



Distributed on-demand clustering algorithm for lifetime optimization in wireless sensor networks

Amrita Ghosal^{a,*}, Subir Halder^a, Sajal K. Das^b

^a Department of Mathematics, University of Padua, Padua 35121, Italy

^b Department of Computer Science, Missouri University of Science and Technology, Rolla, MO 65409, USA

ARTICLE INFO

Article history:

Received 8 August 2018

Received in revised form 27 December 2019

Accepted 27 March 2020

Available online 9 April 2020

Keywords:

Energy balance

Linear programming

Network lifetime

On-demand clustering

Wireless sensor network

ABSTRACT

Wireless Sensor Networks (WSNs) play a significant role in Internet of Things (IoT) to provide cost effective solutions for various IoT applications, e.g., wildlife habitat monitoring, but are often highly resource constrained. Hence, preserving energy (or, battery power) of sensor nodes and maximizing the lifetime of WSNs is extremely important. To maximize the lifetime of WSNs, clustering is commonly considered as one of the efficient technique. In a cluster, the role of individual sensor nodes changes to minimize energy consumption, thereby prolonging network lifetime. This paper addresses the problem of lifetime maximization in WSNs by devising a novel clustering algorithm where clusters are formed dynamically. Specifically, we first analyze the network lifetime maximization problem by balancing the energy consumption among cluster heads. Based on the analysis, we provide an optimal clustering technique, in which the cluster radius is computed using alternating direction method of multiplier. Next, we propose a novel On-demand, optimal Clustering (OPTIC) algorithm for WSNs. Our cluster head election procedure is not periodic, but adaptive based on the dynamism of the occurrence of events. This on-demand execution of OPTIC aims to significantly reduce computation and message overheads. Experimental results demonstrate that OPTIC improves the energy balance by more than 18% and network lifetime by more than 19% compared to a non-clustering and two clustering solutions in the state-of-the-art.

© 2020 Elsevier Inc. All rights reserved.

1. Introduction

With the emergence of Internet of Things (IoT), the significant of Wireless Sensor Network (WSN) has increased immensely [43], since WSN has a great potential towards providing a cost effective solution to the various IoT services under different environmental scenarios for a variety of reasons, such as ease of deployment in different environments without cabling, and flexibility to couple them with sophisticated but cheap sensors with different modalities. Moreover, information gathered by each sensor node can be collaboratively processed in a distributed or centralized way to evaluate performance metrics for improving our daily life and society. In IoT, WSN covers a wide application range starting from military to industry, environment monitoring to security and surveillance, home automation to smart healthcare, and so on [16]. In a typical deployment, the sensor nodes (henceforth, nodes) are battery-powered microsystems that embed a variable number of transducers to monitor their surroundings. In many IoT applications, WSNs need to work unattended for long periods

of time, either to reduce the cost of maintenance, or because they are deployed in hardly accessible (or even hostile) places. As IoT nodes are expected to rise by 50 billion in 2020, so green networking plays a crucial role in IoT to reduce energy consumption, lessen pollution and emissions. It is estimated that if one node could reduce 1% of its energy consumption, it can save the world \$1 billion worth of electricity [43]. All these factors have motivated manufacturers to produce low power consuming nodes and researchers to design energy-efficient data collection mechanisms.

One important way of energy-efficient data collection in IoT is through organizing the nodes into clusters [43] as shown in Fig. 1. Unlike non-clustering algorithm, generally, clustering avoids energy dissipation due to idle listening, collisions and overhearing [17]. Particularly, in every cluster, a dedicated time slot is assigned to Member Nodes (MNs) for sending its sensed data to the Cluster Head (CH). Apart from awaking in a particular time slot of Time Division Multiple Access (TDMA) frame, a MN does not need to be activated all the time, since the MN knows when to transmit message. In this way clustering prevents the energy depletion due to idle listening and overhearing. Further, to conserve energy, CHs aggregate the data received from MNs and send them to the sink, either directly or via a multi-hop path through

* Corresponding author.

E-mail addresses: amrita.ghosal@math.unipd.it (A. Ghosal), subir.halder@math.unipd.it (S. Halder), sdas@mst.edu (S.K. Das).

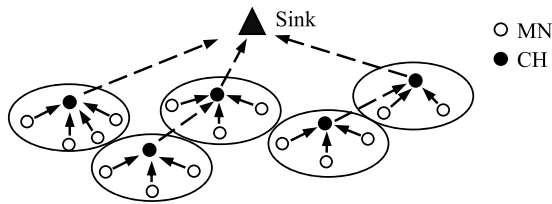


Fig. 1. Clustered IoT.

neighboring CHs. Apart from data aggregation, the network may be reclustered periodically to select energy abundant nodes to function as CHs, thus distributing the load uniformly among all the nodes.

In clustering paradigm, the large difference between the energy consumption of MNs and CHs is a huge motivation for researchers to design energy balancing solutions for avoiding energy-hole problem [23]. Specifically, CHs nearer to the sink bear huge traffic load resulting in faster energy consumption than the other CHs as well as MNs, resulting in energy-hole problem. Due to this energy-hole problem, the whole network leads to premature decrease in network lifetime. Many clustering techniques have been proposed [43] in which network lifetime prolongation by mitigating energy hole problem is the major objective. Based on the clustering properties, the existing techniques can be classified into two categories, namely, uniform clustering and non-uniform clustering [14]. The main philosophy in uniform clustering algorithms [45] is to form the clusters with relatively equal sizes, to keep the number of clusters as small as possible, evenly distribute them across the network, and typically, provide minimum overlapping among them. The major problem in uniform clustering is that the traffic load is not evenly distributed among all the nodes. On the contrary, in non-uniform clustering [7,23], based on the distance between the nodes and the sink, variable sizes of clusters are formed across the network to evenly distribute the traffic load among the nodes (i.e., both CH and MN). Since the traffic load among the nodes is distributed evenly, non-uniform clustering strategy is more promising with respect to energy efficiency than uniform clustering strategy. In this work, we consider non-uniform clustering.

To reduce energy consumption during data gathering, in this work, we consider event driven based data gathering. It is worth mentioning that, depending on the particular application, traffic generated in the nodes can be classified as time driven, event driven, sink initiated and hybrid [14]. Among these methods, event driven is the most energy efficient and challenging, since the occurrence of events is entirely unpredictable, resulting in arbitrary traffic pattern. We also consider in this work that cluster formation is triggered by occurrence of event, i.e., on-demand based. In the on-demand based clustering, a node wakes up only when an event is detected or another node wants to communicate with it. Example of application scenarios where on-demand based clustering strategy is most suitable includes modern parking lot monitoring, battlefield surveillance. Specifically, in battlefield surveillance where nodes are deployed in a large area, each node may be equipped with a camera to take a photograph once an intruder is detected. In presence of obstacles, a node may capture a limited view of the intruder. Nodes thus can form clusters in on-demand basis. Thereafter, CH can collect those captured images from nodes (or, MN) to build the complete view of the region that is being monitored along with the intruder.

Contribution. Many clustering techniques [23,37] have been designed for WSNs to balance energy consumption by dividing the network into non-uniform clusters. Most of the proposed

strategies consider time driven based WSNs and ideal channel model. Furthermore, in existing clustering approaches, CHs are either pre-assigned [37] or selected based on a single parameter such as geographical position [6] or residual energy [23]. The major problems of pre-assigned clustering approaches are that they are neither dynamic nor energy efficient and have limited applications. In contrast, our proposed clustering strategy is dynamic and CHs are selected based on the average energy usage and residual energy level. To the best of our knowledge, there exists no literature that considers optimization of cluster size by including realistic channel model to minimize and balance energy consumption among CHs, while devising on-demand clustering algorithm. We extended the earlier works in several aspects. The major contributions of this paper are as follows:

- First, we study the energy balancing approach that aims directly at maximizing the network lifetime by considering a number of realistic factors like fading model, battery model, the impacts of both inter-cluster and intra-cluster data traffic.
- Based on the analysis, we derive the principle of optimal clustering technique. Specifically, we provide an optimal clustering technique, in which the optimal cluster radius is computed using Alternating Direction Method of Multiplier (ADMM).
- Next, we propose a novel On-demand, oPTimal Clustering (OPTIC) algorithm, where CHs are selected dynamically based on the average energy usage and residual energy level. The invocation of OPTIC is based on the detection of an event of interest. This reduces the unnecessary system updates and hence computation and message overheads.
- Finally, we conduct extensive simulation experiments to evaluate the performance of the proposed OPTIC algorithm by considering realistic scenarios. Simulation results show that OPTIC algorithm can significantly increase the network lifetime without compromising the network performance metrics like throughput, end-to-end delay.

In [14,17], authors proposed similar type of clustering techniques. However, the current work is significantly different from the existing works in many respects. Particularly, as an alternative to existing algorithms [14,17], in this work, we proposed an on-demand clustering algorithm to significantly reduce computation and message overheads. Further, different from [14,17], we determine the optimal cluster radius using ADMM in this work. Finally, in order to increase the reliability of the simulation results, we performed the simulation experiments using one of the popular simulators, i.e., Castalia. We summarized the different key features of the existing works and our proposed scheme in Table 1.

Organization. The rest of this paper is organized as follows. Section 2 discusses the related works. The system model considered for the present work is described in Section 3. Section 4 theoretically analyzes the network lifetime maximization problem and obtains the optimal clustering strategy. The proposed clustering strategy is described in Section 5, after which Section 6 provides an analysis of the proposed clustering strategy. Section 7 presents experimental results under indoor and outdoor environment conditions. Finally, concluding remarks are given in Section 8.

2. Related works

Based on the clustering properties, we classified existing clustering technique into two categories, namely uniform and non-uniform. Here, we briefly summarize the existing uniform clustering (Section 2.1) and non-uniform clustering (Section 2.2) strategies most relevant in our context.

2.1. Energy efficient uniform clustering

In one of the earlier work, Younis and Fahmy [45] proposed a Hybrid, Energy-Efficient, Distributed (HEED) clustering scheme. In HEED, CHs are selected at regular intervals based on two parameters viz. residual energy and intra-cluster communication cost. In one of the potential work's, Xu et al. [42] proposed a clustering technique based on the gradient field to deliver the data packet to the sink with minimum hops. Here, initially, every node in the network obtains its gradient to find the gradient field so that a data packet reaches the sink with minimum hops. Afterwards, CH selection procedure is initiated, where a node is selected as CH based on the gradient, number of neighbors and residual energy. In [14], a static clustering algorithm was proposed to maximize the network lifetime. Initially, they analyzed the energy balancing approach and derived the principle for clustering algorithm. Finally, they proposed a deployment strategy to distribute CHs and MNs in priori known locations. Unlike the earlier solutions, the authors in [22] proposed an Instantaneous Clustering Protocol (ICP). Here, to reduce the cluster formation time, the CH is determined by a pre-assigned probability and its present status. Initially, ICP computes the probability of a node to become a CH and pre-assigns this probability within nodes. Here, the probability is calculated based on the expectation number and the redundancy coefficient. More recently, Karunanithy and Velusamy [20] proposed a distributed clustering method for reliable and stable data transmission. In the proposed method, initially, a set of tentative CH candidates is selected based on the residual energy. Finally, a set of fuzzy rules is applied which takes number of parameters, e.g., number of neighbor nodes, and average connection time to determine the ultimate CH. Hu and Niu [19] proposed an energy-efficient overlapping clustering protocol for enhancing network lifetime. In the proposed protocol, CH is selected based on the residual energy and higher node degree/proximity. Interestingly, to reduce data forwarding load on CHs, the authors introduced relay nodes located at the boundary in the overlapping cluster area.

Different from earlier approaches, Radhika and Rangarajan [35] proposed a fuzzy logic based clustering scheme for lifetime enhancement. In the proposed scheme, the CH is selected based on the residual energy. Interestingly, the authors designed a fuzzy inference system to decide the interval of reclustering procedure. Finally, to reduce data transmission load, the proposed work considered a machine learning technique, where for every set of similar data readings, one reading is sent to the sink. Mittal et al. [28] proposed a fuzzy extended grey wolf optimization algorithm based clustering protocol to prolong lifetime of WSN. Similar to [35], in this work the CH is selected based on the residual energy. However, in order to deal with uncertainties during CH selection, Fuzzy Inference System (FIS) has been adopted along with extended grey wolf optimization to provide the chance value for each node. Liu et al. [25] proposed a stochastic process based clustering technique. In the proposed technique, CH is selected based on the residual energy. Further, the authors proposed a self-adaptive cluster regulation algorithm to improve the robustness of cluster from any unexpected energy depletion. Particularly, the cluster regulation algorithm oversees each node in parallel during the interval of reclustering procedure. If residual energy of any node crosses a threshold, it disconnects the node from the cluster. Neamatollahi et al. [31] proposed a distributed energy-efficient scheme for clustering in WSN. Specifically, the proposed scheme schedules clustering task to enhance energy efficiency and network lifetime. Unlike the popular round robin policy based cluster task scheduling, the proposed scheme uses a dynamic hyper round based scheduling.

2.2. Energy efficient non-uniform clustering

In one of the earlier works, Chen et al. [7] proposed a Unequal Cluster-based Routing (UCR) protocol to mitigate energy hole problem in WSNs. In UCR, initially, based on certain predefined threshold value, nodes form a set of probable CHs. A node from the set of probable CHs becomes CH if its residual energy is maximum. Further, the authors proposed a greedy routing protocol, where a CH chooses a forwarder node based on the highest residual energy and minimum link cost. Shu et al. [37] explored the network lifetime maximization problem by balancing the energy consumption among CHs. To obtain balanced energy consumption, two mechanisms are proposed, viz. routing-aware optimal cluster planning and the clustering-aware optimal random relay. The first method uses a clustering approach that is developed under the perspective of shortest path inter-cluster routing. The second method is a routing strategy for load-balanced clustered topologies. In [23], a cluster size arrangement technique was developed for network lifetime maximization. They also proposed an energy-aware clustering scheme, called COCA, where CHs are elected based on the residual energy. Recently, Sabor et al. [36] proposed a transmission range adjustable-based immune clustering protocol for efficient data delivery to the sink. In the proposed protocol, the cluster size is determined based on the speed of the mobile sensor and CH is selected based on the residual energy. A CH determines its suitable position within the cluster based on an immune optimization algorithm, which takes mobility, energy consumption, connectivity, residual energy and link connection time as input parameters.

As an alternative to periodic reclustering, a static clustering algorithm is proposed in [17], where role of each node is designated at the time of deployment. As the nodes are aware about their function, no separate algorithm is required for the CH election. Further, they proposed a predetermine deployment algorithm to place CHs and MNs in some selected locations. The work in [39], instead, proposed an energy-aware distributed dynamic clustering protocol, where CHs are selected according to on-demand basis. In the proposed protocol, tentative CHs are selected by considering delay times. Finally, the authors use fuzzy logic to assess the fitness cost of the tentative CH to become a CH. The fitness cost is derived considering two metrics viz. node degree and node centrality. Similar to Taheri et al. [39], Baranidharan and Santh [2] proposed a clustering algorithm, named Distributed Unequal Clustering using Fuzzy logic (DUCF), where CHs are selected using fuzzy logic rules. Nevertheless, DUCF uses FIS to elect CHs. Specifically, DUCF takes the residual energy, node degree and distance to sink as inputs during fuzzification. The fuzzified input parameters are evaluated using 27 fuzzy if-then rules in a fuzzy inference system. The defuzzified outputs are the chance of a tentative node to become CH, and maximum number of nodes to be its MNs. Li et al. [24] proposed a clustering technique to reduce the data delivery delay to the sink. Through theoretical analysis, authors show that small cluster size located far from the sink is beneficial for efficient data aggregation. The effectiveness of the proposed clustering technique is measured thorough simulation experiments considering scheduling length and network lifetime as performance metrics. In [6], Bozorgi et al. proposed a hybrid clustering technique for increasing network lifetime. Interestingly, to reduce the unnecessary transmission control messages, authors proposed to use static CH for a number of rounds, where CH is decided by the sink. Later, if the delay time reaches a certain threshold, a new CH election phase is triggered, where a CH is elected based on the minimum delay time, highest residual energy and location at the center of the cluster.

In Table 1, we summarize the key features of the existing works and our proposed scheme. In our specific context of non-uniform clustering approaches, we have the following observations: (1) although, on-demand based clustering can improve

Table 1
Comparison of typical clustering techniques.

Scheme	Formation	Selection rule	Channel model	Network lifetime
Uniform clustering				
Younis and Fahmy [45]	Time driven	RE & Node degree	Ideal	Prolongation
Ghosal and Halder [14]	Time driven	Pre-assigned	Rayleigh	Maximization
Xu et al. [42]	Time driven	RE & Node degree	Ideal	Prolongation
Kong et al. [22]	Time driven	Pre-assigned	Ideal	Prolongation
Karunanithy and Velusamy [20]	Event driven	RE & Fuzzy rules	Ideal	Prolongation
Hu and Niu [19]	Time driven	RE & Node degree	Ideal	Prolongation
Radhika and Rangarajan [35]	Time driven	RE	Ideal	Prolongation
Mittal et al. [28]	Time driven	RE & FIS	Ideal	Prolongation
Liu et al. [25]	Time driven	RE	Ideal	Prolongation
Neamatollahi et al. [31]	Event driven	RE	Ideal	Prolongation
Non-uniform clustering				
Halder and Ghosal [17]	Time driven	Pre-assigned	Rician	Maximization
Li et al. [23]	Event driven	RE	Ideal	Prolongation
Shu and Krunz [37]	Time driven	Pre-assigned	Rayleigh	Maximization
Taheri et al. [39]	Event driven	Fuzzy rules	Ideal	Prolongation
Bozorgi et al. [6]	Time driven	Hybrid	Ideal	Prolongation
Sabor et al. [36]	Event driven	RE	Ideal	Prolongation
Baranidharan and Santhi [2]	Time driven	Fuzzy rules	Ideal	Prolongation
Li et al. [24]	Time driven	Pre-assigned	Ideal	Prolongation
OPTIC	Event driven	Energy usage & RE	Log-normal	Maximization

RE: Residual Energy; GP: Geographical Position.

energy conservation, except Taheri et al. [39], none of the works consider the on-demand based clustering approach as a solution to maximize the network lifetime. Therefore, we are motivated to devise an on-demand based clustering strategy. (2) In majority of the literature, residual energy is considered as the only parameter to select a CH, thus overlooking the possible conflict of several nodes with same residual energy. Motivated by this fact, we considered average energy usage in addition to residual energy level while selecting CH. (3) Except Shu and Krunz [37], Halder and Ghosal [17], Ghosal and Halder [14], none of the works considered realistic channel model while proposing clustering algorithm. In contrast, to the best of our knowledge, we are the first to consider the log-normal path loss model while devising the on-demand, optimal clustering algorithm.

3. System model

In this section, we describe the models used in this work. In particular, Section 3.1 presents the network model. Section 3.2 discusses the network operation model. We then introduce the energy consumption model in Section 3.3. Section 3.4 presents the path loss model. Finally, in Section 3.5, we describe the network lifetime model.

3.1. Network model

We consider a network area χ of radius D which is covered by a disk sector of angle ϕ as shown in Fig. 2 [17]. Nodes are randomly and uniformly deployed across χ with density ρ . The disk sector is partitioned into i ring sectors or slices, where $i = 1, \dots, N$. The sink is considered to be located at the vertex, as shown in Fig. 2, and responsible for gathering data from nodes. We assume a slice based network area, and argue that this slice shape is general enough to estimate many other shapes like rectangle, triangle, and square [17,37]. For example, as shown in Fig. 3, we can approximate triangle, circle, rectangle and square shape network area as slice based network area by setting $\phi = 60^\circ$, $\phi = 360^\circ$, $\phi = 180^\circ$ and $\phi = 90^\circ$, respectively. It is also worth mentioning that Olariu et al. [32] provided an in-depth discussion on real life implementation of the slice based network area. Further, in [26], Luo and Hubaux proved that, for enhancement of network lifetime, the best position for a sink is

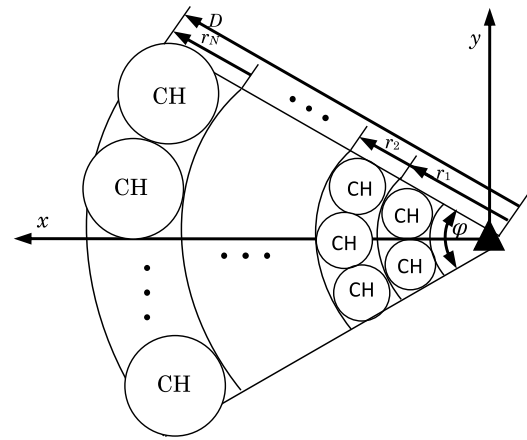


Fig. 2. Network area division into slices.

the center of the circle in slice based network area. Primarily, these two works [26,32] along with other relevant works [17,37] motivate us to consider the present network model.

In our work, we consider deployment of static nodes, where nodes can adjust its communication range. At present, nodes with adjustable communication ranges are commercially available. For example, in [21], the authors have designed a real-time human traffic monitoring system using nodes with adjustable communication range. Further, we consider continuous monitoring applications like battlefield monitoring, intrusion detection in which each node always remains active and senses the environment continuously, however it generates data packets only when it has sensed an event. Once an event is detected, the cluster setup phase is triggered and nodes that detect an event only take part. Rest of the nodes remain in active mode for detecting future event(s) and/or forwarding data packets. For the sake of simplicity, in this paper, the cluster setup phase is divided into two sub-phases, namely, CH selection, and cluster building. After cluster setup, each MN acquires data from the surrounding environment, generates n bits data packet, and transmits the data packet to its CH at every round [14,17]. On the other hand, the CH fuses and forwards the data packets received from both its MNs and neighboring CHs which are far away from the sink. This

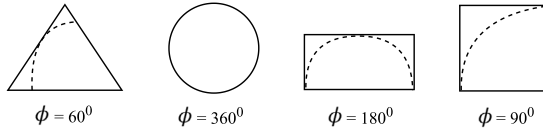


Fig. 3. Network areas that can be approximated by a slice based network area.

process repeats till the data packets arrive at the sink from a MN through intermediate CHs. As we divide the area χ into N slices, thus N types of clusters exist in the network.

In particular, the clusters located nearest to the sink are placed in the 1st slice (1st type), and the ones farthest from the sink are placed in the N th slice (N th type). In this work, we assume that the MNs whose distances from the sink fall in $(r_{i-1}, r_i]$ are organized into clusters of the i th type, where $1 \leq i \leq N$ and $r_1 < \dots < r_N = D$. Therefore, the disjoint clusters of i th slice cover the area $\{(x, y) | r_{i-1}^2 < x^2 + y^2 \leq r_i^2, (x, y) \in \chi\}$. Accordingly, the CHs of i th slice are deployed along the circle $\{(x, y) | x^2 + y^2 = d_i^2\}$, where $d_i = (r_{i-1} + r_i)/2$, with equal angular gap between adjacent CHs. Further, we assume a WSN, where nodes are divided proactively into many clusters at the time of deployment phase [37]. In theoretical analysis, for the sake of simplicity, an ideal Medium Access Control (MAC) layer with no collision and retransmission is assumed. The reason for not considering real MAC in theoretical analysis are the complexities involved in real MAC such as need of control packet (TDMA scheduling), retransmission in case of collision (CSMA/CA) etc. Moreover, some of the overheads such as retransmission are real time parameters which need probabilistic formulation adding to further complexity in theoretical analysis. However, in our simulation, we implemented two MAC protocols, viz. TDMA and CSMA/CA, for investigating the impact caused by them to make the assumption more realistic.

3.2. Network operation

In this work, we assume Collection Tree Protocol (CTP) [15] as a routing protocol for delivering data packet to the sink. CTP is widely considered as an efficient routing protocol in WSN and provides “best-effort anycast datagram communication to one of the collection roots in a network” [15]. In recent past, CTP has been implemented and available in number of realistic WSN simulators, e.g., TinyOS 2.1 [13], Castalia [9]. Here, we briefly recall the philosophy of data collection and routing path formation methods of CTP.

In CTP, once the nodes are deployed, network set-up phase starts. In this phase, each node learns about their one-hop neighbors by exchanging beacons among themselves. In particular, based on the message exchange, a node first computes the link quality between the current node and one-hop neighbor node through a metric, Expected Transmission (ETX) count. Once the network finishes the set-up phase, a tree rooted at the sink is formed and sets ETX for the root as 0. In CTP, the other node calculates its ETX by determining the sum of the ETX of its parent and ETX of its link to its parent, and store ETX in the memory. After the tree construction, network is aware about the parent set of node and child set of node. A child node determines a routing path to forward the data packet to a parent node towards the sink which has the lowest ETX count. Next, the parent node employs the same procedure to choose the next forwarder node for sending its data packet. This process is repeated till the data packet arrives at the sink. Detailed procedure for calculation of link usages and optimal flow can be found in [10].

3.3. Energy consumption model

In this work, we determined the energy consumption in a WSN as a function of its various components, e.g., radio transceiver. Particularly, we used energy consumption model of one of the most popular sensor nodes, i.e., TelosB node [34] in this work. The TelosB node embeds a Texas Instruments CC2420 radio transceiver operating at the 2.4 GHz band. According to the data sheet of TelosB node, the radio transceiver consumes 18.80 mA current during the receiving model (I_{rx}), whereas, 17.40 mA current during the transmission mode (I_{tx}). It is worth noting that TelosB node comprises with a 3 V power supply, i.e., two 1.5 V batteries. Based on these power consumption specifications, we estimated the energy required to transmit a message as $e_{tx} = T_{tx}P_{tx}$, where T_{tx} is the required time to send a message. Similarly, we estimated the energy required to receive a message as $e_{rx} = T_{rx}P_{rx}$, where T_{rx} is the required time to receive a message.

We determined the P_{tx} and P_{rx} as functions of the current consumption of the radio transceiver. For a TelosB node, we estimated using the following equations:

$$P_{tx} = V_b I_{tx} = V_b \times 17.4,$$

$$P_{rx} = V_b I_{rx} = V_b \times 18.8,$$

where V_b is the battery voltage. In this work, instead of classical ideal battery, we assumed battery model based on technical specifications [11], i.e., (1) battery voltage varies based on the internal resistance, residual capacity and instantaneous current draw, and (2) battery voltage discharge curve changes with the residual capacity and current draw value. Based on this battery model, we estimated the battery residual capacity V_r at time $t + \Delta t$ using the following equation:

$$V_r(t + \Delta t) = V_r(t) - I_e \times \frac{\Delta t}{3600},$$

where $V_r(t)$ is the residual capacity at time t and I_e is the equivalent current draw. We estimated I_e through the following equation:

$$I_e = \frac{C_n}{C_r} \times I(t),$$

where C_n (in mA) is the nominal battery capacity, C_r (in mA) is the relative battery capacity and $I(t)$ (in mA) is the instantaneous current. It is worth mentioning that here the value 3600 used for transforming Δt into hours.

3.4. Path loss model

In this work, for the theoretical and experimental analysis, we need a realistic channel model for WSN [27]. In WSN, where the separation distance between any pair of nodes generally varies from 2 m to 100 m, the researchers have shown through empirical studies that the log-normal path loss model provides a realistic assessment of communication characteristics [46]. Further, they have shown that one can use this model for both large and small coverage systems as well as provides more accurate path loss model than Rayleigh and Nakagami. Motivated by [27,46], in this paper, we use a log-normal path loss model to describe the channel between two CHs and also between CH and the sink. In this channel model, the received power, P_r (in dB), for a transmitter–receiver separation distance l is given by:

$$P_r(l) = P_o - PL(l_0) - 10\eta \log_{10} \left(\frac{l}{l_0} \right) + \mathcal{X}_\sigma, \quad (1)$$

where P_o is the output power, $PL(l_0)$ is the power decay of the close-in distance l_0 , η is the path loss exponent ($2 \leq \eta \leq 6$),

$\mathcal{X}_\sigma$ is a Gaussian random variable with mean 0 and variance σ^2 (due to multipath effects). Since, $\mathcal{X}_\sigma$ is random, correct reception of a signal can be guaranteed only when it is represented on a probabilistic basis. Accordingly, in our work, reliable reception of a signal is represented as $\Pr\{e_{ri} \geq \tau\} \geq \delta_r$, where e_{ri} is the energy of the received signal, τ is a predefined energy threshold and δ_r is the required link reliability.

In log-normal path loss model, for a transmitter–receiver separation distance l , the Signal-to-Noise-Ratio (SNR), γ_{SNR} (in dB), at the receiver end is a random variable. From (1), we can calculate γ_{SNR} as:

$$\begin{aligned}\gamma_{SNR} &= P_r(l) - P_\eta \\ &= P_o - PL(l_0) - 10\eta \log_{10} \left(\frac{l}{l_0} \right) + \mathcal{X}_\sigma - P_\eta \\ &= \mathcal{X}_\sigma(\mu(l), \sigma),\end{aligned}$$

where P_η is the noise power and $\mu(l)$ is given as:

$$\mu(l) = P_o - PL(l_0) - 10\eta \log_{10} \left(\frac{l}{l_0} \right) - P_\eta.$$

3.5. Network lifetime model

In the existing literature, various network lifetime models, e.g., event driven (or, on-demand in our case) and time driven models have been used to determine the lifetimes of WSNs. In on-demand based model, the pre-processor remains always in on mode, whereas the node remains in a sleep mode. The node switches to the awake state if and only if an event is detected. On the contrary, in time driven model, the duty cycle of a node describes the awake time or duty period. Generally, most of the time, the node remains in the sleeping mode and it continues until the node wake-up timer expires. Since in this work cluster formation is triggered by the event or on-demand basis, hence, we assume one of the most popular event driven based network lifetime models. In particular, similar to [12,33], we define the network lifetime through a Markov chain. Generally, five different states, i.e., awake, sleep, monitor, processing and communication are used to define lifetime using Markov chain. According to this model, network lifetime is defined as the time required to reach a state, say $s_{0,0}$ from the initial state, say $s_{4,0}$ and it is calculated as:

$$L(s_{p,q}) = S_{oj}(s_{p,q}) + \sum_{s_{p,q} \rightarrow s_{p',q'}} P[s_{p,q}, s_{p',q'}] L(s_{p',q'}),$$

where $S_{oj}(s_{p,q})$ is the sojourn time in state $s_{p,q}$ and $P[s_{p,q}, s_{p',q'}]$ is the state transition probability from state $s_{p,q}$ to state $s_{p',q'}$, $\forall p, 0 \leq p \leq 4$ and $\forall q, 0 \leq q \leq 4$. We refer the reader to [12,33] for a more detailed discussion on the network lifetime model based on Markov chain.

4. Analysis on network lifetime maximization

In this section, in contrast to the existing heuristic load balancing based clustering technique, we investigate the clustering technique based on an analytical approach to maximize network lifetime. In our analysis, we consider deterministic topology model, where nodes are arbitrarily placed, however, their locations are known. With out loss of generality, during our investigation, we assume that a node consumes energy for data transmission and reception. This is due to the fact that in TeloB node [34], the energy consumption is dominated by the wireless transmissions and receptions. However, during experiment (see Section 7), we consider a node consumes energy for sensing, computing, idle in addition to transmission and reception.

Let c_i and u_i be the average intra- and inter-cluster traffic load carried by the CHs of i th slice, respectively, where $i = 1, \dots, N$. According to our network model, c_i and u_i are denoted by $\pi(r_i^2 - r_{i-1}^2)\rho n\phi/2\pi$ and $\pi(r_N^2 - r_i^2)\rho n\phi/2\pi$, respectively. Note that by network model, the number of CHs in the i th slice is approximately given by $\frac{2r_i}{r_i - r_{i-1}} \frac{\phi}{2\pi}$. Let E_i^{jk} be the expected energy consumed by CH j in the i th slice for transmitting all of its traffic to the one-hop neighbor CH k . Therefore, the expected energy consumed by the CH j in the i th slice is given as:

$$\begin{aligned}E_i^{jk} &= \frac{(c_i + u_i)}{2\phi r_i/2\pi(r_i - r_{i-1})} (e_{tx} + e_{rx}) \\ &= \frac{[\pi(r_i^2 - r_{i-1}^2)\rho n\phi/2\pi] + [\pi(r_N^2 - r_i^2)\rho n\phi/2\pi]}{2\phi r_i/2\pi(r_i - r_{i-1})} (e_{tx} + e_{rx}) \\ &= \frac{\pi(r_N^2 - r_{i-1}^2)(r_i - r_{i-1})}{2r_i} \rho n(e_{tx} + e_{rx}).\end{aligned}\quad (2)$$

It is worth noting that the expected energy consumed by a CH in the N th slice can be calculated applying (2) and using the standard convention that a sum of terms is zero if its lower index is greater than its upper bound.

Let e_{ti} be the over-the-air RF energy consumed when transmitting one bit from CH j in the i th slice to CH k located at distance R_{jk} . It is worth mentioning that e_{ti} is a function of the distance between two communicating CHs. So, (2) can be rewritten as:

$$E_i^{jk} = \frac{\pi(r_N^2 - r_{i-1}^2)(r_i - r_{i-1})}{2r_i} \rho n(e_{tx} + e_{rx} + e_{ti}).\quad (3)$$

From channel model (1), the over-the-air RF energy consumed for receiving one bit, e_{ri} , is given as:

$$e_{ri} = e_{ti} L(l_0) \left(\frac{R_{jk}}{l_0} \right)^{-\eta} \mathcal{X}_\sigma.$$

For a log-normal path loss model, the link reliability requirement can be expressed as:

$$\begin{aligned}\delta_r &= \Pr\{e_{ri} \geq \tau\} \\ &= \Pr\left\{\mathcal{X}_\sigma \geq \frac{\tau}{e_{ti} L(l_0) \left(\frac{R_{jk}}{l_0} \right)^\eta}\right\} \\ &= e^{\frac{-\tau}{e_{ti} L(l_0) \left(\frac{R_{jk}}{l_0} \right)^\eta}}.\end{aligned}$$

From the above expression, we can express e_{ti} as:

$$e_{ti} = \frac{-\tau}{L(l_0) \log \delta_r} \left(\frac{R_{jk}}{l_0} \right)^\eta.$$

According to our routing model, let h be the maximum number of links of an end-to-end path. Hence, to generate the constraint on path reliability, δ_p , the minimum link reliability must be:

$$\delta_r = \delta_p^{\frac{1}{h}}.$$

Therefore,

$$e_{ti} = \frac{-h\tau}{L(l_0) \log \delta_p} \left(\frac{R_{jk}}{l_0} \right)^\eta = \beta R_{jk}^\eta,$$

where $\beta = -h\tau/L(l_0)l_0^\eta \log \delta_p$ and it is a constant. Consequently, the energy consumed by CH j in the i th slice, given in (3), can be rewritten as:

$$E_i^{jk} = \frac{\pi(r_N^2 - r_{i-1}^2)(r_i - r_{i-1})}{2r_i} \rho n(e_{tx} + e_{rx} + \beta R_{jk}^\eta).\quad (4)$$

In order to maximize the network lifetime, our objective is to compute $\{r_i\}$, where $i = 1, \dots, N$, for minimizing the maximum energy consumption rate among all the CHs. The calculation procedure can be formulated by the following optimization

problem:

$$\max \min \left\{ \frac{e_{in}}{E_1^{jk}}, \dots, \frac{e_{in}}{E_N^{j'k'}} \right\}$$

where e_{in} is the initial energy of a CH. Introducing an auxiliary variable t , where $t \leq \max \{E_1^{jk}, \dots, E_N^{j'k'}\}$, the above objective function can be transformed into the following optimization problem:

$$\min_{r_1, r_2, \dots, r_N} t \quad (5)$$

subject to

$$\left[\pi \left(r_i^2 - r_{i-1}^2 + \sum_{h=i}^N (r_{h+1}^2 - r_h^2) \right) \rho n \frac{\phi}{2\pi} \right] - \pi \sum_{h=i-1}^N (r_{h+1}^2 - r_h^2) \rho n \frac{\phi}{2\pi} = 0, \forall 1 \leq i \leq N, \quad (6)$$

$$t \left[\frac{\pi (r_N^2 - r_{i-1}^2) (r_i - r_{i-1})}{2r_i} \rho n (e_{tx} + e_{rx} + \beta R_{jk}^\eta) \right] \leq e_{in}, \quad (7)$$

$$\pi (r_i^2 - r_{i-1}^2) \rho n \frac{\phi}{2\pi} > 0, \frac{\phi r_i}{2\pi (r_i - r_{i-1})} > 0. \quad (8)$$

The constraint (6) guarantees inter-cluster flow preservation, i.e., all data packets generated at or forwarded to a slice are pushed out of it. The constraint (7) specifies that in the lifetime t no node consumes more than its available energy e_{in} . The constraint (8) specifies that the intra-cluster data flow and number of clusters in a slice is non-negative. If we examine both the objective function and the constraints, the problem is convex. As per the convex optimization theorem [5], our defined problem has a unique global optimal solution. To obtain optimal solution, we used a widely accepted efficient method, i.e., ADMM. In ADMM, a convex optimization problem is solved distributively, where the problem is decomposed into several subproblems. In view of the decentralized nature of WSN, we chose ADMM in this work to solve convex optimization problem. Apart from the decentralized nature, another major advantage of ADMM is the higher convergence rate compared to other existing methods, e.g., subgradient method [41]. Thus, using ADMM, we can solve the above problem efficiently. To solve (5)–(8), one needs to know network size and network information, e.g., packet size. It is easy to check that the nature of the objective function is convex. It is worth reminding that we consider the nodes in a slice report data to the sink in shortest path, i.e., lowest ETX count (see Section 3.2), hence, the derived lifetime of CH provides the upper bound of the network lifetime.

5. The proposed clustering technique: OPTIC

The proposed OPTIC strategy consists of evaluation of optimal cluster radius, node deployment, cluster setup, and optimal data gathering and routing path formation phases. The evaluation of optimal clustering radius, and optimal data gathering and routing path formation are already discussed in Sections 4 and 3.2, respectively. This section describes node deployment (Section 5.1), cluster setup (Section 5.2) and data collection (Section 5.3) phases.

5.1. Deployment

In this paper, we consider the stochastic deployment, where nodes are dropped by helicopter in battlefields or unfriendly environments. In particular, we consider that nodes are uniformly

and independently distributed in different slices around the sink. The probability f_a that a point is covered by nodes is:

$$f_a = 1 - e^{-\rho \pi R_s^2}, \quad (9)$$

where R_s is the sensing range of a node.

Our objective is to deploy nodes in the ring sectors based network area (Fig. 2). To achieve this goal, we propose to decompose the i th slice into multiple circular domains so that each circular domain is regarded as a cluster. Afterwards, the nodes are uniformly and independently distributed within each cluster using Eq. (9). After deployment of nodes, without loss of generality, we assume that the WSN passes thorough a training process where each node learns about the slice and the region to which it belongs. Recently, a number of training protocols were proposed. In this work, we assume a training protocol as proposed in [3].

5.2. Cluster setup

The information related to the radius of clusters and number of clusters in i th slice can be obtained by solving (5)–(8). Unlike time driven CH selection, in OPTIC, we propose a novel on-demand based CH selection strategy to reduce the extra message overhead during cluster formation. All the nodes in the network are eligible to become CHs and they are selected through a proposed novel process. This CH selection process is divided into two sub-phases: CH selection and cluster building.

5.2.1. CH selection sub-phase

The proposed CH selection strategy is triggered once an event of interest (e.g., detection of an intruder) is detected. Node(s) that detects an event (hereafter referred to as detected node(s)) only takes part in the CH selection and cluster building processes. Rest of the nodes remain in active mode for detecting future event(s) and/or forwarding data packets. Upon detecting an event, a detected node poses itself as acting CH and broadcast *Hello*(ID, E_i , E_r) message around the cluster radius to rest of the detected nodes, where ID, E_i and E_r denote the node identification number, expected energy consumption rate and the residual energy level of the detected node, respectively. It is worth mentioning that the cluster radius is determined by solving our optimization problem given in Section 4. However, during inter-cluster communication a CH uses higher power level so that the transmitted data packet can reach at least two or more cluster diameters. Henceforth, for the sake of simplicity, we call “rest of the detected nodes within cluster radius” as neighbors of the acting CH. After a detected node receives the *Hello* message, all the neighbors send *Hello_Ack*(ID, E_i , E_r) message to the acting CH. The acting CH calculates average energy consumption (AE_i) and average residual energy (AE_r) based on the number of *Hello_Ack*(ID, E_i , E_r) messages, N_e , received from the neighboring nodes. The acting CH nominates the k th node from the neighbors within the cluster radius as new CH whose E_i is minimum, i.e., $AE_i > \dots > E_i^{\min}$, and E_r is maximum, i.e., $AE_r < \dots < E_r^{\max}$. If there is more than one node with the same ($E_i^{\min}$, $E_r^{\max}$), one of them is chosen randomly. The (AE_i) and (AE_r) are calculated as follows:

$$AE_i = \sum_i \frac{E_i}{N_e}, AE_r = \sum_i \frac{E_r}{N_e}.$$

5.2.2. State information of a node

In OPTIC, the node may be in any of the following four states, namely: ON, acting CH (ACH), CH and MN. For the sake of convenience, the state diagram of the nodes in the CH selection sub-phase is shown in Fig. 4. After node deployment phase, all the nodes are considered to be in ON state. If an event is detected

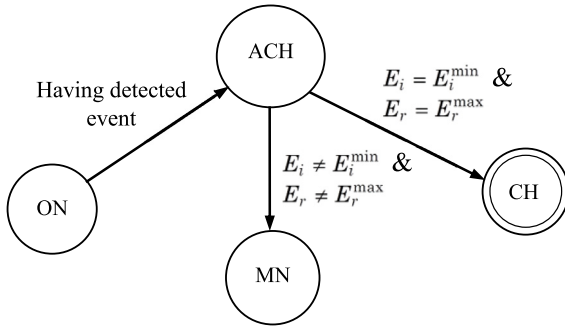


Fig. 4. The state diagram of a node in the CH selection sub-phase of OPTIC.

Algorithm 1: Cluster Setup Phase

```

// Upon detecting an event by a node j
if IsNode(j) = TRUE then
    calculate  $E_i$ ;
    calculate  $E_r$ ;
    send Hello(ID,  $E_i$ ,  $E_r$ ) to all neighbours;
    receive Hello_Ack(ID,  $E_i$ ,  $E_r$ ) from all neighbours;
    calculate  $AE_i$ ;
    calculate  $AE_r$ ;
    if  $AE_i > \dots > E_i^{\min}$  and  $AE_r < \dots < E_r^{\max}$  then
        advertise Head_Msg;
    else
        send Join_Msg to nearest CH ;
        receive Join_Msg from all neighbours ;
    end
end
end

```

by the nodes, they pose themselves as acting CHs and enter ACH state. The acting CHs participate in the CH selection competition. An acting CH could become a CH if it passes the expected energy consumption rate and residual energy condition, otherwise it enters MN state. In the next round of CH selection sub-phase again all nodes enter the ON state.

5.2.3. Cluster building sub-phase

After CH selection, the newly selected CH broadcasts a selection message, *Head_Msg*, within the cluster radius with communication range R_i . The *Head_Msg* message contains node ID, present energy consumption rate, and residual energy. After receiving *Head_Msg* message, a node responds to the newly selected CH by sending *Join_Msg* message. If a node receives more than one *Head_Msg* message from its neighboring CH, it will choose to join the nearest CH by sending *Join_Msg* message. Here, we assume that a node calculates the distances between itself and neighboring CHs based on received signal strength. Detailed discussion of distance calculation based on received signal strength is out of the scope of this work. For detailed discussion, the readers may refer to [1,40]. The CH on receiving *Join_Msg* message acknowledges the joined node by sending *Head_Acpt* message and designate that joined node as MN. As a CH can regulate its communication range, thus no sensor in the network is left away from the cluster framework. Algorithm 1 illustrates the action performed by the node in the cluster setup phase.

5.3. Data collection

After the cluster formation, each CH in the network generates a TDMA schedule for its MNs. In TDMA, the available bandwidth is normally divided into frames and each frame is divided into

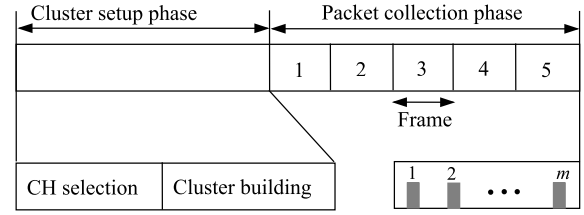


Fig. 5. Time line of OPTIC.

time slots. The length of the frame depends on the number of MNs and time allotted for each MN for data transmission. In this work, we assumed that all the MNs are assigned the same amount of time slots and the length of the frame is decided based on the number of received *Join_Msg* messages. After receiving the schedule, each MN sends its sensed data to its CH by following the TDMA schedule. MNs are awake only during their allotted time slot(s) for transmission and sleep for rest of the time. However, a CH remains in an active/awake state to receive sensory data from its MNs. Since the data gathered by the CHs are highly correlated, hence each CH aggregates the received data packets into a single data packet. Finally, at the end of the frame, the aggregated data packets are sent to the sink through intermediate CHs. In case, if there is no CHs, then intermediate nodes from the optimal DAG (see Section 3.2) are chosen to forward the data packets. A typical time line of operation of our proposed algorithm is illustrated in Fig. 5.

In WSN, due to the broadcast nature of wireless links, when a MN transmits a data packet to its CH, in some cases, the data packet may reach the nearby CHs. To avoid this inter-cluster interference, in data collection phase, we assumed that the MN uses spread code. Particularly, when the MNs send the data to their respective CHs, the spread code of the concerned cluster is attached to avoid inter-cluster interference.

6. Analysis on OPTIC

This section analyzes the message and time complexities of OPTIC. In addition, the CH selection policy of OPTIC is analyzed and shown that it is optimal.

6.1. Message complexity

Let T be the total number of nodes deployed in the network and T^p be the number of nodes participating in the CH selection sub-phase. After detecting an event, a node poses itself as acting CH and broadcast *Hello*(ID, E_i , E_r) message ($O(1)$). In response to the *Hello*(ID, E_i , E_r) message, $(T^p - 1)$ nodes respond back with *Hello_Ack*(ID, E_i , E_r) messages to acting CH which will be $(T^p - 1)$ messages ($O(1)$). Finally, a selected CH responds back to $(T^p - 1)$ nodes with *Head_Msg* message and $(T^p - 1)$ nodes, in turn, respond by sending *Join_Msg* messages ($O(1)$). Therefore, the total number of messages exchanged in the network is given by:

$$1 + 3(T^p - 1) = 3T^p - 2 = O(T^p).$$

It is worth noting from the above discussion that, in an iteration of OPTIC, the number of messages exchanged in the network is $(3T^p - 2)$, i.e., $O(T^p)$. Alternatively, in OPTIC, the order of control message transmissions is $O(1)$ per node, and $O(T^p)$ in the network which is similar to most of the popular clustering algorithms like Younis and Fahmy [45], Ye et al. [44] and Baranidharan and Santhi [2]. This is one of the desirable features in most of the WSNs applications.

6.2. Time complexity

In OPTIC, during cluster setup phase, an acting CH needs to calculate E_i and E_r (lines 3–4). The calculation of E_i and E_r takes $O(1)$ time. In addition, an acting CH needs to calculate AE_i and AE_r (lines 7–8) based on the information received from the nodes which participated during cluster setup phase. Similar to previous calculation, the calculation of AE_i and AE_r takes $O(1)$ time. During clustering, an acting CH can accommodate $(T^p - 1)$ nodes as MN, hence receiving messages from neighboring nodes (line 6) takes $O(T^p)$. Since the calculation of AE_i and AE_r takes $O(1)$ time, as a consequence, line 10 requires $O(1)$ time. In line 13, similar to line 6, receiving messages from neighboring nodes takes $O(T^p)$ time. Since T^p number of nodes are participating in the clustering phase and all the other operations (except mentioned earlier) within the algorithm take $O(1)$ time, the total time complexity of OPTIC is obtained as $O(T^p)$.

6.3. Optimality

This section analyses the optimality of CH selection policy in OPTIC. Let $RE_i^m(\lambda)$ and $RE_i^k(\lambda)$ be the residual energy of node m and k , respectively in λ th round located in a CH of i th slice. Further, let E_{CH} and E_{MN} be the energy consumption by a CH and MN, respectively in each round. To prove the optimality of CH selection policy, we first find the upper and lower bounds of energy levels of a node in any round. Subsequently, we show that the proposed CH selection policy P is optimal and provides maximum network lifetime time, L^{max} .

Theorem 1. For given two rounds t and $t + \lambda$, if $RE_i^k(t) = RE_i^m(t)$ then $|RE_i^k(t + \lambda) - RE_i^m(t + \lambda)| \leq E_{CH} - E_{MN}$, where $1 \leq t \leq L^{max}$ and $1 \leq (t + \lambda) \leq L^{max}$.

Proof. Since the proof is similar to that of [Cheng et al. [8], Theorem 1], we do not repeat it here. $\square$

Theorem 2. For given two rounds t and $t + \lambda$, if $RE_i^k(t) > RE_i^m(t)$ then $RE_i^k(t + \lambda) - RE_i^m(t + \lambda) \leq E_{CH} - E_{MN}$, where $1 \leq t \leq L^{max}$ and $1 \leq (t + \lambda) \leq L^{max}$.

Proof. Since the proof is similar to that of [Cheng et al. [8], Theorem 2], we do not repeat it here. $\square$

Based on Theorems 1 and 2, we can now prove the optimality of the CH selection policy of OPTIC.

Theorem 3. The proposed CH selection policy P is the optimal selection policy to maximize the network lifetime.

Proof. Let us assume that the proposed CH selection policy P is not optimal. Then, there exists an optimal CH selection policy P^* for which one can obtain maximum network lifetime L^* , where $L^* > L^{max}$. Alternately, P is neither able to schedule a node m as a CH nor as a MN at round $L' + 1$, however, P^* can do it. It is worth noting from the assumption that the total energy of node m ranges between $E_{CH} \cdot L'$ and $E_{MN} \cdot L'$. Based on the initial energy, we consider the following possible cases:

- **Case I:** $RE_i^m(1) = E_{MN} \cdot L'$. In this case, node m serves as MN during the interval of $[1, L']$ in policy P . Therefore, it is impossible to schedule node m at round $L' + 1$, since the remaining energy of node m is only adequate to act as a MN L' times.

- **Case II:** $E_{MN} \cdot L' \leq RE_i^m(1) \leq E_{CH} \cdot L'$. Since a node under policy remains active or runs out of energy at round L^{max} , therefore node m should work as a CH at least once during the interval of $[1, L^{max}]$. Now, to conserve energy so that node m stays alive at round $L^{max} + 1$, policy P^* needs to decrease the number of times in which node m acted as CH. It is worth noting that if node m can conserve at least E_{CH} amount of energy then it can serve at round $L^{max} + 1$. Let us consider that policy P selects node m as CH and node k as MN at λ th round. To conserve energy of amount $(E_{CH} - E_{MN})$ for node m , the policy P^* should change the role of node m and k at λ th round. According to the proposed policy P , we know that if node m is acting as CH and node k is acting as MN at λ th round, then $RE_i^m(\lambda) \geq RE_i^k(\lambda)$. Based on Theorems 1 and 2, at the end of λ th round, we know that the difference in remaining energy between node m and k is $(E_{CH} - E_{MN})$. Therefore, if the policy P^* interchanges the role of node m and k at λ th round, then node k will be drained out at round L^{max} . $\square$

Therefore, from the above discussion, we can conclude that the proposed policy P is optimal and works efficiently to prolong the network lifetime.

7. Experimental evaluation

In this section, we compared the performance of the OPTIC with two state-of-the-art unequal clustering algorithms, namely, COCA [23] and UCR [7]. Further, to show the efficacy of our algorithm, we compare OPTIC with a state-of-the-art non-clustering approach, called ORR [38], where a node takes forwarding decision based on its residual energy. A brief description of ORR is as follows:

- **ORR.** In ORR, periodically, the sink gathers the information about the current energy status and neighbor information of each node. Based on these information, the sink forms a forwarder set and distributes the forwarder set information to each node. A node selects a forwarder node from the forwarder set having maximum residual energy. The same process is repeated until the data reaches the sink.

To measure the performance of all the competing schemes, we use a real-world publicly-available environmental monitoring dataset [30].

7.1. Experimental setup

In this work, for performing the experiments, we used version 1.3 of Castalia [4], a specialized simulator for WSN. Castalia simulator not only provides bundled support for the CC2420 radio transceiver, but also provides realistic: (i) radio models, e.g., received signal strength indicator, carrier sensing, (ii) wireless channels, e.g., path loss, interference, and (iii) node behavior, particularly, related to the radio access. In our experiments, the nodes are deployed randomly and uniformly in an area consisting of 6 slices at an angle $\phi = 60^\circ$. We simulate ours and the competing schemes under realistic scenario. In realistic scenario, a node not only consumes energy for transmissions and receptions, but, also consumes energy for sensing, computing and idle. Further, during simulation, each MN and CH runs CTP as routing protocol and TDMA (see Section 5.3) as MAC layer protocol, on top of which we implement all competing schemes. Whereas, during communication between CH and sink, a node runs CSMA/CA as MAC layer protocol for OPTIC, COCA and UCR. On the contrary, in case of ORR, each node runs CSMA/CA as MAC layer protocol and routing protocol as stated earlier. Further, unlike OPTIC, COCA

Table 2
Parameters values used in simulation.

Parameters	Values
Battery power	3.3 V
Radio transceiver receive power	38 mW
Radio transceiver transmit power at 0 dBm	35 mW
Radio transceiver power in idle mode	3 μ W
Microcontroller computing power	41 mW
Sensing power	3 mW
Close-in distance (l_0)	10 m
Path loss exponent (η)	4
Antenna gain (G_t, G_r)	1
Predefined energy threshold (τ)	10^{-17} J
Link reliability (δ_r)	0.99
Wavelength of the carrier signal (ω)	0.125m (2.40 GHz)
R_s	10 m
Packet size (n)	44 bytes
Duration of each round	20 s

and UCR, we consider a rectangle shaped network area while simulating ORR. To make the comparison fair, we deployed the same number of nodes and a sink is deployed at the coordinate (0, 0) for all the schemes. Further, we assume that occurrence of event is independent and identically distributed in the network area and generating an event per minute. During simulation, for all the schemes, we consider that once a node detects an event of interest, it generates and transmits data packets at the rate of 250 kbps (constant bit rate). In addition, we selected data source nodes randomly from the set of nodes. During the experiments, we assumed that a node operated by a non-rechargeable battery consists of two capacitors of 70 F, 2.1 V in a serial connection. It offers a global capacity of 35 F and provides 3.3 V of power supply, i.e., maximum operating voltage of TelosB node [29]. Considering this battery specification, similar to [29], we derive the initial energy budget of each node as 190 J. Further, we set the cut-off voltage of the battery model to 1.5 V (or, 39 J) [29]. All the parameters and their corresponding values used during experiment are based on the TelosB node specification [29,34] and are listed in Table 2. Extensive simulation has been performed with a 95% confidence level and 5% accuracy while plotting the simulation results. To achieve this, we collect experimental results in groups of 200 observations per group. For each group, we calculate a mean out of 200 observations collected. We run at least 5 groups to obtain a minimum of 5 means. Next, we calculate the grand mean and estimate the difference of the grand mean from the true mean with 95% confidence. If the accuracy obtained is greater than 5%, we run more groups and collect more observations until the specified 5% accuracy requirement is achieved.

7.2. Performance metrics

We first measure the impact of training protocol on the performance of OPTIC. To measure the impact of training protocol, we consider convergence time or total time required to complete the training process and energy consumed by each node during the training process as the performance metrics [3]. We next divided the performance evaluation of OPTIC, UCR, COCA and ORR into two stages. First, the energy conservation capability of the schemes is evaluated considering energy balance and network lifetime as performance metrics. Second, network performance of the schemes is measured considering end-to-end delay and throughput as metrics of interest. We considered network lifetime as defined in Section 3.5. Two more parameters, namely, average energy consumption per CH and standard deviation of residual energy among CHs are defined for evaluating the extent of energy balance in the network:

- **Average Energy Consumption Rate per CH (Avg ECR per CH)** is defined as energy consumption by a CH per unit time. This parameter shows how evenly the energy consumption is taking place among the CHs. It is evaluated using (5).
- **Standard Deviation of Residual Energy among CHs (SDRE)** is defined as the deviation of residual energy at the end of network lifetime. This quantity indicates how efficiently a CH utilizes energy, the scarcest resource. The smaller the value of the standard deviation, the better is the capability of utilizing reservoir energy more efficiently.

It is worth mentioning that due to absence of CH in ORR, we measured average energy consumption rate of each forwarder node (or, parent node) instead of avg ECR per CH. In particular, during simulation, for ORR, we assume forwarder node acting as a CH, similar to other competing schemes. Further, we defined the two network performance metrics mentioned earlier as follows:

- **End-to-End Delay** is defined as the difference in time when one data packet is generated at the source node and when this packet is received at the sink [18].
- **Throughput** is defined as the amount of data (in terms of bits) received at the sink per unit time [18].

7.3. Impact of training protocol on OPTIC

In this section, we evaluate the time required to finish the training process of our proposed algorithm, OPTIC. During simulation, we assume that nodes might either wakeup or go to sleep for just a single time slot based on the slice to which they belong. Further, we consider the duration of a node active time as 2 ms based on the fact that lower bound of the length of node active time must be more than the time allowed for both radio startup and shutdown [3]. We notice from Fig. 6(a) that the total time required to converge the training protocol for all the nodes in the network consisting of 6 slices, is around 428 ms. Further, we measure the energy consumption of each node while executing the training protocol and plotted in Fig. 6(b). We notice from Fig. 6(b) that maximum 177 mJ energy consumed by a node for executing training protocol. Since we assume TeloB node of initial energy budget as 190 J, hence, the entire training task consumes around 1/1000 of the total energy budget.

7.4. Energy conservation capability

We conducted two sets of experiments for evaluating the energy conservation capability of the various schemes. One set of experiments measures energy balance in the network and the second one measures optimal network lifetime.

7.4.1. Energy balance

In this section, we evaluate the energy balance of all schemes in terms of two parameters defined in Section 7.2.

(a) **Average ECR per CH:** Fig. 7 shows avg ECR per CH of different competing schemes. The avg ECR per CH is calculated using (4) for OPTIC. Here, we observe that the plot of avg ECR per CH is fairly constant for all slices in OPTIC. For example, avg ECR per CH is 47.95 mJ/sec. It is worth noting that, in OPTIC, avg ECR per CH is significantly less compared to all the competing schemes. Particularly, the plot shows that OPTIC outperforms the other schemes. Further, we notice that, among all the schemes, avg ECR per CH is highest in ORR and lowest in OPTIC. For example, avg ECR per CH of OPTIC is 16.21%, 33.23% and 54.61% less than that of COCA, UCR and ORR, respectively. This is due to the fact that, in OPTIC, the CH selection process is dynamic and energy efficient compared to COCA and UCR. Further, we notice that CHs in 1st slice of both COCA, UCR and ORR have

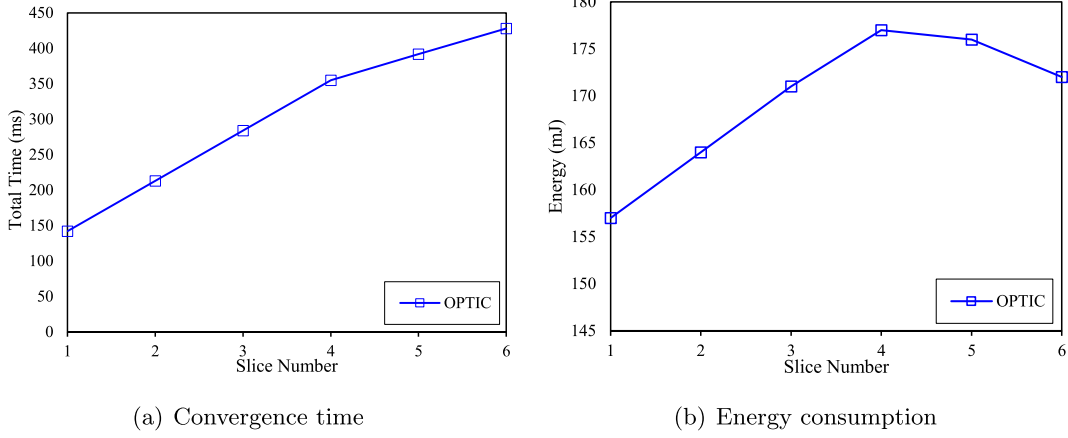


Fig. 6. Convergence time and energy consumption of training protocol.

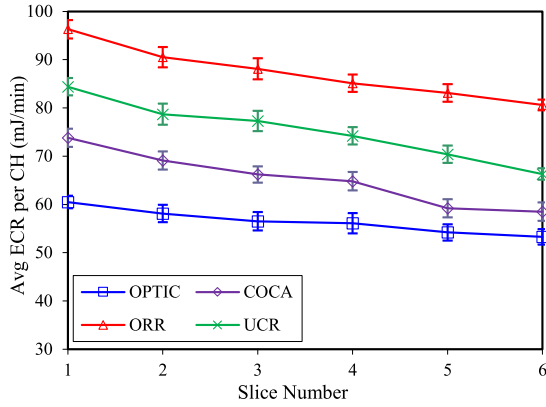


Fig. 7. Average energy consumption rate per CH located at different slices.

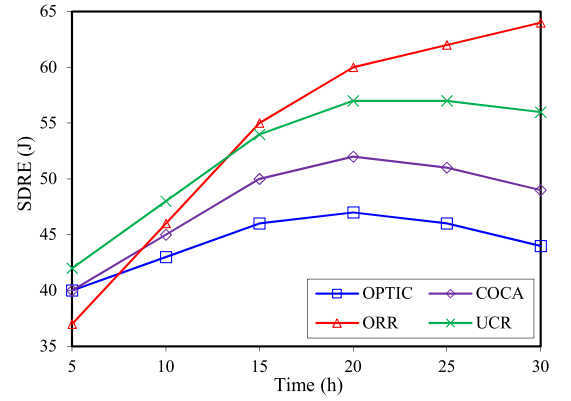


Fig. 8. Standard deviation of residual energy among CHs.

maximum avg ECR per CH, whereas CHs in the farthest slice have lowest avg ECR per CH. In particular, ORR shows the worst performance among the competing schemes. It is due to the fact that ORR is a non-clustering scheme which does not maintain a topology and the nodes located near to the sink handle more data pressure. Therefore, nodes located in slices nearer to the sink get their reserved energy, drained out quickly in comparison to nodes located in slices far away from the sink. Interestingly, avg ECR per CH plot of OPTIC is relatively steady throughout the slices. This indicates that OPTIC is more energy balanced compared to COCA, UCR and ORR.

(b) *SDRE*: Fig. 8 illustrates the comparison of OPTIC with COCA, UCR and ORR considering standard deviation of residual energy as a performance metric. We observe that there are two distinct trends among the schemes. When time increases up to 15 h, the SDRE of OPTIC and COCA increases; it then decreases for time > 20 h. In contrast, for both UCR and ORR, the SDRE increases with respect to time. Interestingly, initially, the performance of ORR is the best among the competing schemes. The possible reason is that nodes located near to the sink bear less forwarding load initially, hence, judicious energy consumption take place among the nodes. However, as time progresses, with the increase of forwarding load, imbalance energy consumption take place, resulting in higher SDRE with increasing time. On the contrary, the plot of UCR indicates that although initially it utilizes energy to a greater extent, however, as the time progresses, it fails to utilize energy judiciously. One possible reason is that energy consumption by the CHs in the UCR is still balanced. Clearly, it indicates that both UCR and ORR fail to utilize reserved energy efficiently, resulting

in significant amount of energy wastage. On the contrary, OPTIC has decreasing trends of SDRE with time, which indicates that all the CHs in different slices utilize reservoir energy more efficiently than all the competing schemes. Further, we noticed that OPTIC leaves less residual energy compared to COCA, UCR and ORR. In particular, OPTIC has better capability of utilizing reservoir energy more efficiently than both the competing schemes. Therefore, we can say that OPTIC is energy balanced and utilizes energy, the scarcest resource, more efficiently than COCA, UCR and ORR.

7.4.2. Network lifetime

We conduct two sets of experiments for evaluating the network lifetime. The first set measures network lifetime for different slices, whereas, the other set measures network lifetime by varying number of nodes.

Fig. 9 shows the network lifetime of all the competing schemes. We observed from the plot that OPTIC outperforms the other schemes. In particular, it is observed from Fig. 9 that the network lifetime of OPTIC is 17.53%, 33.09% and 54.36% more than that of COCA, UCR and ORR, respectively. Moreover, in OPTIC, nearly flat nature of the plot ensures that in all the slices, network lifetime terminates in more or less the same time as compared to COCA, UCR and ORR. It is due to the fact that, compared to the other schemes, the clusters are formed dynamically in OPTIC, i.e., based on the occurrence of events. Further, in OPTIC, clusters are formed in such a manner that data forwarding loads are evenly distributed among the CHs, resulting in balanced energy consumption. Alternately, in OPTIC, energy consumption for CH-to-CH relay is much lesser than COCA, UCR and ORR. This ensures

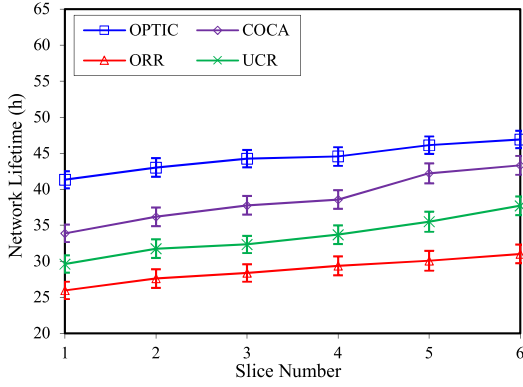


Fig. 9. Network lifetime for different slices.

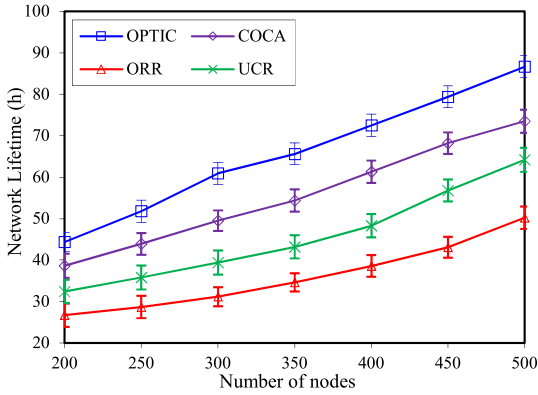


Fig. 10. Network lifetime measurement by varying number of deployed nodes.

that energy is balanced in OPTIC to a greater extent than both the competent schemes.

In Fig. 10, we observe that, as the number of deployed nodes increases, the network lifetime of all the schemes increases. It is expected that, even without energy balancing, data routing paths from CHs to the sink are more diverged, resulting in less energy consumption among CHs. More interestingly, we noticed that the rate of increase of OPTIC is much faster than COCA, UCR and ORR. Specifically, the network lifetime of OPTIC is 19.05%, 44.08% and 82.21% more than that of COCA, UCR and ORR, respectively. It is due to the fact that, in OPTIC, the clusters are formed using optimal radius profiling. In addition, clusters are formed dynamically throughout the network, i.e., based on the occurrence of an event. We further noticed that the performance of the ORR is the worst among the competing schemes. One possible reason is that ORR is a non-clustering scheme which does not maintain a network topology, leading to uneven energy consumption among the forwarder nodes.

7.5. Evaluation of network performance

This section is dedicated to observe network performance. Two sets of experiments are conducted for evaluating the network performance. The first set measures end-to-end delay while the other set measures throughput.

7.5.1. Throughput

Fig. 11 shows data throughput of OPTIC, COCA, UCR and ORR as measured at the sink under the multi-hop benchmark. It is observed from Fig. 11 that, initially, the throughput for all the schemes steadily increases, however, after certain time interval it becomes steady with increasing time. More interestingly,

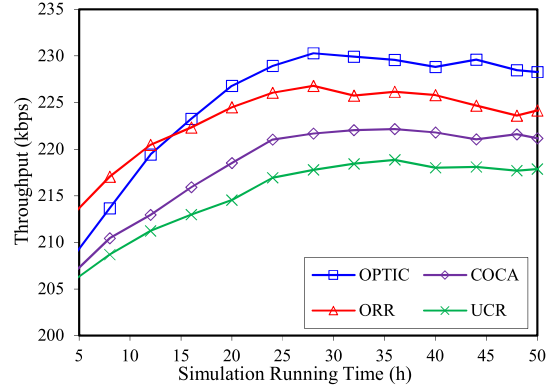


Fig. 11. Throughput comparison over simulation running time.

ORR shows more throughput initially. It is due to the fact that forwarder nodes in ORR are chosen opportunistically, as a consequence more data delivery paths from a node to the sink. However, as time progresses, congestion occurs, resulting in packet loss and degrading the overall throughput. We also observed that, except the first 15 h, OPTIC outperform the competing schemes. The reason is that, in OPTIC, based on inter and intra cluster data traffic, optimal number of clusters are derived in each slice. As a result traffic intensity is judiciously handled by the CHs which reduces the possibility of packet collision and congestion. We noticed from the plot that overall the average throughput of OPTIC is higher than all the competing schemes. In particular, the average throughput of OPTIC is 11.02%, 7.11% and 14.62% more than that of COCA, UCR and ORR, respectively.

7.5.2. End-to-end delay

We also analyzed the end-to-end delays of our scheme along with the competing schemes and results are plotted in Fig. 12. It is observed from the plot that, irrespective of schemes, the end-to-end delay increases with the increase in distance of slices from the sink. It is obvious because of the packets traversing from the nodes located in the slices far away from the sink involve higher propagation time than packets coming from near to the sink. Further, except UCR, we observe that the end-to-end delay of other three schemes is almost overlapping with each other. Interestingly, the end-to-end delay of ORR is slightly better than the OPTIC and COCA. It is expected as non-clustering ORR has more diverge paths from nodes to the sink. Moreover, the differences in end-to-end delays among the schemes are due to the deployment of different number of nodes in a particular slice, causing different magnitude of congestion and resulting in different end-to-end delays. Finally, we observed that the average end-to-end delay of OPTIC is 16.67% and 41.65% less than that of COCA and UCR, respectively.

8. Conclusion

In this work, we addressed the problem of lifetime maximization in WSNs by devising a novel dynamic clustering. We first analyzed the network lifetime maximization problem by balancing the energy consumption among cluster heads. Based on the analysis, we provided an optimal clustering technique, in which the optimal clustering radius of each level is computed using ADMM. Next, we developed OPTIC, an on-demand, distributed optimal clustering algorithm, where CHs are selected dynamically for efficient energy management among the nodes. The use of on-demand based clustering mechanism reduces clustering overhead because no clusters are maintained unless they

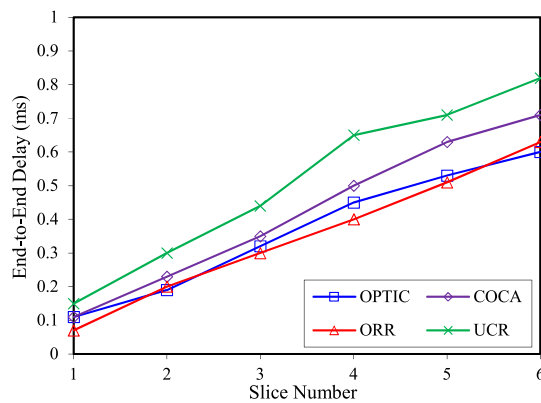


Fig. 12. End-to-end delay for packets.

are needed. Experimental results clearly demonstrate that our proposed scheme can significantly increase the network lifetime without compromising the network performance metrics like throughput, end-to-end delay compared to a non-clustering [38] and two clustering solutions [7,23] in the state-of-the-art.

As future work, we seek to investigate the lifetime optimization problem in WSNs considering more general network architecture. Further, we envision to measure the performance of the designed clustering algorithm in publicly available real-world testbeds such as ORBIT.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Amrita Ghosal: Conceptualization, Methodology, Formal analysis, Writing - original draft, Data curation. **Subir Halder:** Methodology, Software, Validation, Formal analysis, Writing - original draft. **Sajal K. Das:** Supervision, Writing - review & editing.

References

- [1] F. Bandiera, A. Coluccia, G. Ricci, A cognitive algorithm for received signal strength based localization, *IEEE Trans. Signal Process.* 63 (7) (2015) 1726–1736.
- [2] B. Baranidharan, B. Santhi, DUCF: distributed load balancing unequal clustering in wireless sensor networks using fuzzy approach, *Appl. Soft Comput.* 40 (2016) 495–506.
- [3] F. Barsi, A.A. Bertossi, C. Lavault, A. Navarra, S. Olariu, M.C. Pinotti, V. Ravelomanana, Efficient location training protocols for heterogeneous sensor and actor networks, *IEEE Trans. Mobile Comput.* 10 (3) (2011) 377–391.
- [4] A. Boulis, Castalia User Manual (version 1.3), 2011, (online), <http://cpham.perso.univ-pau.fr/ENSEIGNEMENT/PAU-UPPA/INGRES-M1/Castalia%20-%20User%20Manual.pdf>.
- [5] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, U.K., 2004.
- [6] S.M. Bozorgi, A.M. Bidgoli, HEEC: a hybrid unequal energy efficient clustering for wireless sensor networks, *Wirel. Netw.* (2019) 1–22, <http://dx.doi.org/10.1007/s11276-018-1744-x>.
- [7] G. Chen, C. Li, M. Ye, J. Wu, An unequal cluster-based routing protocol in wireless sensor networks, *Wirel. Netw.* 15 (2) (2009) 193–207.
- [8] B.C. Cheng, H.H. Yeh, P.H. Hsu, Schedulability analysis for hard network lifetime wireless sensor networks with high energy first clustering, *IEEE Trans. Reliab.* 60 (3) (2011) 675–688.
- [9] U. Colesanti, S. Santini, A performance evaluation of the collection tree protocol based on its implementation for the castalia wireless sensor networks simulator, Technical report, Swiss Federal Institute of Technology Zurich, Department of Computer Science, 2010, p. 681.
- [10] D.S. De-Couto, D. Aguayo, J. Bicket, R. Morris, A high-throughput path metric for multi-hop wireless routing, *Wirel. Netw.* 11 (4) (2005) 419–434.
- [11] W. Dron, S. Duquenois, T. Voigt, K. Hachicha, P. Garda, An emulation-based method for lifetime estimation of wireless sensor networks, in: *Proc. of IEEE International Conference on Distributed Computing in Sensor Systems, DCOSS*, 2014, pp. 241–248.
- [12] C. Eris, M. Saimler, V.C. Gungor, E. Fadel, I.F. Akyildiz, Lifetime analysis of wireless sensor nodes in different smart grid environments, *Wirel. Netw.* 20 (7) (2014) 2053–2062.
- [13] R. Fonseca, O. Gnawali, K. Jamieson, S. Kim, P. Levis, A. Woo, TinyOS Enhancement proposal (TEP) 123: The collection tree protocol (CTP), 2006, (online), www.tinyos.net/tinyos-2.x/doc/pdf/tep123.pdf.
- [14] A. Ghosal, S. Halder, Lifetime optimizing clustering structure using archimedes' spiral based deployment in wsns, *IEEE Syst. J.* 11 (2) (2017) 1039–1048.
- [15] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, P. Levis, Collection tree protocol, in: *Proc. of 7th ACM Conference on Embedded Networked Sensor Systems, SenSys*, 2009, pp. 1–14.
- [16] H. Habibzadeh, T. Soyata, B. Kantarci, A. Boukerche, C. Kaptan, Sensing, communication and security planes: A new challenge for a smart city system design, *Comput. Netw.* 144 (2018) 163–200.
- [17] S. Halder, A. Ghosal, A predetermined deployment technique for lifetime optimization in clustered wsns, in: *Proc. of 15th International Conference on Algorithms and Architectures for Parallel Processing, ICA3PP*, in: LNCS, vol. 9531, 2015, pp. 682–696.
- [18] S. Halder, A. Ghosal, A location-wise predetermined deployment for optimizing lifetime in visual sensor networks, *IEEE Trans. Circuits Syst. Video Technol.* 26 (6) (2016) 1131–1145.
- [19] Y. Hu, Y. Niu, An energy-efficient overlapping clustering protocol in WSNs, *Wirel. Netw.* 24 (5) (2018) 1775–1791.
- [20] K. Karunanithy, B. Velusamy, Reliable location aware and cluster-tap root based data collection protocol for large scale wireless sensor networks, *J. Netw. Comput. Appl.* 118 (2018) 83–101.
- [21] M. Keally, G. Zhou, G. Xing, D.T. Nguyen, X. Qi, A learning-based approach to confident event detection in heterogeneous sensor networks, *ACM Trans. Sensor Netw.* 11 (2014) 1–28.
- [22] L. Kong, Q. Xiang, X. Liu, X.Y. Liu, X. Gao, G. Chen, M.Y. Wu, ICP: instantaneous clustering protocol for wireless sensor networks, *Comput. Netw.* 101 (2016) 144–157.
- [23] H. Li, Y. Liu, W. Chen, W. Jia, B. Li, J. Xiong, COCA: constructing optimal clustering architecture to maximize sensor network lifetime, *Comput. Commun.* 36 (3) (2013) 256–268.
- [24] Z. Li, Y. Liu, A. Liu, S. Wang, H. Liu, Minimizing convergencast time and energy consumption in green internet of things, *IEEE Trans. Emerg. Topics Comput.* (2019) 1–14.
- [25] K. Liu, J. Peng, L. He, J. Pan, S. Li, M. Ling, Z. Huang, An active mobile charging and data collection scheme for clustered sensor networks, *IEEE Trans. Veh. Technol.* 68 (5) (2019) 5100–5113.
- [26] J. Luo, J.-P. Hubaux, Joint mobility and routing for lifetime elongation in wireless sensor networks, in: *Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM*, vol. 3, 2005, pp. 1735–1746.
- [27] I. Minakov, R. Passerone, A. Rizzardi, S. Sicari, A comparative study of recent wireless sensor network simulators, *ACM Trans. Sensor Netw.* 12 (3) (2016) 1–40.
- [28] N. Mittal, U. Singh, R. Salgotra, B.S. Sohi, An energy efficient stable clustering approach using fuzzy extended grey wolf optimization algorithm for WSNs, *Wirel. Netw.* 25 (8) (2019) 5151–5172.
- [29] J.M. Mora-Merchan, D.F. Larios, J. Barbancho, F.J. Molina, J.L. Sevillano, C. Leon, Mtosim: a simulator that estimates battery lifetime in wireless sensor networks, *Simul. Model. Pract. Theory* 31 (2013) 39–51.
- [30] N. C. Data Center, NOAA National Climatic Data Center station in Kingston, RI, 2017, <http://www.ncdc.noaa.gov/crn/qcdatasets.html>.
- [31] P. Neamatollahi, M. Naghibzadeh, S. Abrishami, M.-H. Yaghmaee, Distributed clustering-task scheduling for wireless sensor networks using dynamic hyper round policy, *IEEE Trans. Mob. Comput.* 17 (2) (2018) 334–347.
- [32] S. Olariu, A. Wada, L. Wilson, M. Eltoweissy, Wireless sensor networks: leveraging the virtual infrastructure, *IEEE Netw.* 18 (4) (2004) 51–56.
- [33] R.W. Pazzi, A. Boukerche, R.E. De-Grande, L. Mokdad, A clustered trail-based data dissemination protocol for improving the lifetime of duty cycle enabled wireless sensor networks, *Wirel. Netw.* 23 (1) (2017) 177–192.
- [34] J. Polastre, R. Szewczyk, D. Culler, Telos: enabling ultra-low power wireless research, in: *Proc. of 4th International Symposium on Information Processing in Sensor Networks, IPSN*, 2005, pp. 1–6.
- [35] S. Radhika, P. Rangarajan, On improving the lifespan of wireless sensor networks with fuzzy based clustering and machine learning based data reduction, *Appl. Soft Comput.* 83 (2019) 105610.
- [36] N. Sabor, S.M. Ahmed, M. Abo-Zahhad, S. Sasaki, ARBIC: an adjustable range based immune hierarchy clustering protocol supporting mobility of wireless sensor networks, *Pervasive Mob. Comput.* 43 (2018) 27–48.

- [37] T. Shu, M. Krunz, Coverage-time optimization for clustered wireless sensor networks: a power-balancing approach, *IEEE/ACM Trans. Netw.* 18 (1) (2010) 202–215.
- [38] J. So, H. Byun, Load-balanced opportunistic routing for duty-cycled wireless sensor networks, *IEEE Trans. Mobile Comput.* 16 (7) (2017) 1940–1955.
- [39] H. Taheri, P. Neamatollahi, O.M. Younis, S. Naghibzadeh, M.H. Yaghmaee, An energy-aware distributed clustering protocol in wireless sensor networks using fuzzy logic, *Ad Hoc Netw.* 10 (7) (2012) 1469–1481.
- [40] S. Tomic, M. Beko, R. Dinis, 3-d target localization in wireless sensor networks using rss and aoa measurements, *IEEE Trans. Veh. Technol.* 66 (4) (2017) 3197–3210.
- [41] E. Wei, A. Ozdaglar, Distributed alternating direction method of multipliers, in: *Proc. of IEEE 51st IEEE Conference on Decision and Control, CDC, 2012*, pp. 5445–5450.
- [42] Z. Xu, L. Chen, C. Chen, X. Guan, Joint clustering and routing design for reliable and efficient data collection in large-scale wireless sensor networks, *IEEE Internet Things J.* 3 (4) (2016) 520–532.
- [43] L. Xu, R. Collier, G.M. O'Hare, A survey of clustering techniques in WSNs and consideration of the challenges of applying such to 5G IoT scenarios, *IEEE Internet Things J.* 4 (5) (2017) 1229–1249.
- [44] M. Ye, C. Li, G. Chen, J. Wu, An energy efficient clustering scheme in wireless sensor networks, *Ad Hoc Sensor Wirel. Netw.* 3 (2–3) (2007) 99–119.
- [45] O. Younis, S. Fahmy, HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, *IEEE Trans. Mobile Comput.* 3 (4) (2004) 366–379.
- [46] M. Zuniga, B. Krishnamachari, Analyzing the transitional region in low power wireless links, in: *Proc. of 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, SECON, 2004*, pp. 517–526.



Amrita Ghosal obtained her Ph.D. degree in computer science and engineering from Indian Institute of Engineering Science and Technology, India in 2015. She received her M. Tech. degree in computer science and engineering from Kalyani Govt. Engineering College, India in 2006.

She is currently a Postdoctoral Researcher at University of Padua, Italy. Prior to that, she was Assistant Professor in the Department of Computer Science and Engineering, Dr. B. C. Roy Engineering College, India. Her current research interests include security and

privacy in wireless resource-constrained mobile device and smart grid, network modeling and analysis. She has published research works in reputed conference proceedings and journals in her field. She also has co-authored a number of book chapters.



Subir Halder received his M. Tech. and Ph.D. degrees in computer science and engineering from Kalyani Government Engineering College and Indian Institute of Engineering Science and Technology, India in 2006 and 2015, respectively.

He is currently a Postdoctoral Researcher at University of Padua, Italy. Prior to that, he was Assistant Professor in the Department of Computer Science and Engineering, Dr. B. C. Roy Engineering College, India. He has coauthored more than 25 papers in international peer-reviewed conferences and journals in his field. He

has also coauthored 5 book chapters. His research interests include security and privacy in next generation networking including WSN, IoT, network modeling and analysis, and performance evaluation and optimization.



Sajal K. Das, an IEEE Fellow, is a professor of Computer Science and the Daniel St. Clair Endowed Chair at the Missouri University of Science and Technology, where he was the Chair of Computer Science Department during 2013–2017. His research interests include wireless and sensor networks, mobile and pervasive computing, mobile crowd sensing, cyber-physical systems and IoTs, smart environments, distributed and cloud computing, and cyber security. He has published extensively in these areas with over 700 research articles in high quality journals and refereed conference proceedings. He holds 5 US patents and coauthored 4 books. His h-index is 85 with more than 31,500 citations according to Google Scholar. He is a recipient of 10 Best Paper Awards in prestigious conferences like ACM MobiCom and IEEE PerCom, and numerous awards for teaching, mentoring and research including the IEEE Computer Society's Technical Achievement Award for pioneering contributions to sensor networks and mobile computing, and University of Missouri System President's Award for Sustained Career Excellence. He serves as the founding Editor-in-Chief of Elsevier's *Pervasive and Mobile Computing Journal*, and as Associate Editor of several journals including the *Journal of Parallel and Distributed Computing*, *IEEE Transactions on Dependable and Secure Computing*, *IEEE Transactions on Mobile Computing*, and *ACM Transactions on Sensor Networks*.