LBS: A Beacon Synchronization Scheme with Higher Schedulability for IEEE 802.15.4 Cluster-tree based IoT Applications

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

***Abstract*—The IEEE 802.15.4 standard is one of the most widely used link layer technology for building Internet of Things. It specifies several physical layer options and MAC layer for meeting low-power and low-rate requirements of devices deployed in a network of IoT. The standard also specifies a synchronization scheme for devices connected in a star topology, operating in beacon-enabled (BE) mode using periodic beacons. The BE mode facilitates synchronization among devices for data transmission and is suitable for large networks to establish low duty-cycles. Absence of a such a scheme for a cluster-tree network has confined its application only to non-beacon mode. The challenge here is to schedule beacon frame transmissions of multiple devices in a non-overlapping manner to avoid beacon collisions. This paper tackles the problem of synchronization by proposing localized beacon synchronization (LBS) scheme, a distributed technique for beacon scheduling in cluster-tree network topologies. LBS uses 2-hop information and association order to compute beacon transmission offsets that better utilize the available time slots, incur fewer transmissions, and is highly scalable. Further, we analytically show that the schedulability of the proposed scheme is higher compared to other related schemes. In addition, we also address the important issue of re-synchronization that has been ignored in all of the prior works. The proposed re-synchronization mechanisms consider the inter-dependencies between synchronization and duty-cycling schemes and are shown to significantly lower the synchronization overhead when synchronization among devices is lost.**

***Index Terms*—IEEE 802.15.4, P2P, cluster-tree, energy conservation, beacon synchronization.**

I. Introduction

The rapid pace at which the Internet of Things (IoT) technology is being adopted into our daily lives through applications like smart homes, smart buildings, smart cities, smart industries, smart health etc., it is key that the enabling communication and network technologies keep up with the varying application requirements. For example, a smart industry that is comprised of hundreds of machines, devices, sensors, and people to connect and communicate with each other via the Internet of Things, requires a reliable communication technology that meets the requirements of low-power, low data-rate and energy efficiency. The IEEE 802.15.4 [1] standard is one such key link-layer technology that has been designed to meet the aforesaid requirements. Based on the this standard, various IoT applications [2]–[7] have been realized. The standard supports two topologies: star and cluster-tree. A typical deployment of IoT devices can benefit by adopting the low power communication mechanisms of IEEE 802.15.4 [8]. The standard specifies mechanisms for devices to synchronize their transmissions with the help of beacons. But, a limitation is that the above schemes are devised for devices operating in a star-topology. On the other hand, applications like industrial IoT that would benefit from a cluster-tree topology [1] need a synchronization scheme to schedule beacon frame transmissions of multiple devices in a non-overlapping manner. In addition, a synchronization scheme is also necessary to efficiently operate other MAC sub-layer energy saving avenues like duty-cycling [9] [10]–[12].

The IEEE 802.15.4 networks are comprised of fully functional devices (FFD) and reduced functional devices (RFD), forming a low-rate wireless personal area network (LR-WPAN). The FFDs can serve either as a PAN coordinator (PANC), coordinator, or a cluster-head (CH). Coordinators may allow other devices to associate with it. An RFD is resource constrained and associates with a single FFD. It acts as an end device in the network topology. Periodic beacons are transmitted by coordinators that allow the associated devices to schedule their data transmission and sleep periods in a network operating in beacon-enabled mode [1]. Synchronization between these IEEE 802.15.4 devices is achieved with the help of a superframe structure [1]. The structure of a superframe is described within beacon frames as shown in Fig. 1. They are used for achieving synchronization and scheduling sleep cycles.

A. Superframe Structure

In beacon-enabled mode, a superframe structure is used for achieving synchronization among devices. The structure of a superframe is shown in Fig. 1. The time interval between two consecutive beacons is the beacon interval (BI) and consists of an active period and an optional inactive period (sleep period). In the active period (contention access period and contention free period), which is divided into 16 equal duration slots, devices transmit data frames, keeping its transceiver powered on. Slotted CSMA/CA is used for medium access in the contention access period (CAP), whereas, dedicated

N. Choudhury and R. Matam are with Department of Computer Science & Engineering, Indian Institute of Information Technology Guwahati, Guwahati, India (e-mail: nikumani.choudhury.2014@ieee.org, rakesh@iiitg.ac.in).

M. Mukherjee is with the Guangdong Provincial Key Laboratory of Petrochemical Equipment Fault Diagnosis, Guangdong University of Petrochemical Technology, Maoming 525000, China (e-mail: m.mukherjee@ieee.org).

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

access is possible in contention free period (CFP) through GTSs. Superframe duration (SD) is the active period starting from the beacon frame transmission. Optionally, devices may sleep in the active period until the beginning of the next superframe structure, forming a superframe cycle. Two parameters namely macBeaconOrder (BO) and macSuperframeOrder (SO) together define the structure of superframe as,

$$BI = \text{aBaseSuperframeDuration}.2^{\text{BO}} \tag{1}$$

$$SD = \text{aBaseSuperframeDuration}.2^{\text{SO}} \tag{2}$$

where, SO and BO refers to the duration of active period along with beacon transmission time and the cyclic time period when the coordinator communicates using beacons, respectively. aBaseSuperframeDuration is defined as the number of symbols constituting a superframe when the SO is set to zero. With $0 \leq \text{SO} \leq \text{BO} \leq 14$ and BO $= 15$ implies non-beacon mode.

B. Motivation

Cluster-tree is the preferred network topology for IoT applications like industrial automation and control, environment monitoring, smart agriculture etc. that usually have a wide network area and would benefit from multi-hop set-up like cluster-tree [13]. However, in such a topology, the presence of multiple coordinators necessitates a mechanism for maintaining synchronization between the transmissions of different coordinators. Such a mechanism prevents overlapping of beacon transmission offsets, and in the absence of it may result in collisions and thereby orphaned nodes. Synchronization among devices is straight forward in a star topology as all communications are through the PAN coordinator (PANC) only. Despite the challenge of implementing a synchronization scheme for a cluster-tree network, it is favoured over star topology as the former allows the network to scale. Industrial applications need such a topology whereby data needs to be transmitted across multiple devices, hop by hop, towards a base station or PANC.

Now, considering a synchronized cluster-tree network, multiple factors like changes in network topology or active duty-cycling schemes are shown to affect synchronization [10]. Duty-cycling impacts synchronization as it acts on same superframe parameters (BO and SO) that the synchronization scheme uses to build a transmission schedule. The loss of synchronization results in attempts to re-synchronize the network, typically using the same synchronizing scheme that is employed during network set-up. This adds to energy consumption overhead as this process basically redoes the entire synchronization process from the beginning. This is not necessary as the changes in network topology or loss of synchronization due to duty-cycling can be handled with minimal re-synchronization efforts. A lightweight mechanism that incurs minimal overhead is needed for re-synchronizing the devices' transmission periods. To the best of our knowledge, no previous work has been carried out in-depth analysis to address the issue of synchronization considering the factors affecting it, along with its inter-dependencies with a duty-cycling scheme.

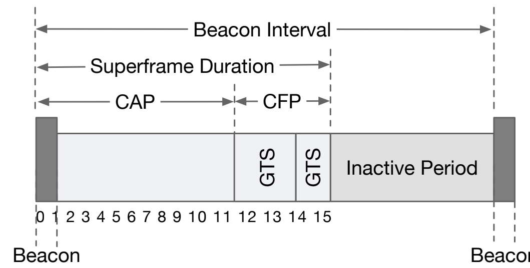


Fig. 1. IEEE 802.15.4 superframe structure [1].

C. Contributions and Organization

In this paper, we present a distributed beacon synchronization mechanism that emphasizes on minimizing the network transmission overhead and restricts beacon frame collisions between neighboring coordinators. The proposed localized beacon synchronization scheme utilizes the channel slots efficiently, resulting in higher schedulability. The higher schedulability of the proposed scheme in comparison to other related schemes is analytically proven. In this work, we also address the largely ignored issue of loss of synchronization, using effective re-synchronization mechanisms. Reasons for loss in synchronization and detailed analysis regarding the same is given in Section VI. Finally, the experimental analysis of proposed synchronization and re-synchronization mechanisms are shown depicting the impact on the network.

We summarize the main contributions of this paper as follows.

- A localized beacon synchronization mechanism for IEEE 802.15.4 cluster-tree networks is proposed.
- A schedulability analysis of the proposed scheme is presented that highlights the effectiveness of a synchronization scheme.
- We further present re-synchronization mechanism to reduce synchronization overhead that occurs due to changes in either network topology or superframe parameters.

The rest of the paper is organized as follows. Section III describes the network model considered. The proposed synchronization mechanism is described in Section IV. The schedulability analysis is presented in Section V. In Section VI, re-synchronization mechanisms are proposed followed by the overhead analysis of re-synchronization mechanism in Sub-Section VI-D. The simulation results are presented in Section VII. Finally, the conclusions are drawn in Section IX.

II. Related Research

The IEEE 802.15.4-2011 standard has been designed for LR-WPANs and WSNs that are typically comprised of resource-constrained devices. It specifies several physical layer options and MAC sub-layer for low-data-rate wireless connectivity among devices with limited power. It is one of the widely adopted standard for realizing IP based IoT applications that have flexible throughput requirements [1]. Several enhancements and improvements have been implemented after carrying out active research on IEEE 802.15.4 by addressing different shortcomings in the standard. A revised version

IEEE 802.15.4-2015, released in 2016, includes the 802.15.4e amendment that aims to support time-critical applications with strict quality-of-service requirements. In this regard, the standard facilitates deterministic communication using the Deterministic and Synchronous Multichannel Extension (DSME) and multi-channel frequency hopping using Time Slotted Channel Hopping (TSCH). Also, it supports low-latency applications using Low Latency Deterministic Network (LLDN) based on time division multiple access to meet timing guarantees. The performance of these mechanisms in comparison to IEEE 802.15.4-2011 is presented in [14]. The authors showcase that IEEE 802.15.4e provides improved throughput through channel hopping and lower latency with dedicated time-slots and multi-superframes. They also emphasize on its limitations in terms of lack of protocol implementations, security and the availability of supporting hardware. In addition, the limitations of TSCH mode that include lack of energy-efficient scheduling algorithms and the need for routing algorithms are highlighted. On the other hand, commercial implementation of IEEE 802.15.4-2011 like ZigBee, 6LoWPAN and WirelessHART have adopted 802.15.4 as the underlying standard. Further, industries like gas and petroleum refinery [15]–[17], agriculture [18], [19], smart city applications [20] and Industry 4.0 [21] continue to take advantage of the IEEE 802.15.4-2011 standard mainly due to the low-complexity in its implementation. Also, if the latency and QoS requirements of an IoT applications are not very strict, then the 802.15.4-2011 is more suited due to aforesaid reasons. In this paper, we pursue with the issue of synchronization in IEEE 802.15.4-2011 cluster-tree network topologies.

The problem of beacon synchronization in P2P network topologies has been tackled either using a centralized or a distributed mechanism. In centralized schemes, a central node called the PAN coordinator (PANC), is responsible for managing the beacon scheduling for all the other coordinators in the network topology. During network initialization, the beacon-transmitting coordinators transmit their information to the PANC. In other words, it is assumed that the PANC is aware of the entire network topology. The PANC computes the time-offset for beacon transmissions and broadcasts this to all the coordinators in a hop-by-hop manner. The centralized schemes provide an efficient schedule in static and small-(or even medium-) sized networks. The central node requires high computational capabilities and sufficient amount of energy. Further, such mechanisms need the support of a network layer for the multi-hop transmissions. It is observed that devices that are closer to the PANC, requiring constant data relay towards the central device, need to be more active compared to the other devices. Such devices suffer from constant energy dissipation. Contrarily, distributed mechanisms do not rely heavily on a single device. The benefits of a distributed scheme over a centralized scheme is the distribution of computation and transmission load from the centralized device to all the coordinators. All coordinators compute their own beacon transmission offset with the locally available information. Due to the self-organizing nature, distributed schemes are easily scalable and network size can be dynamic. Nonetheless, distributed mechanism suffers from extra overhead arising from the maintenance of tree routes, beacon information, or switching radio channels in case of a multi-channel approach.

In [22], the authors presented a centralized scheme known as superframe duration scheduling (SDS) that analyzes a set of superframes with different SD and BI, and provides a cyclic schedule if the set is schedulable. The PANC collects the BO and SO parameter values from all its associated coordinators. However, this is practically infeasible in large-scale real-time networks as frames travel in a multi-hop fashion towards the PANC, where a total delay and transmission overhead increases with each passed hop [23]. The authors propose the use of vertex coloring as means for coordinators that are far enough from their respective transmission ranges, to transmit their beacons simultaneously. Since, vertex coloring is a well known NP-complete problem, solving such a problem is complex. Also, prior knowledge of the entire network topology is necessary thus limiting its efficiency in networks that may undergo changes in their network structure. Another centralized scheme, SABTS [24], initially computes the transmission offset of the PANC and later the transmission offsets of the other coordinators are adjusted by a factor, which is a combination of the beacon length, symbol rate and the superframe duration of that node. In [25], in order to overcome network scalability and interference, same transmission slots are used for transmitting beacons and data frames at the same time using different channels. In [26], a similar collision-free multichannel superframe scheduling problem is considered. Most of these centralized schemes are generally known to suffer from low schedulability.

In distributed schemes like [27]–[31] coordinator devices compute their beacon transmission offset using local information from the neighboring coordinators. In [27], each device collects the beacon transmission offsets from its 2-hop neighbors and chooses an unoccupied empty time slot. The device broadcasts the selected slot to all its neighbors for any conflicts. The LABS [29] mechanism is similar to MeshMAC [27] in the initialization work, however, LABS can allocate a higher number of time slots on demand. In [30], the authors used a dedicated period called Beacon Only Period (BOP) that is used for beacon transmissions at the beginning of the superframe structure. The PAN coordinator defines the maximum number of slots in the BOP. Coordinators choose a contention-free slot from the BOP and transmit its beacons. The data transmission period follows after the BOP. Neighbors' table is used by the coordinator to identify the occupied slots of BOP and then chose an empty slot which is advertised as being temporarily reserved. In the STSS [31] scheme, the beacon broadcasting time slot is shifted between each coordinator in the cluster. An empty slot is computed based on the number of coordinators in the network, the sequence of beacon transmission offsets and the maximum possible number of beacon transmission slots. The authors have modified the beacon frame format to accommodate the requisite changes.

More recent works like [23], [32]–[34] describe approaches for scheduling in IEEE 802.15.4 networks. In [32], the authors propose a semi-dynamic and a dynamic scheduling approach. In semi-dynamic algorithm, coordinators are assigned beacon time slots only if their ID (address) is pre-stored by the PANC. However, end-devices are allocated beacon transmission slots by their respective parent coordinator based on their SD only if a schedulable schedule can be formed. To enhance the scalability and reduce latency, dynamic scheduling was also proposed where all coordinators and end-devices will be assigned their time slots dynamically without the need of any pre-defined sensor ID. The authors in [23] uses vertex coloring algorithm and breadth-first search to provide a non-conflicting transmission schedule in a cluster-tree topology. In [33], scheduling of the TDMA slots is followed by the contention period. The contention for TDMA slot is reduced by grouping coordinators into wake-up and sleep groups. This grouping is done by the PANC by broadcasting a scheduler table at the commencement of the superframe. The authors in [34], propose the scheduling of time-constrained data flows with opposite directions for IEEE 802.15.4 cluster-tree network topologies. The proposed heuristic algorithm is based on graph theory and combinatorial optimization problems.

Finally, with the introduction of IEEE 802.15.4e, several works [35]–[37] have been carried out to achieve scheduling in LLDN and TSCH mode. [35] proposes a priority-aware, scalable scheme called PriMulA that introduces a PriMulA message containing priority of the frames based on the deadline of message consumption. Higher schedulability is achieved through multi-channel communication. However, in this paper, we refrain ourselves from comparing with the works based on IEEE 802.15.4e.

In view of this, we address the beacon synchronization problem for cluster-tree topologies in IEEE 802.15.4 networks using a low overhead distributed mechanism. Also, we perform a schedulability analysis of the proposed scheme and compare with related schemes. Further, as a part of the solution, re-synchronization overhead has been addressed, which has been ignored in the prior works. The paper is distinct as it considers inter-dependencies during operation alongside other energy saving avenues like duty-cycling.

In [28], we presented a preliminary version of the proposed beacon synchronization algorithm. The differences of this work and [28] are as follows: 1) This paper addresses issues of re-synchronization arising from changes in network topology and alteration of superframe parameters. It proposed LBS re-synchronization mechanisms that incur low overhead. Such analysis was not present in the earlier version [28]. 2) Schedulability analysis of the proposed LBS is presented in this paper. 3) Finally, we present more detailed experimental results of the proposed beacon synchronization algorithm. This paper considers the aspects associated with beacon synchronization in a cluster-tree IEEE 802.15.4 network topology and presents energy-efficient measures to re-synchronize when transmission synchronization is lost among the devices.

III. Network Model

We consider an IEEE 802.15.4 cluster-tree network topology as the network model shown in Fig. 2. It comprises of coordinators and end devices. One of the selected coordinators acts as overall coordinator of the network called the PANC.

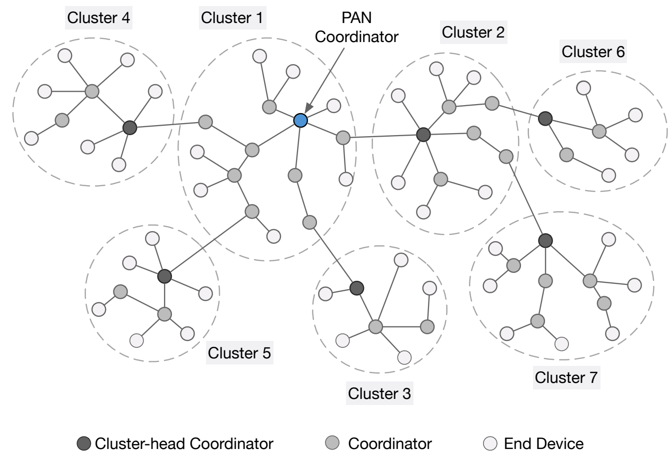


Fig. 2. Cluster-tree topology.

TABLE I
MAIN NOTATION DEFINITION

Symbols	Definition
n_{SDS}	The number of coordinators scheduled under SDS
n_{s}	The maximum number of devices that can be scheduled within a $2\,r$ range for MeshMAC
n'_{s}	The maximum number of devices that can be scheduled within a $2\,r$ range for LBS
N_{k}	The total number of coordinators scheduled by LBS
$\mathcal{O}_{\text{cen}}$	The synchronization overhead in the centralized schemes
$\mathcal{O}_{\text{dis}}$	The synchronization overhead in the distributed schemes
$N_{\text{sub-tree}}$	The number of nodes within the 2-hop neighborhood.
$nchild_{\text{dissociated}}$	The number of child nodes of the dissociated node
$nhigh_{\text{siblings}}$	The Number of siblings with higher AO

All coordinators are entrusted with the additional functionality of synchronizing associated nodes with the help of periodic beacons. The devices join the network by associating itself with a coordinator. The PAN coordinator forms the first cluster by choosing an unused PAN identifier and broadcasting beacon frames to neighboring devices. The coordinators transmit beacon frame at the beginning of their superframe cycle, which is followed by the active period. These beacons carry essential synchronization information like BO, SO parameters (stating the active and sleep period of the coordinator), without which the entire network will fail to have power saving functionality. End devices are devoid of routing capabilities and simply associated with a neighboring coordinator. They transmit all the sensed data to the associated parent coordinator. They are not permitted to transmit beacon frames. A group of coordinators and end devices form a cluster, that may execute a common function. A cluster head is chosen among the coordinators in each cluster for operational simplicity. The main notations in this paper are summarized in Table I.

We consider an association order (AO) for every coordinator. The AO field uses 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 AO is computed by the parent coordinator for all the associated child coordinator devices. A parent coordinator grants association to a device when current resources available on the PAN are sufficient to allow another device to associate. On successful association, the parent coordinator determines the AO based upon the relative association time of its child coordinator devices. This is stored against the short address of the associated coordinator in its neighbor list (defined in the standard). The neighbor list is also updated with the BO, SO parameters of the associated devices with respect to their short addresses. The association field determines the relative association order (AO) of the coordinators. The default AO of the grandparent is 0, parent coordinator is 1 and $\{2, 3 \text{ or } 4\}$ for the peers (siblings) sorted based on their association time. This facilitates coordinators in choosing a beacon transmission slot without any conflict with respect to its sibling coordinators.

IV. Proposed Distributed Beacon Synchronization Scheme

We aim to propose a localized beacon synchronization scheme where a coordinator computes its beacon transmission offset after obtaining the BO and SO values of their neighboring coordinators. The synchronization scheme is detailed as follows.

A. Localized Beacon Synchronization (LBS) Scheme

In the proposed beacon synchronization scheme, a coordinator that intends to compute its beacon transmission schedule, obtains the BO and SO of its parent through the parents' beacon. The BO and SO values of other neighbors that include parent of the parent coordinator (grandparent) and the sibling coordinators (other children of the parent coordinator) are retrieved from the neighbor list transmitted by the parent. This is done with the help of MAC layer management entity (MLME) association request and response messages of IEEE 802.15.4. The neighbor list consists of short addresses of associated coordinators, their respective BO and SO values, and their respective association order. Based on these values, a coordinator computes its own beacon transmission offset.

Algorithm 1 is executed by any coordinator to evaluate a transmission offset relative to the other coordinators that are already transmitting beacons, i.e., compute a synchronized beacon schedule. Algorithm 1 lists the steps involved in computing a non-overlapping beacon schedule. Initially, the coordinator retrieves the BO and SO information from the received beacon frame and the neighbor list. The respective BI and SD values are determined through equations 1 and 2 for all the 2-hop neighbors. The coordinator sorts all the BI into a set $\mathcal{B}$, based on the order of association and chooses the maximum BI as the $\mathrm{BI}_{\mathrm{max}}$ to fix the beacon schedule time cycle. This time schedule is further divided into smaller time slots, where each small time slot equals to the minimum SD. Next, the time slot for each coordinator (from the sorted $\mathcal{B}$ list), i.e., the superframe duration, given by SD_{i}, is allocated based on first empty time within the $\mathrm{BI}_{\mathrm{max}}$ slot. Depending upon BI_i, SD_i is allocated periodic slots until $\mathrm{BI}_{\mathrm{max}}$ is reached. This mechanism allows a coordinator to schedule its beacon transmission in a way that is synchronized with all the neighboring coordinators by recreating a map of their beacon transmissions.

Algorithm 1: Localized beacon synchronization algorithm

```
From parent beacon and neighbor list, obtain BO, SO,
  and association order for all the 2-hop neighbors.
Compute BI and SD for all received BO and SO,
  represented by set B, for all BI_i.
Compute BI_min = 2^(BO_min) and BI_max = 2^(BO_max).
Sort B based on association order for each BI.
Set time-line= BI_max, where slot = min(SD_i), 1 <= i <= N
for each i ∈ B do
    find the first available consecutive time slots >= SD_i
    fix (i) of SD_i in consecutive time slots beginning
      with first empty slot
    if (fix (i) of SD_i in consecutive time slots beginning
      with first empty slot= false) then
        return Not schedulable.
    end
end
return The coordinators transmission time slot.
```

Every coordinator that aims to compute such a schedule carries out this localized process, thereby synchronizing the entire network. The overall network synchronization time is minimal because of its localized and distributed nature, and just depends upon the time required to obtain the neighbor list from its parent coordinator. The time required to compute the transmission offset is dependent on the number of 2-hop neighbors with lower AO than itself. Once the transmission offset is determined, a coordinator can start its beacon transmission from the next superframe cycle. Also, the parent coordinator updates its neighbor list with this new information. Thus, the time required for computing and transmitting its first beacon is within two BIs of the parent coordinator.

The proposed LBS mechanism primarily relies upon the superframe parameters in order to schedule the transmissions of the coordinator devices. Based on these BO and SO values, transmission slot duration is allocated for the coordinators. Although the order of transmission schedules depends upon AO of the coordinator devices, any variation in the superframe parameter values can result in a different transmission schedule for the coordinators.

B. Illustrative Example of Proposed LBS

We consider an illustrative example to understand the working of the localized beacon synchronization mechanism. We assume a simple hierarchy of four coordinators with coordinator c2 associated with coordinator c1, and coordinators c3, c4 are associated with coordinator c2. Also, let the coordinators $\{\mathrm{c1, c2, c3}\}$ be already synchronized and transmitting beacons. At this instance, another coordinator c4 needs to compute an non-overlapping beacon transmission offset. It collects the requisite BO, SO and association order parameters from its parent (c2's) beacon payload. The configuration of the coordinators is shown in Table II. Based on received parameters, coordinator c4 computes the corresponding BI and SD for each

TABLE II
CONFIGURATION OF THE COORDINATORS

Coordinator	SD	BI	Association order
c1 (grand parent)	3	32	0
c2 (parent)	2	8	1
c3 (sibling)	1	8	2
c4	2	8	3

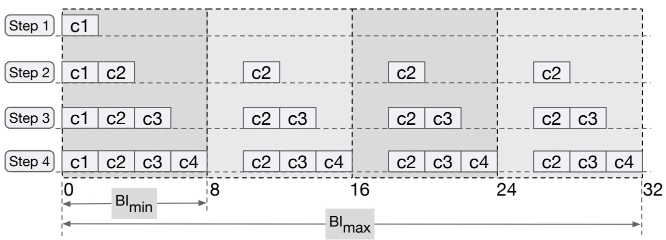


Fig. 3. Beacon transmission schedule for coordinators c1, c2, c3, and c4.

coordinator. The maximum BI is chosen as the $\mathrm{BI}_{\max} = 32$ and minimum BI, $\mathrm{BI}_{\min} = 8$. Then, it arranges the BI values with respect to their association order forming a ordered set $\mathcal{B}$ = $\{32(\mathrm{c1}), 8(\mathrm{c2}), 8(\mathrm{c3}), 8(\mathrm{c4})\}$.

Each instance of SD of the corresponding coordinator, from the set $\mathcal{B}$, is scheduled by allotting the first available slot of size SD time slots. This is accomplished without overlapping with other coordinators' SD. The subsequent instances of the coordinator are allotted at a distance equal to a multiple of its BI. In the considered example, c1 is placed at the beginning of the first horizontal line as it has the lowest AO. Then c2 is allotted after the instance of c1. Next, the instance of c3 is positioned in the third horizontal line after c2. Lastly, the instance of c3 is followed by c4. In accordance with the respective coordinators' BI, the instances are repeated. The transmission schedule hence formed is periodically repeated after a slotted time-line of 32 slots ($\mathrm{BI}_{\max}$). The final beacon transmission schedule computed by c4 is shown in Fig.3.

Now, if we change the configuration of c3 to {SD=3, BI=8}, then the set of superframe parameters is not schedulable. Although empty slots will be available within the $\mathrm{BI}_{\max}$, consecutive allocation of time slots after each BI_i is not possible. Again, if we change the configuration of c3 to {SD=5, BI=8}, then the set of the superframe is again not schedulable. Here, the necessary condition for a set to be schedulable is violated [22].

In the next subsection, we describe the schedulability of synchronization schemes and analytically discuss the schedulability of the LBS mechanism.

V. Schedulability Analysis of the Proposed Synchronization Scheme

Given a set of BO and SO for a group of coordinators, then a schedule is said to be schedulable if all the coordinators in the set can be scheduled in an non-overlapping fashion accommodating within the highest BI. In schedulability analysis, we determine the set of all superframes that are schedulable with a given scheduling algorithm. A schedulability comparison is drawn between centralized and distributed schemes.

A. Schedulability of Centralized and Distributed Synchronization Schemes

For any centralized scheme like SDS, each of the superframe slots are allocated in an exclusive way and there are no simultaneous communications. Therefore, a necessary condition for the superframe set to be schedulable is that the sum of all the duty cycles is lower than 1 [22], i.e.,

$$\sum_{i=1}^{n_{\mathrm{SDS}}} DC_i = \sum_{i=1}^{n_{\mathrm{SDS}}} \frac{\mathrm{SD}_i}{\mathrm{BI}_i} \leq 1, \tag{3}$$

where n_{SDS} is the number of coordinators scheduled using SDS algorithm. However, this is not a sufficient condition for a set to be schedulable. Such a set will have the empty slots available within the $\mathrm{BI}_{\max}$, however, consecutive allocation of time-slots for another coordinators' superframe after each BI_i is impossible.

On the other hand, the distributed scheme such as MeshMAC, simultaneously schedules the superframes of different coordinators that are at least 2-hop away. Thus, the beacon scheduling algorithm limits the scheduling to every node's 2-hop neighbors. This allows a higher number of coordinators in the network to schedule their transmissions without being constrained to overlapping time-slots of coordinators outside their collision domain. Hence, this increases the overall schedulability of the scheme in the network. Further, the necessary condition , i.e., (3) for a set to be schedulable still holds true within a 2-hop neighborhood with locally operating the MeshMAC scheme. We express the maximum number of devices that can be scheduled within a range of 2-hop as [27]

$$n_s = 2^{\mathrm{BO}-\mathrm{SO}} - 1\,. \tag{4}$$

We assume that the values of SD and BI are same for all the coordinators in SDS scheme. Therefore, we rewrite (3) as $\mathrm{BI} \geq n_{\mathrm{SDS}}\,\mathrm{SD}$.

For the distributed schemes, let i denote a 2-hop neighborhood relative to the coordinator. Then, in the ith 2-hop neighborhood, we have $\mathrm{BI}_i \geq n_{s_i}\,\mathrm{SD}_i$. In the same way, for the $(i+1)$th neighborhood, $\mathrm{BI}_{(i+1)} \geq n_{s_{(i+1)}}\,\mathrm{SD}_{i+1}$ holds, for the $(i+2)$th neighborhood, $\mathrm{BI}_{(i+2)} \geq n_{s_{(i+2)}}\,\mathrm{SD}_{(i+2)}$, and so on. Therefore, the total number of coordinators scheduled among all the 2-hop neighborhoods is expressed as

$$n_{s_i} + n_{s_{(i+1)}} + n_{s_{(i+2)}} + \ldots > n_{\mathrm{SDS}}\,. \tag{5}$$

However, MeshMAC assumes common BO and SO parameters throughout the network. In fact, due to the uniformity of the superframe parameters in the network, (4) will yield identical values in any 2-hop neighborhood. Thus, the number of coordinators scheduled at a distinct 2-hop neighborhoods will be same. For the sake of brevity, we consider three arbitrary 2-hop neighborhood expressed as $n_{s_i} = n_{s_{(i+1)}} = n_{s_{(i+2)}}$.

Therefore, the total number of coordinators scheduled in the entire network will be the sum of coordinators scheduled in the distinct 2-hop neighborhoods and expressed as $\sum_{i=1}^{k} n_{s_i} = k\,n_{s_i}$, where k denotes the number of distinct 2-hop neighborhoods and $k \geq 1$. Hence, using (5), we obtain

$$k\,n_{s_i} \geq n_{\mathrm{SDS}}\,. \tag{6}$$

The above shows that the MeshMAC is able to schedule higher number of coordinators than the centralized scheme, SDS. Another distributed scheme LABS [29],have similar schedulability to MeshMAC. In [29], the maximum number of devices that can be scheduled is exactly equal to that of MeshMAC. In [32], in the first semi-dynamic approach, only pre-chosen coordinators can schedule their beacons, i.e., a fixed number of coordinators can be scheduled. Thus, the schedulability will be lower than MeshMAC. However, this constraint is removed in the dynamic approach, allowing all heterogeneous superframes be granted transmission offset, if the set is schedulable. Therefore, in the pure dynamic approach, schedulability will be greater than $k\, n_{s_i}$, i.e., higher than MeshMAC. In [30], schedulability depends upon the number of slots in the BOP ($d_{\max}$), which is pre-defined by the PANC. Choosing a higher $d_{\max}$ value do not necessarily increase the schedulability as it decreases the slots available for data transmission.

B. Schedulability Comparision of Proposed LBS Scheme with MeshMAC Scheme

Unlike MeshMAC, the proposed LBS mechanism is not constrained for setting common BO and SO parameter values among the coordinators in the network. Such settings allow the coordinators to set optimal BO and SO values, based on channel traffic, thereby reducing the idle listening. With unequal SD values, $n'_{s_i} = n'_{s_{(i+1)}} = n'_{s_{(i+2)}}$ may not hold true. Here, n'_{s_i} is the number of coordinators scheduled in the ith 2-hop neighborhood. LBS allows more coordinators to be scheduled than MeshMAC by not putting restrictions on setting common superframe values throughput the network. Therefore, the total number of coordinators scheduled by LBS throughout the network can be expressed as

$$\sum_{i=1}^{k} n'_{s_i} = k\, n'_{s_i} = N_k\,, \tag{7}$$

where N_k is the total number of coordinators scheduled by LBS scheme. For LBS scheme, to obtain higher schedulability than MeshMAC, it is sufficient to prove that $n'_{s_i} \geq n_{s_i}$.

Theorem 1. *Let n_s and n'_s denote the number of coordinators scheduled in a 2-hop neighborhood for MeshMAC and LBS, respectively, then $n'_s \geq n_s$.*

Proof. Within a 2-hop neighborhood, the number of coordinators scheduled in MeshMAC is n_{s_i}. Further, $n_{s_i} = n_{s_{(i+1)}} = n_{s_{(i+2)}} = n_s$ due to common values for SO and BO parameters throughout the network. That is, in every 2-hop coverage, number of coordinators scheduled is equal. Therefore, maximum number of coordinators in a 2-hop neighborhood is $n_s = \mathrm{BI}/\mathrm{SD}$. For full channel utilization, the sum of duty cycles is 1, i.e.,

$$n_s\,\frac{\mathrm{SD}}{\mathrm{BI}} = 1\,. \tag{8}$$

In the proposed LBS mechanism, the maximum number of coordinators scheduled in the ith 2-hop neighborhood for full channel utilization can be expressed as

$$\begin{aligned}&\sum_{i=1}^{n'_s} \frac{\mathrm{SD}'_i}{\mathrm{BI}'_i}\\ &= \frac{\mathrm{SD}'_i}{\mathrm{BI}'_i} + \frac{\mathrm{SD}'_{(i+1)}}{\mathrm{BI}'_{(i+1)}} + \ldots + \frac{\mathrm{SD}'_{(i+n'_s)}}{\mathrm{BI}'_{(i+n'_s)}}\\ &= \frac{\mathrm{SD}'_i + \mathrm{SD}'_{(i+1)} + \ldots + \mathrm{SD}'_{(i+n'_s)}}{\mathrm{BI}} = 1\,.\end{aligned}$$

For simplicity, we consider that length of the superframe cycles of coordinators does not differ drastically, thus,

$$\mathrm{BI}'_i = \mathrm{BI}'_{(i+1)} = ... = \mathrm{BI}'_{(i+n'_s)} = \mathrm{BI}\,,$$

and using (8), we get

$$\frac{\mathrm{SD}'_i + \mathrm{SD}'_{(i+1)} + \ldots + \mathrm{SD}'_{(i+n'_s)}}{\mathrm{BI}} = n_s\,\frac{\mathrm{SD}}{\mathrm{BI}}\,.$$

Since $\mathrm{SD} \geq \mathrm{SD}'_i$, we subsequently get $n'_s\,\frac{\mathrm{SD}}{\mathrm{BI}} \geq n_s\,\frac{\mathrm{SD}}{\mathrm{BI}}$ which leads to $n'_s \geq n_s$. ■

Therefore, from (6), (7) and *Theorem 1*, we conclude

$$N_k \geq k\, n_{s_i} \geq n_{\mathrm{SDS}}. \tag{9}$$

Thus, the schedulability of LBS is higher compared to SDS and MeshMAC schemes. It is important to note that the above analysis do not pose any limitations on either the size of the network or the superframe parameter values for the coordinators.

C. Upper Bound of Schedulability

To estimate the upper bound for total number superframes that can be accommodated within the highest BI, maxBI (not to be confused with $\mathrm{BI}_{\max}$, that was used for computing transmission schedule), in a non-overlapping fashion, we consider the maximum value for BO and minimum SO value for the coordinators. Such a superframe parameter setting will allow the coordinators to operate at the longest BI (i.e, BO $= 14$). On the other hand, SO $= 0$ causes the coordinators to occupy minimum slot duration within the BI.

An illustrative example: Considering $2.4\,\mathrm{GHz}$ frequency band, the symbol duration is $16\,\mu\mathrm{s}$. Then, the minimum superframe duration becomes [1],

$$\begin{aligned}\mathrm{minSD} &= aBaseSlotDuration\\ &\quad \times aNumSuperframeSlots \times 16\,\mu\mathrm{s}\\ &= 60 \times 16 \times 16\,\mu\mathrm{s} = 15.36\,\mathrm{ms}.\end{aligned}$$

Now, for the maximum BI, BO $= 14$, we get

$$\mathrm{maxBI} = \mathrm{minSD} \times 2^{\mathrm{BO}} = 15.36\,\mathrm{ms} \times 2^{14} = 251.66\,\mathrm{s}\,,$$

where $aBaseSuperframeDuration = 15.36\,\mathrm{ms}$. Now, within the maxBI, the total number of superframes that can be accommodated is

$$\frac{\mathrm{maxBI}}{\mathrm{minSD}} = 16\,384\,.$$

Insights: The upper bound of schedulability allows us to infer that a huge number of superframes can be scheduled together without any overlap. However, in practice, this number will be quite low as compared to the upper bound as the active period for data transmissions will be longer. Moreover, coordinators may have shorter sleep periods, resulting in lower BO value. In either case, enough coordinators can be scheduled within a 2-hop neighborhood, eliminating the unlikely scenario of a coordinator not able to compute a non-overlapping beacon offset. However, if such a situation arises, the coordinator seeking a transmission offset has to be denied. The coordinator either has to wait until an existing coordinator leaves the network or some coordinator alters their respective superframe parameters, making its superframe parameters schedulable within the existing schedule. The coordinator repeatedly invokes the scheduling algorithm when either it receives beacons that contain information about the updated superframe parameters, or it stops receiving beacons of a dissociated neighboring coordinator.

VI. Proposed Re-synchronization Scheme for LBS

Synchronization among coordinators in a cluster-tree network can either be lost or the devices may simply need to recompute their synchronized schedules due to changes in the network topology. Also, concurrent operation of other energy saving schemes like duty-cycling that alter superframe parameters (BO, SO) may also result in loss of synchronization. In such a scenario, the synchronization scheme adopted by the network has to be repeatedly invoked resulting in higher synchronization overhead [10]. The authors in [22] discusses possible re-synchronization using both centralized (where all coordinators are affected) as well as distributed manner (all child coordinators under a single parent coordinator are affected). Since no experimental implementation or algorithm was proposed, we do not further consider it in our discussion in this paper. Other works in this direction, only focus on solving the synchronization problem and largely ignore the issue of loss of synchronization. Therefore, it can be assumed that in the event of loss of synchronization, re-synchronization is achieved by employing the existing synchronization scheme. The rationale behind ignoring the issue of re-synchronization can be attributed to the fact that all of the existing schemes have solely focused on synchronization without considering the inter-dependencies with other MAC processes. Herein, we initially determine the factors that may cause loss of synchronization, and later propose mechanisms to re-synchronize the network with minimal overhead. Also, the inter-dependencies between different MAC schemes like synchronization and duty-cycling are thereby addressed through the proposed LBS re-synchronization schemes.

A. Factors affecting synchronization

The foremost reason for the loss of synchronization among coordinator devices is due to changes in the network topology. The network topology may change due to a variety of factors that include new devices taking up the role of a coordinator, or due to malfunctioning of existing coordinators. Every

Algorithm 2: LBS re-synchronization due to topology change

Check the association order (AO) of the dissociated coordinator (coordinator i computes the algorithm).
if $(AO_{dissociated} > AO_i)$ **then**
    continue with existing schedule ;
end
else
    for *each* $SD_{dissociated}$ *to* SD_i **do**
        shift left transmission offset of coordinator i by ($\mathrm{SD}_{\mathrm{dissociated}}$);
        AO $--$;
    end
end

time such a change occurs re-synchronization is necessitated. The other important and interesting reason for the loss of synchronization is due to the dependencies with other energy conserving mechanisms like duty-cycling that may be operating concurrently. The necessity of duty-cycling scheme in the presence of operating synchronization scheme is discussed in [10]. The re-synchronization schemes are discussed as follows.

B. Re-synchronization due to changes in network topology

As listed above, when a device newly takes up the role of a coordinator it aims to compute a synchronized transmission schedule after obtaining the necessary beacon information from its parent coordinator. The newest coordinator has the highest association order and schedules its transmission (if schedulable) only after the transmission of other coordinator nodes. The synchronization of the other coordinators is not affected within the 2-hop neighborhood even with the arrival of a new coordinator.

Alternatively, when a coordinator parts with the network (that includes failures) re-synchronization is necessitated. Depending on the position of the leaving coordinator in the *parent-child* relation of the cluster-tree hierarchy, the employed mechanism varies. Firstly, if the dissociated coordinator is a grand-parent (relative to a coordinator), then all the 2-hop neighbors connected under its sub-tree have to recompute their transmission schedule. Secondly, a parent coordinator dissociation results into its child coordinators adapting their transmission schedule. Thirdly, when a child (sibling) coordinator dissociates, all the other siblings (if any) with higher AO adapt their schedules. Finally, if the dissociated coordinator is a leaf coordinator with the highest depth, then this would have no effect on any other coordinators in the network. The existing schedule can be continued for transmissions. Non-coordinator devices leaving the network, obviously, has no effect on the transmission schedule. However, they may impact the network traffic levels within the network; discussed in the next subsection.

The proposed re-synchronization algorithm to handle the loss of synchronization due to changes in topology is presented as Algorithm: 2. Neighboring coordinators consider a coordinator to be dissociated when they stop receiving the periodic beacons over a specific duration of time. A specified time-limit is considered because a temporarily disconnected coordinator may re-align with their previously connected parent coordinator. In such a scenario, all the associated attributes especially the AO is retained. However, the parent may give away the AO of the dissociated child after reserving it for a fixed number of BI. Firstly, a coordinator checks the AO ($\mathrm{AO}_{\mathrm{dissociated}}$) of the dissociated coordinator, hereby referred to as the affected coordinator. If its AO is lower than that of the affected coordinator, it does not alter its transmission schedule. Contrarily, if the coordinator has a higher AO than the affected coordinator, then it has to adjust its transmission schedule. It can do so by simply utilizing the empty (free) transmission slots made available by the affected coordinator. All the coordinators with higher AO in the neighborhood, similarly adjust their transmission slots by $\mathrm{SD}_{\mathrm{dissociated}}$ slots, in increasing order of their AO. Finally, we update AO of the coordinators.

Algorithm 3: LBS re-synchronization due to changes in superframe parameter values

Check the association order (AO) of the SO adapting coordinator (coordinator i computes the algorithm).
if $(AO_{affected} > AO_i)$ **then**
    continue with existing schedule ;
end
else
    for *each* $SD_{affected}$ *to* SD_i **do**
        if $(new_{SD_{affected}}) < (old_{SD_{affected}})$ **then**
            Shift left transmission offset of the ith coordinator by $(\mathrm{old}_{\mathrm{SD}_{\mathrm{affected}}} - \mathrm{new}_{\mathrm{SD}_{\mathrm{affected}}})$;
        end
        else
            shift right transmission offset of the ith coordinator by $(\mathrm{new}_{\mathrm{SD}_{\mathrm{affected}}} - \mathrm{old}_{\mathrm{SD}_{\mathrm{affected}}})$;
        end
    end
end

C. Re-synchronization due to superframe adaptation

The proposed LBS synchronization scheme uses BO and SO values of a set of coordinators to compute their respective transmission offset. These parameters define the transmission duration and the cycle for the next transmission. Alternatively, duty-cycling aims to optimize energy consumption, also by adjusting the BO, SO parameter values of the coordinators. Coordinators may periodically carry out duty-cycling by adjusting their active and sleep-periods to accommodate changes in the network and varying traffic flows.

Change in the BO parameter determines the frequency of beacon transmissions and the duration of the sleep period, which do not disturb the transmission schedule of other coordinators. However, an increase in the frequency beacon transmissions (lowering of BO) can result in overlapping of transmission period (SD) with other coordinators. This scenario is addressed similar to an increase in SD value, as discussed below. Contrarily, when a coordinator changes its SO parameter, the transmission schedule may be disrupted due to overlapping transmissions. Such transmissions lead to collisions between frames, resulting in loss of synchronization and orphaned devices. Also, channel utilization will decrease when empty slots are present due to the shrinking of the SO parameter. To address this, we present Algorithm 3, that considerably lowers the re-synchronization overhead.

It is worth noting that when a coordinator adapts its SO parameter, this change is communicated to the neighbors through its beacon frames [1]. Initially, a coordinator checks for the AO ($\mathrm{AO}_{\mathrm{dissociated}}$) of the coordinator that adapted the SO parameter. If its own AO is lower than of the affected coordinator, it follows the existing transmission schedule. Otherwise, if the affected coordinator has reduced its SO parameter, all the coordinators with higher AO in the neighborhood left shifts their transmission slots. The number of slots to be shifted is equal to the number of SD slots reduced by the affected coordinator. On the other hand, when the affected coordinator inflates its SD, the other coordinators right shifts their transmission slots. All these transmission shifts occur in an increasing order of their AO. The new SD parameter value of the $\mathrm{AO}_{\mathrm{dissociated}}$ is hereby referred to as $\mathrm{new}_{\mathrm{SD}_{\mathrm{affected}}}$ and the old SD value by $\mathrm{old}_{\mathrm{SD}_{\mathrm{affected}}}$.

D. Overhead due to re-synchronization

Re-synchronization algorithm is triggered when a coordinator (grand-parent, parent or sibling) is dissociated from the network or adapts its superframe parameter.

Centralized synchronization scheme: For a centralized synchronization scheme, the overhead associated with re-synchronization is expressed as [10]

$$\mathcal{O}_{\text{cen}} = r\, d \times \sum_{i=0}^{h} \frac{h-i}{p^i}\,, \tag{10}$$

where p is the maximum number of child leaf-coordinators of any intermediate parent-coordinator and d is the number of leaf-coordinators with a height of h. This overhead is the maximum control message overhead in the transmission for a cluster-tree with all sub-tree at the maximum depth.

Distributed synchronization scheme: The transmission overhead for a distributed scheme for re-synchronization is given by [10]

$$\mathcal{O}_{\text{dis}} = 2\, r\, n\,, \tag{11}$$

where n is the number of coordinators that transmit beacons and r denotes the transmission data-size (in byte) during the synchronization process.

Proposed re-synchronization scheme: In our proposed LBS re-synchronization scheme, the number of slots that is to be shifted is known to the child coordinator as they are synchronized with their parent and grandparent coordinators. Further, the number of slots to be shifted remain constant. For the first case (i.e., the grand-parent), the overhead is equal to the number of coordinators in its 2-hop neighborhood. In the second case (i.e., the parent), the associated overhead is equal to the number of child coordinators of the affected coordinator.

TABLE III
RE-SYNCHRONIZATION OVERHEAD (LBS)

Affected Coordinator	Overhead
Grand parent	$N_{sub-tree}$
Parent	$nchild_{\text{dissociated}}$
Sibling	$nhigh_{\text{siblings}}$

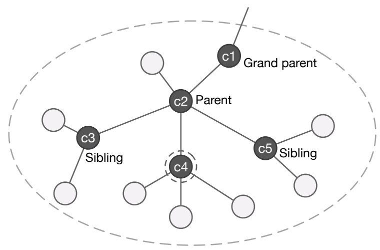


Fig. 4. An illustrative example. The coordinator c1 is the grand-parent, c2 is the parent coordinator, and both c3 and c5 are sibling coordinators with respect to the coordinator c4.

This is because the transmission schedule is formed in the sequence of AO, and the only the coordinators with higher AO (relative to the affected coordinator) shifts their transmission offsets. In the third case (i.e., the sibling), the overhead is equal to the siblings with higher AO. We summarize the overhead in Table III. The overhead is listed in terms of a number of coordinator shifting their transmission slots.

An illustrative example: Consider a cluster-tree topology as shown in Fig. 4 with the association order as shown in Table IV. We compare and summarize the re-synchronization overhead of traditional distributed scheme like MeshMAC with our proposed LBS re-synchronization mechanism in Table V. Typically, the overhead associated with distributed schemes is re-computation of the entire beacon transmission schedule and expressed as $2\,rn$. However, in our proposed scheme LBS, the re-synchronization overhead is due to the only shifting of transmission slots by a few coordinators. Specifically, a coordinator needs to update its beacon transmission slot to keep synchronized. The number of coordinators that updates their transmission slots is the only associated overhead. In the considered topology, when c1 is dissociated from the topology or updates its SO parameter, the coordinators c2, c3, c4 and c5 shift their transmission slots accordingly. Similarly, if the coordinator c2 dissociates or updates its SO parameter, only the coordinators c3, c4 and c5 updates their respective slots. Again, for the coordinator c3, only the sibling coordinators c4 and c5 shift their slots accordingly. Further, for the coordinator c4, only the coordinator c5 (with higher AO than the coordinator c4) needs to shift its transmission offset. Finally, when coordinator c5 dissociates from the network or updates its superframe parameter, none of the other coordinators are being affected. As a result, we observe a significant reduction in re-synchronization overhead with our proposed scheme compared to the MeshMAC.

VII. SIMULATION RESULTS AND DISCUSSION

In this section, the performance of LBS [1] is evaluated and compared with different schemes like SDS [22], MeshMAC [27], LABS [29], TBoPs [30] and a dynamic scheme presented in [32]. The performance metrics considered are a) *transmission overhead:* that is in terms of number of transmissions required for synchronization among the coordinators, b) *energy consumption*, c) *MAC goodput*, d) *schedulability*, and e) *re-synchronization overhead*.

[1]The code are released at https://github.com/Nikumani/synchronization.

TABLE IV
CONFIGURATION OF C4

Coordinator	Association order
c1 (grand parent)	0
c2 (parent)	1
c3 (sibling)	2
c4	3
c5 (sibling)	4

TABLE V
COMPARISON OF RE-SYNCHRONIZATION OVERHEAD

Coordinator	MeshMAC overhead (transmission)	LBS overhead (coordinators)
c1	$8\,r$	4
c2	$8\,r$	3
c3	$8\,r$	2
c4	$8\,r$	1
c5	$8\,r$	0

TABLE VI
SIMULATION PARAMETERS

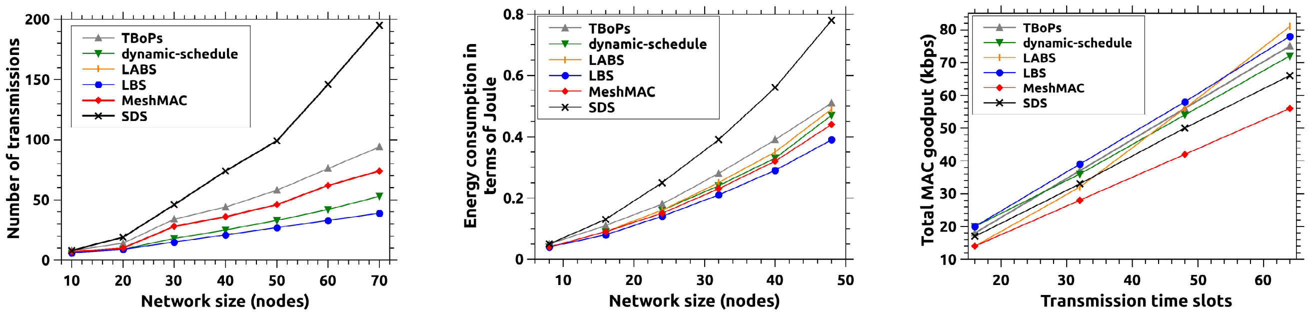
Parameters	**Values**
Frequency band	2.4 GHz
Maximum data rate	250 kbps
Number of nodes	48
Transmission radius	50 m
BO	8
SO	2
Initial Energy	1 J
Energy consumption to receive a frame	0.003 J
Energy consumption to transmit a frame	0.006 J
Energy consumption during the sleep-state	0.000 030 J

A. Simulation Setup

We consider an IEEE 802.15.4 cluster-tree network topology as shown in Fig. 2. The coverage area of this network set-up is $1000\,\mathrm{m}$. The network can be scaled by adding new or by removing existing coordinators. The devices associate with a coordinator and the coordinators forward data towards the PANC. The simulations are carried out on network simulator NS-2.34 [38]. We consider the $2.4\,\mathrm{GHz}$ frequency band that provides a maximum data rate of $250\,\mathrm{kbps}$ [1]. Besides, a transmission radius of $50\,\mathrm{m}$ is chosen to provide sufficient reliability to the transceiver with an optimum energy consumption [39]. We compute the energy consumption in all the schemes for a $3000\,\mathrm{sec}$ duration. We have used two-ray ground radio propagation model and IEEE 802.15.4 PHY and CSMA/CA for MAC sub-layer in our simulation. The other simulation parameters are summarized in Table VI.

B. Transmission Overhead

A coordinator transmits beacons to synchronize transmissions with its associated devices. Existence of multiple coordinators necessitates a synchronization scheme that allows a



(a) Comparison of transmission overhead.

(b) Comparison of energy consumption.

(c) Channel utilization of different schemes.

Fig. 5. Comparison of transmission overhead, energy consumption, and channel utilization.

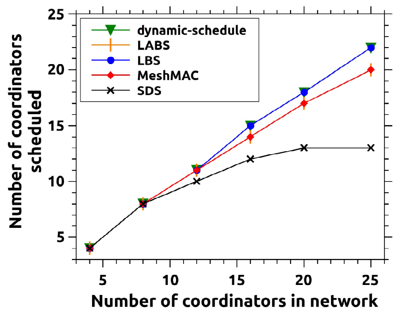
coordinator to compute a non-overlapping beacon transmission offset. The goal is to achieve this with minimal overhead as synchronization is not a one-time network process. Fig. 5(a) compares the number of transmissions required for synchronization in SDS, MeshMAC, LABS, TBoPS, dynamic approach and the proposed LBS scheme. SDS being the foremost scheme and a centralized scheme, has been included and just acts as a reference point. Overhead in MeshMAC is due to reception of messages containing the beacon transmission offset from all the coordinators within 2-hops and also transmission of computed offset to all these 2-hop coordinators. LABS operates similar to MeshMAC by acquiring neighbor list and beacon transmission time of all its 2-hop neighbors. TBoPs incurs significant transmission overhead as it periodically exchanges hello request and response frames among the 2-hop neighbors. Further, the selected slot is broadcasted to resolve slot conflicts. Also, in the dynamic scheme, devices transmit a beacon scheduling frame requesting the parent coordinator to allocate a transmission offset, which in turn broadcasts the allotted slot to the network. In contrary to all the above, LBS significantly reduces the transmission count by relying on its parent coordinator for the neighbor list, thereby restricting the transmission count to 2 for each coordinator. The localized nature of computation does not necessitate conflict resolutions. From the simulation results, we observe that LBS performs better compared to other schemes in term of transmission count necessary for synchronization.

C. Energy Consumption

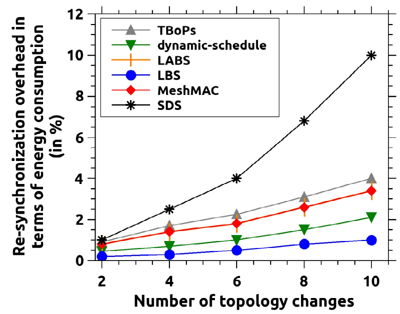
As one of the primary design goals of the IEEE 802.15.4 MAC protocols is to enhance the network lifetime, beacon synchronization schemes lead considerable energy savings by facilitating collision-free transmissions. However, the synchronization scheme in itself should not incur high overhead in the network [10]. In Fig. 5(b), we show the energy consumed by different schemes. As transmission count is directly proportional to the energy consumed, we observe that the proposed LBS mechanism consumes the least energy to determine the beacon schedule. It provides 6%, 13% and 38% energy savings compared to MeshMAC, TBoPs and SDS, respectively. This is because LBS incurs fewer transmissions by depending only on its parent coordinator for the neighbor list. On the contrary, for MeshMAC, the transmission overhead depends on the degree of the coordinator, resulting in higher energy consumption. The synchronization overhead of LABS is comparatively similar to that of MeshMAC as it follows a similar mechanism for computing the transmission offsets. In TBoPS, neighboring coordinators constantly exchange hello request and hello response messages for their updated beacon schedule, SD and BI information. Based on this information, a coordinator randomly chooses an empty slot (if available) and advertises to all its neighbors. These transmissions results in higher energy dissipation compared to schemes like LBS, MeshMAC and LABS. Whereas, the dynamic approach, relying on receiving beacon scheduling frames and broadcasting the allocated slot, initially consumes less energy for transmissions than MeshMAC. However, slot allocation results in parent coordinator rescheduling its associated devices based on the traffic pattern. This leads to an increase in the overall energy consumption.

D. MAC Goodput

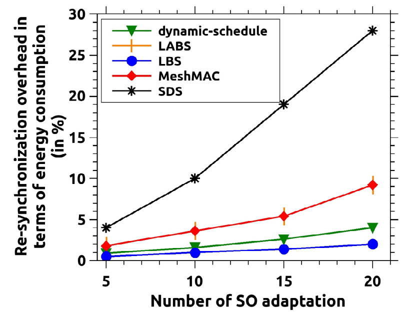
Effective channel utilization is one of the performance metrics of a beacon synchronization scheme as it gives a measure of successfully transmitted bits. Fig. 5(c) depicts the total MAC goodput for all synchronization schemes. We observe that the SDS results in low channel utilization as the PANC assigns effectively non-overlapping schedules even though the coordinators in contention are not in collision range of each other. On the other hand, MeshMAC and LABS also report sub-optimal schedule due to the assumption of common BO and SO parameter values over the entire network. However, LBS, ToBoPs and the dynamic schemes do not impose any restrictions on BO and SO values to addresses the synchronization transmission conflicts. Unlike MeshMAC, LABS can schedule extra slots (if available) for improving throughput. TBoPs reserves a fixed number of slots for BoP, that reduces its channel utilization. As a result, it is observed that even in the dynamic network settings with a frequent change of BO and SO parameters, the proposed LBS provides 9%, 15% and 28% increase in channel utilization compared to TBoPs, SDS and MeshMAC schemes, respectively. The difference in channel utilization among LBS, LABS and the dynamic approach is observed to be minimal.



(a) Schedulability of the schemes.



(b) Re-synchronization overhead during change in network topology.



(c) Re-synchronization overhead during superframe parameter adaptation.

Fig. 6. Schedulability performance and re-synchronization overhead.

E. Schedulability Performance

The choice of a synchronization scheme also depends on its schedulability. A higher schedulability also reflects better channel utilization. Fig. 6(a) shows the schedulability of various schemes. Distributed schemes compute the transmission schedule such that the coordinators that are not in collision range may transmit at an overlapping time. This allows a higher number of coordinators to schedule their respective transmissions within the given time period. Slight exceptions include MeshMAC and LABS that have a slot reserved for broadcast transmission. In addition, these schemes also need usage of uniform BO and SO values in the network. This results in wastage of slots as uniform BO and SO result in same BI and SD irrespective of requirement. On the contrary, the proposed LBS is does not restrict, leading to higher schedulability compared to these three synchronization schemes. Upon purely using only the dynamic approach in [32], schedulability is higher than SDS, MeshMAC and LABS. We have restrained from comparing the schedulability of TBoPs as its BoP approach do not align with the other schemes. Further, its schedulability primarily depends upon the number of slots in the BoP, chosen by the PANC.

F. Re-synchronization Overhead

Lastly, we compare the re-synchronization overhead. The synchronization between the coordinators may be lost when the network topology changes or when coordinators alter their superframe parameters. Different scenarios in which the above can happen and their effects are discussed in Section VI. As existing schemes [22], [27], [29], [30], [32] do not address this issue exclusively, we consider the basic synchronization scheme to handle the issue of loss of synchronization. Fig. 6(b) and Fig. 6(c) shows re-synchronization overhead of the various schemes during topology change and SO adaptation, respectively. Here, change in topology means association or dissociation of coordinator devices that disrupt the existing transmission schedule. In MeshMAC and LABS, with a transmission overhead of $2\,rn$ as observed in (11), the re-synchronization overhead is affected within its 2-hop neighbors. In the dynamic scheme, all the sibling coordinators will have to re-compute their schedule to avoid loss of synchronization. Re-synchronization overhead in TBoPs is also restricted within its 2-hop neighbors when a coordinator parts with the network (including failures). However, as it is based upon the BoP approach, it do not consider superframe parameters for beacon schedule computation. Our proposed LBS scheme minimizes the re-synchronization overhead through shifting of the transmission slots by the selective coordinators with respect to their AO. This is achieved by identifying the certain coordinators that require to adjust their transmission offsets while other coordinators can keep their existing schedule.

G. Discussion and Limitations of LBS

The proposed LBS mechanism differs from other distributed mechanisms like MeshMAC, LABS and TBoPs through their computation of an exclusive, non-conflicting beacon transmission offset for each coordinator. The AO guarantees that no two neighbors (within the considered 2-hops) will chose the same slot at any given time. Other schemes choose an empty slot and broadcasts the chosen slot among the neighbors for any possible conflicts. This primarily occurs as neighboring coordinators may simultaneously choose the same empty slot for beacon transmission. Further, LBS relies only on its parent coordinator for retrieving information about all its neighbors. MeshMAC and LABS depends upon all its neighbors, while, TBoPS periodically exchanges messages among all its neighbors. Moreover, LBS allows heterogeneity in the choice of BO and SO parameter values of the coordinators unlike MeshMAC and LABS.

LBS considers collisions only upto 2-hop neighbors, i.e., coordinators more than 2-hop away may transmit simultaneously. However, in scenarios where coordinators' transmission or sensing range is beyond 2-hop distance, LBS may still schedule transmissions at the same time, resulting into beacon and data frame collisions. Dense network topologies are prone to such a scenario. The collision probability of such a topology was studied in [28]. In LBS, some set of superframes may be declared as unschedulable for two reasons. Firstly, if sum of all the duty cycles is greater than one, i.e., no empty slots are available for transmissions. Secondly, even if empty slots are present, they may not be sufficiently long to accommodate, or not available at regular time intervals. Thus, some empty slots may still remain un-schedulable under the given set of superframes. Further, unlike LABS, extra slots cannot be

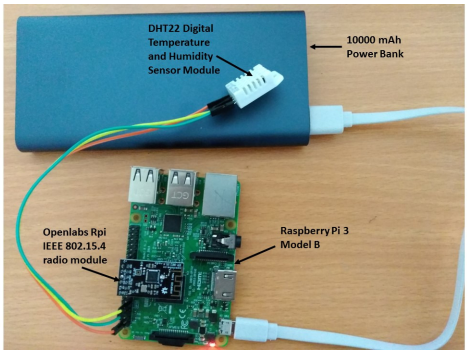


Fig. 7. Raspberry Pi with IEEE 802.15.4 radio for testbed setup.

provided for a particular coordinator incurring heavy traffic flows. However, it is possible for such a coordinator to increase its SD slots through an operational duty-cycling mechanism, provided that the new set is schedulable.

VIII. Implementation of LBS on Real Devices

A proof-of-concept LBS implementation was realized with a small testbed setup to verify the feasibility of LBS. For the setup, Raspberry Pi 3 (RPi) and Openlabs RPi 802.15.4 radio are used. Openlabs offers an antenna add-on module that turns a RPi into a 6LoWPAN border router solution. The module consists within in a chip antenna, crystal oscillator and an Atmel AT86RF233, that includes the transceiver to get an 802.15.4 solution. It provides IEEE 802.15.4-compliant physical layer that adopts the 2.4-GHz PHY with a data rate of 250 kb/s. The module can be plugged directly onto the pins 15-26 of the RPi's GPIO. We use 10000 mAh battery to power the RPi. We use DHT22 digital temperature and humidity sensor module to collect data and later transmit these data frames to the parent coordinator. Fig. 7 shows the Raspberry Pi set up with Openlabs RPi 802.15.4 radio. The implementation is realized using Linux WPAN-tools. A network topology comprising of one PANC, two coordinators and two end-devices was considered. Preliminary experiments were conducted to verify the working of the proposed mechanism.

IX. Conclusion

In this paper, we have addressed the issue of beacon synchronization in an IEEE 802.15.4 based cluster-tree network. Here, we proposed a localized beacon-synchronization scheme that facilitates a coordinator to compute a non-overlapping beacon transmission schedule based on its association order. It relies on BO and SO values of all the 2-hop neighboring coordinators to compute a beacon transmission offset. The proposed scheme is shown to better utilize the available time slots with fewer frame transmissions. Also, the schedulability analysis of the proposed scheme shows that LBS can schedule a higher number of coordinators compared to SDS or MeshMAC. In addition to the above, we also analyze the factors affecting synchronization and propose lightweight re-synchronization schemes to handle the issue of loss of synchronization. We have considered the transmission overhead as a parameter in re-synchronization of a network that loses synchronization. Most importantly, the proposed re-synchronization schemes identify the coordinators that need to shift their transmission-slots while the other coordinators use their existing transmission offset. Therefore, the proposed LBS re-synchronization mechanisms significantly reduce the transmission overhead due to changes in network topology and superframe parameter values of the coordinator devices, in comparison to other related synchronization schemes. We further aim to design a duty-cycling scheme that either has a minimum impact on the existing transmission schedule or quickly converges the network to a synchronized schedule. Alternatively, it is also necessary to develop a duty-cycling scheme considering the inter-dependencies with the synchronization scheme.

Acknowledgement

This work is supported by Science and Engineering Research Board, DST, Govt. of India Under ECR, 2016. Grant Number: 2016/001651.

References

[1] "IEEE Std 802.15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs)," http://www.ieee.org/Standards/, accessed 15 Jan. 2019.

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

[3] 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.

[4] W. Li, C. Zhu, V. C. Leung, L. T. Yang, and Y. Ma, "Performance comparison of cognitive radio sensor networks for Industrial IoT with different deployment patterns," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1456–1466, Sept. 2017.

[5] B. Ünlü, B. Özceylan, and B. Baykal, "DIVVY: An efficient shared cell scheduling method and algorithm for 6TiSCH based IoT networks," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 1, pp. 170–179, Mar. 2019.

[6] X. Zhai, X. Guan, C. Zhu, L. Shu, and J. Yuan, "Optimization algorithms for multi-access green communications in Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1739–1748, June 2018.

[7] R. Lacuesta, J. Lloret, M. Garcia, and L. Penalver, "A secure protocol for spontaneous wireless ad hoc networks creation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 629–641, Apr. 2013.

[8] M. Salayma, A. Al-Dubai, I. Romdhani, and Y. Nasser, "Reliability and energy efficiency enhancement for emergency-aware wireless body area networks (WBAN)," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 3, pp. 804–816, Mar. 2018.

[9] N. Choudhury, R. Matam, M. Mukherjee, and L. Shu, "Impact of synchronization scheme on duty cycling in IEEE 802.15. 4 cluster tree networks," in *Proc. IEEE IECON*, Nov. 2017, pp. 8471–8475.

[10] 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.

[11] A. Rajandekar and B. Sikdar, "A survey of MAC layer issues and protocols for machine-to-machine communications," *IEEE Internet Things J.*, vol. 2, no. 2, pp. 175–186, Apr. 2015.

[12] N. Choudhury, R. Matam, M. Mukherjee, and L. Shu, "Poster: Adaptive duty cycling in IEEE 802.15.4 cluster tree networks using MAC parameters," in *Proc. ACM MobiHoc*, July 2017, pp. 37–38.

[13] J. Lloret, M. Garcia, D. Bri, and J. R. Diaz, "A cluster-based architecture to structure the topology of parallel Wireless Sensor Networks," *Sensors*, vol. 9, no. 12, pp. 10 513–10 544, Dec. 2009.

[14] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "IEEE 802.15. 4e in a nutshell: Survey and performance evaluation," *IEEE Commun. Surveys Tut.*, vol. 20, no. 3, pp. 1989–2010, 3rd quarter 2018.

[15] F. Chraim, Y. B. Erol, and K. Pister, "Wireless gas leak detection and localization," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 768–779, Apr. 2016.

[16] M. Gaafar and G. G. Messier, “Petroleum refinery multiantenna propagation measurements,” *IEEE Antennas and Wireless Propagat. Lett.*, vol. 15, no. 1, pp. 1365–1368, Dec. 2016.

[17] A. Shukla and H. Karki, “Application of robotics in onshore oil and gas industry- A review Part I,” *Robot. and Autonomous Syst.*, vol. 75, no. Part B, pp. 490–507, Jan. 2016.

[18] M. Bacco, A. Berton, A. Gotta, and L. Caviglione, “IEEE 802.15. 4 air-ground UAV commun. in smart farming scenarios,” *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1910–1913, Sept. 2018.

[19] 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.

[20] A. S. Bharadwaj, R. Rego, and A. Chowdhury, “IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application,” in *Proc. IEEE INDICON*, Dec. 2016, pp. 1–6.

[21] M. Wollschlaeger, T. Sauter, and J. Jasperneite, “The future of Industrial communication: Automation networks in the era of the Internet of Things and industry 4.0,” *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.

[22] 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.

[23] E. Y. Golubnichaya and B. Y. Likhttsinder, “Slot data routing in cluster wireless sensor networks based on time synchronization of cluster heads,” in *Proc. SYNCHROINFO*, July 2018, pp. 1–5.

[24] M. U. H. Al Rasyid, F. A. Saputra, and M. R. Ismar, “Performance of multi-hop networks using beacon and non-beacon scheduling in wireless sensor network (WSN),” in *Proc. IEEE IES*, Sept. 2015, pp. 195–199.

[25] E. Toscano and L. L. Bello, “Multichannel superframe scheduling for IEEE 802.15.4 Industrial wireless sensor networks,” *IEEE Trans. Ind. Informat.*, vol. 8, no. 2, pp. 337–350, Sept. 2012.

[26] X. Jin, Q. Zhang, P. Zeng, F. Kong, and Y. Xiao, “Collision-free multichannel superframe scheduling for IEEE 802.15.4 cluster-tree networks,” *Inderscience Int. J. Sensor Netw.*, vol. 15, no. 4, pp. 246–258, Jan. 2014.

[27] 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. Springer AdHoc Netw.*, Sept. 2009, pp. 561–575.

[28] N. Choudhury, R. Matam, M. Mukherjee, and L. Shu, “Distributed beacon synchronization mechanism for 802.15.4 cluster-tree topology,” in *Int. Wireless Internet Conf., Springer*, Dec. 2016, pp. 10–20.

[29] M. Baseri, S. A. Motamedi, and M. Maadani, “A Load-Adaptive Beacon Scheduling algorithm for IEEE 802.15.4 mesh topology improving throughput and QoS in WMSNs,” in *Proc. IEEE ICCCNT*, July 2014, pp. 1–5.

[30] 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 DCOSS*, May 2012, pp. 341–346.

[31] C.-M. Wong, “A novel beacon frame scheduling algorithm based on cluster-tree IEEE 802.15.4 wireless sensor networks,” in *Proc. ICCS*, Nov. 2012, pp. 285–289.

[32] B. A. Alghamdi, M. Ayaida, and H. Fouchal, “Scheduling approaches in beacon-enabled mode for wireless sensor networks,” *Concurrency and Computation: Practice and Experience*, vol. 29, no. 23, p. e4284, Sept. 2017.

[33] 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, pp. 2273–2287, Apr. 2019.

[34] A. Ahmad and Z. Hanzálek, “An energy efficient schedule for IEEE 802.15. 4/zigbee cluster tree wsn with multiple collision domains and period crossing constraint,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 12–23, Jan. 2018.

[35] G. Patti and L. L. Bello, “A priority-aware multichannel adaptive framework for the IEEE 802.15. 4e-LLDN,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 10, pp. 6360–6370, Oct. 2016.

[36] M. Meng, Z. Yujun, Z. Dadong, and C. Fan, “Scheduling for data transmission in multi-hop IEEE 802.15. 4e TSCH networks,” *Mobile Networks and Applications*, vol. 23, no. 1, pp. 119–125, Feb. 2018.

[37] R.-H. Hwang, C.-C. Wang, and W.-B. Wang, “A distributed scheduling algorithm for IEEE 802.15. 4e wireless sensor networks,” *Computer Standards & Interfaces*, vol. 52, no. 1, pp. 63–70, May 2017.

[38] “NS2-the network simulator,” http://www.isi.edu/nsnam/ns.

[39] F. Xia, A. Vinel, R. Gao, L. Wang, and T. Qiu, “Evaluating IEEE 802.15. 4 for cyber-physical systems,” *EURASIP J. Wireless Commun. Netw.*, vol. 2011, no. 1, pp. 1–14, Dec. 2011.

Nikumani Choudhury [S’14] 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 is currently pursuing Ph.D. degree at Computer Science & Engineering department, IIIT Guwahati, India. He was a Project Fellow at IIT Guwahati. His research interests include low-power wireless sensor networks and Internet of Things.

Rakesh Matam [M’14] received the bachelor’s degree in computer science from Jawaharlal Nehru Technological University at Hyderabad, Hyderabad, the masters degree from Kakatiya University Warangal, India, and the Ph.D. degree in computer science from IIT Patna in 2014. In 2014, he joined the Department of Computer Science, IIIT, as an Assistant Professor. He is currently a member of the Design and Innovation Center, IIIT Guwahati, and a Principal Investigator of a funded research project sponsored by the Government of India. His research interests include wireless networks, network security, and cloud computing. He has also served as a member in organizing committee in numerous international conferences.

Mithun Mukherjee [S’10–M’16] received the Ph.D. degree in electrical engineering from the Indian Institute of Technology Patna, Patna, India, in 2015. Currently, an assistant professor with the Guangdong Provincial Key Laboratory of Petrochemical Equipment Fault Diagnosis, Guangdong University of Petrochemical Technology, Maoming, 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 communications, fog computing, and ultra-reliable low-latency communications.

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 20102012 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 an 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.