This paper has been presented at:

LCN 2019 : The 44th IEEE Conference on Local Computer Networks.

October 14-17, 2019, Osnabrück, Germany

https://www.ieeelcn.org/index.html

IEEE 802.15.4 TSCH in Sub-GHz: Design Considerations and Multi-band Support

Martina Brachmann*, Simon Duquennoy*, Nicolas Tsiftes*, and Thiemo Voigt*†
*RISE Research Institutes of Sweden, Stockholm, Sweden
†Uppsala Universitet, Uppsala, Sweden
firstname.lastname@ri.se

***Abstract*—In this paper, we address the support of Time-Slotted Channel Hopping (TSCH) on multiple frequency bands within a single TSCH network. This allows to simultaneously run applications with different requirements on link characteristics and to increase resilience against interference. To this end, we first enable sub-GHz communication in TSCH, which has been primarily defined for the 2.4 GHz band. Thereafter, we propose two designs to support multiple physical layers in TSCH on the same nodes. Our experimental evaluation shows that TSCH is applicable in a wide range of data rates between 1.2 kbps and 1000 kbps. We find that data rates of 50 kbps and below have a long communication range and a nearly perfect link symmetry, but also have a 20x higher channel utilization compared to higher data rates, increasing the risk of collisions. Using these findings, we show the advantages of the multi-band support on the example of synchronization accuracy when exchanging TSCH beacons with a low data rate and application data at a high data rate.**

***Index Terms*—Sub-GHz communication, IEEE 802.15.4, TSCH, multi-band support, timeslot duration**

I. Introduction

Using sub-GHz bands instead of the widely deployed 2.4 GHz frequencies in low-power wireless networks promises a longer communication range [1], [2], [3], [4], less interference with co-existing technologies like BLE, Wi-Fi, or other IEEE 802.15.4 applications [5], [6], [7], and thus, a higher reliability and fewer packet retransmissions. Many protocols and technologies like SigFox, LoRa, IEEE 802.11ah, Z-Wave, and the IEEE 802.15.4 standard already employ sub-GHz communication [2], [8], [9], [10], [11].

In 2012, Time-Slotted Channel Hopping (TSCH) was introduced to the MAC layer of IEEE 802.15.4e [12]. It combines time-slotted scheduling (for collision avoidance) with channel hopping (for frequency diversity). However, TSCH was primarily designed for the use in the 2.4 GHz band. The default duration of a TSCH timeslot at 2.4 GHz is sufficiently long to transmit a data frame and to receive an acknowledgment. However, the data rate at sub-GHz can be much lower, with the consequence that transmitting a single frame in the sub-GHz band can exceed the default duration of a TSCH timeslot.

Further, TSCH is designed to run in networks with only a single data rate on one frequency band, limiting the network to a specific application. Multi-band support would allow a low-power wireless network to serve multiple applications with different requirements. For instance, alarm messages and actuation commands have relatively limited throughput requirements, whereas the packet loss from node to gateway should be preferably low. Hence, a long range and a reliable link, i.e., a low data rate at sub-GHz, is desired in such applications. By contrast, data dissemination and sensor data streaming requires high throughput, but the packet reliability requirements are typically not as strict. Thus, these applications run at a high data rate. The scenarios described above typically run in two separate networks, whereas with multi-band support only a single network is needed. Additionally, multi-band support enhances the reliability when the network is challenged by interference or other changes in the environment [13].

In this paper, we address the use of TSCH in multiple bands in a single network and make the following contributions:

- We dissect TSCH timeslots to enable sub-GHz TSCH, and present a method to derive suitable timings for a given radio hardware and physical layer (PHY) configuration. We show that for high data rate configurations, the effective data rate is limited by the computation time rather than the time the frame is in the air. As a consequence, increasing the data rate from 250 kbps to 1000 kbps does not even double the effective data rate with our hardware.
- We present two different designs for combining multiple bands in a TSCH network: (1) a single-template design in which all slots share the same timing template based on the slowest PHY, and (2) a multi-template design in which the fastest PHY determines the slot length to use for all PHYs, whereas slower PHYs are scheduled to have logical slots spanning multiple real slots.
- We implement multi-PHY TSCH within the TSCH implementation [14] in the Contiki-NG operating system, and evaluate the performance of TSCH with different PHY configurations through a 25-node deployment in an office environment.

Our results show that TSCH is applicable on a wide variety of data rates in the sub-GHz band, from 1.2 kbps to 1000 kbps. Additionally, we demonstrate that multiple PHYs can be combined within a single schedule, for applications with heterogeneous performance requirements. We show that single-hop beacons sent with a low data rate can co-exist with multi-hop RPL traffic sent at a high data rate in a single network. Hence, we can reduce the synchronization errors, which are known to accumulate with hop counts in TSCH networks [15], [16].

The rest of the paper is structured as follows. We give an overview of TSCH in Section II and describe how TSCH can be adapted to various data rates in Section III. Thereafter, we explore the design space for multi-PHY TSCH networks in Section IV, and experimentally evaluate our work in Section V. Lastly, we cover related work in Section VI and conclude the paper in Section VII.

II. Background

We first review the design of TSCH, and in particular how TSCH timeslots are structured. This structure will be important for our work in Section III, where we will carefully modify certain elements to enable TSCH on different data rates.

A. TSCH Overview

TSCH is a MAC protocol for low-power wireless communication defined in the IEEE 802.15.4 standard [17]. It employs frequency-division and time-division multiplexing to enable energy-efficient and deterministic communication. The 6TiSCH Working Group at IETF is further working toward standardizing the use of IPv6 over TSCH [18].

A TSCH network is created by a Private Area Network (PAN) coordinator node, and other nodes can join the network after hearing periodic beacons. Nodes schedule their communication using a globally synchronized timeslot counter, the Absolute Slot Number (ASN). The ASN is typically propagated from the PAN coordinator in a Directed Acyclic Graph (DAG) topology, in which each node has a dedicated time-source neighbor. Data packets and acknowledgment (ACK) packets can be time-stamped, allowing nodes to adjust the time when receiving such packets from their time-source neighbor.

All communication in TSCH is coordinated through a schedule, as opposed to asynchronous duty-cycling methods such as Low-Power Listening (LPL) and Low-Power Probing (LPP) [19], [20]. The TSCH schedule determines when and on which channel a node can communicate. The method to build the schedule is not specified by the standard; it can either be computed centrally to satisfy some specific application requirements and network topology, or it can be formed autonomously among nodes within the network [21].

A schedule consists of a set of slotframes that are repeated indefinitely throughout the network's lifetime. Slotframes may be configured with different lengths; a longer slotframe can result in a lower energy consumption, but also a higher latency and a lower achievable data rate. Each slotframe in turn consists of timeslots of uniform length that determine which nodes should listen for packets, transmit packets (if any are enqueued), or turn the radio off to preserve energy. Timeslots are further divided into timing-sensitive parts to support the transmission of a data frame up to the maximum size, and for the receiver of the data frame to send a corresponding ACK.

B. The TSCH Timeslot Timing Template

The structure of a TSCH timeslot is illustrated in Figure 1. In the following, we describe the different events in case of an acknowledged unicast transmission. We start with the sender that transmits (TX) a frame:

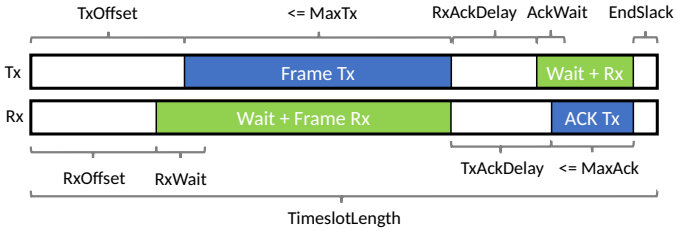

Fig. 1. Simplified TSCH timeslot timing template. Note that all timings except `EndSlack` are part of the timings in the IEEE 802.15.4 standard. We use the `EndSlack` label for simplification.

The timeslot starts with `TxOffset`. During `TxOffset`, the sender turns the radio on, possibly performs Clear Channel Assessment (CCA), prepares (i.e., adds link layer headers), secures (i.e., encrypts and authenticates) and copies the frame to transmit into the radio buffer. `TxOffset` must be longer than the sum of the time needed for preparation, encryption, enqueuing, and PHY synchronization header (i.e., preamble and Start of Frame Delimiter (SFD)) transmission of a maximum sized packet on any possible hardware that will join a network with a given timing template. The transmission duration of the frame that follows the SFD must be equal or smaller than the duration required to transmit a maximum sized frame, indicated as `MaxTx`. The maximum frame length in IEEE 802.15.4 is 128 bytes including the frame's length field. After transmitting the frame, the sender turns the radio to receive mode in `RxAckDelay` and starts listening for an ACK from the receiver. In case no ACK is detected, the sender stops listening after `AckWait`. If the sender successfully receives the ACK, it de-secures (i.e., de-crypts and checks authenticity) it and turns the radio off within `EndSlack`.

A TSCH timeslot for a node that listens and possibly receives (RX) a frame has the following events:

The TSCH timeslot starts with `RxOffset`. During `RXOffset`, the receiver turns its radio on and starts listening for incoming frames. In case no frame is detected it stops listening after `RxWait`. Otherwise, it leaves the radio turned on to receive the incoming frame. After the frame reception, the receiver de-secures (i.e., de-crypts and checks authenticity) the frame, prepares and secures the ACK, turns the radio to transmit mode and sends the synchronization header during `TxAckDelay`. The duration of the transmission of the ACK must be smaller or equal than the maximum time required to send a maximum length ACK, indicated as `MaxAck`. The receiver turns the radio off in `EndSlack`.

III. Enabling Sub-GHz TSCH

In this section, we discuss how to enable sub-GHz communication with TSCH. As TSCH was originally intended for the 2.4 GHz band, its timing configuration has been carefully tuned for these frequencies. A TSCH timeslot is by default 10 ms, which is enough time to transmit a 128 byte packet and receive an ACK at 2.4 GHz. The data rate at sub-GHz

TABLE I
DEFINITIONS AND EQUATIONS FOR TIMESLOT TIMING DERIVATION.

ByteTime	The air-time of one byte; $8\ bit/data\ rate$
SyncHeaderLen	The length of the PHY synchronization header; 5 bytes in our case (3 byte preamble, 2 byte SFD)
SyncHeaderTime	The air-time of the PHY synchronization header; $ByteTime \times SyncHeaderLen$
MaxFrameLen	The maximum number of bytes in a frame; 128 byte in our case (1 byte length, 127 byte payload incl. 2 byte CRC)
MaxACKLen	The maximum number of bytes in an ACK; 10 bytes in our case (1 byte length, 9 byte payload incl. 2 byte CRC)
GuardTime	The guard time for frame reception; we use 2.2 ms, as suggested in the IEEE 802.15.4 standard [17]
ACKGuardTime	The guard time for ACK reception; we use 0.2 ms by default
TxOffset	Measured
RxOffset	$TxOffset - SyncHeaderTime - GuardTime/2$
RxWait	$GuardTime + SyncHeaderTime$
MaxTx	$ByteTime \times MaxFrameLen$
TxAckDelay	Measured
RxAckDelay	$TxAckDelay - SyncHeaderTime - ACKGuardTime/2$
AckWait	$AckGuardTime + SyncHeaderTime$
MaxAck	$ByteTime \times MaxAckLen$
EndSlack	Constant, we use 0.5 ms

can be much lower than at 2.4 GHz. Sending a single byte at 1.2 kbps already takes 6.7 ms. Hence, the current timeslot provisioning precludes the transmission of even the smallest packets. Therefore, we must derive new timeslot configurations for the different sub-GHz bands that we aim to support.

A. Method for Deriving Timings

By examining the timeslot template described in Section II-B, we can derive that sending and receiving a frame and an ACK only depends on the data rate. `EndSlack` is constant while `TxOffset` and `TxAckDelay` include both computing operations as well as operations relying on the data rate. During `TxOffset` and `TxAckDelay` the duration of computing operations should be constant and only the time to transmit the PHY synchronization header should vary depending on the data rate. Hence, in initial experiments we expected $TxOffset - SyncHeaderTime$ to be constant. However, we found in these initial experiments that the computing time increased with decreasing data rates on our reference platform, the Zolertia Firefly running at a clock speed of 32 MHz. The reasons might be internal radio calibration processes that depend on the symbol rate, and thus, on the data rate.

Our method to derive the timings, therefore, consists in quantifying experimentally the achievable delay for both `TxOffset` and `TxAckDelay`. In particular, we timestamp packets and ACKs in consecutive TSCH timeslots. Consecutive slots can usually remain synchronized within one clock tick—e.g., $\pm 30.5\ \mu s$ using the stable 32 kHz clock. After adding the timestamping error, this method results in $\pm 61.0 \mu s$. An alternative for quantifying `TxOffset` and `TxAckDelay` is to use a logic analyzer and to measure the rising pin after the SFD has been successfully received. From the measurements of both `TxOffset` and `TxAckDelay`, we derive the timeslot timing template as shown in Table I. Simply put, the offset to receive a frame/ACK is based on the corresponding transmission offset, minus the air-time of the PHY synchronization header and half the guard time. By doing so, we ensure that the receiver listens long enough to decode a full preamble, plus the guard, centered on the expected arrival time of the SFD.

TABLE II
PHY CONFIGURATIONS.

Name	Channel separation	Modulation	# Channels
1.2 kbps @868 MHz	0.2 MHz	2-GFSK	34
8 kbps @868 MHz	0.2 MHz	2-GFSK	34
50 kbps @868 MHz	0.2 MHz	2-GFSK	34
250 kbps @868 MHz	1 MHz	2-GFSK	7
1000 kbps @868 MHz	1.67 MHz	4-GFSK	4
250 kbps @2.4 GHz	5 MHz	OQPSK	16

B. Examples from 1.2 kbps to 1000 kbps

For our case study, we use the TI CC1200 radio chip and explore PHY configurations with data rates ranging from 1.2 kbps to 1000 kbps, covering various link characteristics [22] that are interesting for a wide range of applications having different requirements on, e.g., communication range and reliability. We use setups produced with SmartRF Studio 7 (v2.7.0), the tool provided by TI for configuration and evaluation of their chips. The configurations we use are summarized in Table II.

We port Contiki-NG's TSCH implementation [14] to the CC1200, using the Zolertia Firefly platform. We program nodes with each configuration shown in Table II and measure—but without performing CCA—`TxOffset` and `TxAckDelay` experimentally. From the measurements, we derive full timeslot timings, as shown in Table III.

The total timeslot length varies widely, from just over 1 s to below 6 ms. As indicated in Table I, the `MaxTx` value in each timeslot is provisioned to be able to transmit a 128-byte frame. Using the derived timeslot lengths, we calculate the effective data rate in kbps as $\frac{128 \times 8}{TimeslotLength}$. The effective data rates for the data rates under consideration in this paper are shown in the bottom row in in Table III. For low data rate configurations, the effective data rate is close to that of the PHY, because the timeslot timings are mostly bound by the byte air-time. For example, in the 1.2 kbps case, the effective data rate is 1 kbps (83% effectiveness). By contrast, at higher data rates, the timings are mostly limited by computation time, not by the air-time of a packet. This results in a lower relative effectiveness; e.g., in the 1000 kbps case, we achieve an effective data rate of 179.5 kbps (18% effectiveness). As a consequence, increasing the data rate fourfold, e.g., from 250 kbps to 1000 kbps does not even double the effective data rate with our hardware.

TABLE III
TIMESLOT TIMINGS FOR EACH DATA RATE AT 868 MHZ. *MEASURED.

Data rate (kbps)	1.2	8	50	250	1000
ByteTime (ms)	6.667	1.000	0.160	0.032	0.008
SyncHeaderTime (ms)	33.333	5.000	0.800	0.160	0.040
GuardTime (ms)	2.200	2.200	2.200	2.200	2.200
ACKGuardTime (ms)	0.400	0.400	0.400	0.400	0.400
EndSlack (ms)	0.500	0.500	0.500	0.500	0.500
TxOffset* (ms)	55.000	10.100	3.800	3.700	2.200
RxOffset (ms)	20.567	4.000	1.900	2.440	1.060
RxWait (ms)	35.533	7.200	3.000	2.360	2.240
MaxTx (ms)	853.334	128.000	20.480	4.096	1.024
TxAckDelay* (ms)	45.000	8.300	3.000	2.100	1.900
RxAckDelay (ms)	11.467	3.100	2.000	1.740	1.660
AckWait (ms)	33.733	5.400	1.200	0.560	0.440
MaxAck (ms)	66.667	10.000	1.600	0.320	0.080
TimeslotLength (ms)	1020.500	156.900	29.380	10.716	5.704
Effective data rate (kbps)	1.0	6.5	34.9	95.6	179.5

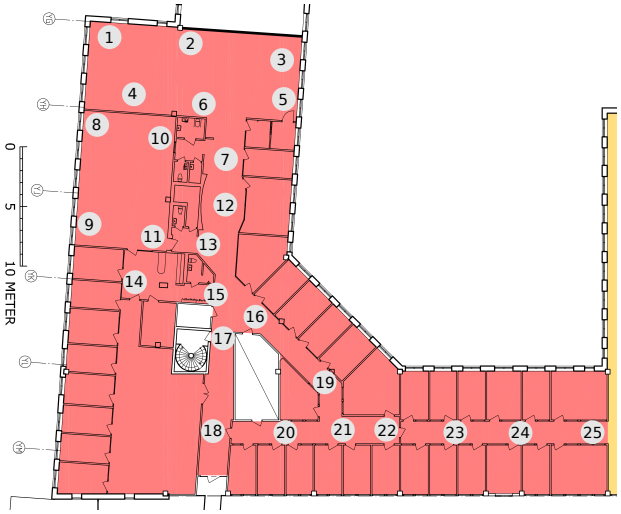

Fig. 2. Map of the 25-node testbed used in the experiments.

C. Standard-compliance Considerations

The timings described above can be implemented in compliance with the IEEE 802.15.4 standard, with one exception, in the 1.2 kbps case. The standard defines Information Elements (IEs) that are embedded in the beacons to disseminate network information such as timeslot timing, hopping sequence, and schedule. For the *TSCH Timeslot IE*, all timeslot timing elements are included, in microseconds. The standard provisions two bytes for each element, and optionally extends `MaxTx` and `TimeslotLength` to three bytes. In our 1.2 kbps template, however, the element `MaxAck` is 66,667μs, which is more than can be represented in two bytes. This means that such timings cannot be disseminated in the network dynamically. Instead, they have to be hard-coded and known prior to deployment. For future versions of IEEE 802.15.4, we suggest extending `MaxAck` to three bytes.

IV. MULTI-PHY DESIGN

After deriving the timings for sub-GHz bands, we now explore the design space for multi-PHY TSCH networks. The idea is to enable schedules where transmissions at different data rates and modulations can co-exist. This is beneficial because different PHYs offer different trade-offs in reliability, range, and capacity. For instance, synchronization beacons do not require a high data rate, but can benefit from extended range. The same applies to urgent and occasional alerts in a network, e.g., in a fire detection scenario. On the other hand application traffic can require higher data rates, and benefit from faster PHYs, even if the shorter range necessitates multi-hop routing.

For the timings in different slots, we identify two main design directions: single template and multi-template.

A. Single-template Design

In the single-template design, all slots share the same timing template, regardless of their PHY. To enable this, all slots are downgraded to the slowest template. For instance, in a network where 1.2 kbps and 1000 kbps slots co-exist, all will have to run on 1 s long slots (see Table III). The main benefit of this design is simplicity and compliance with the single-PHY design covered by existing TSCH standard documents. The main drawback is, however, that running on the longest slots results in severely reduced network capacity.

B. Multi-template Design

We mitigate the capacity limitation of the single-template design with a multi-template design. Note that this design is not strictly compliant with current standard documents.

The idea in a multi-template design is to pick the shortest timeslot length as global time unit (ASN counter) for slots in the networks. Slots that use the fastest PHY can directly be scheduled as conventional slots. Slots that use slower PHYs require more time for each communication event. We enable such slots by spanning them over multiple slots. For instance, if the schedule is built with 8704 μs slots ($5704 + 3000$), 1000 kbps slots will use exactly one slot, while 1.2 kbps will span as many as 118 slots ($1,023,500/8704 \approx 117.6$).

V. EVALUATION

We experimentally evaluate our work with both the single-template design and the multi-template design using our derived TSCH timeslot timings for different bands. In the first part, we study the channel properties in a multi-PHY TSCH network, for six different PHY layer configurations using the single-template design. In the second part, we demonstrate a proof-of-concept multi-PHY network, where bands are switched at run-time using the multi-template design.

In our design, slots are simply tagged with information about what PHY they should use. At the beginning of the slot, the radio is re-configured to the new PHY before transmitting or receiving. Note that depending on the hardware and implementation, this may require extra slot time. In our implementation for the CC1200, we provide 3 ms for PHY

re-configuration, i.e., we use the timings in Table III, but with a slot length extended by 3 ms.

A. TSCH Wireless Medium Characterization

We use a testbed consisting of 25 Zolertia Firefly nodes deployed in an office environment, as depicted in Figure 2. The nodes combine a CC2538 and a CC1200 chip. The former is a System-on-Chip (SoC) comprising a Cortex-M3 MCU and a 2.4 GHz IEEE 802.15.4 radio. The latter is a sub-GHz radio chip that supports a wide range of data rates and modulation schemes. We experiment with all PHY configurations presented in Table II.

The additional results for the 2.4 GHz band are only provided as a base case, and are not directly comparable to the sub-GHz experiments. The CC1200 radio chip is equipped with an external antenna and operates with an output power of +14 dBm, while the CC2538 relies on an on-board ceramic antenna and operates at +7 dBm in our experiments.

To enable a systematic characterization, we implement TSCH with support for sub-GHz and the single-template design. We use one slotframe with a length of 157 timeslots (à 1 second length, see Table III), with one timeslot for each of the six PHYs and for each of the 25 nodes. Note that in TSCH it is recommended that slotframes use a prime number as length [18], hence the choice of 157: a prime number slightly greater than 6×25. Using a prime number ensures that the slots iterate over the channel hopping sequence uniformly. In each slotframe, each node sends a 26-byte broadcast packet exactly once on each PHY. All nodes listen at every slot. During our 4-hour experiments, we log all transmission and reception events, as well as an ambient RSSI scan, for every slot. This enables us to keep track of which nodes receive from which, on any given PHY and for any given channel.

1) Reach per Source and per Channel: We first discuss the single-hop transmission range observed in each configuration and *find that data rates of 50 kbps and below give a longer range compared to higher data rates, independently of the used channel*. Figure 3 and Figure 4 show, in each PHY configuration, the *reach* measured per source node and per channel, respectively. We define the *reach* as the number of receivers of broadcast transmissions. In our 25-node testbed, the maximum reach is hence 24, highlighted in Figure 3 and Figure 4 as solid, red line.

Figure 3 shows the reach per source. As depicted in the figure, sub-GHz PHYs with a data rate of 50 kbps and below exhibit a similar range, with most nodes above a reach of 20. The range starts to deteriorate at 250 kbps and beyond.

Figure 4 also shows the reach, but this time per-channel instead of per-node. In order to avoid any possibility of confusion with the channel naming for the different PHY configurations, we use labels and summarize the label-to-frequency mapping in Table IV. Similar to our findings in Figure 3, at data rates of 50 kbps and below, the vast majority of nodes have a reach over 20 on any given channel. The range decreases at 250 kbps and 1000 kbps, resulting in a reach of over 10 for each channel.

TABLE IV
MAPPING OF LABELS TO FREQUENCIES. THE FREQUENCIES CORRESPOND TO THE CHANNELS FOR DIFFERENT PHY CONFIGURATIONS.

Label	Freq. (MHz)	Label	Freq. (MHz)	Label	Freq. (MHz)
M0	863.125	M18	866.725	N2	866.465
M1	863.325	M19	866.925	N3	868.135
M2	863.525	M20	867.125	G0	2405
M3	863.725	M21	867.325	G1	2410
M4	863.925	M22	867.525	G2	2415
M5	864.125	M23	867.725	G3	2420
M6	864.325	M24	867.925	G4	2425
M7	864.525	M25	868.125	G5	2430
M8	864.725	M26	868.325	G6	2435
M9	864.925	M27	868.525	G7	2440
M10	865.125	M28	868.725	G8	2445
M11	865.325	M29	868.925	G9	2450
M12	865.525	M30	869.125	G10	2455
M13	865.725	M31	869.325	G11	2460
M14	865.925	M32	869.525	G12	2465
M15	866.125	M33	869.725	G13	2470
M16	866.325	N0	863.125	G14	2475
M17	866.525	N1	864.795	G15	2480

The 2.4 GHz configuration is also shown in Figure 3 and Figure 4, respectively, for reference. We observe that it offers significantly poorer reach per source and per channel than the 868 MHz, 1000 kbps configuration. We can attribute this to (1) the different propagation properties of different carriers, (2) the fact that our testbed has external antennas for the 868 MHz chip, but only on-board ceramic antennas for the 2.4 GHz chip, and (3) the lower transmission power for 2.4 GHz.

2) Link Asymmetry: We examine the asymmetry of wireless links in the testbed, for different PHY configurations and *find that data rates of 50 kbps and below achieve nearly perfectly symmetric links*. A high asymmetry reduces the possibility for reliable two-way communication, and poses challenges for routing protocols [23]. We compute the asymmetry of a link as the absolute difference between the Packet Reception Ratio (PRR) in one direction and in the opposite direction. An asymmetry of 1 means that the link is perfect in one direction, but broken in the other, whereas an asymmetry of 0 means that the link has identical PRRs in both directions.

Figure 5 shows the asymmetry for different PHYs. At data rates between 1.2 kbps and 50 kbps, nearly all links are perfectly symmetric: only outliers are above 0. As the data rate increases, so does asymmetry. This suggests that higher data rates not only result in a shorter range (see Section V-A1), but also in more varying link qualities across directions. This has practical implications for certain higher-layer protocols such as MAC and routing, which may rely on the assumption that underlying links are symmetric. Note that even in the worst case (2.4 GHz) though, the asymmetry remains manageable, with only a few outliers exceeding a value of 0.4.

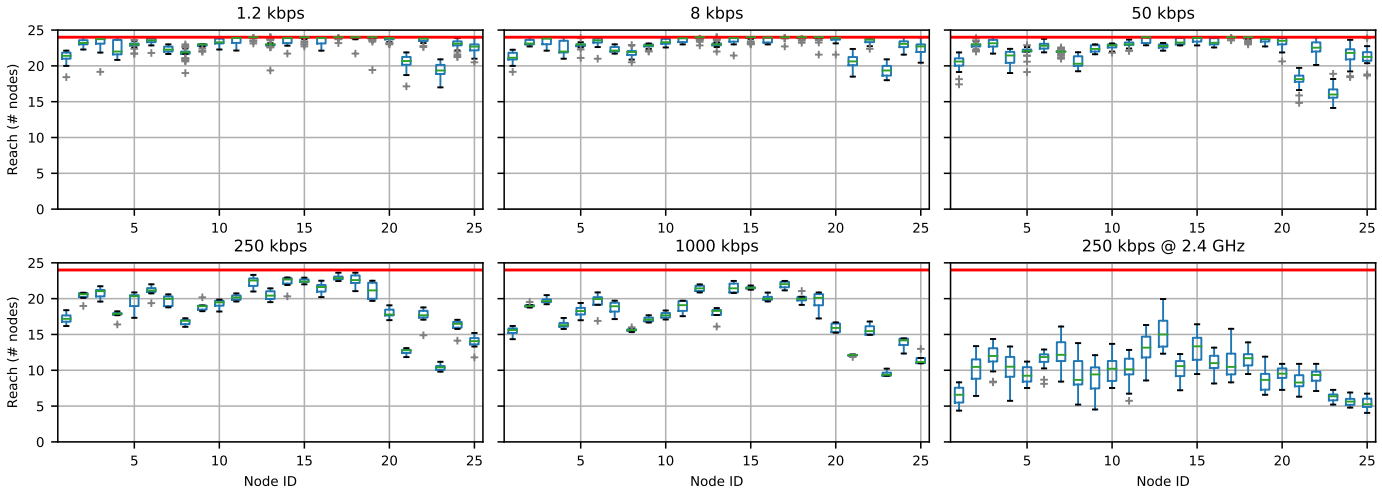

Fig. 3. Number of nodes reached per source in a single-hop transmission. Data rates of 50 kbps and below have most nodes achieving a reach above 20. With higher data rates the single-hop transmission range decreases, resulting in a lower reach. The red lines in the top of the figures indicate the maximum number of nodes that can be reached in perfect conditions.

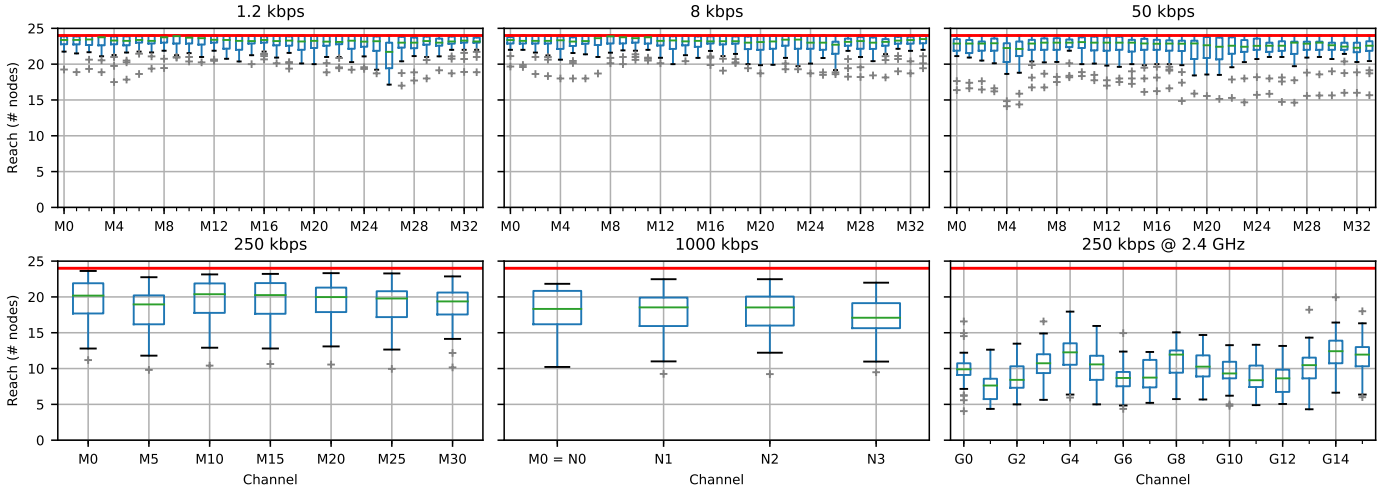

Fig. 4. Number of nodes reached per channel. PHY configurations with data rates of 50 kbps and below have a reach of mostly 20 or above for each channel while at data rates above 50 kbps, most nodes achieve a reach above 10. The red lines in the top of the figures indicate the maximum number of nodes that can be reached in perfect conditions.

B. Multi-PHY Proof of Concept

Our proof-of-concept consists in a network with two co-existing PHYs: the 1.2 kbps and the 1000 kbps configurations shown in Table III. We use a data rate of 1.2 kbps for TSCH beacons, enabling long-range, single-hop time synchronization in so-called *beacon band* timeslots. We further use the 1000 kbps data rate for data traffic in so-called *data band* timeslots. Both bands are used in the multi-template design. Above TSCH, we run a low-power IPv6 stack, including 6TiSCH, 6LoWPAN, and the RPL routing protocol. The data traffic consists mostly of RPL control traffic. This design is inspired by how WiFi handles different data rates: sending beacons at the slowest rate, and then possibly accommodating data traffic at higher rates. The payload excluding length and CRC of the beacon packets is 26 bytes and varies for the data packets from 24 bytes to 100 bytes with an average payload length of 78 bytes.

We run the experiments in the 25-node testbed described in Section V-A. We pick node #18 as network coordinator and RPL root. This guarantees single-hop connectivity at 1.2 kbps and two hops at 1000 kbps. We perform channel hopping on all four possible channels for the data band, but only use channels M0, M1 and M2 in the beacon band. The reason is that nodes scan all available channels for beacons to be able to join the network. Thus, forming a network takes a long time with 34 channels, as outlined by Muñoz et al. [24]. We run the experiment in the full testbed for 1 hour, three times. All runs exhibit similar properties; we discuss here one selected, representative run.

1) Multi-PHY usage pattern: Figure 6 shows the usage patterns of both beacon band and data band and *we find that the beacon band at 1.2 kbps has a 20x higher channel occupancy*

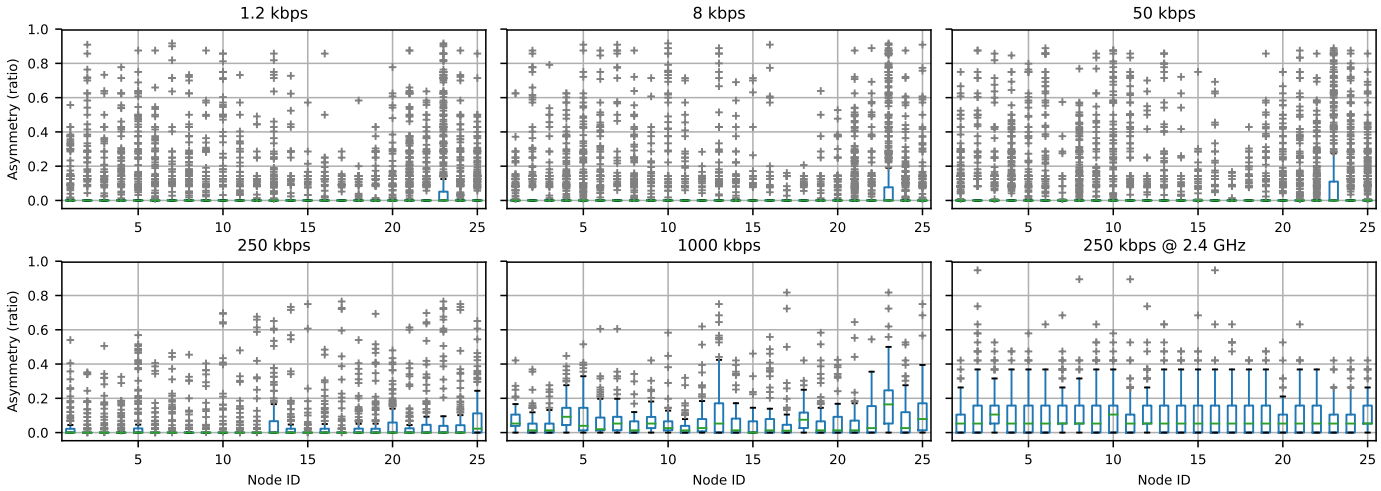

Fig. 5. Link asymmetry. The lower data rate configurations between 1.2 kbps and 50 kbps have highly symmetric links, with nearly all links having averages close to 0. The asymmetry increases with higher data rates.

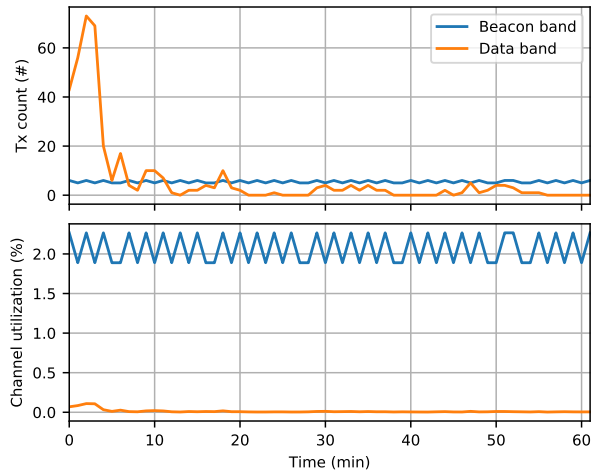

Fig. 6. Channel utilization in the multi-template design. The beacon band uses 1.2 kbps slots for TSCH beacons in order to support long-range, single-hop time synchronization. The data band uses 1000 kbps slots for short-range data traffic. The higher data rate of the data bands contributes to making its channel utilization lower than 0.1%, thereby reducing the risk of collisions.

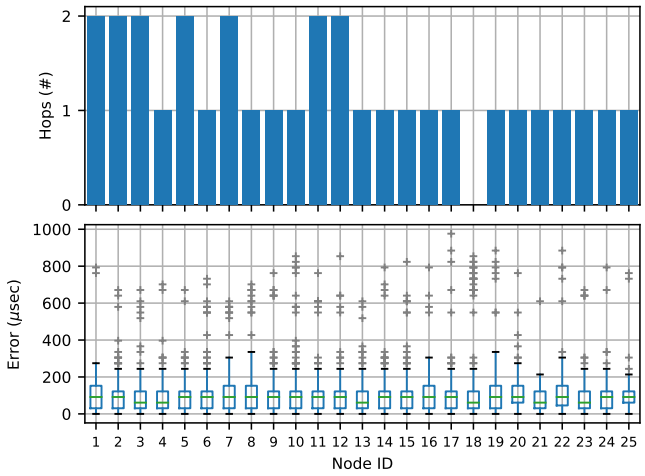

Fig. 7. Synchronization accuracy at the data band among nodes at different hop distances from the coordinator (node #18). Since the beacon band is configured for long-range, sub-GHz communication, the nodes are able to receive synchronization beacons directly from the coordinator. The synchronization accuracy is thus not affected by the hop distances in the data band.

compared to the data band at 1000 kbps. The top figure shows the transmission count over time. The beacon band has a nearly constant usage (periodic beaconing) while the data band starts with an initial peak and then flattens out with 0–5 packets per minute in the whole network. The bottom figure shows the same data, but with the perspective of channel utilization. The channel utilization is the average amount of time a node spends for transmission. It highlights that because of its lower data rate, the beacon band has a significantly higher channel occupancy than the data band (about 2% vs. less than 0.1%). A low channel utilization reduces the risk of collisions in a network and interference with other communication technologies operating on overlapping frequencies in the same area.

2) Synchronization Accuracy: One of the objectives of this proof-of-concept is to stress the trade-offs between different PHYs, and to demonstrate that single-hop beacons can coexist within a multi-hop network. In particular, *we find that the multi-hop data band benefits from the synchronization in the single-hop beacon band*. Figure 7 shows the synchronization accuracy resulting from such single-hop synchronization. At the top, we show the hop count of every node in the data band, ranging between 1 and 2. Since node #18 is the RPL root, its hop count is 0. The bottom figure shows the synchronization error at the data band—which was synchronized in the single-hop beacon band. Notice how the accuracy is independent from the node's hop count: 2-hop nodes are synchronized just as tightly as 1-hop nodes. This is as opposed to conventional TSCH networks where the synchronization error is known to accumulate with hop count [15], [16].

VI. Related Work

We review related work in three categories: (1) multi-band support, (2) TSCH timeslot adjustments, and (3) wireless medium characterization.

Multi-band support. Using multiple frequency bands within a low-power wireless network has previously been explored in different contexts. For example, wake-up radios often use a sub-GHz radio to listen for potential wake-up messages while the actual communication uses the 2.4 GHz frequencies [25], [26]. Kusy et al. [13] propose a dual radio network architecture where nodes transmit and receive identical packets at different frequency bands to increase communication robustness. Kim et al. [27] propose a hybrid network with nodes collecting small sensor data at 400 MHz from simple sensor devices to decrease the chance of packet retransmissions while nodes and gateways use 2.4 GHz for exchanging large data due to the higher data rate. We extend the authors' approach by providing a scheduling method using TSCH that can orchestrate traffic from multiple frequency bands. While our work mainly focuses on the timeslot duration and supporting multiple bands in the same TSCH network, Muñoz et al. [24] identify further design considerations across the 6TiSCH protocol stack when using a different frequency band than the default 2.4 GHz.

TSCH timeslot adjustments. Different authors have also examined the TSCH timeslot template for different purposes. Sciancalepore et al. [28] explore the impact of link-layer security on timeslot durations. Their evaluation shows the required timeslot durations for a variety of security levels. Tavakoli et al. [29] propose a new TSCH timeslot called *hybrid timeslot* that allows to efficiently support time-varying data traffic. The hybrid timeslot is primarily used as a dedicated slot for a single *owner* node. However, any other node can transmit in this slot after a backoff time when the owner node is not using it. This requires extra time within the timeslot. The authors propose to either increase the timeslot length for all timeslots or to decrease the maximum packet length. By contrast, we experimentally derive timeslot durations for different data rates.

Wireless medium characterization. Srinivasan et al. investigate link characteristics in IEEE 802.15.4 networks at 2.4 GHz [30]. Their study highlights the need to understand the different link characteristics when designing protocols, as earlier designs have in multiple cases made unwarranted assumptions that hamper their real-world performance. Woehrle et al. [5] conduct a similar study for the 868 MHz frequency band, and find that the noise floor in the 868 MHz band is lower and more stable than the 2.4 GHz band due to the absence of interference caused by Wi-Fi, Bluetooth, and other devices such as microwaves. Nevertheless, the link quality in 868 MHz is still influenced by multi-path propagation and human activity.

Sandoval et al. [1] compare the performance of IoT-based smart grids in 915 MHz and 2.4 GHz. They find that in a single-hop network, when both frequency bands are exposed to a comparable level of noise, a small network performs better in terms of mean network delay and power consumption per packet transmission at the 2.4 GHz band due to the high data rate. The 868 MHz band is, however, the better option in large networks because of a higher packet reception rate, and thus, fewer packet re-transmissions. Muñoz et al. [22] empirically investigates the performance of the 868 MHz frequency band using different modulation schemes and data rates in outdoor environments. These works serve as a motivation for our extension of TSCH's usability for different PHYs. By contrast, we demonstrate the successful operation and performance of TSCH with multiple channels and data rates.

VII. Conclusion

In this paper, we design and implement multi-band support for IEEE 802.15.4 TSCH, so as to enable networks to make use of the different characteristics provided by the different bands. Our analysis of the necessary timing configurations for different bands shows that TSCH is applicable not just to the 2.4 GHz band, but can also be used for communication on sub-GHz bands. Additionally, we propose two designs for multi-band support, single-template and multi-template, for IoT devices that are equipped with more than one type of radio chip. Our experimental evaluation not only demonstrates the successful operation of multi-band support, but also highlights the performance trade-offs that can be achieved. Hence, this enhances the possibilities to fine-tune the communication of IoT networks in case of different application requirements.

Acknowledgment

This work was financed by the H2020 collaborative Europe/Taiwan research project 5G-CORAL (grant num. 761586), the ERCIM Alain Bensoussan postdoc fellowship program, and the distributed environment E-care@home, funded by the Swedish Knowledge Foundation.

References

[1] R. M. Sandoval, A. J. Garcia-Sanchez, F. Garcia-Sanchez, and J. Garcia-Haro, "Evaluating the More Suitable ISM Frequency Band for IoT-Based Smart Grids: A Quantitative Study of 915 MHz vs. 2400 MHz," *Sensors*, vol. 17, 2016.

[2] W. Sun, M. Choi, and S. Choi, "IEEE 802.11ah: A Long Range 802.11 WLAN at Sub 1 GHz," *Journal of ICT Standardization*, vol. 1, 2013.

[3] C. Zhou, T. Plass, R. Jacksha, and J. A. Waynert, "RF Propagation in Mines and Tunnels," *IEEE Antennas and Propagation Magazine*, vol. 57, 2015.

[4] E. Kjeldsen and M. Hopkins, "An Experimental Look at Rf Propagation in Narrow Tunnels," in *IEEE Military Communications Conference (IEEE MILCOM)*, 2006.

[5] M. Woehrle, M. Bor, and K. Langendoen, "868 MHz: A noiseless environment, but no free lunch for protocol design," in *Proceedings of the International Conference on Networked Sensing Systems (INSS)*, 2012.

[6] V. Iyer, M. Woehrle, and K. Langendoen, "Chrysso – A Multi-channel Approach to Mitigate External Interference," in *The IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (IEEE SECON)*, 2011.

[7] G. Zhou, J. A. Stankovic, and S. H. Son, "Crowded Spectrum in Wireless Sensor Networks," in *Proceedings of the IEEE Workshop on Embedded Networked Sensor Systems (IEEE Emnets)*, 2005.

[8] M. Lauridsen, H. Nguyen, B. Vejlgaard, I. Z. Kovacs, P. Mogensen, and M. Sorensen, "Coverage Comparison of GPRS, NB-IoT, LoRa, and SigFox in a 7800 km^2 Area," in *IEEE Vehicular Technology Conference (IEEE VTC)*, 2017.

[9] E. D. Poorter, J. Hoebeke, M. Strobbe, I. Moerman, S. Latré, M. Weyn, B. Lannoo, and J. Famaey, "Sub-GHz LPWAN Network Coexistence, Management and Virtualization: An Overview and Open Research Challenges," *Wireless Personal Communications*, vol. 95, 2017.

[10] B. Vejlgaard, M. Lauridsen, H. Nguyen, I. Z. Kovacs, P. Mogensen, and M. Sorensen, "Coverage and Capacity Analysis of Sigfox, LoRa, GPRS, and NB-IoT," in *IEEE Vehicular Technology Conference (IEEE VTC)*, 2017.

[11] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *IEEE Communications Surveys & Tutorials*, vol. 19, 2017.

[12] "IEEE 802.15.4e-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 Computer Society, 2012.

[13] B. Kusy, C. Richter, W. Hu, M. Afanasyev, R. Jurdak, M. Brünig, D. Abbott, C. Huynh, and D. Ostry, "Radio Diversity for Reliable Communication in WSNs," in *Proceedings of the International Conference on Information Processing in Sensor Networks (ACM/IEEE IPSN)*, 2011.

[14] S. Duquennoy, A. Elsts, B. A. Nahas, and G. Oikonomou, "TSCH and 6TiSCH for Contiki: Challenges, Design and Evaluation," in *Proceedings of Distributed Computing in Sensor Systems (DCOSS)*, 2017.

[15] T. Chang, T. Watteyne, K. Pister, and Q. Wang, "Adaptive Synchronization in Multi-Hop TSCH Networks," *Computer Networks*, vol. 76, 2015.

[16] A. Elsts, X. Fafoutis, S. Duquennoy, G. Oikonomou, R. Piechocki, and I. Craddock, "Temperature-resilient Time Synchronization for the Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 14, 2018.

[17] "IEEE Std 802.15.4-2015 - IEEE Standard for Low-Rate Wireless Networks," IEEE Computer Society, 2015.

[18] T. Watteyne, M. Palattella, and L. Grieco, "Using IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH) in the Internet of Things (IoT): Problem Statement," IETF, RFC 7554, 2015.

[19] R. Musaloiu-E., C.-J. M. Liang, and A. Terzis, "Koala: Ultra-Low Power Data Retrieval in Wireless Sensor Networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (ACM/IEEE IPSN)*, 2008.

[20] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. M. Liang, and A. Terzis, "A-MAC: A Versatile and Efficient Receiver-Initiated Link Layer for Low-Power Wireless," *ACM Transactions on Sensor Networks*, vol. 8, 2012.

[21] S. Duquennoy, B. A. Nahas, O. Landsiedel, and T. Watteyne, "Orchestra: Robust Mesh Networks Through Autonomously Scheduled TSCH," in *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, 2015.

[22] J. Muñoz, T. Chang, X. Vilajosana, and T. Watteyne, "Evaluation of IEEE802.15.4g for Environmental Observations," *Sensors*, vol. 18, 2018.

[23] G. Zhou, T. He, S. Krishnamurthy, and J. Stankovic, "Impact of Radio Irregularity on Wireless Sensor Networks," in *Proceedings of The International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2004.

[24] J. Muñoz, X. Vilajosana, and T. Chang, "Problem statement for generalizing 6tisch to multiple phys," 2018, internet-draft, work in progress.

[25] F. Sutton, B. Buchli, J. Beutel, and L. Thiele, "Zippy: On-Demand Network Flooding," in *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, 2015.

[26] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond Duty Cycling: Wake-up Radio with Selective Awakenings for Long-Lived Wireless Sensing Systems," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2015.

[27] A. Kim, J. Han, T. Yu, and D. S. Kim, "Hybrid Wireless Wensor Network for Building Energy Management Systems Based on the 2.4 GHz and 400 MHz bands," *Information Systems*, vol. 48, 2015.

[28] S. Sciancalepore, M. Vucinic, G. Piro, G. Boggia, and T. Watteyne, "Link-layer Security in TSCH Networks: Effect on Slot Duration," *Transactions on Emerging Telecommunications Technologies*, vol. 28, 2017.

[29] R. Tavakoli, M. Nabi, T. Basten, and K. G. W. Goossens, "Hybrid Timeslot Design for IEEE 802.15.4 TSCH to Support Heterogeneous WSNs," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2018.

[30] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An Empirical Study of Low Power Wireless," *ACM Transactions on Sensor Networks*, vol. 6, 2010.