Wireless Pers Commun (2013) 73:731–751
DOI 10.1007/s11277-013-1213-x

Achieving Energy Conservation by Cluster Based Data Aggregation in Wireless Sensor Networks

S. Siva Ranjani · S. Radha Krishnan · C. Thangaraj · K. Vimala Devi

Published online: 29 May 2013

Abstract In Wireless Sensor Networks (WSN), one of the major issues is to maximize the network lifetime. Since all sensor nodes directly send the data to the Base station, the energy requirement is very high. This reduces the lifetime of the network. One of the solutions is to partition the network into various clusters which avoids direct communication. In this paper we propose an Energy efficient Cluster Based Data Aggregation (ECBDA) scheme for sensor networks. In this algorithm, Cluster members send the data only to its corresponding local cluster head, there by communication overhead is reduced. Data generated from neighboring sensors are often redundant and highly correlated. So the cluster head performs data aggregation to reduce the redundant packet transmission. In our approach, clusters are formed in a non-periodic manner to avoid unnecessary setup message transmissions. Re-clustering is performed only when CH needs to balance the load among the nodes. The simulation results show that our approach effectively reduces the energy consumption and hence the network lifetime is also increased.

Keywords Wireless sensor networks · Aggregation · Energy efficiency · Cluster · Layer

S. Siva Ranjani (✉) · S. Radha Krishnan
Department of Computer Science and Engineering, Kalasalingam University,
Krishnankoil 626 126, Tamil Nadu, India
e-mail: s.sivaranjani@klu.ac.in

S. Radha Krishnan
e-mail: srk@klu.ac.in

C. Thangaraj
Anna University of Technology Chennai, CPT Campus, Tharamani,
Chennai 600 113, Tamilnadu, India
e-mail: vc@annatech.ac.in

K. Vimala Devi
Department of Computer Science and Engineering, P.S.R. Engineering College,
Sivakasi, Tamilnadu, India
e-mail: k.vimaladevi@gmail.com

1 Introduction

Wireless sensor networks (WSN) have wide applications in a broad spectrum of areas including military, environment and medical systems. Sensor node is small, inexpensive, self-contained and battery-powered device. Sensor node includes the sensing module, data processing module and communication module. Sensing component may vary based on the application. Since most of the applications use the sensor nodes to monitor the remote field, there is no chance for maintenance and battery replacement. The lifetime of a sensor node is determined by its remaining battery power which is significantly affected by the communication among nodes. In common sensor network scenario, data is collected by the sensor nodes from the area it is placed. The neighboring sensor nodes have the redundant or correlated data. It is not necessary to transfer the redundant data to the base station. Data generated by different sensor nodes can be jointly processed when forwarding towards the sink. Generally sensor nodes have the limited resources. Major energy is needed for data transfer between nodes. Hence, we need methods for combining data to avoid the redundant transmission at intermediate sensor nodes which results in reduced energy consumption and bandwidth [1]. This can be accomplished by the cluster based data aggregation method [2–5]. Aggregation functions may be classified according to the representation of intermediate results and the function properties.

Classification of aggregation functions are [6],

- *Duplicate sensitive*—If the aggregation result changed when the measured value of a particular device is used more than once in an aggregation function then it is said to be duplicate sensitive aggregate function. Otherwise it is duplicate insensitive.
- *Summary or exemplary*—An exemplary aggregate is a single, in some sense representative, value out of a set of values. A summary aggregate is a function of the entire set and does not depends on the individual values.
- *Composable*—An aggregation function f is said to be composable if the result of f applied to a set W of measurements can be computed by applying f to some portions of $W = W1 \cup W2$, using a helper function g. Formally,

$$f(W) = g\left(f(W1), f(W2)\right)$$

The large network field is divided into set of clusters. Each cluster has a Cluster Head (CH). All cluster members communicate to its CH. The CH will aggregate the data and transfer it to Base Station (BS). Clustering can be classified into two models: Static clustering and dynamic clustering [7]. Cluster formation is carried out for only one time in static clustering model. Any Protocol which follows the static clustering approaches uses the fixed clusters, whereas in dynamic clustering, cluster are modified over a period of time. This helps to avoid the overload on single CH for the whole network life time. Static clustering is suitable for heterogeneous networks and dynamic clustering is suitable in homogeneous networks. In this approach we use the homogeneous wireless sensor networks so we use the dynamic clustering model. Energy efficient Cluster Based Data Aggregation (ECBDA) uses a novel approach to form the clusters and choose a right node as a CH to perform the data aggregation. ECBDA split the sensing region into set of layers and from each layer it forms K clusters by using the K-means clustering method. From each cluster, one node is elected as a CH. All cluster members transfers its data to its corresponding CH by using TDMA MAC protocol. The CH performs the data aggregation there by it removes the redundancy and forward the aggregated packet to the BS via other CHs as the intermediate nodes. Table 1 provides the illumination of the symbols used in our approach.

Table 1 List of symbols

Symbol	Explanation
BS	Base station
CH	Cluster head
WSN	Wireless sensor network
N	Number of sensor nodes
K	Number of clusters
$E_{tx}(b)$	Energy to transmit b bits of data
E_{elec}	Transmission power
d	Distance
E_{amp}	Amplification power
$E_{rx}(b)$	Energy to receive b bits
L	Number of layers
R	Network region
Σ	Maximum distance
sn	Sensor node
T_r	Radio transmission range
$n(l)$	Number of nodes in lth layer
Kl	Number of clusters in lth layer
E_c	Communication energy
E_r	Residual energy
P_{CH}	Probability to become a cluster head
t_{round}	Time for a round
t_{tf}	Time for TDMA frame
μ	Aggregation energy
$n(C)$	Number of nodes in the cluster C
mn	Member node of a cluster
CH_{nh}	Next hop node of a cluster head
ACK	Acknowledgement
N_{init}	Initial number of nodes in a cluster
N_{curr}	Number of nodes alive currently
E_{th}	Threshold energy
E_{cost}	Edge cost

The remainder of this paper is organized as follows: Sect. 2 deals with the related works. Section 3 shows the system model. Section 4 illustrates the details of ECBDA algorithm. In Sect. 5, we discuss the simulations. Finally we conclude this paper in Sect. 6.

2 Related Works

Cluster based data aggregation [2–5] is addressed by many researchers. Clustering is one of the key solutions to achieve the energy efficiency in sensor networks. Sensor nodes are organized into clusters where each cluster is having a "cluster head" as the leader. The communication within a cluster must happen via the cluster head, which is then forwarded to the base station.

Low Energy Adaptive Clustering Hierarchy (LEACH) [9] is one of the initial and popular clustering algorithms. The operation of LEACH is divided into various rounds. Each round begins with a set-up phase in which the clusters are organized. It is followed by a steady-state phase where data are transferred from the nodes to the cluster head and then to the BS. Nodes with higher energy should be cluster head more often than the nodes with less energy. This can be achieved by setting the probability (*Pi(t)*) of becoming a cluster head as a function of a node's energy level relative to the aggregate energy remaining in the network. All cluster members use the CSMA MAC protocol to transfer the data to its CH. Non-cluster-head nodes will turn off their RF transceiver completely except during their TDMA time slot. However, Clusters are not evenly distributed due to its randomized rotation of local cluster-head.

Due to the poor performance of LEACH distributed algorithm, the same authors have introduced the LEACH centralized (LEACH—C) [9] algorithm. In this protocol, cluster formation is performed in a centralized approach. All the nodes send their remaining energy details and location to the BS. The BS uses the simulated annealing algorithm for cluster formation in the set up phase. Steady state phase is identical to LEACH protocol.

Sunhee Yoon et al. [10] proposed Clustered AGgregation (CAG) technique that forms cluster of nodes sensing similar values within a given threshold (spatial correlation), and these clusters remain unchanged as long as the sensor values stay within a threshold over time (temporal correlation). CAG ignores the redundant data and quickly generate a summary of the data distribution with significant energy savings. CAG works in two modes: interactive and streaming. In the interactive mode, CAG generates a single set of responses for a query. In the streaming mode, periodic responses are generated in response to a query. The interactive mode of CAG exploits only the spatial correlation of sensed data. The streaming mode of CAG takes the advantage of both spatial and temporal correlations of data. Cluster size estimation enables CAG to assign appropriate weights to the cluster head values while computing aggregates. Together, these two techniques enable CAG to compute results efficiently with high accuracy and guarantee that the results are within the user-provided thresholds regardless of data distribution. CAG can save a significant number of transmissions by avoiding global communications and adjusting clusters only using local communications.

Chang and Kuo [11] proposed MECH (Maximum Energy Cluster Head) protocol. It has self-configuration and hierarchical tree routing properties. MECH propose a hierarchical routing scheme to improve the disadvantages of LEACH. In this scheme the cluster constructing method was changed. Cluster constructing method avoids the uneven member distribution for clusters. The hierarchical routing scheme avoids the long range direct communication between cluster-heads and the base station. MECH constructs clusters based on radio range and the number of cluster members. Each round is divided in three phases: Setup phase, Steady phase and forwarding phase. MECH outperforms LEACH by a more balanced cluster distribution and by reducing the non-uniform cluster topology. Cluster members communicate its cluster-head via hierarchical tree. Time synchronization between the nodes is the major problem in MECH.

Time synchronization is the major problem in all distributed systems like wireless sensor networks. Sivrikaya and Yener [12] presents a survey on time synchronization in the field of wireless sensor networks. Reasons for addressing the synchronization problem are: (1) Sensor nodes need to coordinate their operations and collaborate to achieve a complex sensing task. (2) Synchronization can be used by power saving schemes to increase network lifetime. For example, sensors may sleep at appropriate times, and wake up when necessary. (3) Scheduling algorithms such as TDMA can be used to share the transmission medium in the time domain to eliminate transmission collisions and conserve energy. Thus, synchronization is an essential part of transmission scheduling. Time synchronization problem is addressed and solved in many research works [13–16]. Jabbarifar et al. [15] proposed a cluster based L-SYNC method. It uses linear regression method to calculate clock offset and skew. So it can estimate the local time of remote nodes in future and past. But L-SYNC is suffered with the cluster formation overheads.

K-Means [17–19] is the conventional method to form the clusters. Most of the existing works [17,18] use the centralized algorithm which takes more time to form the K clusters. Clusters are formed for each round in many existing techniques [20–22]. It dissipates the energy by broadcasting the cluster setting messages in each round.

Min et al. [23] have performed cluster formation by layering approach. The authors assumed that the base station is in the center of the sensing area but this assumption is not suitable for all the applications. In most of the applications like forest fire detection, base station node is kept far away from the sensor field. Clustering method proposed in [23] is not suitable when the base station is not in the middle of sensing field. Our proposed method works well for any network scenario. Figure 1a, b show the two different kinds of network scenario.

3 System Model

3.1 Network Model

ECBDA algorithm makes use of the following network properties.

- N numbers of sensor nodes are randomly distributed in a region R.
- All nodes have the same initial energy, equal communication range and sensing capability.
- All nodes have the sufficient processing power to support the protocol.
- BS knows the location and distance of all sensor nodes deployed in the field.

3.2 Energy Model

We use the first order radio model [8].

- RF signals are received and transmitted through the radio transceiver.
- ECBDA uses the radio transceiver with four states namely Transmit state, Receive state, Idle state and Sleep state.
- Energy required to transmit b bits of data from node i to node j is,

$$E_{tx}(b_{i,j}) = (E_{elec} \times b) + (d(i,j)^2 \times E_{amp} \times b) \tag{1}$$

where E_{elec} is the power for transmitter or receiver circuit, $d(i,j)$ is the distance between node i and j. E_{amp} is the power required for the amplifier circuit.

$$E_{amp} = \alpha_{amp} + \beta_{amp}$$

α_{amp} & β_{amp} are constants which are depends upon the amplifier architecture.

- Energy required to receive b bit data is,

$$E_{rx}(b) = E_{elec} \times b$$

4 Cluster Based Data Aggregation

ECBDA consists of five phases' viz. Cluster formation, Cluster head election, Cluster head route discovery, Data aggregation and Maintenance. Cluster formation phase is used to split the network into number of clusters. The network terrain area is divided into layers based on the communication range. K clusters are formed in each layer and the Cluster Head Election phase has to be executed. From each cluster, one node is selected as a CH by using its residual energy and the communication cost factor. Once a node is elected as a CH, it broadcasts the CH message to its cluster members, to other CHs and to BS. Then the CH assigns a TDMA time slot to each cluster member. Route discovery phase is used to establish the communication path between each CH and the BS. Actual data forwarding is performed in the fourth phase. In the Data aggregation phase, all cluster members send its sensed data to their CH during its allotted time slot. The CH will wait until its TDMA frame ends. After receiving the data from all its cluster members, CH starts the aggregation process. Each CH forwards the aggregated packet to BS via the forwarding nodes. Maintenance phase check the CH's residual energy at each round. If the residual energy is less than the required threshold value then new cluster head will be elected from the same cluster. Re-clustering is also performed in the maintenance phase. The flow diagram of ECBDA is given in Fig. 2.

4.1 Cluster Formation Phase

BS initiates the cluster formation phase in which the whole network is divided into L layers. Then each layer is divided into set of clusters. In the network region R, σ is the distance of a node which is far away from the BS

$$\sigma = \max\left(\overset{N}{\underset{i=1}{\forall}} d(sn_i, BS)\right)$$

where $d\,(sn_i, BS)$ is distance between the BS and the sensor node sn_i. To provide the communication between the CHs in two adjacent layers, BS uses the minimum radio transmission range to split the network into layers. Number of layers L depends upon this σ and the minimum radio transmission range, T_{r-min} of the sensor node.

$$\therefore L = \frac{\sigma}{T_{r-\min}}$$

$$Let\, l = \left\lfloor \frac{d(sn_i, BS)}{T_{r-\min}} \right\rfloor; \quad sn_i \in l^{th}\, layer; \quad where\, 1 \geq l \leq L$$

The node with minimum distance belongs to the closer layer. One hop distance node belongs to first layer, two hop distance node belongs to second layer and l hop distance node belongs to lth layer. Here the hop distance is based on the minimum transmission range of the node. As a result, all N nodes will fit into any one of the L layers. Now each layer l is further divided into K clusters. Since all CH reach the BS via the intermediate CHs, the CHscloser

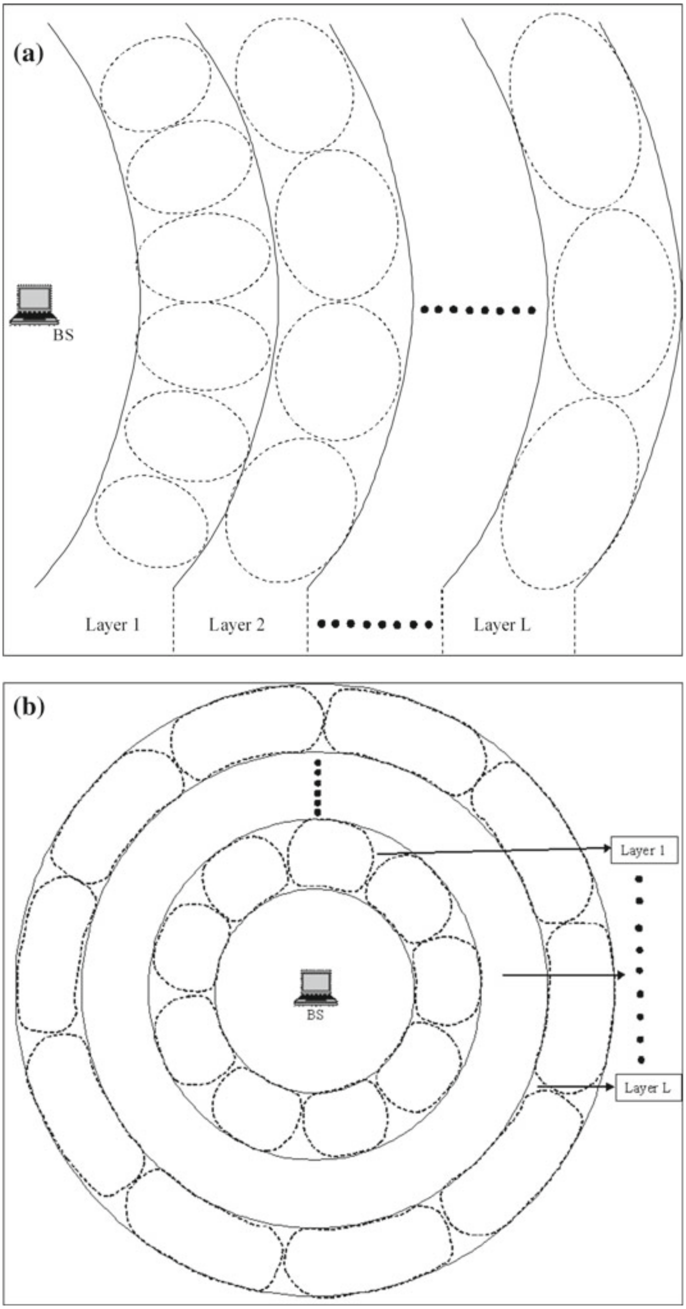

Fig. 1 Sensor network scenario. **a** BS is away from the sensing field. **b** BS is inside the sensing field

to the BS need to be in transmit state. So the CHs closer to the BS drain their energy soon. To overcome this issue we need to assign the K value cautiously. The layers which are closer to the BS have larger K, i.e. $K1 > K2 > K3 > \cdots > KL$. Number of cluster K is directly

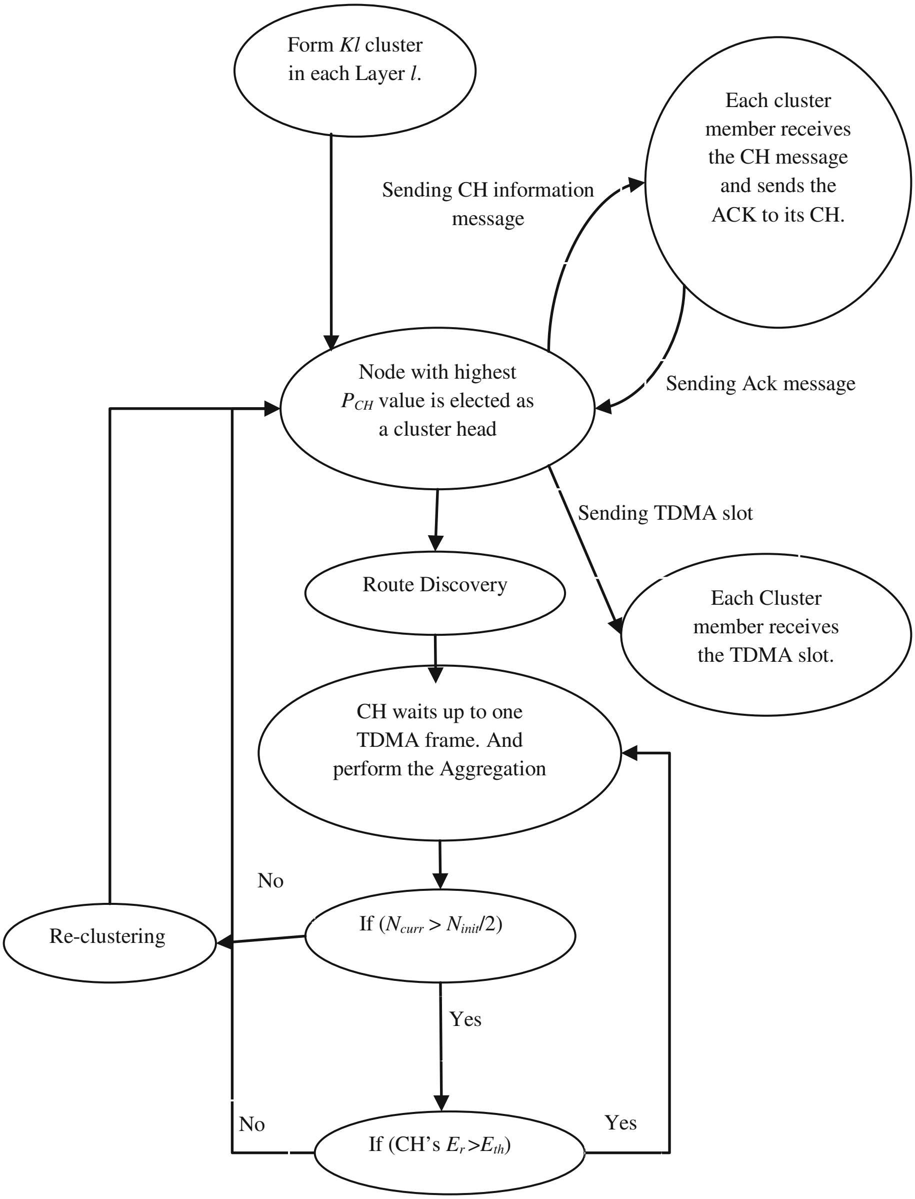
Form Kl cluster in each Layer l.
Sending CH information message
Each cluster member receives the CH message and sends the ACK to its CH.
Node with highest P_{CH} value is elected as a cluster head
Sending Ack message
Sending TDMA slot
Route Discovery
Each Cluster member receives the TDMA slot.
CH waits up to one TDMA frame. And perform the Aggregation
No
Re-clustering
If ($N_{curr} > N_{init}/2$)
Yes
No
Yes
If (CH's E_r>E_{th})

Fig. 2 ECBDA flow diagram

proportional to the layer density and inversely proportional to the layer number. It means that, if a layer is having more number of nodes, it should have more number of clusters. Layer with the smaller layer number should have the greater K and with the greater layer number should have the smaller K. According to these facts, the number of cluster K for the layer l is defined as:

$$Kl = \frac{n(l)}{(l \times L)}$$

where $n(l)$ is the number of nodes in layer l. In each layer l, Kl number of clusters will be formed by using the *K-means* algorithm. *K-Means* clustering generates a specific number of disjoint, flat clusters. The method for the cluster formation is presented in Algorithm 1. This cluster formation is carried out only once in the network and thus ECBDA reduces the message transmissions and increases the lifetime of network. But in the maintenance phase cluster setup is altered if the cluster has too small number of alive nodes. Once the clusters are finalized, ECBDA elect an optimum node as a cluster head.

Algorithm 1 ClusterFormation

```
/* BS initialize the clustering */
$\sigma = 0$
For $i = 1$ to $N$ do          /* N indicates total number of nodes */
    If ( $\sigma < d(sn_i, BS)$ ) then
        $\sigma = d(sn_i, BS)$
    End if
End for
/* Finding number of layers L */
$L = \sigma / T_{r\text{-}min}$          /* $T_{r\text{-}min}$ is the minimum radio transmission range */
For $l = 1$ to $L$ do
    $n[l] = 0$
End for
/* Finding Layers of each sensor node sn */
For $i = 1$ to $N$ do
    $layer[sn_i] = d(sn_i, BS) / T_{r\text{-}min}$
    $n[layer[sn_i]]{+}{+}$          /* Counting number of nodes in each layer */
End for
/* Forming k number of clusters in each layer l */
For $l = 1$ to $L$ do
    $Kl = n[l] / (l * L)$
    Apply K-means algorithm to form Kl number of clusters
End for
```

4.2 Cluster Head Election Phase

Cluster head is chosen based on the node's residual energy (E_r) and the energy to communicate (E_c) with other nodes in the cluster and the BS. Algorithm 2 represents the method for Cluster Head selection within a Cluster. Nodes in a cluster calculate their probability to become a cluster head (P_{CH}) as:

$$P_{CH}(sn) = 1 - \left(\frac{\left(\left(t_{round}/t_{tf} \right) \times (E_c + \mu) \right) + \sum_{j=1}^{n(C)} E_{tx}\left(b_{sn,mn_j} \right)}{E_r} \right)$$

where μ is the energy to perform the data aggregation. t_{round} is the time period of each round. t_{tf} is the TDMA frame period. So $\left(t_{round}/t_{tf}\right)$ is the total number of times that the CH needs to do the aggregation.

$$\sum_{j=1}^{n(C)} E_{tx}(b_{sn,mn_j})$$

Algorithm 2 FindClusterHead

```
/* Choose an optimum node as a CH for the cluster ‘C’ */
$maxProb = 0$
For $c = 1$ to $n[C]$ do          /* n[C] indicates number of nodes in the cluster ‘C’ */
        $P_{CH}[sn_c] = computeProb()$
        If ( $maxProb < P_{CH}[sn_c]$ ) then
                $maxProb = P_{CH}[sn_c]$
                $CH[C] = sn_c$
End if
End for
```

Function 1 computeProb()

```
Input :
$E_c$ : Total energy to communicate with the nexthop CH and energy to receive data from all cluster members.
$\mu$ : Aggregation Energy
$t_{round}$ : Time taken for each round
$t_{tf}$ : TDMA frame period
$E_{tx}$ : Energy to transmit TDMA slot
$E_r$ : Node’s residual energy
Output : Probability to become a CH
If ( $E_c < E_r$ ) then
$TTrEg = 0$        /* TTrEg indicates total transmission energy */
For $i = 1$ to $n[C]$ do
                $TTrEg$ += $E_{tx}(b(sn_i, mn_i))$
End for
$E_{round} = ( t_{round} / t_{tf} ) * ( E_c + \mu )$
$Prob = 1 – ( ( E_{round} + TTrEg ) / E_r )$
Else
        $Prob = 0;$
End If
Return $Prob$
```

is the total energy required to communicate with its cluster members (*mn*) to transmit the TDMA slots. $n(C)$ indicates the total number of nodes in the cluster. The node's residual energy E_r will get reduced for each b bit data transmission and reception. If a node *sn* wants to become the CH, then it should have the capability to communicate with its cluster members and also the next hop CH in order to transmit the aggregated data of the current round. If the node's E_r is less than its E_c value then that node cannot participate in the CH election.

$$E_c = \left\{ \begin{array}{l} Energy\, to\, transmit\, b\, bits \\ of\, data\, to\, its\, next\, hop \\ cluster\, head \end{array} \right\} + \left\{ \begin{array}{l} Energy\, to\, receive\, b\, bits\, of \\ data\, from\, its \\ cluster\, members \end{array} \right\}$$

$$E_c = E_{tx}(b_{sn,CH_{nh}}) + \sum_{j=1}^{n(C)} E_{rx}(b)$$

From the Eq. (1), Energy to transmit b bits of data to the next hop cluster head (CH_{nh}) is,

$$E_{tx}(b_{sn,CH_{nh}}) = (E_{elec} \times b) + (d(sn, CH_{nh})^2 \times E_{amp} \times b)$$

and energy to communicate with its cluster members (*mn*) is,

$$E_{tx}(b_{sn,mn}) = (E_{elec} \times b) + (d(sn, mn)^2 \times E_{amp} \times b)$$

$n(C)$ is the number of nodes in the C^{th} cluster of lth layer. *mn* is the member node of the cluster. The node which is having highest P_{CH} becomes the CH for its cluster. Route discovery phase is used to establish the routing between the CH and the BS.

The CH allocates the TDMA slots for all its cluster members to avoid collision. On the reception of TDMA slot, all cluster members identify its CH and its time slot. Member nodes send the data to its CH only at the allocated TDMA slot. CH will wait for one TDMA frame to receive the data from its cluster members. Time synchronization is the major problem in using TDMA technology. In order to provide the time synchronization between the CH and the cluster members we are using the energy efficient time synchronization protocol [15]. All cluster members are needed to be in active state only in its TDMA slot. In the remaining time they will be in sleep state. As a result, ECBDA increases the lifetime of the network

4.3 Cluster Head Route Discovery Phase

We propose the energy efficient routing between the CHs and the BS. Once CHs are elected, all CHs send the information about the remaining energy to the BS in the route request message. BS sends the route reply after it finds the shortest path to each CH by using dijkstra's algorithm [24].

Let the CHs and the BS are represented as a connected graph G (V, E), where vertex set V ∋ (CHs, BS) and edge set E represents the connection between the CHs. Weight of the each edge in the edge set E is defined as the function of load and energy requirement of the link. The edge cost, Ecost of the edge between two vertices i and j is,

$$E_{\cos t}(i, j) = \frac{\Delta \times E_{tx}(i, j)}{E_r(i)}$$

E_{tx}(i, j) is the energy required to transfer the data from the node i to j. Δ represents the load of the node i and the load is defined as the in degree of the node i. To find the minimum cost connection between the source vertex BS and all vertices and the CHs, we have used the standard dijkstra's algorithm to set up the shortest path based on the edge weight. Once the path is confirmed, the BS send the route reply to each CHs separately. By receiving the reply messages all CHs will identify its next hop neighbor to reach the BS.

4.4 Data Aggregation Phase

Data aggregation is defined as the process of aggregating the data from multiple sensors to eliminate redundant transmission. In this paper, the CH works as the aggregator node. All cluster members send the monitored information to its CH during its TDMA slot allotted to them. Each CH will wait for one TDMA frame to collect the information from its member nodes. After each TDMA frame, CH will aggregate the received information and forward the aggregated data to the base station. In this work, we used the max function as the aggregation function in the CH.

In existing techniques, CHs directly transfer the data to the base station node. Energy for the direct transmission is high. But in ECBDA, each CH will make use of its next hop CH as a forwarding node to base station as a result the transmission energy is reduced.

4.5 Maintenance Phase

ECBDA evenly distribute the load for all nodes in the cluster. If the cluster head is static, then that head node will always dissipate more energy and it will die quickly. So ECBDA uses this maintenance phase to change the cluster head over a time period. The algorithm for the cluster maintenance is presented in Algorithm 3. This algorithm is executed for each round. If the total number of nodes alive (N_{curr}) in a cluster is less than 50 % of its initial number of nodes (N_{init}), then that cluster members join with other closer clusters in the same layer. This process of modifying the cluster setup is called as *Re-clustering*. Figure 3 shows the example scenario of a cluster setup. Initially cluster B has 18 nodes and if suppose after some rounds total number of alive nodes is 8. When the cluster B enters into the maintenance phase, it is identified that its N_{curr} is less than its $N_{init}/2$. So the members of cluster B join with the closer clusters A and C. Figure 4 shows this re-clustering scenario. If a node is joining with a closer cluster, then the number of nodes of the new cluster will be increased. At each round the algorithm checks the cluster head's remaining energy Er with the threshold value Eth.

$$E_{th} = \left(\frac{t_{round}}{t_{tf}}\right) \times (E_c + \mu) + \sum_{j=1}^{n(C)} E_{tx}(b_{sn,mn_j})$$

t_{round} is the time period of each round. t_{tf} is the TDMA frame period. E_{th} is calculated based on the number of times the CH node needs to perform the aggregation process. If E_r is greater than E_{th} then the same node can continue as a CH for the next round. If E_r is less than E_{th} then the cluster enters into the cluster head selection phase and route discovery phase. Another node with high probability will be elected as the cluster head for the cluster. Since the load is balanced among the cluster members, ECBDA improves the lifetime of the network.

5 Simulation Study

In this section, we discuss about the simulation study and performance evaluation of the proposed technique. Castalia 2.3 is used for simulation to evaluate our proposed method. Table 2 shows the simulation parameters.

Algorithm 3 MaintainCluster

```
/* Balancing the load of all nodes in the cluster ‘ C’ */
If ( $N_{curr} < N_{init} / 2$ ) then
    reClustering ( Cluster C )
else
    If ( $E_r < E_{th}$ ) then
        Call FindClusterHead algorithm
    End if
End if
```

Function 2 reClustering(Cluster C)

```
Input :
List of cluster members and its radio range
Cluster heads of 2 closer clusters (C-1) & (C+1) in the same layer l
For $i = 1$ to n(C) do
    If ( $d(sn_i , CH(C-1)) < d(sn_i , CH(C+1))$ ) then
        $Sn_i$ joins with Cluster (C-1)
        n(C-1)++
    Else
        $Sn_i$ joins with Cluster (C+1)
        n(C+1)++
    End if
End for
```

We use the random deployment model for the sensor network topology setup. BS is placed in the location which is away from the sensor field. Random deployment of sensor network is shown in Fig. 5. We have compared our results with LEACH and LEACH—C [9] protocols. Energy is one of the major issues in wireless sensor network. Figure 6 shows the energy consumption of the network. In LEACH protocol, clusters are not evenly distributed. So the

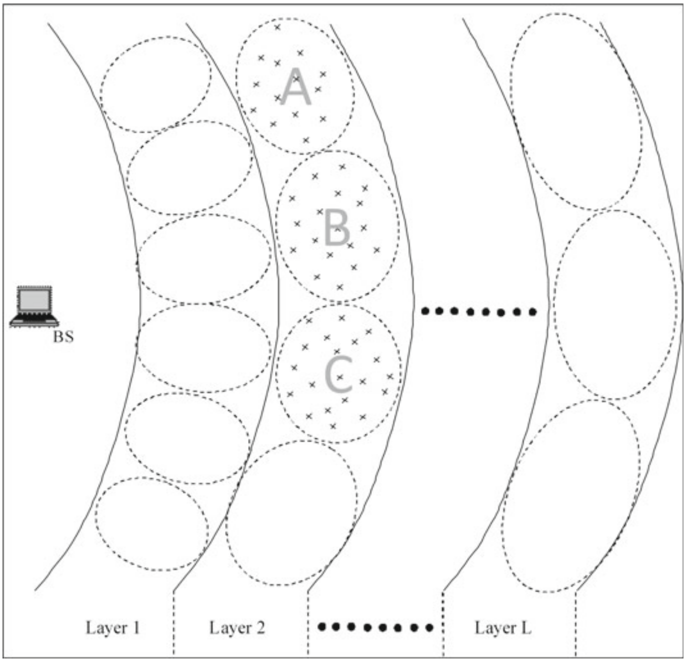

Fig. 3 Initial cluster setup of *A*, *B* and *C*

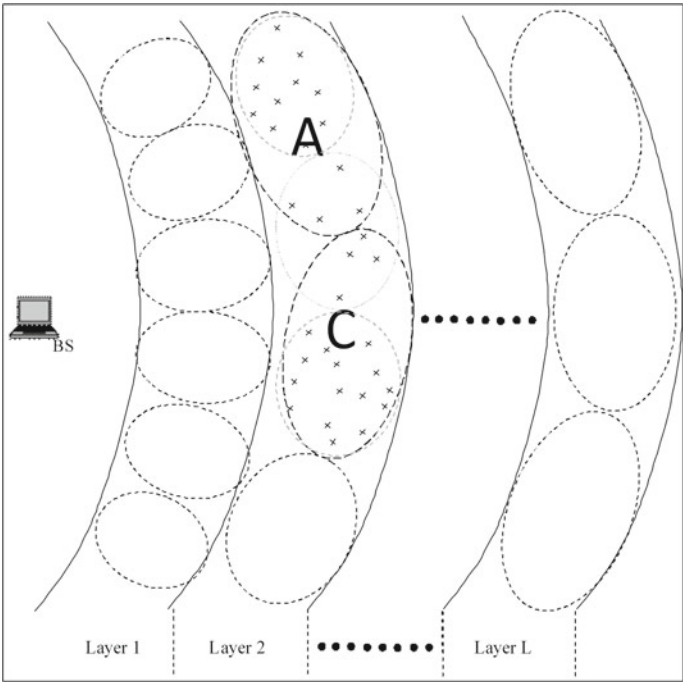

Fig. 4 Re-clustering after maintenance phase cluster B in "Fig. 3" is merged with cluster *A* & *C*

Table 2 Simulation parameter setting

Parameter	Value
Number of nodes	100
Network field	100 m × 100 m
Initial energy	50 J
Receive power	22.2 mW
Transmit power	31.2 mW
Sleep power	0.0006 mW
Data packet size	40 bytes
Noise bandwidth	30 KHz

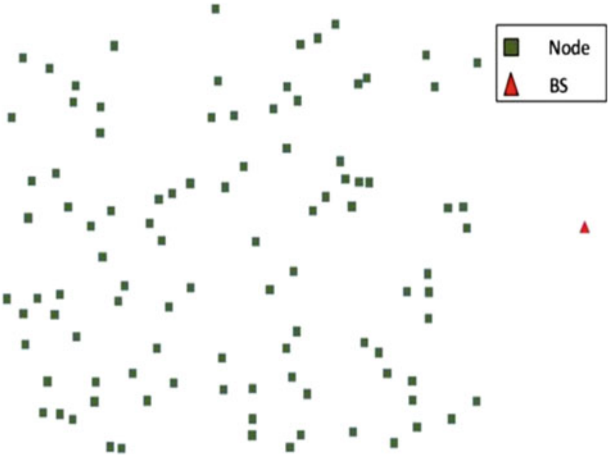

Fig. 5 Topology of WSN

cluster members spent more energy to transmit a packet to its cluster head. In LEACH—C clusters are evenly distributed but at each round all nodes sent its information to the central BS and it is more expensive. But in our method, since clusters are formed based on the K-means algorithm, all the clusters are evenly distributed and hence the cluster members are closer to its cluster head. So the energy consumption for ECBDA is less compared to LEACH protocols.

In our method, the cluster formation is performed only when it is necessary. But in LEACH and LEACH—C, cluster heads are elected for each round. Based on the new cluster head, clusters are formed. So it consumes additional energy for transferring the setup messages. In our approach, clusters are updated only if necessary. In direct transmission, there is no clustering in the network so the energy consumption is high.

From Fig. 7, the average packet error rate in ECBDA is comparatively lower than LEACH and LEACH—C. Packet Error Rate depends on the number of incorrect packets received at the BS. If any one of the data bit is incorrect then that packet is treated as the incorrect packet. In many applications, sensor nodes are deployed in the harsh environment. So the noise level will be high. We use the max function as the aggregation function. In the direct approach, all the nodes directly transfer the data to the BS. BS finds the maximum value from the received packets. Because of the high network congestion in direct approach, it loses many packets

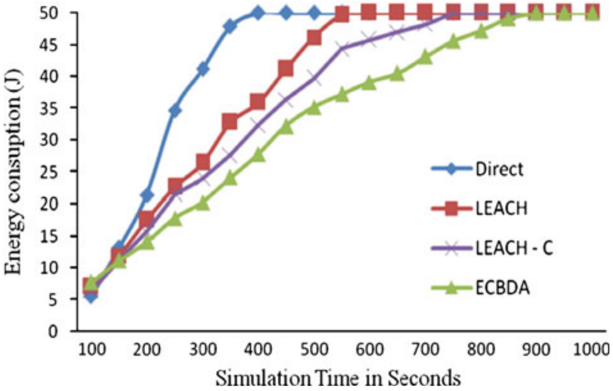

Fig. 6 Comparison of energy consumption

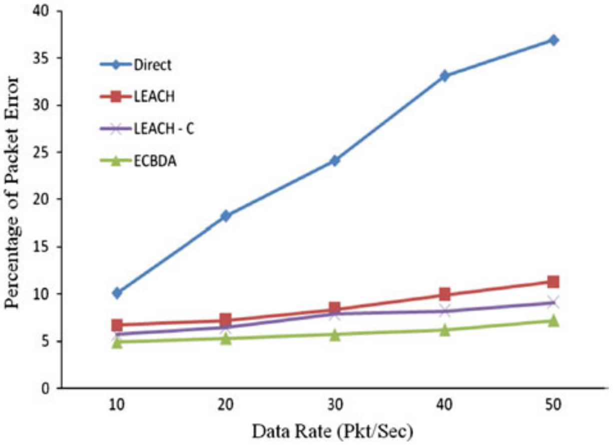

Fig. 7 Comparison of accuracy

and the max function is performed only with the received values which produces the error result. In direct communications the effect of noise is high. So packet error is increased in direct approach.

In LEACH and LEACH—C protocols, all cluster head nodes directly transfer its aggregated data to the sink node. So the error rate for the cluster heads more than 4 hop distances is high. But in our method, cluster heads which are far away from sink use its intermediate cluster heads as a next hop node to forward the data to the sink node. Since ECBDA uses shortest path to transmit the data, the error rate is controlled. In LEACH and LEACH—C, the CH may die during the period it collects the data from its cluster members. So cluster members elect new CH but the previous CH data are lost. So the data error is increased in LEACH and LEACH—C. But in our approach, a node becomes a CH only when it has the capability to do the aggregation for the entire round in order to compute the aggregated data without any error.

Figure 8 shows the percentage of packets successfully delivered to the sink. In LEACH and LEACH—C, many intermediate nodes drain its total energy and the communication becomes too weak hence the packet loss is more.

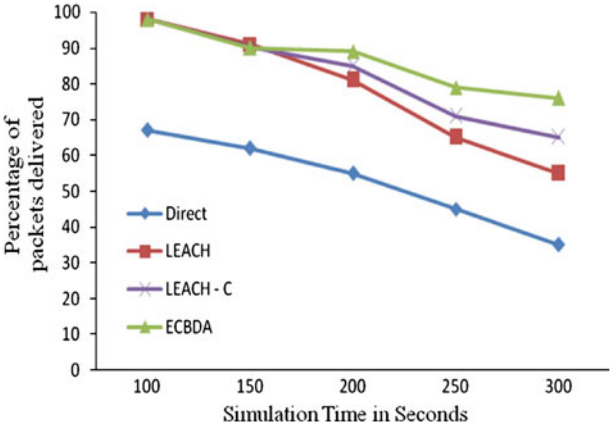

Fig. 8 Comparison of packet delivery ratio

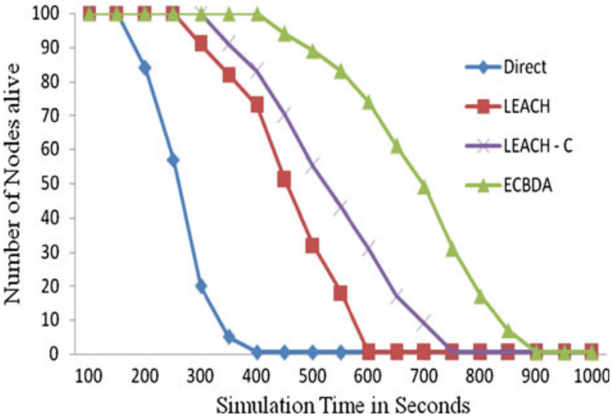

Fig. 9 Comparison of network lifetime

We have also shown that the network life time is improved in ECBDA. In all phases of ECBDA, energy is considered as a major source. Figure 9 shows the number of nodes alive. LEACH and ECBDA are same up to 280 s but after 280 s nodes using LEACH started going out of power. LEACH—C is little better than LEACH because of the right selection of CH. But still rapid node death occurs in LEACH and LEACH—C because Cluster formation phase is executed for each round which consumes more amount of energy. Cluster heads far away from base station will lose its energy quickly. So the LEACH and LEACH—C lose its nodes 30 and 25 % faster than ECBDA respectively. Our proposed method significantly extends the lifetime of the network.

Delay is another vital issue in sensor network. Many applications like forest fire detection and patient monitoring systems need the immediate report to the sink. Figure 10 shows the end to end average delay. With low congestion direct method is performing well but when the data rate is increased then ECBDA out performs LEACH, LEACH—C and Direct transmission. When the data packet generation rate is increased, then automatically congestion is also increased in direct transmission. Direct approach gets higher delay because of high congestion. Various factors such as time needed to setup the cluster for each round and

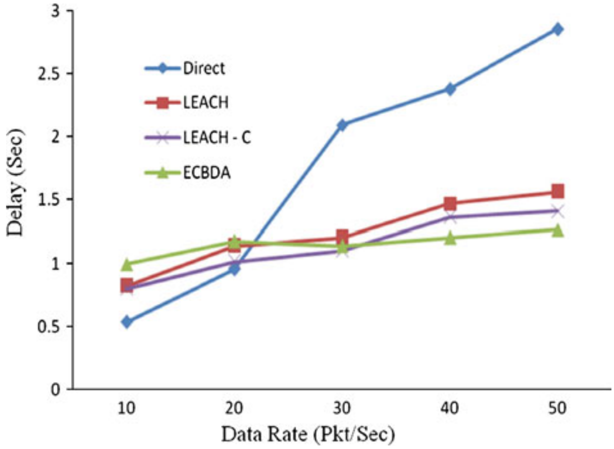

Fig. 10 Comparison of average delay

high communication loss due to frequent node death, increases the LEACH and LEACH—C protocols' average delay.

6 Conclusion

In this paper, we have studied the energy efficient data transmission in wireless sensor networks. We have proposed a cluster based data aggregation method. Our proposed method consists of five phases. As clustering process is performed only when it is necessary, the setup message transmission is reduced. Cluster heads are elected in an optimum way. A node can become the cluster head only when it has the higher capability to communicate with its cluster members throughout the entire round. Results of our simulation study indicate that the proposed method provides the energy efficiency and the network lifetime is also increased compared to the existing methods. This work can be enhanced by addressing the reliability and security [25] issues.

References

1. Dressler, F. (2007). *Self-organization in sensor and actor, networks* (pp. 185–202). New York: Wiley.
2. Rajagopalan, R., & Pramod, K. V. (2006). Data aggregation techniques in sensor networks: A survey. *IEEE Communications Surveys and Tutorials*, *8*(4), 48–63.
3. Abbasi, A. A., & Younis, M. (2007). A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, *30*(14–15), 2826–2841.
4. Tarng, W., Ou, K. L., Huang, K. J., Deng, L. Z., & Lin, H. W. (2011). Applying cluster merging and dynamic routing mechanisms to extend the lifetime of wireless sensor networks. *International Journal of Communication Networks and Information Security (IJCNIS)*, *3*(1), 8–16.
5. Hussaini, M., Bello-Salau, H., Salami, A. F., Anwar, F., Abdalla, A. H., & Islam, M. R. (2012). Enhanced clustering routing protocol for power-efficient gathering in wireless sensor network. *International Journal of Communication Networks and Information Security (IJCNIS)*, *4*(1), 18–28.
6. Karl, H., & Willig, A. (2007). *Protocols and architectures for wireless sensor networks*. New York: Wiley.
7. Bajaber, F., & Awan, I. (2008). Dynamic/static clustering protocol for wireless sensor network. *Second UKSIM European symposium on computer modeling and simulation* (pp. 524–529)

8. Heinzelman, W., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor, networks. In *Proceedings of the 33rd international conference on system sciences (HICSS'00)* (pp 1–10).
9. Heinzelman, W., Chandrakasan, A., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, *1*(4), 660–669.
10. Yoon, S., & Shahabi, C. (2007). The clustered AGgregation (CAG) technique leveraging spatial and temporal correlations in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN), 3*(1), Article 3.
11. Chang, R.-S., & Kuo, C.-J. (2006). An energy efficient routing mechanism for wireless sensor networks. In *20th international conference on advanced information networking and applications* (Vol. 2 (AINA'06), pp. 308–312).
12. Sivrikaya, F., & Yener, B. (2004). Time synchronization in sensor networks: A survey. *IEEE Network*, *18*(4), 45–50.
13. Wu, Y.-C., Chaudhari, Q. M., & Serpedin, E. (2011). Clock synchronization of wireless sensor networks. *IEEE Signal Processing Magazine*, *28*(1), 124–138.
14. Kim, H., Kim, D., & Yoo, S. (2006). Cluster-based hierarchical time synchronization for multi-hop wireless sensor networks. In *20th international conference on advanced information networking and applications* (Vol. 2, pp. 318–322).
15. Jabbarifar, M., Sendi, A. S., Pedram, H., Dehghan, M., & Dagenais, M. R. (2010). L-SYNC: Larger degree clustering based time-synchronisation for wireless sensor network. *SERA*, *2010*, 171–178.
16. Jabbarifar, M., Sendi, A. S., Sadighian, A., Jivan, N. E., & Dagenais, M. (2010). A reliable and efficient time synchronization protocol for heterogeneous wireless sensor network. *Wireless sensor networks*, *2*(12), 910–918.
17. Hua, M., Ki Lau, J., & Pei, K. Wu. (2009). Continuous K-means monitoring with low reporting cost in sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, *21*(12), 1679–1691.
18. Hui Li, X., & Ling Fang, K. (2009). A clustering algorithm based on K-means for wireless indoor monitoring system. *International Conference on Information Technology and Computer Science*, *1*, 488–492.
19. Datta, S., Giannella, C. R., & Kargupta, H. (2009). Approximate distributed K-means clustering over a peer-to-peer network. *IEEE Transactions on Knowledge and Data Engineering*, *21*(10), 1372–1388.
20. Patel, R. B., Aseri, T. C., & Kumar, D. (2009). EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks. *International Journal of Computer Communications, Elsevier*, *32*(4), 662–667.
21. Younis, O., & Fahmy, S. (2004). HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, *3*(4), 366–379.
22. Ishmanov, F., & Kim, S. W. (2009). Distributed clustering algorithm with load balancing in wireless sensor networks. *WRI World Congress on Computer Science and Information Engineering, CSIE 2009* (Vol. 1, pp. 19–23).
23. Min, X., Wei-ren, S., Chang-jiang, J., & Ying, Z. (2010). Energy efficient clustering algorithm for maximizing lifetime of wireless sensor networks. *AEU-International Journal of Electronics and Communications*, *64*(4), 289–298.
24. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. *Introduction to algorithms* (2nd ed. pp. 658–664). The MIT Press: Cambridge.
25. Alrajeh, N. A., Khan, S., Lloret, J., & Loo, J. (2013). Secure routing protocol using cross-layer design and energy harvesting in wireless sensor networks. *International Journal of Distributed Sensor Networks, 2013*, 11 pp. Article ID 374796.

Author Biographies

S. Siva Ranjani holds M.E. in Computer Science and Engineering from Anna University, Chennai, India, 2007. She is pursuing her Ph.D. degree in the area of Wireless Sensor Networks. Presently she is an Assistant Professor in department of computer science and engineering, Kalasalingam University, Krishnankoil, India. Her areas of interest include wireless sensor networks, data structure and algorithms and data mining.

S. Radha Krishnan holds a Ph.D. in Biomedical Engineering from Institute of Technology, Banaras Hindu University, India, 1992. Presently he is a senior professor in department of computer science and engineering, Kalasalingam University, Krishnankoil, India. His research areas are Routing Algorithms, Sensor Networks, Soft Computing Techniques, Mobile and Ad-hoc Networks. He has guided 12 Ph.D. Thesis, 10 M.Phil. Thesis and around 60 M.Tech Thesis. He has more than 40 publications to his credit. He is the life member of Computer Society of India, Indian Society of Technical Education and Society of Biomedical Engineering, India.

C. Thangaraj holds a Ph.D. in Civil Engineering from Indian Institute of Technology, Delhi, India, 1985. Currently he is the Vice Chancellor in Anna University of Technology, Chennai, India. His main research areas are Large Scale Systems, Parallel Programming, Remote Sensing and GIS and Water Resources Systems. He has published 45 papers in National and International Journals and Conferences. He has won a best Principal award, Seva Ratna Award, Hem Prabha-Gupta Award for best paper by Institution of Engineers (India). He is the life member of Indian Society for Technical Education and Member of American Society of Photogrammetry and Remote Sensing.

K. Vimala Devi is a Professor in the Department of Computer Science and Engineering, P.S.R. Engineering College, Sivakasi, Tamil Nadu, India. She received her Ph.D. degree during 2008, at the Department of Computer Science and Engineering, Anna University, Chennai, India. She did her MCA at the University of Madras and an M.E. in Computer Science and Engineering at College of Engineering, Anna University, Chennai. She has more than 15 years' experience in teaching and research. Her current research interests include computer networks, active networks and management, wireless sensor networks, Distributed and Cloud Computing. She is a life member of ISTE and editorial member for the IEEE Communications letter, International Journal of Network Management-John Wiley publications and for the Aircsse journals.