Received: 23 September 2022 | Accepted: 22 November 2022
DOI: 10.1002/dac.5420

RESEARCH ARTICLE

WILEY

DEICA: A differential evolution-based improved clustering algorithm for IoT-based heterogeneous wireless sensor networks

Sandip K. Chaurasiya[1] | **Arindam Biswas**[2] | **Anand Nayyar**[3] | **Noor Zaman Jhanjhi**[4] | **Rajib Banerjee**[5]

[1]Department of Cybernetics, School of Computer Science, University of Petroleum & Energy Studies (UPES), Dehradun, India

[2]School of Mines & Metallurgy and Centre for IoT and AI Integration with Education Industry Agriculture, Kazi Nazrul University, Asansol, India

[3]Duy Tan University, Da Nang, Vietnam

[4]Taylors University School of Computer Science and Engineering, Taylors University, Subang Jaya, Malaysia

[5]Department of Electronics & Communication Engineering, Dr. B. C. Roy Engineering College, Durgapur, India

Correspondence
Rajib Banerjee, Department of Electronics & Communication Engineering, Dr. B. C. Roy Engineering College, Durgapur, West Bengal, India.
Email: rajib123banerjee@gmail.com

Funding information
SERB, Grant/Award Number: CRD/20201000284

Summary

With the evolution of technology, many modern applications like habitat monitoring, environmental monitoring, disaster prediction and management, and telehealth care have been proposed on wireless sensor networks (WSNs) with Internet of Things (IoT) integration. However, the performance of these networks is restricted because of the various constraints imposed due to the participating sensor nodes, such as nonreplaceable limited power units, constrained computation, and limited storage. Power limitation is the most severe among these restrictions. Hence, the researchers have sought schemes enabling energy-efficient network operations as the most crucial issue. A meta-heuristic clustering scheme is proposed here to address this problem, which employs the differential evolution (DE) technique as a tool. The proposed scheme achieves improved network performance via the formulation of load-balanced clusters, resulting in a more scalable and adaptable network. The proposed scheme considers multiple parameters such as nodes' energy level, degree, proximity, and population for suitable network partitioning. Through various simulation results and experimentation, it establishes its efficacy over state-of-the-art schemes in respect of load-balanced cluster formation, improved network lifetime, network resource utilization, and network throughput. The proposed scheme ensures up to 57.69%, 33.16%, and 57.74% gains in network lifetime, energy utilization, and data packet delivery under varying network configurations. Besides providing the quantitative analysis, a detailed statistical analysis has also been performed that describes the acceptability of the proposed scheme under different network configurations.

KEYWORDS
clustering, differential evolution, energy efficiency, Internet of Things, network lifetime, wireless sensor network

1 | INTRODUCTION

The wireless sensor network (WSN) is defined as a set of sensors connected wirelessly to one another to pursue application-specific operations. The deployed sensors read the surroundings, process the measurement, and communicate that to some central base station (BS). Moreover, the sensors are equipped with tiny power units that come up with a limited amount of energy. Once a node is depleted of its energy, it becomes unable to contribute to the network operations. Thus, if not appropriately handled, early death of the sensors might cause the early network death.[1,2]

Since the very inception of IoT, WSN has always been an integral part of it.[3] WSN serves as a backbone network in a wide variety of IoT-enabled monitoring and sensing systems as depicted in Figure 1. Be it a centralized or decentralized model of IoT, the increasing demand for the IoT infrastructure has enforced the dense deployment of WSN wherein some sensors are equipped with additional features to enable the data exchange over the Internet via a centralized BS. Such network is termed IoT-based heterogeneous WSN (HWSN).

Among all the possible network operations, transmission consumes most of the nodes' power, and hence, the development of energy-efficient schemes pertaining to the transmissions and routing, especially at the physical and network layer, has attracted the researchers a lot for the success of IoT-based HWSN.

In this regard, clustering has been recognized as the most significant tool which enables energy-efficient operations leading to the improved network lifetime. As in previous studies,[4–8] clustering refers to the process of grouping the nodes with some common attributes like nodes' proximity, nodes' remaining energy levels, and node-to-sink distance. In a clustered network, the network is partitioned into a finite number of clusters. The nodes in such network architecture are categorized as cluster heads (CHs) and cluster members. There would be precisely one CH to serve the members in every cluster. The primary responsibility of a cluster member is to read the surroundings and transfer the measurements to the respective CH. The CH in turn collect the data from all such members assigned to it and transmit it to the BS after performing data aggregation. From the above discussion, it is pretty obvious that the CHs perform more energy-intensive tasks than that by the member nodes. Since the CHs are selected from the same set of sensors, the additional responsibilities being executed by the CHs may drain the battery very quickly. This may lead to the early network death. To counter this critical challenge, many works such as previous studies[9–11] have reported the introduction of energy-enriched control nodes, termed gateways or relay nodes to act as CH. These control nodes usually have higher energy levels in comparison to the normal nodes and hence, are more suitable for energy-intensive network operations. This introduction of specialized nodes with higher energy makes the network a two-level heterogeneous network. However, the deployed nodes of such IoT-based HWSN are battery operated, which necessitates intelligent usage of the available resources to ensure long and steady network operation over time.

Here, the specially deployed control nodes, acting as the CHs, get associated with the normal nodes to form a finite number of clusters in the network. In this cluster formation process, an appropriate scheme for the assignment of normal nodes to energy-enriched control nodes (or gateways) must be formulated very carefully. As a matter of fact, if not handled adequately, the randomly distributed sensor nodes may result in poor clusters like some clusters might become

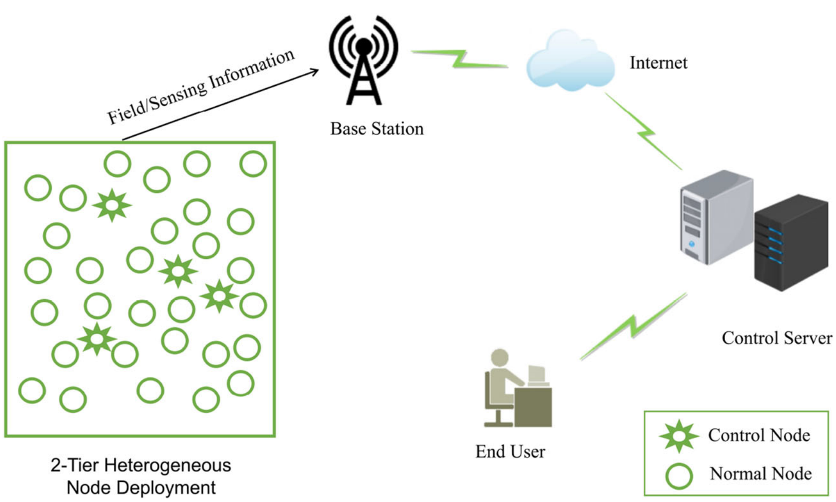


FIGURE 1 IoT-based two-tier heterogeneous wireless sensor network

overcrowded and some might have a lesser number of nodes. This, in turn, forces the deployed control nodes to suffer from highly varying energy consumption as some may be overloaded, and some might be assigned with a few members only. Such an uneven distribution of nodes among the clusters may get the deployed control nodes suffering from highly varying energy consumption. Some may be overloaded, and some be assigned with a few only. Thus, in addition to energy consumption by the normal and control nodes, the assignment of the normal sensors to the control nodes or equivalently to the CHs matters substantially huge for the development of energy-efficient clustering solutions in the IoT-based HWSN.

As indicated above, clustering problem requires consideration of multiple parameters, it can also be thought of as a multivariate optimization problem. For solving this class of problems, problem-independent metaheuristic techniques have proved their significance to an appreciable extent. A huge number of metaheuristic algorithms, namely, genetic algorithm (GA), particle swarm optimization (PSO), genetic programming (GP), teaching–learning-based optimization (TLBO), differential evolution (DE), evolutionary programming (EP), and evolution strategies (ES), are already being used for the above-mentioned purpose of clustering in IoT-based HWSN.

The main motivation for this proposed work is to develop a load balancing bio-inspired DE-based clustering technique to enhance network longevity. DE-based strategies have proved their worth over their peers such as GA and PSO[12] and have attained huge popularity in this regard due to their simplicity, robustness, and faster rate of convergence.

1.1 | Major contributions and organization of the paper

The proposed scheme contributes majorly in

- Formulating a fitness function that enables
  - clusters with even load distribution.
  - minimized communication cost among the nodes of a cluster.
  - maximum resource utilization.
- Developing an improved intermediate phase to support the construction of the load-balanced clusters.
- Developing a DE-based clustering scheme that employs the formulated fitness function and the intermediate phase to improve the network performance.
- Analyzing the performance of the proposed scheme to establish its efficacy over existing state-of-the-art schemes in respect of
  - quality of clusters under varying network configurations.
  - statistically justified results.
  - various network performance criteria like energy consumption, network lifetime, and throughput.

The remaining paper is arranged in a way that the existing works have been listed in Section 2. Section 3 briefs the general scheme of DE. In addition, Section 3 also discusses the other models and assumptions used. The proposed scheme is explained along with all its constituting phases in Section 4. Section 5 analyzes the simulation results to prove the proposed scheme's supremacy over its existing peers. At last, Section 6 summarizes the entire work while stating the future scope.

2 | LITERATURE REVIEW

Due to their significant impact in efficiently solving a complex problem, metaheuristic schemes like GA, DE, EP, ES, PSO, TLBO, and ant colony optimization (ACO) are being used in almost every field of engineering. These schemes have also proved their significance in the domain of IoT-based WSNs, and hence, numerous contributions based on these techniques can be easily found in the literature like in previous studies.[13–19]

This section lists some of the significant contributions developed to serve the intended purpose of featuring optimized clusters to enable energy-efficient network operations as follows:

Bari et al[9] offered a clustering solution for the two-tier HWSN based on the GA. As stated in Section 1, the two-tier HWSN refers to a network that deploys two types of sensor nodes—traditional normal sensors and energy-enriched control nodes or gateways. In their proposed scheme, the authors adopted the method of roulette-wheel selection to select the individuals from the population. The roulette-wheel selection method ensures that the individuals with higher fitness values are selected. Here, the fitness function considers only a single parameter, network lifetime, in its formulation. For the production of new offspring, k-point crossover or uniform crossover is implemented randomly. Moreover, the scheme applies the mutation to improve an individual's fitness further. The scheme performs well over the traditional multihop schemes; however, provisioning only the network lifetime for the definition of its fitness function limits the scheme's performance.

In one of the works,[11] a DE-based clustering algorithm (DECA) for a two-tier WSN was proposed. In the proposed scheme, the network comprises two types of nodes—one with higher energy nodes and another with normal energy levels. The authors attempted balanced cluster formation via an appropriate fitness function in their proposed scheme. The proposed fitness function considers the gateway nodes' lifetime and distance from their respective gateways. A local improvement phase was also introduced that reassigned a node from the overloaded gateway to a control node with a comparatively low load. However, neither the fitness function nor the local improvement phase took the cluster size of cluster length into account.

Osamy et al[19] proposed an intelligent data collection technique for IoT-enabled HWSN in smart environments. Osamy et al[19] emphasize on achieving energy-efficient network operation to enhance the overall network lifetime by figuring out the energy-aware disjoint dominating sets acting as data collector nodes in each round of the network operation. For this, the authors employ a swarm intelligence-based scheme under the aforesaid title. The network operation comprises two phases—the collector nodes selection phase and data gathering path formation and collection phase. Osamy et al[19] also apply the idea of sleep–awake scheduling to improve the network lifetime further.

Li et al[20] proposed a metaheuristic routing scheme, DE-LEACH. DE-LEACH is a DE version of LEACH for environmental monitoring WSN. Li et al[20] take the parameters such as residual energy and spatial distribution of nodes into account for network partitioning into a finite number of clusters. The network operation consists of four phases—initial cluster partitioning, nodes' status collection through the use of auxiliary CHs within their respective clusters, DE-based optimized CHs determination, and optimized cluster formation.

Li et al[20] prove its supremacy over the LEACH,[21] and LEACH-C[22] through various simulation results.

In another work,[23] the author proposed a hybrid metaheuristic technique that calls the DE and simulated annealing (SA) together to improve the network lifetime in WSN. DE provides the locally optimized solution, which can be further improved as a global optimum solution via SA. The scheme proposes four phases of network operation as in any conventional DE-based scheme—population initialization, mutation, crossover, and greedy selection. Instead of using the traditional random population-initialization approach, it calls the "opposite point method" as in Brest et al[24] for the population vectors' initialization. Moreover, Potthuri et al[23] propose run-time selection of applicable mutation scheme based on a threshold value (0.5 here) from two available options—$DE/rand/1$ and $DE/target-to-best/1$. The scheme randomly generates a number, say $\zeta \in (0,1)$. If it is less than the chosen threshold value, the former one is applied otherwise, the latter one. For the formulation of the fitness function, Potthuri et al[23] take the ratio of a node's energy to its cluster's energy. A Gaussian distribution-based blending rate is used for the crossover purpose.

Potthuri et al[23] prove its supremacy over the customary DE-based scheme with respect to the network throughput, energy consumption, and network lifetime.

In Randhawa and Jain,[25] a swarm intelligence-based multiobjective metaheuristic technique termed multiobjective load balancing clustering (MLBC) is proposed. The scheme employs multiobjective PSO (MOPSO) to address two major problems: network reliability and energy efficiency in WSN. The scheme provisions energy-efficient network operations by appropriately taking the average residual energy of the CHs into account. Randhawa and Jain[25] minimize the communication cost among the nodes in their respective clusters and achieve network reliability. MLBC considers just a single parameter in its objective function definition—CHs' average residual energy. Moreover, MLBC shuffles the role of the next-hop nodes and CHs in each iteration to ensure even load distribution among the sensor nodes.

Gupta and Saha[26] proposed a novel scheme that utilizes a mixture of two popular metaheuristic schemes—DE and artificial bee colony (ABC). The authors considered three decision parameters—average energy of CHs, average intracluster distance, and the transmission delay in defining the scheme's fitness function. One of the main objectives of their proposed scheme, ABC with DE (ABC-DE) is the even distribution of network load among the CHs. Gupta and Saha[26] also provision run-time repositioning of the mobile sinks to achieve energy efficiency further.

Iwendi et al[27] proposed a scheme that addresses energy optimization in an IoT-enabled WSN. Iwendi et al[27] use a hybrid approach in which it calls for the SA and whale optimization algorithm (WOA) methods to locate the best possible CH candidates for each clusters. The corresponding fitness function in Iwendi et al[27] considers a set of parameters like nodes's remaining energy, node-to-BS distance, temperature, load, and delay to select the fittest node for the role of CH. The fitness function in Iwendi et al[27] appoints the sensors with the maximum remaining energy and minimum load, delay, node-to-BS distance, and temperature as the CHs in each round of network operation.

In another work,[28] the authors proposed a quorum system based on artificial intelligence (AI) to facilitate the energy-efficient network operations in WSN. In their proposed AI-based system, the authors have attempted fastening the process of neighbor discovery while lowering the latency in the network. Moreover, the weighted load balancing feature adopted in the scheme, Ponnan et al[28] minimize the energy consumption and thus, enhances the network lifetime. The authors have proven the outperformance of Ponnan et al[28] over its peers in respect of network lifetime, coverage, energy efficiency, and latency through an extensive set of experimentations.

Sackey et al[29] proposed a brain storm optimization (BSO)-inspired metaheuristic energy-efficient clustering technique entitled energy-efficient clustering-BSO (EEC-BSO). The scheme is based on the swarm intelligence and applies the human brainstorming process to locate the most appropriate solutions in the search space. Sackey et al[29] focus on devising energy-efficient clusters. It has been provisioned that the nodes not participating in the information exchange would be sent to sleep mode to minimize the energy consumption. Moreover, the fitness function in Sackey et al[29] considers three major parameters for network partitioning: residual energy, packet data rate, and coverage. Using various simulations, Sackey et al[29] confirm its outperformance over schemes like LEACH, LEACH-C, LEACH-BSO, and energy-efficient clustering schema (EECS) with respect to reduced energy consumption, improved coverage, and improved data packet rate.

A DE-based energy-efficient clustering method for WSN is proposed in Ghahramani and Laakdashti.[30] The proposed scheme targeted the DECA for further improvement. In their proposed scheme, the authors updated the mutation function to consider the target vector along with the best vector and two other randomly chosen population vectors. The proposed scheme's fitness function considers nodes' energy along with the gateways' energy as the main decision parameters. However, Scheme-[30] (here, Scheme-[30] refers to the scheme implemented in Ghahramani and Laakdashti[30]) does not consider the cluster population in deciding the clusters like its predecessor DECA.

For the energy-efficient network operations, formation of load-balanced clusters in the network is a key that has been the least addressed in above-mentioned works. The present work emphasizes on the development of energy-efficient clusters for a two-level IoT-based HWSN such that it ensures:

- load-balanced cluster formation.
- reduced intracluster communication among the nodes and their respective CH to save the node's energy.
- maximum resource utilization.
- consistent performance under varying network configuration.

3 | PRELIMINARIES

3.1 | DE—A brief description

The DE scheme is a stochastic problem-independent metaheuristic technique that facilitates an optimized solution from the available search space.[31,32] It starts with a randomized set of initial population vectors, termed target vectors. Each target vector goes through the mutation process followed by a crossover phase. The target vector undergoing the mutation phase is called the donor or the mutant vector. After going through the crossover phase, the donor vector becomes a trial vector. The new offspring comprises vectors that come out as the result of greedy selection in terms of fitness values between the target vector and trial vector pairs.

More illustratively, each pair of target vector and the respective trial vector is checked against the value of the fitness function defined in the scheme, and whosoever has the higher fitness value finds a place in the offspring of the next generation. It is to be noted here that the donor/mutant vectors are never considered for the new offspring.

Since the inception of DE, many variants have been introduced. These variants are represented by a notation similar to that being used for queueing system, like $DE/x/y/z$. Here, DE refers to DE, x refers to the type of vectors being mutated (like random vectors or best vectors or the target vector), y refers to the number of difference vectors being used for mutation purpose, and finally, z refers to the crossover technique being followed. The popular crossover techniques are the binomial crossover and the exponential crossover.

A few of the popular DE schemes are $DE/rand/1/exp$, $DE/rand/2/exp$, $DE/rand/1/bin$, $DE/best/1/exp$, $DE/best/2/bin$, $DE/target-to-best/1/bin$, and so on. In this work, what has been followed is $DE/best/1/bin$.

This work adopts $\mathbf{DE/best/1/bin}$.

3.2 | Models and assumptions

In this section, the various assumptions along with the adopted network and energy-consumption models are described.

3.2.1 | Network model

As in Xiang et al[33] and Ramteke et al,[34] the operational WSN in the IoT-based HWSN is adopted here as a digraph, $G=(V,L)$, where V refers to the set of various sensor nodes deployed for monitoring and control purpose and L refers to the set of directed links. Here, V comprises two types of nodes—normal nodes and control nodes. The control nodes are equipped with higher energy compared to the normal nodes' energy, and hence, yielding two-level heterogeneous networks. The normal nodes are deployed for environment sensing, and control nodes are appointed for communicating the field data to the BS. L denotes the set of all the communication links between the normal nodes to control nodes and control nodes to the BS.

The detailed assumptions about the network in the proposed scheme can be listed as follows:

- The sensor nodes in the IoT-based HWSN are distributed randomly across the field, and once deployed, nodes cannot change their respective locations.
- Each participating nodes in the deployed network is assigned with a universal identification number (UIN).
- Two-tier network architecture deploys two types of nodes with different initial energy. The control nodes equipped with higher initial energy act as CHs, and those with comparatively low initial energy are solely responsible for environment sensing and monitoring. Moreover, all the energy-intensive tasks are to be performed by the energy-enriched control nodes.
- Both kind of nodes are equipped with the power control features.
- The BS is static and situated in the middle of the sensing field.
- The sensor nodes are periodically sensing the environment in order to generate the data. In brief, the continuous data flow model is adopted for the working of the nodes.

3.2.2 | Energy-consumption model

As in many of the schemes like previous studies,[11,26,30,34–39] the proposed scheme uses the *First-Order Radio Model* for the computation of all energy expanses in the network.

4 | PROPOSED SCHEME—A DE-BASED IMPROVED CLUSTERING ALGORITHM (DEICA)

DEICA employs the centralized BS for suitable cluster formation as in DECA and Scheme-[30]. Once the clusters are formed, further network operations are taken care of by the network nodes themselves.

At the beginning of the scheme, network setup occurs first. The network setup process comprises two different phases- the bootstrapping and clustering phases. In the bootstrapping phase, the normal nodes start broadcasting their UINs to be sensed by the control nodes in their communication range. The control nodes then send the collected information to the centralized BS for balanced cluster formation. In the clustering phase, clusters are formulated by following the DE strategy detailed below. The resulting clusters are then improved through a local improvement phase (Section 4.5) ensuring further load balancing in the network. Once the clusters are formed, each control node is informed of its responsibility and respective member nodes. The normal nodes are then briefed on their respective clusters and TDMA schedules by the respective CHs. After that, the network operations are divided into rounds wherein the data exchange is provisioned.

Moreover, the detail schema of the proposed scheme is described in the Figure 2 and has been explained in the subsequent subsections. The proposed scheme also incorporates a local improvement phase like in DECA but with an updated policy to ensure more balanced clusters.

4.1 | Initialization

The initialization scheme is inherited from Kuila and Jana[11] wherein each population vector indicates the complete assignment of network nodes to every control node. For example,

$$\vec{\Lambda}_{i,T} = [\lambda_{1,i,T}, \lambda_{2,i,T}, \lambda_{3,i,T}, \ldots, \lambda_{N,i,T}], \tag{1}$$

where $\vec{\Lambda}_{i,T}$ is the ith target vector in the population of Tth generation; $\{\lambda_{1,i,T}, \lambda_{2,i,T}, \lambda_{3,i,T}, \ldots, \lambda_{N,i,T}\} \in (0,1)$ are chosen randomly, where $\lambda_{j,i,T}$ refers to s_j^{th} node assigned to one of control nodes, say m, as follows:

$$l = ceiling(v_{j,i,T} * |CommCH(s_j)|). \tag{2}$$

Here, $CommCH(s_j)$ refers to the set of all the control nodes found in the communication range of the node, s_j. And

$$CH_m = index(CommCH(s_j), l). \tag{3}$$

Once this initialization phase is complete, the fitness value of each target vectors are recorded before going ahead with the mutation phase.

The fitness function is described in the next subsection.

4.2 | Fitness function

The prime objective of this work is to devise a fitness function that assures network longevity via the load-balanced clusters. Since the network lifetime is defined as the death of the first control node/gateway, it is important to bring it under consideration in the definition of fitness function, along with other parameters like distance of the nodes from their respective control nodes, cluster energy, node degrees, and size of the clusters. It can be intuitively observed that a quality cluster requires its member nodes to be at a minimum distance from the control node and to be characterized

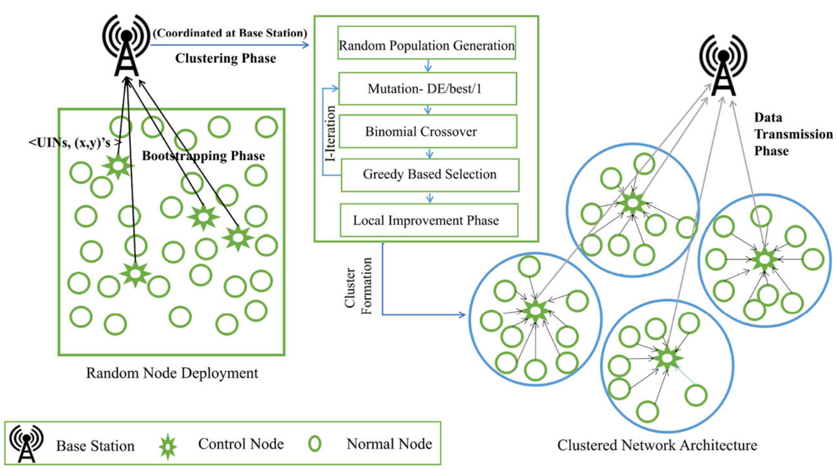


FIGURE 2 DEICA—flowchart

with a higher node degree to reduce intracluster communication costs. Similarly, for a balanced network partitioning in a finite number of clusters, the number of member nodes per cluster must be almost the same inducing clusters with approximately the same level of energy too. Thus, the fitness function must incorporate the standard deviation of the control nodes' lifetime, the standard deviation of average cluster distance, the standard deviation of average cluster size, the standard deviation of average cluster energy, cluster energy, and node degree in its formulation.

The fitness function (ϕ_f) characterizing all the aforementioned required considerations can be derived as follows:

4.2.1 | Standard deviation of the lifetime of control nodes

For the balanced network partitioning, it is considered that all the cluster-inducing control nodes should have similar lifetimes where the lifetime of a control node (L_{CN}) can be defined as follows:

$$L_{CN_i} = \frac{ResidualEnergy_{CN_i}}{EnergyConsumption_{CN_i}}, \tag{4}$$

where $ResidualEnergy_{CN_i}$ and $EnergyConsumption_{CN_i}$ refer to the remaining energy and energy consumption by the ith control node. The control node consumes its energy in performing numerous network operations, such as receiving data packets from the member normal nodes, aggregating them, and forwarding the aggregated data to the BS repeatedly in successive network rounds. More illustratively,

$$EnergyConsumption_{CN_i} = ClusterSize_i * (E_{RX} + E_{DA}) + E_{TX}(CN_i, BS), \tag{5}$$

where $ClusterSize_i$ is the size of the ith cluster; E_{RX} and E_{DA} are the energy consumed in receiving and aggregating a data packet; and $E_{TX}(CN_i, BS)$ is the energy consumed by the ith control node (acting as CH) in transmitting the data packet to the BS. Now, for the balanced lifetime of the control nodes, their standard deviation from the average lifetime should be the minimum, that is, if

$$\mu_{Life_{CN}} = \frac{1}{k} \cdot \sum_{i=1}^{k} Life_{CN}, \tag{6}$$

where k is the number of CHs and $\mu_{Life_{CN}}$ is the mean lifetime of the control nodes.

Then,

$$\sigma_{Life_{CN}} = \sqrt{\frac{1}{k} \sum_{i=1}^{k} (\mu_{Life_{CN}} - Life_{CN_i})^2}, \tag{7}$$

where $\sigma_{Life_{CN}}$ is the standard deviation of the lifetime of the CHs.

And it can be intuited easily that

$$\phi_f \propto \frac{1}{\sigma_{Life_{CN}}}. \tag{8}$$

4.2.2 | Standard deviation of the average cluster distance

Since the member normal nodes are only to communicate with their respective control nodes throughout their life, it is quite obvious that lesser the distance between the member and control node, lesser the energy consumption. The same can be mapped to fitness function (ϕ_f) in such a way that if $AvgDist_{CN_i}$ be the average distance of a node from the control node in the ith cluster and $\mu_A vgDist_{CN}$ be the mean of the average cluster distances for all the clusters formed defined below:

$$\mu_{AvgDist_{CN}} = \frac{1}{k}\sum_{i=1}^{k} AvgDist_{CN_i}. \tag{9}$$

Then,

$$\sigma_{AvgDist_{CN}} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(\mu_{AvgDist_{CN}} - AvgDist_{CN_i}\right)^2}, \tag{10}$$

where $\sigma_{AvgDist_{CN}}$ is the standard deviation of the average cluster distance.

And accordingly,

$$\phi_f \propto \frac{1}{\sigma_{AvgDist_{CN}}}. \tag{11}$$

4.2.3 | Standard deviation of the average cluster size

For partitioning the network into equally loaded clusters, consideration of this metric can be a key. If $IdealCS$ is the suggested number of nodes per cluster defined as N/k where N being the number of normal nodes and k is the number of control nodes deployed, then standard deviation of the average cluster size can be defined as follows:

$$\sigma_{IdealCS} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(IdealCS - ClusterSize_{CN_i}\right)^2}, \tag{12}$$

where $ClusterSize_{CN_i}$ is the size of the cluster associated with ith control node.

And

$$\phi_f \propto \frac{1}{\sigma_{IdealCS}}. \tag{13}$$

4.2.4 | Standard deviation of the average cluster energy

To further ensure the load-balanced clusters, standard deviation of the average cluster energy (σ_{AvgCE}) for every clusters can also be considered as a key parameter. It can be intuitively concluded that load-balanced clusters refer to the clusters with approximately same level of energy; thus, indicating clusters with lower standard deviation of the average cluster energy. More illustratively,

$$\sigma_{AvgCE} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(\mu_{CE} - CE_i\right)^2}, \tag{14}$$

where μ_{CE} is the mean cluster energy for the entire network partitioning and CE_i is the cluster energy for the ith cluster.

Moreover,

$$\phi_f \propto \frac{1}{\sigma_{AvgCE}}. \tag{15}$$

4.2.5 | Cumulative average cluster energy

Cumulative average cluster energy (C_{AvgCE}) refers to the summation of average cluster energies for each of the clusters. Average cluster energy ($AvgCE_i$) for the ith cluster can be defined as follows:

$$AvgCE_i = \frac{\sum_{j=1}^{m}(ResidualEnergy_j{}^i + ResidualEnergy_{CN_i})}{(m+1)}, \tag{16}$$

where m is the cluster length.

And

$$C_{AvgCE} = \sum_{i=1}^{k} AvgCE_i. \tag{17}$$

For the better fitness function, C_{AvgCE} needs to be maximized as it ensures nodes are adequately distributed over the clusters formed. Hereby,

$$\phi_f \propto C_{AvgCE}. \tag{18}$$

4.2.6 | Cumulative average node degree per cluster

The degree of a node (deg) refers to the number of normal nodes which are in close proximity of the node say in the sensing range of the node. To restrict a node from being a member of a more distant control node, it is expected to group the nodes which are in close proximity. Thereby, for a better fitness function, cumulative average node degree (C_{AvgND}) per cluster must be maximized.

$$C_{AvgND} = \sum_{i=1}^{k}\left(\frac{1}{m}\sum_{j=1}^{m} deg_j^i\right). \tag{19}$$

And accordingly,

$$\phi_f \propto C_{AvgND}, \tag{20}$$

From Equations (8), (11), (13), (15), (18), and (20),

$$\phi_f \propto \frac{C_{AvgCE} * C_{AvgND}}{\sigma_{Life_{CN}} * \sigma_{AvgDist_{CN}} * \sigma_{IdealCS} * \sigma_{AvgCE}}. \tag{21}$$

Or

$$\phi_f = \frac{K * C_{AvgCE} * C_{AvgND}}{\sigma_{Life_{CN}} * \sigma_{AvgDist_{CN}} * \sigma_{IdealCS} * \sigma_{AvgCE}}, \tag{22}$$

where K is the proportionality constant, and without loss of generality, it can be set to 1. And hence,

$$\phi_f = \frac{C_{AvgCE} * C_{AvgND}}{\sigma_{Life_{CN}} * \sigma_{AvgDist_{CN}} * \sigma_{IdealCS} * \sigma_{AvgCE}}, \tag{23}$$

or equivalently, from Equations (7), (10), (12), (14), (17), and (19),

$$\phi_f = \frac{k^2 * \sum_{i=1}^{k} \frac{\sum_{j=1}^{m}(ResidualEnergy_j{}^i + ResidualEnergy_{CN_i})}{+(m+1)} * \left(\frac{1}{m}\sum_{j=1}^{m} deg_j^i\right)}{\sqrt{\sum_{i=1}^{k}(\mu_{Life_{CN}} - Life_{CN_i})^2 * \sum_{i=1}^{k}(\mu_{AvgDist_{CN}} - AvgDist_{CN_i})^2 * \sum_{i=1}^{k}(IdealCS - ClusterSize_{CN_i})^2}} * \frac{1}{\sqrt{\sum_{i=1}^{k}(\mu_{CE} - CE_i)^2}}. \tag{24}$$

Every population vector is to be evaluated against this fitness function: the higher the value, the better the suitability of the candidate vector.

4.3 | Mutation

In the proposed scheme, $DE/best/1$ has been opted as the mutation strategy. More illustratively, the mutation strategy can be well interpreted by the following expression:

$$\vec{\Psi}_{i,T} = \vec{\Lambda}_{best,T} + F(\vec{\Lambda}_{r_1,T} - \vec{\Lambda}_{r_2,T}), \tag{25}$$

where $\vec{\Lambda}_{best,T}$ and $\vec{\Lambda}_{r_1,T}, \vec{\Lambda}_{r_2,T}$ denote the best vector and the two randomly chosen vectors from Tth generation such that i, r_1, and r_2 are the three integers $\in [1,P]$ (P being the population size) and $i \neq r_1 \neq r_2$. F refers to the scaling factor that can assume any value between (0,2).

4.4 | Crossover

After getting through the mutation phase, each vector passes through the crossover or recombination phase which has been chosen here a binomial one as follows:

$$\omega_j = \begin{cases} \psi_j & \text{if } \zeta \leq Cross_r \text{ OR } j{=}\rho, \\ \lambda_j & \text{if } \zeta > Cross_r \text{ AND } j \neq \rho, \end{cases} \tag{26}$$

where $Cross_r$ is the crossover rate, ρ is a randomly chosen index from the set $\{1,2,3,...,|\text{decision variable}|\}$, $\zeta \in (0,1)$ is chosen randomly, and ω_j, ψ_j, and λ_j refer to the jth variable in trial vector, donor vector, and target vector, respectively. Once the crossover is over, the best of the target vector and corresponding trial vector is chosen on the basis of their fitness value to be a part of the population vector for the next generation.

4.5 | Local improvement phase

Similar to Kuila and Jana,[11] DEICA also provisions local improvement phase. However, instead of executing it at the end of every iteration, DEICA calls it to improve the quality of the finally obtained generation only. In other words, DEICA improves the last set of offspring through this phase.

In this phase, the ideal cluster size can be computed by dividing the total number of nodes by the number of control nodes deployed. Then, the above-formed DE-based clusters are checked for their respective sizes. If the size of the cluster is found greater than the ideal one, the excess nodes are distributed randomly over the nearby suitable clusters. Here, the suitability of the clusters for the nodes reallocation is determined based on two considerations—the size of the destination cluster must be less than the ideal size, and the destination control node must be at the least possible distance among all control nodes. The process is applied over each resultant cluster formed successively.

4.6 | Data transmission phase

In this phase, the member nodes transmit the data to their respective control node acting as CHs. Afterward, the control nodes aggregate the data packets collected and then forward to the BS. The process continues over the rounds till the nodes and controls nodes have sufficient residual energy.

Moreover, the proposed scheme is also summarized into the following algorithm.

4.7 | Algorithm—DEICA

Input:

- N_s: Number of normal sensors deployed randomly
- k: Number of deployed control nodes (CN)
- $\phi_f()$: Fitness evaluation function
- F: Mutation or Scaling factor
- I: Number of iteration
- $Cross_r$: Crossover rate

Algorithmic Steps:

◇ START

%% BOOTSTRAPPING & CLUSTERING PHASE %%

◇ for i ← 1 : N_s
◇   Status Broadcast by $Node_i$
◇ endfor
◇ for i ← 1 : k
◇   Node's Information Transmission(CN_i → BS)
◇ endfor
◇ Generation of initial population (P) wherein every vector (say Λ_i) denotes the complete assignment of the normal sensors to k control nodes
◇ for i ← 1 : P_size

%% P_size → Size of the population

◇ $\phi_f(\Lambda_i)$

%% fitness value of i^{th} vector in P

◇ endfor
◇ for i ← 1 : I
◇   for j ← 1 : P_size
◇ $\Psi_j = \Lambda_{best} + F(\Lambda_{r_1} - \Lambda_{r_2})$
◇ $\Omega_j = [\omega_j^l]$

$$\text{where, } \omega_j^l = \begin{cases} \psi_j^l, & \zeta \leq Cross_r \;\; \| \;\; l = \rho \\ \lambda_j^l, & \zeta > Cross_r \;\; \&\& \;\; l \neq \rho \end{cases}$$

◇   endfor
◇   for j ← 1 : P_size
◇ $\phi_f(\Omega_j)$

%% fitness value of j^{th} trial vector

◇   if ($\phi_f(\Omega_j) > \phi_f(\Lambda_j)$)
◇   Update P

%% Greedy selection strategy to update the population

◇   endif
◇   endfor
◇ endfor
◇ $LocalImprovement([Cluster])$

%% Local Improvement Phase accepting current cluster formation for further refinement

%% DATA TRANSMISSION PHASE %%

◇ while(all the CNs are alive)
◇   for i ← 1 : k
%% for every cluster
◇   for j ← 1 : cs_i
%% $cs_i \rightarrow$ size of the i^{th} cluster
◇   DataTransmission($Node^i_j \rightarrow CN^i$)
◇   endfor
◇   DataTransmission($CN^i \rightarrow BS$)
%% aggregated data transmission to the base station by i^{th} CN
◇   endfor
◇ endwhile
◇ END

Local Improvement([Cluster])

o for i ← 1 : k
o   if $sizeof(Cluster(i)) > IdealCS$
o   for j ← 1 : k
o   if i==j
o   continue;
o   else
o   if $sizeof(Cluster(j)) < IdealCS$ && $dist(CN_j, CN_i) == min^K_{l=1\&l\neq j,i}[(dist(CN_j, CN_l)]$
o   shift $min(IdealCS - sizeof(Cluster(i)),\ IdealCS - sizeof(Cluster(j)))$ random nodes in $Cluster(j)$ from $Cluster(i)$
o   endif
o   endif
o   endfor
o   endif
o endfor

5 | PERFORMANCE ANALYSIS

This section describes and analyzes the various results obtained by performing an extensive set of experimentations to support the claim for DEICA's supremacy over the existing schemes.

5.1 | Experimental environment

The experiments have been conducted under varying network configurations like with different node deployments in the field, say 100, 200, 300, 400, and 500. In addition to the normal nodes mentioned above, four different clustering scenarios have been explored by deploying 15, 20, 25, and 30 control nodes (gateways). The BS is situated at the center of the $200 \times 200\ m^2$ sensing field precisely at (100 m, 100 m).

Figure 3 describes an exemplary simulation interface for the network operation with two different deployment instances, networks with 15 control nodes and 30 control nodes. The nodes labeled with "G" represent the control nodes (or gateways), and that labeled with "BS" refers to the base station. Every other node is representing the normal nodes in the two-tier IoT-based HWSN.

All the experiments and analyses have been performed in MATLAB.

Through the various experimentations, we have demonstrated the following:

1. The efficacy of the proposed scheme, DEICA in formulating the load-balanced clusters with respect to the ones obtained from DECA and Scheme-[30] (detailed in Section 5.3.1).
2. The supremacy of DEICA with respect to the various network performance measurement criteria like network energy consumption, network lifetime, and packet delivery at the BS over the existing schemes- DECA and Scheme-[30] (detailed in Section 5.3.2).

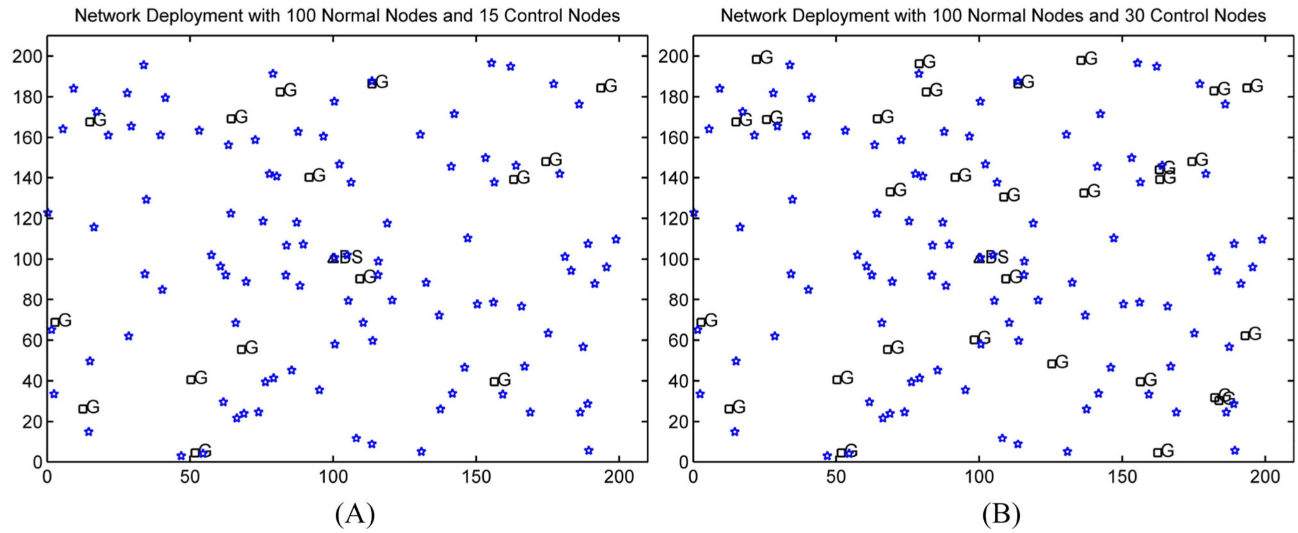


FIGURE 3 Simulation interface for network operation. (a) Network with 15 control nodes. (b) Network with 30 control nodes

TABLE 1 Simulation parameters

Parameter	Parameter's Value
Area of sensing field	200×200 m^2
Location of the base station	(100 m, 100 m)
Node deployment strategy	Random
Number of normal nodes	{100,200,300,400}
Number of deployed control nodes (gateways)	{15,20,25,30}
Normal node's initial energy	2 J
Control node's initial energy	10 J
Data packet size	4000 bits
Packet header size	200 bits
Energy consumption in data aggregation (ε_{da})	5 nJ/bits/signal
Energy consumption in the transceivers' circuitry (E_{elec})	50 nJ/bit
Free space amplification factor (ε_{fs})	10 pJ/bit/m^2
Amplification factor-multipath fading model (ε_{mp})	0.0013 pJ/bit/m^4
Mutation factor (F)	0.5
Crossover rate ($Cross_r$)	0.7

5.2 | Simulation parameters

The set of simulation parameters adopted for the performance comparison of the proposed scheme, DEICA with DECA and Scheme-[30] is described in Table 1.

5.3 | Results and discussion

5.3.1 | Formation of load-balanced clusters

This first set of experiments compares the performance of DEICA with that of DECA, Scheme-[30], TDE_1, and TDE_2. In this comparison, TDE_1 and TDE_2 represent the traditional differential versions of DECA and the proposed scheme,

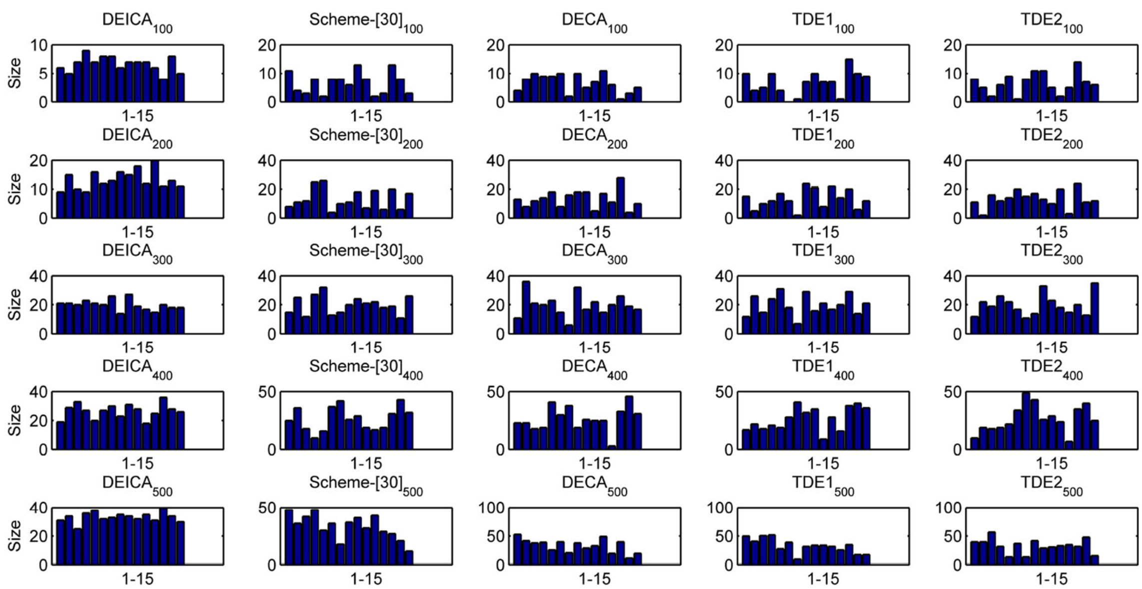


FIGURE 4 Cluster formation with 15 control nodes

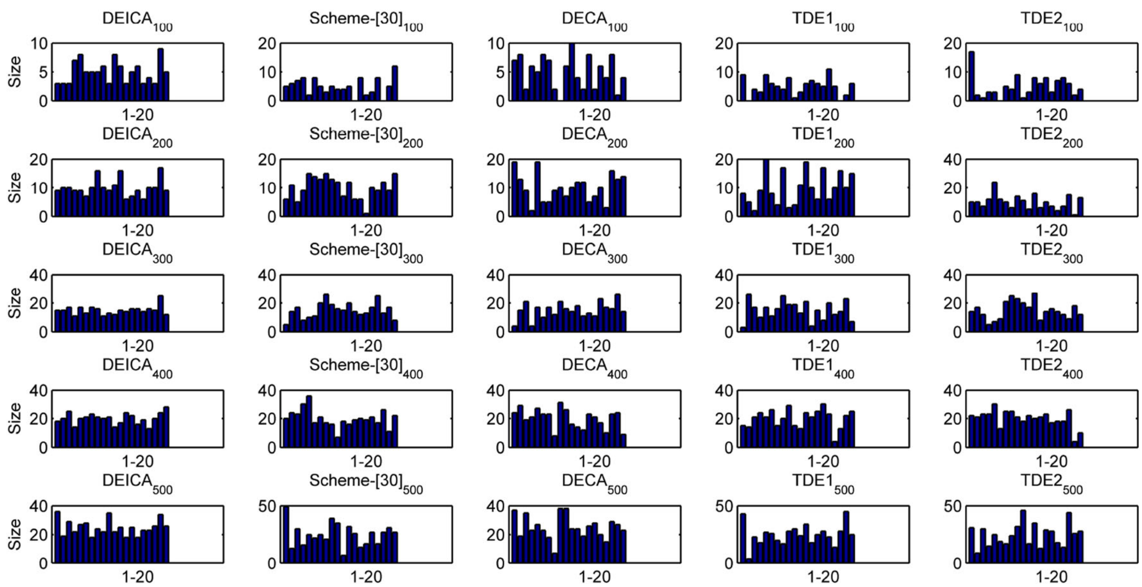


FIGURE 5 Cluster formation with 20 control nodes

DEICA without local improvement phase. The performance comparison based on the quality of formed clusters has been recorded in Figures 4–7. Here, the quality of clusters is measured in terms of their count such that clusters with approximately equal count of normal nodes are considered to be a more quality clusters or load-balanced clusters.

Figures 4–7 depict that the clusters produced as per the proposed scheme ensure comparatively more balanced clusters with respect to DECA, Scheme-[30], TDE_1, and TDE_2. Here, x axis and y axis in the figures refer to the number of control nodes and member normal nodes enrolled in the respective clusters. Performance of the aforesaid schemes have been recorded in the varying network configurations like with different number of normal nodes and control nodes. For example, $DEICA_{100}$,$Scheme$-$[30]_{100}$,$DECA_{100}$,$TDE1_{100}$, and $TDE2_{100}$ with 15 control nodes refer to the schemes—DEICA, Scheme-[30], DECA, TDE_1, and TDE_2 with 100 normal nodes. The similar interpretations can be developed with regard to other notations used in Figures 4–7.

These figures clearly demonstrate the fact that the proposed scheme distributes the nodes more evenly over the formed clusters when compared to other schemes. The observed variation in cluster sizes (that is the differences in the

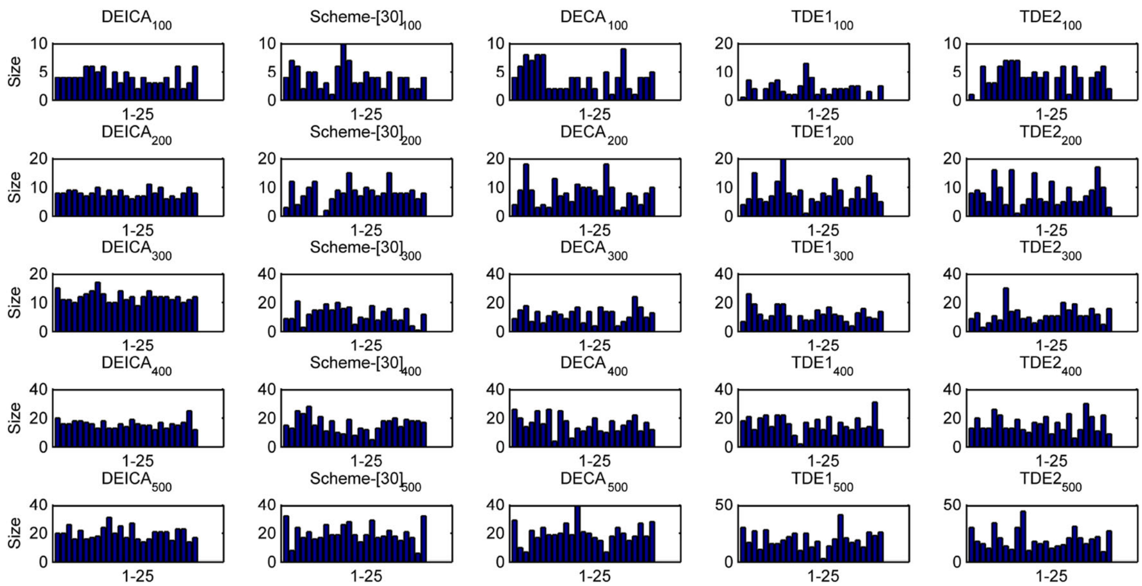


FIGURE 6 Cluster formation with 25 control nodes

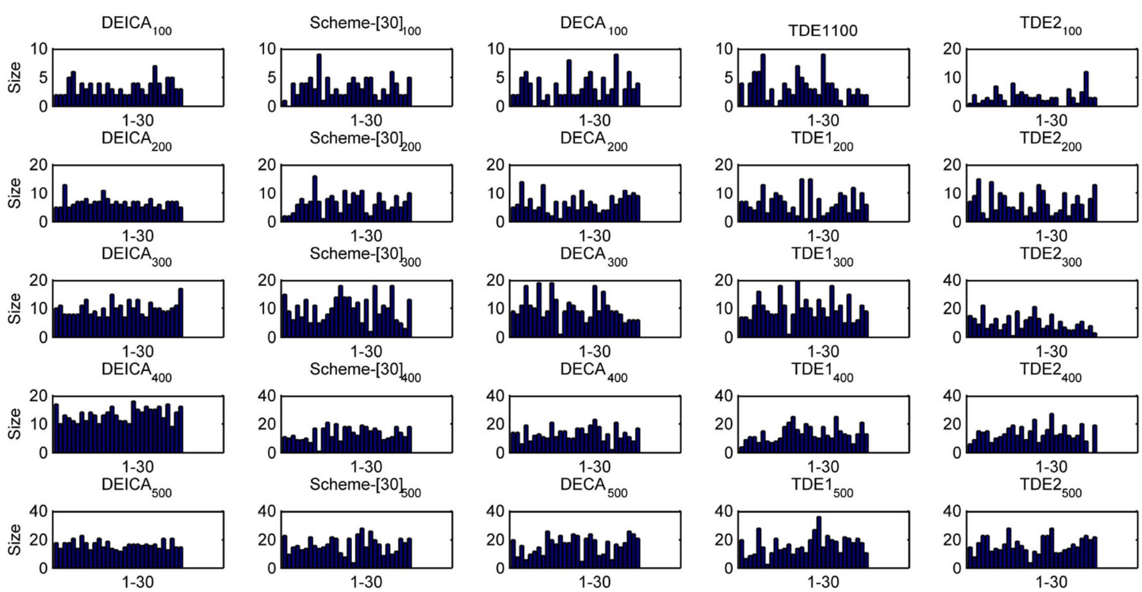


FIGURE 7 Cluster formation with 30 control nodes

number of member nodes in clusters) in DEICA is lower than in DECA, Scheme-[30], TDE_1, and TDE_2 under every network scenarios.

Moreover, the effect of local improvement phase is also apparent from Figures 4–7. It can be easily observed that when TDE_2 was executed solely in the network, the variations in the cluster sizes are comparatively higher than those obtained via DEICA.

Statistical analysis

To further strengthen the claim in favor of DEICA, a statistical analysis is presented in Figure 8. It shows the respective performance of the schemes in terms of standard deviation towards the formulation of load-balanced clusters. Here, standard deviation of the average cluster size refers to how the resultant clusters deviate from the ideal distribution of nodes. Ideal distribution of nodes requires each cluster to contain equal number of normal nodes. The supremacy of the proposed scheme is evident from Figure 8 as DEICA produces clusters with the least standard deviation with respect to others in every possible network configuration. The least standard deviations of the average cluster size clearly indicate that the clusters obtained in DEICA are more balanced in terms of member count and cluster energy. The quantitative values in favor of the aforesaid claim can be observed from Table 2.

Moreover, Table 2 also indicates the success of the proposed scheme in terms of load-balanced cluster formation over the state-of-the-art schemes via confidence interval which justifies the probability of node deployment within specified range of values. For this purpose, the confidence intervals with 95% and 99% confidence intervals are computed, respectively, for four different clustering scenarios with 15, 20, 25, and 30 control nodes with variable node counts-100, 200, 300, 400, and 500. Node counts can be easily computed by dividing the number of nodes by number of control nodes deployed. For example, in the first clustering scenario with 15 control nodes with 100, 200, 300, 400, and 500 nodes, each cluster should contain 6.67, 13.33, 20, 26.67, and 33.33 nodes, respectively, in their ideal cases.

Table 2 confirms the supremacy of the proposed scheme, DEICA, over its peers under varying network configuration. Like, when 100 normal sensors are deployed with 15 control nodes (gateways), DEICA ensures its clusters to contain [6.40, 6.93] and [6.32, 7.01] normal nodes with 95% and 99% confidence levels, respectively, with respect to ideal 6.67 normal nodes per cluster. On the other hand, Scheme-[30] ensures [5.95, 7.39] and [5.73, 7.61] normal nodes per cluster and DECA ensures [6.06, 7.28] and [5.86, 7.48] normal nodes per cluster with 95% and 99% confidence levels, respectively, under the same network configuration. Similarly, the consistency of the proposed scheme, DEICA, can be observed over its peers in terms of load-balanced cluster formation in Table 2.

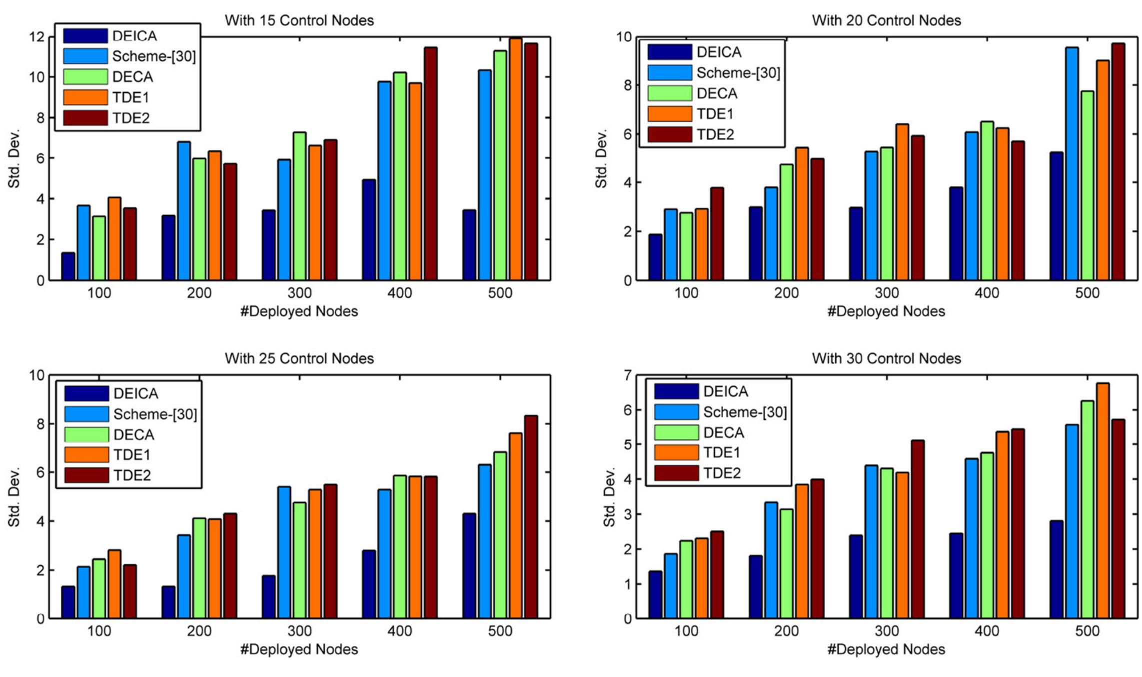


FIGURE 8 Standard deviations in cluster formation

TABLE 2 Standard deviations and confidence intervals table for the generated clusters

		Standard deviation (σ)				Interval estimate with 95% confidence level			
#GW	#Nodes	DEICA	Scheme-[30]	DECA	TDE1	TDE2	DEICA	Scheme-[30]	DECA
15	100	1.3416	3.6606	3.1305	4.0579	3.5308	[6.40 6.93]	[5.95 7.39]	[6.06 7.28]
	200	3.1728	6.8166	5.9722	6.3613	5.7096	[12.89 13.77]	[12.39 14.27]	[12.50 14.16]
	300	3.4254	5.9104	7.2847	6.6332	6.9089	[19.61 20.39]	[19.33 20.67]	[19.18 20.82]
	400	4.9193	9.7673	10.2144	9.6919	11.4397	[26.19 27.15]	[25.71 27.63]	[25.67 27.67]
	500	3.4351	10.3312	11.2813	11.9304	11.6419	[33.03 33.63]	[32.42 34.24]	[32.34 34.32]
20	100	1.8708	2.8983	2.7568	2.9155	3.7815	[6.30 7.03]	[6.10 7.24]	[6.13 7.21]
	200	2.9833	3.7947	4.7329	5.4406	4.9598	[12.92 13.74]	[12.80 13.86]	[12.67 13.99]
	300	2.9665	5.2820	5.4498	6.4031	5.9245	[19.66 20.34]	[19.40 20.60]	[19.38 20.62]
	400	3.7947	6.0745	6.5115	6.2450	5.7009	[26.30 27.04]	[26.07 27.27]	[26.03 27.31]
	500	5.2536	9.5341	7.7460	9.00	9.7005	[32.87 33.79]	[32.49 34.17]	[32.65 34.01]
25	100	1.3266	2.1354	2.4495	2.8142	2.2091	[6.41 6.93]	[6.25 7.09]	[6.19 7.15]
	200	1.3266	3.4293	4.1183	4.0792	4.3081	[13.14 13.51]	[12.85 13.81]	[12.76 13.90]
	300	1.7664	5.3963	4.7582	5.2839	5.4918	[19.80 20.20]	[19.39 20.61]	[19.46 20.54]
	400	2.8000	5.2839	5.8583	5.8241	5.8172	[26.40 26.94]	[26.15 27.19]	[26.10 27.24]
	500	4.3081	6.2992	6.8176	7.6158	8.3331	[32.95 33.71]	[32.79 33.88]	[32.73 33.93]
30	100	1.3663	1.8619	2.2361	2.3094	2.5033	[6.40 6.94]	[6.30 7.03]	[6.23 7.10]
	200	1.8074	3.3367	3.1411	3.8471	3.9917	[13.08 13.58]	[12.87 13.79]	[12.89 13.76]
	300	2.3944	4.3894	4.3050	4.1873	5.1251	[19.73 20.27]	[19.50 20.50]	[19.51 20.49]
	400	2.4495	4.5826	4.7539	5.3728	5.4467	[26.43 26.91]	[26.22 27.12]	[26.20 27.14]
	500	2.8048	5.5737	6.2557	6.7626	5.7213	[33.08 33.58]	[32.84 33.82]	[32.78 33.89]

TABLE 2 (Continued)

	Interval estimate with 95% confidence level	Interval estimate with 99% confidence level					
#GW	TDE1	TDE2	DEICA	Scheme-[30]	DECA	TDE1	TDE2
15	[5.87 7.47]	[5.98 7.36]	[6.32 7.01]	[5.73 7.61]	[5.86 7.48]	[5.62 7.71]	[5.76 7.58]
	[12.45 14.21]	[12.54 14.12]	[12.75 13.91]	[12.09 14.57]	[12.24 14.42]	[12.17 14.49]	[12.29 14.37]
	[19.25 20.75]	[19.22 20.79]	[19.49 20.51]	[19.12 20.88]	[18.91 21.08]	[19.01 20.99]	[18.97 21.03]
	[25.72 27.62]	[25.55 27.79]	[26.04 27.30]	[25.41 27.93]	[25.35 27.99]	[25.42 27.92]	[25.19 28.15]
	[32.28 34.38]	[32.31 34.35]	[32.93 33.73]	[32.14 34.52]	[32.03 34.63]	[31.95 34.71]	[31.99 34.67]

TABLE 2 (Continued)

#GW	Interval estimate with 95% confidence level	Interval estimate with 99% confidence level					
	TDE1	TDE2	DEICA	Scheme-[30]	DECA	TDE1	TDE2
20	[6.10 7.24]	[5.93 7.41]	[6.18 7.15]	[5.92 7.42]	[5.96 7.38]	[5.92 7.42]	[5.69 7.65]
	[12.58 14.08]	[12.64 14.02]	[12.79 13.87]	[12.64 14.02]	[12.47 14.19]	[12.33 14.32]	[12.43 14.24]
	[19.28 20.72]	[19.33 20.67]	[19.56 20.44]	[19.21 20.79]	[19.19 20.81]	[19.05 20.95]	[19.12 20.88]
	[26.06 27.28]	[26.11 27.23]	[26.18 27.16]	[25.89 27.45]	[25.83 27.51]	[25.86 27.48]	[25.93 27.41]
	[32.54 34.12]	[32.48 34.18]	[32.72 33.93]	[32.23 34.43]	[32.44 34.22]	[32.29 34.37]	[32.21 34.45]
25	[6.12 7.22]	[6.24 7.10]	[6.33 7.01]	[6.12 7.22]	[6.04 7.30]	[5.94 7.40]	[6.10 7.24]
	[12.77 13.90]	[12.73 13.93]	[13.09 13.57]	[12.70 13.96]	[12.58 14.08]	[12.59 14.07]	[12.54 14.11]
	[19.40 20.54]	[19.38 20.62]	[19.74 20.26]	[19.20 20.80]	[19.29 20.71]	[19.21 20.79]	[19.19 20.82]
	[26.10 27.24]	[26.10 27.24]	[26.31 27.03]	[25.99 27.35]	[25.91 27.43]	[25.92 27.42]	[25.92 27.42]
	[32.66 33.40]	[32.60 34.06]	[32.83 33.83]	[32.60 34.05]	[32.54 34.12]	[32.45 34.21]	[32.37 34.29]
30	[6.22 7.12]	[6.18 7.16]	[6.31 7.02]	[6.19 7.15]	[6.09 7.25]	[6.07 7.27]	[6.02 7.32]
	[12.80 13.86]	[12.78 13.88]	[13.00 13.66]	[12.72 13.94]	[12.76 13.90]	[12.63 14.03]	[12.60 14.06]
	[19.53 20.47]	[19.42 20.58]	[19.64 20.36]	[19.35 20.65]	[19.36 20.64]	[19.37 20.62]	[19.24 20.76]
	[26.14 27.19]	[26.14 27.20]	[26.35 26.99]	[26.08 27.26]	[26.06 27.28]	[25.98 27.36]	[25.97 27.37]
	[32.74 33.92]	[32.83 33.83]	[33.00 33.65]	[32.67 33.97]	[32.61 34.05]	[32.55 34.11]	[32.67 33.99]

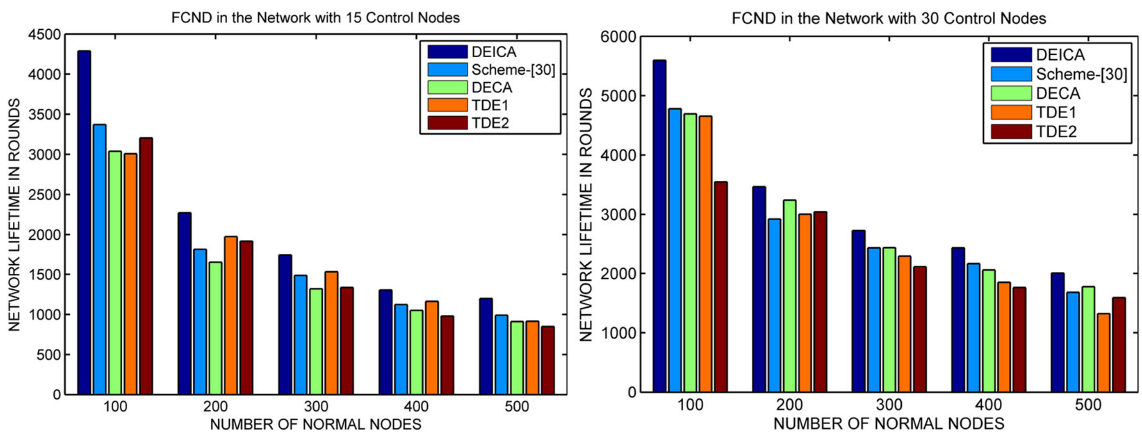
5.3.2 | Network performance

In this next set of experiments, performance of the proposed scheme, DEICA has been compared with that of Scheme-[30], DECA, TDE_1, and TDE_2 in terms of network lifetime, nodes' death rate, energy consumption rate, node's average residual energy, and packet delivery, respectively, in Figures 9–12. Performance of the aforesaid schemes has been recorded under varying node distributions for two exemplary network scenarios-one with the 15 control nodes and another with 30 control nodes, respectively.

Network Lifetime

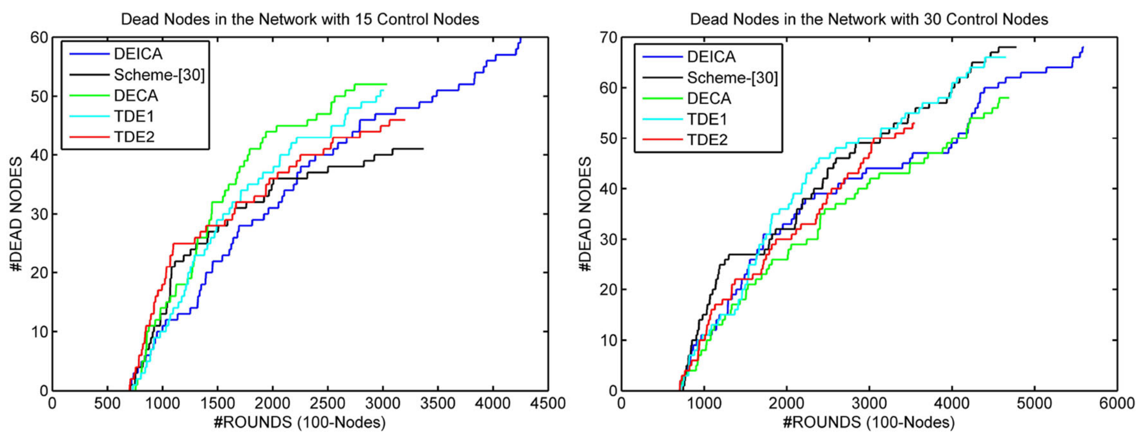
The definition of the network lifetime is adopted from previous studies[9–11,30] as the time when the first control node (or equivalently, the CH) dies in the network. It is evident from Figure 9 that DEICA outperforms the rest of the schemes under varying network configurations with respect to network lifetime.

In a network with 15 control nodes and 100, 200, 300, 400, and 500 normal nodes, the first control node death (FCND) occurs at 4289, 2270, 1750, 1304, and 1199 rounds, respectively, in DEICA. For the above-mentioned network configuration, FCND occurs at 3369, 1816, 1485, 1122, and 989 rounds in Scheme-[30]; at 3037, 1659, 1320, 1050, and



(A) (B)

FIGURE 9 Network lifetime measured as first control node death. (A) First control node death in N/W with 15 control nodes. (B) First control node death in N/W with 30 control nodes



(A) (B)

FIGURE 10 Network stability in terms of normal node's death rate. (A) Node's death rate in N/W with 15 control nodes. (B) Node's death rate in N/W with 30 control nodes

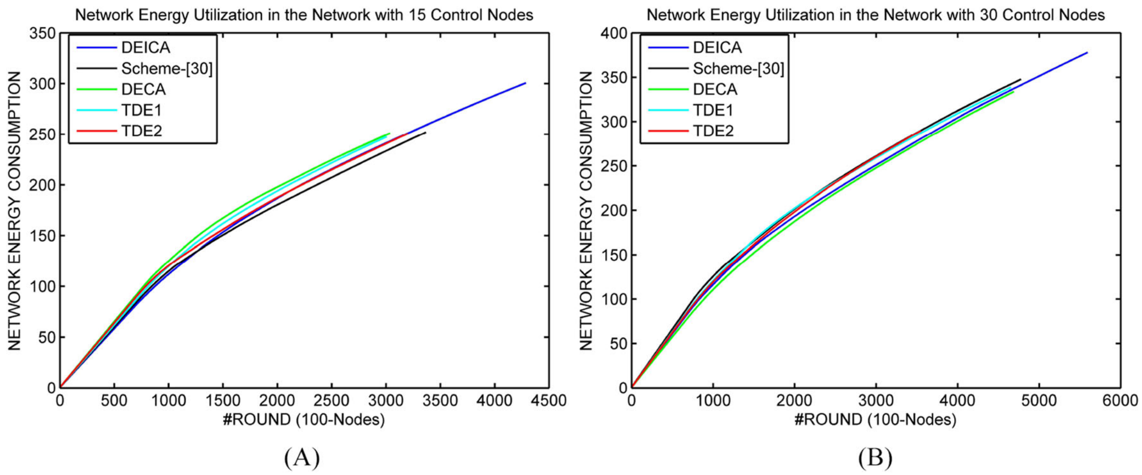


FIGURE 11 Network energy resource utilization over the network rounds. (A) Energy resource utilization in N/W with 15 control nodes. (B) Energy resource utilization in N/W with 30 control nodes

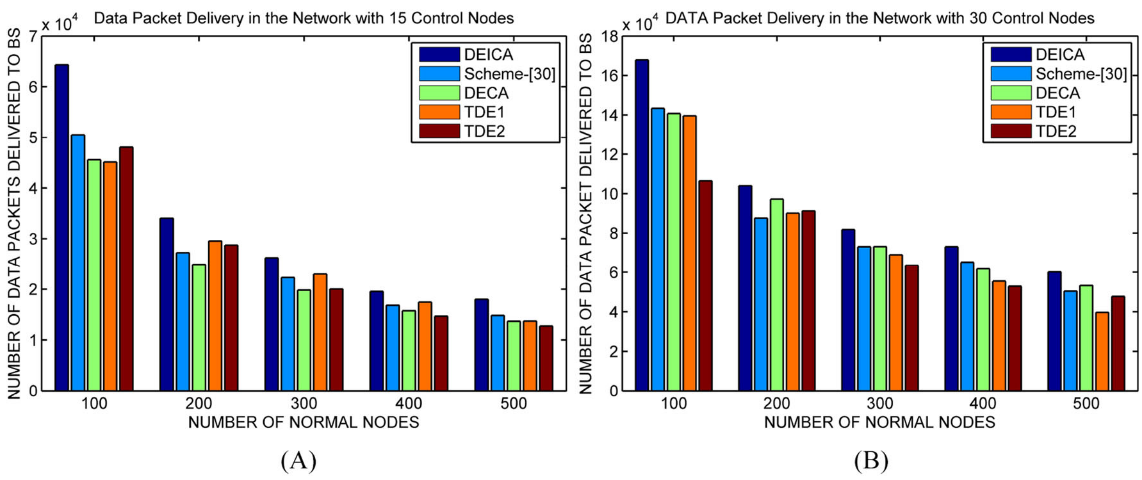


FIGURE 12 Data packet delivery at the base station. (A) Data packet delivery in N/W with 15 control nodes. (B) data packet delivery in N/W with 30 control nodes

TABLE 3 Percentage gain in network lifetime

		Percentage Gain in Network Lifetime			
No. of GWs	**No. of nodes**	**Scheme-[30]**	**DECA**	TDE_1	TDE_2
15	100	27.27%	41.42%	42.55%	33.92%
	200	24.94%	36.77%	15.06%	18.42%
	300	17.78%	32.5%	14.24%	30.91%
	400	16.22%	24.19%	12.12%	33.33%
	500	21.23%	31.61%	31.04%	40.21%
30	100	17.10%	19.30%	20.25%	57.69%
	200	18.69%	6.97%	15.45%	13.97%
	300	12.03%	11.89%	18.66%	28.62%
	400	11.95%	18.32%	31.62%	38.04%
	500	19.26%	12.89%	51.85%	25.93%

911 rounds in DECA; at 3008, 1972, 1531, 1163, and 915 rounds in TDE_1; and at 3202, 1916, 1336, 978, and 848 rounds in TDE_2.

Similarly, in a network with 30 control nodes and with 100, 200, 300, 400, and 500 normal nodes, the FCND occurs at 5595, 3468, 2728, 2435, and 2006 rounds, respectively, in DEICA. For the above-mentioned network configuration, FCND occurs at 4778, 2921, 2435, 2175, and 1682 rounds in Scheme-[30]; at 4690, 3241, 2438, 2058, and 1777 rounds in DECA; at 4653, 3003, 2299, 1850, and 1321 rounds in TDE_1; and at 3548, 3042, 2121, 1764, and 1593 rounds in TDE_2.

Table 3 summarizes the above-obtained results while describing the percentage gain in network lifetime for the proposed scheme, DEICA over Scheme-[30], DECA, TDE_1, and TDE_2, respectively. It can be easily observed that under varying network configuration, DEICA assures up to 57.69% gain in terms of network lifetime over the other competing schemes.

In addition to the FCND taken as the criterion to measure the network lifetime, normal nodes' death rate in the deployed IoT-based HWSN can also be considered an important parameter for evaluating the performance performance. Figure 10 describes the same for the proposed scheme, DEICA with respect to that in Scheme-[30], DECA, TDE_1, and TDE_2. In their respective network lifetimes, DEICA, Scheme-[30], DECA, TDE_1, and TDE_2 consume 60, 41, 52, 51, and 46 normal nodes in the first network scenario (with 100 nodes and 15 control nodes) and 68, 68, 58, 66, and 53 normal nodes in the second network scenario (with 100 nodes and 30 control nodes). Here, the lesser number of dead nodes in the schemes other than DEICA indicate that the normal nodes are uncovered due to early death of their respective control nodes.

Network energy utilization

In the above-prescribed two-tier network where the energy-intensive tasks are handled by the energy-enriched control nodes and normal nodes are kept reserved for environment sensing and monitoring, it is more desirable to keep the network operating for a longer time. In other words, normal nodes must keep on sensing and monitoring activities for a longer time and hence available network energy resource must be utilized for a longer duration for the fulfillment of the intended objectives. For example, from Figure 11A, considering a network scenario with 100 normal nodes and 15 control nodes, it can be easily observed that DEICA utilizes 300.69 J (out of 350 J) network energy with a percent energy utilization of 85.71%. This is because DEICA is able to engage most of the normal nodes for network operation. Contrary to this, schemes—Scheme-[30], DECA, TDE_1, and TDE_2—have percent energy utilization of 72.00%, 71.70%, 70.69%, and 71.61%, respectively. Similarly, Figure 11B depicts that DEICA utilizes 377.89 J (out of 500 J) network energy with percent energy utilization of 75.57% in comparison to 69.56%, 66.77%, 67.73%, and 57.90% percent energy utilization due to Scheme-[30], DECA, TDE_1, and TDE_2, respectively. This establishes the supremacy of DEICA in terms of network energy utilization over the other competing schemes.

Table 4 further describes the supremacy of DEICA by detailing the percentage gain in network energy utilization over the schemes—Scheme-[30], DECA, TDE_1, and TDE_2. From Table 4, it can be observed that DEICA outperforms

TABLE 4 Percentage gain in network energy utilization

		Percentage gain in network energy utilization			
No. of GWs	**No. of nodes**	**Scheme-[30]**	**DECA**	TDE_1	TDE_2
15	100	19.31%	19.80%	21.52%	19.98%
	200	26.40%	29.73%	19.39%	18.67%
	300	9.59%	24.90%	8.76%	23.85%
	400	13.23%	20.40%	9.71%	33.16%
	500	9.66%	18.75%	21.18%	28.29%
30	100	8.65%	13.19%	11.57%	30.52%
	200	10.01%	1.45%	8.76%	7.72%
	300	5.52%	2.63%	9.95%	15.91%
	400	11.52%	15.40%	22.41%	26.51%
	500	7.89%	5.34%	30.15%	11.34%

TABLE 5 Percentage gain in data packet delivery at base station

No. of GWs	No. of nodes	Percentage gain in data packet delivery at base station			
		Scheme-[30]	DECA	TDE_1	TDE_2
15	100	27.32%	41.24%	42.60%	33.96%
	200	25.01%	36.85%	15.12%	18.49%
	300	17.86%	32.60%	14.31%	31.01%
	400	16.32%	24.30%	12.22%	33.46%
	500	21.36%	31.76%	31.81%	41.56%
30	100	17.12%	14.44%	20.27%	57.74%
	200	18.73%	7.00%	15.49%	14.00%
	300	12.78%	11.94%	18.72%	28.68%
	400	12.00%	18.37%	33.37%	14.86%
	500	19.33%	12.95%	51.97%	26.00%

other competing schemes in terms of network energy utilization by assuring up to 33.16% more utilization under variable network configurations.

Packet delivery at BS

In this last set of experimentation, the performance of the proposed scheme, DEICA has been compared with that of Scheme-[30], DECA, TDE_1, and TDE_2 with respect to total number of data packet delivered to the BS. Figure 12 describes the comparative performance of DEICA over the other schemes.

DEICA ensures delivery of 64,320, 34,035, 26,235, 19,560, and 17,985 data packets to the BS in a network with 15 control nodes and 100, 200, 300, 400, and 500 nodes, respectively. For the above-mentioned network configuration, the total number of data packets delivered to the BS are 50,520, 27,225, 22,260, 16,815, and 14,820 in Scheme-[30]; 45,540, 24,870, 19,785, 15,735, and 13,650 in DECA; 45,105, 29,565, 22,950, 17,430, and 13,710 in TDE_1; and 48,015, 28,725, 20,025, 14,655, and 12,705 in TDE_2.

Similarly, for the network with 30 control nodes and 100, 200, 300, 400, and 500 nodes, the number of data packets delivered to the BS are 167,850, 104,010, 81,840, 73,050, and 60,180 in DEICA; 143,310, 87,600, 73,020, 65,220, and 50,430 in Scheme-[30]; 146,670, 97,200, 73,110, 61,710, and 53,280 in DECA; 139,560, 90,060, 68,940, 54,770, and 39,600 in TDE_1; and 106,410, 91,230, 63,600, 52,890, and 47,760 in TDE_2.

The above-mentioned results are summarized in Table 5 that describes the percentage gain in data packet delivery at BS for DEICA over the schemes—Scheme-[30], DECA, TDE_1, and TDE_2. It is self-evident from Table 5 that DEICA enables up to 57.74% more data packets to be delivered at the BS under varying network configuration.

Hence, from the above experimentation, it can be concluded that DEICA outperforms the schemes—Scheme-[30], DECA, TDE_1, and TDE_2 in terms of quality cluster formation, network longevity, network resource utilization, and data packet delivery.

6 | CONCLUSION AND FUTURE WORKS

In the present work, a DEICA is proposed for the IoT-based two-level heterogeneous WSNs. DEICA utilizes the straightforward and fast converging DE scheme via the BS to distribute the nodes evenly among the clusters led by the specialized energy-enriched control nodes. Various parameters have been considered for the formation of load-balanced clusters, such as the lifetime of the control nodes, intracluster communication cost for the normal sensor nodes, and cluster density. In addition to the algorithmic steps of DE with a specially devised fitness evaluation function, the proposed scheme also applies a local improvement phase to improve the formulated clusters further. The supremacy of DEICA is established through an extensive set of experimentations over the existing state-of-the-art schemes—Scheme-[30], DECA, and traditional DE schemes with respect to clusters' quality, network lifetime improvement, network resource utilization, and data packet delivery. It has been demonstrated that DEICA achieves considerable efficacy in terms of gains in network lifetime, network energy utilization, and data packet delivery, respectively, over its peer schemes under variable network configurations.

As a future version of this work, IoT-based HWSN with more levels of energy-heterogeneity might be investigated, which is getting wide popularity due to the networks involving nodes with varying features and functionality.

ACKNOWLEDGMENT

This work is partially supported by SERB under ASEAN-India Collaborative Research Project (CRD/20201000284) entitled “Environmental Monitoring in Underground Mines using Sensor-enabled Internet of Things.”

CONFLICT OF INTEREST

The authors declare that they have no competing interests.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Rajib Banerjee https://orcid.org/0000-0003-0685-5977

REFERENCES

1. Akyildiz IF, Weilian S, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. *IEEE Commun Mag*. 2002;40(8):393-422. https://doi.org/10.1109/MCOM.2002.1024422
2. Dumka A, Chaurasiya SK, Biswas A, Mandoria HL. *A Complete Guide to Wireless Sensor Networks: From Inception to Current Trends*. 1st ed.: CRC Press; 2019. ISBN 9781032401416.
3. Shahraki A, Taherkordi A, Haugen Ø, Eliassen F. A survey and future directions on clustering: from WSNs to IoT and modern networking paradigms. *IEEE Trans Netw Serv Manag*. 2021;18(2):2242-2274. https://doi.org/10.1109/TNSM.2020.3035315
4. Chaurasiya S, Pal T, Bit SD. An enhanced energy-efficient protocol with static clustering for WSN. In: Proceedings IEEE Xplore, Int'l Conf. of Information Networking (ICOIN), Kuala Lumpur, Malaysia. IEEE; 2011:58-63.
5. Chaurasiya S, Sen J, Chatterjee S, Bit SD. EBLEC: an energy-balanced lifetime enhancing clustering for WSN. In: Proceeding IEEE Xplore, 14th Int'l Conf. on Advanced Communication & Technology-2012, PyeongChang, Korea (South). IEEE; 2012:189-194. INSPEC Accession Number: 12656578.
6. Chaurasiya SK, Mondal J, Datta S. Field-of-view based hierarchical clustering to prolong network lifetime of WMSN with obstacles. In: Proceeding IEEE Xplore, Int'l Conf. on Electronics, Communication, and Computational Engineering, Chennai, INDIA; Nov. 2014:72-77. 10.1109/ICECCE.2014.7086638.
7. Chaurasiya SK, Biswas A, Banerjee R. Metaheuristic multilevel heterogeneous clustering technique for heterogeneous wireless sensor networks. In: 2021 10th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON). IEEE; 2021:1-6.
8. Chaurasiya SK, Biswas A, Bandyopadhyay PK, Banerjee A, Banerjee R. Metaheuristic load-balancing based clustering technique for wireless sensor networks. *Wirel Commun Mob Comput*. 2022;2022:8911651. https://doi.org/10.1155/2022/8911651
9. Bari A, Wazed S, Jaekel A, Bandyopadhyay S. A genetic algorithm based approach for energy efficient routing in two-tiered sensor networks. *Ad Hoc Netw*. 2009;7(4):665-676.
10. Chakraborty UK, Das SK, Abbott TE. Energy-efficient routing in hierarchical wireless sensor networks using differential-evolution-based memetic algorithm. *2012 IEEE Congress on Evolutionary Computation (CEC)*: IEEE; 2012:1-8.
11. Kuila P, Jana PK. A novel differential evolution based clustering algorithm for wireless sensor networks. *Appl Soft Comput*. 2014;25:414-425.
12. Vesterstrom J, Thomsen R. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *Proc Congr Evol Comput*. 2004;2:1980-1987.
13. Bounceur A, Bezoui M, Lounis M, Euler R, Teodorov C. A new dominating tree routing algorithm for efficient leader election in IoT networks. In: 15th IEEE Annual Consumer Communications & Networking Conference (CCNC). IEEE; 2018.
14. Kumar N, Vidyarthi DP. A green routing algorithm for IoT-enabled software defined wireless sensor network. *IEEE Sens J*. 2018;18(22):9449-9460.
15. Alshehri MD, Hussain FK, Hussain OK. Clustering-driven intelligent trust management methodology for the Internet of Things (CITM-IoT). *Mob Netw Appl*. 2018;23(3):419-431.
16. Souissi I, Azzouna NB, Said LB. A multi-level study of information trust models in WSN-assisted IoT. *Comput Netw*. 2019;151:12-30. https://doi.org/10.1016/j.comnet.2019.01.010
17. Yarinezhad R, Sabaei M. An optimal cluster-based routing algorithm for lifetime maximization of Internet of Things. *J Parallel Distrib Comput*. 2021;156:7-24. https://doi.org/10.1016/j.jpdc.2021.05.005

18. Nandan AS, Singh S, Malik A, Kumar R. A green data collection & transmission method for IoT-based WSN in disaster management. *IEEE Sens J*. 2021;21(22):25912-25921. https://doi.org/10.1109/JSEN.2021.3117995
19. Osamy W, Salim A, Khedr AM, El-Sawy AA. IDCT: intelligent data collection technique for IoT-enabled heterogeneous wireless sensor networks in smart environments. *IEEE Sens J*. 2021;21(18):21099-21112. https://doi.org/10.1109/JSEN.2021.3100339
20. Li X, Xu L, Wang H, Song J, Yang SX. A differential evolution based routing algorithm for envirnomental monitoring wireless sensor networks. *Sensors*. 2010;10(6):5425-5442.
21. Heinzelman WR, Chandrakasan A, Balkrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of 33rd Hawaii International Conference on System Science, Vol. 2. IEEE; 2000:1-10.
22. Heinzelman WR, Chandrakasan A, Balkrishnan H. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans Wireless Communication*. 2002;1(4):660-670. https://doi.org/10.1109/TWC.2002.804190
23. Potthuri S, Shankar T, Rajesh A. Lifetime improvement in wireless sensor networks using HYbrid differential evolution and simulated anneailing (DESA). *Ain Shams Eng J*; 2018(9):655-663.
24. Brest J, Greiner S, Boskovic B, Mernik M, Zumer V. Self-adapting control parameters in differential evoultion: a comparative study on numerical benchmark problems. *IEEE Trans Evol Comput*. 2006;10(6):646-657.
25. Randhawa S, Jain S. MLBC: multi-objective load balancing clustering technique in wireless sensor networks. *Appl Soft Comput*. 2019;74:66-89. https://doi.org/10.1016/j.asoc.2018.10.002
26. Gupta GP, Saha B. Load balanced clustering scheme using hybrid metaheuristic technique for mobile sink based wireless sensor networks. *J Ambient Intell Human Comput*. 2020. https://doi.org/10.1007/s12652-020-01909-z
27. Iwendi C, Maddikunta PKR, Gadekallu TR, Lakshmanna K, Bashir AK, Piran MJ. A metaheuristic optimization approach for energy efficiency in the IoT networks. *Softw: Pract Exper*. 2021;51(12):2558-2571. https://doi.org/10.1002/spe.2797
28. Ponnan S, Saravanan AK, Iwendi C, Ibeke E, Srivastava G. An artificial intelligence-based quorum system for the improvement of the lifespan of sensor networks. *IEEE Sens J*. 2021;21(15):17373-17385. https://doi.org/10.1109/JSEN.2021.3080217
29. Sackey SH, Ansere JA, Anajemba JH, Kamal M, Iwendi C. Energy efficient clustering based routing technique in WSN using brain storm optimization. In: 2019 15th International Conference on Emerging Technologies (ICET). IEEE; 2019:1-6.
30. Ghahramani M, Laakdashti A. Efficient energy consumption in wireless sensor networks using an improved differential evolution algorithm. In: 10th International Conference on Computer and Knowledge Engineering (ICCKE). IEEE; 2020:018-023.
31. Santosh T, Chaurasiya SK. Computational intelligence in wireless sensor networks: a thorough review. In: Chaurasiya SK, Dutta J, Biswas A, Dutta G, Sarkar MK, eds. *Computational Intelligence for Wireless Sensor Networks—Principles and Applications*. 1st ed.: CRC Press; 2022.
32. Chaurasiya SK, Ghosh S, Banerjee R. Metaheuristic clustering in wireless sensor network. In: Chaurasiya SK, Dutta J, Biswas A, Dutta G, Sarkar MK, eds. *Computational Intelligence for Wireless Sensor Networks—Principles and Applications*. 1st ed.: CRC Press; 2022.
33. Xiang W, Wang N, Zhou Y. An energy-efficient routing algorithm for software-defined wireless sensor networks. *IEEE Sens J*. 2016; 16(20):7393-7400.
34. Ramteke R, Singh S, Malik A. Optimized routing technique for IoT enabled software-defined heterogeneous WSNs using genetic mutation based PSO. *Comput Stand Interfaces*. 2022;79:103548. https://doi.org/10.1016/j.csi.2021.103548
35. Banerjee R, Chaterjee S, Bit SD. An energy saving audio compression scheme for wireless multimedia sensor networks using spatio-temporal partial discrete wavelet transform. *Comput Electr Eng*. 2015;48:389-404. https://doi.org/10.1016/j.compeleceng.2015.09.009
36. Banerjee R, Bit SD. Low-overhead video compression combining partial discrete cosine transform and compressed sensing in WMSNs. *Wirel Netw*. 2019;25:5113-5135. https://doi.org/10.1007/s11276-019-02119-y
37. Banerjee R, Bit SD. An energy efficient image compression scheme for wireless multimedia sensor network using curve fitting technique. *Wirel Netw*. 2019;25:167-183. https://doi.org/10.1007/s11276-017-1543-9
38. Kaddi M, Khalili Z, Bouchra M. A differential evolution based clustering and routing protocol for WSN. In: 2020 International Conference on Mathematics and Information Technology, Adrar, Ageria, Feb. 18–19. IEEE; 2020:190-195.
39. Chaurasiya SK, Biswas A, Bandyopadhyay PK. Heterogeneous energy-efficient clustering protocol for wireless sensor networks. In: Mishra B, Tiwari M, eds. *VLSI, Microwave and Wireless Technologies*, Lecture Notes in Electrical Engineering, vol. 877: Springer; 2023, https://doi.org/10.1007/978-981-19-0312-0_16

How to cite this article: Chaurasiya SK, Biswas A, Nayyar A, Zaman Jhanjhi N, Banerjee R. DEICA: A differential evolution-based improved clustering algorithm for IoT-based heterogeneous wireless sensor networks. *Int J Commun Syst*. 2023;e5420. doi:10.1002/dac.5420