Peer-to-Peer Networking and Applications
https://doi.org/10.1007/s12083-020-01016-y

Construction of multi-level data aggregation trees for energy efficiency and delivery delay in machine-to-machine communications

Prasad Challa [1] · **B. Eswara Reddy** [1]

Received: 10 January 2020 / Accepted: 7 October 2020

Abstract
Machine-to-Machine (M2M) communications have gone forth as the newest technology for succeeding in communication generations. The M2M connections use the sensor nodes to capture an event into data packets and relayed through a network. The sensor nodes consume more energy whenever the increase in data packets transmitted from the sensor nodes in the system. To reduce the energy utilization applying the data aggregation is essential. We proposed a comprehensive model for calculating energy utilization and delay-tolerance by using Multi-Level Data Aggregation Trees (MLDAT). In the proposed scheme, the first stage is about the construction of Multi-Level Data Aggregation Tree, which aggregates the data originated from various wireless sensor nodes in the communication network. In the second stage, a delay-tolerant scheduling algorithm for controlling the delivery delay for user queries presented. Ultimately, the simulation results of the proposed scheme show that the suggested algorithms have better performance than the existing state-of-the-art approaches significantly.

Keywords Sensors · Machine-to-machine(M2M) communication · DAT · Energy consumption · Network delay

1 Introduction

The Machine-to-Machine(M2M) communication network involves thousands of sensors connected to the core network via a select node called *Sink node or gateway node*. The battery-operated devices monitor the environmental conditions corresponding to their position, transfer the collected data to the *gateway node*, process this data, and deliver information to Application Server. The sensor nodes can organize by themselves, providing access outside and watchdog environments. In this way, the M2M communications are getting attractively for several application areas such as health-monitoring, farming, aviation, water contamination- monitoring, crowd-monitoring and Building Monitoring Systems (BMS), etc.,

However, thousands of devices, combined with sensors, result in multiple target analysis or unnecessary data transmissions in the communication network. Therefore, the more nodes squander energy very quickly. The approaches, such as dynamic routing, medium access control, and resource allocation of communications, reduce the energy consumption of sensor nodes [1, 2]. The scheduling algorithms of the wireless sensor nodes can efficiently manage the power consumption of the communication network [3–7]. The information collecting mechanism can utilize data aggregation functions to reduce unnecessary communications. The data sensed by multiple wireless sensor devices have temporal and spatial interactions [8, 9] due to only a few sensor devices are scheduled to transmit the data packets. The other sensor nodes are in the sleep state.

Several empirical studies stated that hierarchical network topology is efficient for sensor nodes to collect data and transmit to the *gateway node* [10]. A tree-based topology is expensive for storing routing tables at each node with limited resources compared to an arbitrary network topology [11]. The data aggregation tree structure's saving ability to implement aggregation functions is likewise referred to as data aggregation trees (DATs). DATs have *gateway node* as root, these are directed trees and receive a distinct route from every sensor device to *the gateway node*. Furthermore, in data aggregation trees (DAT), the sensor nodes gather data from various wireless sensors that are fused at intermediate wireless sensor nodes, as stated by the data aggregation functions such as MAX, MIN, SUM, AVERAGE, COUNT, etc. [12].

✉ Prasad Challa
chprasad18@gmail.com

B. Eswara Reddy
eswar.cse@jntua.ac.in

1 Department of CSE, JNTUA, Anantapuramu, Andhra Pradesh, India

Published online: 21 October 2020

Data Aggregation technique is applied to lower total energy consumption in the communication networks. Mottola et al. described an energy-saving routing algorithm, which states that the data aggregation becomes better up to 70% of network life for the entire wireless network the real-world experiment results. But there is a significant effect of data aggregation, where the delays for processing the user queries with data aggregation increase about four times. Shin-Yeh Tsai et al. studied timing control, in which time is taken to buffer packets. The maximal number of buffered data packets is to take into account data aggregation [13].

The rest of the paper is structured as follows. Section 2 specifies the literature survey on data collection and data aggregation in the M2M Communication networks. Section 3 presents the details of the problem statement and the system environment. Section 4 presents the aggregation tree construction algorithm and delay tolerance model and examine development issues. The simulation environment and results are analyzed in Section 5. The conclusion and future directions are given in Section 6.

The primary contributions of this article are pruned as follows:

- We distinguish the problems when constructing the Multi-level data aggregation tree (MLDAT) for the M2M communication network. Furthermore, we proposed two algorithms to measure minimum data transmissions and also minimal latency for user queries.
- Construction of the Multi-level Data Aggregation Tree problem, it is a well-known NP-Complete. We work out this problem in two stages. On the stage1, we compute aggregating nodes obtained by the combination of Maximal Independent Set with minimal load, LBMIS(I), and set connectors denoted as CMIS(C). Next, in stage2, Load Balanced Parent Node Assignment LBPNA, which is NP-Complete, we provide an approximation solution.
- In addition to this, we proposed a Delay Tolerant Scheduling Algorithm based on the periodic per-hop method.

2 Related work

In the Data Aggregation Trees (DAT), most research studies focus on scheduling algorithms to transfer the data packets from sensors to the *gateway node*. The Data Aggregation Tree construction problem is NP-Complete Problem only fewer approaches are available for constructing the Optimistic Data Aggregation Trees. By this motivation, we try to study the construction of Data Aggregation Tree for the objective of energy efficiency and delivery latency in the wireless communications network.

The energy consumption of sensor nodes is cut by the dynamic nature of the wireless channel consuming energy due to loss of packets and retransmissions [14]. Cognitive Radio sensors may be efficient in adapting varying channel situations, which would surge transmission efficiency and lower the power consumption used to transmit and receipt [15, 16].

Many researchers have conducted several experiments on node scheduling approaches in the M2M communications, and all those schemes work based on the assumption that sensor nodes are stationary. So, these systems cannot be applied to the M2M applications in which there is a requirement of mobility in wireless communication devices, for example, health control. In [17–20], the studies are illustrated cluster-based scheduling algorithms for the need for movement.

In [4], a cluster-based energy-efficient, mobility-centric node scheduling scheme is (CENM) proposed to network coverage by keeping minimal node inactive and minimizing the total failures by selecting different cluster gateway nodes. Several Data Aggregation Models have aimed for the development of Data Aggregation Trees (DAT) for WSNs. Ding, Min et al. recommended energy-aware distributed heuristic to generate the aggregation tree (EADAT) [21]. It makes use of neighboring broadcast scheduling and distributed competition with neighbors. Kumar Padmanabh et al. introduced an algorithm that is flexible to opt one of the aggregations depends on the scheme and degree of data aggregation based on network traffic [22].

Yao Lu et al. proposed a spanning tree called Multi-Objective Steiner Tree (MOST) based on some standard metrics. A heuristic method Jumping particle swarm optimization is suitable for a static objective, but in practice, if the targets are frequently changed, then in incurs more overhead [23]. Randhawa et al. also suggested analyzing the results of tiny aggregation to reduce energy consumption and increase the network lifetime. It is only suitable for fixed connections in small networks. In the case of lossy link radio connections, it results in poor performance [24].

Mohsenifard et al. proposed a data aggregation tree (DAT) using a cuckoo optimization algorithm that did not consider the load–balance factor [25]. In [26], monitoring hierarchical trees became easy. Still, when moving out from the cluster head, the number of nodes increased enhanced relay routing in which the data packets relayed among sensor nodes, taking too much delay for user queries. In the case of duty-cycled WSN, Le et al. [27] proposed a level order tree-based scheduling scheme for delay tolerance, which having more computations overhead for scheduling.

Zhaohui et al. [28] and Lu, Yao, et al. [29] proposed a data aggregation scheme for heterogeneous wireless sensor networks concentrating on local tree topology and maintenance. Also, in [30] Kale et al., proposed scheduling technique with local heuristics approach both are having improvements in the centralized and static network topology. Still, in practice, WSN links are dynamic, and connections are lossy link nature.

Compressive Sensing is an advanced signal processing technique, in which data sets process efficiently and accurately by acquiring, storing, fusing, and processing slowly. This method, which links data gathering, compression, dimensionality reduction, and optimization, has drawn significant attention from researchers and applied scientists in several fields. The CS-based aggregation solution improves energy efficiency [31]. The energy consumption models are summarized in Table 1.

3 System model and problem definitions

In this section, we depict the overall construction of the MLDAT under the lossy link network model. First, we listed the assumptions commonly present in the network model, followed by the formal definition of the problem and some remarks on the issue.

3.1 Assumptions

We assume the M2M Communication network as a connected graph G(V, E), where $V =$ Error ! Bookmark not defined. are the set of sensor nodes, and E denotes the lossy wireless connections among the nodes. In the sensor nodes, v_0 indicates the *gateway node*, unlike the other nodes, the *gateway node* has a continuous power supply. Every node except the *gateway node* (v_0) powered by a non-removable battery.

All the nodes at any point in time they can determine their residual energy. We assume every node in the network has the same transmission range $\mathbb{R}$. If there is an edge between any two nodes $i, j \in V$ then the distance between them is denoted by $l_{ij} \leq \mathbb{R}$. In addition to this, we assume no node failure and also no duty-cycles.

The n nodes monitor the environment in the deployed area and periodically send the gathered data to *gateway node* v_0 along with the MLDAT routing structure. Every node prepares a data packet of size B-bits during each report. Non-leaf sensors aggregate different incoming data packets, along with its data packet into a single outgoing data packet. Also, we assume the data gathering rate of any node v_i is γ_i. Also, $\mathbb{R}$ represents the maximum data receiving rate of all the sensors.

3.2 The network model

In a lossy-link network model, we denote the communication network as connected undirected graph G(V, E, P(E)), where $V = V_s \cup \{v_0\}$ and $V_s = \{v_1, v_2, \ldots v_n\}$ is a collection of $n + 1$ nodes, any sensor node is denoted by v_i, where $0 \leq i \leq 1$. The communication links indicated as E is the set of lossy links $\forall v_i, v_j \in V_s$, there exists a link (v_i, v_j) in G iff:

1) Both v_i and v_j stay within wireless radio transmission range, also.

2) $l_{ij} > 0$, as every link $(v_i, v_j) \in$ E, l_{ij} denotes the probability that v_i can straightly transfer a data packet towards v_j successfully; as well as

$$\mathrm{P(E)} = \left\{ l_{ij} \mid \left(v_i, v_j\right) \in E, 0 \leq l_{ij} \leq 1 \right\} \tag{1}$$

In the network all the sensor nodes can send data packets by forwarding through the intermediate nodes, in that case, we define 1-hop neighborhood $\{N_1(v_i)\}$ as the data packets can reach by one-step to the destination also k-hop neighborhood $\{N_h(v_i)\}$ set as the data packets can reach by k-steps to the target in the wireless sensor network.

3.3 Problem definition

The primary goal of the proposed Multi-Level Data AggregationTree (MLDAT) construction problem is to reduce energy consumption and prolong the network lifetime to accomplish this measurement of every node's traffic pattern and

Table 1 Summary of the Related Work

S. No	Approach	Energy consumption	Delivery Delay	Network lifetime	Scalability
1	Cognitive radio	✓	×	×	×
2	Sleep node scheduling	✓	×	×	×
3	Relay nodes	✓	✓	✓	×
4	Deterministic Network Model	✓	×	✓	✓
5	Periodic simple and Periodic Per-Hop	✓	✓	×	×
6	MAC: EER-ACK	✓	×	×	×
7	Approximation or Heuristics Algorithms	×	✓	✓	✓
8	Clusters Hierarchical Aggregation	✓	×	×	×
9	Compressive Sensing (CS)	✓	×	×	✓

workload network. We can distinguish the network traffic load for internal nodes and leaf nodes. Compared to the leaf nodes, Internodes have more workload on them. We can measure the potential traffic load of every node by using the *potential load* factor of that node.

As earlier stated in Section 1, based on the number of neighboring nodes connected to a node is an indicator of its traffic load. However, in the lossy-link wireless sensor network, some other components also there to cause *potential load* on sensor nodes. Consider the example, if $l_{ij} = 0.25$, then the expected number of transmissions is $\frac{1}{0.25} = 4$ to guarantee v_i to transfer one data packet to v_j. If l_{ij} valueless, then potential traffic load on v_j from v_i is more. Thus, a more admissable as well as formal definition of the *potential load* presented as follows:

Definition 3.1: Potential Load (ρ_i). $\forall V_i \in V_s$, the potential load of v_i defined as:

$$\rho_i = \sum_{v_j \epsilon N_{1(v_i)}} \left\lceil \frac{B}{\gamma_i} \right\rceil \quad \frac{1}{l_{ij}} \tag{2}$$

The construction of the MLDAT problem solved in two stages; in stage1, we find out the internal nodes of the given M2M communication network. To achieve this, first compute the Maximal Independent Set with minimum *potential load* LBMIS(I), but finding the LBMIS is well known NP-Hard. We can map LBMIS problem just like integer Linear Program LP^*_{LBMIS} As follows:

Definition 3.2.

Let $\omega_{i,}$ $\forall\, i \in V$ is a binary decision variable set to 1 *if* the node is independent; the other is 0.

$$Max \;\; v = min\{\, \rho_i | \;\; \forall v_i \epsilon I\} \tag{3}$$

$$s.t \;\;\; \omega_0 = 1;$$

$$\omega_i + \sum_{v_j \epsilon N_1(v_i)} \omega_j \geq 1;$$

$$\textstyle\sum_{v_j \epsilon N_1(v_i)} \omega_{ij} = 0;$$

$$\omega_i \geq \omega_{ij} \; ; \; \omega_j \geq \omega_{ij};$$

$$\omega_i + \omega_j - 1 \leq \omega_{ij};$$

$$\omega_i, \omega_j, \omega_{ij} \epsilon \{0, 1\}, \;\; \forall v_{i,} v_j \epsilon v_s$$

In Eq. (3), the objective function for selecting the set of nodes such that the nodes having minimal potential load with the specified constraints represented as relaxed Integer Linear Problem.

Next, we can arrange the LBMIS(I) into several partitions as disjoint sets. Those connected by selecting some other nodes as connectors, which are denoted by C. Finally, the combination of both I and C are called internal nodes or dominating set or Connected Maximal Independent Set CMIS of the network. In the second phase, we can assign the parent nodes to the internal nodes and the parent nodes to the leaf nodes denoted with A_I and A_L.

Definition 3.3: Parent Node Assignment (PNA) for internal nodes (A_I),

$$A_I = \{I(v_i) \;\; | \forall v_i \epsilon D, 1 \leq i \leq m\} \tag{4}$$

In Eq. (4), the internal nodes are assigned with parents from the dominant set(D) towards the root node. Also, assigning the parent links for the leaf nodes is a well known NP-Complete. The assignment of leaf nodes to their parent node is done by considering the minimum *actual load* of that particular node.

Let formally define the *actual load* among the *dominating set(D)* of nodes.

Definition 3.4: Actual Load (α_i). The actual load of an internal node v_i is:

$$\forall v_i \epsilon D, \quad \alpha_i = \sum_{v_j \in \{\; L(v_i) \cup I(v_i) |\; i \neq j \;\}} \left\lceil \frac{B}{\gamma_i} \right\rceil \quad \frac{1}{l_{ij}} \tag{5}$$

Definition 3.5: Assignment Parent Nodes to leaf nodes (A_L) of lossy-link wireless sensor network denoted by the graph G(V, E, P(E)) along with CMIS, $D = \{v_1, v_2, \ldots v_m\}$.

It is required to obtain m disjoint sets from V, denoted by L(v_1), L(v_2), …, L(v_m), so that:

1) Each set L(v_1) ($1 \leq i \leq m$) contains exactly one non-leaf node v_i.

2) $\overset{m}{\underset{i=1}{\cup}} L(v_i) = V$, and $L(v_i) \cap L(v_j) = \varnothing$,   ($1 \leq i \neq j \leq m$).

3) $\forall v_u \in$ L(v_i) ($1 \leq i \leq m$) and $v_u \neq v_i$, such that (v_i, v_1) $\in$ E.

4) Assign v_1 ($1 \leq i \leq m$) as the parent node of the nodes in L(v_1)\{ v_1}.

Therefore formally we can denote Parent Node Assignment (PNA) for leaf nodes as,

$$A_L = \{\mathrm{L}(v_1) \; | \forall v_1 \epsilon \mathrm{D}, 1 \leq i \leq m\}.$$

We define a decision variable β_i to denote whether the sensor v_i is an internal node or a leaf node. β_i sets to be 1 *iff* the sensor v_i is an internal node. For remaining nodes, β_i assigned as 0. Moreover, Let select a random variable ξ_{ij} to every edge connecting an *internal node* v_i, a *leaf node* v_j in graph G formed with the lossy-link wireless communication network.

We can define the parent node assignment as standard linear programming LP^*_{LBPNA}

$$Max \; \upsilon = \min \left\{\alpha_i \;\;\; | \forall \mathrm{v_i} \epsilon D \right\} \tag{6}$$

$$such \; that \sum_{v_i \in N_1(v_j)} \left(\beta_i \xi_{ij} = 1\right) \forall \notin \mathrm{D}$$

$$\xi_{\mathrm{ij}} \epsilon \{0, 1\}$$

Table 2 Notations

ρ_i	Potential load of node i
α_i	The actual load of node i
β_i	The decision variable associated with the node i
P(E)	Probability of lossy-link connectivity
A_I	Parent node assignment for Internal nodes
A_L	Parent node assignment for leaf node
I	Maximal Independent Set
C	Connected maximal Indent Set
D	Dominating Set
$N_2(v)$	Two-hop neighbors of node v
B	The size of the data packet (bits)
ξ_{ij}	The random variable associated to the edge between i and j
$L(v)$	Level of vertex v
ω_i	Decision Variable of node i
γ_i	The rate of data packets generated by node i

Therefore, the overall Parent Node Assignment with minimum *actual load* (LBPNA-A^*) is a combination of both A_I *and* A_L. i.e.,$A^* = \{A_I, A_L\}$. After LBPNA is decided, by assigning a direction of each link in the constructed tree structure, we obtain an MLDAT.

In the next section, the solutions for solving the load-balanced MIS(I), connected MIS(C), load-balanced PNA(A^*), and finally, we build the MLDAT.

4 Construction of multi-level data aggregation trees

The construction of MLDAT is designed in two stages. In the first stage, we obtain the *dominating set (D)* of the given Lossy-link wireless communication network graph G(V, E, P(E)). In other words, the *dominating set (D)* contains the internal nodes of the MLDAT, including the root node. We can find out the *dominating set (D)* by solving the problem of LBMIS (I) and selecting the *connectors* (C) of set I. We can represent the LBMIS(I) problem as an Integer Linear Program (ILP) and solve by using the Linear relaxation technique. After that, we can choose some connectors (C) to form the *dominating set (D)*.i.e., $D = \{I \cup C\}$.

In the second stage, we have to form parent-child links by keeping the load balance among the nodes. First, we assign the parent nodes among the internal nodes (D) as A_I next, we assign the parent nodes for leaf nodes as A_L. The problem of Parent Node Assignment with minimum *actual load* (LBPNA) of leaf nodes A_L is formally defined as an Integer Linear Program, and it solved by using the random rounding technique.

Table 2 shows the notations used in this paper.

4.1 Approximation algorithm for MIS with minimum traffic load LBMIS(I)

The basic idea of the solution in Algorithm 1 described as given below:

Let solve the linear program of LP* $_{\text{LBMIS}}$ to obtain an optimal solution, which is denoted by

$\left(\omega^*, v^*\right)$, (where)

$\omega^* = \langle \omega_1^*, \omega_2^* \dots \omega_n^* \rangle$, also round-off ω_i^* values into integers ω_i render to the six steps.

presented through lines 3–14 of Algorithm 1, also.

v^* denotes the corresponding node.

1) Sort sensor nodes by the ω_i^* value (*where* $1 \le i \le n$) in the decreasing order.
2) Set the sink node to be the independent node, i.e., $\omega_0 = 1$.
3) Fix for all $\widehat{\ }$ωi with 0's.
4) Begin with the first node in sorted array P. If no node selected as an independent node in $v_i{}'s$ 1-hop neighborhood, then set $\widehat{\omega}_i = 1$ with probability $p_i = \omega_i^*$.
5) Repeat step (4) until the end of array A.
6) Again repeat steps 4) and 5) as $\frac{8\ln(n)}{\min\left\{\omega_i^* \middle| \ v_i \ \epsilon \mathbb{V}, \omega_i^* > 0 \ \right\}}$ many times.

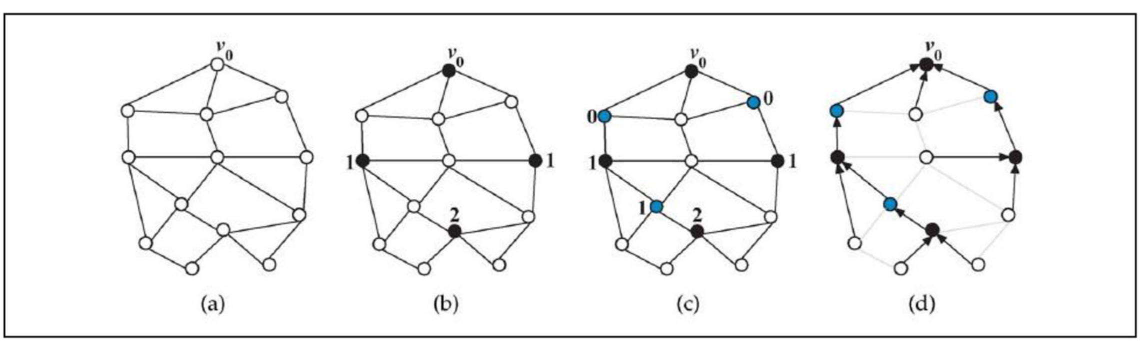


Fig. 1 Illustration of (**b**) LBMIS, (**c**) CMIS, and (**d**) MLDAT construction process

Algorithm 1: Approximation Algorithm to obtain load-balanced MIS(I)

Solve LP^{*}_{LBMIS}, Let(ω^{*}, v^{*}) be the optimal solution, Where $\omega^{*} = \langle \omega_1^{*}, \omega_2^{*}, \ldots \omega_n^{*} \rangle$, $v^{*} = \min\{\rho_i \mid \forall v_i \in V\}$;
Sort all ω^{*} values in a non-increasing array. The arranged ids put into an array denoted by P[n].
$\widehat{\omega_0} = 1$;
for $i = 1$ through n **do**
$\widehat{\omega_i} = 1$;
$counter = 0$;
while $counter \leq \tau$ where $\tau = \frac{8\ln(n)}{\min\{\omega_i^{*} \mid v_i \in \mathbb{V}, \omega_i^{*} > 0\}}$
$k = 0$;
    while $k < n$ **do**
        $i = P[k]$;
            if $\forall \boldsymbol{v_j} \in \boldsymbol{N_1(v_i)}, \boldsymbol{\widehat{\omega_j} = 0}$ then
                $\widehat{\omega_i} = 1$ with probability $p_i = \omega_i$;
$k = k + 1$
$counter = counter + 1$;
return $\left(\boldsymbol{\widehat{\omega_i}, \widehat{v_i} = min\left\{\sum_{j:\widehat{\omega_i} l_{ij} > 0} \left\lceil \frac{\beta}{\gamma_i} \right\rceil \frac{1}{l_{ij}} \;\middle|\; \forall v_i \in V\right\}}\right)$

4.2 Connecting load-balanced MIS(C)

After finding the LBMIS(I) one more step is required to obtain the dominating set (D). We have to select the *connectors(C)*. To do this by a similar procedure as in [32], to obtain a minimum set of connected MIS (C) to connect the load-balanced MIS(I).

Partition the LBMIS(I) into disjoint node sets by using the following criterion:

$$I_0 = \{v_0\}(\text{and})$$

$$I_l = \left\{v_i \,|v_i \in I, \exists v_j \in I_{l-1}, v_i \in N_2\left(v_j, v_i \notin \cup_{k=1}^{l-1} I_k\right)\ \right\} \tag{7}$$

The *gateway node* is put into I_0. It acts like the root of the MLDAT, i.e., $|I_0| = 1$. All of the other nodes present in I in a 2-hop neighborhood out from nodes in I_{l-1} are placed in I_l. So, l is termed as the *level* of the independent node. I_l Represent collection of separate nodes in *level l* of G corresponding to a node in I_0. In addition to this, L denotes the maximum allowed levels of independent nodes.

For each level i, where $i \in [0, L-1]$, assume S_i is a set of neighboring nodes to at least one node in I_i also at least one node in I_{i+1}. Next, find a minimum-sized set of nodes $C_i \subseteq S_i$ to cover all nodes in I_{i+1}.

Let $C = \cup_{i=0}^{l-1} C_i$ and.

Therefore, finally, D = {I ∪ C} is the CMIS of the original graph G.

We use the M2M communication network shown in Fig. 1a to illustrate the construction process of a Dominating Set (D). In Fig. 1a, all circles denote the sensor nodes as we specified previously, the construction of MLDAT T having two steps. In the step1, we solve the LBMIS(I) problem using Algorithm 1, which is represented in Fig. 1b as black circles. In step2, we choose appropriate LBMIS connectors (C), expressed as grey nodes in Fig. 1c, for connecting every node in I to form a dominating set (or) Internal nodes of G, denoted as CMIS (D).

i.e., $D = \{I \cup C\}$.

4.3 Load balanced PNA for internal nodes(A_I)

After d*ominating set (D)* finds out, in the next stage, we dedicate to find a load balance PNA for Internal nodes A_I. The following procedure is given:

1. $\forall v_i \in C_0$, their parent is to be the *gateway node* v_0.
2. As per the *index* non-decreasing order, all $v_i \in C_l$, and $l > 0$, their parent connected to the neighboring node $v_j \in I_{l-1}$, which is having the minimum load.
3. As per the *index* non-decreasing order, all $v_i \in I_l$, and $l > 0$ their parent connected to the neighboring node $v_j \in C_{l-1}$, which is having the minimum load.

4.4 Integer linear program formulation of load-balanced PNA of leaf nodes(A_L)

As we know that, finding an arbitrary data aggregation tree having maximum network lifetime is NP-Complete [27]. Similarly, we can prove that PNA with minimal load for leaf nodes A_L is also an NP-complete. The following randomized algorithm used to solve the load-balanced PNA for Leaf Nodes, A_L. The formal definition of the load-balanced PNA for Leaf Nodes A_L is explained in definition 3.4. The problem can solve by using the following randomized algorithm.

Algorithm 2: Randomized Approximation Algorithm for load-balanced PNA for Leaf Nodes $A_{L.}$

Solve LP^*_{LBPNA}. Let (ξ^*, v^*) be the optimal solution.
Sort the ξ^*_{ij} values in each row (for every node) of ξ^* in the non-increasing order and next to keep the corresponding j (v_j's IDs) in two-dimensional array denoted by P[n][m]
$\widehat{\xi_{ij}} = 0$
while $k \leq \kappa = \frac{7\log(n)}{\delta^2 \min\{\xi^*_i \mid 1<i<n, 1\leq j\leq m, \xi^*_{ij}>0\}}$ **do**
$k = 0, l = 0;$
    while $k < n$ **do**
     $i = k;$
        while $l < m$ **do**
            $j = P[k][l]$
           if $v_j \in N_1(v_i)$ and $\hat{\xi} = 0$ then
                $\widehat{\xi_{ij}} = 1$ with probability ξ^*_{ij} ;
            break;
      $k = k + 1$
return $(\hat{\xi}, \hat{v}) = min\{\alpha_i \mid \forall_i \in D\}$

5 Delay-tolerant scheduling model

Once the Multi-level Data Aggregation Tree (MLDAT) is constructed, the aggregation time control mechanism is applied by the Delay-Tolerance function. Consider the following M2M Communication network shown in Fig. 2. Every node gathers information in its location and sent it to the *gateway node* (Fig. 2b), which further transmits the data to Application Server, as shown in Fig. 2c.

The source node is $N(i)$, $0 \leq i \leq k-1$ relays data m_i, and sends to the *gateway node* $N(k)$. The data m_i collected by $N(i)$ to be sent to the user by application server by following the Multi-level DAT structure. We present a periodic per-hop aggregation time scheduling algorithm present at each node $N(i)$. The time control for the aggregation scheduling algorithm and delay tolerance process consisting of the following steps:

Algorithm 3: Delay-Tolerant Scheduling Algorithm

1. When $N(i)$ receives the data from its child node $N(i-1)$ else prepares a data packet to have the data $\boldsymbol{m_i}$, gathered by $N(i)$ by its own, it keeps the data packets in its buffer.
2. $N(i)$ checks its buffered packets, if the buffered packets are empty, $N(i)$ begins a refresh timer with count T.
3. Suppose the packets present in the buffer is extended to the maximal number of buffered data packets. In that case, $N(i)$ extracts the buffered data packets and aggregates the gathered information into a new data packet (or) If the *refresh* time reaches 0, then $N(i)$ stops the buffering timer to count T and save the original parcel to its next parent node $N(i+1)$.

Aggregation Timing Control Scheduling Algorithm considers both the number of buffered packets and the refresh timer for transmitting fresh data packet at a node to overcome long delays for user queries. After collecting data from all nodes, the *gateway node* $N(k)$, restores the gathering data present in the application server. The gateway node resets the Application Server to minimize the signal overhead in the core network after a new refresh time value denoted by T_{ref}.

6 Performance evaluation

In the construction of Data Aggregation Trees, most of the earlier research works focus on the classical Shortest Path Tree, Minimal Spanning Tree, and Load Balanced Cluster Head Approaches. In this paper, we selected the most dominating tree-based routing models, such as Load-Balanced Data Aggregation Trees (LBDAT) [33], Two-Tier Adaptive Model Aggregation (TTAMA) [34] and Energy Efficient Spanning tRee (EESR) [35]. The Proposed MLDAT approach is suitable to compare with the selected routing tree structures. We compare all the four algorithms in terms of energy consumption and delay tolerance.

The Simulation environment is having all sensors having the same transmission range of 40 m. Every sensor device randomly distributed in a rectangular area of $200m \times 400m$.

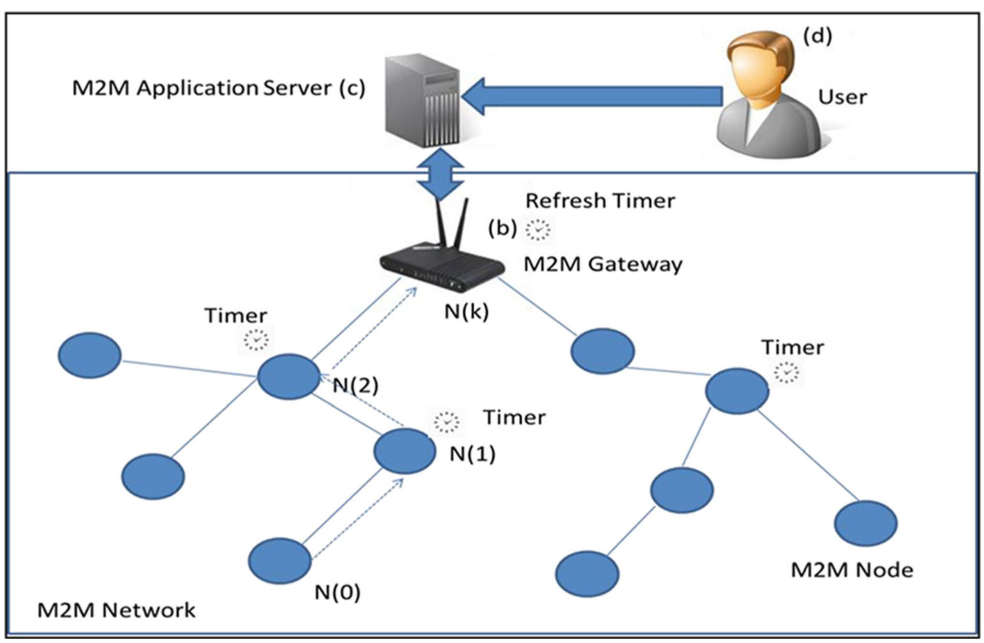


Fig. 2 The Architecture of Machine-to-Machine(M 2 M) Communication Network

Table 3 Configuration parameters

Parameter	Value
Random value for Wireless Links	[0.5–1.0]
Gateway Position	Top Left
Topology (Grid)	$200m \times 400m$
Network Size	150
Transmission Range	40 m
Energy for receiving	1Unit
Energy for Transmitting	2Units

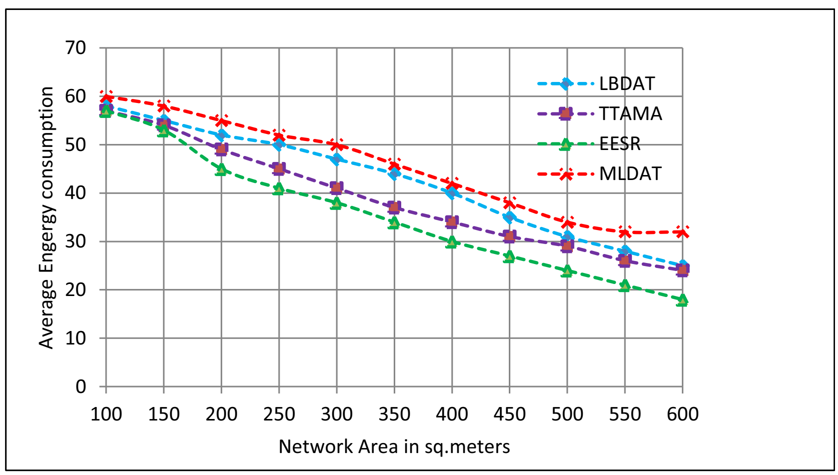
The results are getting for every different setting, 50 instances obtained, and rounded to integer values. In addition to this, a random value function is taken to assign links among the nodes. If a node is within the transmission range, then the transmission value is[0.5, 1] else it is [0, 0.5]. All the configuration parameters have listed Table 3.

Among all the parameters network Area, number of nodes, and transmission range are tunable parameters. The implementation of the proposed MLDAT model is evaluated using NS-3 tools and Python programming language, and the configuration parameters parsed by using an XML File.

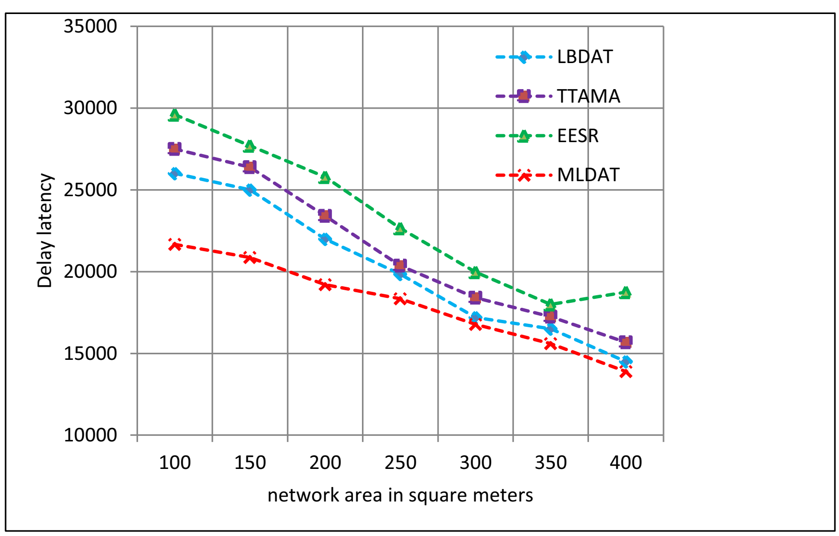
We evaluate the performance of MLDAT with the state-of-art DAT models such as LBDAT, Energy Efficient Spanning Tree (EEST), Two-Tier Aggregation Multi-target application Trees (TTAMA). The proposed MLDAT Model has better performance for energy consumption and delivery delay than the existing algorithms.

Scenario 1:Network Coverage Area

In this case, 150 sensors are distributed randomly and evenly in the rectangular network area. Each node has a transmission range of 30 m. The side length of the rectangle area is varied from 100 m to 400 m by increasing by step of 50 m. As increasing the length of the side, the network turned into thinner, and higher internal nodes are required to support the connectivity. The number of transmissions reduced in each



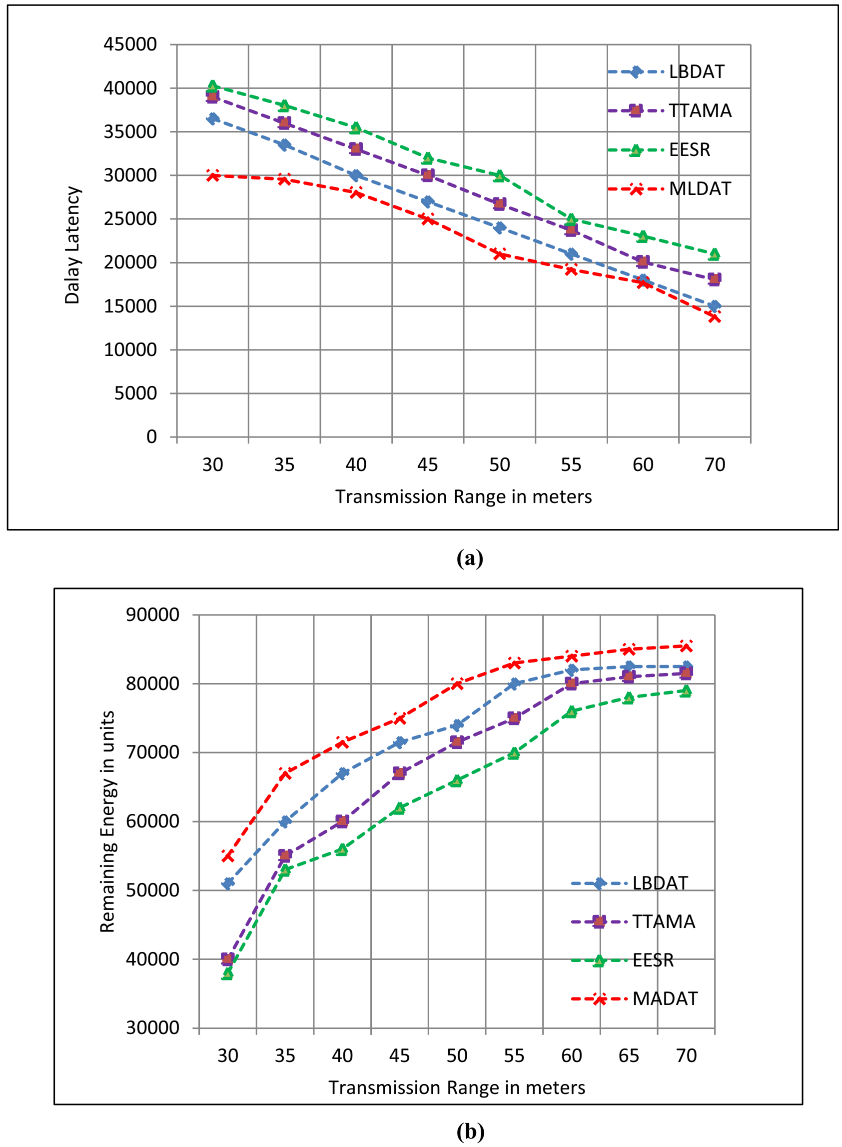
(a)



(b)

Fig. 3 **a**. Average Energy consumption. **b**. The average Delay latency against the network area

Fig. 4 **a** Residual energy for different transmission ranges. **b** Delay latency for different transmission ranges



communication round the energy consumption is also reduced, as shown in Fig. 3a. In the other models, also it is the same trend happens.

In Fig. 3b, the average delay latency for user queries is shown. When the network becomes thinner, the nodes choose available buffers packets to transmit to the gateway node. In MLDAT, we use the periodic per-hop scheduling mechanism, which improves the average delay due to the availability of buffered packets.

Scenario 2: Transmitting Range

In this scenario, the transmitting range is tuned from 30 m to 70 m by increasing a step of 5 m. The 100 nodes are deployed randomly uniformly in the network area.

We consider the $140m \times 300m$ rectangular area of grid topology. In Fig. 4a, the Residual energy for different transmission ranges is shown. In Fig. 4a, we can observe the decrease in the residual energy because the network becomes denser, which means more nodes are present within the circles of node transmissions. So even the connectivity of internal nodes is maintained although with less number of internal nodes. For the other EESR and TTAMA algorithms is very close to both use Minimal Spanning tree method to maintain aggregating node connectivity.

In Fig. 4b, the delay is measured for all the algorithms. As the network becomes denser, the delay occurred for MLDAT is better than the other models due to *periodic per-hop scheduling* which is used tunable parameter refresher timer, due to updating the refresher timer it can achieve shorter delay. In contrast, the other models use minimum latency scheduling algorithms (MLSA) that are suffering from their static time slots.

Scenario 3: Number of Sensor Nodes

In this scenario, In a rectangular area size of $200m \times 400m$, the number of sensors that are deployed randomly as 50 to 500 with unevenly. Each node transmission range as 50 m.

In Fig. 5a, it is shown that the residual energy is proportional to the number of nodes becoming more and more. It is because the redundant nodes are available for connecting CMIS. We have to perform the data aggregation in a dense area, the Parent Node Assignment (PNA) can improve the energy consumption significantly due to a lot of redundant sensors.

In Fig. 5b, as the number of nodes increasing the delay is also increased for all the traditional algorithms (LBDAT, TTAMA, and EESR) due to the dense network, the number of aggregation points are more and more, but it is different for MLDAT because the constraints on the Parent Node Assignments MLDAT selects the optimistic aggregation points with more leaf nodes.

Scenario 4: Observation of Network Lifetime

The number of sensor nodes is deployed randomly in a 200×200 $sq.$ m area, ranging from 10 to 100 with step value ten and the number of sensors 100–1000 with step value 100 and the transmission range, R = 30m fixed. The observation is that the number of nodes increasing from up to 100 the network lifetime of MLDAT is having a similar report with other approaches which is shown in Fig. 6a. In contrast, the number of nodes increasing from 100 to 1000 shows a significant improvement in the network lifetime of MLDAT compared to the LBDAT, TTAMA, and EESR. The reason is that the actual load in Eq. (5) of nodes increased in EESR and

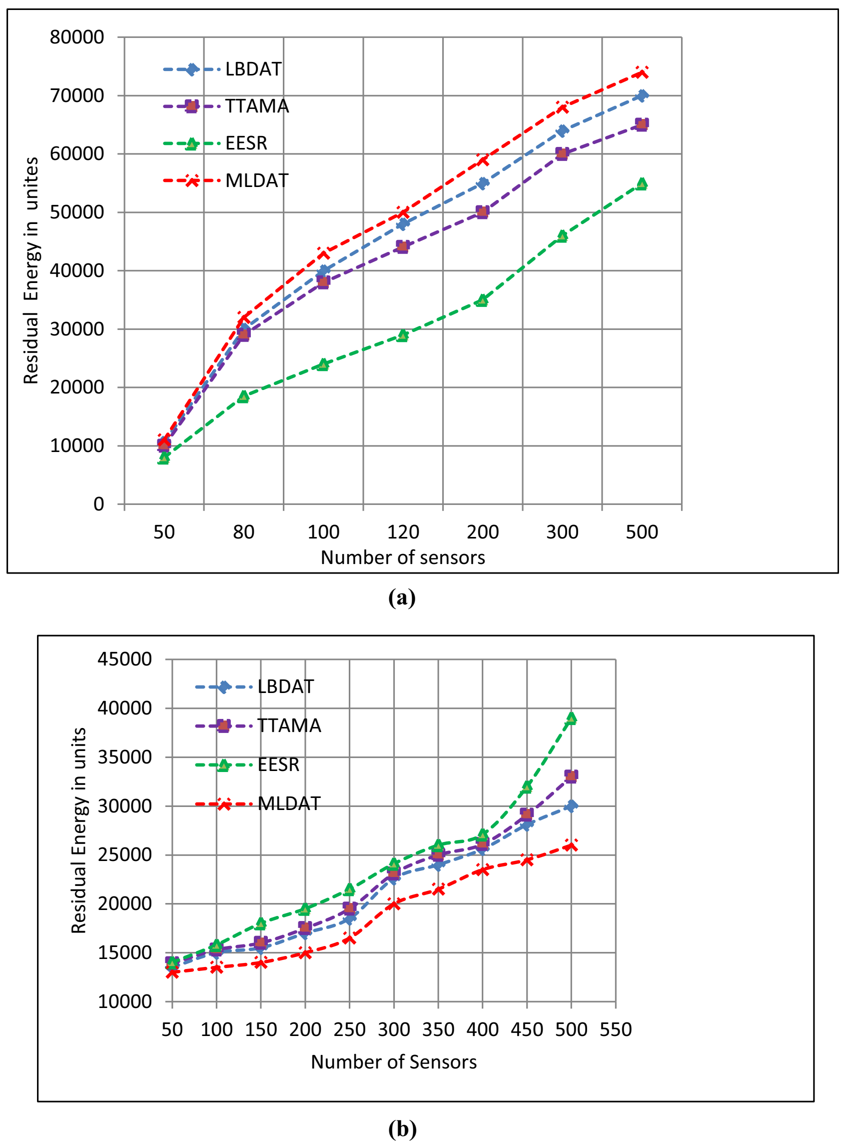
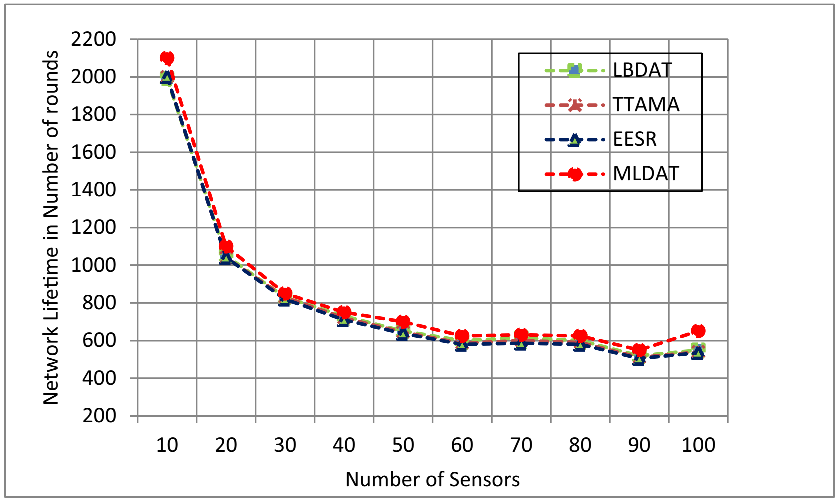
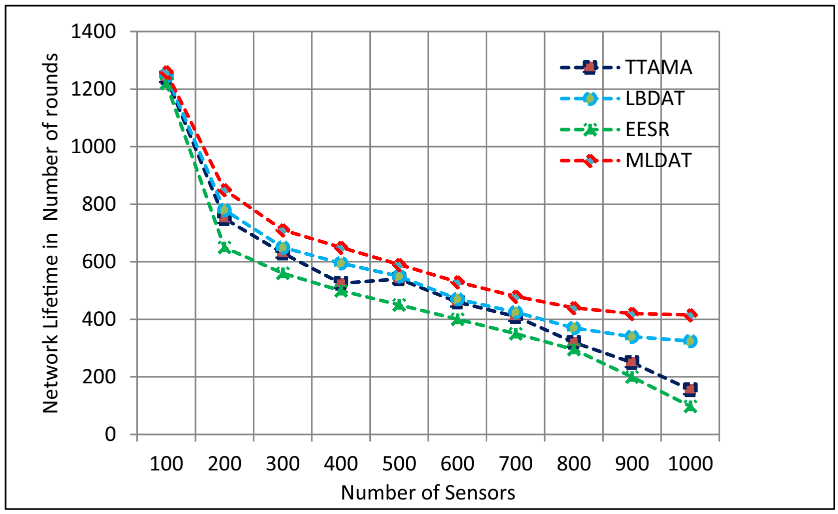


Fig. 5 **a**. Energy Consumption w.r.t the number of nodes. **b**. Network delay concerning the number of nodes

Fig. 6 **a**. Network Life Time for Number of nodes. **b**.Network Life Time concerning Number of nodes



(a)



(b)

TTAMA compared to MLDAT and LBDAT as the number of nodes increasing the overall network is shown in Fig. 6b.

7 Conclusion

In this article, we address the fundamental problem of modeling the Data Aggregation Trees in Machine-to-Machine(M2M) communication Networks. We Proposed a solution for CMIS and PNA Problems. In addition to this, Aggregation Time control Scheduling appropriately schedules the nodes to reduce the redundant transmissions. However, the proposed Multi-Level Data Aggregation Tree model with the Delay-Tolerant Scheduling algorithm for the Machine-to-Machine(M2M) communication shows better performance than the existing state-of-art solutions. Distributed algorithms can improve the solution for collecting data from sensor nodes and compressed sensing techniques to improve energy efficiency.

References

1. Ye Z, Mohamadian H (2014) Adaptive clustering based dynamic routing of wireless sensor networks via generalized ant Colony optimization. IERI Procedia 10:2–10
2. Zhou Z, Guo Y, He Y, Zhao X, Bazzi WM (2019) Access control and resource allocation for M2M communications in industrial automation. IEEE Trans Ind Inf 15(5):3093–3103
3. Manap Z, Ali BM, Ng CK, Noordin NK, Sali A (2013) A review on hierarchical routing protocols for wireless sensor networks. Wirel Pers Commun 72(2):1077–1104
4. Al-Kahtani MS (2015) Efficient cluster-based sleep scheduling for M2M communication network. Arab J Sci Eng 40(8):2361–2373
5. Nguyen N-T, Liu BH, Pham VT, Liou TY (2018) An efficient minimum-latency collision-free scheduling algorithm for data aggregation in wireless sensor networks. IEEE Syst J 12(3):2214–2225
6. Haque M, Ahmad T, Imran M (2018) Review of hierarchical routing protocols for wireless sensor networks, In Intell Commun Comput Tech, Springer. p. 237–246
7. Chen Q, Gao H, Cai Z, Cheng L, Li J (2018) Energy-collision aware data aggregation scheduling for energy harvesting sensor

networks. In IEEE INFOCOM 2018-IEEE Conference on Computer Communications, pp 117–125. IEEE
8. Zheng J, Cai Y, Shen X, Zheng Z, Yang W (2015) Green energy optimization in energy harvesting wireless sensor networks. IEEE Commun Mag 53(11):150–157
9. Lee DJ, Zhu Z, Toscas P (2015) Spatio-temporal functional data analysis for wireless sensor networks data. Environmetrics 26(5): 354–362
10. Mann PS, Singh S (2017) Energy-efficient hierarchical routing for wireless sensor networks: a swarm intelligence approach. Wirel Pers Commun 92(2):785–805
11. Beal J, Pianini D, Viroli M (2015) Aggregate programming for the internet of things. Computer 48(9):22–30
12. Rahman H, Ahmed N, Hussain I (2016) Comparison of data aggregation techniques in Internet of Things (IoT). In 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), pp 1296–1300. IEEE
13. Tsai S-Y, Sou S-I, Tsai M-H (2014) Reducing energy consumption by data aggregation in M2M networks. Wirel Pers Commun 74(4): 1231–1244
14. Brummet R, et al (2018) A flexible retransmission policy for industrial wireless sensor actuator networks. In 2018 IEEE International Conference on Industrial Internet (ICII). IEEE
15. Kliks A (2015) Application of the cognitive radio concept for M2M communications: practical considerations. Wirel Pers Commun 83(1):117–133
16. Akan OB, Karli OB, Ergul O (2009) Cognitive radio sensor networks. IEEE Netw 23(4):34–40
17. Toor AS, Jain A (2019) Energy aware cluster based multi-hop energy efficient routing protocol using multiple mobile nodes (MEACBM) in wireless sensor networks. AEU Int J Electron Commun 102:41–53
18. Chandirika B, Sakthivel N (2018) Performance Analysis of Clustering-Based Routing Protocols for Wireless Sensor Networks, in Advances in Big Data and Cloud Computing, Springer. p. 269–276
19. Haseeb K, Bakar KA, Abdullah AH, Darwish T (2017) Adaptive energy aware cluster-based routing protocol for wireless sensor networks. Wirel Netw 23(6):1953–1966
20. Kumar R, Kumar D (2016) Multi-objective fractional artificial bee colony algorithm to energy aware routing protocol in wireless sensor network. Wirel Netw 22(5):1461–1474
21. Ding M, Cheng X, Xue G (2003) Aggregation tree construction in sensor networks. In 2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall (IEEE Cat. No. 03CH37484). IEEE
22. Padmanabh K, Vuppala SK (2014) Method and system for adaptive aggregation of data in a wireless sensor network, Google Patents
23. Lu Y, Chen J, Comsa I, Kuonen P, Hirsbrunner B (2014) Construction of data aggregation tree for multi-objectives in wireless sensor networks through jump particle swarm optimization. Procedia Comput Sci 35:73–82
24. Randhawa S, Jain S (2017) Data aggregation in wireless sensor networks: previous research, current status and future directions. Wirel Pers Commun 97(3):3355–3425
25. Mohsenifard E, Ghaffari A (2016) Data aggregation tree structure in wireless sensor networks using cuckoo optimization algorithm. Inf Syst Telecommunication 4(3):182–190
26. John N, Jyotsna A (2018) A survey on energy efficient tree-based data aggregation techniques in wireless sensor networks. In 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), pp 461–465. IEEE
27. Le DT, Lee T, Choo H (2018) Delay-aware tree construction and scheduling for data aggregation in duty-cycled wireless sensor networks. EURASIP J Wirel Commun Netw 2018(1):1–15
28. Zhang Z, Li J, Yang X (2020) Data aggregation in heterogeneous wireless sensor networks by using local tree reconstruction algorithm. Complexity, 2020
29. Lu Y, Zhang T, He E, Comşa IS (2018) Self-learning-based data aggregation scheduling policy in wireless sensor networks. J Sens 2018:1–12
30. Kale PA, Nene MJ (2019) Scheduling of data aggregation trees using local heuristics to enhance network lifetime in sensor networks. Comput Netw 160:51–64
31. Sun Z, Wang H, Liu B, Li C, Pan X, Nie Y (2018) CS-FCDA: a compressed sensing-based on fault-tolerant data aggregation in sensor networks. Sensors 18(11):3749
32. Mohanty JP, Mandal C (2017) Connected Dominating Set in Wireless Sensor Network, in Handbook of Research on Advanced Wireless Sensor Network Applications, Protocols, and Architectures, IGI Global p 62–85
33. He J et al (2013) Constructing load-balanced data aggregation trees in probabilistic wireless sensor networks. IEEE Trans Parallel Distrib Syst 25(7):1681–1690
34. Riker A et al (2015) A two-tier adaptive data aggregation approach for m2m group-communication. IEEE Sensors J 16(3):823–835
35. Hussain S, Islam O (2007) An energy efficient spanning tree based multi-hop routing in wireless sensor networks. In 2007 IEEE Wireless Communications and Networking Conference, pp 4383–4388. IEEE

Prasad Challa is currently working as a Lecturer in department of Computer Science of *Visvodaya Govt. Degree College, Venkatagiri*, SPSR Nellore (Dt.) Andhra Pradesh,India. Also he is a research scholar in the department of CSE from JNTUA Ananthapuramu, A.P,India. He completed Post Graduation in computer science (M.Tech) from *National Institute of Technology,Tiruchirapalli*,Tamil Nadu, India. He has several publications in Computer Netwoks and WSNs.

Prof. **B. Eswara Reddy** graduated in B. Tech (Computer Science and Engineering) from Sri Krishna Devaraya University in 1995. He received Masters Degree in M.Tech. (Software Engineering) from JNT University, Hyderabad, in 1999. He received Ph.D in Computer Science & Engineering from JNT University, Hyderabad, in 2008. He has guided several research scholars for Ph.D. He served as Assistant Professor from 1996 to 2006 and as Associate Professor from 2006 to 2012. He is working as Professor of CSE Department in JNTUA College of Engineering, Anantapuram since 2012 and currently he is Director, Software Development Center in JNTUA Anantapuram. He has more than 121 Publications in various International Journals and Conferences. He is one of the author's of the textbooks titled '*Programming with Java*' published by Pearson/Sanguine Publishers and '*Data Mining*' published by Elsevier India. His research interests include Pattern Recognition & Image Analysis, Data Mining and Cloud Computing. He is a life member of ISTE, ISCA, CSI, Fellow IE (India) and IEEE.