A Beacon and GTS Scheduling Scheme for IEEE 802.15.4 DSME Networks

Nikumani Choudhury, *Member, IEEE,* Rakesh Matam, *Member, IEEE,* Mithun Mukherjee, *Senior Member, IEEE,* and Jaime Lloret, *Senior Member, IEEE*

***Abstract*—The IEEE 802.15.4 standard is one of the widely adopted networking specification for realizing different applications of Internet of Things (IoT). It defines several physical layer options and Medium Access Control (MAC) sub-layer protocols for low-power devices supporting low-data rates. One such MAC protocol is the Deterministic and Synchronous Multi-channel Extension (DSME), which addresses the limitation on maximum number of Guaranteed Time Slots (GTSs) in 802.15.4-2011 MAC, and provides channel diversity to increase network robustness. However, beacon scheduling in peer-to-peer networks suffers from beacon slot-collisions when two or more coordinators simultaneously compete for the same vacant beacon slot. In addition, the standard does not explore DSME-GTS scheduling across multiple channels. This paper addresses the beacon slot collision problem by proposing a non-conflicting beacon scheduling mechanism using association order. Further, a distributed multi-channel DSME-GTS schedule is proposed that optimally assigns DSME-GTSs across different channels. The objective is to minimize the number of times-lots used while maximizing the usage of available channels. Through simulations, the proposed mechanisms' performance is analyzed in terms of energy efficiency, transmission overhead, scheduling efficiency, throughput, and latency and is shown to out-perform the other existing schemes.**

***Index Terms*—IEEE 802.15.4, Internet of Things, DSME, DSME-GTS, beacon scheduling, multi-superframe.**

I. INTRODUCTION

INTERNET of Things (IoT) has emerged as one of the leading enabling technology that connects smart objects and devices over IP networks [1]–[4]. It has rapidly evolved over the years, supporting different real-time and industrial applications spanning home automation [5] to healthcare [6], smart buildings [7] to intelligent transportation [8], smart agriculture [9], smart cities [10], smart logistics [11], etc. Several of these applications require Quality of Service (QoS) in terms of low and deterministic latency, high throughput, scalability, and reliability. The IEEE 802.15.4 [12] standard is one of the widely adopted specifications for realizing different applications of IoT.

Corresponding author: Mithun Mukherjee, email: m.mukherjee@ieee.org

N. Choudhury is with Department of Computer Science & Information Systems, Birla Institute of Technology & Science, Pilani, Hyderabad Campus, Hyderabad, India (e-mail:nikumani@hyderabad.bits-pilani.ac.in).

R. Matam is with Department of Computer Science and Engineering, Indian Institute of Information Technology Guwahati, Guwahati, India (e-mail: rakesh@iiitg.ac.in).

M. Mukherjee is with the School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: m.mukherjee@ieee.org).

J. Lloret is with the Universitat Politecnica de Valencia, Spain (e-mail: jlloret@dcom.upv.es).

The IEEE 802.15.4-2011 [13] supports time-critical data by providing guaranteed bandwidth using its GTS mechanism [14]. However, a maximum of 7 GTS is permitted, thereby limiting the MAC protocol's scalability in real-time applications. The enhancement to this protocol, the IEEE 802.15.4e [15] and the recently ratified IEEE 802.15.4-2020 [12] (hereby referred to as IEEE 802.15.4) was able to address this shortcoming by defining MAC modes such as Time Slotted Channel Hopping (TSCH) and DSME [16]. Like the IEEE 802.15.4-2011 MAC, DSME facilitates both CSMA/CA and GTS to support the best effort and time-critical communications, respectively. The DSME MAC facilitates stringent delay and throughput requirements, including enhanced scalability and robustness through multi-channel access. The MAC mode operates on beacon-enabled mode [12], where the associated devices synchronize their transmissions using the beacon frames. These beacons coordinate a DSME multi-superframe structure that defines a devices' transmissions and channel access mechanism.

A. Multi-superframe Structure

Devices in a DSME network schedule their transmissions within a time structure known as multi-superframe. This is an extension of the IEEE 802.15.4 superframe structure [13]. The superframe has a total of sixteen timeslots (equal duration) of which, the first slot is used for the transmission of beacons. The next eight slots are used for transmissions using the CSMA/CA mechanisms. This transmission period is known as the Contention Access Period (CAP). This is followed by the Contention Free Period (CFP), wherein DSME-GTSs over multiple channels are used for transmissions. The length of the transmission period (consisting of CAP and CFP) is known as Superframe Duration (SD). The parameter Beacon Interval (BI) is defined as the time interval between two consecutive beacons. The BI consists of multi-superframes, and each of them contains multiple superframes. A new parameter defines a multi-superframe termed multi-superframeOrder (MO). MD is the multi-superframe duration, signifying the length of all the individual superframes in the multi-superframe. Two other superframe parameters macBeaconOrder (BO) and macSuperframeOrder (SO) together defines the structure of the multi-superframe, as shown in Fig. 1. These parameters are related to each other by: $0 \leq SO \leq MO \leq BO \leq 14$. The parameter values for BI, SD, and MD is derived by the following expressions,

$$\mathrm{BI} = \mathrm{aBaseSuperframeDuration} \times 2^{\mathrm{BO}} \tag{1}$$

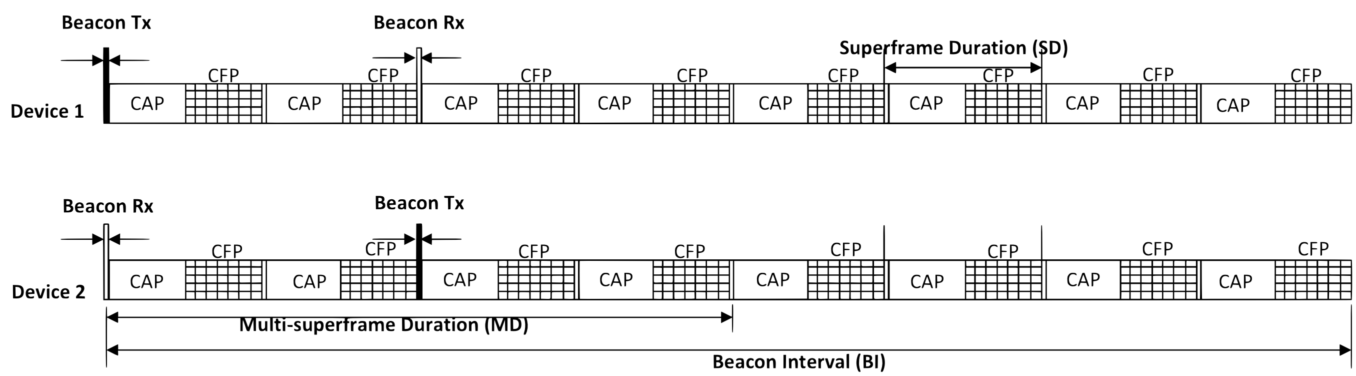


Fig. 1. DSME multi-superframe structure.

$$\mathrm{SD} = \mathrm{aBaseSuperframeDuration} \times 2^{\mathrm{SO}}, \tag{2}$$

$$\mathrm{MD} = \mathrm{aBaseSuperframeDuration} \times 2^{\mathrm{MO}}, \tag{3}$$

where aBaseSuperframeDuration is defined as the number of symbols constituting a superframe when the SO is set to zero. There are $2^{\mathrm{BO - SO}}$ superframes and $2^{\mathrm{BO - MO}}$ multi-superframes in the BI. The PAN coordinator sets the BO, SO, and MO parameter values during network initialization and is broadcast to associated devices through the Enhanced Beacons (EB) [12]. The EB is responsible for maintaining the synchronization in the network. From Fig. 1, it can be observed that a device transmits beacon during the Beacon Tx slots and a device receives beacons from another coordinator during a Beacon Rx slot. In Fig.1, initially Device 1 transmits beacon (Tx slot) and Device 2 receives the beacon from Device 1 in the Rx slot at the same time instant. At a later time instant, Device 2 transmits beacon in its Beacon Tx slot, and Device 1 receives the transmitted beacon in its Beacon Rx slot.

B. DSME MAC

Applications like industrial/factory automation and process control, health monitoring, and several commercial automation (home and smart building) can be sensitive to data loss or require stringent time constraints. In addition, all these applications demand high scalability and robustness. DSME addresses the needs for the applications with this kind of stringent QoS requirements. This is facilitated through channel hopping and channel adaptation techniques. It provides multi-channel access for all transmissions in the CFP period. Multi-channel access can be improvised through the CAP reduction technique [12]. When CAP reduction is applied, all CAPs except the one in the first superframe of the multi-superframe are substituted with DSME-GTSs [12]. DSME overcomes the issues related to interference from other wireless networks through proper coordination between the devices for channel usage. DSME MAC employs the channel hopping mechanism wherein the hopping sequence is pre-determined, and the same hopping pattern is repeated till the end of the data transmission. On the other hand, channel adaptation in DSME allows devices to hop over the channels based on the identified link quality.

In DSME beacon scheduling, each prospective coordinator initially performs a scan procedure over the available channels to receive the EBs from the neighboring coordinators. Each coordinator maintains a SD index table that contains SD information of the neighboring coordinators. Additionally, the SD index data is represented as a bitmap, included in a macSDBitmap field of the beacon frame. When a coordinator wants to select an empty beacon slot, it inspects the SD index from the received beacons. It selects a vacant slot, represented by ”0” in the received macSDBitmap. The coordinator then sets the corresponding bit to ”1” and broadcasts a DSME beacon allocation notification command frame to its neighbors. Upon receiving this frame, the neighboring coordinators verify whether any other neighboring coordinators are already using the selected SD index. If the slot is available for use, the neighboring coordinators update their individual SD index table accordingly. However, a collision might occur when two or more coordinators simultaneously attempt to choose the same empty slot. To address this problem, DSME uses an additional frame, a DSME beacon-collision notification command. The neighboring coordinators notify the prospective coordinators of the collision in slot selection. The beacon slot selection procedure is repeated in the next beacon cycle of the parent coordinator. Eventually, this procedure can avoid overlapping allocation of SD indexes among neighboring nodes. But, the major challenge herein is the amount of overhead incurred in carrying out the collision notification and repeated beacon slot allocation processes that impact the network lifetime.

C. Motivation

The presence of multiple beacon transmitting coordinators in p2p networks necessitates a mechanism to assign beacon slots in a non-overlapping fashion. A newly associated coordinator (or a device taking up the role of a coordinator) randomly selects an empty beacon slot from the received beacon bitmap (through the EB frames) from their neighboring devices [12] and advertises the selected slot. If the selected slot is also simultaneously chosen by another coordinator, it results in the beacon slot collision problem. Beacon collisions result in loss of synchronization between neighboring coordinators, leading to loss of data and incurring overhead in terms of transmission of control frames necessary for the association process. Additionally, this leads to an increase in energy dissipation by all the devices in the network. The standard defined mechanism of collision notification and re-selection of beacon selection is a energy-draining process and is not a fast network formation mechanism. Therefore, there is a necessity for a beacon slot selection scheme that allots non-overlapping beacon transmission slots to the coordinators and eliminates any slot selection conflict; thereby, addressing the overhead associated with such collisions.

Further, DSME-GTS scheduling is not discussed in the IEEE 802.15.4 standard. Optimally scheduling DSME-GTS slots across different channels is necessary to meet various QoS requirements. Such a QoS-efficient schedule should use the available resources stringently and fully utilize the multi-channel capabilities of DSME. In addition, the scheduling mechanism needs to have minimal overhead to the network. Prior works [17]–[19] in GTS scheduling are limited to a single channel for the IEEE 802.15.4-2011 MAC. Recent works [20]–[23] in DSME-GTS scheduling incurs high overhead by using routing layer information or simply assigning GTS in a vacant slot without considering resource utilization.

D. Contribution and Organization

This paper presents a distributed non-conflicting beacon scheduling mechanism that emphasizes eliminating beacon slot collision with no extra overhead to the network. We also propose a DSME-GTS scheduling mechanism that assigns DSME-GTSs across different channels. The proposed scheme minimizes the number of timeslots used while maximizing the usage of available channels, resulting in a QoS efficient schedule. The main contributions of the paper are summarized as follows:

- A Distributed Beacon Slot Selection (DBSS) scheme for IEEE 802.15.4 DSME based networks is proposed. We further analyse the schedulability of the proposed scheme and present low-overhead re-synchronization techniques to address the problem of loss in synchronization.
- We further present a DSME-GTS Scheduling (DGS) scheme that makes optimal usage of the available timeslots and fully utilizes the multi-channel capabilities of DSME.
- The performance analysis of the proposed DBSS and DGS mechanisms in terms of energy consumption, transmission overhead, throughput and latency is presented.

The rest of this paper is organized as follows. Section II describes the related works carried in DSME beacon and GTS scheduling. The network model is described in Section III. Section IV and Section V presents the proposed beacon slot selection and the DSME-GTS scheduling mechanisms, respectively. The simulation results are presented in Section VI. Finally, the conclusions are drawn in Section VII.

II. Related Research

The current specification of the standard was revised in 2015 and 2020, primarily focusing on throughput and latency [24]. In other words, it extends the functioning of IEEE 802.15.4-2011 to suit applications with stringent QoS requirements. The newly defined MAC modes like DSME guarantees stringent latency and throughput requirements including robust communication through multi-channel frequency hopping. In the IEEE 802.15.4-2011 MAC-based p2p networks, the beacon scheduling problem [25]–[28] have been well addressed. Due to the limitation of a single shared channel for transmissions, both the beacon and data transmission needs to be scheduled in a non-overlapping manner for coordinators (at least for 2-hop neighbors). The transmissions (active period of a coordinator, i.e., SD) are scheduled in the inactive period of its neighbors to minimize collisions. In contrast, the current specification of IEEE 802.15.4 allows multi-channel communication in the CFP (only) using the DSME-GTS mechanism. However, beacon transmissions (single channel in CAP) need to be scheduled so that no other adjacent coordinators are transmitting their beacons at the same instant of time (same timeslot). The absence of an optional sleep period (inactive period) for coordinators necessitates transmissions in the CAP period for all coordinators to occur simultaneously using the CSMA/CA mechanism. The multi-channel feature in the CFP presents the scope of effectively scheduling the DSME-GTS across the available channels.

Works like [16], [21], [29]–[35] have been carried out focusing on several aspects of the DSME MAC behavior and performance, but the problem of scheduling (both beacon and GTS) in DSME MAC has not been sufficiently explored. Hwang et al. [29] explored the problem of beacon slot selection and discussed several shortcomings in the DSME beacon slot selection technique that resulted in its performance degradation. The authors compared the DSME procedure with other selection methods like the Least Available Bit (LAB), Most Available Bit (MAB), and random schemes, but all these methods resulted in an excessive collision among the notification frames. In view of this, the authors defined a new DSME beacon scheduling that was based on limited permission procedure [29]. However, it addresses only the problem of collisions among the notification frames and not selecting the same beacon slot. Moreover, the proposed scheme restructures the IEEE 802.15.4 superframe structure to achieve collision reduction. In [34], the authors proposed a decentralized and low-complexity mechanism, termed Decentralized Beacon Scheduling Algorithm (DBSA), which was based on learning techniques. However, the network is still observed to suffer from beacon collisions prior to reaching a collision-free operation phase. In addition, the scheme relies on a feedback mechanism for several cycles incurring network overhead and considerable time prior to non-overlapping slot selection.

Symphony [20] is a routing aware algorithm that provides a schedule based on the routing information retrieved from the RPL [32] algorithm. The optimal assignment of timeslots and frequencies, which is done by Symphony is considered to be an NP-Hard problem. In [36], the authors build a centralized scheduling algorithm, termed as Maximum Dedicated Timeslot (MDT), aims for on-time delivery of data in WSNs prone to link failures. Several dummy GTSs slots were allocated to occupy the transmissions in case of a transmission failure. However, this approach can impact the overall delay of the network. The authors used a control layer

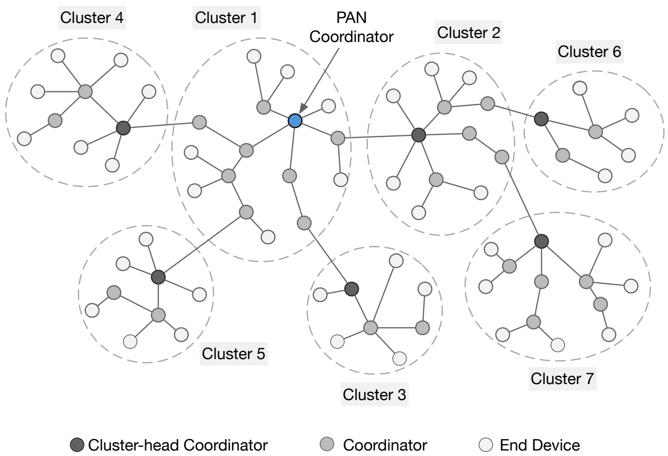


Fig. 2. Cluster-tree topology.

that schedules the timeslots and channel frequency during the CFP period. A Distributed Scheduling Algorithm (DSA) was proposed in [21] for cluster-tree network topologies, wherein the individual coordinators compute their transmission slots in the respective cluster. The devices within the cluster periodically exchanges control frames to gain knowledge about the queue length of the neighboring devices and the assigned transmission slots. It supports mobility and a neighboring stationary device is responsible for sharing the aforementioned information to the other devices in the cluster. Devices with higher number of data frames advertises a longer queue length and will have the right to steal some timeslots. In [23], the authors proposed a shared-DSME scheme in order to schedule GTSs as well as to enhance DSME-MAC scalability. In addition, the authors in [22] used linear programming to develop several scheduling mechanisms with an objective to reduce latency and energy. However, complex computations are challenging in the resource constrained IoT devices.

The aforementioned scheduling schemes incur high transmission overhead due to the maintenance of routing information [20], control frames [21], [36]. In addition, high computation overhead is observed in [20], [22]. In view of this, a light-weight distributed beacon and DSME-GTS scheduling mechanisms are proposed that optimizes the usage of the available resources.

III. Network Model

In this section, we present the network model that consists of coordinators and end-devices, associated in the form of a cluster-tree topology (as shown in Fig. 2). The topology is initiated by a coordinator known as the PAN coordinator (PANC), which acts as the root of the network. Coordinators allow other devices to associate with themselves (to scale up the network) by broadcasting periodic beacons and maintaining synchronization with their associated devices. On the other hand, end-devices associate with a coordinator and are not allowed to transmit beacons. Hence, they act as the leaf devices of the cluster-tree. The data sensed by these end-devices are directly transmitted to the associated parent coordinator. A cluster is formed by a group of coordinators and end-devices. In each of the clusters, a cluster-head (CH) is chosen among the available coordinators for data aggregation and transmission. The PANC also acts as the CH in the first cluster.

Algorithm 1: DBSS: Distributed Beacon Slot Selection

```
From received EB frames, retrieve the beacon bitmap
  and the AO for all the 2-hop neighbors ;
Compute β ;
if (β > 0) then
  | skip β vacant slots from the first empty slot in the
  |   bitmap;
  | Choose the next vacant slot for transmission ;
else
  | Choose the next vacant slot for transmission ;
end
```

A. Association Order

The association order (AO) of each coordinator is an integer value that represents the order in which the devices associate themselves to a parent coordinator. The AO is determined by a parent coordinator based on the time instant at which a child device associates, with coordinators associating first having a lower AO. For example, in a cluster-tree topology, the AO of the grandparent is 0, followed by the parent as 1 and {2, 3 or 4} for the peers (siblings), ordered based on their association time. The primary objective of AO is to facilitate coordinators in choosing beacon slots and channels avoiding conflict among the sibling coordinators.

The AO parameter uses the requisite bits within 0x03-0x7f reserved bits available in the association status field of the association response command in the MAC command frame, defined in the standard. This parameter is stored against the short address of the associated coordinator in its neighbor list (defined in the standard). The neighbor list is periodically updated through the received beacon frames from the neighboring coordinators. Each coordinator with its neighbor list is aware of the AO values of all its neighbors. The AO value of a coordinator can uniquely determine the hierarchy of association between coordinators. In this paper, it is used to avoid conflicts between coordinators in choosing beacon slots and transmission channels for DSME-GTS scheduling. In the proposed mechanisms, each coordinator keeps an account of the AO values of its 2-hop neighbors while performing the necessary computations (in Algorithm 1 and Algorithm 2).

IV. Proposed DSME Beacon Slot Selection Scheme

A Distributed Beacon Slot Selection (DBSS) mechanism is proposed based on the AO of the coordinators. A newly associated coordinator or a device that has taken up the coordinator's role retrieves the beacon bitmap (through the received EB frames) from their neighboring devices. The beacon schedule information is expressed in a bitmap sequence representing the schedule of EB frames transmitted from all neighboring

TABLE I
AO Configuration.

Coordinator	AO
A	0
B	1
C	2
D	3
E	4
F	5

devices. The corresponding bit in the bitmap shall be set to one if a beacon is occupied in that beacon slot. It chooses an empty beacon slot according to its AO within its 2-hop neighborhood. That is, rather than randomly selecting an empty slot (that may result in slot conflicts), beacon slots are chosen based on the AO of the coordinator. Therefore, no two or more coordinators simultaneously select the same vacant slot. Once the new coordinator selects a vacant slot, it uses the slot as its own beacon slot.

Algorithm 1 is executed by any coordinator to evaluate its beacon transmission slot. The coordinator chooses an empty beacon slot, skipping unselected/vacant slots for coordinators with lower AOs than itself if any. Note that β is applicable only in scenarios wherein a coordinator is yet to choose an empty beacon slot and another coordinator (with higher AO) tries to select one such slot. This is done to have an uniformity in beacon slot selection with respect to the association order of the coordinators. The number of such empty slots skipped by the coordinator is given by β, which is defined as the number of other coordinators (having lower AO) with no slot in the beacon bitmap. A coordinator through its neighbor table (defined in the standard) is aware of the number of coordinators having lower AO and the number of the coordinators occupying beacon slots in the received beacon bitmap. The difference in the number of such coordinators is the β value.

A. Illustrative Example

We consider an illustrative example to understand the working of the proposed DBSS scheme as shown in Fig. 3. Let coordinators A, B, C, D, E, and F be 2-hop neighbors. Coordinators E and F try to select a vacant beacon transmission slot simultaneously. The rest of the coordinators transmit beacons as per the schedule, as shown in Fig. 4a. The DBSS mechanism ensures that E and F choose different vacant slots by following the neighboring coordinators' AO, i.e., F selects a vacant slot after reserving a slot for E. This is shown in Fig. 4b. The final beacon bitmap schedule is shown in Fig. 4c.

B. Schedulability

Schedulability refers to the number of coordinators that can have non-overlapping beacon transmissions within a single BI. A higher schedulability signifies that a given beacon scheduling scheme schedules many coordinators without any transmission overlap. In general, distributed schemes would have higher schedulability compared to centralized schemes. This is because distributed schemes simultaneously schedule

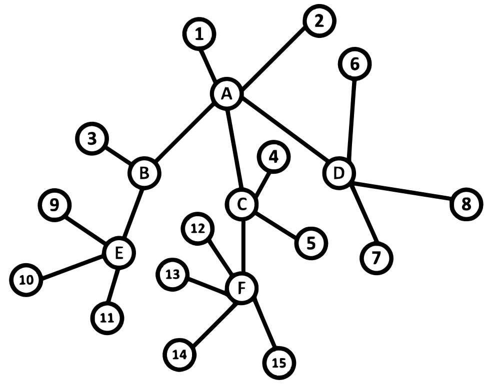
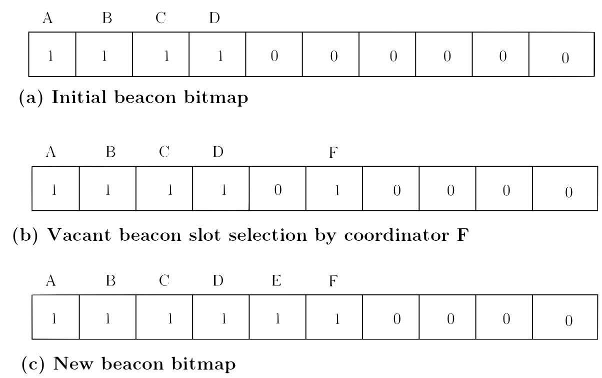


Fig. 3. Illustrative example



(a) Initial beacon bitmap

(b) Vacant beacon slot selection by coordinator F

(c) New beacon bitmap

Fig. 4. DBSS: Beacon transmission schedule.

the beacon frames of different coordinators at least 2-hop away. Therefore, the beacon scheduling schemes limit the scheduling to every node's 2-hop neighbors. This leads to a higher number of coordinators' beacons being scheduled in a non-overlapping manner outside their collision domain, thereby increasing the network's overall beacon schedulability.

The proposed DBSS mechanism schedules the beacon frame transmissions based on the AO of the coordinators within a 2-hop neighborhood. Whereas, DBSA [34] scheme is centralized in nature, thereby having a limitation on the schedulability. Although the E-DSME [29] scheme is distributed in nature, the probability of beacon collisions during the simultaneous request for slot allocation reduces the scheme's schedulability. On the other hand, the proposed scheme, through the use of AO, nullifies beacon collisions within the 2-hop neighborhood.

C. Addressing Re-synchronization Issues

The coordinators in a network can lose synchronization when the devices may simply need to recompute their synchronized schedules due to changes in the network topology. In dynamic network topology, topology changes may arise from various factors that include new devices taking up a coordinator's role or due to malfunctioning of the existing coordinator. Each time such a topology change occurred, the need for re-synchronization is observed. This results in attempts to re-synchronize the network, typically with the same beacon scheduling mechanism that was employed during network setup. The energy overhead associated with the beacon scheduling mechanism is added up each time the network encounters a loss in synchronization.

The DBSS scheme does not require the coordinators to recompute their beacon transmission schedule. If the synchronization loss is due to some network issues and the topology remains intact, the coordinators can continue with their original beacon schedule. However, if synchronization was lost due to changes in network topology, the coordinators require to adjust their beacon transmission slot as per their AO. If a new coordinator joins the network or a device newly takes up a coordinator's role, it receives the highest AO in its 2-hop neighborhood. Thus, it schedules its beacon transmission slot only after other coordinators. Therefore, the rest of the coordinators' synchronization in the network is not affected by a new coordinator joining a network.

On the other hand, when a coordinator parts away from the network topology, maybe due to device malfunction or complete power drainage, synchronization is affected. The dissociation of a node is typically detected by its associated coordinators if they fail to receive periodic beacons from the dissociated node. Depending upon the dissociated coordinator's hierarchy (relative to a coordinator), the rest of the coordinator adapts accordingly, i.e., the employed mechanism varies.

1) If the dissociated coordinator is hierarchically a grand-parent, then all the coordinators under its sub-tree need to adjust their beacon slots in the bitmap. The coordinators shift left by one slot in the new beacon bitmap to maintain synchronization and avoid vacant slots.
2) If the dissociated coordinator is hierarchically a parent coordinator, then all its child coordinators shift left by one slot in the new beacon bitmap. The grand-parent (parent of the dissociated coordinator) continues with its original transmission slot.
3) If the dissociated coordinator is a child (sibling) coordinator, it results in all the other siblings (if any) with higher AO left shifting their transmission slots by one. Rest all the coordinators continue transmitting on their original beacon bitmap slots.
4) Finally, if a leaf coordinator dissociates, none of the coordinators need to adjust their transmission slots. This is because the dissociated coordinator had the highest AO and its slot was at the rightmost in the beacon bitmap.

Since the proposed re-synchronization mechanism is a distributed scheme, each of the affected coordinators (coordinators that needs to change beacon slots) shifts left by one slot according to the beacon bitmap. Thus, the proposed re-synchronization technique enables the overall network to reach a stable state after the dissociation of a node.

V. Proposed DSME-GTS Scheduling Scheme

The DSME-GTSs can be scheduled across multiple available channels. This paper proposes a DSME-GTS scheduling mechanism where the devices are optimally assigned DSME-GTS slots based on their AO. Specifically, channel assignments among the coordinators in a 2-hop neighborhood are based on their AO. The DGS scheme is built upon the following premises.

i No two 2-hop neighbors are assigned the same channel.
ii A parent coordinator (assuming single transceiver) can receive data from only one child at a time.
iii A channel can be assigned to every child device for parent-child communication.
iv Different channels can be assigned for different pairs of parent-child devices.
v Transmissions arising from a coordinator and towards that coordinator should be placed in different or adjacent slots.

The DGS scheme can be divided into two steps, namely, the channel assignment step followed by the identification of the blacklisted timeslot.

A. Channel Assignment

In the channel assignment step, the AO of the coordinators are used to assign a particular channel to each of the coordinators. This channel is used by the immediate child devices for DSME-GTS transmissions. Let $N_{\text{2-hop}}$ be the total number of coordinators in a 2-hop neighborhood, and n_{chan} be the total number of available channels. Therefore, channel assignment for each coordinator, i, is expressed as,

$$F(i) = AO_i \pmod{n_{chan}} \tag{4}$$

The function $F(i)$ gives us a subset of coordinators for each of the available channels.

B. Blacklisted Timeslot

Once a timeslot is assigned to a coordinator for transmission, the same timeslot (across all the channels) cannot be further used for any communication by that coordinator. We define such timeslots as *blacklisted* for the particular coordinator. Note that the blacklisted timeslot is relevant only to the pair of coordinators involved in communication. The blacklisted timeslot is however, available to other coordinators and end-devices for DSME-GTS transmissions.

C. DGS Algorithm

Algorithm 2 presents the proposed DGS mechanism that is executed at each coordinator in the network. Initially, a coordinator determines a channel for DSME-GTS allocations for its child devices as per Eq. (4). This channel will be used by its associated devices to transmit data frames using DSME-GTS. The assigned channel is communicated to all its neighboring devices through the EB frames. A coordinator tries to allocate a single DSME-GTS to all its associated child devices. This is achieved by allocating timeslots sequentially (sorted according to AO) in the selected channel. Note that the CFP contains at most 7 timeslots per channel; therefore, only 7 DSME-GTS can be scheduled by each coordinator. However, in a 2-hop neighborhood, if the number of coordinators is higher than the number of available channels, Eq. (4) will assign the same channel for different coordinators. In such

Algorithm 2: DGS: DSME-GTS Scheduling

```
chan_i = 0;
Compute F = AO_i % n_chan ;
while (chan_i <= n_chan) do
    if (chan_i == F) then
        for each d_i ∈ n_i_child do
            while (DSME-GTS is scheduled) do
                if (time_slot <= 7) then
                    if (time_slot != black_listed_i) then
                        Find the first empty timeslot.
                        Allocate timeslot for
                          DSME-GTS transmissions.
                    end
                    else
                        time_slot ++ ;
                    end
                end
                else
                    F ++ ;
                end
            end
        end
    end
    chan_i++ ;
end
```

a case, DSME-GTSs will be scheduled in the remaining timeslots of the assigned channel. Suppose timeslots are not available or insufficient for scheduling (in the selected channel) all its associated devices. In that case, the remaining allocations can be made in the adjacent channel following the same procedure. All DSME-GTS scheduling must consider the blacklisted timeslots prior to allocations. Whenever a blacklisted timeslot is encountered for a particular device, it is skipped, and the next timeslot is selected. This procedure is repeated until the schedule is complete for all child devices for the coordinator. For the coordinators with coinciding channels, the coordinator with higher AO schedules transmissions of its child devices only after considering the schedules of the coordinator with lower AO. This is achieved by collecting the total number of child devices of such coordinators (coinciding channels) with lower AO.

It may be observed that coordinators with low AO may occupy more timeslots than others as AO sorts the schedule. This can be justified because higher transmissions are expected in coordinators with low AO in a cluster-tree network topology.

D. Illustrative Example

We consider a cluster-tree topology with coordinators A,B,C,D,E,F, and end-devices 1, 2, …, 14, 15, as shown in Fig. 3. Let the total number of available channels for operation be 5:{0, 1, 2, 3, 4}. The assigned channels for the coordinators are {A:0, B:1, C:2, D:3, E:4, F:0}. Coordinator A schedules DSME-GTS to its child devices (B, C, D, 1, 2) starting from the channel-0, timeslot-1 as shown in Fig. 5 Next, coordinator B schedules GTS to its associated devices

Channels \ Timeslots	1	2	3	4	5	6	7	
0	B->A	C->A	D->A	1->A	2->A	12->F	13->F	A, F
1		E->B	3->B	14->F	15->F			B
2	F->C		4->C	5->C				C
3	6->D	7->D		8->D				D
4	9->E		10->E	11->E				E

Fig. 5. Illustrative example: DGS schedule.

TABLE II
SIMULATION PARAMETERS

Parameters	Values
Beacon order	8
Superframe order	2
Multisuperframe order	2
Number of channels	12
Frame Length	75 Bytes
Number of nodes	50
Data rate	250 kbps

in channel 1. However, the first timeslot (T1) is a blacklisted channel as it is allotted by A in channel 0. Therefore, T1 is skipped, and coordinator B starts scheduling from T2 onwards. Similarly, coordinators, C, D, and E schedules DSME-GTS for its associated devices. Finally, coordinator F, which was assigned channel 0, schedules its end devices (12, 13, 14, 15) at T6, T7 in channel 0, followed by T4 and T5 in channel 1. Coordinator F is aware of the channels assigned for A and B as every coordinator communicates its channel number to all its 2-hop neighbors through the beacon frames. Hence, F selects empty slots after considering the slots occupied by the child devices of coordinators A and B.

VI. SIMULATION RESULTS

The Network Simulator OMNeT++ [37] is used to evaluate the performance of the proposed schemes. OpenDSME [38], is used to implement the IEEE 802.15.4 DSME MAC mode in OMNeT++. The parameters that are used in the experiments are listed in Table II. We considered a cluster-tree comprising of seven clusters (Fig.2) covering a radius of $1000\,\mathrm{m} \times 1000\,\mathrm{m}$. The 2.4GHz frequency band is selected as it provides the maximum data rate of 250kbps [13]. The simulation duration is fixed at $3000\,\mathrm{seconds}$ duration. For any given combination of simulation parameters, we ran 30 different simulations and finally averaged over all the 30 different results. The performance metrics considered are: 1) transmission overhead, 2) successful allocation, 3) energy consumption, 4) channel utilization, 5) scheduling efficiency, 6) latency, and 7) MAC goodput. We compare the proposed DBSS mechanism with IEEE 802.15.4 DSME scheme, E-DSME scheme [29], and DBSA [34]. The proposed DGS scheme is compared with schemes like Symphony [20], MDT scheduling [36], and DSA [21].

A. Transmission Overhead

Beacon slot selection and DSME-GTS scheduling mechanisms are necessary to effectively assign beacon slots and

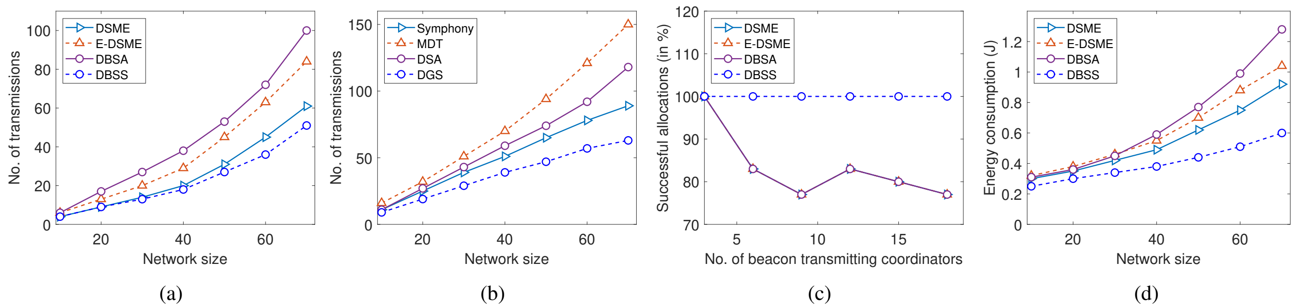


Fig. 6. (a) Transmission count for beacon slot scheduling schemes, (b) transmission count for DSME-GTS scheduling schemes, (c) successful beacon slot allocation, and (d) energy consumption of various beacon slot scheduling schemes.

DSME-GTS slots across the channels, respectively. However, the goal is to achieve this with minimal overhead to the network. Firstly, we compare the transmission count of different beacon slot selection schemes (as shown in Fig. 6(a)), and later, the number of transmissions required for the DSME-GTS scheduling schemes (shown in Fig. 6(b)). From Fig. 6(a), it can be observed that DBSS has the lowest transmission overhead arising from the transmission of control frames in achieving non-overlapping beacon transmission slots for all the coordinators in the network. This is because the DBSS mechanism relies upon the AO to avoid any conflict in empty beacon slot selection. Therefore, unnecessary beacon slot conflicts and transmission of control frames arising from such slot conflicts are averted. On the other hand, the standard defined DSME MAC suffers from beacon slot conflicts which are then resolved through transmissions of conflict notification frames resulting in higher transmission overhead. Similarly, beacon slot scheduling schemes like E-DSME and DBSA also suffer from the higher transmission and network overhead. These schemes are involved in frequent message exchanges to address the beacon slot conflicts. These schemes repeat the process of slot selection with either a limited number of coordinators or learning techniques that involve considerable transmission overhead.

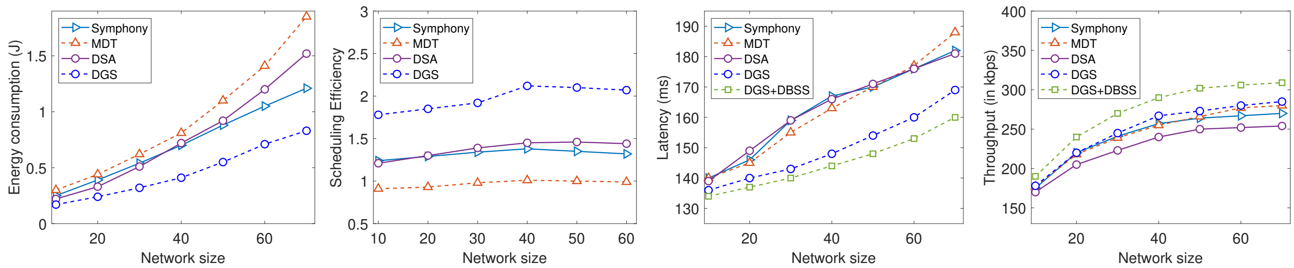
Fig. 6(b) shows that on scaling up the network, Symphony, DSA, and MDT scheduling schemes suffer from high transmission overhead. Herein, all the coordinators exchange control frames for maintaining knowledge about their queue length, the slots assigned and routing information. In Symphony, routing information is collected through the use of RPL in the higher layer, which also contributes to the transmission overhead. In contrast, the proposed DGS mechanism relies upon the AO to select a channel, and the assigned channel is communicated to all its neighboring devices through the EB frames. A transmission overhead is incurred in collecting the number of child devices of coordinators (lower AO only) with coinciding channels. However, such cases' probability is very low as the total number of available channels (27 channels in total) is generally higher than the number of coordinators in a 2-hop neighborhood, except for dense network topologies.

B. Successful Allocation

The main objective of any beacon slot selection mechanism is to successfully allot non-overlapping slots for beacon transmission, i.e., no transmission conflicts with other coordinators. In Fig. 6(c), we show the percentage of successful beacon slot allocation (in the first attempt) for coordinators without any slot conflict and collisions. The proposed DBSS mechanism always successfully allocates beacon transmission slots to all the coordinators. This is possible with the assistance of AO that guarantees that no two coordinators compete for the same transmission slot. On the other hand, DSME MAC, E-DSME, and DBSA schemes focus on the process of re-computing a beacon transmission slot following a beacon slot conflict notification. Hence, such schemes initially suffer from slot conflicts which may or may not be addressed in the next attempt to allocate a transmission slot. The percentage of unsuccessful slot allocations is the same for all the aforesaid schemes (except the proposed scheme) as two or more coordinators simultaneously competing to select a slot invariably selects the same vacant slot resulting in slot conflicts. The respective schemes then propose measures to address the issue. The slot selection mechanism is repeated several times until non-conflict slots are allotted. Operating these schemes additionally increases the transmission and network overhead.

C. Energy Consumption

Scheduling schemes can significantly reduce energy-savings by facilitating collision-free beacon slot selection and transmissions. This contributes to improving the network lifetime, which is one of the primary design goals for DSME MAC. But, these schemes in themselves should not incur high overhead in the network in terms of energy consumption [39]. As the transmission count is directly proportional to the energy consumption, it can observed that both the proposed proposed DBSS and DGS mechanisms consume the least energy compared with their respective scheduling schemes. Fig. 6(d) shows that the DBSS scheme achieves 8% energy savings compared to the other related schemes. This can be attributed to fewer transmissions of control messages as discussed in the subsection VI-A. Further, schemes like E-DSME and DBSA spend excess energy on transmissions arising from slot conflicts. Additional energy consumption is observed from beacon collisions and thereby loss in synchronization.



(a) (b) (c) (d)

Fig. 7. (a) Energy consumption of various DSME-GTS scheduling schemes, (b) comparison of scheduling efficiency, (c) latency comparison of various scheduling schemes, and (d) throughput comparison of various scheduling schemes.

From Fig. 7(a) it can be observed that the DGS scheme achieves 3-7% energy savings compared to Symphony, DSA, and MDT scheduling schemes. To schedule the DSME-GTSs, Symphony acquires the routing information (using RPL) that results in significant energy consumption. Schemes like DSA and MDT suffer from higher energy consumption than the proposed mechanism as these schemes incur high transmissions in maintaining the knowledge about the slots assignments of other coordinators and routing information. In contrast, the proposed scheme achieves higher energy savings through fewer transmissions among the coordinators to build a DSME-GTS schedule. Coordinators independently choose a channel according to their AO, and this information is broadcast through the beacon frames. Energy dissipation through transmissions is observed during the collection of child devices with coinciding channels.

D. Scheduling Efficiency

We define scheduling efficiency as the optimal usage of available resources, i.e., minimal timeslots and maximum channels, with respect to the allocation of DMSE-GTS slots. This helps us achieve a reduction in channel wastage, and the overall network throughput and scalability can be significantly increased as well as simultaneously minimize latency [20]. The transmission overhead associated with the scheduler is also accounted for as given in Eq. (5) below:

$$\text{Scheduling efficiency}, \eta = \frac{\text{GTS allotted} \times \text{Channels used}}{\text{Timeslots used} \times \text{Overhead}}, \tag{5}$$

where overhead refers to the transmission overhead incurred by a scheduling scheme. The number of DSME-GTS allotted by a scheduling scheme is one of the primary parameters to infer scheduling performance. However, optimal usage of timeslots and available channels significantly contributes to the scheduling efficiency. It is desired that the scheduling scheme should maximize the use of available channels, i.e., distribute the DSME-GTS slots allotments over different channels considering non-overlapping transmissions among the neighboring devices. Also, the scheduling scheme should aim to optimize the use of timeslots while allotting DSME-GTSs.

Table III shows the details of channels and timeslots used by each of the considered scheduling schemes along with the number of DSME-GTS slots allotted for 20 devices in the network topology. Fig. 7(b), shows the comparison of the scheduling efficiency of DGS, Symphony, MDT, and DSA schemes with varying network size. The proposed scheme successfully distributes the available channels among the coordinators and their child devices with respect to their AO parameter, irrespective of the network's size. The MDT scheme focuses on scheduling DSME-GTS slots as substitute links for transmission failure. Therefore, the scheduling efficiency is lower than the other schemes. Also, the other schemes like Symphony and DSA suffer from network overhead on scaling up the network size. This is due to the higher number of control frames exchanged among the coordinators to maintain the knowledge of the slot assignments, routing (RPL), and queue length of the devices. It is interesting to note that DSA allows coordinators to utilize (steal) DSME-GTS slots allotted to other devices if the former has a higher queue length. This allows the network to build an traffic-aware schedule over a period of time. Therefore, with an increase in network size, the scheduling efficiency of DSA improves over Symphony.

TABLE III
SCHEDULING DETAILS OF DIFFERENT SCHEDULING SCHEMES FOR A NETWORK WITH 20 DEVICES.

Scheduling Scheme	DSME-GTS	Timeslots	Channels
Symphony	84	15	7
MDT	79	19	7
DSA	89	14	7
DGS	89	12	7

E. Latency and MAC goodput

Latency and throughput are one of the essential performance metrics of scheduling schemes as it gives a measure of the time consumed in transmission, and successfully transmitted bits. Fig. 7(c) and Fig. 7(d), depicts the latency and total MAC goodput for all the DSME-GTS scheduling schemes, respectively. Latency can be minimized if several transmissions can be scheduled in the same timeslot but with different channels. That is, optimizing the channel allocations or utilizing the full potential of the available channels. Latency is further minimized through synchronization among the coordinators as this avoids unnecessary collisions during data transmissions.

Hence, we combine the proposed DBSS and DGS mechanisms to get maximum benefits in QoS performance. On the other hand, we implemented the standard defined DSME beacon slot selection mechanism for Symphony, DSA, and MDT scheduling mechanism. Additionally, we considered a scenario with DGS and the basic beacon slot selection mechanism. Symphony and DSA have higher latency than the proposed scheme as the former schemes' scheduling efficiency are lower, resulting in sub-optimal channel usage. Further, these schemes suffer from beacon slot collisions resulting in a loss in synchronization, thereby degrading the QoS performance of the network in terms of latency and throughput.

Throughput is an important parameter to measure the performance of the proposed scheduling mechanism. It can be seen from Fig.7(d), that the throughput of the network improves when both the proposed mechanisms are used together. Higher throughput is observed for the proposed mechanisms as the schemes complement each other to provide an efficient DSME-GTS schedule and collision-free beacon and data transmissions (in the CAP). Symphony, MDT, and DSA suffer from possible collisions at the beginning of network formation or during network topology changes, resulting in lower throughput. On the other hand, MDT allocates extra available slots for links encountering failures due to interference or network error. This helps to maintain a steady network throughput.

VII. Conclusion

The paper addresses the problem of slot conflict during beacon slot selection in DSME based cluster-tree networks. The proposed DBSS mechanism facilitates a coordinator to chose a non-conflict vacant slot for beacon transmission by using its AO. It relies on the beacon bitmap received from its 2-hop neighboring devices' EB frames to select the transmission slot. The proposed scheme does not add extra overhead to the network in terms of control frames. In addition to the above, we also propose a DSME-GTS scheduling scheme that assigns GTSs over different channels considering several factors like optimal resource utilization (minimum timeslots and maximum channels) and interference from neighboring devices. The channel assignments for the coordinators are based on their AO. The proposed mechanism minimizes channel switching overhead by assigning channels for every coordinator. Most importantly, the scheduling efficiency of the proposed DGS mechanism is higher compared to other related schemes.

As part of the future work, we will expand our experiments to a testbed consisting of a cluster-tree network topology and implement the proposed scheduling mechanisms. The hardware, known as M3OpenNode, that consists of an Atmel AT86RF231 radio chip and an ARM Cortex M3 STM32F103REY will be used. The Contiki OS with 6LoWPAN stack will be used by replacing the existing CSMA/CA MAC with the openDSME implementation. We further aim to consider delay constraints and flow deadlines while developing schedules for applications with strict latency requirements.

References

[1] M. Moness and A. M. Moustafa, "A Survey of Cyber Physical Advances and Challenges of Wind Energy Conversion Systems: Prospects for Internet of Energy," *IEEE Internet Things J.*, vol. 3, no. 2, pp. 134–145, Apr. 2016.

[2] T. A. Al-Janabi and H. S. Al-Raweshidy, "An Energy Efficient Hybrid MAC Protocol with Dynamic Sleep-Based Scheduling for High Density IoT Networks," *IEEE Internet Things J.*, vol. 6, no. 2, Apr. 2019.

[3] J. A. Stankovic, "Research Directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[4] A. Moravejosharieh and J. Lloret, "Performance Evaluation of Co-located IEEE 802.15.4-based Wireless Body Sensor Networks," *Annals of Telecommun.*, vol. 71, no. 9-10, pp. 425–440, Oct. 2016.

[5] D. Marikyan, S. Papagiannidis, and E. Alamanos, "A systematic review of the smart home literature: A user perspective," *Elsevier Technological Forecasting and Social Change*, vol. 138, pp. 139–154, Jan. 2019.

[6] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26 521–26 544, Nov. 2017.

[7] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, and B. B. Gupta, "Efficient iot-based sensor big data collection–processing and analysis in smart buildings," *Elsevier Future Generation Computer Systems*, vol. 82, pp. 349–357, May 2018.

[8] P. Šolić, A. Leoni, R. Colella, T. Perković, L. Catarinucci, and V. Stornelli, "Iot-ready energy-autonomous parking sensor device," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4830–4840, Oct. 2021.

[9] N. Ahmed, D. De, and I. Hussain, "Internet of Things (IoT) for Smart Precision Agriculture and Farming in Rural Areas," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4890–4899, Dec. 2018.

[10] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[11] Y. Song, F. R. Yu, L. Zhou, X. Yang, and Z. He, "Applications of the internet of things (iot) in smart logistics: A comprehensive survey," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4250–4274, Oct. 2021.

[12] "IEEE Std. 802.15.4- Low-Rate Wireless Networks," http://www.ieee.org/Standards/, pp. 1–800, July. 2020, [Online accessed: 25-Aug-2020].

[13] "IEEE Std. 802.15.4- Low-Rate Wireless Personal Area Networks (WPANs)," http://www.ieee.org/Standards/, pp. 1–314, Sept. 2011, [Online accessed: 25-Jan-2020].

[14] A. Moravejosharieh and J. Lloret, "A Survey of IEEE 802.15.4 Effective System Parameters for Wireless Body Sensor Networks," *Intl. J. Commun. Syst.*, vol. 29, no. 7, pp. 1269–1292, Dec. 2016.

[15] "IEEE Std. 802.15.4:2015- Low-Rate Wireless Networks, IEEE Standard ," http://www.ieee.org/Standards/, pp. 1–709, Apr. 2016, [Online accessed: 25-Jan-2020].

[16] N. Choudhury, R. Matam, M. Mukherjee, and J. Lloret, "A Performance-to-Cost Analysis of IEEE 802.15.4 MAC With 802.15.4e MAC Modes," *IEEE Access*, vol. 8, pp. 41 936–41 950, Feb. 2020.

[17] C. Na, Y. Yang, and A. Mishra, "An optimal GTS scheduling algorithm for time-sensitive transactions in IEEE 802.15.4 networks," *Computer Networks*, vol. 52, no. 13, pp. 2543–2557, Sept. 2008.

[18] Y. Ding and S. H. Hong, "CFP scheduling for real-time service and energy efficiency in the industrial applications of IEEE 802.15.4," *J. Commun. Netw.*, vol. 15, no. 1, pp. 87–101, Mar. 2013.

[19] Y.-K. Huang, A.-C. Pang, and H.-N. Hung, "An adaptive GTS allocation scheme for IEEE 802.15.4," *IEEE Trans. Parallel Distr. Sys.*, vol. 19, no. 5, pp. 641–651, Mar. 2008.

[20] H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "Symphony: routing aware scheduling for DSME networks," *ACM Sigbed Review*, vol. 16, no. 4, pp. 26–31, Jan. 2020.

[21] Y.-S. Lee and S.-H. Chung, "An Efficient Distributed Scheduling Algorithm for Mobility Support in IEEE 802.15.4e DSME-based Industrial Wireless Sensor Networks," *Intl. J. Distrib. Sensor Netw.*, vol. 12, no. 2, pp. 1–14, Feb. 2016.

[22] F. Meyer and V. Turau, "Delay-Bounded Scheduling in IEEE 802.15.4e DSME Using Linear Programming," in *International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2019, pp. 659–666.

[23] F. Battaglia, M. Collotta, L. Leonardi, L. L. Bello, and G. Patti, "A scalable approach for periodic traffic scheduling in IEEE 802.15.4-DSME networks," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, July 2019, pp. 1747–1750.

[24] J. Wu, S. Guo, J. Li, and D. Zeng, "Big Data Meet Green Challenges: Big Data Toward Green Applications," *IEEE Syst. J.*, vol. 10, no. 3, pp. 888–900, Sept. 2016.

[25] A. Koubâa, A. Cunha, M. Alves, and E. Tovar, "TDBS: A Time Division Beacon Scheduling Mechanism for ZigBee Cluster-tree Wireless Sensor Networks," *Springer Real-Time Syst.*, vol. 40, no. 3, pp. 321–354, Dec. 2008.

[26] N. Choudhury, R. Matam, M. Mukherjee, and J. Lloret, "LBS: A Beacon Synchronization Scheme with Higher Schedulability for IEEE 802.15.4 Cluster-tree Based IoT Applications," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8883–8896, Oct. 2019.

[27] B. Nefzi, D. Khan, and Y.-Q. Song, "TBoPS: A Tree based Distributed Beacon Only Period Scheduling Mechanism for IEEE 802.15.4," in *Proc. IEEE Intl. Conf. Distrib. Comput. Sensor Syst.*, May 2012, pp. 341–346.

[28] P. Muthukumaran, R. de Paz, R. Spinar, and D. Pesch, "MeshMAC: Enabling Mesh Networking Over IEEE 802.15.4 through Distributed Beacon Scheduling," in *Proc. Intl. Conf. on AdHoc Networks*, Sept. 2009, pp. 561–575.

[29] K.-i. Hwang and S.-w. Nam, "Analysis and enhancement of ieee 802.15. 4e dsme beacon scheduling model," *Journal of Applied Mathematics*, vol. 2014, May 2014.

[30] P. Sahoo, S. Pattanaik, and S.-L. Wu, "A Reliable Data Transmission Model for IEEE 802.15.4e Enabled Wireless Sensor Network under Wifi Interference," *Sensors*, vol. 17, no. 6, p. 1320, June 2017.

[31] G. Alderisi, G. Patti, O. Mirabella, and L. L. Bello, "Simulative Assessments of the IEEE 802.15.4e DSME and TSCH in Realistic Process Automation Scenarios," in *Intl. Conf. on Ind. Informat.*, Oct. 2015, pp. 948–955.

[32] H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "RPL over DSME: A Technical Report," CISTER-Research Centre in Realtime and Embedded Computing Systems, Tech. Rep., 2018. [Online]. Available: http://www.cister.isep.ipp.pt/docs/rpl_over_dsme__a_technical_report/1347/view.pdf.

[33] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "An Efficient Approach to Multisuperframe Tuning for DSME Networks," in *ACM/IEEE Intl. Conf. Inf. Processing in Sensor Netw.*, Oct. 2018, pp. 162–163.

[34] C. Cano and D. Malone, "A learning approach to decentralised beacon scheduling," *Ad Hoc Networks*, vol. 49, pp. 58–69, June 2016.

[35] F. Kauer, M. Köstler, T. Lübkert, and V. Turau, "Formal analysis and verification of the ieee 802.15. 4 dsme slot allocation," in *Proc. ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Nov., pp. 140–147.

[36] J. Lee, W. Jeong, and B. Choi, "A multi-channel timeslot scheduling algorithm for link recovery in wireless multi-hop sensor networks," in *International Conference on Information and Communication Technology Convergence (ICTC)*, Oct. 2016, pp. 871–876.

[37] "OMNeT++ Network Simulator," https://omnetpp.org/download/, [Online accessed: 25-Jan-2020].

[38] "OpenDSME: Open Source Portable Implementation of IEEE 802.15.4 DSME," https://github.com/openDSME/inet-dsme/, [Online accessed: 25-Jan-2020].

[39] N. Choudhury, R. Matam, M. Mukherjee, and L. Shu, "Beacon Synchronization and Duty-cycling in IEEE 802.15.4 Cluster-tree Networks: A Review," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1765–1788, June 2018.

Nikumani Choudhury [S'14–M'20] received the B.Tech degree in Information Technology from Assam University in 2012, and the M.Tech degree in Information Technology from Gauhati University in 2014. He received his Ph.D. degree from Computer Science & Engineering department, IIIT Guwahati, India. Currently, he is with the department of Computer Science, Birla Institute of Technology & Sciences, Pilani, Hyderabad Campus as an Assistant Professor. Previously, he was a Project Fellow at IIT Guwahati. He is also a editorial board member of the IEEE Smart Cities Publications. His research interests include low-power wireless sensor networks and Internet of Things.

Rakesh Matam [M'14] received B.Tech. degree in computer science from Jawaharlal Nehru Technological University at Hyderabad, Hyderabad, India, in 2005, M.Tech. degree from Kakatiya University, Warangal, India, in 2008, and the Ph.D. degree in computer science from the Indian Institute of Technology (IIT) Patna, Patna, India, in 2013. In 2014, he joined the Department of Computer Science and Engineering, IIIT Guwahati, Guwahati, as an Assistant Professor, where he is currently a member of the Design and Innovation Center. He is a Principal Investigator of two funded research projects sponsored by SERB and DST, Government of India. His research interests include wireless networks, IoT, network security, and fog-cloud computing.

Mithun Mukherjee (S'10-M'15-SM'20) received the Ph.D. degree in electrical engineering from the Indian Institute of Technology Patna, Patna, India, in 2015. Currently, he is a Professor with the School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, China. Dr. Mukherjee was a recipient of the 2016 EAI WICON, the 2017 IEEE SigTelCom, the 2018 IEEE Systems Journal, and the 2018 IEEE ANTS Best Paper Award. He has been a guest editor for IEEE Internet of Things Journal and IEEE Transactions on Industrial Informatics. His research interests include wireless networks, mobile edge computing, and intelligent edge computing.

Jaime Lloret [M'07–SM'10] received his M.Sc. in physics in 1997, his M.Sc. in electronic engineering in 2003, and his Ph.D. in telecommunication engineering in 2006. He is an associate professor at Universitat Politecnica de Valencia, chair of the Research Institute IGIC, and the Head of the Active and Collaborative Techniques and Use of Technologic Resources in the Education (EITACURTE) Innovation Group. He has authored more than 450 research papers and book chapters. He was the Vice-Chair for the Europe/Africa Region of Cognitive Networks Technical Committee of the IEEE Communications Society for the term 2010–2012 and the Vice-Chair of the Internet Technical Committee of the IEEE Communications Society and the Internet Society for the term 2011–2013. He is also the Chair of the Working Group of the Standard IEEE 1907.1. He has been the Internet Technical Committee Chair of the IEEE Communications Society and the Internet Society for the term 2013–2015. Since 2016, he has been the Spanish Researcher with highest h-index in the Telecommunications journal list according to the Clarivate Analytics Ranking. He is Editor-in-Chief of Ad Hoc and Sensor Wireless Networks and Network Protocols and Algorithms. He is an ACM Senior and IARIA Fellow.