

Research Article

Clustering Approaches for Pragmatic Two-Layer IoT Architecture

J. Sathish Kumar  and **Mukesh A. Zaveri** 

Computer Engineering Department, SVNIT, Surat, India

Correspondence should be addressed to J. Sathish Kumar; sathish613@gmail.com

Received 13 September 2017; Revised 8 February 2018; Accepted 11 March 2018; Published 19 April 2018

Academic Editor: Mauro Femminella

Copyright © 2018 J. Sathish Kumar and Mukesh A. Zaveri. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Connecting all devices through Internet is now practical due to Internet of Things. IoT assures numerous applications in everyday life of common people, government bodies, business, and society as a whole. Collaboration among the devices in IoT to bring various applications in the real world is a challenging task. In this context, we introduce an application-based two-layer architectural framework for IoT which consists of sensing layer and IoT layer. For any real-time application, sensing devices play an important role. Both these layers are required for accomplishing IoT-based applications. The success of any IoT-based application relies on efficient communication and utilization of the devices and data acquired by the devices at both layers. The grouping of these devices helps to achieve the same, which leads to formation of cluster of devices at various levels. The clustering helps not only in collaboration but also in prolonging overall network lifetime. In this paper, we propose two clustering algorithms based on heuristic and graph, respectively. The proposed clustering approaches are evaluated on IoT platform using standard parameters and compared with different approaches reported in literature.

1. Introduction

The success of wireless sensor network in the form of technology and applications in different areas like home automation, industrial applications, security and military surveillance, and many more raises the further need for machine-to-machine connectivity and availability of the data or information anytime and anywhere [1]. This requirement leads to the new technology development in the form of Internet of Things (IoT). The IoT allows the connectivity among the devices and helps in acquisition of data or information at any time and from any place. There are numerous applications of IoT coming up using different technologies [2]. Also, IoT enables innovative services in numerous applications like smart transportation, smart home, smart city, smart lifestyle, smart retail, smart agriculture, smart industries, smart emergency, smart health care, smart environment, and many more [3, 4]. The use of these applications and their demand has increased the scope of research and innovation in this domain [2, 5].

The most significant part of above-mentioned applications using IoT needs sensing and monitoring the environment or acquiring the data from different IP-enabled devices or sensors. The sensor devices used for sensing and monitoring are battery-operated and energy-constrained. This implies that power consumption and energy are critical aspects. IP-based communication effectively utilizes more energy, which leads these low-powered devices to deplete rapidly. These huge numbers of devices communicate and collaborate with each other in order to accomplish a given job or task. This raises the need for effective connectivity and efficient communication among these devices in an optimized way, which is a very challenging task. In this context, there is a need for a solution that promises maximal connectivity through minimum communication. The most efficient way to fulfill these needs is to collaborate among the devices or sensors and perform the tasks for a given application. One way to achieve this is through grouping the devices in an efficient way in terms of energy usage and computational complexity.

For any IoT-based application, it is required to collect the data through sensing devices and process these data through different algorithms. Then, the processed information can be accessed through the Internet anywhere and at any time. The grouping of devices or sensors is known as clustering. Clustering helps in carrying out the task of acquiring the information in an efficient way with minimum number of communications within a network [6] and disseminating this information for further processing. Clustering also helps in prolonging the network lifetime and further the lifetime of an IoT-based application that is deployed for a specific task.

For efficient clustering in the IoT environment, there is a need to have (i) classification between the underlying sensor devices and IP-enabled devices and (ii) minimum communication overhead for accessing the data. To achieve this, we introduce two-layer architectural framework consisting of two layers: upper layer that consists of IP-enabled IoT devices and lower layer that is simply sensor devices. In this regard, for grouping the devices, clustering algorithms are employed. In clustering, each cluster chooses one node within the cluster as a cluster head, and further the communication among the nodes in a network is carried out through the cluster head. If two nodes in a network are not within the range of each other, multihop communication is required. Communication through cluster head avoids multihop communication up to some extent. The cluster head also helps in aggregating the data acquired by different nodes in a network. In IoT-based network, as mentioned, two-layer architecture consists of sensor layer and IoT layer. The grouping of these devices is possible in two ways, that is, from sensor layer to IoT layer and vice versa. For accommodating both these approaches, we propose two clustering algorithms, namely, heuristic-based and graph-based clustering, in this paper, respectively.

The rest of the paper is organized as follows. The literature survey with respect to related work regarding the different clustering techniques is described in Section 2. Section 3 introduces a layered architectural framework for IoT. This architectural framework is a system model for our proposed clustering algorithms. Section 4 describes the proposed clustering approaches. The simulation results and experiments are detailed in Section 5. Section 6 concludes the paper.

2. Related Work

In the literature, efforts have been noticed to propose the IoT architecture [7, 8] in the form of generic architecture or as reference model. These architectures have been introduced at conceptual level without any implementation and validation. The generic architecture or reference model is too complex for one-to-one mapping for any real application. Earlier, IoT architecture [9, 10] has been presented in an abstract way and evaluated to measure the performance. In practice, there is a need to integrate ordinary sensors or IoT devices with an application. In this context, a system model using two-layer IoT framework is introduced and the communication cost in this model may be reduced to one or at the most two hops for any node in a network. This two-layer framework is described later in the next section.

As pointed out earlier, the nodes are tiny and energy-constrained. In this context, it is desirable to form a group of devices and the group of devices communicate among themselves through a group leader that ensures the minimum communication overhead and energy-efficient execution using clustering. In the literature, clustering in wireless sensor network (WSN) has been explored very well [11–25]. In our paper, we also address the clustering of sensor nodes but in the Internet of Things environment. There are very important facts to be noted which differentiate the clustering in WSN compared to that in IoT. The clustering in WSN is performed locally in the network, whereas clustering in IoT may be considered as globally in the network. The number of parameters restricted to WSN may be connectivity and density of the nodes, which are either layer-based [26, 27] or without layer and hence very much constrained by the mobility of the nodes. But in the case of IoT environment, the density is not a constraint. Moreover, the mobility of IoT nodes allows on the fly clustering with underlying sensor nodes dynamically in the region of interest, which is not possible in case of WSN. Further, in the literature, there is no method reported for clustering in IoT environment so far, and, for the first time, it is explored using the two-layer architecture in IoT as introduced in our earlier proposed work [9, 10].

In [26], cluster-based architecture has been proposed to structure the topology of different wireless sensor networks to coexist in the same environment by creation of virtual wireless sensor networks. Similarly, layer-based clustering approach has been reported in [27] for routing with homogeneous and densely deployed network using cross-layer interaction. The approach in [26] does not support the discovery of nodes on the fly which is mandatory for the network with mobility to enable anytime and anywhere data accessibility, which can be supported by IoT-based network. In our proposed method, there is no restriction of dense network deployment compared to the method in [27], and, moreover, our proposed approach also supports clustering in the case of heterogeneous environment.

The literature related to different clustering approaches in the wireless domain is referred to here, which motivates us to propose a clustering method for a layered IoT framework. Various clustering algorithms are available in the literature and are proposed by considering the distributed mechanisms with respect to WSN. In low-energy adaptive clustering hierarchy (LEACH) [11], two-level-LEACH [12] and energy-efficient hierarchical clustering (EEHC) [13], residual energy is not considered, which leads to unstable network. Energy-efficient unequal clustering (EEUC) [14], power-efficient and adaptive clustering hierarchy (PEACH) [15], multihop routing protocol with unequal clustering [16], and EEHC [13] require k number of hops to reach cluster head (CH) with overhead in joining the cluster. Hybrid energy-efficient distributed clustering (HEED) [17] has higher overhead for CH election. Sensor Web (S-WEB) [18] is a hybrid technique of centralized and distributed clustering, where most of the tasks are performed by the nodes, except the beacons which are generated from the base station. Ding et al. [25] have proposed distributed weight-based energy-efficient hierarchical clustering (DWEHC) approach to enhance over the HEED

approach by assigning weight to the parameters. Clustering approaches incorporating the mobility have been proposed in [19–24]. LEACH is designed for sensor network without considering the mobility of the nodes. The approach, namely, LEACH-mobile [19, 20], has been proposed incorporating the mobility of the nodes. Similarly, Nagpal et al. have also proposed CLUBS clustering approach [21, 22] for WSN by incorporating the mobility of the nodes. In this approach, the clusters are formed by using local broadcasting and local density of nodes is used as criteria for cluster formation. In CLUBS approach [21], the cluster formation has been carried out considering three properties: (i) every node should come under a cluster; (ii) maximum value of diameter should be equal for each cluster; and (iii) intracluster communication must be allowed. Manjeshwar and Agrawal proposed an event-driven clustering approach called threshold-sensitive energy-efficient sensor network protocol (TEEN) [28]. In this approach, the sensed data is forwarded to the base station only if some event occurs, which is based on two thresholds, that is, soft and hard. The disadvantage of this approach is that the node responds only if the change in the attributes crosses these threshold values, which makes the approach less applicable in dynamic environment, as the selection of two threshold values is very sensitive and difficult for real applications. The user may keep on waiting to get response and does not get any information about the status of the node, which makes this approach not suitable for the applications, where the periodic updates are required. Later, this approach has been enhanced and proposed as adaptive threshold-sensitive energy-efficient sensor network (APTEEN) [29]. APTEEN combines event-driven approach of TEEN and periodic approach of LEACH to address the problems occurring in TEEN. APTEEN is good for periodic applications, but the complexity of the approach increases due to inclusion of extra threshold function and count time. Cluster head election using fuzzy logic (CHEF) has been proposed by Kim et al. [30], which is an enhanced approach over LEACH using two parameters, that is, distance and residual energy for cluster head selection. This approach does not consider the intercluster communication. Bagci and Yazici [31] proposed energy-aware unequal clustering with fuzzy (EAUCF) approach, which is a distributed competitive unequal clustering algorithm. In this approach, the selection of cluster head is performed based on the competition radius, which is determined by distance and energy among the different nodes using probabilistic and fuzzy techniques. This algorithm assumes that working area of each node is directly proportional to its energy; otherwise node will die rapidly.

Many distributed approaches are also available in literature, but it is not possible to use these algorithms in the current form for IoT framework. The reason is that these algorithms do not consider different parameters that are necessary for IoT environment like heterogeneity, support for IP, and mobility. Mobility requires dynamic cluster formation. It is also important to note that IoT network consists of IP-enabled devices and, in such cases, one hop communication is an ideal case and the most desirable one. Multiple IoT nodes act as base stations for underlying sensor nodes, which enables one hop communication. These devices acquire the

information through sensing and send it to the CH. CHs communicate the received information to IoT node. Besides, all the cluster members are at one hop to CHs and all the CHs are at one hop to IoT node, which leads to minimum communication overhead.

In brief, the cluster head performs receiving and forwarding of sensed data or aggregated data. The aggregation of data reduces overall communication within a network, as CH communicates to base station instead of all nodes deployed in the network. Cluster members also help in scheduling communication whenever it is essential in order to save energy consumption of the network. The cluster-based network topology is simple and easy to manage and is conducive for running an application in distributed manner. The cluster is also very helpful in a dynamic environment, where a node leaves or enters the cluster and becomes part of a cluster which is adaptive in nature in the sense that each node decides whether to join the cluster or become a CH. If CHs and cluster formation are changed with time or rounds, then it is called dynamic cluster formation, whereas in static cluster formation, once the clusters are created, they remain the same throughout the network lifetime. Dynamic clustering implies frequent changing of the cluster formation and the respective CH selection at different time intervals. Although frequent cluster formation is carried out due to adaptive nature of the network, there is also an advantage of it with respect to energy. As dynamic clustering allows selecting different CHs at different time intervals, the energy level of different nodes can be incorporated while making such selection as battery lifetime of the node is very critical for overall existence of the network. Thus, dynamic clustering promises to balance the energy level among the different nodes across the different clusters in the network.

To overcome the above limitations and for accommodating the various requirements for minimum communication and maximum collaboration among the sensing nodes and IoT nodes, in this paper, we propose two clustering approaches that are most suitable for IoT architectural framework. These two approaches are, namely, heuristic- and graph-based clustering, respectively. For evaluating the architectural framework and clustering methods introduced in this paper, the proposed algorithms are compared thoroughly with the existing methods in literature using the standard set of parameters like power consumption, communication cost, number of cluster heads, and number of hops. The experiments are executed in IoT-based platform using Contiki-2.7 operating system and Cooja simulator [32]. For summarizing the contribution in this paper, we would like to emphasize that our proposed IoT-based clustering is based on only two-layer architecture, that is, underlying sensor layer and IoT layer, which allows clustering globally in the network in the region of interest and provides the anytime, anywhere data accessibility and on the fly dynamic clustering due to the inclusion of mobility in the approach at IoT layer. There are approaches in the literature reported for clustering with mobility in the wireless sensor network but these approaches are operated locally in the network, whereas IoT-based approach supports the global network scenario anytime and anywhere.

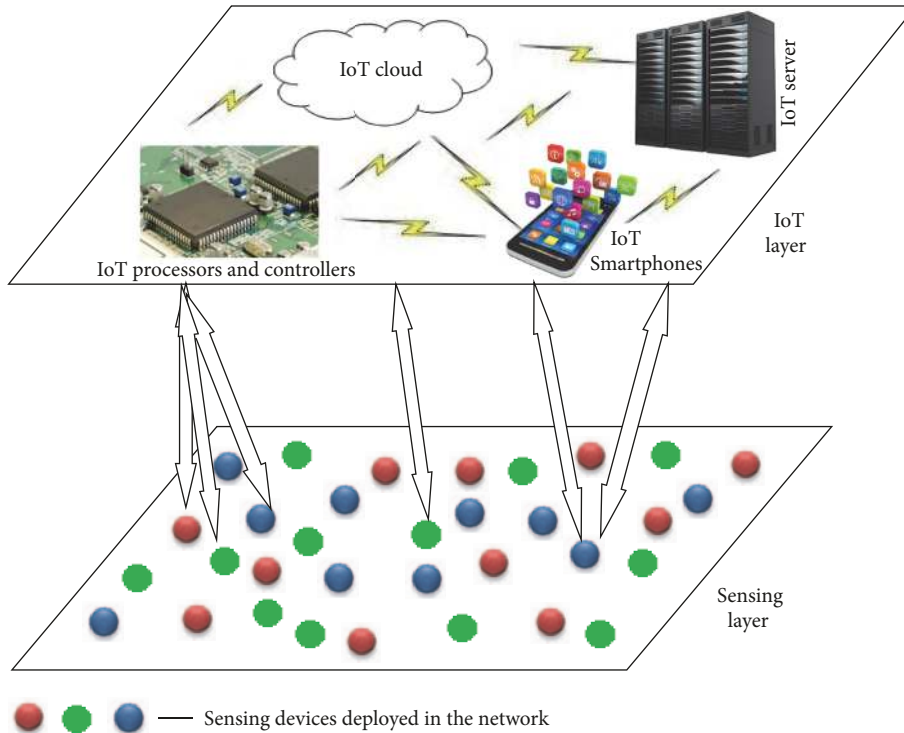


FIGURE 1: Two-layered hierarchical IoT architectural framework.

3. System Model

In this section, the two-layer IoT architectural framework, which represents a system model for proposing the clustering algorithms, is described. The sensor network is mainly used for acquiring the data from the surrounding environment depending upon the applications. IoT network consists of many sensing devices that are not necessarily to be connected to the Internet. In this view, it is preferred to have a framework that differentiates between IP-enabled devices and non-IP-enabled devices, that is, IoT devices and simple sensing nodes without IP capability. This kind of framework provides a layered architecture and is efficient in terms of communication for exchanging the data, which is validated through simulations described later in this paper.

Generally, in wireless sensor network, the nodes that are more than one hop away from the base station or access point consume energy rapidly [33–38]. In IoT framework, if sensor nodes or nodes are static to acquire the data which should be accessible anytime and anywhere, the multihop communication should be handled carefully for optimizing network resources for prolonging the lifetime of an application deployed for a specific task. It is preferred that IoT devices that have more energy and higher end processors as compared to underlying ordinary sensor nodes should be available to underlying sensor nodes for communication for at most two hops. The IoT devices may have mobility and this provides further flexibility to underlying static nodes for communication, so that many such static nodes in the network may be covered by mobile IoT devices. Thus,

ordinary nodes may communicate through local group leader called cluster head (CH) to IoT device and the data acquired through ordinary sensor node may now be accessible anywhere and anytime. For dynamic scenario, the mobility may be incorporated in few IoT nodes whenever it is necessary [39]. In this view, a two-layered IoT architectural framework is introduced.

The architectural framework for IoT applications consists of two layers, namely, IoT layer and sensing layer. Sensing layer is deployed with devices that are either IP-enabled or ID-enabled, depending on the requirement of the application. In this layer, deployed devices include sensors, actuators, and RFID devices. An IoT layer device comprises IP-enabled devices with IoT protocol stack which is significant in nature, because the IoT protocol stack [40] has been introduced to operate in energy-constrained environment at any layer in the network hierarchy, that is, at data link using IEEE 802.15.4e [41], networking using 6LowPAN [42], routing using RPL [40], or application layer using CoAP [43], which results in a huge difference in developing the clustering approach in IoT as compared to that in WSN.

These IoT devices are expected to have longer battery life and storage with the ability to perform real-time processing and communication, as compared to the functionalities provided by ordinary nodes. These features of IoT devices are also necessary for availability of the data or information acquired anywhere and anytime. An important job of mobile IoT nodes is to monitor and collect the information from CHs in the sensing layer. The two-layered IoT framework is shown in Figure 1.

Communication between the devices in the IoT layer and sensing layer consists of different possibilities. As shown in Figure 1, one CH may communicate with one IoT node; two CHs may communicate with one IoT node; and one CH may communicate with two IoT nodes, depending on whether the IoT node is within its range of transmission or not. Although every node in sensing layer may not be in the range of other nodes, still all the sensing nodes are capable of understanding the scenario of entire network through communicating and collaborating with IoT nodes. Assuming that thousands of devices are scattered in the real-time network and are going to take part in accomplishing the given task, communication and connectivity should be addressed in an energy-efficient way. In this regard, the proposed clustering mechanisms for our two-layer architecture are more suitable for any real-time IoT-based application, which is elaborated in detail in the following section.

From this discussion, it is clear that the IoT network may generate a huge amount of data over the time, and thus accessing the same anytime and anywhere is a challenging task. In such a scenario, it is important to have an IoT-based cloud environment that facilitates storing and processing the sensing information in the cloud, where IoT nodes perform real-time processing to accomplish a given task and forward the data to corresponding static IoT nodes or IoT cloud. IoT cloud is connected to IoT servers and IoT nodes are located physically anywhere. IoT server is basically an IoT node that is responsible for high level data processing that helps to make appropriate decisions. Similarly, IoT smartphone is also an IoT node that can be used to access the acquired information and to control the applications remotely. Likewise, IoT processors and controllers are also considered as IoT nodes for small scale card sized computers or embedded processors. For simplicity, the IP-based processors, controllers, and vehicles may be called IoT microcomputers, IoT controllers, and IoT vehicles, respectively. Also, they can be used as parameters on IoT cloud and can be utilized for various applications. Based on applications, the device mobility also plays an important role in IoT environment. Few of these devices may be static and others may be mobile. For instance, IoT cloud and IoT server can be considered as a static node, whereas IoT smartphones, IoT vehicles, IoT microcomputers, and IoT controllers can be considered as mobile nodes. This implies that these IoT nodes are portable and their positions vary at different times. This IoT-layered framework is used for developing the clustering algorithms, which is detailed in the following section.

4. Proposed Approaches

As discussed earlier in this paper, the devices in the network are going to perform various tasks based on application. An important issue is to have energy-efficient connectivity and communication. This can be achieved through clustering or grouping various devices in the layered IoT framework. Any clustering algorithm needs to perform two steps: (i) cluster formation and (ii) CH selection. The cluster formation and CH selection can be carried out in two ways. Based on the

proposed two-layer framework, the clusters can be formed by either of the two layers. In the sensing layer, the nodes form the cluster and determine the CH without intervention of IoT nodes, according to the typical and traditional clustering mechanism. The approach is called *heuristic-based clustering algorithm* that works from bottom to top. The other way of cluster formation and electing CH is by IoT nodes. Assuming that the IoT node has information of sensing nodes that are within the range of the IoT node, it forms the cluster and decides optimally the CH. After cluster formation and CH selection, the sensor nodes are able to communicate with the IoT node through the CH of respective cluster. This approach can be considered as top to bottom approach as IoT layer performs the task of cluster formation and CH selection. To perform this task, a *graph-based clustering algorithm* is proposed, where clustering is carried out in the sensing layer with the involvement of IoT nodes. In the graph-based clustering approach, IoT nodes form the clusters and select CH, which reduces overall communication cost at sensing layer and hence reduces energy consumption, which helps in prolonging the network lifetime. The two proposed approaches are generic and can be applied for any application depending on the need. Further, the proposed approaches work on layered architecture that, in turn, helps manage the nodes, performs tasks optimally, and reduces the communication overhead.

The clustering is performed based on a number of parameters. Generally, these parameters are connectivity, distance between the nodes, residual energy of a node, and so forth. For our proposed algorithm, we consider two parameters, namely, the number of neighbors and residual energy of a node. The reason behind considering the number of neighbors as one of the parameters is that the cluster should be formed using the neighbor nodes only. In our approach, the neighbor node is decided based on the radius of transmission range and the nearby nodes, whose beacon satisfies a certain SNR that is more justified way to decide whether the given node is a neighbor node or not. Simply, the connectivity-based clustering results in the set of nodes in the cluster which are multiple hops apart; still it forms a cluster. This is not a good solution from energy point of view, as multihop communication within a cluster consumes more energy. The best way is to have one hop neighbor and form a cluster. The same scenario is possible using the selection of neighbor nodes based on transmission range and SNR as mentioned above. Moreover, the residual energy is very important parameter as the node with more residual energy should be chosen as cluster head. In our approach, we consider the residual energy as another parameter and the selection of cluster head based on it allows one hop communication to IoT layer, which results in energy-efficient communication.

If a given node has more residual energy, then its transmission range and SNR with respect to other nodes will be more and hence the chance of inclusion of such nodes in the cluster is more. In this way, the residual energy is related to the selection of the number of neighbor nodes. Further, the more the number of neighbor nodes in the cluster is, the more the coverage of the network deployed in the region of interest

is. Due to lesser number of cluster heads, the computational complexity is minimized as minimum number of cluster heads are communicating with the IoT devices in upper layer. It is important to note that this communication takes place with one hop only, which is a major advantage of our proposed hierarchical two-layer architecture.

The number of neighbor nodes is calculated as follows. The number of neighbors for each sensor node i is determined by the following equation:

$$\text{Neighbours Count (i)} = \sum_{j=1}^k X_j, \quad (1)$$

where $X_j = 1$, if node j is within the radius of transmission range with desirable SNR and $X_j = 0$, otherwise.

The term k denotes the total number of nodes within default transmission range. All nodes within the transmission range of a node i do not have desirable SNR and hence those nodes that have desirable SNR are considered for neighbor counting. The desirable SNR can be set to a particular value depending on the application in hand. The residual energy is calculated using radio energy dissipation model [44] and power loss that is proportional to (distance)² in the free space and to (distance)⁴ in the case of multipath fading, respectively. The distance is represented by d . The energy consumption for transmitting m bits message over distance d can be formulated as shown in (2) and (3), respectively.

$$E_{Tx} = m \times (E_{elec} + \epsilon_{fs} \times d^2), \quad \text{if free space} \quad (2)$$

$$E_{Tx} = m \times (E_{elec} + \epsilon_{amp} \times d^4), \quad \text{if multipath}, \quad (3)$$

where ϵ_{fs} is power consumption of the free space propagation, ϵ_{amp} is power consumption of multipath propagation, and E_{elec} represents the residual energy of network. To receive m bits of information, the radio expends as shown in the following equation:

$$E_{Rx} = m \times E_{elec}. \quad (4)$$

Therefore, the residual energy for each sensor node i , $RE(i)$, is determined in our approach by using the following equation:

$$RE(i) = E_0(i) - (E_{Tx}(i) + E_{Rx}(i) + E_{Computation}(i)), \quad (5)$$

where E_0 is initial energy of the node and $E_{Computation}$ is the amount of energy consumption in local processing while executing cluster head selection algorithm on respective nodes.

4.1. Heuristic-Based Clustering Approach. Heuristic-based clustering approach is introduced using the parameters described above, namely, the neighbors count and residual energy. Neighbors count also indicates connectivity of the nodes. The cluster formation is within the radio range transmission of a node. Let us assume that the nodes are deployed randomly in the network with unit disk graph medium (UDGM) model [45]. The network is assumed to be dynamic; that is, the nodes may have mobility. The algorithm

performs the cluster formation and elects CH in a sequence of three steps. These three steps are broadcasting, multicasting, and unicasting, respectively. As discussed earlier, in our IoT framework, there are two types of devices: non-IP-enabled devices and IP-enabled devices. The non-IP-enabled devices are addressed using identification number locally by assigned ID and for IP-enabled devices, IP addresses are assigned using IoT protocol stack [40]. Each step is described as follows:

- (i) In broadcasting step, every node broadcasts the packets or sends the beacon signals that consist of its ID or IP address, so that each node may be informed of its neighbors within its radio range of transmission. By this way, all nodes come to know about their neighbors in the network. Besides, each node maintains the neighbor list and total neighbors can be determined by the total count using (1).
- (ii) In the second step, multicasting is performed. Each node sends the neighbor count and residual energy to its neighbors along with their ID or IP address within its radio range of transmission as determined using (1) to (5).
- (iii) Finally, in third step, cluster formation and selection of CH are carried out. Hence, after receiving the information about their neighbors, every node maintains the neighbors list and neighbors count. By determining the maximum count of the number of neighbors and residual energy (RE), CH is elected. Consequently, the cluster is formed by the elected cluster head using the neighbors list it has, which consists of all nodes within its radio range of transmission. The flowchart of the proposed algorithm is presented in the Figure 2. The corresponding algorithmic steps are presented in the form of pseudocode in Algorithm 1.

The proposed algorithm is demonstrated using an illustrative example with three nodes as follows. Let us assume three nodes in the network, as shown in Figure 3, such that node 1 within its range has only one neighbor, that is, node 2. Node 2 has two neighbors within its range, that is, node 1 and node 3. Likewise, node 3 has only one neighbor, that is, node 2. Now, every node exchanges *node* (NC and RE) with all other neighbors and calculates the maximum of NC and RE values, where NC is neighbors count and RE is residual energy.

These maximum values are used as reference values in cluster head selection process. If nodes have maximum NC value but less RE value is selected as cluster head, then the network becomes unstable due to CH die-out over the time. Similarly, if nodes have maximum RE but less NC value, then it may result in more number of clusters in the network and, hence, result in more communication overhead. So, it is required to consider both values simultaneously for cluster head selection. For this, $\max(\text{NC}, \text{RE})$ function is defined, which returns the node number i having maximum value of NC and RE combined together.

Node 1 determines $\max(\text{Node 1 (NC} = 1, \text{RE)}, \text{Node 2 (NC} = 2, \text{RE)})$. It finds the maximum and determines

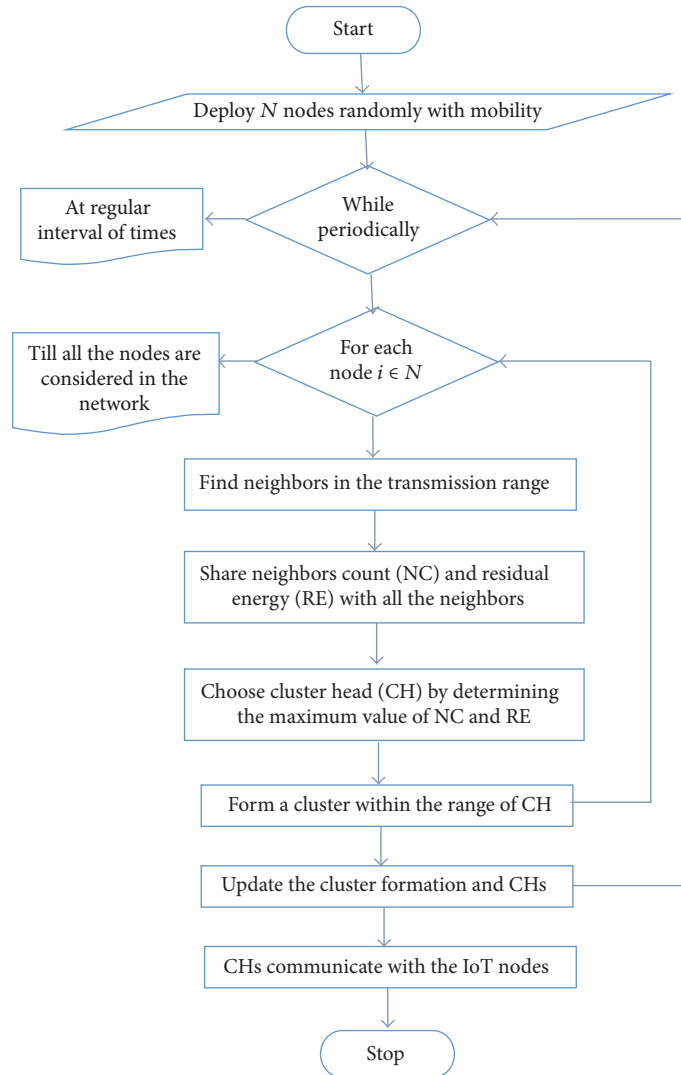


FIGURE 2: Flowchart of heuristic algorithm.

Input Data: N //Number of nodes in sensing layer
Result: CH's election and Cluster's formation

- (1) Deploy the nodes randomly with UDGm model;
- (2) Set the node mobility parameters;
- (3) **while** Periodically **do**
- (4) **for each node** $i \in N$ **do**
- (5) Step 1 → Send ("random messages to find the neighbors");
- (6) Step 2 → Send (NC, RE) to all neighbors;
 // Where NC = Neighbor count and RE = Residual Energy are calculated using equation (1) to (5);
- (7) Step 3 → Determine CH = $\max(NC_j, RE_j)$, //where j is total number of neighbors of i
- (8) Form a cluster within the range of CH.;
- (9) **end**
- (10) Update the Cluster formation and CHs;
- (11) **end**
- (12) CH's Communicate and collaborate with the IoT nodes;

ALGORITHM 1: Heuristic-based clustering algorithm.

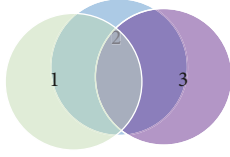


FIGURE 3: Example of cluster head selection and cluster formation.

CH as node 2. Likewise, node 2 computes $\max(\text{Node 1 (NC} = 1, \text{RE)}, \text{Node 2 (NC} = 2, \text{RE)}, \text{Node 3 (NC} = 1, \text{RE)})$ and determines node 2 as CH. Node 3 calculates $\max(\text{Node 2 (NC} = 2, \text{RE)}, \text{Node 3 (NC} = 1, \text{RE)})$ and determines CH as node 2. Hence, CH is selected as node 2 and cluster is formed within its radio range of transmission. In this example, it is assumed that all nodes have same residual energy; that is, $\text{RE}_1 = \text{RE}_2 = \text{RE}_3 = \text{RE}$. RE_i is residual energy of node i . In real scenario, this is not possible, so our *Max* function evaluates the nodes using both NC and RE values simultaneously and returns the corresponding node number. This is possible because NC and RE values are exchanged among all nodes within the transmission range of a particular node. Suppose that if node 2 has lesser residual energy, then it cannot be selected as cluster head, though the NC is maximum compared to that of other nodes. In such a case, instead of node 2, the node that has the maximum residual energy may be selected as cluster head. In the above mentioned example, either node 1 or node 3 is elected as cluster head and forms a cluster. In this way, the cluster head is selected and the cluster formation is performed. Likewise, remaining nodes form other clusters.

Further, in our proposed system, we accommodate the formation of clusters that are based on heterogeneous and homogenous nodes. The reason is that a particular application may need to exchange the data acquired by the different type of the nodes that sense different parameters. In such cases, formation of heterogeneous cluster using different type of nodes and homogenous cluster using similar type of nodes helps to serve the particular functionality of an application. Similar types of nodes or homogeneous nodes are categorized based on the type of sensors or sensed values. For example, consider the cluster of nodes in which all nodes sense the temperature data and another cluster of nodes senses humidity value. Depending upon the need of an application, the cluster may be activated and the data may be acquired for further processing. Even the same node may have temperature and humidity sensors mounted on it. Such nodes may be a part of both clusters with overlapping.

In this context, for various applications in IoT [2, 4, 5, 39], two cases can be defined. In case 1, nodes want to communicate with their own family type. For example, few nodes are deployed to determine the light intensity and other few of them are for detection of carbon monoxide, which implies that the job descriptions of the nodes are different. In such cases, cluster formation and CH selection processes are applied to their own family type. CH is selected from one among the same type of family. If a node broadcasts packets within its range of transmission, the other family

nodes simply reject them and the packets must be received by their family members only.

In case 2, nodes need to communicate with other family types. In this case, different varieties of nodes communicate and collaborate across different family type of nodes. This is a typical scenario representing heterogeneous environment. Nodes must be having the same type of protocol for communication. In the context of IoT, if these nodes are IP-enabled to communicate with other node types, then this leads us to a nonoverlapping cluster formation. CH is selected from the clusters depending on the energy and connectivity. For instance, assume that temperature and humidity nodes are deployed in the network. In such a situation, nodes can communicate with each other to determine weather conditions. The nodes have to communicate and process the information in a collaborative fashion to make the decisions effectively. IP-enabled IoT devices allow communicating among the different family types of nodes through IoT layer, which is an advantage of two-layer framework, even in the case of heterogeneous network. The advantage of our proposed heuristic-based clustering algorithm is that it is simple, easy to implement, and provides effective communication and connectivity within the network as well as across the network through IoT layer.

The heuristic-based clustering is carried out mainly by the sensing layer and the IoT layer has information about CH in the sensor layer. It results into overhead for sensing layer as both the tasks of clustering and data acquiring are to be handled by the sensing layer only. This scenario is good enough for static clustering; that is, nodes are static, and once the cluster is formed, it remains unchanged. In case of dynamic network, where nodes are mobile in the sensing layer, the clustering process needs to be performed whenever the topology changes due to the movement of these nodes. So, it may result into more energy consumption in sensing layer using heuristic-based clustering approach. To overcome this limitation, in the following section, the graph-based clustering algorithm is proposed. In this approach, clustering is performed by IoT layer instead of sensing layer, which is detailed in the next subsection.

4.2. Graph-Based Clustering Approach. An energy-efficient method for clustering is a need for IoT-based applications to extend the network lifetime in real scenario. Assuming that the location information of sensing nodes is available to the IoT nodes [46] and using this location information, IoT nodes can form the clusters and perform CH selection. As mentioned earlier, this strategy is also called top to bottom approach, that is, from IoT layer to sensing layer. Generally, IoT layer has high end processing capability as compared to ordinary nodes in the sensing layer. This strategy helps us deal with the issues of (i) optimal routing in sensing layer for communication across the clusters and (ii) less overhead on sensing devices for cluster formation and CH selection.

In our proposed algorithm, the graph theory is used for clustering and for routing across the two layers, which provides an energy-efficient solution in the IoT environment as evident from the simulation results. Graph theory has been used to solve many real-time problems in an optimal

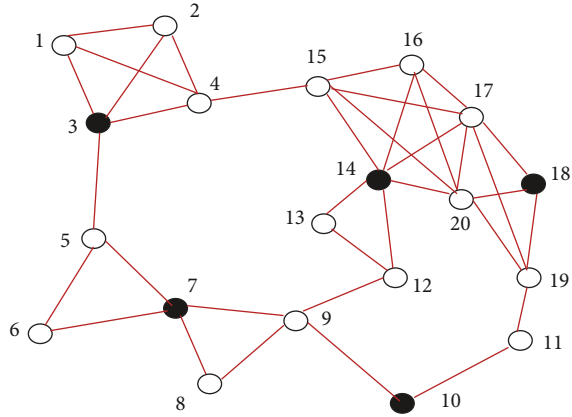


FIGURE 4: Example of dominating set.

way [47]. In the proposed approach, dominating set and bipartite graphs are utilized. Assume that graph $G = (V, E)$ is an undirected graph with V as set of vertices and E as set of edges representing a network, where the vertices are the sensor nodes and edges are the communication links in the network, respectively. The proposed clustering algorithm initially divides V into a collection of subsets $\{V_1, V_2, \dots, V_k\}$ but not inevitably disjoint, where $V = \bigcup_{i=1}^k V_i$, such that each subset V_i induces a connected subgraph of G . Overlapping of these induced subgraphs could be possible. Then, every such subset vertex is a cluster. Here, k is the total number of subgraphs in graph G .

Let us say that $G = (V, E)$ is an abstracted graph constructed such that for every vertex $v_i \in V$ represents a subset V_i . In general, in each cluster, a particular vertex is selected, which acts as cluster head or cluster leader and that corresponding subset forms the cluster. A graph $G = (V, E)$ is said to be a dominating set (DS), if it is a subset $DS \subseteq V$, such that every vertex $v \in V$ is either adjacent to a vertex of DS or in DS. For instance, as shown in Figure 4, the sensing layer has two types of color nodes indicating the CHs and cluster members. The black color nodes are CHs, that is, 3, 7, 10, 14, and 18, which form a dominating set of the graph because these vertices cover all the remaining white color nodes. Further, in this approach, the residual energy of the nodes is considered in the selection of dominating set.

The advantage of having residual energy as one of the criteria for cluster head selection is that if these nodes are selected as CHs in the network, the communication for information collection from sensors and for controlling them can be uniformly distributed to each and every vertex in just one hop. The white nodes communicate with CHs and CHs to IoT nodes and vice versa.

Dominating set problem is classical NP-complete decision problem [47] but many approximation algorithms exist, which provide optimal solutions up to a certain factor. A simple approach is using the greedy strategy by selecting the node that has highest degree with maximum residual energy and removes its neighbors from the set. The algorithm starts with empty DS and greedily adds nodes to DS until all nodes are covered. The natural greedy approach for DS is given

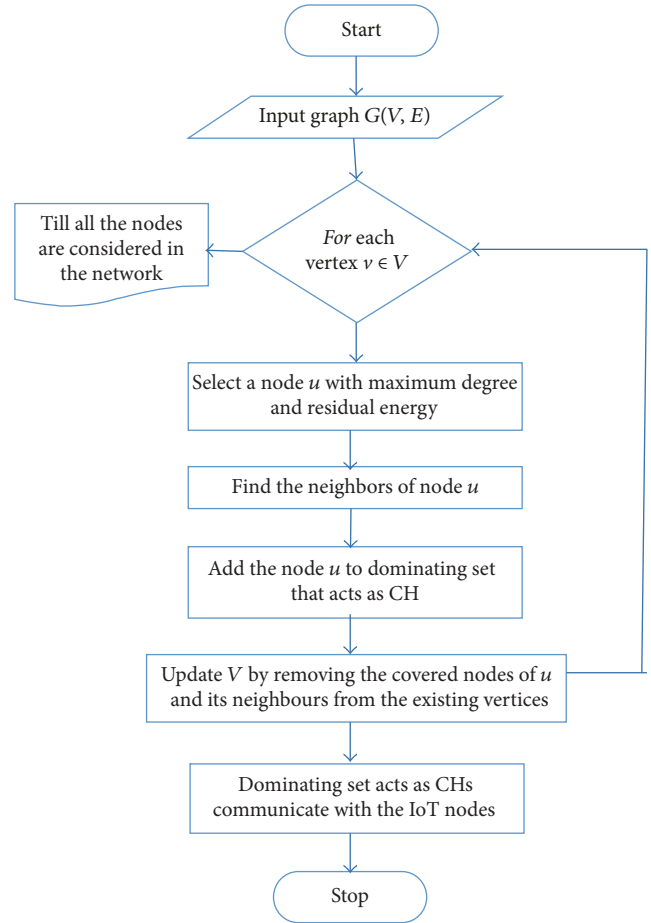


FIGURE 5: Flowchart of graph-based algorithm.

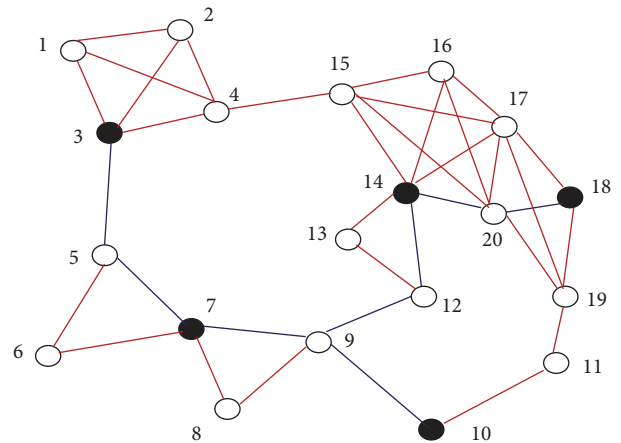


FIGURE 6: Connected dominating set.

in Algorithm 2. The flowchart of the proposed graph-based algorithm is shown in Figure 5.

If you connect all the vertices in the dominating set, then it is called connected dominating set, as shown in Figure 6. For routing computation in the network, connected dominating sets are helpful. This routing application can be achieved by using a small connected dominating set that acts

```

Input Data: Graph  $G(V, E)$ 
Result: DS (Dominating set)
//Greedy algorithm for identifying the dominating set
(1) for each  $v \in V$  do
(2)    $u = \text{Select}(\Delta(G), \max(\text{RE}))$ . //where  $\Delta(G)$  is maximum degree and RE is residual energy;
(3)   Find  $N(u)$ ; //Where  $N(u)$  indicates is neighbors of node  $u \{u \mid u \in G\}$ ;
(4)    $\text{DS} = \text{DS} \cup u$ . //where DS indicates dominating set and adding  $u$  into set;
(5)    $V = \text{update}(V) \{V \mid V \in N(u), u\}$ . //Remove from the covered nodes from the existing vertices;
(6) end
(7)  $\text{CHs} = \text{DS}$ . //Dominating set nodes acts as CHs.;

```

ALGORITHM 2: Graph-based clustering algorithm.

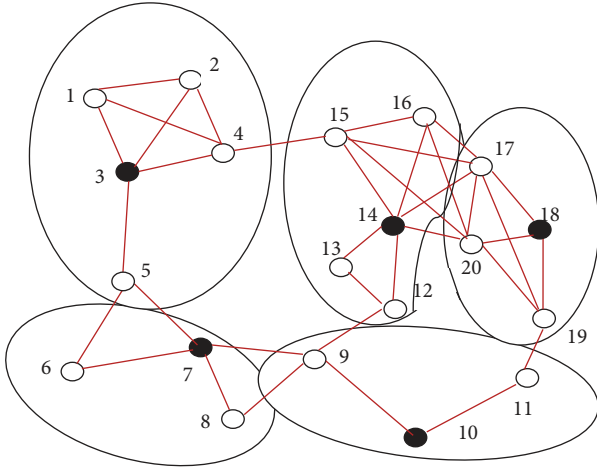


FIGURE 7: Cluster formation on layer 2 by IoT nodes.

as a backbone for communication. For example, nodes 3, 5, 7, 9, 10, 12, 18, and 20 act as the routing path of the network. The remaining nodes that are not in the set of connected path can have the communication by passing messages through neighbors. The cluster formation after applying dominating set in the network is shown in Figure 7.

Because of the mobility of nodes, it is desirable to avoid the same node to be elected as CH again and again. Besides, the significant reason for using residual energy as another parameter is that the possible candidates for CHs are the members of the dominating set by default. It is easy to find a node with maximum residual energy as a cluster head using (2) to (5).

Further, bipartite graph [48, 49] is used to represent the communication between IoT nodes and CHs as shown in Figure 8. A bigraph or bipartite graph is a graph such that the vertices are divided into two disjoint sets of classes, say I and C . These two disjoint classes are independent sets, such that every edge connects a vertex in I to one in C , where I is a set of IoT nodes, that is, $I = \{I_1, I_2, \dots, I_n\}$, and C is set of CHs, that is, $C = \{\text{CH}_1, \text{CH}_2, \dots, \text{CH}_n\}$. There are many advantages of using bipartite graph representation for communication among the two layers. For instance, resource management issues can be handled optimally and network communications can be balanced by periodically

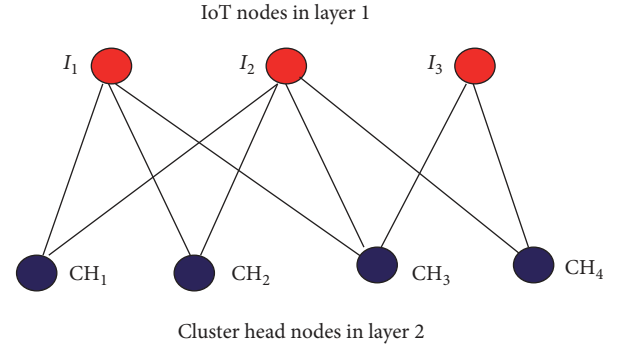


FIGURE 8: Bipartite graph representation of two layers.

reconfiguring the positions of IoT nodes, even in dense deployment of sensing devices. This helps in monitoring and understanding the events occurring in the network.

In the absence of layered architecture, the network is a flat topology. In flat topology, to communicate from one node to the base station or from a node in one cluster to another node in another cluster, multihop communication is needed and this leads to a situation where the nodes nearer to the base station die out faster than the remaining nodes [6, 34], whereas the proposed approaches take at most two hop communications from node to CH and CH to IoT node, reducing the communication cost drastically to a greater extent. Further, the communication for CH selection and cluster formation is carried out for only two times. From the discussion, it is clear that our proposed approach is simple and efficient.

5. Simulation Results

For simulation, the Cooja simulator [32] is used. For evaluating algorithms in IoT environment, all communication should be performed using IoT stack, which is designed specifically for energy-constrained environment for Internet of Things [40–43]. IoT protocol stack is completely different from the normal WSN protocol stack [9, 10, 40]. The best part of the work is that the whole system is evaluated using Cooja simulator [32] for emulating actual IoT environment. All the nodes are deployed randomly in the network. For simulation,

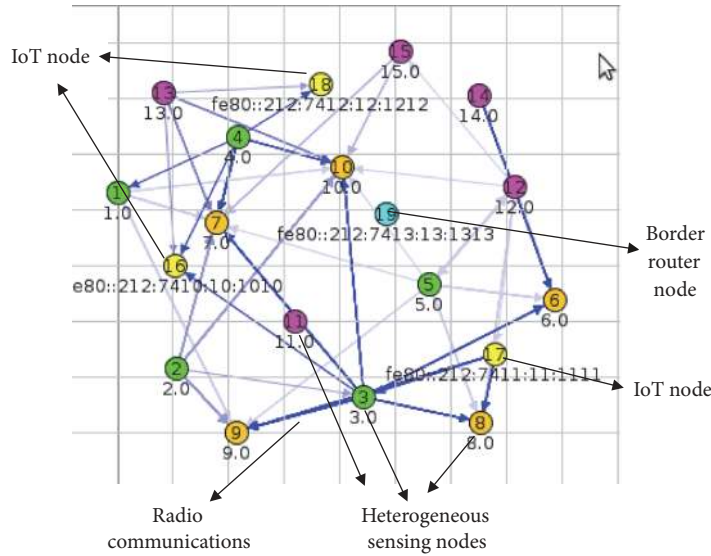


FIGURE 9: Deployment of 15 sensing motes and 3 IoT motes in the network.

the radio transmission range of each node is set to 50 meters and these nodes are mobile in nature. As shown in Figure 9, the simulation is carried out using three types of motes, namely, sky-mote, wis-mote, and Z1 mote, respectively.

These motes are built-in feature of the Cooja simulator. In Figure 9, 15 sensing nodes are deployed, which are in the lower layer. Among these 15 nodes, five nodes are of the type sky-mote, which are in green color, five nodes are of the type wis-mote, which are in purple color, and five nodes are of the type Z1-mote, which are in orange color. Similarly, in the upper layer, three IoT nodes are deployed with IPv6 addresses and can be seen as yellow-colored nodes. For the purpose of displaying the sensed values on browser, border router node is deployed, which is highlighted in blue color. This border router node acts as a bridge between the IoT nodes and web page. The sensed information like temperature and light intensity values from sensing layer is communicated to IoT layer. The corresponding values of the readings are as shown in Figure 10. By accessing the IPv6 address of IoT nodes in the web browser, it is possible to view and analyze the average sensed information across the scattered network.

In the literature, large numbers of clustering algorithms have been reported for wireless sensor network. For better evaluation of the proposed algorithms, it is necessary to compare the methods with relevant approaches reported in the literature. So far, there is no method reported for clustering in IoT environment in the literature. In such a case, the comparison seems to be difficult but there are standard clustering algorithms available for WSN, which are very baseline approaches in nature and robust in performance. Various variants of these baseline approaches have been modified for hierarchical network in the wireless sensor network, even with mobility of the nodes. This leads us to compare our algorithms for better evaluation with existing baseline standard algorithms, like LEACH and its variants with mobility, which is described later in this section.

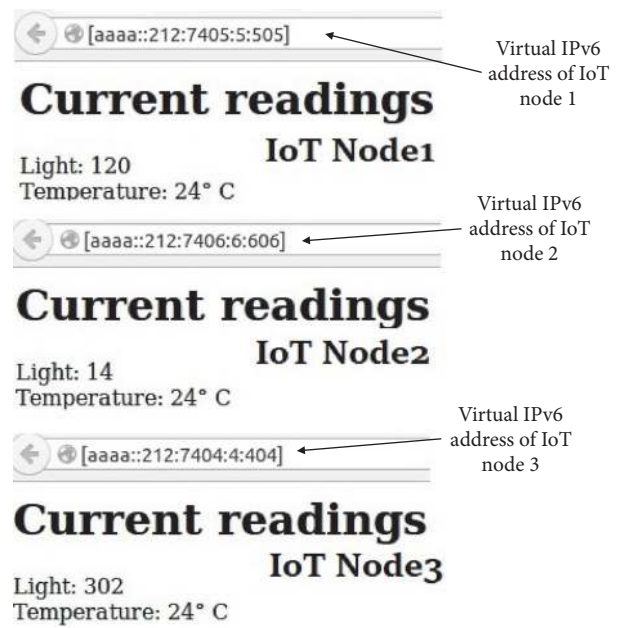


FIGURE 10: Sensed values of the network by IoT nodes.

A network, whether IoT or WSN, inherits common characteristics in itself; that is, both are wireless in nature. But, as pointed out earlier, clustering in the IoT-based network or wireless sensor network has to address the same set of issues for real-time applications. Hence, the evaluation criteria or performance parameters of clustering algorithm remain the same for both networks. In the absence of IoT-based clustering algorithms in literature, the proposed clustering algorithms are compared with standard and most widely used clustering methods in wireless sensor network using various

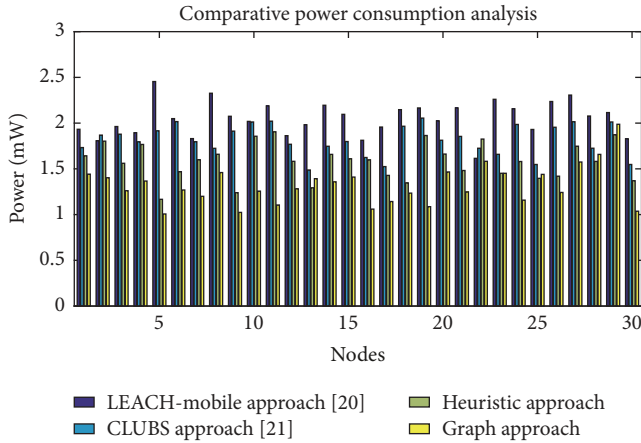


FIGURE 11: Comparative analysis of power consumption for clustering using different approaches.

parameters like number of hops, number of cluster heads, power consumption, and communication cost [19–31].

LEACH [11] is the standard algorithm and is widely used for clustering in various applications of WSN. As discussed in the literature, there are other variants of LEACH approach [19–22], which address the concept of mobility in the case of WSN. In this context, our proposed approaches are compared and evaluated with LEACH-mobile approach [20] and CLUBS approach [21], respectively. The simulation is performed with varying number of nodes. Figure 11 shows the power consumption at each node in the network using all four algorithms.

The dark blue color line indicates the result using LEACH-mobile approach [20] and light blue color line indicates the power consumption of CLUBS approach [21]. The green color represents the output using heuristic-based clustering approach and the yellow color shows the result using graph-based clustering approach. The power consumption of 30 nodes is calculated for our proposed approach and compared with CLUBS [21] and LEACH-mobile [20] approaches as shown in Figure 11. From Figure 11, it is clear that the power consumption using our approaches, namely, heuristic- and graph-based approaches, is very less as compared to CLUBS [21] and LEACH-mobile [20] approaches. It is observed that, for all 30 nodes, LEACH-mobile and CLUBS algorithms consume more power than our proposed approaches. The efficacy of the proposed approach is demonstrated. It is important to notice that the graph-based approach consumes less power than the other three approaches, because the clustering is carried out by the IoT layer which reduces the overall overhead of sensing nodes.

Another criterion for evaluation is the number of CHs elected in the network while clustering. It is preferred that the network should have maximum coverage with minimum CHs. Indirectly, it reduces overall communication cost and helps in prolonging the network lifetime. The comparative analysis of CHs selection using LEACH-mobile [20], CLUBS [21], and proposed algorithms is shown in Figure 12. The reason behind the more number of CHs in the case of

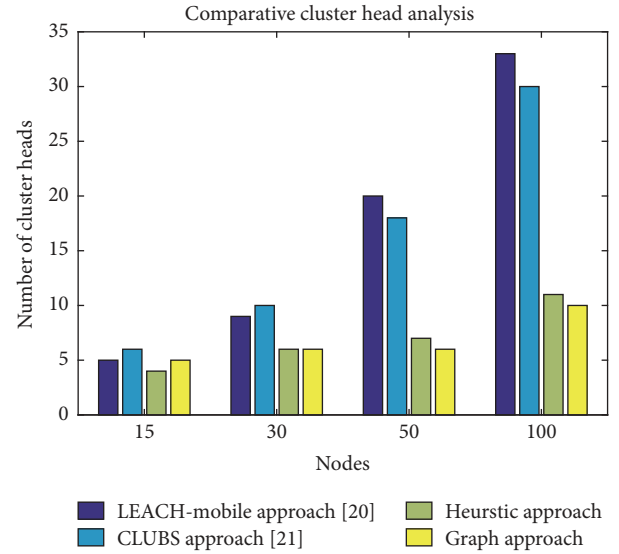


FIGURE 12: Comparison for number of cluster heads with variation in density of nodes.

LEACH-mobile [20] and CLUBS [21] approaches is illustrated as follows.

The operation of LEACH is divided into rounds. Each round begins with a setup phase when the clusters are organized, followed by a steady-state phase when data is transferred from the nodes to the cluster head and onwards to the base station. LEACH-mobile approach forms clusters by using a distributed algorithm, where nodes make autonomous decisions without any centralized control. In LEACH-mobile approach, because of the random election of cluster heads, the optimal number and distribution of cluster heads cannot be ensured [50]. The above complete process is performed in the static environment. But, as stated earlier, we also incorporate the mobility, which results into varying number of nodes from each round to another, and hence the number of cluster heads is more in each round. This is mainly due to mobility and randomness in electing the cluster heads.

In the proposed algorithms, the residual energy is used for CH selection. The simulation results indicate that the stability of the network is preserved, which implies that the whole network lifetime is extended. In Figure 12, it can be noticed that, for small number of nodes like 15 and 30, the numbers of CHs are almost the same for the proposed and existing LEACH-mobile and CLUBS approaches, but as the number of nodes increases to 50 and 100, proposed algorithms have less number of CHs compared to LEACH-mobile and CLUBS approaches. Since LEACH-mobile and CLUBS approaches elect the CHs based on randomness and with certain probability, the number of CHs is more. LEACH-mobile and CLUBS approaches do not consider the residual energy in the process of CH election. But, in our proposed approaches, the residual energy is considered for CH election, which is more practical approach and justified compared to randomly choosing the CH for real-time application.

Also, in terms of network coverage, using our approaches, the minimum CHs are required as depicted in Figure 13.

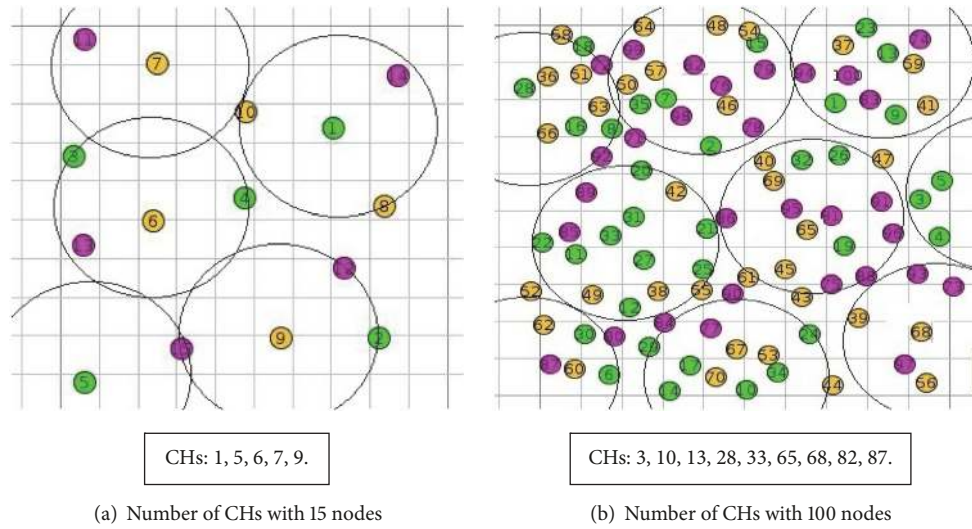


FIGURE 13

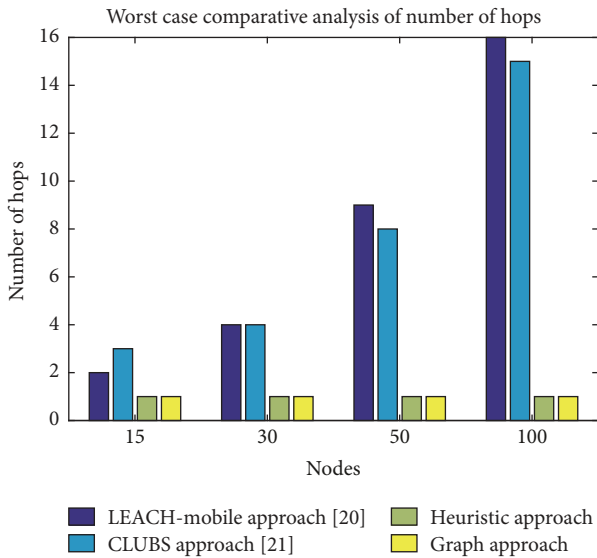


FIGURE 14: Worst case comparison for number of hops with variation in density of nodes.

For instance, as shown in Figure 13(a), in clustering and CH election with 15 nodes, nodes 1, 5, 6, 7, and 9 are elected as CHs, whereas, in the same network, assuming that 100 nodes are deployed, 9 CHs are elected covering all the nodes, that is, 3, 10, 13, 28, 33, 65, 68, 82, and 87, which is depicted in Figure 13(b). Further, the proposed algorithm is evaluated in terms of the number of hops. As shown in Figure 14, irrespective of number of nodes, the number of hops is one. This is because cluster formation is carried out using the node transmission range. All the cluster members are able to communicate to CH in a single hop.

Another comparison among these four algorithms performed is based on the number of hops required to reach the destination, that is, the base station in the given network.

Figure 14 depicts the result related to the number of hops occurring while communicating to destination and base station in the network using our proposed methods and LEACH-mobile [20] and CLUBS [21] approaches.

From the figure, it is clear that the number of hops required using our proposed methods is less compared to the other two approaches, LEACH-mobile and CLUBS. The reason is that LEACH-mobile approach operates with fixed and single base station and the number of hops to reach the base station from all the cluster heads is more. Considering the worst case, the cluster heads are randomly selected, which are farther from the base station; then, this may even result in an increased number of hops to reach the base station. However, in our proposed approach, due to two-layer architecture, all nodes are able to reach the upper layer, that is, IoT nodes that act as base station, in one hop only. The comparison is demonstrated for different set of nodes in the network. From Figure 14, the worst case scenario of choosing number of hops is carried out and it is clear that the numbers of hops are 2, 4, 9, and 16 for LEACH-mobile approach and 3, 4, 8, and 15 for CLUBS approach for the 15, 30, 50, and 100 numbers of nodes, respectively. It is evident from the above discussion that our proposed approaches outperform the existing approaches.

Another important parameter for evaluating the proposed algorithms compared to other methods reported in the literature is overall communication cost. The communication cost using two-layer topology and flat topology is compared. In typical WSN, the cost of communication overhead is 50–70% more than the cost of computation [44, 45]. Lesser number of communications leads to increasing the network lifetime. Since our proposed approach has less number of CHs and less number of hops, this leads to reduced number of communications.

As shown in Figure 15, the proposed approaches ensure less communication cost, which leads to prolonged network lifetime. The stability in the network depends on the battery of the sensors. More communication implies that sooner the

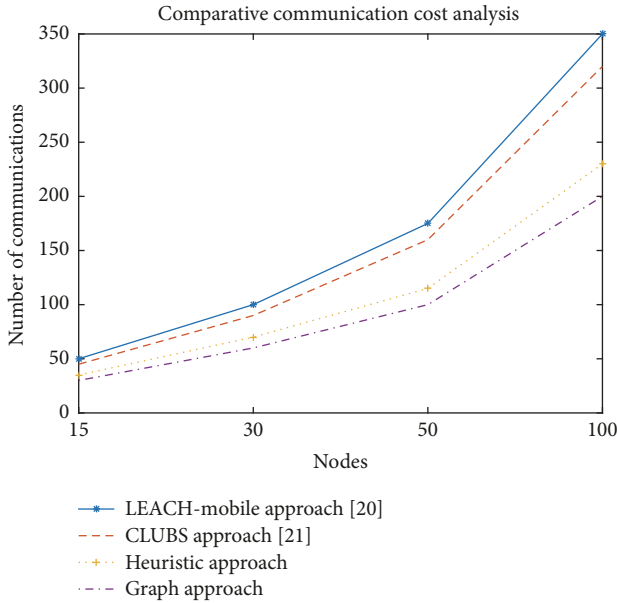


FIGURE 15: Comparison of communication cost with variation in density of nodes.

network becomes unstable due to battery drain of the node. It is observed that the number of communications in the case of LEACH-mobile [20] and CLUBS [21] approaches is significantly more than our proposed approaches. Since in our proposed approach we use only two times communication for CH election and cluster formation, the communication cost is very less, whereas, in the LEACH-mobile and CLUBS approaches, they form the cluster using broadcasting of random messages for minimum three times, which results in more communication. From the above discussion, it can be concluded that the proposed approaches are much better in terms of communication cost and result in more stability or prolonged lifetime of the network.

6. Conclusion

Due to advancement in the technology and need for machine-to-machine connectivity, the Internet of Things overplays the role compared to wireless sensor network. In this context, different applications based on IoT need to be executed efficiently in terms of energy and communication. To achieve this, there is a need to collaborate among various devices at various levels, which can be achieved by the grouping of these devices, that is, through the clustering. In this view, we introduced a pragmatic architectural framework for IoT in this paper and subsequently proposed two clustering algorithms, that is, heuristic- and graph-based algorithms, respectively.

The proposed two-layer framework is justified as most of the applications using IoT are sensing based static or dynamic using IP-based or non-IP-based devices. Secondly, the two-layer architecture allows one to have the sensed values or information anywhere and anytime using IoT devices. For any IoT application having huge number of nodes at sensing layer and IoT layer, it must have effective connectivity and

efficient communication to have data availability anywhere and anytime with prolonged lifetime of the network. Two clustering algorithms have been proposed using the neighbors count and residual energy. The heuristic- and graph-based approaches allow performing the clustering in bottom to top and top to bottom ways depending on the need of an IoT application. The proposed algorithms are evaluated using a number of standard parameters and compared with existing algorithms. The proposed layered IoT framework provides hierarchical structure for efficient connectivity. Our proposed clustering algorithms also allow forming different clusters in presence of mobility and heterogeneity. The dynamic cluster formation and efficient cluster head selection ensure effective connectivity and efficient communication. From the simulation results, it is clear that the proposed clustering algorithms outperform the existing algorithm in terms of power consumption and the number of hops required for communication across the network. The number of neighbor nodes and residual energy are proven to be very effective for cluster formation and cluster head selection. The graph-based approach with dominant set and bipartite graph allows forming a cluster optimally and also provides a routing path for communication.

The algorithm implementations are carried out using IoT-based Cooja simulator, which ensures that the proposed algorithms are evaluated in IoT environment. Comparative analysis of communication cost with respect to the flat topology and layered topology draws to the conclusion that layered topology provides better results. The crucial factor of the network depends on the residual energy of the nodes, which is taken into consideration for cluster formation, which concludes that the proposed approaches are feasible. Results indicate that the optimal number of CHs has been elected and thus it indicates maximum network coverage. Minimum number of communications and the amount of power consumption in the network promise prolonged network lifetime. The proposed algorithm can be extended to connect with IoT cloud to realize the computation in fog environment. Various parameters can be incorporated to define the characteristics of heuristic or graph edges for clustering depending on the applications on hand.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work is supported by the Ministry of Electronics and Information Technology (MeitY), funded by Ministry of Human Resource Development (MHRD), Government of India (Grant no. 13(4)2016-CC&BT).

References

- [1] J. S. Kumar and D. R. Patel, "A survey on internet of things: security and privacy issues," *International Journal of Computer Applications*, vol. 90, no. 11, pp. 20–26, 2014.

- [2] M. A. Zaveri, S. K. Pandey, and J. Sathish Kumar, "Collaborative service oriented smart grid using the Internet of Things," in *Proceedings of IEEE at International conference on communication and signal processing*, pp. 1412–1418, April 2016.
- [3] S. Penga, W. Yao-huib, and C. Chaoc, "IoT core technology and its applications," *Journal of Communications Technology*, vol. 5, p. 35, 2011.
- [4] K. S. Yeo, M. C. Chian, T. C. W. Ng, and D. A. Tuan, "Internet of things: Trends, challenges and applications," in *Proceedings of the 14th International Symposium on Integrated Circuits, ISIC 2014*, pp. 568–571, December 2014.
- [5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [6] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Journal of Computer communications Elsevier*, vol. 30, pp. 2826–2841, 2007.
- [7] R. Khan, S. U. Khan, and R. Zaheer, "Future internet: the internet of things architecture, possible applications and key challenges," in *Proceedings of the FIT*, pp. 257–260, December 2012.
- [8] "Architectural reference model (ARM) for internet of things," <http://iotforum.org/wp-content/uploads/2014/09/120613-IoT-A-ARM-Book-Introduction-v7.pdf>, 2011.
- [9] J. S. Kumar and M. A. Zaveri, "Clustering for collaborative processing in IoT network," in *Proceedings of the Second International Conference on IoT in Urban Space, ACM*, pp. 95–97, 2016.
- [10] J. S. Kumar and M. A. Zaveri, "Hierarchical Clustering for Dynamic and Heterogeneous Internet of Things," *Elsevier Procedia Computer Science*, vol. 93, pp. 276–282, 2016.
- [11] H. Gou and Y. Yoo, "An energy balancing LEACH algorithm for wireless sensor networks," in *Proceedings of the 7th International Conference on Information Technology: New Generations (ITNG '10)*, pp. 822–827, April 2010.
- [12] V. Loscri, G. Morabito, and S. Marano, "A two-levels hierarchy for low-energy adaptive clustering hierarchy (TL-LEACH)," in *Proceedings of the IEEE Vehicular Technology Conference*, vol. 3, p. 1809, 2005.
- [13] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proceedings of Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 3, pp. 1713–1723, 2003.
- [14] C. Li, M. Ye, G. Chen, and J. Wu, "An energy-efficient unequal clustering mechanism for wireless sensor networks," in *Proceedings of IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, pp. 1–8, 2005.
- [15] S. Yi, J. Heo, Y. Cho, and J. Hong, "PEACH: power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks," *Computer Communications*, vol. 30, no. 14-15, pp. 2842–2852, 2007.
- [16] B. Gong, L. Li, S. Wang, and X. Zhou, "Multihop routing protocol with unequal clustering for wireless sensor networks," in *Proceedings of IEEE and ISECS International Colloquium on Computing, Communication, Control, and Management, CCCM'08*, vol. 2, pp. 552–556, August 2008.
- [17] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.
- [18] H. Le, D. Hoang, and R. Poliah, "S-Web: An efficient and self-organizing wireless sensor network model," in *Springer Lecture Notes in Computer Science*, vol. 5186, p. 179, 2008.
- [19] G. S. Kumar, V. P. Mv, and K. P. Jacob, "Mobility metric based LEACH-Mobile protocol," in *Proceedings of 16th International Conference on Advanced Computing and Communications*, pp. 248–253, India, 2008.
- [20] G. Renugadevi and G. M. Sumithra, "An analysis on LEACH-mobile protocol for mobile wireless sensor networks," *International Journal of Computer Applications*, vol. 65, no. 21, 2013.
- [21] R. Nagpal and D. Coore, "An algorithm for group formation in an amorphous computer," in *Proceedings of the 10th International Conference on parallel and distributed systems*, LV, NV, USA, October 1998.
- [22] T. Firdaus and M. Hasan, "A survey on clustering algorithms for energy efficiency in wireless sensor network," in *Proceedings of the 3rd International Conference on Computing for Sustainable Global Development, INDIACOM 2016*, pp. 759–763, March 2016.
- [23] F. D. Tolba, W. Ajib, and A. Obaid, "Distributed clustering algorithm for mobile wireless sensors networks," in *Proceedings of the IEEE SENSORS*, pp. 1–4, 2013.
- [24] D.-S. Kim and Y.-J. Chung, "Self-organization routing protocol supporting mobile nodes for wireless sensor network," in *Proceedings of the 1st International Multi-Symposiums on Computer and Computational Sciences (IMSCCS '06)*, vol. 2, pp. 622–626, June 2006.
- [25] P. Ding, J. A. Holliday, and A. Celik, "Distributed energy-efficient hierarchical clustering for wireless sensor networks," in *Proceedings of the IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS'05)*, 2005.
- [26] J. Lloret, M. Garcia, D. Bri, and J. R. Diaz, "A cluster-based architecture to structure the topology of parallel wireless sensor networks," *Sensors*, vol. 9, no. 12, pp. 10513–10544, 2009.
- [27] S. Rani, R. Talwar, J. Malhotra, S. H. Ahmed, M. Sarkar, and H. Song, "ICCBP: Inter cluster chain based protocol with cross layer interaction for randomly deployed wireless sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 36, no. 1-4, pp. 257–284, 2017.
- [28] A. Manjeshwar and D. P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," in *Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS '01)*, 2001.
- [29] A. Manjeshwar and D. P. Agrawal, "APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Proceedings of the IEEE at the International Parallel and Distributed Processing Symposium (IPDPSi02)*, pp. 1530–2075, 2002.
- [30] J. Kim, S. Park, Y. Han, and T. Chung, "CHEF: Cluster Head Election mechanism using Fuzzy logic in wireless sensor networks," in *Proceedings of the ICACT*, pp. 654–659, 2008.
- [31] H. Bagci and A. Yazici, "An Energy Aware Fuzzy Unequal Clustering Algorithm for wireless Sensor Networks, Fuzzy Systems (FUZZ)," in *Proceedings of the IEEE International Conference on Digital Object Identifier*, pp. 1–8, 2010.
- [32] "Cooja simulator on contiki2.7 operating system," <http://www.contiki-os.org>, 2013.
- [33] P. Kuila and P. K. Jana, "Energy efficient clustering and routing algorithms for wireless sensor networks: particle swarm optimization approach," *Engineering Applications of Artificial Intelligence*, vol. 33, pp. 127–140, 2014.

- [34] O. Boyinbode, H. Le, and M. Takizawa, "A survey on clustering algorithms for wireless sensor networks," *International Journal of Space-Based and Situated Computing*, vol. 1, no. 2-3, pp. 130–136, 2011.
- [35] D. S. Mantri, N. R. Prasad, and R. Prasad, "Mobility and Heterogeneity Aware Cluster-Based Data Aggregation for Wireless Sensor Network," *Journal of Springer*, vol. 86, no. 2, pp. 1–19, 2015.
- [36] A. M. Krishnan and P. G. Kumar, "An effective clustering approach with data aggregation using multiple mobile sinks for heterogeneous wsn, Wireless Personal Communications," *Journal of Springer*, pp. 1–12, 2015.
- [37] H. Karvonen, J. Suhonen, J. Petäjäjärvi, M. Hämäläinen, M. Hännikäinen, and A. Pouttu, "Hierarchical architecture for multi-technology wireless sensor networks for critical infrastructure protection," *Wireless Personal Communications*, vol. 76, no. 2, pp. 209–229, 2014.
- [38] S. Jabbar, A. A. Minhas, A. Paul, and S. Rho, "Multilayer cluster designing algorithm for lifetime improvement of wireless sensor networks," *The Journal of Supercomputing*, vol. 70, no. 1, pp. 104–132, 2014.
- [39] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Journal of Computer networks in Elsevier*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [40] M. R. Palattella, N. Accettura, X. Vilajosana et al., "Standardized protocol stack for the internet of (important) things," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1389–1406, 2013.
- [41] N. Salman, I. Rasool, and A. H. Kemp, "Overview of the IEEE 802.15.4 standards family for low rate wireless personal area networks," in *Proceedings of 7th International Symposium on Wireless Communication Systems (ISWCS)*, pp. 701–705, UK, 2010.
- [42] Z. Shelby and C. Bormann, *6LoWPAN: The wireless embedded Internet*, vol. 43, John Wiley and Sons, 2011.
- [43] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (coap)," <https://tools.ietf.org/html/rfc7252>, 2014.
- [44] W. Si, H.-Y. Shi, and P. Liu, *A Collaborative Data Gathering Mechanism Based on Fuzzy Decision for Wireless Sensor Networks*, vol. 11 of *SAGE Publications Sage*, 2015.
- [45] S. Funke, A. Kesselman, U. Meyer, and M. Segal, "A simple improved distributed algorithm for minimum CDS in unit disk graphs," *ACM Transactions on Sensor Networks*, vol. 2, no. 3, pp. 444–453, 2006.
- [46] S. K. Pandey and M. A. Zaveri, "Quasi random deployment and localization in layered framework for the Internet of Things," in *The Computer Journal*, vol. 61, pp. 1–21, Oxford university press, 2017.
- [47] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to The Theory of NP-Completeness*, WH Free. Co., San Fr, San Fr, Calif, USA, 1979.
- [48] A. S. Asratian, T. M. Denley, and R. Haggkvist, *Bipartite Graphs and Their Applications*, vol. 131, Cambridge University Press, 1998.
- [49] J. E. Hopcroft and R. M. Karp, "An $n^5/2$ Algorithm for Maximum Matchings in Bipartite Graphs," *SIAM Journal on Computing*, vol. 2, no. 4, pp. 225–231, 1973.
- [50] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the ACM MobiCom*, 1999.



Hindawi

Submit your manuscripts at
www.hindawi.com

