6TiSCH – IPv6 Enabled Open Stack IoT Network Formation: A Review

ALAKESH KALITA and MANAS KHATUA, Indian Institute of Technology Guwahati

The IPv6 over IEEE 802.15.4e TSCH mode (6TiSCH) network is intended to provide reliable and delay bounded communication in multi-hop and scalable Industrial Internet of Things (IIoT). The IEEE 802.15.4e Time Slotted Channel Hopping (TSCH) link layer protocol allows the nodes to change their physical channel after each transmission to eliminate interference and multi-path fading on the channels. However, due to this feature, new nodes (aka pledges) take more time to join the 6TiSCH network, resulting in significant energy consumption and inefficient data transmission, which makes the communication unreliable. Therefore, the formation of 6TiSCH network has gained immense interest among the researchers. To date, numerous solutions have been offered by various researchers in order to speed up the formation of 6TiSCH networks. This article briefly discusses about the 6TiSCH network and its formation process, followed by a detailed survey on the works that considered 6TiSCH network formation. We also perform theoretical analysis and real testbed experiments for a better understanding of the existing works related to 6TiSCH network formation. This article is concluded after summarizing the research challenges in 6TiSCH network formation and providing a few open issues in this domain of work.

CCS Concepts: • **Networks** → **Link-layer protocols**; **Network simulations**; **Network performance analysis;**

Additional Key Words and Phrases: Industrial Internet of Things, IEEE 802.15.4e, 6TiSCH, network formation

ACM Reference format:
Alakesh Kalita and Manas Khatua. 2022. 6TiSCH – IPv6 Enabled Open Stack IoT Network Formation: A Review. *ACM Trans. Internet Things* 3, 3, Article 24 (July 2022), 36 pages.
https://doi.org/10.1145/3536166

1 INTRODUCTION

The **Internet of Things** (**IoT**) allows physical objects (referred to as "*things*" or "*nodes*" in IoT) to communicate among themselves. IoT connects almost every physical object that is found in homes, healthcare systems, industries, and various transportation systems [1–3]. Nodes exchange their surrounding information among themselves, which helps the computational system interact with the physical world. The information received from the nodes is used for analyzing the current environmental situation and based on the analysis, IoT changes the behavior of the environment if needed. Various application domains such as industry [4, 5], health-care [6, 7], smart home [8, 9], smart city [10, 11], smart grid [12, 13], smart transportation [14], and smart precision agriculture [15] are the few applications of IoT. The nodes in IoT are called *smart* [1, 16] in three aspects:

Authors' addresses: A. Kalita and M. Khatua, Indian Institute of Technology Guwahati, Computer Science and Engineering Department, Assam 781039, India; emails: alakesh.kalita1025@gmail.com, manaskhatua@iitg.ac.in.

2577-6207/2022/07-ART24 $15.00
https://doi.org/10.1145/3536166

(i) they can *sense or monitor* their surroundings, (ii) they can *communicate* or *exchange* their information with other nodes, (iii) they can *change the behavior* of their surroundings. Nodes use different sensors for sensing or monitoring the environment, and most commonly, it usages wireless communication for exchanging information. The behaviors of the environment are changed (known as *actuation*) using some mechanical devices which work based on the input provided by the nodes [17, 18]. In brief, IoT can improve the efficiency, safety, and security of its various applications in the physical world through sensing, communicating, and performing actuation on the environment.

Most of the IoT devices[1] are resource-constrained in terms of processing capacity (which has only a few **kilobytes** (**KBs**) of RAM), memory (also in KBs), and energy (mainly battery operated) [19]. Nevertheless, IoT should provide high reliability and delay bounded energy-efficient communication in most of its applications. Therefore, it is very challenging to implement and fulfill various requirements of different IoT applications with the resource-constrained devices.

In recent years, because of the increasing demand for wireless technology for both sensing and actuating, many standards have been introduced for the IoT networks based on the requirements of different applications [19–21]. They include *ZigBee* [22], *IEEE 802.15.4* [23], *Bluetooth* [24], *WirelessHART* [25], and *ISA-100.11a* [26]. These standards use 2.4 *Ghz* frequency band, whereas **Narrowband Internet of Things** (**NB-IoT**) [27], IEEE 802.11.ah [28], **Low Power and Long Range Wide Area Network** (**LoRaWAN**) [29] are the few standards that use $Sub-1\ GHz$, frequency band. Among these standards, *IEEE 802.15.4* is the most widely used standard for resource-constrained ***low power and lossy networks*** (**LLNs**) [19]. It defines both the ***Physical*** (**PHY**) and ***Medium Access Control*** (**MAC**) layer. However, many works such as [30, 31] mentioned from their performance evaluations of the IEEE 802.15.4 standard that it suffers from unbounded delay, low reliability, less protection against interference, and multi-path fading, hence, not suitable for most of the IoT applications. The main reason for these issues is the usage of a single channel in 2.4 *GHz* frequency band. On the other hand, ZigBee suffers from interoperability and scalability issues as it does not use existing infrastructure for communication and it is completely proprietary. Similarly, WirelessHART and ISA-100.11a suffer from scalability issues as both of them follow centralized approaches for establishing and maintaining the communication among the nodes. The $Sub-1\ GHz$-based standards are designed for long-range but low data rate oriented applications. Note that, to achieve higher data rates but, at the cost of range, LoRaWAN can be operated on 2.4 *GHz*.

Therefore, to fulfill the increasing demands of sensor/actuator-based technologies and deal with the issues of existing standards, ***The Institute of Electrical and Electronics Engineers Standards Association*** (**IEEE-SA**) published *IEEE 802.15.4e* [32] amendment in 2012. The IEEE 802.15.4e is an extension of the existing IEEE 802.15.4 [23]. IEEE 802.15.4e enhances the functionalities of both the PHY and MAC layers compared to the existing IEEE 802.15.4 to support different application requirements. For this, IEEE 802.15.4e mentioned five different MAC behavior modes based on different applications' requirements. The MAC behaviors are **Time Slotted Channel Hopping** (**TSCH**), **Deterministic and Synchronous Multi-Channel Extension** (**DSME**), **Low Latency Deterministic Network** (**LLDN**), **Asynchronous Multi-Channel Adaptation** (**AMCA**), Radio Frequency Identification Blink (BLINK). A detailed survey of IEEE 802.15.4e and all MAC modes is given in [33]. Note that among these MAC behaviors, both the TSCH and DSME are included in the recently revised version of IEEE 802.15.4-2016 [34] because of their applicability in different application domains.

[1]We use the terms device, mote, and node interchangeably in this manuscript as all of them mean the same.

IEEE 802.15.4e supports multiple transmissions at a time using multiple physical channels (max. *16 channels*) using the 2.4 *GHz* frequency band. Hence, it increases the overall throughput of the networks. Apart from this, IEEE 802.15.4e also allows the nodes to change their physical channels after every transmission, which is known as *channel hopping* to get rid of multi-path fading and interference on the channels. So, IEEE 802.15.4e also increases communication reliability.

Among the five MAC behavior modes, TSCH gains attention among the researchers. It provides deterministic channel access to the nodes, and so provides reliable and delay bounded communication [35]. Therefore, TSCH mainly opts in industrial automation, process monitoring, oil, and gas industry as a part of the *Industry 4.0* revolution where information should be transmitted reliably within a specific time period [32, 34, 36–38]. In this article, we mainly focus on TSCH.

As IEEE 802.15.4e defines only the PHY and MAC layers for providing IPv6-based Internet connectivity to the devices, the ***Internet Engineering Task Force*** **(IETF)** created the ***IPv6 over the TSCH mode of IEEE802.15.4e*** **(6TiSCH) Working Group (WG)**. This WG aims at providing interoperability between the IEEE 802.15.4e TSCH link layer and IP-based upper layer protocol stack used for IPv6 connectivity. The 6TiSCH networks use these TSCH link layer and upper layer protocols for establishing multi-hop communication among the nodes [39].

Nevertheless, several issues have been left open by the standard such as resource allocation during the formation of 6TiSCH network, mechanism/scheduling for exchanging data, and maintaining synchronization among the nodes, to name a few. Network formation of 6TiSCH network gains the attention of the researchers because TSCH allows each node to change its physical communication channel after each transmission irrespective of whether the previous transmission is successful or not. On the other hand, the pledges[2] do not know in which channel transmission of control packets is happening and at what time. Furthermore, various control packets are required by the pledges to join a network. Therefore, pledges need to randomly scan each physical channel to receive valid control packets. For that, pledges need to keep their radios active all the time, which in turn causes huge energy consumption. As the devices used in IoT has limited power source and it is not always feasible to change the power source such as the battery, so huge energy consumption by the nodes during its network joining process can severely reduce its lifetime, and so the lifetime of the entire network. Moreover, nodes need to contend to access the transmission channel to transmit their control packets as the channels are shared for control packet transmission. Hence, the possibility of packet collision in the wireless medium increases, and the chance of accessing a transmission channel by a node decreases with the increasing number of nodes in the network. This results in increasing joining time and energy consumption of the pledges. Therefore, many researchers aimed at reducing the joining time of the pledges so that their energy consumption can be reduced, which in turn increases the lifetime of the entire network. Further, a node is allowed to transmit its sensory data packet and control packets only after it completely joins the 6TiSCH network. Hence, the joining time of the pledges is critical to improving the efficiency and reliability of the networks, and so the applications. Therefore, the initial network formation task needs to be done efficiently in order to improve the lifetime and reliability of the networks.

1.1 Contributions

Although few recently published survey works such as [4, 30, 39] considered IEEE 802.15.4e but these works only gave an overview of the enhancement of IEEE 802.15.4e over the existing IEEE 802.15.4. These works were not specifically dedicated to IEEE 802.15.4e TSCH or any research issues related to 6TiSCH networks. On the other hand, few works such as [41–43] surveyed the

[2]IETF used the term "pledge" to designate a new joining node, which has not yet completed the join process in a given secure network and is therefore not trusted by the network [39, 40]. So, we refer to a new node as a pledge in this article.

Table 1. List of Frequently used Terms with their Corresponding Meanings

Term	Description
Timeslot	It is the smallest fixed time duration to transmit one packet and receive its acknowledgment. The timeslot duration is the same for all the nodes in a network.
Slotframe	It is a collection of several timeslots. Slotframe repeats one after another and a timeslot repeats after each slotframe. The slotframe length is same for all the nodes in a network.
Slot offset	It denotes the timeslot (index number) within a slotframe which ranges from (0 to slotframe length -1)
Channel offset	It is an integer number that ranges from (zero to maximum number of channel used - 1) in a network where each integer is mapped with a particular physical channel that is used in the network.
Cell	A cell is represented by the touple [`slot offset` and `channel offset`], which basically denotes at what time, which channel should be used for transmitting either data packet or control packet by the nodes.
Shared cell	In shared cell, nodes transmit their control packets by performing TSCH-CSMA/CA channel access mechanism to access the channel associated with the cell.
DIO	DODAG Information Object. This control packet is multicasted/unicasted by the joined nodes' RPL layer to construct the DODAG.
EB	Enhanced Beacon frame which is broadcasted by the TSCH MAC layer.

scheduling approaches used for data transmission in 6TiSCH networks. However, these works did not consider the formation of 6TiSCH networks. As per our knowledge, till now, only one published article [41], which partially surveyed the works related to network formation in IoT. The work in [41] considered only a few works that are related to the initial step of 6TiSCH network formation i.e., TSCH node association/synchronization. Note that the association/synchronization of a node in a TSCH network does not mean that the node has completely joined in the 6TiSCH network. TSCH node association is the initial step or first step of 6TiSCH network formation. Nodes need to receive a few more control packets to join the 6TiSCH networks completely after associating or getting synchronized with the TSCH networks. Please remind that we use the words association and synchronization interchangeably in this article. More discussion about TSCH node association/synchronization and 6TiSCH network formation are given in Section 2.4.

Recently, few works had been published which considered the complete formation of 6TiSCH networks. Therefore, in this article, a brief discussion about almost all the works related to both the TSCH node association and 6TiSCH network formation is provided. Apart from the literature review and a state-of-the-art survey on the works related to TSCH synchronization and 6TiSCH network formation, comparison-based performance evaluation of the recent works related to 6TiSCH network formation using Markov Chain-based analytical model and testbed experiments are provided. For the testbed experiments, at first, we implement the existing schemes on open source Contiki-NG **Operating System** (**OS**) [44], which is mainly designed for IoT devices, and compile the schemes to generate executable binary files for the IoT devices. Subsequently, executable binary files are flashed on the devices available on FIT IoT-LAB [45], which is an open IoT testbed. The results obtained from the testbed experiments are provided and discussed briefly so that it helps the researchers to design more efficient algorithms for 6TiSCH network formation in the future. In brief, the major contributions of this article are as follows:

—At the beginning, this article provides a comprehensive overview of the IEEE 802.15.4e TSCH based 6TiSCH network followed by its formation process.
—A detailed literature review and state-of-the-art survey, that includes almost all the published works related to TSCH synchronization and 6TiSCH network formation process is provided.
—An Markov Chain-based analytical model is provided for the theoretical analysis of the existing schemes related to 6TiSCH network formation.
—We analyze the performance of a few recently published works related to 6TiSCH network formation in terms of joining time of the pledges and their energy consumption using testbed experiments at FIT IoT-LAB.
—Finally, we discussed about the future research scope and directions for the 6TiSCH network formation.

The rest of the article is organized as follows. In Section 2, a brief overview of IEEE 802.15.4e TSCH followed by the discussion about 6TiSCH network protocol stack and its network formation process are provided. Section 3 provides the literature survey on TSCH network synchronization and 6TiSCH network formation in two different subsections. Subsequently, we provide theoretical analysis in Section 4 and testbed experimental analysis in Section 5 of the existing schemes. Finally, a brief discussion of the open research challenges for the 6TiSCH network formation is provided in Section 6, and the conclusions are drawn in Section 7.

2 BACKGROUND

2.1 Overview of IEEE 802.15.4e TSCH

TSCH divides time into small and fixed duration *timeslot* that repeats over time. A single timeslot duration is long enough, usually 10 *ms*, to transmit a packet (max. 127 *Bytes* long) and receive its ***acknowledgment*** (**ACK**), if needed. The collection of several timeslots is known as *slotframe*, which repeats over time, and after each slotframe, one particular timeslot repeats. A node may be in a ***receiving*** (**Rx**), ***transmitting*** (**Tx**), or *idle* state in a timeslot. A pair of nodes decide their communication timeslot using a *scheduling* function. Therefore, the timeslot used by a pair of nodes is not used by other pairs in the network. In this way, TSCH provides deterministic channel access to the nodes to exchange their information. This results in deterministic delay bounded data packet delivery. However, sometimes more than one node is allowed to transmit in a timeslot, which is known as a *shared timeslot*. Basically, nodes perform TSCH-CSMA/CA (*carrier-sense multiple access with collision avoidance*) mechanisms in a shared timeslot to transmit their unicast packet in order to reduce collision and only control packets are transmitted in the shared timeslot.

To deal with the problems of interference and multi-path fading on a particular channel, IEEE 802.15.4e TSCH takes the advantage of the availability of multiple channels for communication. It allows a node to change its physical communication channel after each timeslot. This mechanism is called channel hopping. Therefore, a pair of nodes also need to schedule one transmission channel along with the timeslot. The nodes computed the physical channel for their communication by using Equation (1), which provides a pseudo-random channel hopping sequence.

$$f = F[(ASN + \texttt{channel offset}) \; mod \; N_{channels}], \tag{1}$$

where f denotes the calculated `channel index` associated with a particular frequency, F denotes the sequence of channel hopping i.e., `lookup table` for channels. *ASN* is the *absolute slot number*, which denotes the number of total timeslots that have elapsed from the starting of the network. So, *ASN* is an integer number that starts from 0 and gets incremented by one after each timeslot.

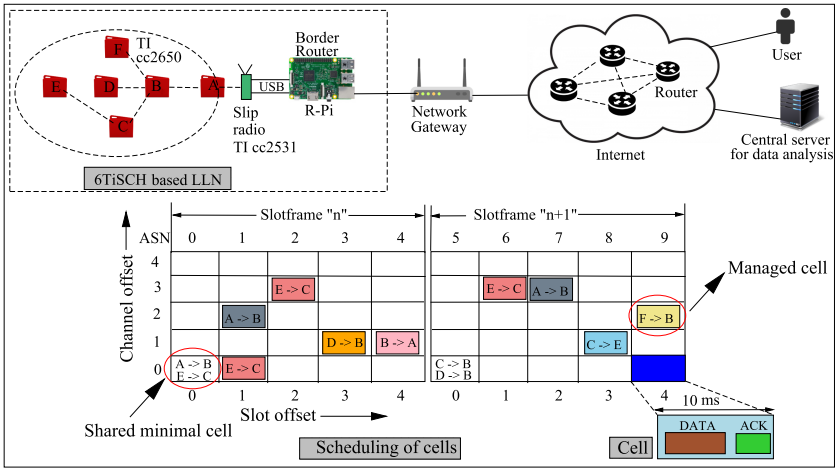


Fig. 1. A 6TiSCH network with IPv6-based Internet connectivity and scheduling of communication cell.

The `channel offset` is a fixed integer assigned to both the sender and receiver by the scheduling algorithm, and $N_{channels}$ denotes the number of channels used in the network.

The combination of timeslot and its associated channel is called a *cell*. The underlying scheduling algorithm needs to decide both the timeslot and channel i.e., cell for a pair of nodes for their communication. Figure 1 shows a 6TiSCH-based LLN connected with the traditional IPv6-based network through a network gateway, where the user(s) can access the data in real-time, or data can be stored in the central server(s) for further processing using IPv6 connectivity. The figure also shows how the communication cells are managed/scheduled by the underlying scheduling algorithm in the repeated slotframes for the same multi-hop 6TiSCH network. The cell at slot offset 0 and channel offset 0 is the *shared cell* where all the nodes transmit their control packets.

2.2 6TiSCH Protocol Suite

In 2007, IETF formed ***IPv6 over Low power Wireless Personal Area Network*** **(6LoWPAN) Working Group (WG)** to provide IPv6-based Internet connectivity to the IEEE 802.15.4 networks [46]. For this, the *6LoWPAN adaptation layer* is added on top of Layer-2 [47]. IETF has recently created the 6TiSCH-WG to provide Internet connectivity to the devices that run TSCH in their link-layer with the 6LoWPAN adaptation layer. Therefore, 6TiSCH bridges the IETF's upper layer protocol stack with the IEEE 802.15.4e TSCH to provide IPv6-based Internet connectivity to the LLN devices. The overall protocol stack is shown in Figure 2, where the 6TiSCH stack is rooted on the top of the IEEE 802.15.4 physical layer [19, 39, 48]. RFC8180 [49] is released for 6TiSCH network bootstrapping or formation, which provides the basic procedure to form a network and information about minimum resource in terms of cells used for forming the multi-hop 6TiSCH networks. On the other hand, works in [50], RFC9031 [40] describe the procedure for secure enrollment of nodes during their joining process. RFC8137 [51] and RFC8480 [52] described the distributed cell management protocols and [53–56] are a few scheduling schemes used for scheduling the data packet transmission cells. The 6LoWPAN adaptation layer includes stateless header compression and fragmentation technique in RFC4944 [57] and contextual header compression techniques in RFC6282 [58], RFC8025 [59]. RFC8138 [60]) is used for providing the IPv6-based Internet connectivity to 6LoWPAN. The **Routing Protocol for Low Power Lossy Networks (RPL)** [61] is used in the network layer to organized the network in the form of **Destination Oriented Directed**

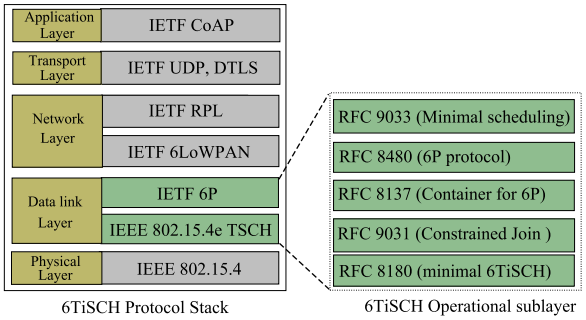


Fig. 2. The complete 6TiSCH network protocol stack.

Acyclic Graph (**DODAG**) topology following the RFC6552 [62]. On the other hand, **Constrained Application Protocol** (**CoAP**) [63] is used in the application layer for enabling low-overhead secure RESTful interaction. 6TiSCH WG is working on providing interoperability between IEEE 802.15.4e TSCH and IETF's upper layer protocol stacks. This WG deals with the scheduling of communication cells among the nodes based on the received signals from the TSCH link layer and other upper layers. Communication cells are scheduled for all types of packets (i.e., data or control packets). Apart from this cell scheduling, 6TiSCH WG also deals with resource allocation (i.e., number of shared cells per slotframe) for efficient 6TiSCH network bootstrapping [49], which will be discussed briefly in this article.

2.3 Current Development in 6TiSCH Network

Most of the RFC standards used in 6TiSCH protocol stack such as RFC8180 [49], RFC4944 [57], RFC6282 [58], RFC8025 [59] and RFC8138 [60], RFC6550 [61] are already implemented, whereas implementation of some of the RFCs such as RFC9031 [40], RFC8480 [52] are under development. The complete implementations of 6TiSCH protocol stack are available on various open-source platform such as Contiki-NG (Contiki) [44], OpenWSN [64], RIOT [65]. For the evaluation of the 6TiSCH networks with different network settings without any real-world deployment, different emulator/simulator such as Cooja [44], OpenSim [64], and 6TiSCH simulator [66] are already developed. Apart from this, nowadays, there are many commercial hardware devices such as TelosB, Texas Instrument's CC2650/1350, OpenMote-CC2538, IoT Lab M3/A8, Zolertia, Jenic JN516x, to name a few, available in the market that supports the 6TiSCH protocol stack. Hence, the available open-source implementations of 6TiSCH protocol stack in different platform and commercial devices which support 6TiSCH expedite the use of 6TiSCH network in various IoT applications such as smart home, smart industry, and smart city.

2.4 Formation of 6TiSCH Network

The formation process of a 6TiSCH network is initiated by its **join registrar/coordinator**[3] (**JRC**) (or RPL root) by periodically broadcasting the **Enhanced Beacon** (**EB**) frame. EB contains the basic information of a network such as JRC-id, duration of a timeslot, number of timeslots in a slotframe, channel hopping sequence, and location of the shared cell. In the beginning, the pledges do not have such prior information (that an EB contains) to join a network, specifically, the time

[3]Please note that the JRC can be co-located with the RPL DODAG root or can be in the cloud [39]. Most of the works consider JRC is co-located with the RPL root, and so, in this article, we also consider that JRC is co-located with RPL root. In brief, we denote both the JRC and RPL root as a single node throughout the article.

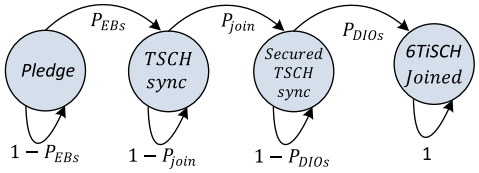


Fig. 3. Different states of a pledge during its network joining.

and the channel where the joined nodes transmit their control packets. Therefore, the pledges turn on their radios and randomly start scanning on all the available channels one by one to receive EB frame. The pledges switch their scanning channel from one channel to another after a specific amount of time and keep scanning until they receive valid EB. When a pledge receives a valid EB from an already joined node (it can be the JRC or any other already joined nodes), it becomes a TSCH synchronized node. If the pledge receives several EBs from different joined nodes, it selects the best-joined node as its parent based on the value mentioned in the `Join Metric` field of the received EB frames. Now, the TSCH synchronized node knows the time and channel (i.e., `[slot offset, channel offset]` in a slotframe) where the control packets will be transmitted from the received EB. Therefore, from now onward, it actives its radio only in the shared timeslot, which saves energy. Note that the cell present in the shared timeslot is called a *shared cell*. After getting synchronized with the TSCH network, the node exchanges **join request** (**JRQ**) and **join response** (**JRS**) frames with the JRC by taking the help of its parent node (called *join proxy*) for secure enrollment [50], RFC9031 [40]. Once the secure enrollment is done, the TSCH synchronized node should receive a valid DIO packet to become a 6TiSCH joined node. DIO packets are *multicasted/unicasted* by the routing layer (RPL) of the joined nodes. It contains all the necessary routing information required to join the DODAG routing tree constructed by the RPL for both upward and downward routing operations. As a TSCH synchronized node joins the RPL's DODAG topology after receiving the DIO packet, the 6TiSCH joined node is also called RPL joined node. Now, the 6TiSCH joined node is eligible to transmit its control packets for further expansion of the network. The formation of the entire network gets completed when all the pledges complete their joining process one by one. In brief, the journey of a pledge can be depicted by the following states: new node/pledge → TSCH synchronized node → TSCH secured joined node → RPL/6TiSCH joined node. During this whole journey, the pledge receives mainly EB, JRS, and DIO control packets, and it transmits JRQ, and DIS control packets. These states are shown in Figure 3.

As a pledge moves from one state to another state depending on the control packet receiving probability in the previous state, therefore, different control packet receiving probabilities in a slotframe by a pledge such as P_{EB_S}, P_{DIO_S}, and P_{join} are shown in the figure. In brief, P_{EB_S}, P_{DIO_S}, and P_{join} denote the probabilities of successfully receiving EB frame, DIO packet, and exchanging of JRQ and JRS frames together in a slotframe, respectively. In the last state (called the absorbing state) of the state diagram, a pledge becomes a 6TiSCH joined node.

Please note that throughout the full life cycle of 6TiSCH networks, nodes need to re-join (or re-synchronize) the network several times in different occasions such as de-synchronization of the nodes due to clock drift, DODAG reset, DODAG version changed, link lost, to name a few. During re-joining, the de-synchronized nodes behave like fresh pledges. Hence, the de-synchronized nodes follow the same procedure what a new joining node follows, which requires some amount of time and energy to complete the process. Therefore, the required amount of time and energy consumption during network formation and nodes' re-joining play an important role in the overall network lifetime and its efficiency.

3 TSCH AND 6TISCH NETWORK FORMATIONS

Researchers have separately considered the TSCH synchronization state and 6TiSCH joining state of the pledges during the formation of 6TiSCH networks. In the beginning, a pledge needs to keep its radio active all the time for the EB frame which causes more energy consumption. Therefore, researchers tried to reduce the EB scanning time. This scanning time can be called TSCH synchronization time because a pledge becomes a TSCH synchronized node after receiving its first EB. Subsequently, to form a multi-hop 6TiSCH network, the TSCH synchronized nodes should get valid DIO packets from their selected parents. Only the 6TiSCH joined nodes are allowed to transmit their control packet for further expansion of the multi-hop network. Furthermore, quick construction of DODAG also helps in efficient sensory data packet transmission [67, 68]. Therefore, it is important that the TSCH synchronized node should quickly join the DODAG constructed by RPL. Accordingly, researchers tried to reduce this 6TiSCH joining time of the TSCH synchronized nodes, so that a multi-hop 6TiSCH network can be constructed in less time. Note that the construction of DODAG and the formation of 6TiSCH network happen together because a TSCH synchronized node becomes a 6TiSCH joined node after receiving the DIO packet while the DIO packet is used to construct the DODAG.

Therefore, we divide our literature study on 6TiSCH network formation into two parts based on prior publications on the topic. The studies that considered TSCH network synchronization are included in the first section. These studies focused solely on the transmission of EB frames in order to create single-hop TSCH networks. The works that considered the formation of a multi-hop 6TiSCH network, on the other hand, are included in the second half. For 6TiSCH network formation, the transmission of both the EB frame and DIO packet is required. Hence, 6TiSCH network formation can be described as an extension of TSCH network formation in which nodes wait for DIO packets after getting an EB. Before proceeding towards the detailed review of the existing works on TSCH network synchronization and 6TiSCH network formation, in the next section, we provide our search methodology to carry out our literature search.

3.1 Related Literature Searching

The related literature searching was carried out using the “SCOPUS” and “Web of Science” databases. However, both the databases yielded the same articles many times when we searched using related keywords such as “TSCH”, “6TiSCH”, “joining”, and “formation” in the title, abstract and keyword fields. For example, SCOPUS, and Web of Science had shown 27 and 14 research articles, respectively, when we used the search keywords “TSCH” and “joining”. The results were then manually filtered to remove the articles which are neither related to TSCH network formation nor 6TiSCH network formation. For example, we excluded 7 and 3 articles from the list of articles that are found using the keywords “TSCH” and “joining” in SCOPUS and Web of Science databases, respectively. These excluded articles mainly focus on different areas such as TSCH or 6TiSCH scheduling, performance comparison of MAC protocols, simulation study, and energy harvesting. In addition to these search results, we included the 6TiSCH minimal configuration (RFC8180) [49] as it describes minimum resource allocation during 6TiSCH network formation. Finally, after keyword based filtering and manual removal of duplicate entries, we selected 22 research articles for our review. These selected research articles were published during 2014–2021. Out of these selected articles, 13 articles are related to TSCH network formation and the rest of the 9 articles are related to 6TiSCH network formation. In this article, we have analyzed these 22 research articles to show the comparative study from the viewpoint of 6TiSCH network formation.

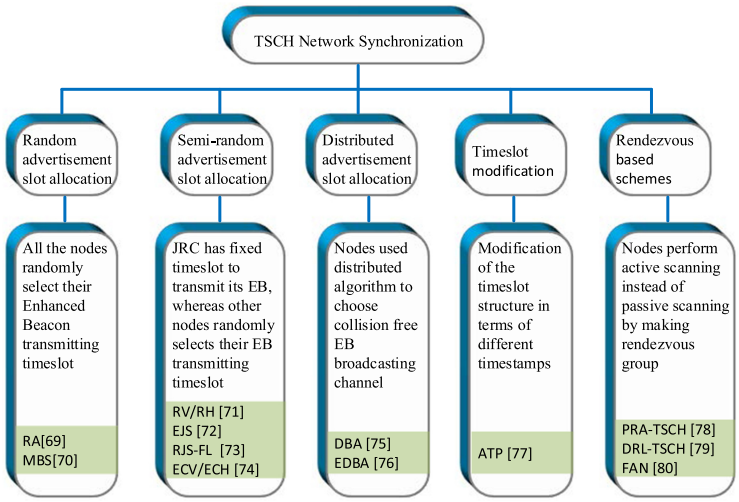


Fig. 4. Taxonomy of TSCH network synchronization schemes.

3.2 TSCH Network Formation

This section discusses the works that considered TSCH network synchronization, i.e., considered only the broadcasting of the EB frame during the synchronization process. When a pledge receives its first EB frame, it becomes a TSCH synchronized node, and the time between when it starts scanning for EB frame and when it receives the first EB frame is referred to as the pledge's TSCH synchronization time. As shown in Figure 3, a pledge becomes a TSCH synchronized node when it successfully receives EB frame (i.e., with the probability P_{EB_S}) from any of the joined nodes(s) and reach the state 2. So, state 2 represents the state of a TSCH synchronized node in Figure 3 and the time required to reach this state from the first state (i.e., pledge) is called TSCH synchronization time.

Researchers considered different design characteristics while designing their proposed approaches for faster synchronization of the TSCH network. Figure 4 shows the taxonomy of the existing TSCH network synchronization schemes, where researchers mainly considered different advertisement slot allocation policies such as random [69, 70], semi-random [71–74], and deterministic [75, 76]. In random advertisement slot allocation, already synchronized nodes transmit their EB frame by randomly choosing any slot as their advertising slots. Note that control packet (specifically, EB frame) transmitting cells are called advertisement slots. On the other hand, in semi-random allocation, fixed advertisement slot(s) are assigned to the JRC whereas other synchronized nodes chose their advertisement slots randomly. In deterministic advertisement slot allocation, fixed advertisement slots are assigned to the synchronized nodes. Apart from advertisement slot allocation, few works modified the advertisement timeslot format [77], and few other works proposed (e.g., [78–80]) rendezvous-based active TSCH synchronization schemes. All these existing works related to the TSCH network synchronization are briefly discussed one by one as follows:

3.2.1 Random Advertisement Algorithm (RA). Guglielmo et al. performed a theoretical analysis of TSCH network synchronization in [69]. The authors designed a Markov Chain-based analytical model to find the relation between the average synchronization time of the pledges and beacon transmission probability of already synchronized nodes. Authors observed from their analysis that

the synchronization time of the pledges is inversely affected by the number of channels used for beacon transmission. Subsequently, the authors suggested to use more channels when more nodes join the network. Following that, the authors proposed a ***random advertisement algorithm*** (**RA**). Using their scheme, the authors observed that the overall synchronization time decreases when the number of nodes increases in the network. Although the authors suggested various important points for improving the TSCH synchronization time, their work is limited to the analytical modeling, which used only one channel for EB transmission. However, TSCH network allows multiple channels to get rid of interference, and multi-path fading on a single channel, and improve the overall network throughput.

3.2.2 Random Vertical (RV) and Random Horizontal (RH). Vogli et al. proposed ***random vertical*** (**RV**) filling and ***random horizontal*** (**RH**) filling schemes for faster synchronization of TSCH networks in [71]. The authors used the term *multi-slotframe* (multi-SF), which is nothing but a collection of several slotframes together and repeats one after another. In RV, the coordinator (or JRC) transmits at the first slot of a multi-SF using the channel offset 0. On the other hand, other synchronized nodes transmit using any random channel offset except the channel offset 0. Therefore, the EBs transmitted by the synchronized node(s) never collided with the EB transmitted by the JRC. Note that using RV, all the nodes transmit their EBs simultaneously, i.e., at the first timeslot of a multi-SF. It increases the EB reception probabilities of the pledges. However, as the nodes transmit their EBs in different channel offsets, this can lead to de-synchronization between parent-child pairs. On the other hand, in RH, all the nodes, including the JRC, use the same channel offset 0, but use different slotframes of a multi-SF to transmit their EBs. De-synchronization between the parent-child pairs is also possible in this RH scheme. Furthermore, collision is possible in both RV and RH schemes when two or more nodes select the same channel offset, and slotframe, respectively. The authors designed probabilistic-based theoretical models for both the RV and RH schemes to estimate the synchronization time of the pledges. The designed analytical models were later validated using simulation experiments. The results from both the theoretical and simulation experiments showed that the performance obtained by both RV and RH is almost the same when the multi-SF length and number of used channels are the same. However, no comparison-based evaluation with any benchmark scheme is done by the authors. They only showed the relation between the TSCH synchronization time and the number of synchronized nodes.

3.2.3 An Efficient Joining Scheme (EJS). The authors in [72] proposed a lightweight synchronization scheme for TSCH networks. In **Efficient Joining Scheme** (**EJS**), a slotframe is divided into two parts, named as advertisement and communication planes. The advertisement plane contains the advertisement slot(s) and the communication plane is reserved for transmitting the data packets. The author's pre-configured the number of advertisement slot(s) at the beginning of their experiment and compared their scheme with the existing RV and RH schemes. Their obtained results showed improvement over the RV and RH schemes. However, the authors did not calculate the optimal number of advertisement slots for a particular network configuration. Rather, they used pre-configured static values for their experiments. Again, the energy consumption of the nodes, throughput, and end-to-end packet delivery latency of the network get affected using more advertisement slots. It is because the nodes need to keep their radios active in all the advertisement slots, and the communication slots are converted into advertisement slots. Furthermore, in EJS, collision is possible when the allocated advertisement slots are not scheduled properly.

3.2.4 Model-based Beacon Scheduling (MBS). Guglielmo et al. [70] proposed a beacon scheduling algorithm based on their proposed discrete time Markov Chain model for TSCH network formation. The authors described the behavior of a pledge during its network joining process using

a Markov Chain model, and subsequently, formulated an optimization problem. The optimization problem minimizes the average synchronization time by efficiently scheduling the EB transmission slot. Later, the solution of the optimization problem is used in the **Model-based Beacon Scheduling (MBS)** scheme for determining the unique beacon transmission link of the nodes. The authors analyzed their MBS scheme theoretically and showed that MBS outperforms the existing schemes such as RA, RV, and RH in terms of average synchronization time. It is noteworthy that MBS behaves like RV when both the schemes used a same number of channels. On the other hand, MBS behaves like RH, when both the schemes used the same slotframe length. The drawbacks of MBS are like it affects the nodes' energy consumption, data transmission schedule as it uses several EB transmission slots per slotframe. Furthermore, the de-synchronization between the parent-child pairs is possible in MBS as the nodes use different channels to transmit their EB frames.

3.2.5 A Rapid Joining Scheme based on Fuzzy logic (RJS-FL). P. Duy et al. proposed a fuzzy logic-based adaptive beacon transmission scheme in [73] by extending their previous work EJS [72]. This scheme determines the EB rate of the TSCH synchronized nodes dynamically depending on the number of available advertiser nodes in their surroundings. The authors also took care of the energy consumption of the nodes as the EB rate is directly related to the energy consumption of the nodes. Like the EJS scheme, a slotframe is also divided into two parts, i.e., advertisement and communication planes, where advertisement slots are kept together in the advertisement plane. A synchronized node randomly chooses a channel for broadcasting its EB frame using a slot from the advertisement slots. The authors used fuzzy logic to determine the required number of advertisement slots per slotframe depending on node density. So, the used fuzzy logic takes the number of synchronized nodes as input and provides the number of advertisement slots per slotframe as output. The authors performed theoretical and testbed experiments and showed that **Rapid Joining Scheme-based on Fuzzy logic (RJS-FL)** improves the network synchronization time compared to the RV scheme. However, the idea of a random selection of advertisement slots can lead to a collision.

3.2.6 Enhanced Random Vertical (ECV) and Enhanced Random Horizontal (ECH). In [74], Vogli et al. proposed an enhanced version of their previous approaches i.e., RV and RH, named them as **Enhanced Random Vertical (ECV)** and **Enhanced Random Horizontal (ECH)**. In RV, the JRC uses the channel offset 0, whereas other synchronized nodes use different channel offsets, and all the nodes transmit at the same timeslot i.e., at timeslot 0 of the first slotframe of a multi-SF. However, when the number of synchronized nodes exceeds the total number of channel offsets, collision becomes inevitable. To mitigate this problem, a node can choose any channel offset other than channel offset 0 from the next slotframe of a multi-SF when the previous slotframe gets filled in ECV. Furthermore, the JRC can transmit its EB in every slotframe of a multi-SF using the channel offset 0, considering it has enough power source. On the other hand, in ECH, the next channel offset is used by the nodes when there is no free slotframe available in the multi-SF using the previous channel offset. Thus, both the ECV and ECH reduced the EB collision probability in dense networks by using more channel offsets and slotframes, respectively. The authors designed analytical models for both the ECV and ECH and compared their analytical results with the experimental results performed using hardware devices. Although both the ECV and ECH improve the performance of TSCH network formation, these two schemes suffer from de-synchronization issues between the parent-child pairs, and high energy consumption of the nodes. Furthermore, it is not always true for every IoT application that JRC has enough energy sources. Again, the throughput and latency of the network are also be affected as many number of advertisement slots are used in both the schemes.

3.2.7 A Fast Joining Technique (ATP). Karalis et al. proposed a technique (***advertisement timeslot partitioning*** (**ATP**)) to transmit more number of EBs without allocating extra advertisement slots per slotframe in [77]. EBs are generally broadcasted in the advertisement slot, where the receiver does not transmit an ACK. As the TSCH timeslot length is long enough to transmit a packet and receive its ACK, half of the total timeslot duration gets wasted when an EB is transmitted. Therefore, ATP divides a timeslot into two parts and utilizes both the parts for advertising EB instead of allocating extra advertisement slots per slotframe. Hence, ATP can broadcast two EB frames in a single advertisement slot. Thus, ATP improves the EB transmission rate in the network without increasing the number of advertisement slots per slotframe. The authors performed simulation experiments on a python based simulator, and their obtained results show that ATP can improve the synchronization time of the nodes. However, in real-life scenarios, some control packets require ACKs such as unicast DIS request, and keep-alive. Therefore, ATP will not be suitable in presence of those control packets for constructing multi-hop 6TiSCH networks.

3.2.8 Deterministic Beacon Advertising Scheme (DBA). The authors Khoufi et al. proposed DBA algorithm for TSCH networks in [75]. Basically, this work reduces the EB transmission collision by distributing all the available channels among the nodes. It results in deterministic channel access to the nodes for transmitting their EBs. **Deterministic Beacon Advertising** (**DBA)** finds the unique channels for all the nodes and allows the nodes to transmit their EBs in the regularly spaced advertisement slots using the assigned channels. The advertisement nodes calculate their unique channel offset during their association using the DBA scheme to achieve collision-free beacon broadcasting. The simulation results showed that the DBA outperforms the RV and RH in terms of joining time. The results considered TSCH synchronization time of different network topologies as a function of a number of beacons transmitting nodes and beacon transmitting interval.

3.2.9 Enhanced Deterministic Beacon Advertising Scheme (EDBA). Khoufi et al. further improved their previous work (i.e., DBA [75]) in [76]. The authors proposed an EDBA for better collision-free EB broadcasting. **Enhanced Deterministic Beacon Advertising** (**EDBA)** transmits beacons using all the available channels without any collision efficiently. For validating EDBA, the authors developed a Markov Chain-based analytical model for EDBA and compared with MBS [70]. The authors also performed simulation-based performance evaluation. Their results showed that EDBA outperforms MBS in terms of the joining time of the nodes. However, no clear distinction is visible between both the DBA and EDBA schemes. Both DBA and EDBA used the same method to add collision-free advertisement slots per slotframe.

3.2.10 Parallel Rendezvous-Based Association (PRA-TSCH). In [78], Algora et al. proposed a ***parallel rendezvous-based association for TSCH networks*** (**PRA-TSCH**) scheme is to reduce the random channel scanning time of the pledges during their TSCH association. Unlike the previously mentioned works, PRA-TSCH allows the pledges to perform active scanning for EB instead of always waiting (i.e., scanning) for EB on random channels. It is done by allowing the non-associated nodes to turn up for their rendezvous during their channel scanning process. For this, the non-associated (i.e., pledge) nodes form a network among themselves by exchanging ***rendezvous beacon*** (**RB**) frames. When a pledge receives an EB from an already synchronized node, the pledge broadcasts the basic information carried by the EB among the other non-associated nodes who have participated in rendezvous. This helps the non-associated nodes to join the network quickly. Furthermore, as the pledges in the rendezvous group scan on different channels, it increases the EB reception probabilities of the pledges. Both theoretical and simulation experimental results showed that PRA-TSCH improves the TSCH association (or synchronization) time compared to

the **6TiSCH minimal configuration** (**6TiSCH-MC**) standard [49] (6TiSCH-MC is explained in Section 3.4).

However, PRA-TSCH scheme may affect the existing TSCH schedule performed by the already synchronized nodes for control and data packet transmission. Furthermore, this scheme could lead to the higher energy consumption of the nodes by unnecessarily broadcasting RB frames in sparse networks. On the other hand, collisions among the packets transmitted by the associated and non-associated nodes are possible in dense networks. Again, there is a possibility that the non-associated nodes miss the EB frame broadcasted by the already associated nodes while gossiping among themselves in the rendezvous group.

3.2.11 2-Way Parallel Rendezvous for Faster TSCH Synchronization. The authors Byeong-Hwan Bae and Sang-Hwa Chung proposed the ***distributed radio listening*** (**DRL-TSCH**) scheme in [79] to form the TSCH networks quickly. The authors used a similar concept of exchanging RB among the non-associated nodes as mentioned in PRA-TSCH [78]. However, in the DRL-TSCH scheme, nodes also respond to the received RB frames by mentioning their own channel scanning sequences (i.e., their channel index, channel array, and ID), so that a node does not scan the same channel used by the other nodes in their group. As a result, all the channels are sub-divided among the pledges, which creates the rendezvous chain. Hence, EB receiving probability increases in DRL-TSCH compared to the PRA-TSCH. It is because, in PRA-TSCH, two or more nodes may scan on the same channel, which reduces the EB reception probability. But, DRL-TSCH has over overcome this by distributing the channels among the nodes. Their simulation results show that DRL-TSCH outperforms the PRA-TSCH and 6TiSCH-MC in terms of TSCH association/synchronization time. However, the above-mentioned issue related to PRA-TSCH can also be feasible in DRL-TSCH.

3.2.12 Fast and Active Network Formation (FAN). Recently, Mohamadi et al. proposed a ***fast and active network formation*** (**FAN**) scheme for faster TSCH synchronization in [80]. Following the works in [78, 79], FAN also allows the pledges to transmit ***enhanced beacon requests*** (**EBRs**) to accelerate the (re)association process. However, unlike PRA-TSCH and DRL-TSCH, FAN takes care of the collision free EBR broadcasting. This collision-free method avoids collision among the EBR and data packets transmitted by the underlying TSCH schedule. Apart from this, FAN also proposed to use trickle-based EB transmission instead of using fixed periodic EB transmission. This trickle-based EB transmission strategy allows the nodes to change their EB transmission rate depending on the current network condition. Authors performed simulation-based experiments to evaluate their proposed scheme i.e., FAN, and evaluate it in terms of joining time as a function of number of EB transmitting nodes, a number of used channels, and channel packet delivery rate. Results showed that FAN outperforms the RV, RH schemes. However, EBRs may cause a large number of EB transmissions in the network because of the use of trickle-based EB transmission.

3.2.13 A Theoretical Model for TSCH Association Time. The authors Algora et al. designed a theoretical model for the nodes' joining process in TSCH network in [81]. The model estimates the time required by a pledge to get associated with a synchronizer node. The authors also considered the link failure possibility in their theoretical model. Basically, their proposed theoretical model finds the upper time limit to form a network with an arbitrary topology. The authors used Contiki OS-based Cooja simulator to validate their proposed theoretical model. From both the theoretical analysis and simulation experiments, the authors concluded that the formation of a TSCH network greatly depends on the number of channels used in the network and the EB transmission intervals set by the already associated nodes in the network.

Table 2. Comparison Among the Existing Works on TSCH Network Synchronization

Schemes	EB transmission rate	Number of advertisement slot	Advertisement slot position	Resource Allocation Type	Channel offset used for advertisement	Modify slotframe (SF)	Modify timelsot	EB scanning type	Improved Over
RA [69]	1/EB cycle	NIA	NIA	NIA	All	No	No	Passive	NIA
RV & RH [71]	1/Multi-SF	RV: 1 RH: >1	RV: 1^{st} slot of multi-SF RH: 1^{st} slot of each SF	Random	RV: Random RH: 0	Yes	No	Passive	NIA
EJS [72]	Implementation specific	>1	Beginning of a SF	Pre-configured	All	Yes	No	Passive	RV
MBS [70]	Implementation specific	>1	Anywhere	Random	Any	No	No	Passive	RV & RH
RJS-FL[73]	1/Multi-SF	>1	Beginning of a SF	Random	All	Yes	No	Passive	RV
ECV & ECH [74]	1/Multi-SF	ECV: >1 ECH: >1	ECV: 1^{st} slot of each SF ECH: 1^{st} slot of each SF	Random	ECV: All ECH: All	Yes	No	Passive	RV & RH
ATP [77]	Implementation specific	NIA	Implementation specific	Pre-configured	NIA	No	Yes	Passive	NIA
DBA [75]	NIA	>1	Anywhere	Distributed	All	No	No	Passive	RV & RH
EDBA [76]	NIA	>1	Anywhere	Distributed	All	No	No	Passive	MBS
PRA-TSCH [78]	1/SF	1	slot offset 0 channel offset 0	Centralized (6TiSCH-MC)	0	No	No	Active	6TiSCH-MC
DRL-TSCH [79]	1/SF	1	slot offset 0 channel offset 0	Centralized (6TiSCH-MC)	0	No	No	Active	6TiSCH-MC PRA-TSCH
FAN [80]	1/SF	1	slot offset 0 channel offset 0	Centralized (6TiSCH-MC)	0	No	No	Active	6TiSCH-MC RV & RH

3.3 Lessons Learned from TSCH Network Synchronization

It can be seen that most of the TSCH network synchronization schemes reduced the synchronization time by scheduling the EB frame, increasing the number of advertisement slots per slotframe, and the EB transmission rate. However, each of these three methods have significant drawbacks. First, as every node schedules its EB frame to make collision-free broadcasting, the parent-child pair nodes might lose the synchronization to each other while transmitting their EB frames in different channels. None of the works considered this de-synchronization issue, and so did not try to overcome it. De-synchronization is an inevitable problem when nodes do not maintain communication for a long time. Secondly, adding more advertisement slots per slotframe affects the data communication cell, which hinders the performance of the networks in terms of throughput and end-to-end delay. On the contrary, TSCH is designed to provide deterministic delay bounded and reliable communication with minimum end-to-end latency to the various IoT applications. Furthermore, collision is possible in the advertisement slots as in most of the schemes, the nodes chose their advertisement slots randomly. Therefore, it is necessary to a maintain schedule for efficient collision-free EB broadcasting considering nodes energy, and existing data communication schedule. Thirdly, the ill effect of transmitting more EB frames by a node directly falls upon the node's energy consumption, which further affects the lifetime of the overall network.

Although the rendezvous-based association schemes (i.e., [78–80]) look promising, the synchronization between the parent-child pairs needs to be addressed. Furthermore, the energy consumption of the nodes needs to be taken care of while the nodes transmit their EB request frame before associating with the TSCH network. In Table 2, we summarize all the works related to TSCH network synchronization considering different design parameters.

3.4 6TiSCH Network Formation

The studies that considered multi-hop 6TiSCH network formation are briefly described in this section. These works considered that the pledges should receive both EB and DIO packets to join in multi-hop 6TiSCH networks. After receiving the first EB, a pledge becomes a TSCH synchronized node, and when the TSCH synchronized node receives a valid DIO packet from its parent, it becomes a 6TiSCH joined node. In brief, as shown in Figure 3, a pledge becomes a 6TiSCH joined node when it reaches that final state of the state diagram and the time required to reach the final state from the initial state is called 6TiSCH joining time. A network is said to be completely formed when all the pledges present in the network become 6TiSCH joined nodes. That means

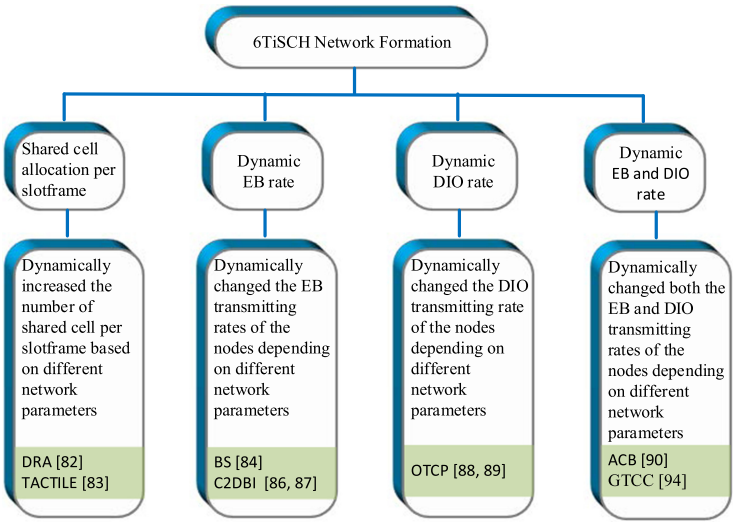
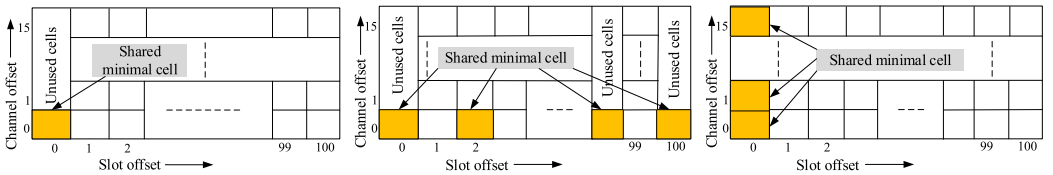


Fig. 5. Taxonomy of the 6TiSCH network formation schemes.

all the pledges received both EB and DIO packets and successfully exchanged their JRQ and JRS frames with the JRC.

The 6TiSCH WG published the 6TiSCH-MC standard [49] for allocating minimum resources during multi-hop 6TiSCH network formation. This standard mentioned three important points which need to be followed by the pledges and already joined nodes during the formation of the 6TiSCH network. The first point is that all types of control packets (such as EB, DIO, DIS, and keep-alive) should be broadcasted/unicasted in a shared cell. The control packet can not be transmitted in the dedicated cell. Only data packets are allowed to transmit in the dedicated cells. As the transmission of control packets happens only in shared cells, nodes perform TSCH CSMA/CA channel access mechanism before transmitting their control packets. The main difference between the traditional CSMA/CA and TSCH CSMA/CA is in setting the *backoff* timer by a node when it finds that the channel is busy i.e., after performing ***clear channel assessment*** **(CCA)**. The backoff timer is set in terms of number of shared cells in TSCH CSMA/CA instead of continuous time period as in traditional CSMA/CA. The second point mentioned by the standard is the number of shared cell per slotframe for transmitting all the generated control packets by all the nodes. The standard mentioned that only one shared cell per slotframe should be used for transmitting all types of control packets in the network. Note that this static allocation is made by the standard irrespective of the slotframe length. The default location of the shared cell is set by 6TiSCH-MC at `slot offset=0, channel offset=0` for all the nodes present in multi-hop 6TiSCH networks as show in Figure 6(a). This commonly shared cell helps the nodes to get tightly synchronized with each other within the network. The last but not the least important point mentioned by the standard is that a pledge should receive both EB and DIO packets to complete its joining process. Furthermore, a node is allowed to transmit its control packet for further expansion of the network only after becoming a 6TiSCH joined node. Otherwise, the standard mentioned that there is a possibility of occurring an unstable network, which can hamper the throughput, reliability, and also affect in network formation and nodes' energy consumption.

After 6TiSCH-MC several schemes have been proposed by different researchers considering different design characteristics. Taxonomy of the existing 6TiSCH network formation schemes is presented in Figure 5, where the main design characteristics are a number of shared cells per



(a) 6TiSCH-MC [49] (b) DRA [82] (c) TACTILE [83]

Fig. 6. Minimal cell allocation by different existing schemes.

slotframe, EB transmission rate, DIO transmission rate, and both EB and DIO transmission rates together. Note that although the design approaches of the existing schemes are different, their design objective is same i.e., reduce the congestion in a shared cells. In the below subsections, the summary of the existing works are presented with respect to their design characteristics as follows:

3.4.1 Dynamic Resource Allocation (DRA). Vallati et al. [82] tried to improve the 6TiSCH formation process by considering 6TiSCH-MC as a benchmark scheme. At the beginning, the authors showed that 6TiSCH-MC does not provide enough resources (i.e., shared cells) per slotframe during the formation of 6TiSCH networks using both theoretical and simulation experiments. Following their analysis, the authors proposed the **Dynamic Resource Allocation** (**DRA)** scheme to deal with the insufficient resource allocation problem. DRA increases the number of shared cells per slotframe as shown in Figure 6(b) and this allocation is done depending on the number of control packets generated in the network. So, unlike 6TiSCH-MC, DRA allocates the shared cells dynamically in every slotframe. For this, all the nodes exchange the information about the total number of buffered control packets in their EBs. Accordingly, every node calculates the required number of shared cells to transmit all the generated control packets in their surroundings. Finally, the nodes set the location of the added shared cells without any external signaling overhead. The authors mentioned that their proposed scheme significantly improves the formation time as well as network stability compared to 6TiSCH-MC by performing testbed experiments.

However, we observe that DRA has several drawbacks. DRA converts the dedicated cells into shared cells which makes more unusable cells per slotframe. It is because the nodes need to keep their radios active in the shared cells. Hence, the rest of the channel offsets become unusable apart from the channel offsets associated with the allocated shared cells. This decreases the network efficiency in terms of throughput and end-to-end latency. Furthermore, dedicated cells need to be rescheduled again because there is a possibility that some of the dedicated cells scheduled for data transmission are converted to shared cells. Apart from this, using DRA, the ***radio duty cycles*** (**RDC**) of the nodes get increased as they need to keep their radios active in all the allocated shared cell. Note that RDC denotes the amount of time a node keeps its radio active per unit time. Again, nodes are used to add shared cells based on the information received in EBs. Therefore, if a node does not receive an EB from its neighbor node(s) due to collision or link failure, the node may lose important information. Thus, this can result in an unstable network topology.

3.4.2 Broadcasting Strategy (BS). Vucinic et al. [84] performed simulation experiments on the formation of a multi-hop 6TiSCH networks considering the same resource allocation policy stated by 6TiSCH-MC. From the simulation experiments, the authors mentioned that the EB transmission probability per slotframe should be 0.1 in order to reduce the congestion in a shared cells. This EB broadcasting probability is set irrespective of the number of nodes and node density in the networks. The authors also considered the DIO transmission probability per slotframe is 0.3 in

their simulation experiments. Nevertheless, the DIO transmission rate is governed by the Trickle Algorithm [85], which increases the transmission rate of the DIO packet when there is an inconsistency in the networks. Note that during the formation of the networks, the nodes behave like there is an inconsistency in the network i.e., the nodes frequently transmit DIS packets. This forces the Trickle algorithm of the joined nodes to transmit more DIO packets in less time, which results in congestion in the shared cell. Therefore, in RPL-based 6TiSCH networks, the formation time could be wrongly estimated when fixed rate of DIO transmission is considered. Moreover, the pledges need to wait for more time to receive EB frame when the EB transmission interval is more. This waiting time becomes worst when only few nodes are deployed i.e., in a sparse network topology. So, a longer EB transmission interval can increase the TSCH synchronization time of the nodes, and so their energy consumption. Subsequently, it can increase the overall formation time of multi-hop 6TiSCH networks. Therefore, the idea of using a fixed and low EB rate all the time in order to reduce congestion in a shared cell is not always suitable for all types of networks.

3.4.3 Channel Condition Based Dynamic Beacon Interval (C2DBI). As mentioned above, the DRA scheme [82] increases the energy consumption of the nodes, and also affects the overall throughput of the networks. On the other hand, the BS scheme [84] may not be a suitable solution for sparse networks or where the number of nodes present in the networks is less. Therefore, Kalita and Khatua [86] proposed the **Channel Condition Based Dynamic Beacon Interval (C2DBI)** scheme for 6TiSCH network formation considering the same resource allocation as 6TiSCH-MC. Initially, the authors modeled the formation of a single-hop 6TiSCH network considering the transmission of all types of control packets such as EB, DIO, JRQ, and JRS. Using the analytical model and simulation experiments on Cooja Simulator, the authors showed that the formation of 6TiSCH network degrades when more nodes join the networks. The authors mentioned the reason as the increasing congestion in shared cell. Therefore, to deal with the congestion problem, the authors proposed to vary the EB generation interval dynamically i.e., depending on the congestion in shared cell. C2DBI increases the EB generation interval when there is more congestion in the shared cell and vice versa. Their simulation experimental results show that C2DBI improves the network formation performance over 6TiSCH-MC.

Later, the authors extended the same work in [87]. In this work, the authors redesigned the analytical model for multi-hop 6TiSCH networks, and also provided an analytical model for energy consumption of both the joined node and pledge during network formation. Furthermore, the authors changed the beacon generation interval method so that a small change in the congestion in shared cell can vary the beacon generation rates of the nodes. The authors performed testbed experiments in the FIT IoT-LAB testbed [45] apart from the simulation experiments. Both simulation and experimental results showed that the C2DBI improves the joining time compared to 6TiSCH-MC but fails to outperform DRA [82]. It is because C2DBI can not reduce the congestion in shared cell significantly by varying the beacon generation interval. On the other hand, DRA reduces the congestion significantly using more shared cells per soltframe. However, C2DBI outperforms both 6TiSCH-MC and DRA in terms of energy consumption because using C2DBI, nodes need to keep their radios active only in a single shared cell per slotframe like 6TiSCH-MC.

3.4.4 Opportunistic Transmission of Control Packets (OTCP). In [88], Kalita and Khatua highlighted the importance of the DIO control packets during the formation of 6TiSCH networks. 6TiSCH-MC mentioned that the EB has the highest priority over other control packets such that DIO, DIS, and keep-alive. However, Kalita and Khatua have shown in [88] that because of the EB's highest priority, the performance of 6TiSCH network formation degrades. Sometimes, the TSCH synchronized nodes need to wait a longer time to receive a DIO packet. It is because a newly generated EB frame replaces the already buffered DIO packet. This results in increasing the

6TiSCH joining time of the nodes, which further affects the overall formation time of the network. Furthermore, the authors also have shown an example where a node needs to wait for a maximum DIO generation interval to receive a DIO packet due to congestion in the shared cell and EB's highest priority. To overcome these two problems, the authors dynamically change the priority of the control packets as per the requirement of the network instead of always considering EB as the highest priority packet. The authors also proposed to assign highest priority to a DIO packet over an EB frame, when a joined node receives either JRQ frame or DIS request packet from its child node. Upon receiving these two control packets, the authors also allowed the joined nodes to reset their Trickle algorithms to generate DIO packets quickly. Note that in this work also, authors followed the same resource allocation as 6TiSCH-MC. Simulation experimental results showed that dynamic priority assignment and conditional Trickle reset mechanism improved the performance of 6TiSCH network formation compared to 6TiSCH-MC.

Apart from these two problems, the authors also considered the problem of uncertainty in accessing the shared cell in their extended version of the previous work in [89]. In addition to the above-mentioned problems and their solutions, the authors proposed another scheme for the nodes which have urgent DIO packets (i.e., when the node receives either JRQ frame or DIS request) in their transmission buffers to access the shared cell quickly. The authors improved the channel access probabilities of the nodes by modifying the existing *backoff algorithm*. Urgent DIO containing nodes are allowed to use a small *contention window* size compared to the nodes which do not have an urgent packet to transmit. Using both the simulation and testbed experiments on Cooja simulator and FIT IoT-LAB, respectively, the authors showed that all the proposed schemes together improve network formation performance in terms of pledge joining time and energy consumption compared to 6TiSCH-MC and BS schemes.

3.4.5 Adaptive Control Packet Broadcast (ACB). The same authors i.e., Kalita and Khatua studied the behavior of the Trickle algorithm [85] in 6TiSCH networks in their another work [90]. The authors designed an analytical model to show the effect of different Trickle parameters on the congestion in a shared cells, and so on, 6TiSCH network formation. From the analysis, the author's confirmed that the behavior of the Trickle algorithm needs to be changed dynamically in order to improve the formation of 6TiSCH networks. Accordingly, the authors proposed a few modifications to the existing Trickle algorithm to make it more dynamic. The first modification is done to reduce the burst transmission of DIO packets when nodes receive DIS request so that congestion can be reduced in shared cell. For this, nodes are allowed to only one DIO packet even when they receive a DIS request. To reduce the congestion further, one Trickle parameter i.e., *redundancy constant*, which controls the DIO generation and transmission rate in a network, is set dynamically instead of using a fixed value of it all the time. The value of the redundancy constant is set in such a way that it neither allows all the nodes to transmit their DIOs nor blocks them permanently. The authors also tried to provide equal DIO transmission opportunities to all the nodes by dynamically varying the *listen-only* period of each node. It is done because the default Trickle algorithm does not always provide equal transmission opportunities to all nodes, which affects the overall DODAG construction time and throughput of the networks [91, 92]. Hence, Kalita and Khatua attempted to provide equal DIO transmission opportunities to the nodes by varying the listen-only period of the nodes. As the authors considered the transmission of DIO packets in shared cell, therefore, their solution is different from other existing solutions such as [68, 93] for the same problem.

Apart from the dynamic Trickle algorithm, in order to reduce the congestion in shared cell further and to provide equal control packet (both EB and DIO) transmission opportunities to all the nodes, the authors proposed one ***slotframe window*** (**SW**) based scheme. This SW-scheme

restricts the nodes to transmit control packets more frequently. The size of the SW is varied depending on the present network condition. When the network is likely to generate more control packets, the size of the SW gets increased. This ultimately lowers the congestion in shared cell by restricting the nodes to transmit their control packets for a longer SW period. The authors performed testbed experiments on FIT IoT-LAB in order to validate their proposed scheme considering both the modified Trickle algorithm and SW-based schemes together. The obtained results showed improvement in terms of joining time and energy consumption of the nodes compared to the benchmark schemes such as 6TiSCH-MC and BS.

3.4.6 Non-cooperative Gaming-based Control Packet Transmission. All the above-mentioned works (except the work in [90]) did not consider the impact of the transmission of both EB and DIO packets together on the congestion in the minimal cell, and so on 6TiSCH network formation. The nodes in the 6TiSCH network behave non-cooperatively i.e., a node does not know and care about the control packet transmission rates of its neighboring nodes. Therefore, in some peak network situations like broadcasting of multicast DIS requests, and DODAG resetting, the minimal cell gets congested heavily, and this results in poor network performance. Furthermore, when the number of nodes increases in the network, due to fixed EB rate, congestion becomes inevitable. So, to deal with these problems i.e., to mitigate the congestion due to the heavy transmission control packets, Kalita and Khatua proposed a non-cooperative gaming approach for control packet transmission in [94]. In this gaming model, the joined nodes are considered as players. The optimal solution of the game is to maximize the control packet transmission probability of the nodes. Later, this optimal solution is used in a ***game theory-based congestion control scheme*** (**GTCC**) to calculate the number of slotframe a node should wait to transmit its next control packet. Thus, nodes are forced to restrain their transmissions frequently. Thus, GTCC reduces the congestion in minimal cells and improves the performance of 6TiSCH network formation. The authors performed theoretical analysis and testbed experiments and found that GTCC outperforms both 6TiSCH-MC and C2DBI during the formation of 6TiSCH networks.

3.4.7 Autonomous Allocation and Scheduling of Minimal/Shared Cell (TACTILE). Except for the work DRA [82], all the above-mentioned works follow the same resource allocation constraint set by the 6TiSCH-MC standard i.e., only one shared cell per slotframe. DRA shows improvement over the 6TiSCH-MC and C2DBI in terms of joining time by significantly reducing congestion in shared cell. For this, DRA allocates more shared cells (in different timeslots) per slotframe. However, DRA has several drawbacks as mentioned in Section 3.4.1. Furthermore, none of the previous works utilize all the channel offsets, and so the physical channels available at the shared timeslot (i.e., at `timeslot=0`). In DRA, the unused number of channel offsets increases significantly with the increasing number of added shared cells per slotframe. To address these issues, the work in [83] proposed a scheme namely ***autonomous allocation and scheduling of minimal cell*** (**TACTILE**), to add the multiple shared cells autonomously at `timeslot=0`. Basically, TACTILE allocates multiple shared cells (at most 3) to a node by using different channel offsets as shown in Figure 6(c). However, the node can use only one shared cell at a given slotframe for either transmission or reception. Therefore, to use the right shared cell at the right time for maintaining stable networks, TACTILE proposed a hierarchical scheduling algorithm, named CHOICE. TACTILE allocates the shared cells to each node based on its position in the DODAG tree, hence, the whole tree (or network) is hierarchically divided into a number of sub-tree (sub-network). CHOICE is totally based on the nodes' position in the DODAG. For example, if the parent is transmitting, the child node(s) should listen on the shared cell calculated based on the parent's EUI64 address. Thus, CHOICE schedules the shared cells based on nodes' hierarchical positions in the DODAG, so, the authors called it a hierarchical scheduling algorithm. Both the allocation and scheduling

Table 3. Comparison Among the Existing Works on 6TiSCH Network Formation

Scheme	Shared cell count	Dynamic Congestion Control	Dynamic Beacon Interval	Dynamic Trickle Algorithm	Resource Allocation Type	Complexity	Use of multiple channel at a time	Shows improvement over
6TiSCH-MC [49]	1	No	No	No	Centralized	O(1)	No	Benchmark
DRA [82]	>1	Yes	No	No	Distributed	O(N)	No	6TiSCH-MC
BS [84]	1	No	No	No	Centralized	O(1)	No	6TiSCH-MC
C2DBI [86, 87]	1	Yes	Yes	No	Centralized	O(1)	No	6TiSCH-MC, DRA (Energy)
OTCP [88, 89]	1	No	No	Yes	Centralized	O(1)	No	6TiSCH-MC, BS
ACB [90]	1	Yes	Yes	Yes	Centralized	O(1)	No	6TiSCH-MC, BS
GTCC [94]	1	Yes	No	No	Centralized	O(1)	No	6TiSCH-MC, C2DBI
TACTILE [83]	>1	Yes	No	No	Autonomous	O(1)	Yes	6TiSCH-MC, DRA, C2DBI

of shared cell is done without any control signal overhead. The main benefit of TACTILE is that it increases the number of shared cells per slotframe without increasing the energy consumption of the nodes and affecting the already scheduled cells for data packets, unlike the scheme DRA. This allocation utilizes all the physical channels at the shared timeslot. Apart from the theoretical analysis, the authors also performed testbed experiments in FIT IoT-LAB. Their obtained results showed that TACTILE outperforms the existing schemes such as 6TiSCH-MC, DRA, and C2DBI in terms of joining time and energy consumption of nodes during their joining process. It is because TACTILE increases the number of shared cells per slotframe like the scheme DRA. However, the nodes do not consume more energy using this scheme because all the shared cells are not allocated in the different timeslots; rather, all are allocated at `timeslot=0`.

3.5 Lessons Learned from 6TiSCH Network Formation

Most of the existing works related to 6TiSCH network formation follow the same resource allocation described by the 6TiSCH-MC standard [49] i.e., one shared cell per slotframe for transmitting all the generated control packets. The schemes such as BS [84], C2DBI [87], OTCP [89], ACB [90], GTCC [94] improve the performance of 6TiSCH network formation compared to the 6TiSCH-MC. These schemes basically change the transmission rate of either EB frame or DIO packet to reduce the congestion in the shared cell, and so improve the nodes' joining time. So, both EB and DIO transmission rates play significant role in congestion in the shared cell, and so 6TiSCH network formation. However, the above-mentioned works could not able achieve significant performance improvement compared to the works in DRA [82] and TACTILE [83]. The reason for this performance issue is congestion in the shared cell. Although, the works BS [84], C2DBI [87], ACB [90] and, GTCC [94] reduce congestion in minimal cell but that is not sufficient to achieve significant improvement in network formation. Therefore, from here, it can be inferred that the resource provided by the 6TiSCH-MC is not enough to transmit all the generated packets quickly. The results achieved by the schemes such as DRA [82] and TACTILE [83] is significantly better compared to the other schemes i.e., BS [84], C2DBI [87], OTCP [89], ACB [90], and GTCC [94]. The reason for showing this significant improvement by DRA and TACTILE is nothing but providing sufficient shared cells per slotframe to transmit all the generated control packets. Although DRA consumes more energy and hampers the data communication schedule, TACTILE overcomes these two problems by autonomously allocating the shared cells at the same timeslot and then scheduling them properly for maintaining synchronization between the parent-child pairs. Hence, by efficiently adding more number of shared cells per slotframe can improve the performance of 6TiSCH network during its formation compared to having only one shared cell per slotframe. Comparison among the existing schemes on 6TiSCH network formation is shown in Table 3 considering various parameters that are used for designing the existing schemes related to 6TiSCH network formation.

Table 4. List of Frequently used Symbols

Symbol	Meaning
L	Slotframe length in timeslots
N_c	Number of used channels
I_{eb}	EB frame generation interval
P_{eb}	EB transmission probability in a shared cell
P_{EB_S}	Successful EB transmission probability in a shared cell
P_{dio}	DIO transmission probability in a shared cell
P_{DIO_S}	Successful DIO transmission probability in a shared cell
P_{jrq}	JRQ transmission probability in a shared cell
P_{JRQ_S}	Successful JRQ transmission probability in a shared cell
P_{jrs}	JRS transmission probability in a shared cell
P_{JRS_S}	Successful JRS transmission probability in a shared cell
P_{loss}	Packet loss probability
A, B, C, D	Binary variables
ASF	Average number of slotframes
AJT	Average joining time

4 THEORETICAL ANALYSIS OF 6TISCH NETWORK FORMATION

4.1 Theoretical Modeling

This section provides a generalized analysis of 6TiSCH network formation considering the transmission of EB, JRQ, JRS, and DIO happen in minimal/shared cell. At the outset, the variables used in the analytical model are mentioned in Table 4.

In Section 2.4, a brief discussion about the four states of a pledge during its network joining process is provided and Figure 3 shows these states as a Markov state model. Now, the probability of moving from one state to another state depends on the control packet receiving probability by the pledge in its previous state. For example, moving from the first state (i.e., pledge) to second state (i.e., TSCH synchronized node) depends on the successful EB transmission probability by the joined nodes i.e., P_{EB_S}. Similarly, successfully exchanging the JRQ and JRS frames (i.e., P_{JRQ_S}, and P_{JRS_S}) denoted together by P_{join}, and successful DIO transmission probability P_{DIO_S}, help a pledge to move from state 2 to state 3, and from state 3 to the final absorbing state, respectively. Now, in order to calculate these probabilities i.e., P_{EB_S}, P_{JRQ_S}, P_{JRS_S}, and P_{DIO_S}, initially, the EB and DIO transmission probabilities in a shared cell (i.e., P_{eb} and P_{dio}) need to be calculated. Now, according to 6TiSCH-MC [49] and BS [84] schemes, all the nodes use fixed EB generation intervals, whereas C2DBI [87] proposed to vary the beacon interval. Therefore, P^k_{eb} of a joined node (i) can be calculated as follows:

$$P^i_{eb} = L/I^k_{eb,} \tag{2}$$

where L denotes the slotframe length and I^k_{eb} denotes the EB generation interval of node (k). However, the calculation of P_{dio} is not trivial as P_{eb} because the DIO generation interval varies with different network conditions. DIO generation and transmission are mainly governed by the Trickle algorithm [85]. The work in [82] briefly discussed the procedure to calculate P_{dio} considering the 6TiSCH network. Therefore, we use the same procedure to calculate P_{dio} and the final equation to calculate P_{dio} is as follows:

$$P_{dio} = (1 - P_{eb}) \frac{2^{N_D}(1-P_r)^{N_D} min(\frac{L}{2^{N_D} D_{min}}, 1) + \sum_{i=0}^{N_D-1} P_r 2^i (1-P_r)^i min(\frac{L}{2^i D_{min}}, 1)}{P_r + 2^{N_D}(1-P_r)^{N_D} + \sum_{j=1}^{N_D-1} P_r 2^j (1-P_r)^j}, \tag{3}$$

where D_{min}, P_r and N_D denote the minimum DIO generation interval, Trickle reset probability, and the number of trickle states, respectively. Now, using the values of P_{eb} and P_{dio}, the successful EB transmission probability by a joined node, P_{EB_S} can be calculated as follows:

$$P_{EB_S} = \frac{K}{N_c} \sum_{n}^{M-n} n \left[P_{eb}^{i} \prod_{k=1, k\neq i}^{\alpha} \left(1 - P_{eb}^{k}\right) \right]^{A} [(1 - P_{dio})^{\alpha}]^{B} [(1 - P_{jrs})^{\alpha}]^{C} [(1 - P_{jrq})^{\beta}]^{D} \times (1 - P_{loss}) P(N = n), \tag{4}$$

where, P_{eb}^{k} denotes the EB transmission probability of node (k), M and n denote the total number of nodes, and joined nodes, respectively. Both M and n vary from node to node. The pledges $(M - n)$ join the network one after another in the given network. Therefore, $P(N = n)$ denotes the probability of a number of joined nodes, n at any given time, which is calculated using a uniform probability distribution i.e., $1/M$. In short, Equation (4) can be described as follows: a joined node can successfully transmit its EB frame only when the other nodes do not transmit any kind of control packets. In TACTILE, control packets are broadcasted using all the available channels at the same time. So, the probability collision of the control packet gets reduced by the number of transmitting nodes divided by the number of channels, N_c i.e., $\alpha = \lfloor n/N_c \rfloor$ for TACTILE, for others $\alpha = n$; similarly, $\beta = \lfloor M - n/N_c \rfloor$ for TACTILE, for others $\beta = (M - n)$. P_{jrq} denotes the JRQ transmission probability by a pledge, therefore, $P_{jrq} = P_{EB_S}$ because a pledge transmits its JRQ only after receiving an EB frame. Similarly, $P_{jrs} = P_{JRQ_S}$, where P_{JRQ_S} is the successful JRQ transmission probability (which is calculated below) and P_{jrs} denotes the JRS transmission probability in a slotframe. Note that at the beginning, the values of P_{jrs} and P_{jrq} are considered as 0, otherwise, these would form a cycle among the equations. Now, A, B, C, and D are taken as binary variables (values are either 0 or 1) to designate whether EB, JRQ, JRS, and JRQ are transmitted (i.e., value is 1) or not transmitted (i.e., value is 0) together in a shared cell. Because in TACTILE, JRQ is not transmitted in the same minimal cell where EB, JRS, and DIO are transmitted. Initially, a pledge is not aware of the channel where the control packets are broadcasted by the joined nodes, therefore, the whole probability is divided by the N_c. However, when a scheme uses more than one channel (e.g., TACTILE), the entire probability is increased by the number of joined node times i.e., $K = n$, otherwise $K = 1$. Likewise, the other successful packet transmission probabilities such as P_{DIO_S}, P_{JRQ_S}, and P_{JRS_S} are calculated as follows:

$$P_{DIO_S} = \sum_{n}^{M-n} n[P_{dio}(1 - P_{dio})^{\alpha}]^{B} [(1 - P_{jrs})^{\alpha}]^{C} \left[P_{eb}^{i} \prod_{k=1, k\neq i}^{\alpha} \left(1 - P_{eb}^{k}\right) \right]^{A} [(1 - P_{jrq})^{\beta}]^{D} \times (1 - P_{loss}) P(N = n), \tag{5}$$

$$P_{JRQ_S} = \sum_{n}^{M-n} (M - n)[P_{jrq}(1 - P_{jrq})^{\beta}]^{C} \left[P_{eb}^{i} \prod_{k=1, k\neq i}^{\alpha} \left(1 - P_{eb}^{k}\right) \right]^{A} [(1 - P_{dio})^{\alpha}]^{B} \times [(1 - P_{jrs})^{\alpha}]^{C} (1 - P_{loss}) P(N = n), \tag{6}$$

$$P_{JRS_S} = \sum_{n}^{M-n} n[P_{jrs}(1 - P_{jrs})^{\alpha}]^{C} \left[P_{eb}^{i} \prod_{k=1, k\neq i}^{\alpha} (1 - P_{eb}^{k}) \right]^{A} [(1 - P_{dio})^{\alpha}]^{B} [(1 - P_{jrq})^{\beta}]^{C} \times (1 - P_{loss}) P(N = n). \tag{7}$$

Note that in the above equations, N_c is not used in the denominator because a TSCH synchronized node knows the channel where the EB would be broadcasted by the joined nodes. Now, using these probabilities i.e., P_{EB_S}, P_{JRQ_S}, P_{JRS_S}, and P_{DIO_S}, the ***average number of slotframes*** (**ASF**) required by a pledge to reach the final absorbing state can be computed as follows:

$$ASF = \frac{1}{P_{EB_S}} + \left(\frac{1}{P_{JRS_S}} + \frac{1}{P_{JRQ_S}}\right) + \frac{1}{P_{DIO_S}}. \tag{8}$$

If we consider a multi-hop network where each pledge has a different number of neighbors and in different hop distances from the JRC, then the final ***average joining time*** (**AJT**) of a pledge can be calculated as follows:

$$AJT = PJT + ASF \times L, \tag{9}$$

where PJT is the average joining time of the pledge's parent in the multi-hop network. The energy consumption by the pledges and joined nodes are briefly explained in the work in [87] (Section 4.2), which is directly related to the model discussed above. So, the same analytical model can be used to calculate the energy consumption of the pledges and joined nodes during the formation of the 6TiSCH network. Thus, the joining time and energy consumption of any pledge can be determined also considering its parent's joining time and the number of neighbors. Please note that due to dynamic behavior of schemes such as DRA, OTCP, and ACB, we do not provide a theoretical analysis of these schemes. Moreover, the authors of these works did not provide their theoretical analysis.

4.2 Performance Metrics

Three performance metrics have been considered for the evaluation of the existing schemes. As the energy consumption of the pledges is directly proportional to their channel scanning time, therefore, ***TSCH synchronization time*** of the pledges is considered as the first performance metric. The TSCH synchronization of a pledge is the time when it receives its first EB frame from the time when it has started scanning for EB frame. In brief, it is the time when a pledge reaches the second state of the state diagram shown in Figure 3. A TSCH synchronized node consumes less energy compared to a pledge because the TSCH synchronized node keeps its radio active only in the shared timeslot, unlike the pledges which keep their radios active all the time.

A TSCH synchronized node is not allowed to transmit any control packet for further expansion of the network. A node can transmit EB and DIO control packets only after completely joining the 6TiSCH network. Therefore, the ***6TiSCH joining time*** of a node is considered as our second evaluation metric. The 6TiSCH joining time of a node is the time when it receives both EB and DIO packets i.e., when a pledge reaches the final state of the state diagram shown in Figure 3. Note that DODAG joining time and 6TiSCH joining time is the same for a TSCH synchronized node. A TSCH synchronized node requires a DIO packet to join the DODAG constructed by RPL and the same DIO is used to become a 6TiSCH joined node.

The LLN nodes are constrained in terms of processing capacity, memory, and energy. Therefore, the energy consumption of a pledge during the formation of 6TiSCH networks or after their formation is very much important as it is related to the overall network lifetime. Therefore, the ***energy consumption*** of the pledges and joined nodes during the formation of 6TiSCH network is considered as our third evaluation metric.

4.3 Analytical Results and Discussion

To obtain different analytical results such as TSCH and 6TiSCH joining time of the pledges and energy consumption of the joined nodes and pledges using the above analytical model, we consider a

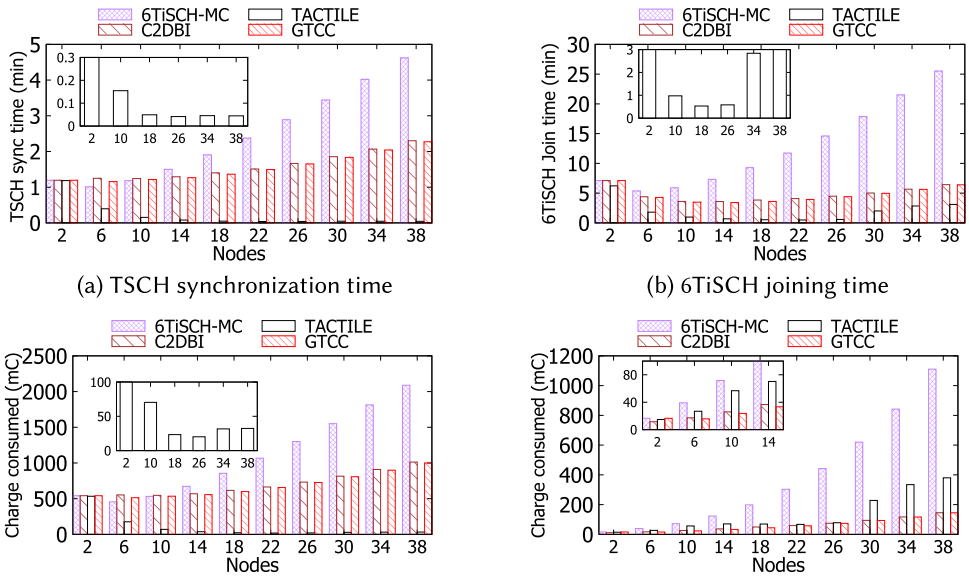


Fig. 7. Comparison-based Analytical Results.

random network with different parameter values as follows: $N_D = 16$, $N_C = 16$, $L = 101$ *timeslots*, $P_{loss} = 0.2$, $I_{eb} = 4 * L$, $D_{min} = 8$ *ms*, $T_i = 10$ *ms*, and $P_r = 0.2$. We follow the work in [87] to calculate the energy consumption of the joined nodes and pledges considering `GINA mote` [95]. A `GINA mote` consumes 69.6, and 72.1 micro coulombs (μC) charge to transmit and receive a packet, respectively. Figure 7(a) and (b) show the average TSCH synchronization time and 6TiSCH joining time of a pledge with respect to varied network size. From the obtained results, it can be seen that both the joining times significantly increase with the increasing number of nodes in the case of 6TiSCH-MC. The reason for this significant performance degradation is the increasing congestion in the shared cell with an increasing number of nodes. When the shared cell severely gets congested, the joined node cannot able to transmit their control packets quickly, and if they do, there will be a high chance of control packet collision. Although C2DBI [87] and GTCC [94] reduce the congestion in the shared cell, that is not sufficient to achieve significant improvement during 6TiSCH network formation. Both C2DBI and GTCC use only shared cell per slotframe like 6TiSCH-MC, and so congestion becomes an inevitable problem for these schemes also. Therefore, increasing the number of shared cells per slotframe is the only way to achieve significant performance improvement of 6TiSCH networks. The work TACTILE [83] increased the number of shared cells per slotframe by intelligently allocating and scheduling the shared cell at the same timeslot. Hence, it outperforms the above-mentioned schemes in terms of both TSCH and 6TiSCH joining times. Furthermore, TACTILE allows the nodes to transmit their EB frames using different physical channels at the same time which significantly increases the EB reception probabilities of the pledges. Therefore, the TACTILE shows significant performance improvement compared to the other schemes.

The average energy consumption of a pledge and joined node are shown in Figure 7(c) and (d), respectively. As the energy consumption of the pledge is directly related to the both the joining times, specially, TSCH synchronization time, therefore, using the 6TiSCH-MC, nodes consume more energy compared to the other schemes. The pledges need to keep their radios active for more time to get an EB frame, which causes more energy consumption in 6TiSCH-MC. On the

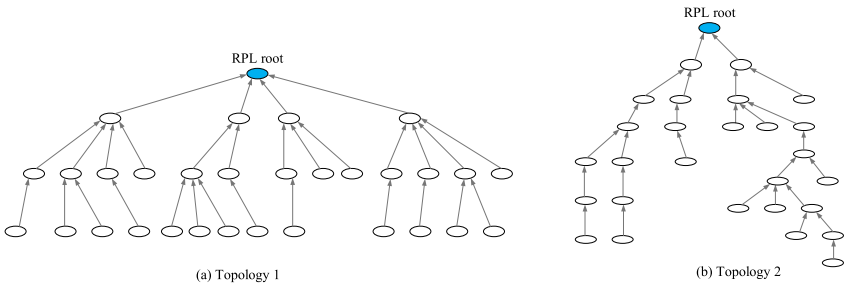


Fig. 8. The tree topologies are used in testbed experiments.

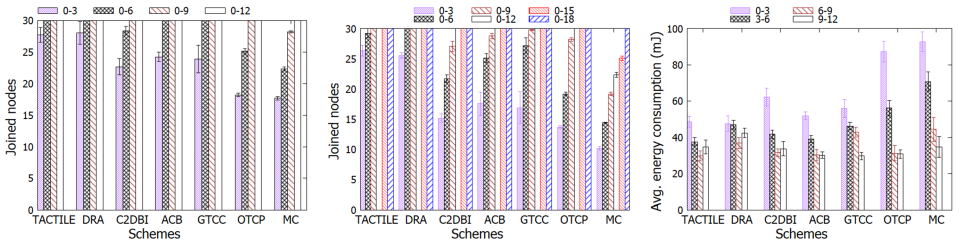
other hand, using TACTILE, pledges do not need to keep their radios active for more time, and so their average energy consumption is very less compared to the other schemes. However, the energy consumption of the joined node is more in TACTILE compared to the other schemes. It is because, using TACTILE, joined nodes get more opportunities to transmit their control packets, and so they transmit, which turns out in more energy consumption.

5 EXPERIMENTAL ANALYSIS

In this section, the existing works on 6TiSCH network formation are analyzed using testbed experiments at FIT IoT-LAB [45]. For testbed experiments, we use the topologies as shown in Figure 8. Note that testbed experimental analysis is done only for the works which considered 6TiSCH network formation i.e., consider both EB and DIO packet for the formation of 6TiSCH network. It is because the works related to TSCH network synchronization are not 6TiSCH-MC compliant, and almost all the schemes would encounter the de-synchronization issue between the parent-child pairs. Hence, the schemes related to TSCH network synchronization can never be able to completely form multi-hop 6TiSCH networks. For the testbed evaluation, in the beginning, the existing works (i.e., 6TiSCH-MC, DRA, C2DBI, ACB, GTCC, OTCP, and TACTILE) have been implemented on the Contiki-NG OS [44]. The binary files generated (either .elf, .bin, or .hex) after successfully compiling the schemes on Contiki-NG have been flashed on the STM32 (ARM Cortex M3) micro-controller based M3 nodes on FIT IoT-LAB. Two different topologies have been used for our testbed experiments to show the results with varied node density and hop distance. In Topology 1, the nodes are deployed densely compared to Topology 2, whereas the maximum hop length is more in Topology 2 compared to Topology 1.

5.1 Results and Discussion

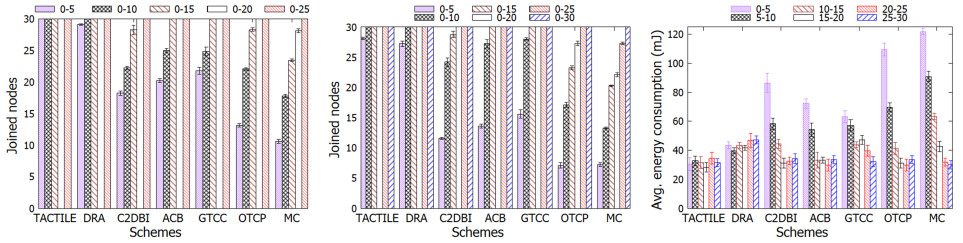
The metrics used for analytical analysis in Section 4.2 are used to show the testbed experimental results. The results are shown using a 95% confidence interval. The metrics are measured under different time intervals like 0–3, 0–9, 0–15 minutes. For example, a total number of pledges that become joined nodes within 3 min after the starting of the network is shown in 0–3 min intervals. Similarly, the total number of pledges joined within 6 min after the starting of the network is shown in the 0–6 min interval. Please note that we activate (i.e., switch on) all nodes at the same time during our experiments, but they join the network in different times. Figure 9(a)–(e) show the number of TSCH joined/synchronized nodes and 6TiSCH joined nodes in various time intervals using Topologies 1 and 2, respectively. The obtained testbed results reveal that TACTILE and DRA surpass all existing schemes in terms of both TSCH synchronization time and 6TiSCH joining time, including C2DBI, ACB, GTCC, OTCP, and 6TiSCH-MC. Please note that we mention 6TiSCH-MC as MC in Figure 9. It is because both the schemes provide enough number of shared cells per



(a) TSCH joined nodes using Topology 1

(b) 6TiSCH joined nodes using Topology 1

(c) Average energy consumption using Topology 1



(d) TSCH joined nodes using Topology 2

(e) 6TiSCH joined nodes using Topology 2

(f) Average energy consumption using Topology 2

Fig. 9. Different Testbed results are shown under various time intervals (e.g., 0–3, 0–9, 0–15). The measuring units of time intervals are minute.

slotframe for transmitting all the generated control packets. From here, we can conclude that a single shared cell is not enough for forming the 6TiSCH networks quickly. Rather, more number of shared cells per slotframe are needed for faster formation. Although the rest of the schemes i.e., C2DBI, ACB, GTCC, OTCP, and 6TiSCH-MC use the same resource during the formation of 6TiSCH networks, the performance of 6TiSCH-MC is the worst among all the schemes. It is because 6TiSCH-MC does not provide any mechanism to deal with congestion in a shared cells and handle the transmission of different control packets depending on the present network condition.

Basically, C2DBI reduces the congestion in the shared cells by varying the EB transmission interval depending on network conditions. In contrast, OTCP provides enough routing information i.e., DIO packet during 6TiSCH network formation by changing the priority of the control packets depending on network conditions and enabling quick transmission of DIO packets. Hence, OTCP even performs less than C2DBI as congestion in the shared cell is the major issue during the formation of 6TiSCH networks. On the other hand, ACB considered efficient transmission of EB and DIO packet considering congestion in shared cell. Hence, ACB performs better than the C2DBI and OTCP. Similary, GTCC also considered the transmission rate of both EB and DIO packets on shared cell congestion, so it shows almost similar results as ACB.

As the energy consumption of the pledges is directly associated with their TSCH synchronization time and overall formation of the network, the energy consumption of TACTILE is very much less compared to the other schemes as shown in Figure 9(c) and (f) for both the topologies. Although in the beginning, DRA consumes less energy, after completely forming the network, it consumes a little bit more energy compared to the other schemes. It is because all the nodes need to active their radios more amount time per slotframe (i.e., in all the allocated shared cells per slotframe) compared to the other schemes where nodes need to active their radios only in a single shared cell per slotframe. TACTILE overcomes this issue of DRA by allocating and scheduling all

the shared cells in a single timeslot. Hence, a node needs to active its radio only in the shared cell, and so, it has similar energy consumption as the other schemes except DRA. However, nodes consume more energy throughout the formation of the networks using the 6TiSCH-MC scheme as they need to wait for more time to receive the EB frame, which forces them to continuously active their radios for a longer duration.

The results in Figure 9(c) and (f) show that 6TiSCH-MC consumes more energy compared to all the other schemes. For example, in our experiment using topology 2, it is observed that TACTILE saves approximately (380.4 − 188.89) = 191.51 mJ (i.e., approximately 51.34%) compared to 6TiSCH-MC. Using this 191.51 mJ, a GINA mote can transmit approximately 1528 packets. Furthermore, during the full life cycle of the networks, sometimes, many nodes need to re-join the networks as fresh pledges several times due to different network issues as mentioned in Section 2.4. Therefore, it is very important to considered the energy consumption of the nodes during 6TiSCH network formation, which also be used for individual node to re-join the network.

6 RESEARCH CHALLENGES AND OPEN ISSUES IN 6TISCH NETWORK FORMATION

6.1 Factors Affecting in 6TiSCH Network Formation

There is a number of factors that affect the formation of 6TiSCH networks. These factors are discussed briefly in the below subsections.

6.1.1 Number of used Communication Channels. One of the main reasons for longer formation of the 6TiSCH network is the channel hopping feature of TSCH MAC protocol. At the beginning, a pledge does not know the time and the channel in which joined nodes transmit their control packets. Hence, a pledge scans for an EB on a random channel. Now, the probability of getting an EB on that random channel is inversely proportional to the number of total communication channels used by the networks. Although IEEE 802.15.4e allows maximum of 16 channels within the 2.4 *GHz* frequency band, some networks may consider less than 16 channels for their communications. In that case, the probability of receiving an EB by the pledge is high compared to the networks which use more communication channels. The overall 6TiSCH network formation time reduces when the pledges receive their EB frames quickly. Thus, the number of communication channels used in the network plays a significant role in the TSCH synchronization time of the pledges, and so their energy consumption and overall formation time of 6TiSCH networks.

6.1.2 Number of Shared Cells Per Slotframe. Including EB and DIO, all types of control packets need to be transmitted in shared cell in 6TiSCH networks. Therefore, congestion in the allocated shared cell(s) increases when more nodes join the network. This congestion in the shared cell significantly affects the formation time of the 6TiSCH networks when the number of allocated shared cell is less and static. It is because the pledges (or joined nodes) need to wait for a longer time to receive (or transmit) their control packets due to packet collision or longer shared cell access time. So, more number of shared cells per slotframe help the joined nodes to transmit their control packets quickly, and so improve the formation time. However, the allocation of more shared cells in a slotframe reduces the number of dedicated cells for data packet transmission. This can affect the performance of the networks in terms of throughput, end-to-end latency, and also can affect the existing scheduling for data packet transmission. Furthermore, the radio duty cycles of the nodes increase with the increasing number of shared cells per slotframe. Therefore, the number of allocated shared cells per slotframe plays a significant role in 6TiSCH network formation, and nodes' energy consumption.

6.1.3 Node Density. As the number of shared cells per slotframe is limited as per the 6TiSCH-MC standard, therefore, the density of the nodes in a 6TiSCH network significantly affects its

formation. Even in the same network, the congestion in the shared cell is more where nodes are densely deployed compared to the area where nodes are sparsely deployed. As it is already mentioned that congestion plays an important role in the performance of the 6TiSCH network formation, hence, node density also plays a significant role in the performance of the 6TiSCH network formation.

6.1.4 Network Topology. The position of the nodes in their deployment area also affects the formation of the networks. For example, the nodes present in a grid topology with an average node's degree will take less time to form the network compared to the networks where a same number of nodes in a linear topology with a single node degree. Here, node degree denotes the number of neighboring nodes. It is because in linear topology each node needs to wait for its parent node to finish its joining time. Furthermore, in the linear topology, the probability of receiving an EB by a pledge is very less as only one joined node is present in its communication range. On the other hand, in the grid topology, nodes have more opportunities to receive control packets from different nodes which increases the control packet receiving probabilities of the nodes. Thus, the topology of a network also affects the formation of 6TiSCH networks.

6.1.5 Default Transmission Rate of EB Frame. Except the works in [84, 87], the other works considered the static rate of the EB frame. As, initially, a pledge does not know the channel where the EB frame is being broadcasted by the joined nodes, therefore, the pledge performs random channel scanning. Now, if the EB rate is high, then the chances of receiving an EB frame gets increased. However, when the transmission rate of the EB frame crosses a certain limits, then the shared cell starts getting congested, results in a higher joining time of the pledge. Furthermore, the energy consumption of joined nodes increases with the increasing transmission rate of the EB frame. Therefore, there should be a tradeoff between the energy consumption of joined nodes and pledges joining time during the formation of 6TiSCH networks.

6.1.6 Default Transmission Rate of DIO Packet. The generation and transmission of the DIO packet in the 6TiSCH network is governed by the Trickle algorithm [85]. The trickle algorithm varies the DIO generation and transmission rate depending on the present network condition, unlike the fixed rate of the EB frame. However, Trickle algorithm mainly depends on a few parameters such as *DIO minimum interval*, *redundancy constant*, and a *number of doubling*, which are set by the network administrator at the beginning and remain unchanged throughout the network lifetime. Therefore, in some situations such as DODAG reset time, inconsistency in the network, transmission of multicast DIS request, the joined nodes transmit several DIO packets within a small span of time if these parameters are not chosen carefully. This rapid transmission of DIO packets can severely congest the shared cell, and so can significantly increase the formation time of 6TiSCH networks. Hence, the default transmission rate of the DIO packet has a significant affect on the 6TiSCH network formation. Furthermore, the challenges associated with the RPL routing layer [96] need to be considered during the formation of 6TiSCH networks.

6.1.7 Node Mobility. None of the existing works considered the presence of mobile nodes during 6TiSCH network formation. Mobile nodes can be expected in some IoT applications [2, 21]. A pledge chooses its parent based on the values mentioned in the `join metric` field of received EB frames. However, the `join metric` field does not contain any information about the mobility of the nodes. Hence, the pledge might choose a mobile node as its parent by considering better value of `join metric`. In such a situation, the pledge again needs to join the network as a fresh pledge when the currently associated parent moves away from the communication range of the pledge. And this can increase the formation time of entire network. Thus, the mobility of the nodes also impacts the 6TiSCH network formation.

6.2 Open Research Issues in 6TiSCH Network Formation

In this section, we highlight some open research issues on 6TiSCH network formation as follows:

6.2.1 Formation of 6TiSCH Networks using Efficient EB Scanning. Although the IEEE 802.15.4e standard mentioned active scanning of EB frame by which the pledges can transmit EB request frame, only a few works i.e., [78–80] considered it. However, as it is already mentioned, these schemes (i.e., [78–80]) have several drawbacks such as active scanning increases the energy consumption of the nodes and packet collision, hampers the TSCH schedule, and creates de-synchronization between parent and child. Furthermore, none of the existing works considered the formation of multi-hop 6TiSCH network i.e., the transmission of DIO packet for constructing the DODAG topology during the formation of 6TiSCH network. Therefore, the researchers can put an effort towards designing efficient active EB scanning process for faster formation of multi-hop 6TiSCH network.

6.2.2 Impact of Clock Drift on Nodes' Re-association. As the nodes use timeslot to communicate with their neighboring nodes for both control and data packet transmission in IEEE 802.15.4e TSCH-based networks, therefore, both the transmitter and receiver's clocks should be aligned to each other [97, 98]. In brief, both the transmitter and receiver clocks should be synchronized within 1ms. However, due to various environmental factors such as fluctuation in temperature as well as supplied voltage, the difference in mote fabrication by different vendors, and both the transmitter-receiver clocks used to drift from each other. Although this drifting happens in terms of microseconds each time, both the transmitter and receiver gets de-synchronized when they are drifted by more than 1ms. And because of this de-synchronization, the child joined node(s) might need to rejoin the network as a fresh pledge. Although various schemes have already been proposed to deal with this clock drifting issue, researchers still can consider clock drifting in various environmental conditions and propose a solution for it. Furthermore, only few research works had been published, which studied the rate of drifting in the presence of multiple timesource. Therefore, clock drifting is still an open and important research problem because the timeslot shared by a parent-child (transmitter-receiver) nodes needs to be closely aligned to receive the transmitted data correctly in the 6TiSCH network. Therefore, clock drifting is a vital research issue that can be explored in more detail from a network formation perspective.

6.2.3 Formation of 6TiSCH Networks in Presence of Malicious Node. The works in [99, 100] studied the 6TiSCH network formation process in the presence of malicious nodes. The authors in [99] mentioned from their simulation experiments that the formation time of the entire network drastically increases in presence of non-cooperative malicious nodes. However, the authors did their experiments considering only the 6P transaction [52], which is used for scheduling the data packet transmission cell in a distributed manner. However, the formation of the 6TiSCH network can be studied considering other security threats which a malicious node can cause, such as jamming the commonly shared cell by cracking the channel hopping sequence, DIS attack [100] or DIS Sybil attack [101], DIO suppression attack, DIO black hole attack, RPL rank attack, and so on. As both the congestion in shared cells and transmission of all types of control packets play a critical role in forming the of multi-hop 6TiSCH networks, therefore, it is very much necessary to investigate the performance of 6TiSCH network formation in the presence of malicious nodes. Furthermore, the other security threads to IoT as mentioned in [102] need to be investigated during the formation of 6TiSCH networks.

6.2.4 Formation of 6TiSCH Networks in Presence of Mobile Node. The IEEE 802.15.4e TSCH based 6TiSCH network is mainly designed for industrial applications. Although some industrial

applications do not have mobile nodes, some industries, such as the petroleum and oil industry, actuation-based industrial monitoring, and waste management, mobile nodes are used. It is noteworthy that none of the published works considered the formation of the 6TiSCH network in the presence of mobile nodes. It is essential because when a joined node moves out of the communication range of the child node(s), all the child node(s) need to again rejoin the network as a pledge. This can severely impact on the overall formation of the network. Therefore, it is another research scope where researchers can think about the formation of the 6TiSCH network in the presence of a mobile node.

6.2.5 Formation of Software-defined Network (SDN) based 6TiSCH Networks. Nowadays, researchers are trying to incorporate **Software-defined Network (SDN)** in LLNs. However, only a few works had been published, integrating SDN in 6TiSCH networks to schedule the communication cell [103, 104]. Implementing a centralized SDN architecture in LLNs will face considerable challenges such as control traffic generated by SDN subject to jitters due to unreliable wireless links and contention in the communication cell, which can further hinder the performance of SDN-based network. Furthermore, the SDN control traffic needs to be transmitted in the communication cell. Therefore, how the control traffic overhead generated by SDN can be transmitted along with the existing 6TiSCH control traffic during the formation of SDN-based 6TiSCH networks needs to be adequately investigated. The resource allocation by 6TiSCH-MC is not sufficient for 6TiSCH control traffic as per the existing works. In this case, the transmission of SDN control traffic along with the existing 6TiSCH network traffic will be some extra burden on the shared cell. Hence, this will surely degrade the performance of the networks during their formations.

7 CONCLUSION

The IEEE 802.15.4e amendment includes TSCH MAC mode to support multi-channel, channel hopping, and time-slotted/time-division channel access, which provide deterministic and delay bounded packet delivery to the mission-critical IoT applications. The formation of the TSCH MAC-based 6TiSCH network is not trivial because of the usage of multiple channels, channel hopping feature, and minimal resource allocation for control packet transmission. Therefore, different researchers proposed many improvements to reduce the network joining time of the pledges so that the network's lifetime can be increased and data packets can be transmitted efficiently. This article surveys the existing works related to the formation of both TSCH and 6TiSCH networks, considering almost all the works published in 2014–2021. Apart from the survey, this article also provides both theoretical analysis and testbed evaluation of most of the schemes related to 6TiSCH network formation. Taking into account both theoretical and experimental results, this article ends with a brief summarization of the factors that affect the 6TiSCH network formation and also provides a few open research issues in 6TiSCH network formation.

REFERENCES

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. 2010. The internet of things: A survey. *Computer Networks* 54, 15 (2010), 2787–2805.

[2] Li Da Xu, Wu He, and Shancang Li. 2014. Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics* 10, 4 (2014), 2233–2243. http://dx.doi.org/10.1109/TII.2014.2300753

[3] John A. Stankovic. 2014. Research directions for the internet of things. *IEEE Internet of Things Journal* 1, 1 (2014), 3–9. http://dx.doi.org/10.1109/JIOT.2014.2312291

[4] Thomas Watteyne, Vlado Handziski, Xavier Vilajosana, Simon Duquennoy, Oliver Hahm, Emmanuel Baccelli, and Adam Wolisz. 2016. Industrial wireless ip-based cyber -physical systems. *Proceedings of the IEEE* 104, 5 (2016), 1025–1038. http://dx.doi.org/10.1109/JPROC.2015.2509186

[5] Kun Wang, Yihui Wang, Yanfei Sun, Song Guo, and Jinsong Wu. 2016. Green industrial internet of things architecture: An energy-efficient perspective. *IEEE Communications Magazine* 54, 12 (2016), 48–54. http://dx.doi.org/10.1109/MCOM.2016.1600399CM

[6] Stephanie B. Baker, Wei Xiang, and Ian Atkinson. 2017. Internet of things for smart healthcare: Technologies, challenges, and opportunities. *IEEE Access* 5 (2017), 26521–26544. http://dx.doi.org/10.1109/ACCESS.2017.2775180

[7] Adam B. Noel, Abderrazak Abdaoui, Tarek Elfouly, Mohamed Hossam Ahmed, Ahmed Badawy, and Mohamed S. Shehata. 2017. Structural health monitoring using wireless sensor networks: A comprehensive survey. *IEEE Communications Surveys Tutorials* 19, 3 (2017), 1403–1423. http://dx.doi.org/10.1109/COMST.2017.2691551

[8] Mussab Alaa, A. A. Zaidan, B. B. Zaidan, Mohammed Talal, and M. L. M. Kiah. 2017. A review of smart home applications based on internet of things. *Journal of Network and Computer Applications* 97 (2017), 48–65. http://dx.doi.org/10.1016/j.jnca.2017.08.017

[9] Eunil Park, Yongwoo Cho, Jinyoung Han, and Sang Jib Kwon. 2017. Comprehensive approaches to user acceptance of internet of things in a smart home environment. *IEEE Internet of Things Journal* 4, 6 (2017), 2342–2350. http://dx.doi.org/10.1109/JIOT.2017.2750765

[10] Saraju P. Mohanty, Uma Choppali, and Elias Kougianos. 2016. Everything you wanted to know about smart cities: The internet of things is the backbone. *IEEE Consumer Electronics Magazine* 5, 3 (2016), 60–70. http://dx.doi.org/10.1109/MCE.2016.2556879

[11] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. 2014. Internet of things for smart cities. *IEEE Internet of Things Journal* 1, 1 (2014), 22–32. http://dx.doi.org/10.1109/JIOT.2014.2306328

[12] Yuke Li, Xiang Cheng, Yang Cao, Dexin Wang, and Liuqing Yang. 2018. Smart choice for the smart grid: Narrowband internet of things (NB-IoT). *IEEE Internet of Things Journal* 5, 3 (2018), 1505–1515. http://dx.doi.org/10.1109/JIOT.2017.2781251

[13] Miao Yun and Bu Yuxin. 2010. Research on the architecture and key technology of internet of things (IoT) applied on smart grid. In *Proceedings of the 2010 International Conference on Advances in Energy Engineering*. 69–72. http://dx.doi.org/10.1109/ICAEE.2010.5557611

[14] Shanzhi Chen, Hui Xu, Dake Liu, Bo Hu, and Hucheng Wang. 2014. A vision of IoT: Applications, challenges, and opportunities with china perspective. *IEEE Internet of Things Journal* 1, 4 (2014), 349–359. http://dx.doi.org/10.1109/JIOT.2014.2337336

[15] Nurzaman Ahmed, Debashis De, and Iftekhar Hussain. 2018. Internet of things (IoT) for smart precision agriculture and farming in rural areas. *IEEE Internet of Things Journal* 5, 6 (2018), 4890–4899. http://dx.doi.org/10.1109/JIOT.2018.2879579

[16] Gerd Kortuem, Fahim Kawsar, Vasughi Sundramoorthy, and Daniel Fitton. 2010. Smart objects as building blocks for the internet of things. *IEEE Internet Computing* 14, 1 (2010), 44–51. http://dx.doi.org/10.1109/MIC.2009.143

[17] Mo Sha, Dolvara Gunatilaka, Chengjie Wu, and Chenyang Lu. 2017. Empirical study and enhancements of industrial wireless sensor-actuator network protocols. *IEEE Internet of Things Journal* 4, 3 (2017), 696–704. http://dx.doi.org/10.1109/JIOT.2017.2653362

[18] Ivan Stojmenovic. 2014. Machine-to-machine communications with in-network data aggregation, processing, and actuation for large-scale cyber-physical systems. *IEEE Internet of Things Journal* 1, 2 (2014), 122–128. http://dx.doi.org/10.1109/JIOT.2014.2311693

[19] Maria Rita Palattella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Luigi Alfredo Grieco, Gennaro Boggia, and Mischa Dohler. 2013. Standardized protocol stack for the internet of (important) things. *IEEE Communications Surveys Tutorials* 15, 3 (2013), 1389–1406. http://dx.doi.org/10.1109/SURV.2012.111412.00158

[20] Adnan Aijaz and A. Hamid Aghvami. 2015. Cognitive machine-to-machine communications for internet-of-things: A protocol stack perspective. *IEEE Internet of Things Journal* 2, 2 (2015), 103–112. DOI: http://dx.doi.org/10.1109/JIOT.2015.2390775

[21] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. 2015. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys Tutorials* 17, 4 (2015), 2347–2376. DOI: http://dx.doi.org/10.1109/COMST.2015.2444095

[22] Stanislav Safaric and Kresimir Malaric. 2006. ZigBee wireless standard. In *Proceedings of the ELMAR 2006.* 259–262. DOI: http://dx.doi.org/10.1109/ELMAR.2006.329562

[23] 2006. IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low rate wireless personal area networks (WPANs). *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)* (2006), 1–320. DOI: http://dx.doi.org/10.1109/IEEESTD.2006.232110

[24] 2005. IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 15.1a: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPAN). *IEEE Std 802.15.1-2005 (Revision of IEEE Std 802.15.1-2002)* (2005), 1–700. DOI: http://dx.doi.org/10.1109/IEEESTD.2005.96290

[25] Jianping Song, Song Han, Al Mok, Deji Chen, Mike Lucas, Mark Nixon, and Wally Pratt. 2008. WirelessHART: Applying wireless technology in real-time industrial process control. In *Proceedings of the 2008 IEEE Real-Time and Embedded Technology and Applications Symposium.* 377–386. DOI: http://dx.doi.org/10.1109/RTAS.2008.15

[26] 2011. Wireless systems for industry automation: Process control and related applications. *ISA-100.11a-2011 Standard* (2011), 1–793.

[27] Almudena Díaz Zayas and Pedro Merino. 2017. The 3GPP NB-IoT system architecture for the internet of things. In *Proceedings of the 2017 IEEE International Conference on Communications Workshops*. 277–282. DOI: http://dx.doi.org/10.1109/ICCW.2017.7962670

[28] 2017. IEEE standard for information technology–telecommunications and information exchange between systems - local and metropolitan area networks–specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 2: Sub 1 GHz license exempt operation. *IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as amended by IEEE Std 802.11ai-2016)* (2017), 1–594. DOI: http://dx.doi.org/10.1109/IEEESTD.2017.7920364

[29] Aloÿs Augustin, Jiazi Yi, Thomas Clausen, and William Mark Townsley. 2016. A study of LoRa: Long range & low power networks for the internet of things. *Sensors* 16, 9 (2016), 1–18. DOI: http://dx.doi.org/10.3390/s16091466

[30] Harrison Kurunathan, Ricardo Severino, Anis Koubaa, and Eduardo Tovar. 2018. IEEE 802.15.4e in a nutshell: Survey and performance evaluation. *IEEE Communications Surveys Tutorials* 20, 3 (2018), 1989–2010. DOI: http://dx.doi.org/10.1109/COMST.2018.2800898

[31] Sofie Pollin, Mustafa Ergen, Sinem Coleri Ergen, Bruno Bougard, Liesbet Van Der Perre, Ingrid Moerman, Ahmad Bahai, Pravin Varaiya, and Francky Catthoor. 2008. Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer. *IEEE Transactions on Wireless Communications* 7, 9 (2008), 3359–3371. DOI: http://dx.doi.org/10.1109/TWC.2008.060057

[32] 2012. IEEE standard for local and metropolitan area networks-part 15.4: Low-rate wireless personal area networks (LR-WPANs) Amendment 1: MAC sublayer. *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)* (April 2012), 1–225. DOI: http://dx.doi.org/10.1109/IEEESTD.2012.6185525

[33] Domenico De Guglielmo, Simone Brienza, and Giuseppe Anastasi. 2016. IEEE 802.15.4e: A survey. *Computer Communications* 88 (2016), 1–24. DOI: http://dx.doi.org/10.1016/j.comcom.2016.05.004

[34] 2016. IEEE standard for low-rate wireless networks. *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)* (April 2016), 1–709. DOI: http://dx.doi.org/10.1109/IEEESTD.2016.7460875

[35] Thomas Watteyne, Maria Rita Palattella, and Luigi Alfredo Grieco. 2015. Using IEEE 802.15.4e Time-slotted channel hopping (TSCH) in the Internet of Things (IoT): Problem statement. *Internet Engineering Task Force* RFC 7554. (2015). DOI: http://dx.doi.org/10.17487/RFC7554

[36] Stig Petersen, Paula Doyle, Svein Vatland, Christian Salbu Aasland, Trond Michael Andersen, and Dag Sjong. 2007. Requirements, drivers and analysis of wireless sensor network solutions for the Oil Gas industry. In *Proceedings of the 2007 IEEE Conference on Emerging Technologies and Factory Automation*. 219–226. DOI: http://dx.doi.org/10.1109/EFTA.2007.4416773

[37] Atis Elsts, Xenofon Fafoutis, George Oikonomou, Robert Piechocki, and Ian Craddock. 2020. TSCH networks for health IoT: Design, evaluation, and trials in the wild. *ACM Transactions on Internet of Things* 1, 2 (2020), 27 pages. DOI: http://dx.doi.org/10.1145/3366617

[38] Rajeev Piyare, George Oikonomou, and Atis Elsts. 2020. TSCH for long range low data rate applications. *IEEE Access* 8 (2020), 228754–228766. DOI: http://dx.doi.org/10.1109/ACCESS.2020.3046769

[39] Xavier Vilajosana, Thomas Watteyne, Tengfei Chang, Mališa Vučinić, Simon Duquennoy, and Pascal Thubert. 2020. IETF 6TiSCH: A tutorial. *IEEE Communications Surveys Tutorials* 22, 1 (2020), 595–615. DOI: http://dx.doi.org/10.1109/COMST.2019.2939407

[40] Mališa Vučinić, Jonathan Simon, Kris Pister, and Michael Richardson. 2021. *Constrained Join Protocol (CoJP) for 6TiSCH*. Technical Report 9031. IETF. DOI: http://dx.doi.org/10.17487/RFC9031

[41] Seema Kharb and Anita Singhrova. 2019. A survey on network formation and scheduling algorithms for time slotted channel hopping in industrial networks. *Journal of Network and Computer Applications* 126 (2019), 59–87. DOI: http://dx.doi.org/10.1016/j.jnca.2018.11.004

[42] Atis Elsts, Seohyang Kim, Hyung-Sin Kim, and Chongkwon Kim. 2020. An empirical survey of autonomous scheduling methods for TSCH. *IEEE Access* 8 (2020), 67147–67165. DOI: http://dx.doi.org/10.1109/ACCESS.2020.2980119

[43] Rodrigo Teles Hermeto, Antoine Gallais, and Fabrice Theoleyre. 2017. Scheduling for IEEE802.15.4-TSCH and slow channel hopping MAC in low power industrial wireless networks: A survey. *Computer Communications* 114 (2017), 84–105. DOI: http://dx.doi.org/10.1016/j.comcom.2017.10.004

[44] Adam Dunkels, Bjorn Gronvall, and Thiemo Voigt. 2004. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks* 455–462. DOI: http://dx.doi.org/10.1109/LCN.2004.38

[45] Cedric Adjih, Emmanuel Baccelli, Eric Fleury, Gaetan Harter, Nathalie Mitton, Thomas Noel, Roger Pissard-Gibollet, Frederic Saint-Marcel, Guillaume Schreiner, Julien Vandaele, and Thomas Watteyne. 2015. FIT IoT-LAB: A large scale open experimental IoT testbed. In *Proceedings of the 2015 IEEE 2nd World Forum on Internet of Things*. 459–464. DOI: http://dx.doi.org/10.1109/WF-IoT.2015.7389098

[46] Geoff Mulligan. 2007. The 6LoWPAN architecture. In *Proceedings of the 4th Workshop on Embedded Networked Sensors*. Association for Computing Machinery, New York, NY, 78–82. DOI: http://dx.doi.org/10.1145/1278972.1278992

[47] Carles Gomez, Josep Paradells, Carsten Bormann, and Jon Crowcroft. 2017. From 6LoWPAN to 6Lo: Expanding the universe of IPv6-supported technologies for the internet of things. *IEEE Communications Magazine* 55, 12 (2017), 148–155. DOI: http://dx.doi.org/10.1109/MCOM.2017.1600534

[48] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert. 2014. 6TiSCH: Deterministic IP-enabled industrial internet (of things). *IEEE Communications Magazine* 52, 12 (2014), 36–41. DOI: http://dx.doi.org/10.1109/MCOM.2014.6979984

[49] Xavier Vilajosana, Kris Pister, and Thomas Watteyne. 2017. *Minimal IPv6 over the TSCH Mode of IEEE 802.15.4e (6TiSCH) Configuration*. RFC 8180. IETF. DOI: http://dx.doi.org/10.17487/RFC8180

[50] Michael Richardson. 2019. *6TiSCH Zero-Touch Secure Join Protocol*. draft-ietf-6tisch-dtsecurity-zerotouch-join-04 [work-in-progress]. Internet Engineering Task Force.

[51] Tero Kivinen and Pat Kinney. 2017. *IEEE 802.15.4 Information Element for the*. Technical Report 8137. IETF. DOI: http://dx.doi.org/10.17487/RFC8137

[52] Qin Wang, Xavier Vilajosana, and Thomas Watteyne. 2018. *6TiSCH Operation Sublayer (6top) Protocol (6P)*. RFC 8480. IETF. DOI: http://dx.doi.org/10.17487/RFC8480

[53] Tengfei Chang, Mališa Vučinić, Xavier Vilajosana, Simon Duquennoy, and Diego Roberto Dujovne. 2021. *6TiSCH Minimal Scheduling Function*. Technical Report 9033. IETF. DOI: http://dx.doi.org/10.17487/RFC9033

[54] Simon Duquennoy, Beshr Al Nahas, Olaf Landsiedel, and Thomas Watteyne. 2015. Orchestra: Robust mesh networks through autonomously scheduled TSCH. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. 337–350. DOI: http://dx.doi.org/10.1145/2809695.2809714

[55] Seungbeom Jeong, Hyung-Sin Kim, Jeongyeup Paek, and Saewoong Bahk. 2020. OST: On-demand TSCH scheduling with traffic-awareness. In *Proceedings of the IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*. 69–78. DOI: http://dx.doi.org/10.1109/INFOCOM41043.2020.9155496

[56] Seohyang Kim, Hyung-Sin Kim, and Chongkwon Kim. 2019. ALICE: Autonomous link-based cell scheduling for TSCH. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks* . Association for Computing Machinery, New York, NY, 121–132. DOI: http://dx.doi.org/10.1145/3302506.3310394

[57] Gabriel Montenegro, Jonathan Hui, David Culler, and Nandakishore Kushalnagar. 2007. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. Technical Report 4944. IETF. DOI: http://dx.doi.org/10.17487/RFC4944

[58] Pascal Thubert and Jonathan Hui. 2011. *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*. Technical Report 6282. IETF. DOI: http://dx.doi.org/10.17487/RFC6282

[59] Pascal Thubert and Robert Cragie. 2016. *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch*. Technical Report 8025. IETF. DOI: http://dx.doi.org/10.17487/RFC8025

[60] Pascal Thubert, Carsten Bormann, Laurent Toutain, and Robert Cragie. 2017. *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing Header*. Technical Report 8138. IETF. DOI: http://dx.doi.org/10.17487/RFC8138

[61] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. P. Vasseur, and R. Alexander. 2012. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550. DOI: 10.17487/RFC6550

[62] Pascal Thubert. 2012. *Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks*. Technical Report 6552. IETF. DOI: http://dx.doi.org/10.17487/RFC6552

[63] Zach Shelby, Klaus Hartke, and Carsten Bormann. 2014. *The Constrained Application Protocol* . Technical Report 7252. IETF. DOI: http://dx.doi.org/10.17487/RFC7252

[64] Thomas Watteyne, Xavier Vilajosana, Branko Kerkez, Fabien Chraim, Kevin Weekly, Qin Wang, Steven Glaser, and Kris Pister. 2012. OpenWSN: A standards-based low-power wireless development environment. *Transactions on Emerging Telecommunications Technologies* 23, 5 (2012), 480–493.

[65] Emmanuel Baccelli, Oliver Hahm, Mesut Günes, Matthias Wählisch, and Thomas C. Schmidt. 2013. RIOT OS: Towards an OS for the internet of things. In *Proceedings of the 2013 IEEE Conference on Computer Communications Workshops*. 79–80. DOI: http://dx.doi.org/10.1109/INFCOMW.2013.6970748

[66] Esteban Municio, Glenn Daneels, Mališa Vučinić, Steven Latré, Jeroen Famaey, Yasuyuki Tanaka, Keoma Brun, Kazushi Muraoka, Xavier Vilajosana, and Thomas Watteyne. 2019. Simulating 6TiSCH networks. *Transactions on Emerging Telecommunications Technologies* 30, 3 (2019), e3494.

[67] Mališa Vučinić, Michał Król, Baptiste Jonglez, Titouan Coladon, and Bernard Tourancheau. 2017. Trickle-D: High fairness and low transmission load with dynamic redundancy. *IEEE Internet of Things Journal* 4, 5 (2017), 1477–1488. DOI: http://dx.doi.org/10.1109/JIOT.2017.2650318

[68] Baraq Ghaleb, Ahmed Y. Al-Dubai, Elias Ekonomou, Imed Romdhani, Youssef Nasser, and Azzedine Boukerche. 2018. A novel adaptive and efficient routing update scheme for low-power lossy networks in IoT. *IEEE Internet of Things Journal* 5, 6 (2018), 5177–5189. DOI: http://dx.doi.org/10.1109/JIOT.2018.2862364

[69] D. De Guglielmo, A. Seghetti, G. Anastasi, and M. Conti. 2014. A performance analysis of the network formation process in IEEE 802.15.4e TSCH wireless sensor/actuator networks. In *Proceedings of the IEEE Symposium on Computers and Communications*. 1–6. DOI: http://dx.doi.org/10.1109/ISCC.2014.6912607

[70] D. De Guglielmo, S. Brienza, and G. Anastasi. 2016. A model-based beacon scheduling algorithm for IEEE 802.15.4e TSCH networks. In *Proceedings of the IEEE 17th International Symposium on A World Wireless, Mobile Multimedia Networks*. 1–9. DOI: http://dx.doi.org/10.1109/WoWMoM.2016.7523517

[71] E. Vogli, G. Ribezzo, L. A. Grieco, and G. Boggia. 2015. Fast join and synchronization scheme in the IEEE 802.15.4e MAC. In *Proceedings of the IEEE Wireless Communications and Networking Conference Workshops*. 85–90. DOI: http://dx.doi.org/10.1109/WCNCW.2015.7122534

[72] Thang Phan Duy and YoungHan Kim. 2015. An efficient joining scheme in IEEE 802.15.4e. In *Proceedings of the 2015 International Conference on Information and Communication Technology Convergence*. 226–229. DOI: http://dx.doi.org/10.1109/ICTC.2015.7354534

[73] Thang Phan Duy, Thanh Dinh, and Younghan Kim. 2016. A rapid joining scheme based on fuzzy logic for highly dynamic IEEE 802.15.4e time-slotted channel hopping networks. *International Journal of Distributed Sensor Networks* 12, 8 (2016), 1–10.

[74] Elvis Vogli, Giuseppe Ribezzo, L. Alfredo Grieco, and Gennaro Boggia. 2018. Fast network joining algorithms in industrial IEEE 802.15.4 deployments. *Ad Hoc Networks*. 69 (2018), 65–75. DOI: http://dx.doi.org/10.1016/j.adhoc.2017.10.013

[75] Ines Khoufi, Pascale Minet, and Badr Rmili. 2017. Beacon advertising in an IEEE 802.15.4e TSCH network for space launch vehicles. In *Proceedings of the 7th European Conference for Aeronautics and Aerospace Sciences*. 1–15.

[76] Ines Khoufi and Pascale Minet. 2018. An enhanced deterministic beacon advertising algorithm for building TSCH networks. *Annals of Telecommunications* 73 (2018), 745–757.

[77] Apostolos Karalis. 2018. ATP: A fast joining technique for IEEE802.15. 4-TSCH networks. In *Proceedings of the 2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks."* 588–599. DOI: http://dx.doi.org/10.1109/WoWMoM.2018.8449759

[78] Carlos M. García Algora, Vitalio Alfonso Reguera, Evelio M. García Fernández, and Kris Steenhaut. 2018. Parallel rendezvous-based association for IEEE 802.15.4 TSCH Networks. *IEEE Sensors Journal* 18, 21 (2018), 9005–9020. DOI: http://dx.doi.org/10.1109/JSEN.2018.2868410

[79] Byeong-Hwan Bae and Sang-Hwa Chung. 2020. Fast synchronization scheme using 2-way parallel rendezvous in IEEE 802.15.4 TSCH. *Sensors* 20, 5 (2020), 1–19. DOI: http://dx.doi.org/10.3390/s20051303

[80] Mohamed Mohamadi, Badis Djamaa, Mustapha Reda Senouci, and Abdelhamid Mellouk. 2021. FAN: Fast and active network formation in IEEE 802.15.4 TSCH networks. *Journal of Network and Computer Applications*. 183–184 (2021), 103026. DOI: http://dx.doi.org/10.1016/j.jnca.2021.103026

[81] Carlos Manuel García Algora, Erik Ortiz Guerra, Samuel Montejo-Sánchez, Evelio M. García Fernández, and Kris Steenhaut. 2021. A theoretical association time model for IEEE 802.15.4 TSCH networks. *IEEE Communications Letters* 25, 2 (2021), 656–659. DOI: http://dx.doi.org/10.1109/LCOMM.2020.3032674

[82] C. Vallati, S. Brienza, G. Anastasi, and S. K. Das. 2019. Improving network formation in 6TiSCH networks. *IEEE Transactions on Mobile Computing* 18, 1 (2019), 98–110. DOI: http://dx.doi.org/10.1109/TMC.2018.2828835

[83] A. Kalita and M. Khatua. 2021. Autonomous allocation and scheduling of minimal cell in 6TiSCH network. *IEEE Internet of Things Journal* 8, 15 (2021), 12242–12250. DOI: http://dx.doi.org/10.1109/JIOT.2021.3062115

[84] Malisa Vucinic, Thomas Watteyne, and Xavier Vilajosana. 2017. Broadcasting strategies in 6TiSCH networks. *Internet Technology Letters* 1, 1 (2017), 1–6. https://doi.org/10.1002/itl2.15

[85] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko. 2011. *The Trickle Algorithm*. RFC 6206. IETF.

[86] Alakesh Kalita and Manas Khatua. 2019. Faster joining in 6TiSCH network using dynamic beacon interval. In *Proceedings of the 2019 11th International Conference on Communication Systems Networks*. 454–457. DOI: http://dx.doi.org/10.1109/COMSNETS.2019.8711460

[87] A. Kalita and M. Khatua. 2021. Channel condition based dynamic beacon interval for faster formation of 6TiSCH network. *IEEE Transactions on Mobile Computing* 20, 7 (2021), 2326–2337. DOI: http://dx.doi.org/10.1109/TMC.2020.2980828

[88] Alakesh Kalita and Manas Khatua. 2020. Opportunistic priority alternation scheme for faster formation of 6TiSCH network. In *Proceedings of the 21st International Conference on Distributed Computing and Networking*. 1–5.

[89] A. Kalita and M. Khatua. 2021. Opportunistic transmission of control packets for faster formation of 6TiSCH network. *ACM Transactions on Internet of Things* 2, 1 (2021), 29 pages. DOI: http://dx.doi.org/10.1145/3430380

[90] Alakesh Kalita and Manas Khatua. 2021. Adaptive control packet broadcasting scheme for faster 6TiSCH network bootstrapping. *IEEE Internet of Things Journal* 8, 24 (2021), 17395–17402. 10.1109/JIOT.2021.3080735

[91] C. Vallati and E. Mingozzi. 2013. Trickle-F: Fair broadcast suppression to improve energy-efficient route formation with the RPL routing protocol. In *Proceedings of the Sustainable Internet and ICT for Sustainability*. 1–9.

[92] S. Aljawarneh, M. B. Yassein, and E. Masa'deh. 2017. A new elastic trickle timer algorithm for Internet of Things. *Journal of Network and Computer Applications* 89 (2017), 38–47.

[93] S. Murali and A. Jamalipour. 2019. Mobility-aware energy-efficient parent selection algorithm for low power and lossy networks. *IEEE Internet of Things Journal* 6, 2 (2019), 2593–2601.

[94] A. Kalita and M. Khatua. 2022. A Noncooperative Gaming Approach for Control Packet Transmission in 6TiSCH Network. *IEEE Internet of Things Journal* 9, 5 (2022), 3954–3961. DOI: 10.1109/JIOT.2021.3101941

[95] X. Vilajosana, Q. Wang, F. Chraim, T. Watteyne, T. Chang, and K. S. J. Pister. 2014. A realistic energy consumption model for TSCH networks. *IEEE Sensors Journal* 14, 2 (2014), 482–489. http://dx.doi.org/10.1109/JSEN.2013.2285411

[96] Hyung-Sin Kim, Jeonggil Ko, David E. Culler, and Jeongyeup Paek. 2017. Challenging the IPv6 routing protocol for low-power and lossy networks (RPL): A survey. *IEEE Communications Surveys Tutorials* 19, 4 (2017), 2502–2525. http://dx.doi.org/10.1109/COMST.2017.2751617

[97] David Stanislowski, Xavier Vilajosana, Qin Wang, Thomas Watteyne, and Kristofer S. J. Pister. 2014. Adaptive synchronization in IEEE802.15.4e networks. *IEEE Transactions on Industrial Informatics* 10, 1 (2014), 795–802. http://dx.doi.org/10.1109/TII.2013.2255062

[98] Tengfei Chang, Thomas Watteyne, Kris Pister, and Qin Wang. 2015. Adaptive synchronization in multi-hop TSCH networks. *Computer Networks* 76 (2015), 165–176. http://dx.doi.org/10.1016/j.comnet.2014.11.003

[99] Yassine Boufenneche, Rafik Zitouni, Laurent George, and Nawel Gharbi. 2020. Network formation in 6TiSCH industrial internet of things under misbehaved nodes. In *Proceedings of the 2020 7th International Conference on Internet of Things: Systems, Management and Security*. 1–6. http://dx.doi.org/10.1109/IOTSMS52051.2020.9340200

[100] Alakesh Kalita, Alessandro Brighente, Manas Khatua, and Mauro Conti. 2022. Effect of DIS attack on 6TiSCH network formation. *IEEE Communications Letters* 26, 5 (2022), 1190–1193. http://dx.doi.org/10.1109/LCOMM.2022.3155992

[101] Cong Pu. 2020. Sybil attack in RPL-based internet of things: Analysis and defenses. *IEEE Internet of Things Journal* 7, 6 (2020), 4937–4949. DOI: http://dx.doi.org/10.1109/JIOT.2020.2971463

[102] Jorge Granjal, Edmundo Monteiro, and Jorge Sá Silva. 2015. Security for the internet of things: A survey of existing protocols and open research issues. *IEEE Communications Surveys Tutorials* 17, 3 (2015), 1294–1312. DOI: http://dx.doi.org/10.1109/COMST.2015.2388550

[103] Michael Baddeley, Reza Nejabati, George Oikonomou, Sedat Gormus, Mahesh Sooriyabandara, and Dimitra Simeonidou. 2017. Isolating SDN control traffic with layer-2 slicing in 6TiSCH industrial IoT networks. In *Proceedings of the 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks*. 247–251. http://dx.doi.org/10.1109/NFV-SDN.2017.8169876

[104] Michael Baddeley, Usman Raza, Aleksandar Stanoev, George Oikonomou, Reza Nejabati, Mahesh Sooriyabandara, and Dimitra Simeonidou. 2019. Atomic-SDN: Is synchronous flooding the solution to software-defined networking in IoT? *IEEE Access* 7 (2019), 96019–96034. http://dx.doi.org/10.1109/ACCESS.2019.2920100

Received September 2021; revised March 2022; accepted May 2022