IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 8, 2018, accepted October 24, 2018, date of publication December 7, 2018,
date of current version December 31, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2882639

Cluster-Tree Routing Based Entropy Scheme for Data Gathering in Wireless Sensor Networks

WALID OSAMY[1], AHMED M. KHEDR[2,3], AHMED AZIZ[1], AND AHMED A. EL-SAWY[1]
[1]Computer Science Department, Faculty of Computers and Informatics, Benha University, Benha 13511, Egypt
[2]Computer Science Department, College of Sciences, University of Sharjah, Sharjah 27272, UAE
[3]Mathematics Department, Faculty of Science, Zagazig University, Zagazig 44519, Egypt

Corresponding author: Ahmed M. Khedr (akhedr@sharjah.ac.ae)

ABSTRACT Wireless sensor networks (WSNs) have captivated substantial attention from both industrial and academic research since last few years. The major factor behind the research efforts in the field of WSNs is their vast range of applications, such as surveillance systems, military operations, health care, environment event monitoring, and human safety. However, sensor nodes are low potential and energy constraint devices; therefore, energy efficient routing protocol is the foremost concern. In this paper, a new Cluster-Tree routing scheme for gathering data (CTRS-DG) is proposed that composed of two layers: routing and aggregation and reconstruction. In aggregation and reconstruction layer, a dynamic and a self-organizing entropy based clustering algorithm for cluster head (CH) selection and cluster formation is proposed. Data is aggregated and compressed at CHs based on compressive sensing technique. In routing layer, a new proposed algorithm to form the routing tree as backbone of the network is proposed. The routing tree is used to forward the compressed data by CHs to the base station (BS). Finally, as a phase of aggregation and reconstruction layer, an effective CS reconstruction algorithm called Bee based signal reconstruction (BEBR) is proposed to improve the recovery process at the BS. BEBR utilizes the advantages of the greedy algorithm and Bees algorithm to find the optimal solution of reconstruction process. Simulation results reveal that the proposed scheme outperforms existing baseline algorithms in terms of stability period, network lifetime, and average normalized mean squared error for compressive sensing data reconstruction.

INDEX TERMS Average normalized mean squared error, clustering-tree based, compressive sensing, entropy coefficient, bees algorithm, network lifetime, stability period, wireless sensor network.

I. INTRODUCTION AND MOTIVATION

A Wireless sensor network (WSN) consists of spatially distributed devices known as sensors. Each senor can perform some basic processing such as gathering sensory information and communicating with other connected sensors in the WSN [55]. Routing schemes can be classified based on the logical used topology into flat-based, cluster-based, chain-based and tree-based routing. In flat-based routing, messages are flooded to find the route between sensor nodes. In Chain-based routing, chain is constructed to connect all the nodes in WSN, where the successor node aggregates the received data from the predecessor with its data and then transmits the aggregated data to its successor toward BS. In tree-based protocols, a child parent relationship is established, where all nodes send their data to their parents and the parents re-transmit the received data after performing process on data (if possible) toward the root node or BS [12], [24], [35], [49]. In cluster-based protocols, nodes are organized into groups called clusters. In each cluster there is one CH that is responsible for receiving data from cluster members (CMs) and performs some operations on that data then forwards it direct or indirect to BS. Low-Energy Adaptive Clustering Hierarchy (LEACH) [17] is the first cluster-based protocol.

LEACH has many variant versions such as [18]–[22], [27], [39], and [41]. Salim *et al.* [39] proposed intra-balanced LEACH protocol to extend LEACH by balancing the energy consumption in WSNS. A heterogeneity-aware energy efficient clustering (HEC) algorithm is suggested in [41]. HEC selects CHs based on the three network lifetime phases. A distributed energy-efficient clustering scheme for heterogeneous WSNs (DEEC) is presented in [36]. To overcome the limitation of DEEC and the lifetime of the network, a Cluster-head Restricted Energy Efficient Protocol (CREEP) is proposed in [45]. A three level heterogeneous network model for WSNs is proposed in [44]. An enhanced developed distributed energy-efficient clustering (EDDEEC) with heterogeneous network model is proposed in [25] that is established on three energy levels of sensor nodes. Modified Enhanced Developed Distributed Energy-Efficient Clustering (MED-DEEC) algorithm has been presented in [42]. A distributed unequal cluster-based routing (DUCR) proposed in [30]. DUCR uses multi-objective optimization technique to assign CMs to an appropriate CH so that the load is balanced. The work in [16] proposed a distributed energy-efficient clustering protocol (DCE) for heterogeneous WSNs. DCE is based on a Double-phase CH Election scheme.

Panag and Dhillon [19] proposed dual head static clustering algorithm (DHSCA) to balance consumed energy of sensor nodes and prolong the lifetime of the network. In [20], a heterogeneous routing protocol based on the adaptive threshold sensitive distributed energy efficient cross layer routing protocol is proposed. Weighted probability is used to elect CH. In [21], a decentralized hierarchical cluster-based routing algorithm for WSNs is proposed. Clusters are formed in such a way that CHs have the most competency in forwarding task of intra-cluster and inter-cluster transmission tree. In [22], multi-hop communication protocol is proposed for finding optimal clusters particularly, when the sensing field is split into hexagonal and voronoi clusters. In [10], an energy efficient concentric circular clustering protocol (EECCCP) has been proposed. EECCCP divides network field into zones of concentric circular and considers energy heterogeneity normal and super nodes having flat topology while advance nodes having clustering topology. Enhance Threshold Sensitive Stable Election Protocol (ETSSEP) [43] is proposed in [43] for heterogeneous WSNs. In ETSSEP the level of residual energy and minimum number of clusters per round are the main parameters to select the CHs. The average energy of the network, remaining energy of nodes and distance between BS and nodes are the main parameters to elect the CHs as proposed in Distance based Enhance Threshold Sensitive Stable Election Protocol (DETSSEP) [38]. In general, clustering approach reduces the size of collected data by keeping only significant information by applying data aggregation techniques at CHs and it also reduces communication overheads and due to effective allocations of resource as a result, decrease the overall energy consumption and reduce the interference among sensor nodes [55].

There are some works that take the benefits of both cluster-based and tree-based techniques such as [5], [6], [11], [50], and [54]. In Cluster-Tree-based Data Dissemination (CTDD) [5], first it forms the clusters inside the grids, then it forms the tree structure over clusters, where, each CH is treated as a tree node in WSN. In [50], a cluster-tree data gathering algorithm (CTDGA) to reduce the energy consumption is proposed. In [11], a Velocity Energy-efficient and Link-aware Cluster-Tree (VELCT) scheme for data gathering in WSNs is proposed. VELCT constructs the Data Collection Tree (DCT) based on the CH location. VELCT reduces the energy exploitation and the end-to-end delay by effective usage of the DCT.

In the proposed scheme, we design two layers routing scheme for gathering data by taking the benefits of both cluster-based and tree-based techniques. The first layer is for data collection and aggregation based on clustering technique and the second layer for forwarding aggregated data at CHs using tree based technique. All previous works consider special attributes of sensor nodes, e.g., remaining energy, distance to BS, ... etc. as criteria for selecting CHs, ignoring the cluster load, i.e., the number of cluster members that can be served by CH or the number of sensor nodes that can be supported by the CH. In our proposed scheme, we consider cluster load as a criterion for selecting CHs combined with the remaining energy, distance to BS and the degree of neighboring sensor nodes. Moreover, we introduce a new method to predict the remaining energy of each sensor node at the end of the next round to select the most appropriate sensor nodes that can continue as CH. In the proposed scheme, three-level energy heterogeneity of sensor nodes are used as in [3] and the entropy weight coefficient is used to come up with optimal selection of CHs.

Information entropy theory such as administration entropy, environment entropy, and economy entropy [51] have been employed in many discrete areas. Entropy indicates the valuable information produced by the data, as a result it can be used to determine the weights. The entropy value becomes smaller when the analyzed objects have fairly big difference between each others on a given specific criterion. However, the weight of the criterion should be smaller when the difference between objects is smaller (larger entropy value). In this paper, clustering algorithm can be adopted in terms of entropy as a election criterion. Moreover, we consider selection of CH node as a decision problem based on multiple criteria such as residual energy, sensor node density, and distance to the BS. Hence, we have multiple alternatives, i.e., set of sensor nodes where each alternative consists of multiple criteria, i.e., sensor node's information and selection decision need to be made according to these critaria. For this situation, Multi-Criteria Decision Analysis (MCDA) methods such as Weighted Product Model (WPM) [48] is utilized for addressing the decision problem and the entropy weight coefficient is used to assess the weight of different criteria [4].

A general Compressive Sensing scenario is presented as follows: a WSN has N sensor nodes that deployed to sense data in a region and send the collected data to CH or BS. Sparse signals are the main assertion to use CS. In CS, if the measurements number is M and the sparse level is S. BS requires $M \geq S \log N/S$ to reconstruct the original data x from the CS measurement y where $y = \Phi x$, $y \in R^{M\times 1}$ and Φ is $M \times N$, $M \ll N$, CS random matrix (Bernoulli, Gaussian, etc.). In cluster-based WSNs, two main scenarios of using CS method exist: plain CS and hybrid CS. In plain CS [7], [8], [33], [40], using CS matrix, each node compresses and sends samples vector of size M to its CH and then each CH compresses and sends M samples to BS. The main disadvantage of plain CS, each sensor node sends M samples in spite of its data size is $< M$, which leads to unnecessary higher traffic. Hybrid CS is proposed to overcome the drawback of plain CS [28], [53], where every CH determines either to use CS or not, according to the collected data size. It will use CS only if the data size $\geq M$. In this paper, we compress the data using hybrid CS after adapting the cluster size and propose a new algorithm to reconstruct the original data at BS with minimum Average Normalized Mean Squared Error. The main contributions of the current work can be summarized as follows:

1. We consider selection of CH node as a decision problem based on multiple criteria. Remaining energy, distance to BS and Intra-to-Inter Distance Ratio of neighboring sensor nodes are used as criteria for selecting CHs.
2. WPM is utilized for addressing the decision problem and the entropy weight coefficient is used to assess the weights of different criteria.
3. We introduce a new algorithm to predict the remaining energy of each sensor node at the end of the next round to select the most appropriate sensor nodes to continue as CHs in the next round.
4. In order to improve the reconstruction process, we propose a new efficient reconstruction algorithm based on Bees Algorithm to reconstruct the original sensors data from the compressed samples.
5. Extensive performance analysis of the proposed scheme and comparison with baseline approaches to demonstrate the effectiveness of the proposed scheme. Our simulation results reveal that the proposed scheme can manage power consumption better than existing algorithms and achieves the desired results for WSNs.

The remainder of this paper is structured as follows: The Information Entropy description is presented in Section II. Section III describes the proposed algorithm. In section IV, the performance analysis and results are presented. Evaluations indicate that the proposed approach efficiently solves the problem and exceeds other existing algorithms. The conclusion of the proposed work is given in Section V. Used notations through the paper are given in Table 1.

We assume that N static sensor nodes with a unique ID for each ID are randomly deployed in a region R of size $M \times M$. Each sensor node knows its communicating neighbors, including their identifications and coordinates, which can be gathered statically via *hello* message, or periodically if frequent changes occur in the topology. In this paper, we assume energy heterogeneity of sensor nodes, i.e., different energy values of sensor nodes.

II. INFORMATION ENTROPY

Entropy in information theory uses the discrete probability distribution to represent the amount of uncertainty. Let X be a discrete random variable with alphabet χ and probability mass function $p(x) = Pr\{X = x\}, x \in \chi$. The entropy $H(X)$ of a discrete random variable X is defined by [9] as follows:

$$H(X) = -\sum_{x \in \chi} p(x) log_2 p(x) \tag{1}$$

The minimum entropy is 0 and it occurs when one of the probabilities is 1 and the rest are 0s, while the entropy achieves the maximum value ($H_{max} = log_2(n)$) when all the probabilities have equal values of $\frac{1}{n}$, where n is the number of outcomes.

Entropy indicates the valuable information produced by the data, as a result it can be used to determine the weights. The entropy value becomes smaller when the analyzed objects have fairly big difference between each others on a given specific criterion. However, the weight of the criterion should be smaller when the difference between objects is smaller (larger entropy value). Consequently, the entropy coefficient method is a target empowering method to determine the weight by calculating the entropy weights of each criterion based on determining variation degree with respect to every criterion value [23], [37], [46], [51].

Using the local information of the sensor nodes such as residual energy, a clustering algorithm can be adopted in terms of entropy as the election criterion. Such a new algorithm is energy efficient. Moreover, we consider selection of

TABLE 1. Table of notations.

Notation	Description
EWC	Entropy Weight Coefficient Method.
e_{CH}	Total energy consumption by CH.
$E_{consumed}(s,r)$	Total energy consumption for node s at round r.
E_0	Initial energy for normal node.
E_{int}	Initial energy for intermediate node.
E_{adv}	Initial energy for advanced node.
E_{Rx}	Energy consumption in data reception.
E_{Tx}	Energy consumption in data transmission.
ϵ_{mp}	Amplification energy to overcome the multi-path.
ϵ_{fs}	Amplification energy to overcome the free space.
E_{elec}	Electronics energy.
D	Distance to BS.
E	Residual energy.
DN	Sum of distances to neighbor nodes.
x	Original Signal.
M	Number of Measurements.
ψ	Transform Matrix.
ϕ	CS Matrix.
K	Sparsity Level.
y	Compressed Samples.
F_i	CH_i's CS Sub-Matrix.
H	Support set.
t	Matrix Transpose.
x'	Estimated Signal.
q	Number of Selected columns.
Q	Set of Indices of CS matrix's Columns.
F_Q	Sub-matrix of columns indices Q From F Matrix.
n_{scouts}	Number of Scouts Bees.
B	Bees Solution Matrix.
e	Size of Elite Site.
T_{max}	Maximum Number of Iterations.
F	Fitness Value.

CH node as a decision problem based on multiple criteria such as residual energy, sensor node density, and distance from the BS. Hence, we have multiple alternatives, i.e., set of, sensor nodes where each alternative consists of multiple criteria, i.e., sensor node's information and we need to make selection decision. For that, one of Multi-Criteria Decision Analysis (MCDA) methods is used to solve the decision problem with multiple criteria. In our proposed strategy, WPM [48] is utilized for addressing the decision problem and the entropy weight coefficient [4], [23], [46], [51] is used to assess the weight of different criteria. Our simulation results reveal that the proposed strategy can manage power consumption better than existing algorithms and achieves the desired results for WSNs.

III. CLUSTER-TREE ROUTING SCHEME FOR DATA AGGREGATION

In this section, we describe Cluster-Tree routing scheme (CTRS-DG) for data gathering. CTRS-DG has two layers: aggregation and reconstruction and routing. The aggregation and reconstruction layer includes three phases: (1) Head Election and Cluster Formation phase: In this phase, we propose a new dynamic and self-organizing Entropy Based Clustering algorithm to select CHs and form clusters; (2) Aggregation phase: where, data will be aggregated and compressed at each node based on compressive sensing technique; and (3) Reconstruction phase: we propose a new efficient BEE based Reconstruction algorithm (BEBR) to reconstruct the original data from the compressed data. The phases of routing layer are as follows: (1) Setup phase: In this phase, we propose a new algorithm to form the routing tree as a backbone for the network; (2) Data Transmission phase: where, every CH selects its parent based on rank value which is computed based on the distance from BS and the remaining energy of CH. The compressed data at CHs will be routed to BS using routing tree. Figure 1 show block diagram of the proposed scheme. First we discuss our proposed Entropy Based Clustering Algorithm (EBCA). EBCA is a dynamic distributed and self-organizing clustering algorithm that prolongs the lifetime of the network and minimizes the consumed energy in the network. Then, the description of the phases of the proposed scheme is given.

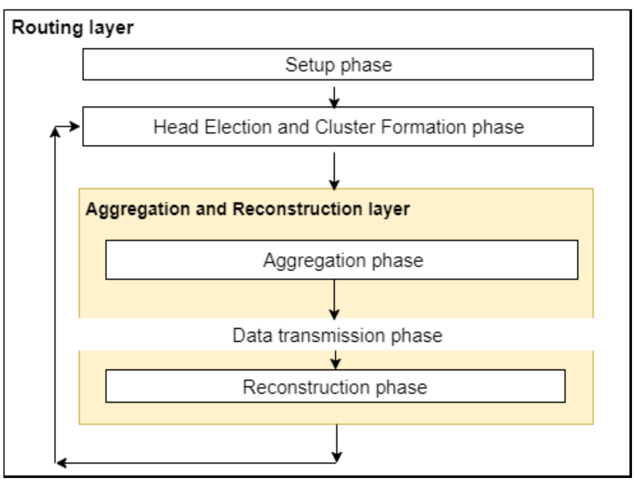

FIGURE 1. **Block diagram of the proposed scheme.**

A. EBCA: ENTROPY BASED CLUSTERING ALGORITHM

In our proposed scheme, WPM along with Entropy Weighted Coefficient Method (EWC) is used for resolving the election decision problem based on the following criteria for node s_j.

- **Residual energy** ($E(s_j)$)**:** $E(s_j)$ is the most important feature of every sensor node where the network lifetime mainly depends on the residual energy among the sensor nodes.
- **Distance to BS** ($D(s_j)$)**:** $D(s_j)$ is important to be considered where as more distance to BS implies more energy consumed to transmit packets to BS.
- **Intra-to-Inter Distance Ratio** ($DR(s_j)$)**:** The objective of $DR(s_j)$ is to minimize intra-cluster distance between CHs and respective CMs; and maximize the minimum inter-cluster distance between two distinct CHs. As cluster's intra-distance and inter-distance are judging the soundness of the clustering methods provided by the routing protocol algorithm [2]. To achieve this objective, the criteria function is defined as ratio of total Euclidean distance of CHs to their CMs and minimum Euclidean distance between any pair of CHs. Intra-distance to inter-distance ratio is given by

$$DR(s_i) = \frac{\sum_{s_j \in N(s_i)} d(s_i, s_j)}{\min_{s_j \in N(s_i)}\{d(s_i, s_j)\}}. \tag{2}$$

Here, $N(s_i)$ is the set of neighbor nodes of s_i.

In next section, we adapt WPM which is a Multi-criteria decision analysis method (MCDA) [48] to address the election of CHs problem.

1) WEIGHTED PRODUCT MODEL (WPM)

In WPM, each alternative $(A_1, A_2, \ldots)$ is compared with others by multiplying a number of ratios for each criterion (c_i). Each ratio is raised to the power equivalent of the relative weight (w) of the corresponding criterion. In order to compare two alternatives A_K and A_L using WPM; the following product (P) has to be calculated for m number of criteria and n number of alternatives:

$$P(\frac{A_K}{A_L}) = \prod_{j=1}^{m}(\frac{a_{Kj}}{a_{Lj}})^{w_j}. \tag{3}$$

Here, $K \neq L, K, L = 1, 2, \ldots, n$, a_{ij} is the actual value of the alternative A_i in terms of j^{th} criterion and w_j is the weight of importance of the j^{th} criterion. If the ratio $P(\frac{A_K}{A_L}) \geq 1$, then it implies that the alternative A_K is more useful than the alternative A_L. The best alternative is the one that is better than or at least equal to all others.

The alternative approach of WPM is the decision maker where we use only products without ratios as follows [48]:

$$P(A_K) = \prod_{j=1}^{m} (a_{Kj})^{w_j}. \tag{4}$$

In this formula, the term $P(A_K)$ indicates the performance value of alternative A_K when all the criteria are evaluated under the WPM model. The weights w_j of different criteria are calculated using entropy coefficient method that will be described in the next subsection.

2) ENTROPY WEIGHT COEFFICIENT METHOD (EWC)

The entropy coefficient method is applied to determine the weights for the criteria. The main steps of calculating the weights of m criteria and n alternatives are as follows:

- Calculate the entropy of each criterion $i = 1, \ldots, m$ that can be derived from Equation 1.

$$H_i = -\frac{1}{log_2 n} \sum_{j}^{n} p_{ij} log_2 p_{ij} \tag{5}$$

  Here, $p_{ij} = \frac{c_i(s_j)}{\sum_j^m c_i(s_j)}$, $j = 1, \ldots, n$, and $c_i(s_j)$ is the performance value of alternative s_j. If $p_{ij} = 0$, $p_{ij} log_2\, p_{ij}$ will be 0.
- Calculate the entropy coefficient weight (w_i) of each criterion i:

$$w_i = \frac{(1 - H_i)}{m - \sum_i^m H_i}. \tag{6}$$

  Here, $0 \leq w_i \leq 1$, $\sum_{i=1}^m w_i = 1$.

3) CLUSTER HEAD ELECTION DECISION

The election decision of CH is made by executing the following steps (The pseudo-code of this procedure is presented in Algorithm 1).

- Using EWC, compute the weight of each criterion.
- Using WPM, compute the product value P of each sensor node s_j considering $E(s_j)$, $D(s_j)$ and $DR(s_j)$ criteria as:

$$s_j.P = (s_j.c_1)^{w_1} \times (s_j.c_2)^{w_2} \times (s_j.c_3)^{w_3}. \tag{7}$$

  Here, w_1, w_2, w_3 are the weights for the criteria $c_1 = (\frac{E(s_j)}{sj.Eo})$, $c_2 = 1 - \frac{D(s_j)}{\sum_{i=1, i \neq j}(D(s_i))}$, $c_3 = 1 - \frac{DR(s_j)}{\sum_{i=1, i \neq j}(DR(s_i))}$ respectively.
- Sensor nodes with highest P values will be elected as CHs.

Algorithm 1 Election Procedure (EP)

Input: S: Set of sensor nodes and
k: number of CHs.
Output: k sensor nodes with highest P values.
$S(j).c_i$: criterion value of sensor node j for criteria i
m: number of criteria
n: number of alternatives (sensor nodes in S)
for $i = 1$ to m **do**
  for $j = 1$ to n **do**
    $p_{ij} = \frac{S(j).c_i}{\sum_{u=1}^n S(u).c_i}$
  end for
  $H_i = -\frac{1}{log_2\, n} \sum_j^n p_{ij} log_2\, p_{ij}$
end for
for $i = 1$ to m **do**
  $w_i = \frac{(1-H_i)}{m - \sum_i^m H_i}$
end for
for $j = 1$ to n **do**
  $S(j).P = \prod_{i=1}^m (S(j).c_i)^{w_i}$
end for
Return the top k nodes of S with highest P values.

B. SETUP PHASE

- **Backbone tree construction Step:** In this phase, BS constructs the backbone tree. The backbone tree is described as follows:
  1. Initially each node i sets its parent to *null* and level to a large number.
  2. BS (s_r) sets its level to zero then broadcasts BUILD message to its neighbors. The BUILD message contains node ID, energy level, and level.
  3. Each node s that receives the BUILD message will do the following:
     (a) If the received level value is less than its level value ($s.level$), s adds the owner of BUILD message to the set of candidate parents C and set $s.level$ to BUILD.level + 1
     (b) If the received level value is greater than or equal to its level value ($s.level$), the received message will be dropped.
     (c) Finally s broadcasts only once a BUILD message with its own level, energy level and ID to its neighbors.
  4. s calculates the rank value of each node in C based on the distance to s_r and the remaining energy using the following equation:

$$rank(s) = \frac{E_{remaining}}{d(s, s_r) * E_{initial}} \tag{8}$$

     This guarantees that the node with higher residual energy and minimum distance to the BS will have higher rank. If multiple parents have the same rank, the node with the higher residual energy is selected as a forwarding node. Parent receives the data from other nodes and forwards it directly to the next hop without aggregation.
  5. Node s sets its parent to the node with the highest rank in C.

C. HEAD ELECTION AND CLUSTER FORMATION PHASE

1. **Initial Step:** BS participates in CHs' selection only in the first round ($r = 1$). The election process starts by executing the selection procedure (Algorithm 1) considering the criteria, E, D and DR, i.e., the BS calls

$EP(S_i, k_i)$ procedure where S_i is the set of all nodes at level i in WSN. BS sends BE_{CH} message to k nodes with the highest P values at each level.

As previously discussed, CS method uses random matrix such as Gaussian matrix as a measurement matrix. The measurement matrix is used in the compression and reconstruction process. The random matrix is generated using seeds. In our approach, BS generates and broadcasts a global seed ξ to the entire network. The generated seed is used by each CH to compress its cluster data and by BS to reconstruct the original data. Upon receiving ξ by each CH j (CH_j), using ξ and the unique identification of CMs (node IDs), CH_j generates the corresponding series of coefficients of its CMs. These coefficients will be regenerated at the BS for reconstruction process where BS has the identifications of all nodes and ξ.

After receiving BE_{CH} message, the elected CHs announce their role by broadcasting advertise message CH_{ADV} and then each CH forms its cluster by executing cluster formation phase below.

For the subsequent round operations, each elected CH decides to stay or retract its role according to the results of executing Cluster Head Election Step.

2. **Cluster Head Election Step:** The consumed energy by CH involves the consumed energy by receiving data from its CMs, aggregating data, and forwarding data to the BS. As a result, we employ energy prediction technique to predict the CH failure due to energy depletion. If we assume that N sensor nodes are uniformly dispersed among k clusters in the topology. Thus, on average, there are N/k sensor nodes per cluster (one CH and (N/k)-1 CMs). Consequently, the total energy consumption by the CH (e_{CH}) for a single round is calculated as follows [17]:

$$e_{CH} = (\frac{N}{k} - 1).E_{Rx}(l) + \frac{N}{k}.l.E_{DA} + E_{Tx}(l, d_{toBS}). \tag{9}$$

CMs only need to transmit data to its corresponding CH. Thus, the total energy consumption by a CM (e_{CM}) during a single frame is calculated as follows:

$$e_{CM} = E_{Tx}(l, d_{toCH}). \tag{10}$$

Here, d_{toBS} is the mean distance between CH and BS, and d_{toCH} is the average distance between CMs and their CH. In an area with size $M \times M$, d_{toBS} and d_{toCH} is given by [36]:

$$d_{toCH} = \frac{M}{\sqrt{2\pi k}}, \quad d_{toBS} = 0.765\frac{M}{2}. \tag{11}$$

The total energy consumption for a cluster can be calculated as follows:

$$E_{consumed} = e_{CH} + e_{CM}. \tag{12}$$

According to the current energy of a node s and Equation 12, the energy consumption ratio (E_{ratio}) is calculated as follows:

$$E_{ratio}(s) = E_{consumed}/E_{residual}(s). \tag{13}$$

Depending on the ratio E_{ratio}, each node determines whether it has the ability to act as a CH or not. Equation 13 shows that the more the total remaining energy, the smaller the ratio E_{ratio} is. Each CH calculates its E_{ratio} and the ratio for each CM. The nodes with ratio less than energy threshold value are added to candidate set ψ. Then, CH executes Algorithm 1 based on the set ψ, then in order to take a decision for the next round. CH has three cases (Algorithm 2, steps 26-38):

(a) If the list of candidate CHs (ψ) is empty, i.e., none of CMs or current CHs can be a CH for the next round because E_{ratio} is smaller than the predefined threshold energy value ($E_{threshold}$). In this case, the current CHs inform their CMs to send their data directly to the BS. By this way, we avoid unreliable and un-predicted behavior of the network by avoiding forming clusters and using the remaining energy of the remaining sensor nodes, they may succeed to send its data to the BS.

(b) If ψ contains only one CH candidate, this CH remains working as a CH for the next round.

(c) If ψ contains more than one CH candidates ($|\psi| > 1$), as a result CH has to execute the election procedure EP with the set ψ ($EP(\psi, 1)$) to take a decision.

If the cardinality of ψ is larger than one, then CH has a set of alternatives. The first alternative is to stay working as CH, and the second alternative is to select one of its CMs to work as a CH. CH executes the election procedure EP considering ψ members and the three criteria, E, D and DR. The product value of each member in ψ is computed and compared, then the sensor node with the highest product value P will be selected as the next CH (Algorithm 2, Step 34).

Each CH sends BE_{CH} message to the elected CHs. The elected CHs announce their role using CSMA. The advertising short message CH_{ADV} contains the ID of CH. Then, cluster formation step is executed.

3. **Cluster Formation Step:** Each CH_i broadcasts advertisement message CH_{ADV} which includes the identifier and the location of the CH to neighbors. To ensure that the concise information will be sent to the BS, each non-CH node selects the CH with minimum communication cost. After the CH announcement, according to the distance between each non-CH and the selected CH, each non-CH determines to which cluster it should join by sending $JOIN_{Request}$ message to the selected CH. The collected data by CMs will use the reverse path of the forwarding advertisement message.

Algorithm 2 EBCA

$G_{CM}(ch)$: List of CMs of ch.
ψ: List of sensor nodes to be nominated as CHs.
{**CH Side.**}
if BE_{CH} message received. **then**
  Advertise your role as CH by transmitting CH_{ADV} message.
  Wait for $JOIN_{Request}$ messages from neighbor sensor nodes.
end if
if $JOIN_{Request}$ message received from s_j **then**
  Add sensor node s_j to $G_{CM}(s_i)$.
end if
Transmit TDMA schedule to CMs.
if Data received from $G_{CM}(s_i)$ members **then**
  Perform data compression using CS and send the result tO BS.
end if
Calculate E_{ratio} for cluster i.
if $E_{ratio} < E_{threshold}$. **then**
  Add ch to ψ.
end if
for each sensor node $cm_i \in G_{CM}(s_i)$. **do**
  if $E_{ratio}(cm_i) < E_{threshold}$. **then**
    Add cm_i to ψ
  end if
end for
if $|\psi| = 0$ **then**
  CH informs $G_{CM}(s_i)$ members to send directly to BS.
  Relinquish your role as CH and send directly to BS.
else
  if $|\psi| = 1$ and $s_i \in \psi$ **then**
    s_i decides to remain working as CH.
  else
    if $|\psi| > 1$ **then**
      s_i executes the election procedure by calling $EP(\psi, 1)$
      s_i transmits BE_{CH} message to the next CH node.
    end if
  end if
end if
{**non-CH Side**}
if CH_{ADV} message received from s_j. **then**
  send $JOIN_{Request}$ to s_j that requires minimum communication cost.
  wait for data request message.
end if
if data request message received. **then**
  Transmit your data to the CH along with (ID, $E_{residual}$.)
end if

D. AGGREGATION PHASE

Normally, each CM sends only its collected data to the corresponding CH in its allocated time slot. However, in our proposed algorithm, the data message of CM includes local information such as ID, residual energy. This local information can be employed to localize the CH rotation in the subsequent rounds.

The task of CMs is to collect the original data and transfer it to CHs. The CHs collect local information, reduce the correlation between CMs, and eliminate the redundancy to increase the transmission rate. BS gets the gathered-data, reconstructs the compressed data and makes decisions. In this phase, we adopt hybrid CS method where the node sends its data without CS to its CH which determines either to use CS or not, according to the collected data size.

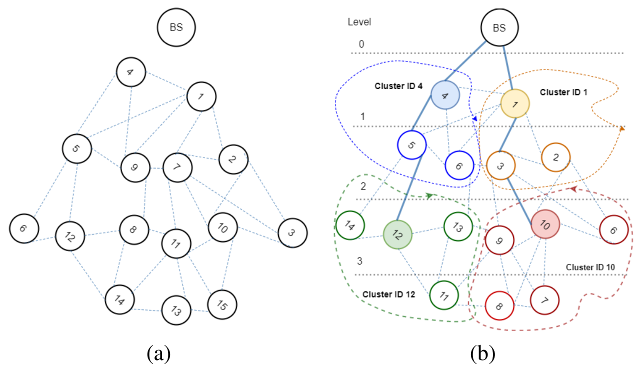

FIGURE 2. (a) A WSN network graph, (b) Cluster tree based example.

For clarification, In Figure 2(b) nodes 6, 7, 8, and 9 as CMs transmit their data x_6, x_7, x_8, and x_9 to their CH CH_{10}. Based on the global seed ξ and the IDs of its CMs, CH_{10} generates the corresponding series of coefficients $\Phi_i = \begin{pmatrix} \Phi_{1,i} \\ \vdots \\ \Phi_{M,i} \end{pmatrix}$, $i = 6, 7, 8, 9, 10$

for node 6, 7, 8, 9 and 10 that will be used to compute the measurement. CH uses Φ_i, $i = 6, 7, 8, 9, 10$ to decrease the amount of sent data. CH gathers its information, and add them with CMs information by computing $\sum_{i=6,7,8,9,10} \Phi_i x_i$. The results then routed to BS using *Data transmission Phase*

At the end of this phase, using piggybacked CM information, each current CH decides whether it would continue as CH or give up its role (as explained in *Head Election and Cluster Formation Phase*).

E. DATA TRANSMISSION PHASE

After each CH_i generates its compressed data, it calculates the rank value of each node in the parent set based on the distance to *BS* and the remaining energy using equation 8. Then, it transfers the data to the parent node with higher rank. The parent node receives the data and repeats the same process to transfer the data to BS. The same operations are performed by every CH. Finally, BS collects the compressed data and get the original data by applying the *Reconstruction Phase*.

F. RECONSTRUCTION PHASE

CS method utilizes the sparsity property in the full IoT sensors signals, due to the high correlations between IoT sensors' data, to reduce the signal size from N to M such that $M \ll N$. So one of the biggest CS method challenges is to reconstruct the full signal (N) from the sub-sampled vector (M). The reconstructiion algorithms can be divide into two categories [15]: 1) convex relaxation algorithms and 2) Greedy algorithms (GA). The convex relaxation reconstruction based algorithms depend on relaxing the problem of the following equation:

$$x = arg\ min\|x\|_0 \quad \text{subject to} \quad y = \Phi x \tag{14}$$

into convex problem by replacing non-convex L_0 by the convex L_1 norm so Equation 14 can be rewritten as follows:

$$x = arg\ min\|x\|_1 \quad \text{subject to} \quad y = \Phi x \tag{15}$$

Then, Equation15 can be solved using any convex problem solvers such as $L1$-magic toolbox [13]. Although the convex reconstruction based algorithms have the stability and the ability to reconstruct the full signal correctly, but they suffer from highly complex computations that make them not suitable for IoT.

On the other hand, Greedy reconstruction based algorithms provide the same reconstructions performance with low computation that make them suitable for IoT. Greedy reconstruction based algorithms can be divided into two types: Reversible and Irreversible GA. Detection of the support-set using matched filter detection and then, the estimation of the original signal based on the detected support-set by solving the least square problem are common steps for the two types. However, during irreversible greedy, like OMP [47] algorithm, once it adds one element to the support set (forward step), this element remains in the sets till the end of the search. In contrast, reversible greedy algorithms, like COSAMP [32] and SP [52] algorithm, gives itself the ability to remove any elements (backward step) that have been added to the support set during the search.

The greedy reconstruction based algorithms have gained significant attention for CS signal recovery. However, the greedy reconstruction based algorithms often doesn't provide optimal solutions to CS reconstruction problem [15]. Bees algorithm [34] is a famous meta-heuristic optimization algorithm, and it proved its efficiency in finding global solutions for a lot of problems such s combinatorial optimization and selection problems. The basic idea of Bees search inspired by the natural behavior of honey bees to find the nest site between many sites by considering both speed and accuracy (optimal solution). This analogues to finding the optimal solution in an optimization process. The Bees algorithm can be described in Figure 3.

First, Bees are initialized by sending number n of bees (scouts bees) to the selected food sites randomly. The fitness value of each site will be calculated and then all the values are sorted from the biggest to the smallest. Bees selects the best m sites which have the highest fitness values to start neighborhood search. The Bees divide the best sites according to their fitness values into elite sites (e) and non-elite sites ($m - e$).

The local search starts by searching around the best sites such that Bees assign more number of bees to search in the elite sites and send $m - e$ bees to search in the non-elite sites. The bee population will be updated by the bee which has the highest fitness value. Then Bees perform a random search on other ($n - m$) sites which called non-best sites. Finally, Bees sort all the new sites location decently according to their fitness values and repeat the previous processes till the global optimum is found or the stopping criteria are reached.

In this section we aim to integrate between the advantages of greedy algorithm (fast and easy implementation) and the advantages of Bees to find the optimal solution to the reconstruction process. The new algorithm called BEE Based Signal Reconstruction algorithm (BEBR). BEBR consists of three steps: initialization, search and stop criteria. BEBR starts like any greedy based algorithm by selecting the K largest amplitude components of $y\Phi^t$, where t means the transpose of the matrix Φ, to initialize the support set H. Then BEBR executes Bees search to select q columns from matrix Φ such that the set $Q = H \cup q$ has the minimum fitness value. The value of q is assigned as $q = \lfloor 0.7\ M \rfloor - 7K/9$ as used in [15]. The BEBR uses the set Q to estimate the signal x' by solving the least square problem. The support set H is updated by K largest amplitude components of x'. Finally, BEBR checks the stopping criteria to decides either to stop and return x' if the number of iterations exceeds the maximum number or $y - \Phi x' \leq 0$, or repeating the search step with the new value of the support set H. Before describing BEBR algorithm, some operations (initialization operations) that are used in the algorithm are defined as follows:

- **Solution orthogonality removal**

$$resid(y, \Phi_Q) \triangleq y - y_p, \quad \text{where} \quad y_p = \Phi_Q^t \Phi_Q^{\dagger} y \tag{16}$$

- **Correlations search**

$$supp(x, k) \triangleq \Big(\text{the set of indices corresponding to k} \\ \text{largest amplitude components of x}\Big) \tag{17}$$

1) FITNESS FUNCTION

We propose the following fitness function which is used by BEBR for the selection process: According to [14], if the sparsity Level K satisfying $K < Spark(\Phi)/2$, where $Spark(\Phi)$ represents number of linear independent columns of matrix Φ, then, CS reconstruction problem can be approximated by:

$$argmin_x \|\Phi x - y\|_2, \quad \text{such that } \|x\|_0 \leq K \tag{18}$$

From the greedy reconstruction based algorithms, the signal x can be estimated by solving the following least square problem:

$$x' = \Phi_Q^{\dagger} y, \quad \text{such that } x'_{N-Q} = 0\ (N = 1, 2, \ldots .n) \tag{19}$$

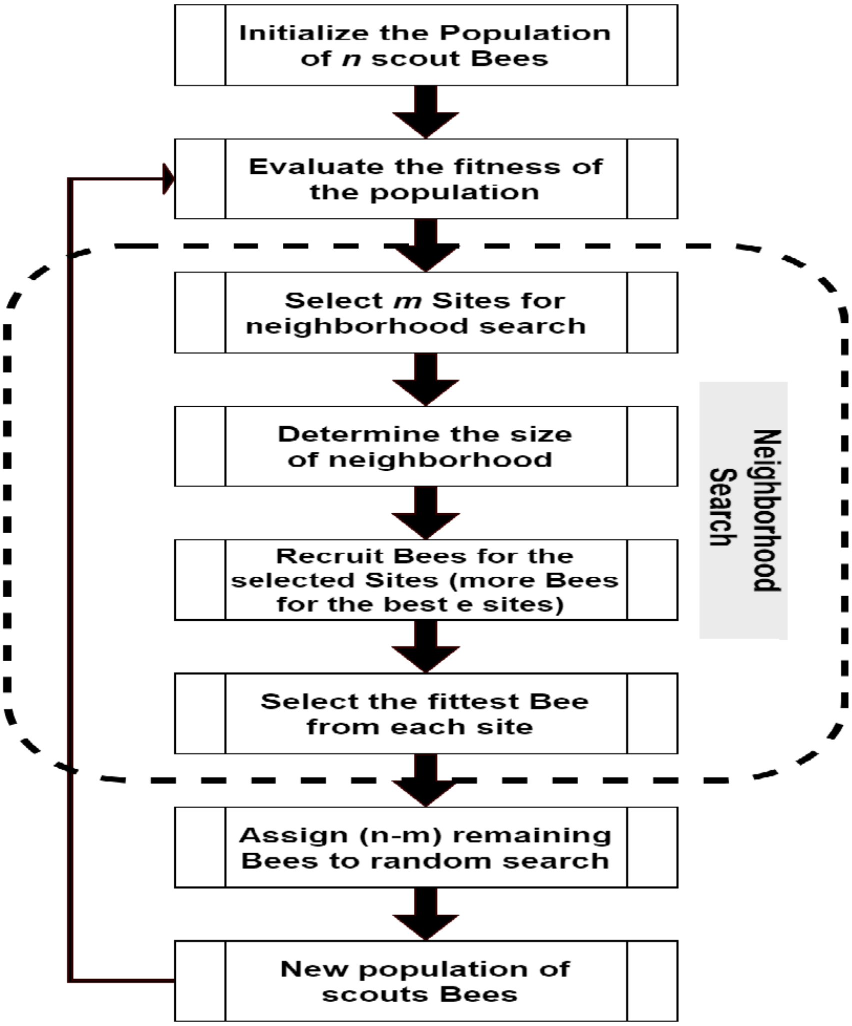

FIGURE 3. Flow Chart of Basic Bees Algorithm.

Here, Φ_Q is a sub-matrix from matrix Φ and $\Phi^\dagger$ is the pseudo inverse of matrix Φ. We can only obtain the exact signal x, i.e., $x' \approx x$ if and only if $\Phi_Q \Phi_Q^\dagger y = y$. Then we can obtain $\|\Phi_Q \Phi_Q^\dagger y - y\|_2 = 0$, therefore, the fitness function can be defined as:

$$F(Q) = \|\Phi_Q \Phi_Q^\dagger y - y\|_2 \tag{20}$$

2) BEBR ALGORITHM DESCRIPTION

Here, we describe BEBR algorithm to solve the problem of CS reconstruction (see Algorithm 3).

During the initialization step all Bees parameters such as number of scouts bees n_{scouts}, size of elite site e, size of non-elite sites $(m - e)$ and population size n are initialized. Then, BEBR creates the bee solution matrix B, where B is $q \times n_{scouts}$, as random matrix of integers in the range $[1, N]$ such that N is the number of matrix Φ's columns.

The set Q is initialized by adding the indices of the largest K amplitude components of $\Phi^t y$. The fitness value will be calculated for each row i of B. Finally, the solutions are sorted in descending order and then set $BestSol$ = fittest bee solution.

During the search step, BEBR executes Bees to find the best bee solution ($BestSol = q$ columns) that nears the estimated signal to the original one. After that, the set Q is updated as the union between H and the new $BestSol$. Based on this new Q, the estimated signal x' is computed.

Finally, the support H is updated by selecting the indices of the K largest amplitude components of x'. The last two steps are repeated until the Stopping Criterion is met, i.e., if the number of iterations $\geq T_{max}$ or the value of E is smaller than or equal to $\theta = 10^{-5}$.

IV. SIMULATION RESULTS

Our algorithm is simulated using MATLAB R2015a. 100 sensor nodes are randomly deployed in a two-dimensional plane region with size 100×100 m^2 with BS is placed at the center. The radio model and the energy parameters are used as

Algorithm 3 BEBR Algorithm

Initialization Step:
Input: sparsity level= K, CS matrix= Φ, q, measurement vector $=y$, and Maximum number of iterations $=T_{max}$.
initialize Bees supp.
for Each row i of matrix B **do**
  Q=Union (H,i)
  F= fitness value of each row i Φ_Q using Equation 20.
end for
Sort the set F in descending order.
$BestSol$ = The position of the fittest bee.
T=1
Search Step:
while Stopping criterion is not met **do**
  for Each Elite site (m) and non-Elite site (m-e) patches **do**
    Recruit the forager bees to find new solutions.
    Q=Union (H, new solution)
    Evaluate the solution of each patch.
  end for
  Assign the remaining bees to search randomly and evaluate their fitness values.
  BestSol= The position of the fittest bee from each patch and updates the matrix B.
  Q=Union(H,BestSol)
  $x' = \Phi_Q^{\dagger} y$ and $x'_{N-q} = 0$
  H= $supp(x', K)$ Equation 17.
  Stopping Criteria Step:
  E= $resid(y, \Phi_H)$ Equation 16.
  if $E \leq \theta || T \geq T_{max}$ **then**
    Stop;
  end if
  $T = T + 1$
end while
Output: $x' = \Phi_H^{\dagger} y$ and $x'_{N-H} = 0$

in many previous works such as [3], [17], and [36]. The main parameters and their values of the simulation are provided in Table 2. The used performance metrics are as follows:

1. First node dies (FND): the time from the beginning of the experiment until the first sensor node dies.
2. Last node die (LND): the time from the beginning of the experiment where all sensor nodes are on until the last sensor node dies.
3. Number of alive sensor nodes per round: The number of alive sensor nodes in the WSN after each round.
4. Average remaining energy per round: the total remaining energy of all sensor nodes divided by the number of nodes.

The scheme has been tested using different random topologies. We assume that the total energy of the network is $102J$.

In the next subsections we evaluate our proposed scheme based on performance metric and compare the performance results with a number of existing baseline algorithms.

TABLE 2. Simulated parameters.

Parameter	Value
Network area size	100×100,
Nodes	100
Initial energy	$0.5\ J, 1.25\ J, 2.0\ J$
E_{elec}	50 nJ/bit
ϵ_{fs}	10 $pJ/bit/m^2$
$\epsilon_{m}p$	0.00013 $pJ/bit/m^4$
d_0	87 m
E_{DA}	5 nJ/bit/signal
Packet size	4000 bits

A. LIFETIME AND RESIDUAL ENERGY EVALUATION

The evaluation and comparison with the baseline algorithms will be according to the following two cases:

1. **Case 1:** Perfect Data Aggregation: where all individual signals can be combined into a single representative signal [17]. In this case we compare the performance results of the proposed scheme CTRS-DG with SILEACH [1], CREEP [45], and SEECP [31] algorithms that consider perfect data aggregation.
2. **Case 2:** Routing based CS Aggregation: in this case we compare the performance results of the proposed scheme CTRS-DG with SEP-CS [26] and LEACH-CS [29] algorithms.

1) CASE 1: PERFECT DATA AGGREGATION

The stability period is defined as the duration from the beginning of the experiment until the first sensor node dies. In the first test, we evaluate the proposed scheme based on first node dies, last node dies, number of alive nodes per round and the average residual energy per round and compare the results with the baseline algorithms CREEP, SEECP and SILEACH.

Figures 4 and 5 show that the proposed scheme CTRS-DG enhances the stability period compared with CREEP, SEECP and SILEACH up to 27%, 37% and 49% more, respectively.

Figure 6 shows the number of alive nodes in the network per round in CTRS-DG, CREEP, SEECP and SILEACH. It is clear that the number of alive nodes in the proposed CTRS-DG are more than those of CREEP, SEECP and SILEACH algorithms.

Figure 7 shows that the average of residual energy of sensor nodes decreases as the number of rounds increases. It is clear that the consumed energy by CTRS-DG is less than the consumed energy by other algorithms. I.e., the proposed scheme CTRS-DG enhances the LND and FND of WSNs better than CREEP, SEECP and SILEACH because the weights are calculated and updated based on the current status of sensor nodes in each round. Moreover, CTRS-DG considers cluster load as a parameter during CH election process and it predicts if the current CH can continue its task as a CH until the end of the next round or it must be replaced by one of

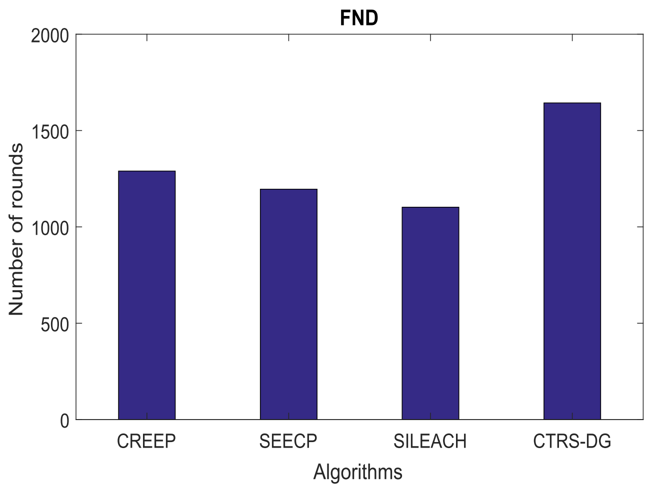

FIGURE 4. Stability (First node dies) in CTRS-DG, SILEACH, CREEP, and SEECP algorithms.

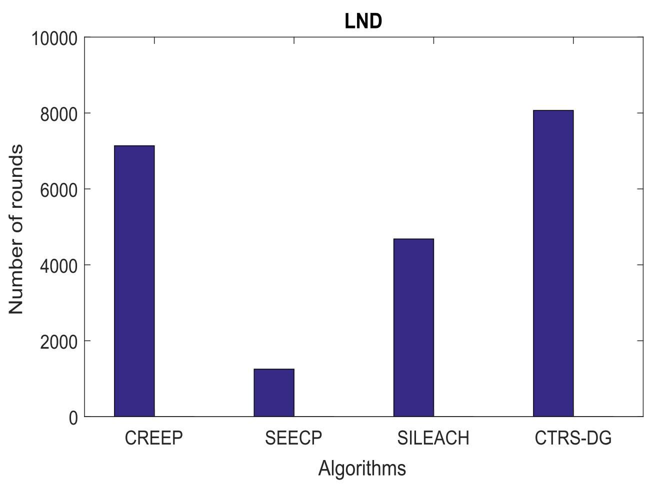

FIGURE 5. Last node dies in CTRS-DG, SILEACH, CREEP, and SEECP algorithms.

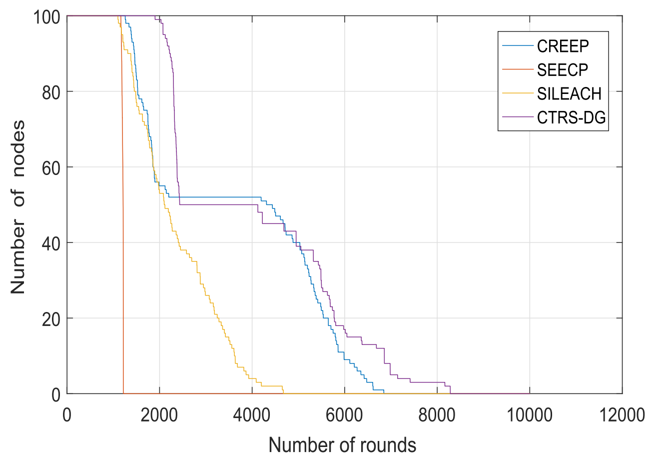

FIGURE 6. Number of alive sensor nodes per round in CTRS-DG, SILEACH, CREEP and SEECP algorithms.

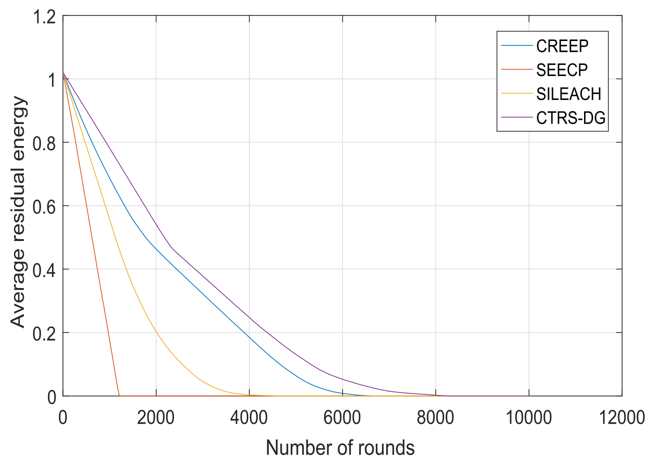

FIGURE 7. Average residual energy per round in CTRS-DG, SILEACH, CREEP, and SEECP algorithms.

its CMs which leads to balance the energy level of the sensor nodes in the network and prolong the lifetime of sensor nodes with low energy level by working as CMs only. The results show that the worst network lifetime is in SEECP because SEECP considers only energy level as the only parameter for CH selection process.

2) CASE 2: ROUTING BASED CS AGGREGATION

In the second test, we evaluate the proposed scheme based on first node dies, last node dies, number of alive nodes per round and the average residual energy per round and compare the results with the baseline algorithms LEACH-CS, and SEP-CS.

Figure 8 shows the lifetime and the percentage of alive nodes in the network per round in CTRS-DG, LEACH-CS, and SEP-CS algorithms.

It is clear from Figures 9 and 10 that the first and the last nodes in CTRS-DG live more than those of LEACH-CS and SEP-CS algorithms, i.e., CTRS-DG reduces the energy consumption over all sensor nodes because the proposed scheme handles the CHs selection as a decision problem using a number of criteria which leads to the best selection of CHs and so save energy consumption.

Figure 11 shows that the proposed scheme has the maximum residual energy after each round compared to other algorithms, i.e., in each round, the energy consumed by CTRS-DG is less than the energy consumed by LEACH-CS and SEP-CS algorithms because CTRS-DG selects the most suitable CHs based on the updated weights which considers the network status at each round. Moreover, Each neighbor node joins the CH that in its transmission range this leads to reduce the energy consumption for transmitting data where each node is in the same transmission range of other cluster nodes.

B. EFFECT OF ENERGY HETEROGENEITY

In this test, the goal is to evaluate the performances of CTRS-DG, SILEACH, CREEP, and SEECP in the two scenarios

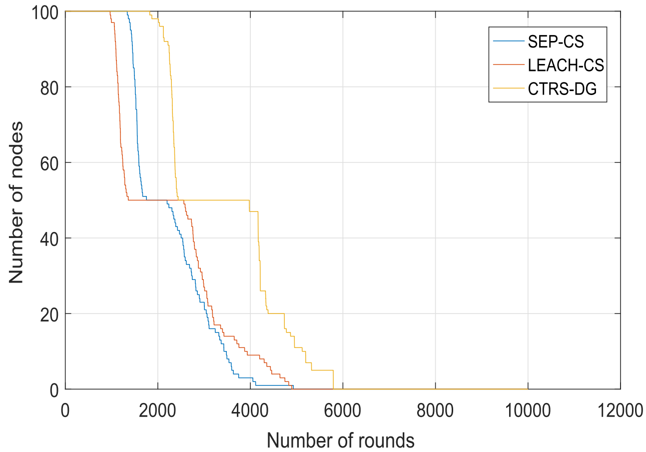

FIGURE 8. Number of alive sensor nodes per round in CTRS-DG with CS, LEACH-CS and SEP-CS.

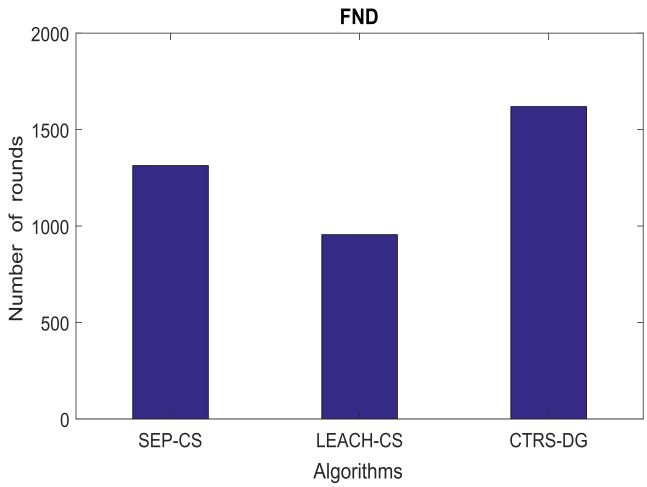

FIGURE 9. Stability (first node dies) in CTRS-DG with CS, LEACH-CS, and SEP-CS.

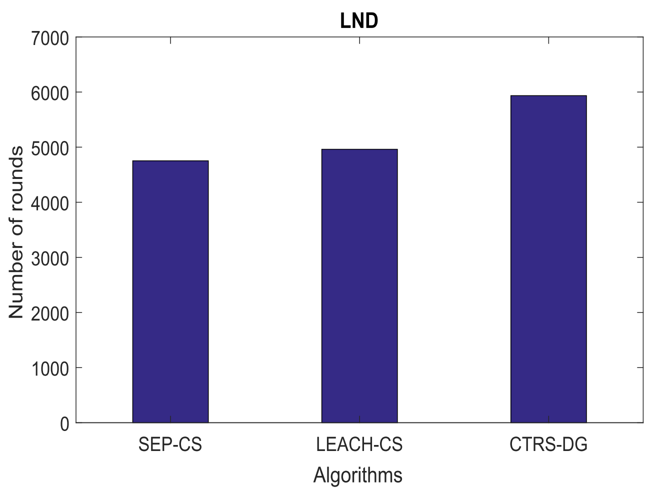

FIGURE 10. Last node dies in CTRS-DG with CS, LEACH-CS, and SEP-CS.

of energy heterogeneity: predetermine and random. In predetermine energy heterogeneity, three different initial energy values of nodes are considered. Table 3 shows the ratio of

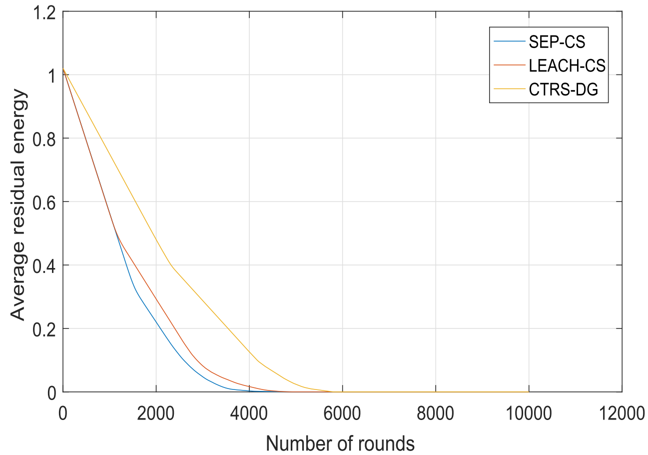

FIGURE 11. Average residual energy per round in CTRS-DG with CS, LEACH-CS and SEP-CS.

TABLE 3. Energy setting of sensor nodes.

% of nodes	Energy
20% of the nodes	2J
30% of the nodes	1.25J
50% of the nodes	0.5 J

sensor nodes and their corresponding energies. In random energy heterogeneity, initial energy of nodes are assigned to a random value with total 102 unit to the network.

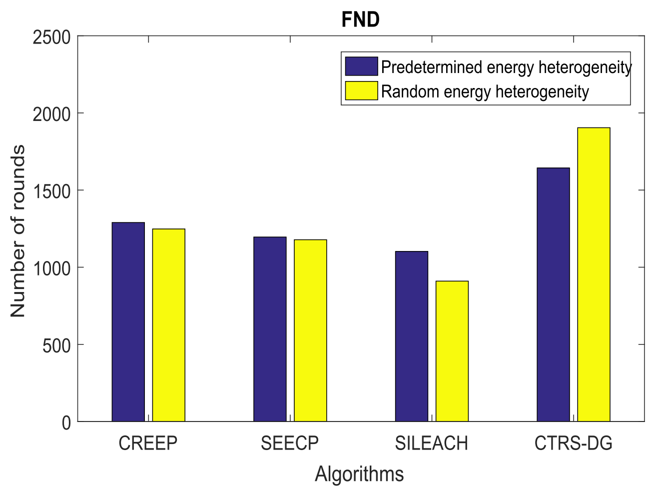

FIGURE 12. Stability (first node dies) in CTRS-DG, SILEACH, CREEP, and SEECP with predetermine and random energy heterogeneity.

Figure 12 shows the network lifetime in terms of FND is enhanced for random energy heterogeneity than predetermine energy heterogeneity while it is decreased for SILEACH, CREEP, and SEECP because in CTRS-DG, entropy coefficient method handles the diverse of node information and updates the weights based on each node criteria while the other algorithms do not handle this diverse. Figures 12 and 13 show that in case of predetermining or random energy

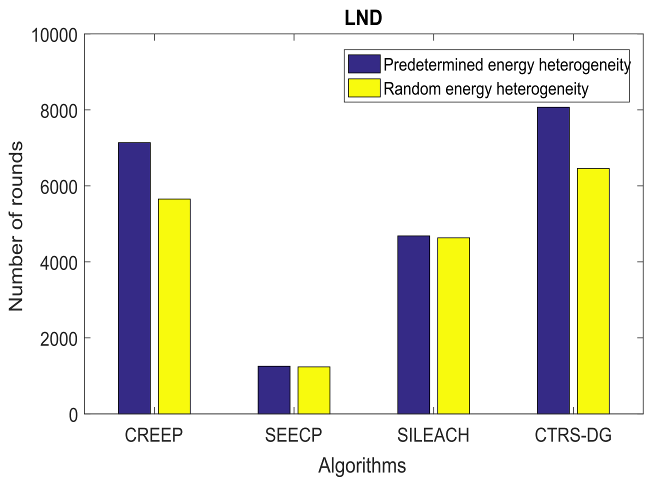

FIGURE 13. Last node dies in CTRS-DG, SILEACH, CREEP, and SEECP with predetermine and random energy heterogeneity.

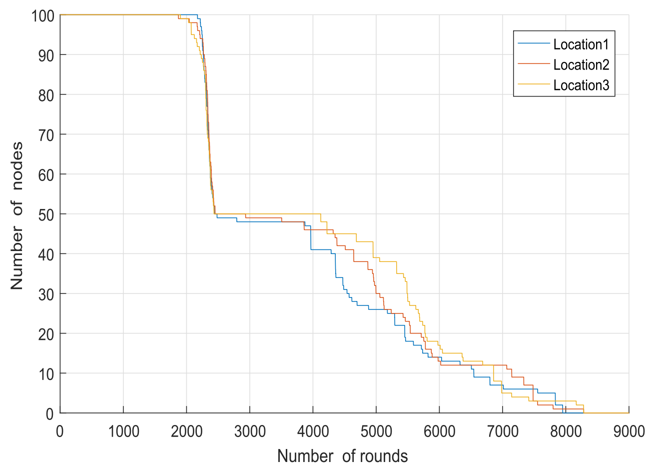

FIGURE 15. Number of alive sensor nodes per round in different network locations of BS.

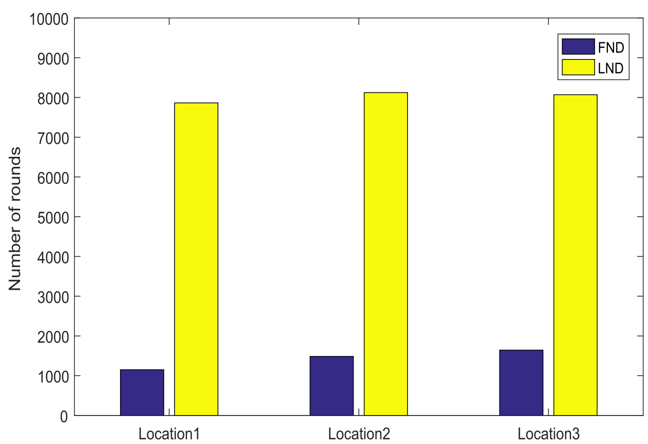

FIGURE 14. FND and LND for different BS Locations.

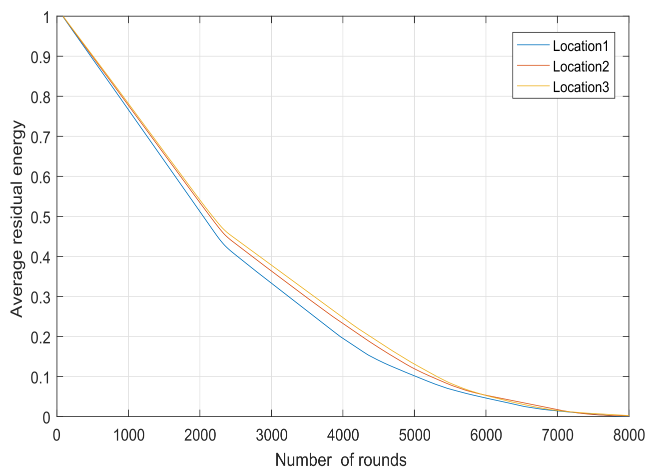

FIGURE 16. Average residual energy per round versus BS locations.

heterogeneity, the performance of the proposed CTRS-DG exceeds the performance of SILEACH, CREEP and SEECP for FND and LND.

C. EFFECT OF BS PLACEMENT

In this test, our goal is to address the effect of BS locations on the performances of the proposed scheme by running the programs with different placements of the BS in the network. We consider 100 sensor nodes with fixed communication range at 30 m are deployed randomly on 100 $m \times 100$ m area, and the different placements of BS are as follows: corner of the network (Location 1), middle of an edge of the network (Location 2), and center of the network (Location 3).

Figures 14, 15 and 16 show that the FND and LND, average residual energy and number of alive sensor nodes per round for different locations of BS. We can note that FND and LND are enhanced because as the BS moves to the network center leads to decrease the length of the paths to the BS and so reduce energy consumption for sensor nodes to forward data to the BS.

D. EVALUATE BEBR ALGORITHM

In this test, we evaluate the performance of BEBR reconstruction algorithm and compare the results with OMP, COSAMP, SP, FBP, BA and PSO algorithms. The experiments considering reconstruction of computer-generated signals for different nonzero coefficient distributions, including Uniform and Gaussian distributions. We investigate reconstruction via Gaussian matrix. The simulations are performed in MATLAB environment and repeated over 500 times using randomly generated K sparse samples of length $N = 100$ from which $M = 60$ random observations are selected via the observation matrix Φ.

Performance Metrics: Average Normalized Mean Squared Error (ANMSE)is used to measure the accuracy of reconstruction algorithms. ANMSE is computed as the average ratio of $\|L\|_2$ norm of the reconstruction error to $\|x\|_2$. We applied an individual observation matrix Φ for each test sample whose entries were drawn from the Gaussian distribution with mean 0 and standard deviation $1/N$.

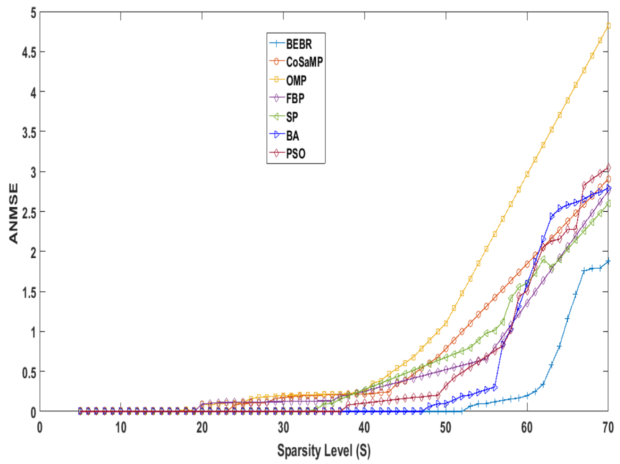

FIGURE 17. ANMSE results over sparsity for Gaussian sparse vectors in CTRS-DG, COSAMP, OMP, FBP, SP, BA and PSO Algorithms.

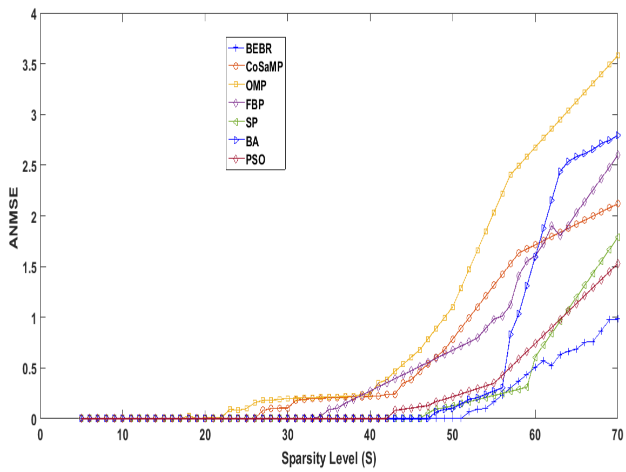

FIGURE 18. ANMSE results over sparsity for Uniform sparse signals in CTRS-DG, COSAMP, OMP, FBP, SP, BA and PSO Algorithms.

Figure 17 shows ANMSE results when the sparse signal's non-zero values are drawn from Gaussian distribution. BEBR algorithm clearly provides lower ANMSE compared with COSAMP, OMP, FBP, LP and SP. In addition, ANMSE in BEBR is started to increase only when $K \geq 53$ while in COSAMP, OMP, FBP, SP, BA and PSO algorithms, ANMSE is started to increase when $K \geq 22$, $K \geq 19$, $K \geq 26$, $K \geq 33$, $K \geq 45$ and $K \geq 37$ respectively as shown in Figure 17 because BEBR combines the advantages of greedy reconstruction based algorithms and BEE algorithm to find the best reconstruction solution.

Figure 18 shows ANMSE results when the sparse signal's non-zero values are drawn from uniform distribution. It shows BEBR algorithm provides the lowest ANMSE compared to COSAMP, OMP, FBP, SP, BA and PSO, because in each round, BEBR supports the search space with the best q columns that helps to find the best solution. ANMSE started to increase when $K \geq 53, K \geq 25, K \geq 21, K \geq 33, K \geq 47$,

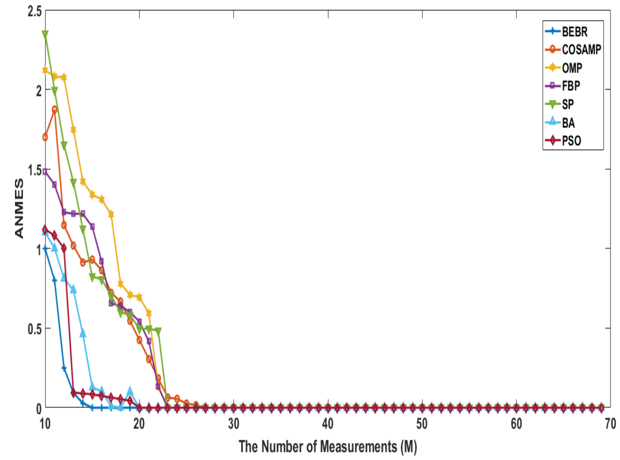

FIGURE 19. ANMSE results over different lengths of measurement vector *M* for Uniform sparse signals using a single Gaussian observation matrix for each *M* in CTRS-DG, COSAMP, OMP, FBP, SP, BA and PSO Algorithms.

$K \geq 47$ and $K \geq 43$ in BEBR, COSAMP, OMP, FBP, SP, BA and PSO respectively.

Finally, we evaluate the performance of BEBR with different lengths of measurement vector M. Sparse signals drawn from uniform distribution with length $N = 100$ is used and M values ranges from 10 to 60 with step size 1. Figure 19 shows that BEBR still provides the lowest ANMSE values compared with COSAMP, OMP, FBP, SP, BA and PSO algorithms.

V. CONCLUSION

In this paper, we have proposed a cluster-based tree routing Scheme for Data Gathering in IoT based heterogeneous wireless sensor networks for periodic applications. The roposed scheme includes a dynamic, distributive, and self-organizing clustering algorithm that benefits from the advantage of the local information of sensor nodes that can be measured in terms of entropy as criteria for CH election and cluster formation. It also includes a new proposed algorithm to form the routing tree as backbone of the network and an effective CS reconstruction algorithm to improve the recovery process at the BS. Our simulation results show that the proposed scheme exceeds the baseline algorithms CREEP, SEECP, and SILEACH in terms of reducing energy consumption and prolonging network lifetime. Moreover, proposed scheme exceeds CS based baseline algorithms LEACH-CS and SEP-CS in terms of reducing energy consumption and prolonging network lifetime. The new proposed BEBR algorithm to reconstruct the original data exceeds the baseline algorithms COSAMP, OMP, FBP, SP, BA and PSO in terms of ANMSE. The proposed scheme achieves our goals to prolong the network lifetime and improves the reconstruction performance.

REFERENCES

[1] S. A. Al-Zubaidi, A. A. Ariffin, and K. A. Al-Qadhi, "Enhancing the stability of the improved-leach routing protocol for WSNs," *J. ICT Res. Appl.*, vol. 12, no. 1, pp. 1–13, Apr. 2018.

[2] B. A. Attea and E. A. Khalil, "A new evolutionary based routing protocol for clustered heterogeneous wireless sensor networks," *Appl. Soft Comput.*, vol. 12, no. 7, pp. 1950–1957, 2012.

[3] F. A. Aderohunmu, J. D. Deng, and M. K. Purvis, "A deterministic energy-efficient clustering protocol for wireless sensor networks," in *Proc. 7th IEEE Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process. (IEEE-ISSNIP)*, Adelaide, SA, Australia, Dec. 2011, pp. 341–346.

[4] S. S. Bhunia, B. Das, and N. Mukherjee, "EMCR: Routing in WSN using multi criteria decision analysis and entropy weights," in *Internet and Distributed Computing Systems* (Lecture Notes in Computer Science), vol. 8729. Cham, Switzerland: Springer, 2014, pp. 325–334.

[5] N. Bagga, S. Sharma, S. Jain, and T. R. Sahoo, "A cluster-tree based data dissemination routing protocol," in *Proc. 11th Int. Multi-Conf. Inf. Process. (IMCIP)*, vol. 54, 2015, pp. 7–13.

[6] E. Callaway, *IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)*, IEEE Standard 802.15.4, 2001.

[7] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large scale wireless sensor networks," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 145–156.

[8] S. Chen, C. Zhao, M. Wu, Z. Sun, H. Zhang, and V. C. M. Leung, "Compressive network coding for wireless sensor networks: Spatio-temporal coding and optimization design," *Comput. Netw.*, vol. 108, pp. 345–356, Oct. 2016.

[9] T. M. Cover and J. A. Thomas, *Elements of Information Theory* (Wiley Series in Telecommunications and Signal Processing). Hoboken, NJ, USA: Wiley, 2006.

[10] A. Chithra and R. S. S. Kumari, "A novel 3-level energy heterogeneity clustering protocol with hybrid routing for a concentric circular wireless sensor network," *Cluster Comput.*, 2017, doi: 10.1007/s10586-017-1310-9.

[11] H. M. N. D. Bandara, A. P. Jayasumana, and T. H. Illangasekare, "A top-down clustering and cluster-tree-based routing scheme for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 7, no. 1, p. 940751, May 2011.

[12] O. Durmaz Incel and B. Krishnamachari, "Enhancing the data collection rate of tree-based aggregation in wireless sensor networks," in *Proc. Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. (SECON)*, Jun. 2008, pp. 569–577.

[13] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, "Signal processing with compressive measurements," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 445–460, Apr. 2010.

[14] X. Du, L. Cheng, and L. Liu, "A swarm intelligence algorithm for joint sparse recovery," *IEEE Signal Process. Lett.*, vol. 20, no. 6, pp. 611–614, Jun. 2013.

[15] X. Du, L. Cheng, and D. Chen, "A simulated annealing algorithm for sparse recovery by l_0 minimization," *Neurocomputing*, vol. 131, pp. 98–104, May 2014.

[16] R. Han, W. Yang, Y. Wang, and K. You, "DCE: A distributed energy-efficient clustering protocol for wireless sensor network based on double-phase cluster-head election," *Sensors*, vol. 17, no. 5, p. 998, 2017,

[17] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. 33rd Hawaii Int. Conf. Syst. Sci.*, Jan. 2000, pp. 1–10.

[18] A. Alsaaffn, A. M. Khedr, and Z. Al Aghbari, "Distributed trajectory design for data gathering using mobile sink in wireless sensor networks," *AEU-Int. J. Electron. Commun.*, vol. 96, pp. 1–12, Nov. 2018.

[19] T. S. Panag and J. S. Dhillon, "Dual head static clustering algorithm for wireless sensor networks," *AEU-Int. J. Electron. Commun.*, vol. 88, pp. 148–156, May 2018.

[20] R. Singh and A. K. Verma, "Energy efficient cross layer based adaptive threshold routing protocol for WSN," *AEU-Int. J. Electron. Commun.*, vol. 72, pp. 166–173, Feb. 2017.

[21] M. Sabet and H. R. Naji, "A decentralized energy efficient hierarchical cluster-based routing algorithm for wireless sensor networks," *AEU-Int. J. Electron. Commun.*, vol. 69, no. 5, pp. 790–799, 2015.

[22] V. Kumar *et al.*, "Multi-hop Communication based optimal clustering in hexagon and Voronoi cell structured WSNs," *AEU-Int. J. Electron. Commun.*, vol. 93, pp. 305–316, Sep. 2018.

[23] S. Hengqiang and Y. Helong, "Application of entropy weight coefficient method in environmental assessment of soil," in *Proc. World Automat. Congr.*, Jun. 2012, Puerto Vallarta, Mexico, pp. 1–4.

[24] O. D. Incel, A. Ghosh, B. Krishnamachari, and K. Chintalapudi, "Fast data collection in tree-based wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 1, pp. 86–99, Jan. 2012.

[25] N. Javaid, T. N. Qureshi, A. H. Khan, A. Iqbal, E. Akhtar, and M. Ishfaq, "EDDEEC: Enhanced developed distributed energy-efficient clustering for heterogeneous wireless sensor networks," in *Proc. Int. Workshop Body Area Sensor Netw. (BASNet)*, vol. 19, 2013, pp. 914–919.

[26] A. M. Khedr and D. M. Omar, "SEP-CS: Effective routing protocol for heterogeneous wireless sensor networks," *Ad Hoc Sensor Wireless Netw.*, vol. 26, nos. 1–4, pp. 211–234, 2015.

[27] D. Kumar, T. C. Aseri, and R. B. Patel, "EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks," *Comput. Commun.*, vol. 32, no. 4, pp. 662–667, 2009.

[28] J. Luo, L. Xiang, and C. Rosenberg, "Does compressed sensing improve the throughput of wireless sensor networks?" in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2010, pp. 1–6.

[29] Y. Liu, W. Zhao, L. Zhu, B. Ci, and S. Chen, "The method of data aggregation for wireless sensor networks based on LEACH-CS," in *Advances in Wireless Sensor Networks* (Communications in Computer and Information Science), vol. 501, L. Sun, H. Ma, D. Fang, J. Niu, W. Wang, Eds. Berlin, Germany: Springer, 2015.

[30] N. Mazumdar and H. Om, "DUCR: Distributed unequal cluster-based routing algorithm for heterogeneous wireless sensor networks," *Int. J. Commun. Syst.*, vol. 30, no. 18, p. e3374, 2017, doi: 10.1002/dac.3374.

[31] N. Mittal, U. Singh, and B. S. Sohi, "A stable energy efficient clustering protocol for wireless sensor networks," *Wireless Netw.*, vol. 23, pp. 1809–1821, Aug. 2017, doi: 10.1007/s11276-016-1255-6.

[32] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.

[33] M. T. Nguyen, K. A. Teague, and N. Rahnavard, "CCS: Energy-efficient data collection in clustered wireless sensor networks utilizing block-wise compressive sensing," *Comput. Netw.*, vol. 106, pp. 171–185, Sep. 2016.

[34] D. T. Pham, M. Castellani, M. Sholedolu, and A. Ghanbarzadeh, "The Bees algorithm and mechanical design optimisation," in *Proc. 5th Int. Conf. Inform. Control, Automat. Robot. (ICINCO)*, 2008, pp. 250–255.

[35] M.-S. Pan, P.-L. Liu, and Y.-P. Lin, "Event data collection in ZigBee tree-based wireless sensor networks," *Comput. Netw.*, vol. 73, pp. 142–153, Nov. 2014.

[36] L. Qing, Q. Zhu, and M. Wang, "Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks," *Comput. Commun.*, vol. 29, no. 12, pp. 2230–2237, 2006.

[37] N. Qiang and X. Qiannan, "Weight optimization method of wireless sensor network based on fuzzy MADMR," in *Proc. 4th Int. Conf. Intell. Comput. Technol. Automat.*, Shenzhen, Guangdong, Mar. 2011, pp. 303–306.

[38] R. Rani, D. Kakkar, P. Kakkar, and A. Raman, "Distance based enhanced threshold sensitive stable election routing protocol for heterogeneous wireless sensor network," in *Computational Intelligence in Sensor Networks*. Berlin, Germany: Springer, 2019, pp. 101–122.

[39] A. Salim, W. Osamy, and A. M. Khedr, "IBLEACH: Intra-balanced LEACH protocol for wireless sensor networks," *Wireless Netw.*, vol. 20, pp. 1515–1525, Aug. 2014.

[40] A. Salim and W. Osamy, "Distributed multi chain compressive sensing based routing algorithm for wireless sensor networks," *Wireless Netw.*, vol. 21, pp. 1379–1390, May 2015.

[41] S. Sharma, R. K. Bansal, and S. Bansal, "Heterogeneity-aware energy-efficient clustering (HEC) technique for WSNs," *KSII Trans. Internet Inf. Syst.*, vol. 11, no. 4, pp. 1866–1888, 2017, doi: 10.3837/tiis.2017.04.003.

[42] M. Shaji and A. S, "Distributed energy efficient heterogeneous clustering in wireless sensor network," in *Proc. 5th Int. Conf. Adv. Comput. Commun. (ICACC)*, Kochi, India, Sep. 2015, pp. 130–134.

[43] K. Shekhar, S. K. Verma, and A. Kumar, "Enhanced threshold sensitive stable election protocol for heterogeneous wireless sensor network," *Wireless Pers. Commun.*, vol. 85, no. 4, pp. 2643–2656, 2015.

[44] S. Singh, A. Malik, and R. Kumar, "Energy efficient heterogeneous DEEC protocol for enhancing lifetime in WSNs," *Eng. Sci. Technol., Int. J.*, vol. 20, no. 1, pp. 345–353, 2017, doi: 10.1016/j.jestch.2016.08.009.

[45] S. Dutt, S. Agrawal, and R. Vig, "Cluster-head restricted energy efficient protocol (CREEP) for routing in heterogeneous wireless sensor networks," *Wireless Pers. Commun.*, vol. 100, no. 4, pp. 1477–1497, 2018.

[46] J. Tian, T. Liu, and H. Jiao, "Entropy weight coefficient method for evaluating intrusion detection systems," in *Proc. Int. Symp. Electron. Commerce Secur.*, Aug. 2008, pp. 592–598.

[47] J. A. Tropp and A. C. Gilbert, ''Signal recovery from random measurements via orthogonal matching pursuit,'' *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[48] E. Triantaphyllou, *Multi-Criteria Decision Making Methods*. Boston, MA, USA: Springer, 2000.

[49] S. Upadhyayula and S. K. S. Gupta, ''Spanning tree based algorithms for low latency and energy efficient data aggregation enhanced convergecast (DAC) in wireless sensor networks,'' *Ad Hoc Netw.*, vol. 5, no. 5, pp. 626–648, 2007.

[50] R. Velmani and B. Kaarthick, ''An efficient cluster-tree based data collection scheme for large mobile wireless sensor networks,'' *IEEE Sensors J.*, vol. 15, no. 4, pp. 2377–2390, Apr. 2015.

[51] Q. Wang *et al.*, ''Assessment of the sustainable development capacity with the entropy weight coefficient method,'' *Sustainability*, vol. 7, no. 10, pp. 13542–13563, 2015.

[52] W. Dai and O. Milenkovic, ''Subspace pursuit for compressive sensing signal reconstruction,'' *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[53] L. Xiang, J. Luo, and A. Vasilakos, ''Compressed data aggregation for energy efficient wireless sensor networks,'' in *Proc. IEEE Sensor, Mesh, Ad Hoc Commun. Netw.*, vol. 11, Jun. 2011, pp. 46–54.

[54] J. Yang, B. Bai, and H. Li, ''A cluster-tree based data gathering algorithm for wireless sensor networks,'' in *Proc. Int. Conf. Autom. Control Artif. Intell. (ACAI)*, Xiamen, China, 2012, pp. 22–25.

[55] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. H. Hanzo, ''A survey of network lifetime maximization techniques in wireless sensor networks,'' *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 828–854, 2nd Quart., 2017.

Authors' photographs and biographies not available at the time of publication.