach that is likely to succeed is the use of a hierarchical structure [2].

Hierarchical organization of sensor networks leads to three different types of routing protocols based on how they communicate data to the sink, which are the cluster-based approach, chain-based approach and tree-based approach.

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. H.A. Shabeer.

* Corresponding author. Tel.: +91 9952312970.
E-mail addresses: malasuamthi@gmail.com (L. Malathi), rkgnanam@yahoo.co.in (R.K. Gnanamurthy), mkchand69@hotmail.com (K. Chandrasekaran).

http://dx.doi.org/10.1016/j.compeleceng.2015.06.019
0045-7906/© 2015 Elsevier Ltd. All rights reserved.

Applications that cover large sensor fields and need frequent data gathering should support data aggregation as a prime candidate for improving the lifetime of the network. Cluster-based configuration has achieved this with minimum overhead. In clustering schemes, sensor nodes are grouped into clusters, a node is selected as the cluster head (CH) and the other nodes are called cluster members (CMs) [3]. Each node in the cluster collects local data from the environment periodically and sends it to its cluster head. After receiving the data from all CMs, the CH aggregates them and transmits to the sink via direct single hop communication or through a multi-hop routing path. The existing clustering algorithms can be static or dynamic depending on the frequency of clustering. In dynamic algorithms, at each round, a new cluster head is elected and the clustering boundary is reconstructed, which increases the clustering overhead. In static algorithms, the cluster is formed once, which reduces the clustering overhead, but the cluster head is overloaded, decreasing the network lifetime.

In this work, a proposal for a hybrid unequal clustering with layering protocol (HUCL) is presented. The HUCL is a hybrid of static and dynamic clustering approaches. In HUCL, the network is divided into layers and precise numbers of cluster heads are selected depending on the nodes residual energy, number of neighbors and centrality among neighbors. Different sizes of clusters are formed locally depending on the distance from the sink.

The author's contribution in this paper: Each round of data transmission is divided into major slots, and each major slot is divided into several mini-slots. 1. In each mini-slot, the CMs sense the environment and send the data to the CH; the CH aggregates and transmits the data to the sink through a multi-hop transmission chain. 2. In every major slot, the current CH selects a node as the new CH within the same cluster boundary depending on the energy. The current CH informs the new CH about its predecessor and successor in the data transmission path. 3. In each round, the clusters are reformed to avoid early drain of some nodes. 4. An in-network data compression algorithm is used to enhance the lifetime of the network. Thus, the proposed approach reduces the clustering overhead and improves the network lifetime.

The rest of the paper is organized as follows. Section 2 presents an overview of clustering in WSN. Section 3 presents earlier works on uneven clustering and compression. Section 4 explores the detailed description of the proposed HUCL algorithm. Section 5 analyses several properties of the HUCL. Section 6 gives the simulation results, and Section 7 concludes the work.

2. Clustering overview

In clustering, the sensor nodes in a WSN are divided into different virtual groups according to a set of rules. Under a cluster structure, sensor nodes may be assigned different functions, such as CH or CM [4]. A cluster head normally serves as a manager for its cluster and provides intra-cluster transmission schedules, collects the data from its members, performs data aggregation/fusion and forwards the aggregated data to the sink. The characteristics and advantages of clustering algorithms were discussed in previous articles [1,2,4–6]. In earlier studies [7,8,10], CH uses single hop communication to transfer the data to sink, but to increase the network lifetime, multi-hop communication has been used in recent years. However, the previous studies suffer from an important problem called the "hot spot" problem. The papers [10–18] address this issue with uneven size clusters. To minimize the clustering overhead of dynamic clustering, the papers [19–22] use a static clustering approach. However, in the static approach, the CHs should have higher power than its members; otherwise, the CH may drain quickly.

3. Related works

In recent years, many algorithms have been proposed in the area of energy-efficient clustering and routing for WSNs. The authors of [7] proposed a clustering protocol called LEACH for periodic data-gathering applications. LEACH is a dynamic, two phase protocol. CHs are distributed in the setup phase, and the CH directly transmits data to the sink in the steady phase. An energy-aware variant of LEACH is proposed in [8]; here the nodes with higher energy are more likely to become CHs. HEED [9] introduces cluster radius to define the transmission power used for intra-cluster broadcasting. The tentative cluster heads are elected based on residual energy, and final heads are selected according to the intra-cluster communication cost. HEED achieves fairly uniform distribution of cluster heads across the network. In EECS [10], unequal size clusters are formed based on the distance from the sink. EECS uses a cluster head competitive algorithm without message exchange iterations. Because it uses single hop inter-cluster communication, clusters farther from the sink have smaller sizes; thus, some energy is preserved for data transmission to the sink.

Soro and Heinzelman [11] examined an unequal clustering model for balancing the energy consumption of CHs in multi-hop sensor networks. They focused on a heterogeneous network where high power cluster heads were deployed at pre-computed locations. In this case, it was easy to control the actual sizes of the clusters. EEUC [12] is a distributed, unequal clustering algorithm that elects tentative cluster heads based on the residual energy of the nodes. The tentative cluster head has a competition radius determined based on the distance from sink. The tentative cluster heads competing to become the final cluster head. The algorithm forms more clusters near the sink. The inter-cluster communication is a multi-hop model; each CH selects another CH in the upward direction to transmit the data to the BS. In [13], Chen extended EEUC to mitigate the hot spot problem and introduced a novel unequal clustering protocol called UCR. The selection and rotation of CHs are based on residual energy. The model increases the network lifetime by decreasing the number of nodes in clusters with higher relay loads near the sink. In [14], the authors addressed the "isolation point" and the hot spot problems. Each node collects its neighbor information for computing the average energy and assigns the waiting time for each node based on residual energy. After the waiting time expires, the node itself assumed as the CH and sends the CH announcement message

within its competition radius. For inter-cluster communication, it uses a multi-hop communication pattern. In [15], multi-hop routing protocol with unequal clustering (MRPUC) was proposed. A node is elected as a CH if it has the highest residual energy among all neighbor nodes. If a node fits into the radius of any CH, it can join as a member. MRPUC constructs an inter-cluster routing tree rooted at BS to save energy. The minimum cost node is considered as the parent node among all neighbors CHs. The hot spot problem is mitigated, but the number of control messages used is high. In [16], the unequal cluster versions of HEED were proposed as UHEED. The unequal size clusters are created based on the distance of the CH from the sink. The competition radius calculation is the same as EEUC; UHEED creates a smaller cluster closer to the sink. The authors of [17] proposed a grid based scheme for energy efficient data collection. It divides the network into grids with unequal sizes and implements CH rotation in each grid. The method balanced energy dissipation by setting proper sizes of grids to adjust the number of nodes that participate in CH rotation in different grids. In ECDC [18], the authors introduced the coverage importance (CI) metrics for area coverage and point coverage. They induced these metrics in the waiting time calculation and to find a data forwarding path to the sink to increase the network lifetime.

In [19], the authors proposed EEPSC (energy-efficient protocol with static clustering), a static clustering based protocol, which eliminates the overhead of dynamic clustering and has high power sensor nodes for power consuming tasks to prolong the network lifetime. The Hausdorff clustering algorithm is a static clustering method, in which the Hausdorff distance between two node sets is used as the clustering metric [20]. It conducts cluster formation based on the minimum Hausdorff distance between nodes of the same cluster and between the nodes of neighboring clusters to ensure connectivity within the network. In [21], the authors proposed ERP-SCDS (energy-efficient routing protocol for wireless networks with static clustering and dynamic structure), which utilizes virtual points in a corona-based WSN, forming static clusters. The clustering process performed by virtual points around the sink. The location of the virtual points is calculated by the sink after receiving location information from all sensor nodes in the network. In [22], the authors proposed virtual concentric circle band based clustering (VCCBC), which divides the sensing field into concentric circles. The node located exactly midway between two concentric circles is elected as the first round candidate CH. A back off timer is used for the final CH selection. The frequency and timing of the CH rotation process are determined by calculating the energy consumption for regular sensor nodes and CH nodes. The taxonomy of energy-efficient clustering protocols is given in Fig. 1.

Compression reduces the energy required for data transmission, but sensor nodes have limited memory and computation power. Therefore, the compression algorithm should require less computation. To meet these challenges, the authors of [23] developed the S-LZW compression algorithm. This algorithm effectively uses the sensor resources, but compression latency is high. In [24], the authors proposed a modified version of run-length encoding named RLE-ST. This method reduces the computation time, but compression ratio is poor. The authors of [25] proposed a routing algorithm with compression to reduce redundant data transmission.

To optimize the problem of clustering overhead in dynamic clustering and the coverage and energy hole problems of static clustering, HUCL was proposed in this paper. The algorithm combines static and dynamic clustering and uses a simple compression algorithm to reduce redundant data transmission.

4. The proposed approach

This section describes the details of the HULC used to meet the requirements of current emerging applications of WSN.

4.1. Network model

A few assumptions are made about the sensor nodes and the network model:

1. There are '*N*' sensor nodes distributed randomly in an $M \times M$ field.
2. All nodes are homogeneous.
3. All nodes and the sink are stationary.
4. All nodes can adjust their transmission power according to the distance.
5. The sink is outside of the sensor field. It can be reached by all sensors and has sufficient energy.
6. CH can perform data aggregation and compression.

The radio model used in HUCL is same as in LEACH. The energy consumed to transmit *l*-bit data a distance *d* is,

$$ETx(l,d) = \begin{cases} l \times E_{elec} + l \times \varepsilon_{fs} \times d^2, & d \leqslant dl_1 \\ l \times E_{elec} + l \times \varepsilon_{amp} \times d^4, & d > dl_1 \end{cases} \quad (1)$$

where ε_{fs} is the energy consumed by a free space model, ε_{amp} is the energy consumed by a multi-path amplifier circuit and dl_1 is the distance for layer 1. Therefore, the limitation is that the layer 1 CHs are the only nodes to transmit data to the sink directly. To receive the message, the expended energy is

$$ERx(l) = l \times E_{elec} \quad (2)$$

In addition, the following assumptions are made: the energy spent by the node for sensing is E_s, and the CH spends E_{agg} energy for performing data aggregation and E_{com} for doing in-network data compression.

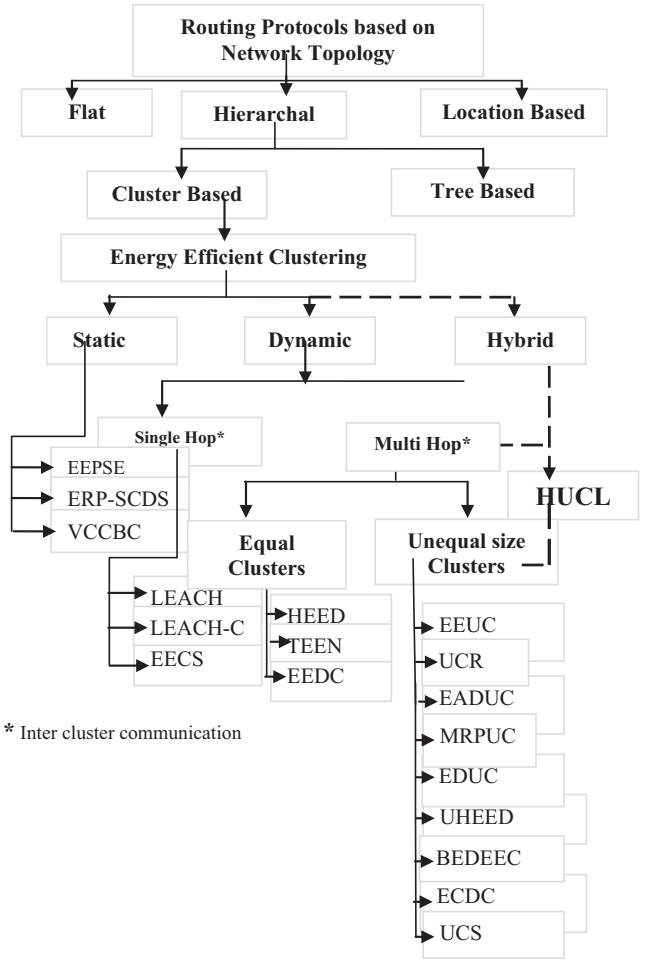


Fig. 1. Taxonomy of energy-efficient clustering protocols.

4.2. Protocol operation

After deployment, the nodes are grouped into layers based on the distance from the sink, and a unique id is assigned to all nodes according to layer. The sink broadcasts a HELLO message with a layer-id using the minimum power; the nodes that receive this message form layer-1 and calculate the distance from BS using the received signal strength. Then, the sink increases the power level and broadcasts the HELLO message again. This layering is used for only inter-cluster communication. The clusters are independent of the layers.

The whole operation of HULC is divided into rounds; each round has a setup phase and a data transmission phase. To reduce the overhead, the data transmission phase is divided into '*M*' major slots, and each major slot contains '*m*' mini slots. A mini slot is the actual data transmission, which covers the TDMA schedule distribution, data collection from sensors by CH, data aggregation by CH and transmitting data to the sink. Major slots perform CH rotation within the cluster boundary and handover the cluster members and the data forwarding path to the new CH. The overall operation of HUCL is given in Fig. 2.

The setup phase has five sub-phases: 1. Neighbor node discovery (NND), 2. Cluster head competition (CHC), 3. Cluster formation (CF), 4. Cluster optimization (CO), and 5. Constructing the data transmission path to the sink (PC).

4.3. Setup phase

In the setup phase, the network is divided into '*k*' number of clusters. It has five sub-phases, as mentioned previously; each requires a specific amount of time to complete, T_1–T_5, respectively. The time taken to complete the cluster formation is T ($T = T_1 + T_2 + \cdots + T_5$). After T, the actual data transmission starts. In neighbor node discovery, each node broadcasts the Neighbor_Msg within radio communication radius T_c. It also receives t Neighbor_Msg from other nodes and updates its Neighbor_List. Each node calculates the random waiting time 'T_w' as

$$T_w(s_i) = \frac{E_{\text{init}}(s_i)}{E_{\text{rem}}(s_i)} \times \frac{1}{\alpha|\text{NL}(s_i)|} \times T_2 \times R \tag{3}$$

where E_{init} and E_{rem} are the initial and residual energy of the node 's_i'. α denotes the number of times the node 's_i' acts as a CH. $|NL(s_i)|$ is the number of neighbors for the node 's_i'. R is a random number between 0.1 and 0.2. After T_1, the cluster head competition phase starts; each node 's_i' waits for time '$T_w(S_i)$' to receive the Head_Msg from other nodes. If no message is received then the node itself is assumed as the head and broadcasts the Head_Msg within its competition radius R_c. R_c is calculated using,

$$R_c(s_i) = \left[1 - c\frac{d_{max} - d(s_i, BS)}{d_{max} - d_{min}}\right] RL_{max} \tag{4}$$

where RL_{max} is the layer-dependent maximum competition radius, which is pre-determined during network initialization, and d_{max} and d_{min} are the maximum and minimum distance between the sink and nodes. C is the weighting factor, which can take a value between 0 and 1.

Algorithm 1: Cluster setup (for each node 'i' in network s[ ])

```
Begin
if S[i].state = "CHP"
  exit
else
  S[i].state ←node
    Broadcast Neighbor_Msg;
    while (CT < T1)
      Receive Neighbor_Msg
      update neighbor List NL [ ]
    whileend
  μ = rand (0.1, 0.2)
  Calculate Tw = Einit/Erem *1/(α*|NL|)*μ
  t = T1 + Tw
  while (CT < T2)
    if(CT > T)
      state = 'CH'
      calculate the Rc
    broadcast Head_Msg
      receive Head_Msg from competition CH
      store in Head_List HL[ ] along with distance
  elseif (received Head_Msg from any neighbor)
      state = 'CM'
      store 'si' in Head_List HL[ ] along with distance
  end
whileend
while (CT < T2)
  if state = CM
    select the nearest CH sj from HL[ ] list
    si.head = sj
    send JoinCluster_Msg to si
  elseif state = CH
    Receive JoinCluster_Msg from CM
      store in CM[ ] List
  end
whileend
while (CT < T4)
  if state = CH and |CM[ ]| = 0
    state = CM
    select the CH from HL[ ]
    send a JoinCluster_Msg
  end
  whileend
  if state = CH
      Prepare and broadcast TDMA to members of CM [ ]
    α = α + 1
    state = 'CHP'
  end
  end
End
```

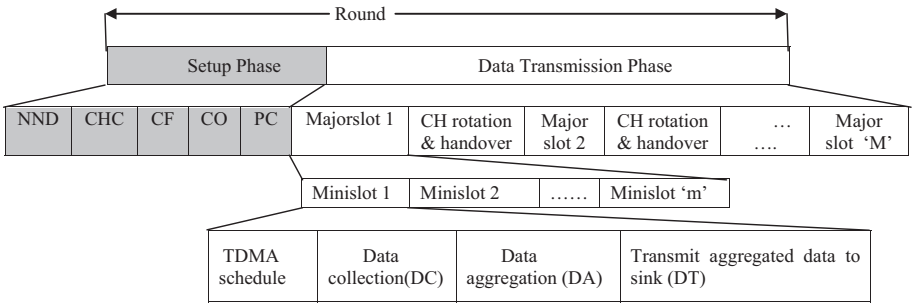


Fig. 2. HUCL operation.

When a node receives the Head_Msg, it updates the Head_List to find the nearest head. After T_2 expires, nodes find the nearest CH and send the JoinCluster_Msg. On receiving the JoinCluster_Msg, the CHs update their Member_List. After T_3 expires, the cluster optimization phase starts, in which member-less clusters (CH alone) and single member (CH + one node) clusters are removed and combine with nearest clusters. After cluster optimization, CH constructs the path for transmitting data to the sink. The time taken to complete the path is T_5. It is divided into two different components (T_{51}) for constructing the forward path through Route_Msg and (T_{52}) for Route_Reply. During this time, T_5 cluster members may be in a sleep state.

After cluster optimization, the layer1 CH broadcasts the Route_Msg with its Node_ID. The layer 2 CH, which receives the Route_Msg, updates its ALL_PATH_List and sends its Route_Msg. This continues for all layers. After T_{51} expires, each CH waits for a random amount of time depending on the layer and starts transmitting the Route_Reply_Msg to the nearest upper layer. On receiving the Route_Reply_Msg, the CH will update its F_PATH_List.

Algorithm 2: Data forwarding chain construction (for each CH Si in the Network)

```
Begin
  if node belongs to Layer 1
        relay ← BS
  else
    Calculate TW2 = T5 × 1/layerindex
    While (Tw2 has not expired)
      Receive Route Message
      Update ALL_PATH_List
whileend
  relay = ALL_PATH[0]
  for each CH Sj in ALL_PATH_List
    if (d(Si,Sj) < d(Si,relay))
      relay = Sj
    end
  forend
    send the Route_Reply to node Srelay
  end
End
```

4.4. Data transmission phase

After the setup phase, data transmission starts by preparing the TDMA schedule by CH and distributing to its members. CH collects the data from its members, aggregates it and transmits it to a higher layer CH. The higher layer CH receives the data from the other CH and checks whether the data are already compressed. If yes, it forwards the data to a higher order CH; otherwise, it performs in-network data compression and transmits the compressed data to the higher layer. The data packet size is assumed to be 900–1000 bytes. The header has a bit to indicate whether the data are already compressed. For compression, the simple 2-hop, in-network compression algorithm is used. The algorithm used for compression is given in Algorithm 3.

Algorithm 3. Data compression (X_1, X_2)

```
//X – Aggregated data
Begin
  if (compression_bit = 0)
    α = Tolerance threshold
    λ = Maximum threshold
    if (X1 + λ > X2 or X1 − λ < X2)
      exit
    else
    Compute μ = X1 + X2/2;
    for each value xi in X
      Compute Δ = sqrt((μ − xi)^2)
    If (Δ < α) data = X1 + 0
      xi.Compression_bit = 1
    elseif(Δ > λ)
      xi.Compression_bit = 0
    else
      data = X1 + Δ
      xi.Compression_bit = 1
    endif
    end
  endif
end
```

4.5. Cluster head rotation and handover

During the last mini slot, the nodes send their remaining energy along with the sensed data. The CH selects the new CH based on the remaining energy and its distance from the current CH. The CH hands over the Member_List, ALL_PATH_List and F_PATH_List to the new CH, which directly sends the Route_Msg to the members of the ALL_PATH_List to find a new path.

5. Protocol analysis

Lemma 1. *The control message complexity of a clustering algorithm is O (N), and it is reduced by at least the factor (m * M/2).*

Proof. Assume there are '*N*' number of nodes, '*m*' mini slots, '*M*' major slots, '*r*' rounds, and '*l*' layers in the network. During layering, the sink broadcasts '*l*' layer messages. For cluster setup, 'Neighbor_Msg' and '*k*' head messages are broadcasted, and $(N - k)$ Join_Cluster messages are transmitted. For cluster optimization, '*x*' messages are needed, where $0 \leqslant x \leqslant k$. the upper limit k is considered. For cluster setup, $N + k + (N - k) + k$ messages are required. □

For constructing the data transmission path, $2k$ messages are needed, and for distributing the TDMA schedule, k control messages are broadcasted.

In all, $N + k + (N-k) + k + 3k = 2N + 4k + l$ messages are required for sending data to the sink. The first major slot may have $2N + 4k + mk + l$ control messages, where *'m'* is the number of mini slots. For CH rotation, a maximum $2k$ control messages are needed. The total number of control messages for a round, which is equal to m, and for M rounds in the existing algorithm is $2N + 4k + mk + M2k$ messages +l message for layering.

The authors [12,14] proved that the total number of messages for each round is $2N + 2k + x$, where x varies from 0 to the number of tentative CHs, depending on the algorithm.

5.1. Numerical example

Consider the example of 100 nodes, 500 rounds and 10 clusters.

5.1.1. Existing approach

Assume x as 0. The number of control messages per round = $2N + 2k$. For r rounds, $r*(2N + 2k)$, that is, for 500 rounds, 100 nodes (N) and 10 clusters (k), it is 111,000.

5.1.2. Proposed approach

Consider 5 minislot (m), 4 majorslot (M), 500 rounds are considered as $500/m \times M = 25$ rounds, and the number of control messages for r rounds = $r/(m * M) * (2N + 4k + mk + M2k) + l = 9254$ messages, which is less than 10,000. For the average of

500 rounds, 100,000 control messages are reduced. Consider a control packet size of 150 bytes and a data packet size of 1000. Form the saved 100,000 control packets of energy, approximately 5000 data packets can be transmitted, which increases the network lifetime.

Lemma 1 shows that the control messages are very low compared to the existing approaches. The reduction factor is $(m * M/2)$. This increases the network lifetime significantly. A proper value of $m * M$ leads to better improvement in lifetime.

Lemma 2. *There is no possibility of having two CHs if one node is in the competition radius of the other node.*

Proof. Eq. (3) ensures that different nodes have different waiting times. If the node si has a shorter waiting time and broadcasts the Head_Msg within radio range 'Rci', then all of the nodes within this range become CMs. Therefore, there is no more than one CH in the range of any CH with radius 'Rci'. □

The protocol uses the CSMA MAC during the exchange of messages. During cluster formation, if the node is the CH of the last round, then it can be in the sleep state for the entire T2 duration. During the slot T5, the cluster members are in the sleep state to save energy. During the data transmission phase, the nodes may wake only at their allotted time slot. Other nodes may be in the sleep state. In each phase of cluster setup, an appropriate time interval should be chosen for proper operation of the algorithm. The protocol also requires time synchronization among the sensors. An assumption is made that all sensors are synchronized and start each phase at the same time. The performance is also controlled by the values of 'c' and RL_{max} in Eq. (4). In addition to the theoretical control overhead, the sink has to send beacon frames for synchronization.

6. Simulation results and discussion

A simulation of the proposed approach was performed using MATLAB, and parameters, including network lifetime, cluster head distribution, FND, and LND, were analyzed. Error-free communication links and ideal MAC were assumed for the simulation. The parameters used for the simulation are given in Table 1.

The experiments were conducted for two scenarios, which are given in Fig. 3.

Table 1
Simulation parameters.

Parameter	Value
Sensor field	100 m × 100 m
BS location	(150, 50)
Number of nodes	100
Initial energy of nodes	2 J
Data packet size	1000 bytes
Packet header size	25 bytes
Control message size	50 bytes
RL_{max}	10–60 m
E_{elec}	50 nJ/bit
E_{fs}	10 pJ/(bit m^2)
E_{amp}	0.0013 pJ/(bit m^4)
E_{agg}	5 nJ/(bit signal)
E_{com}	2 nJ/(bit signal)
E_S	0.5 nJ/(bit signal)

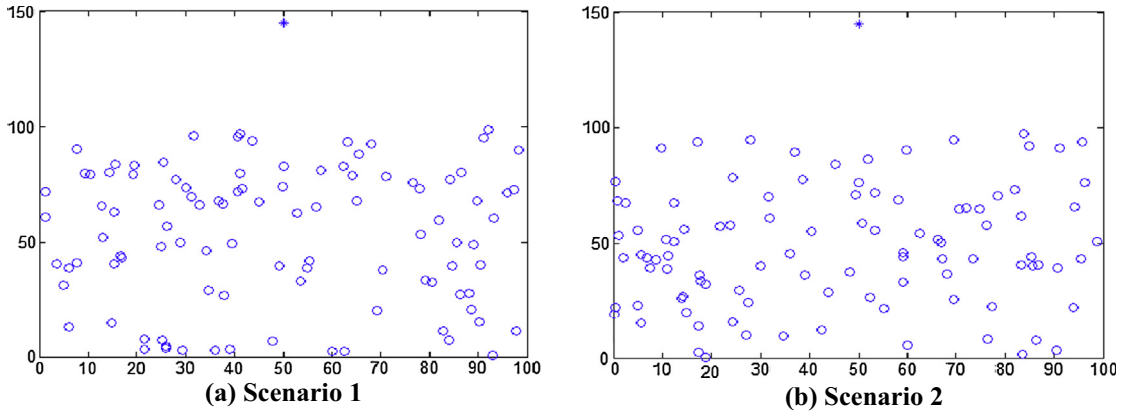


Fig. 3. Simulation scenarios.

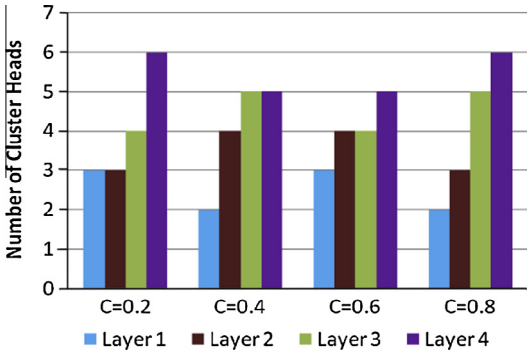


Fig. 4. Cluster head distribution on layers.

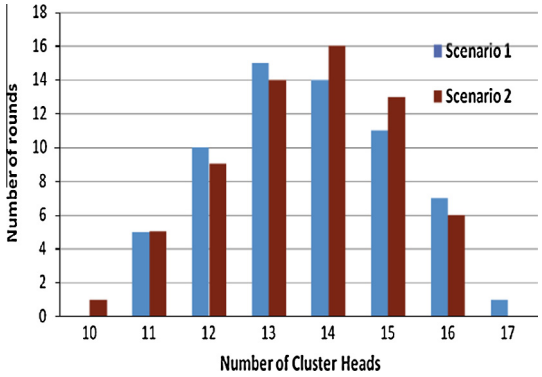


Fig. 5. Cluster head distribution in different scenarios.

For cluster formation, Eq. (4) has a layer-dependent maximum competition radius RL_{max}. For Layer 2 and 3, it is 1.5 times greater than Layer 1 ($RL_{max}2 = RL_{max}3 = 1.25 \times RL_{max}1$.) For Layer 4, it is 1.75 times greater, $RL_{max}4 = 1.75 \times RL_{max}1$. For the simulation, $RL_{max}1$ is considered as $20 \leqslant RL_{max}1 \geqslant 50$. $RL_{max}1$ has an impact on the cluster head distribution. The parameter that affects the competition radius of nodes, as per Eq. (4), is '*C*', which can take values between 0 and 1.

The effect of *C* during cluster setup is given in Fig. 4. $C = 0.6$ gives a normal distribution of CHs. For the simulation, *C* is considered as 0.5. Fig. 5 shows the cluster formed for multiple runs of scenario 1 and scenario 2. On average, 10–12 clusters were formed in each round during the simulation. Sensor nodes spent much of their energy for transmission rather than sensing or processing. The theoretical analysis proved the hybrid scheme reduces the number of control messages. To balance the load among intra-cluster and inter-cluster communication, nodes closer to the sink have smaller size clusters; thus, less energy is spent for intra-cluster communication, which preserves energy for inter-cluster communication.

Next, the energy efficiency of the proposed protocol was analyzed by determining the rounds at which the first node dies and last node dies. The hybrid scheme combines static and dynamic cluster formation in the form of mini and

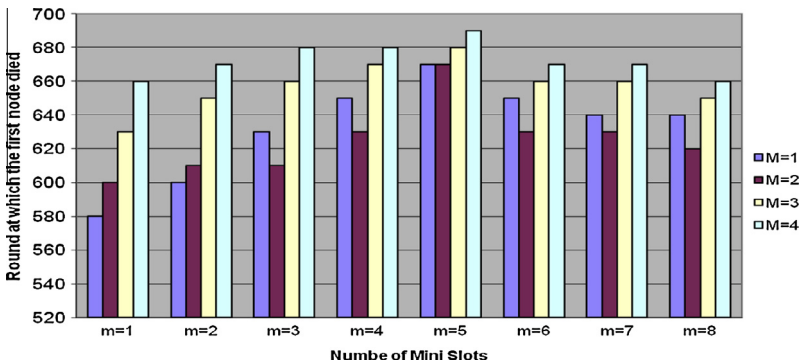


Fig. 6. Network lifetime for different mini-slots '*m*' and major slots '*M*'.

major slot. Selecting the number of mini-slots '*m*' and major-slots '*M*' has a high impact on the network lifetime. Fig. 6 shows the lifetime of the network for scenario 2 on different values of '*m*' and '*M*'. For the simulation, *m* is 5 and *M* is 4.

In practical applications, sensors are distributed randomly; the clustering protocol should extend the lifetime in uniform and non-uniform scenarios. Fig. 7 shows the lifetime of a network when using compression. The result demonstrates that the hybrid approach works equally for both scenarios, and it uses 2-hop compression. The compression improves the network lifetime by 8%. During compression, CHs spend little energy, but compression reduces the inter-cluster communication cost, thereby extending the lifetime. Fig. 8 shows the lifetime of network without compression.

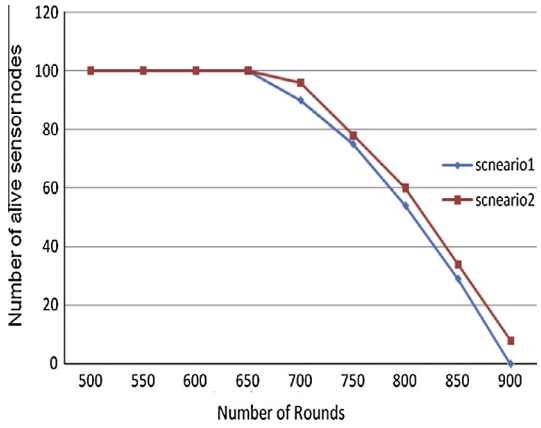


Fig. 7. Network with compression.

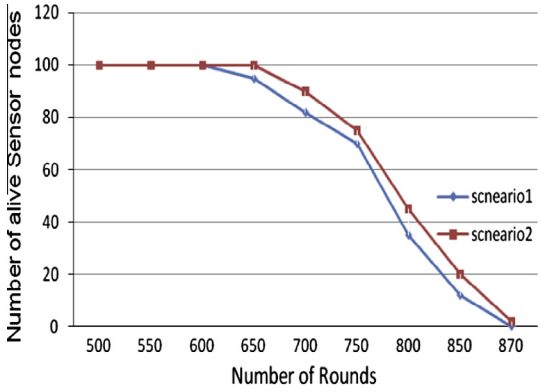


Fig. 8. Network lifetime without compression.

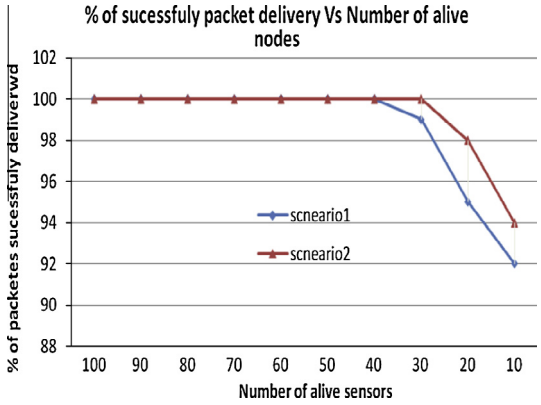


Fig. 9. Successful packet delivery.

In periodic data collection, sensor nodes sense the environment and update the sink with the latest information. The number of such successful transfers is considered an important measure for the routing algorithm. Fig. 9 shows the percentage of successful packets delivered to the sink over the number of active nodes. HUCL achieves 100% success even after 50% of the nodes died.

Apart from the variables of routing algorithms, many factors, such as the position of the sink and the number of clusters, affect the performance of the protocol. One example is given in Fig. 10, which presents how the sink location affects the

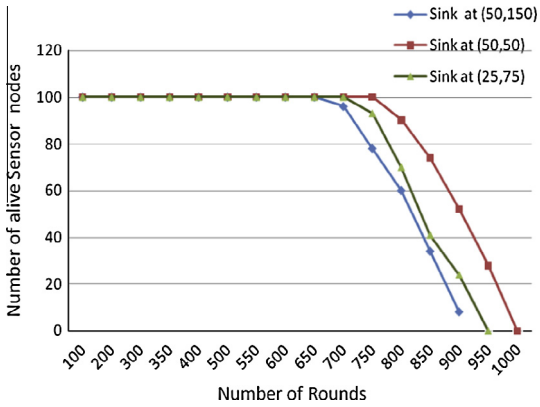


Fig. 10. Network lifetime in uniform distribution at different sink position.

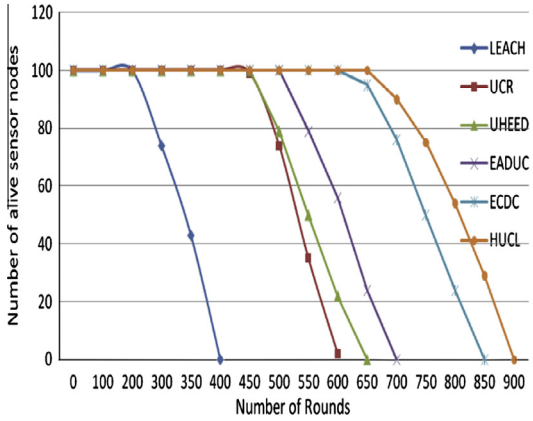


Fig. 11. Network lifetime in non-uniform distribution.

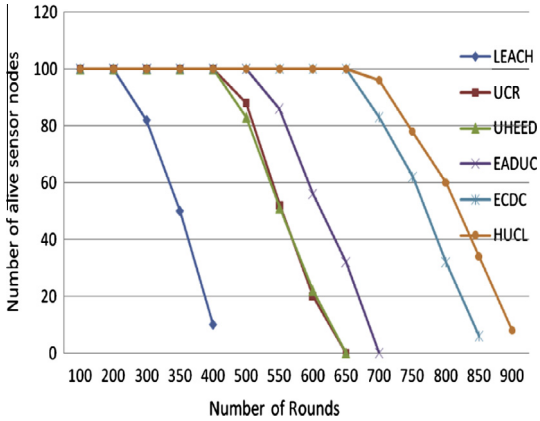


Fig. 12. Network lifetime in uniform distribution.

network lifetime for a uniform node deployment scenario. Fig 10 shows that if the sink is placed at the center of the monitoring area, the lifetime of the network can be improved. However, in many real time applications, the sink is placed far from the monitoring area.

The energy efficiency of the proposed protocol was compared with existing algorithms, including LEACH UCR, UHEED, EADUC and ECDC. UCR, UHEED and EADUC use similar inter-cluster communication methods. EADUC addresses homogeneous and heterogeneous environments. For the comparison, only a homogenous environment is considered. Similarly, for ECDC, only the area coverage problem is considered. In UHEED, the authors concluded that their approach yielded better performance when $C = 0.8$ in the competition radius calculation. The same value is kept for UHEED. Fig. 11 shows the number of surviving nodes in each round in the non-uniform sensor distribution scenario. Fig. 12 shows the number of surviving nodes in each round in the uniform sensor distribution scenario.

7. Conclusion

Even though clustering is an effective technique for periodic data collection from WSNs, clustering parameters and frequency of clustering should be chosen optimally for reducing the clustering overhead. Previous studies exploit either static or dynamic clustering but here hybrid unequal clustering algorithm was proposed to enhance the network lifetime. The proposed algorithm divides the data transmission phase into mini slots and major slots, each major slots uses static clustering with cluster head rotation. Dynamic clustering is performed after completing all the major slots. Through theoretical analysis, it is proved that the proposed hybrid approach reduces the clustering overhead to a great extent compared to other dynamic clustering techniques. The simulation results show that the hybrid approach extends the network lifetime and ensures the uniformity of energy consumption in the network which mitigates the energy hole problem. The network lifetime was enhanced by 67%, 33%, 32%, 24% and 9%, respectively, compared to LEACH, UCR, UHEED, EADUC and ECDC. The proposed work uses simple compression algorithm which also improves lifetime by 2% without using complex algorithm. This work considers only a simple compression algorithm, irrespective of the local information of the node. In the future, the focus will be on extending the lifetime of the network by eliminating redundant data through a compression algorithm that considers the local and global state of the network.

References

[1] Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. Comput Network 2002;38:393–422.
[2] Abbasia Ameer Ahmed, Younisb Mohamed. A survey on clustering algorithms for wireless sensor networks. Comput Commun October 2007;30(15):2826–41.
[3] Zhou X, Wu M, Xu J. BPEC: an energy-aware distributed clustering algorithm in WSNs. J Comput Res Develop 2009;46(5):723–30.
[4] Liu Xu-Xun. A survey on clustering routing protocols in wireless sensor networks. Sensors 2012;12:11113–53.
[5] Ahmed A, Mohammed Y. A survey on clustering algorithms for wireless sensor networks. Comput Commun 2007;30:2826–41.
[6] Malathi L, Chandrasekaran MK, Gnanamurthy RK. A novel routing protocol with lifetime maximizing clustering algorithm for WSN. In: Proceedings of annual IEEE India conference (INDICON), Kochi, India; December 2012, doi:http://dx.doi.org/10.1109/INDCON.2012.6420749.
[7] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33rd annual Hawaii international conference on system sciences, Maui, HI, USA; 4–7 January 2000. pp. 10–19.
[8] Heinzelman W, Chandrakasan A, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wireless Commun 2002;1(4):660–70.
[9] Younis O, Fahmy S. HEED: a hybrid, energy-efficient distributed clustering approach for ad hoc sensor networks. IEEE Trans Mobile Comput 2004;3(4):660–9.
[10] Ye M, Li C, Chen G, Wu J. EECS: An energy efficient clustering scheme in wireless sensor networks. In: Proceedings of the 24th IEEE international performance, computing, and communications conference (IPCCC), Phoenix, AZ, USA; 7–9 April 2005. pp. 535–40.
[11] Soro S, Heinzelman W. Prolonging the lifetime of wireless sensor networks via unequal clustering. In: Proceedings of the 19th IEEE international parallel and distributed processing symposium (IPDPS); April 2005.
[12] Li CF, Ye M, Chen GH, Wu J. 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 conference (MASS), Washington, DC; 7–10 November 2005.
[13] Chen G, Li C, Ye M, Wu J. An unequal cluster-based routing protocol in wireless sensor networks. Wireless Networks J 2009;15(2):193–207.
[14] Jiguo Yu, Yingying Qi, Gang Wang, Qiang Gu, Xin Gu. An energy-aware distributed unequal clustering protocol for wireless sensor networks. Int J Distrib Sensor Network 2011:8. http://dx.doi.org/10.1155/2011/20214. Article ID 202145.
[15] Gong Bencan, Li Layuan, Wang Shaorong, Zhou Xuejun. Multihop routing protocol with unequal clustering for wireless sensor networks. IEEE; 2008.
[16] Luchmun Ever R, Mostarda L, Navarra A, Shah P. UHEED – an unequal clustering algorithm for wireless sensor networks. In: Proceedings of the 1st international conference on sensor networks (SENSORNETS 2012). Rome, Italy; February 24–26. pp. 185–93.
[17] Yue EverJun, Zhang Weiming, Xiao Weidong, Tang Daquan, Tang Jiuyang. A novel unequal cluster-based data aggregation protocol for wireless sensor networks. Przegląd Elektrotechniczny, R. 89 NR 1b/2013, 20–24.
[18] Gu Xin, Yu Jiguo, Yu Dongxiao, Wang Guanghui, Lv Yuhua. ECDC: an energy and coverage-aware distributed clustering protocol for wireless sensor network. Comput Electric Eng 2014;40:384–98.
[19] Zahmati AS, Abolhassani B. An energy-efficient protocol with static clustering for wireless sensor networks. World Acad Sci, Eng Technol 2007:69–72.
[20] Zhu Xiaorong, Shen Lianfeng, Yum Tak Shing Peter. Hausdorff clustering and minimum energy routing for wireless sensor networks. IEEE Trans Vehicular Technol 2009;58(2):990–7. February.
[21] Ferng Huei Wen, Tendean Robby, Kurniawan Arief. Energy efficient routing protocol for wireless networks with static clustering and dynamic structure. Springer J Wireless Personal Commun 2001 (published online), March.
[22] Kumar Ashok, Kumar Vinod, Chand Narottam. Energy efficient clustering and cluster head rotation scheme for wireless sensor networks. Int J Adv Comput Sci Appl 2011;3(5):129–36.
[23] Zhang P, Sadler CM, Lyon SA, Martonosi M. Hardware design experiences in ZebraNet. In: Proceedings of the 2nd international conference on embedded networked sensor systems, Baltimore, MD, USA; 3–5 November 2004. pp. 227–38.

[24] Sadler CM, Martonosi M. Data compression algorithms for energy-constrained devices in delay tolerant networks. In: Proceedings of the 4th international conference on embedded networked sensor systems, Boulder, CO, USA; 31 October–3 November 2006. pp. 265–78.
[25] Arici T, Gedik B, Altunbasak Y, Liu L. PINCO: A pipelined in-network compression scheme for data collection in wireless sensor networks. In: Proceedings of IEEE The 12th international conference on computer communications and networks, Dallas, TX, USA; 20–22 October 2003. pp. 539–44.

Malathi L is research scholar at Anna University, India and working as a Assistant Professor in Vivekanandha College of Engineering for Women, Tiruchengode, India. She completed her Bachelors of Engineering degree in Information Technology during 2003 and the Master of Engineering degree in Computer Science and Engineering department at Anna University, Chennai during the year 2007. Her Research Interest includes Wireless Sensor Networks, Wireless Adhoc Network and Network Security.

Gnanamurthy R.K received his Ph.D in Faculty of Information and Communication Engineering from Anna University, India and M.E degree in Microwave and Optical Engineering from Madurai Kamaraj University, India. Currently, he is a professor and Head at SKP Engineering College, Tiruvannamalai, Tamilnadu, India. He has published more than 100 papers in various international journals and conferences. His research interests are related to signal processing, pattern recognition, adhoc networks and Wireless Sensor Networks.

Chandrasekaran Krishnan received his Ph.D in Computer Science and Engineering during the year 2013. He is working as a Professor and Head in the department of Computer Science and Engineering in Angel College of Engineering and Technology, Tirupur, India. His research interests are related to Wireless Networks, Network Security and Expert Systems.