Enhanced Time-Slotted Channel Hopping in WSNs using Non-Intrusive Channel-Quality Estimation

Rasool Tavakoli*, Majid Nabi*, Twan Basten*†, Kees Goossens*
*Department of Electrical Engineering, Eindhoven University of Technology, the Netherlands
†TNO Embedded Systems Innovation, Eindhoven, the Netherlands
Email:{r.tavakoli, m.nabi, a.a.basten, k.g.w.goossens}@tue.nl

Abstract—**Cross-technology interference on the license-free ISM bands has a major negative effect on the performance of Wireless Sensor Networks (WSNs). Channel hopping has been adopted in the Time-Slotted Channel Hopping (TSCH) mode of IEEE 802.15.4e to eliminate blocking of wireless links caused by external interference on some frequency channels. This paper proposes an Enhanced version of the TSCH protocol (ETSCH) which restricts the used channels for hopping to the channels that are measured to be of good quality. The quality of channels is extracted using a new Non-Intrusive Channel-quality Estimation (NICE) technique by performing energy detections in selected idle periods every timeslot. NICE enables ETSCH to follow dynamic interference well, while it does not reduce throughput of the network. It also does not change the protocol, and does not require non-standard hardware. ETSCH uses a small Enhanced Beacon hopping Sequence List (EBSL) to broadcast periodic Enhanced Beacons (EB) in the network to synchronize nodes at the start of timeslots. Experimental results show that ETSCH improves reliability of network communication, compared to basic TSCH and a more advanced mechanism ATSCH. It provides higher packet reception ratios and reduces the maximum length of burst packet losses.**

Keywords-**IEEE 802.15.4e; TSCH; channel hopping; WSN; interference; channel-quality estimation; whitelisting;**

I. INTRODUCTION

IEEE 802.15.4 [1] provides PHY and MAC layer protocols for low data rate Wireless Personal Area Networks (WPANs) as a light-weight, power-efficient and inexpensive solution for Wireless Sensor Networks (WSNs). It operates in the license-free 2.4 GHz ISM band as well as two sub-gigahertz ISM bands. The 2.4 GHz band is further divided into sixteen channels. This band is used by a wide range of wireless technologies including Wi-Fi [2], Bluetooth [3], and also devices such as microwave ovens. This causes non-uniform cross-technology interference on the frequency band and thus different noise conditions on each of the channels of IEEE 802.15.4 PHY at different locations.

Considering stringent requirements of industrial applications such as reliability of communication links, the 4e [4] amendment was developed aiming to increase the robustness and throughput of IEEE 802.15.4 wireless links. Time-Slotted Channel Hopping (TSCH) is one of the MAC operation modes of this amendment which aims to reduce the effect of interference and multi-path fading through guaranteed medium accesses and channel diversity. It eliminates block-
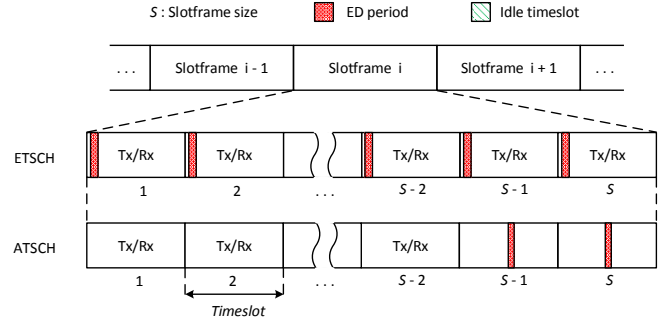

Figure 1. Channel energy detection in ATSCH and ETSCH.

ing of wireless links (repeated dropping of packets due to interference on the operating channel) by hopping over multiple channels. As Fig. 1 depicts, TSCH divides time into fixed time periods called *timeslots*. One timeslot is enough to transmit a single packet and its acknowledgement. A number of timeslots (S) are grouped into a *slotframe* and each link in the network is assigned to one of these timeslots to avoid collisions. Slotframes repeat over time to enable nodes to have periodic access to the network. TSCH uses a global timeslot counter in the network is called the *Absolute Sequence Number* (ASN). By use of the ASN and a global *Hopping Sequence List* (HSL), each node computes the operating channel of each timeslot using Eqn. 1.

$$Channel = HSL[(ASN + Channel\ Offset)\%|HSL|] \qquad (1)$$

$|HSL|$ is the number of channels in the HSL. Different *Channel Offsets* can be assigned to different links in the network to enable parallel communications in one timeslot on multiple channels. The HSL may include all or a subset of 16 channels defined by the IEEE 802.15.4 PHY, to be determined by the upper layers in the protocol stack.

The authors in [5] show that using whitelisting to select the HSL can increase the reliability of TSCH by restricting the used channels to the channels that are measured to be of good quality. When the network conditions vary over time, dynamic whitelisting is required to update the HSL accordingly. Adaptive Time-Slotted Channel Hopping (ATSCH) [6] was proposed to measure quality of the wireless channels at run-time by use of hardware-based Energy Detections (EDs), and then update the HSL periodically. As depicted in Fig. 1, ATSCH reserves two timeslots of each slotframe

to perform one ED per slot on one frequency channel. Using this technique, ATSCH performs EDs on all 16 channels in a period of 8 slotframes. Therefore the quality-estimation sampling period for each channel can be computed as shown in Eqn. 2.

$$T_{sampling}^{ATSCH} = S \times macTsTimeslotLength \times 8 \qquad (2)$$

where *macTsTimeslotLength* is the length of a timeslot. Considering this channel sampling period, for networks with highly dynamic wireless interference (that changes every few seconds), ATSCH may not be sufficiently adaptive. There are many applications in which the interference level and channel conditions are highly dynamic. For instance, in wireless In-Vehicle Networks (IVNs), a network in a moving car may experience dynamic interference from many sources such as Wi-Fi networks along the road. Thus, quality of channels may change before ATSCH uses them to update the HSL.

In this paper we propose an Enhanced version of the TSCH protocol (ETSCH) that uses dynamic channel whitelisting. ETSCH uses a new Non-intrusive Channel-quality Estimation (NICE) technique to measure the quality of IEEE 802.15.4 frequency channels by performing EDs in every timeslot. Fig. 1 shows channel samplings in our technique compared to those of ATSCH. Our ETSCH technique exploits idle time available in 802.15.4 timeslots for EDs. It performs more frequent channel measurement than ATSCH. It does not change the protocol and it does not require non-standard hardware. ETSCH updates HSL frequently and uses a secondary and less dynamic hopping sequence list to broadcast the HSL using periodic *Enhanced Beacons* (EBs). These EBs are defined by IEEE 802.15.4*e* to disseminate setups of the network and synchronize the nodes. These techniques improve reliability of the TSCH protocol by dynamically hopping to less noisy channels as shown by higher Packet Reception Ratios (PRR) and reduced burst packet losses in our experiments.

The paper is organized as follows. The next section gives an overview of related work about channel hopping in WSNs. Section III presents the functionality of our proposed ETSCH mechanism in detail. The NICE technique is introduced in Section IV. Implementations and experimental setups are discussed in Section V. The results are given in Section VI. Section VII concludes.

II. RELATED WORK

The idea of channel hopping is used by a number of protocols and standards including IEEE 802.15.1 Bluetooth [3], WirelessHART [7], ISA100.11a [8], and TSCH which is one of the MAC operation modes of the IEEE 802.15.4e [4] standard. All of these protocols use a time-slotted approach to schedule network communications. At the start of each timeslot, which is synchronized in all network devices, each device hops to a new channel by use of a predefined hopping sequence pattern.

Exploiting a channel hopping technique reduces the probability of not being able to communicate that may occur in single channel communications; it is caused by narrow-band interference and multipath fading [9], [5]. Furthermore, channel hopping improves network performance through higher PRR and better network connectivity compared to a single-channel approach [5].

One enhancement to the channel hopping networks is the whitelisting technique which limits the wireless nodes to only hop between the channels that are known to be of good quality [5]. This technique is used by Bluetooth through defining a subset of channels by a user to be used for hopping. WirelessHART [7] and ISA100.11a [8] are two protocols designed for industrial applications and both use IEEE 802.15.4 radios in the 2.4 GHz ISM band. These protocols also have the option to define a user configurable whitelisting feature at a global scope to control the channel hopping pattern. ISA100.11a has the option to use the history of communications on each link to enable channel whitelisting for that link. The authors of [5] show that use of a static whitelisting can improve performance of a TSCH network in nearly static wireless conditions.

All mentioned user-oriented whitelisting techniques can cope with static wireless conditions. But for wireless environments that experience a level of dynamism [10], [11], [12], these mechanisms may not work well. Also the history-based solution that is exploited by ISA100.11a may not work well, because it needs prior communications on the channel to gain enough knowledge about its condition. To cope with dynamic wireless medium conditions, the Adaptive Times-Slotted Channel Hopping (ATSCH) [6] proposes a dynamic whitelisting/blacklisting mechanism using hardware-based channel energy measurement. This technique works on top of the TSCH protocol and reserves two timeslots of each TSCH slotframe to perform energy samplings (EDs) on the operating channel of these timeslots (Fig. 1). There will be no communications in these timeslots; therefore the gathered values of energy samplings can be considered as noise levels on those channels. These sampling results are used to assign a quality factor to each channel and thus channels can be ranked according to their wireless conditions. A fixed-size subset of the best quality channels is selected periodically as the HSL for the TSCH protocol.

Our proposed ETSCH mechanism also uses EDs to measure the quality of channels and select the best subset of channels as HSL. Although ATSCH and ETSCH both use the same philosophy of channel sampling and whitelisting methods, there are several advances in ETSCH.

1) ATSCH reserves two timeslots of a TSCH slotframe which results in a throughput cost to the network. ETSCH does not use communication parts of timeslots, does not reduce the capacity of the network and requires no change to the TSCH protocol.
2) The rate of sampling in ATSCH is two samples per slot-

frame and is directly affected by the size of slotframes. In contrast, ETSCH introduces the NICE technique to perform energy samplings at least two times per timeslot. It thus has a sampling rate that is at least S times higher than ATSCH. This makes ETSCH perform better in highly dynamic wireless conditions.

3) ATSCH uses all sixteen channels to broadcast EBs (containing the HSL) which may result in EB losses and synchronization loss between nodes. We propose a new method to broadcast EBs in a TSCH network using a secondary hopping sequence whitelist. This technique uses a small and less dynamic hopping sequence list which contains the best quality channels. By using this secondary hopping list for EB transmissions, the probability of EB losses is reduced.

In this paper, wireless IVNs are considered as an example of networks that experience highly dynamic interference. This interference can be caused by in-range Wi-Fi networks along the road, as well as other wireless IVNs working in adjacent cars on the road. These different interference sources work on different channels and each one may be visible for only a short period of time. Therefore, an IVN can experience different interference or noise levels on different parts of the frequency band and at different points in time.

III. ENHANCED TIME-SLOTTED CHANNEL HOPPING

The basic idea of ETSCH is to adaptively select a subset of less noisy channels called *whitelist* and use it as an input for the channel hopping algorithm. ETSCH adds three components to the basic TSCH protocol to do that (Fig. 2). These components are NICE, HSL whitelisting, and EB hopping Sequence List (EBSL) whitelisting.

As Fig. 2 shows, NICE runs in parallel with TSCH on the MAC layer to extract the quality of available channels. To assign a quality factor to each frequency channel of IEEE 802.15.4 on the 2.4 GHz ISM band, we use the EDs introduced in the protocol. An ED is an estimate of the received signal power within the bandwidth of a channel and takes 8 symbol periods (i.e., $128\ \mu s$). The first part of Algorithm 1 shows the process of NICE technique. NICE uses the *silent period* in every timeslot to perform as many EDs on the channels as possible. This silent period is already available based on the timeslot diagram of the TSCH protocol. EDs are performed on successive channels and after 16 EDs, all the channels are sampled. The result of each ED is used to update the assigned Channel-Quality Estimation (CQE) to that channel. The NICE is presented in detail in Section IV.

The output of NICE is used periodically by the two other components which run in the next layer in the protocol stack. These two components configure the TSCH parameters based on the observed wireless condition. Whitelisting is used in these components to select a subset of good quality channels for TSCH hopping procedure. In the following, we present these two components which are the functional parts of the ETSCH.
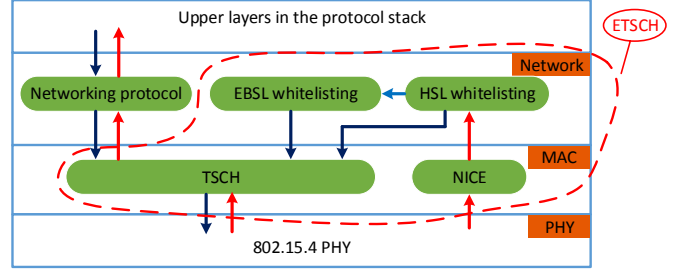

Figure 2.   ETSCH components

Algorithm 1 Procedure of ETSCH components
___
NICE()
Output:
  CQE[]: Channel Quality Estimation results of all channels

  every timeslot **do**
    while it is the *silent period* **do**
      $ch \leftarrow (ch+1)\%16$
      $quality \leftarrow$ ED (ch)
      Update CQE[ch] using measured *quality*
    end while
  end every

___
HSL_whitelisting(CQE)
Output:
  HSL[]: main Hopping Sequence List

  every whitelisting period **do**
    $HSL_{sorted} \leftarrow$ Sort channels in decreasing CQE[]
    HSL $\leftarrow HSL_{sorted}$[1 to sizeof(HSL)]
  end every

___
EBSL_whitelisting(EBSL,HSL,k)
Output:
  EBSL[]: Enhanced Beacon hopping Sequence List

  every whitelisting period **do**
    if EBSL[k] $\notin$ HSL[0 to 3] $\wedge$ EBSL[k] $\neq$ 26 **then**
      $m = \min\{h \mid 0 \le h \le 3 \ \wedge \ $HSL[$h$] $\notin$ EBSL$\}$
      EBSL[k] $\leftarrow$ HSL[m]
    else
      // No update is needed
    end if
  end every

A. Channel Hopping Whitelisting

Whitelisting is performed periodically to select a subset of good quality channels as the HSL for TSCH protocol. There are two approaches to do whitelisting; 1) selecting all channels with a better channel quality than a specified threshold, and 2) selecting a fixed-size subset of channels with the best qualities. The first method can improve the PRR by defining quality thresholds but it cannot guarantee a minimum size of the HSL and thus it may decrease the

number of possible parallel slotframes and throughput of the network. The fixed-size method can guarantee a minimum size HSL, but it may include low quality channels in high interference conditions. For simplicity of presentation, we use the fixed-size whitelisting in describing our ETSCH and later in the experimental evaluation. Our enhancements can be straightforwardly integrated in the other variant though. As the second part of Algorithm 1 shows, every whitelisting-period, the channels are sorted based on their assigned qualities and then a fixed-size subset is selected as HSL. This HSL is used by the TSCH protocol for the hopping procedure. This fixed-size whitelisting technique is also used by the ATSCH to update the HSL periodically.

B. EB Whitelisting

As mentioned, the coordinator device of the ETSCH network periodically uses whitelisting to extract the best HSL with a predefined size. HSL and other information of the network such as link allocations and ASN are disseminated via the EBs defined in the TSCH protocol, in line with e.g. [4]. Same as ATSCH, we setup the coordinator to broadcast EBs periodically in the first timeslot of the slotframe with highest priority. Thus, all devices in the network can be aware of changes in the network setups at the start of each slotframe. Periodic transmission of EBs helps all devices in the network to synchronize with their coordinator at the start of each slotframe.

When a coordinator broadcasts an EB with an updated HSL, there is the possibility of missing this EB in one or more devices. Using unicast and ACK-enabled communications for transmitting EBs comes at a throughput cost to the network. The work in [5] shows that some of IEEE 802.15.4 channels are affected less than others by coexisting Wi-Fi networks which are the main source of interference on IEEE 802.15.4 channels (i.e., channels 15, 20, 25, and 26). Thus we decide to use a second, less dynamic, hopping sequence list consisting of a small subset of channels with the best qualities to disseminate EBs in ETSCH. The EBSL is defined by the coordinator and has a fixed size of 4. Therefore, the operating channel to transmit an EB for a given ASN and size of the slotframe (S) can be computed as:

$$Channel(EB) = EBSL[\lfloor ASN/S \rfloor \% \ 4] \qquad (3)$$

We update this EBSL in a one-channel-per-period manner every time the main HSL is updated. In this method, every time the coordinator wants to broadcast an EB containing an update of the main HSL, it updates the EBSL entry which was used for the last EB transmission (k). The process of updating this list is described in the last part of Algorithm 1. This algorithm finds the best quality channel which is not in the EBSL, then puts this channel in the last used entry of the EBSL. This updating method reduces the possibility of burst EB losses in a joined device by only using best quality channels. Hence, when a device misses an EB which contains

an updated HSL, it has a high chance to receive it in the later slotframes and synchronize its HSL to the network.

Timeslot phase difference caused by clock drift between a device and the coordinator can lead to disconnection of the link between them. This leads to burst EB losses even when the EBSL is the same at both. To solve this problem, we take channel 26, which is a non-overlapping channel with Wi-Fi, as a permanent member of EBSL. This channel is considered to be the less noisy channel in urban environments. Every time a joined device experiences a burst EB loss equal to N_{BL}, it considers this situation as a synchronization loss caused by timeslot phase difference and starts a passive scan on channel 26 to be synchronized again with its coordinator.

IV. NON-INTRUSIVE CHANNEL-QUALITY ESTIMATION

To perform an ED in a frequency channel to estimate its noise level, there should be no transmissions in the network during that measurement. We propose NICE to perform the EDs on different frequency channels without any bandwidth cost to the protocol. In the following, we first look at the TSCH communication diagram and then explain our NICE technique.

The TSCH technique of IEEE 802.15.4e uses synchronized timeslots to establish communication between pairs of nodes. A receiver node should be aware of the start of the sender's timeslot to turn on its radio and listen to the medium before transmission starts. Because of clock drift between the nodes, the synchronization process needs to be continuously performed in order to keep the nodes synchronized. To compensate an amount of timeslot phase differences caused by clock drifts, TSCH defines a diagram for timeslots shown in Fig. 3. The timeslot duration, *macTsTimeslotLength*, is long enough for transmission of a maximum size packet and its ACK. There is an offset at the beginning of a receiver's timeslot before it starts listening to the medium. This *Rx* offset prevents interference from other nodes in the network which are behind for a maximum *macTsRxOffset* and still are transmitting in the previous timeslot. The packet *Tx* offset (*macTsTxOffset*) in a sender is defined with a value greater than the *Rx* offset to make the communication possible when the sender is ahead of the receiver for a *macTsTxOffset* − *macTsRxOffset* period of time. A *macTsCCAOffset* offset is defined for a sender to perform a Clear Channel Assessment before each *Tx* and prevent packet transmission if the channel is busy. When a receiver starts to listen to the medium for a packet reception in a timeslot, it waits for a *macTsRxWait* period of time to receive the packet. If the transceiver cannot detect any packet preamble in this period, the receiver considers this situation as a packet transmission failure and stops listening. The values of these parameters are defined such a way so that *macTsRxOffset*+*macTsRxWait* is greater than *macTsTxOffset*. Thus, the communication can be successful if the receiver is ahead of the sender for maximum the difference of these
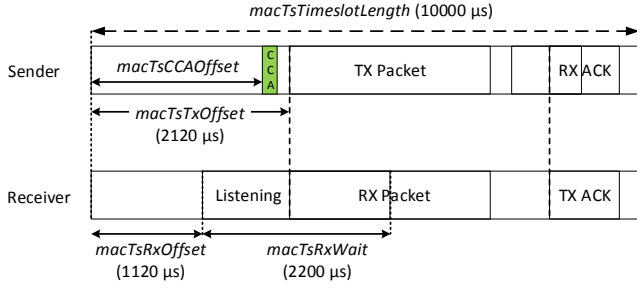
Figure 3. The structure of transmit and receive timeslots in IEEE 802.15.4e TSCH mode.

two values. Some other timings such as *Tx/Rx* durations and *ACK* transmission timings are defined in the protocol but not shown in Fig. 3.

To extract the maximum allowed phase difference for default values of the mentioned offsets, we investigate different cases. As illustrated in Fig. 4(a), if a receiver starts its timeslot $T_{forward} = 1200\ \mu s$ earlier that the coordinator, it still can receive the packet from the coordinator. Also if a receiver starts its timeslot $T_{backward} = 1000\ \mu s$ later than the coordinator, the two nodes can still communicate (Fig. 4(b)). Considering the fact that the coordinator of a WPAN is the main source of synchronization, there is no chance for a device that starts $T_{forward} = 1200\ \mu s$ before the coordinator to communicate with a device that starts $T_{backward} = 1000\ \mu s$ after the coordinator (Fig. 4(c)). To enable bidirectional transmission between each pair of nodes in the network, as shown in Fig. 4(d) and (e), the maximum forward and backward timeslot phase differences should be $T_{backward}^{Max} = T_{forward}^{Max} = 500\ \mu s$.

Each device may start its timeslot at maximum $T_{forward}^{Max}$ earlier or $T_{backward}^{Max}$ later than the coordinator. Therefore, from the coordinator perspective, for a $T_{backward}^{Max}$ time period at the start of each timeslot, there is the possibility of packet transmissions by some nodes in the previous timeslot. Also for a $T_{forward}^{Max}$ time period before *macTsTxOffset*, there is the possibility that some nodes start packet transmissions ahead of the coordinator. Considering these possibilities, there will be no packet transmissions in the network for a T_{silent} period (Fig. 5) that can be computed by Eqn. 4.

$$T_{silent} = macTsTxOffset - T_{backward}^{Max} - T_{forward}^{Max} \qquad (4)$$

For the timing defaults of the TSCH protocol, this value will be $T_{silent} = 1120\ \mu s$. Because of the allowed timeslot phase difference between network devices, they cannot have an exact estimation about the start of transmissions in other devices. The coordinator device is an exception here because it is the source of synchronization for its joined devices. To perform an ED in a network and have an estimation about the noise level of the channels, there should be no transmissions in the network during these measurements. We use this silent period in each timeslot to perform EDs on the coordinator device.
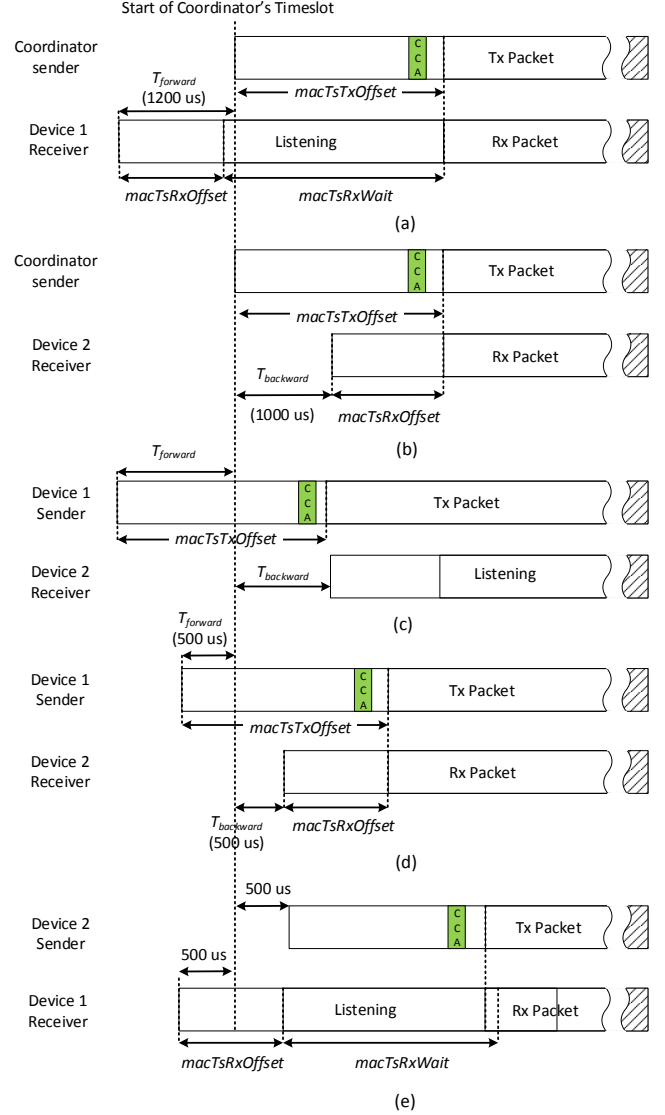

Figure 4. Pair-wise communications in the case of timeslot phase difference; (a) Device 1 starts $T_{forward} = 1200\ \mu s$ ahead the coordinator, (b) Device 2 starts $T_{backward} = 1000\ \mu s$ later than the coordinator, (c) Communication of device 1 with $T_{forward} = 1200\ \mu s$ and device 2 with $T_{backward} = 1000\ \mu s$ fails, (d,e) Successful communication of devices 1 and 2 with $T_{forward} = T_{backward} = 500\ \mu s$.

A wireless device can be receiver, transmitter or an idle node during a timeslot. According to this, EDs during the silent period in the coordinator device can be divided into three types. If the coordinator is a receiver, it should finish the ED process within the *macTsRxOffset* period. The overlap of this period and the silent period can be used for the ED process. When the coordinator is a transmitter, this period will be the overlap of *macTsCCAOffset* and the silent period. The whole silent period can be used for performing EDs when the coordinator has no *Rx/Tx* task. Fig. 5 shows these Silent Energy Detection (SED) periods. The available ED duration for each type of timeslot transmission can be computed as
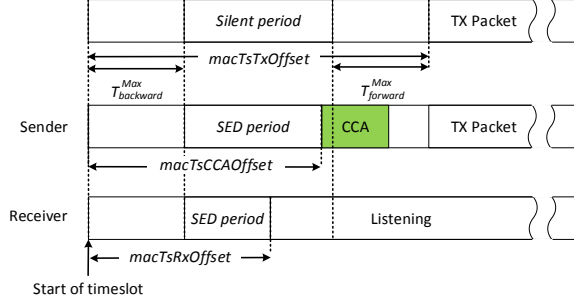
Figure 5. Available time for Silent Energy Detection (SED) when coordinator is a sender or a receiver

Eqn. 5, 6, and 7.

$$T_{SED}^{Rx} = \max(macTsRxOffset, \\ macTsTxOffset - T_{forward}^{Max}) - T_{backward}^{Max} \quad (5)$$

$$T_{SED}^{Tx} = \max(macTsCCAOffset, \\ macTsTxOffset - T_{forward}^{Max}) - T_{backward}^{Max} \quad (6)$$

$$T_{SED}^{Idle} = T_{silent} \quad (7)$$

According to the TSCH protocol defaults, these ED periods will be $T_{SED}^{Rx} = 620~\mu s$, $T_{SED}^{Tx} = T_{SED}^{Idle} = 1120~\mu s$. As mentioned before, each ED takes 8 symbol periods and is the mean of 128 consequent measurements of the signal strength, each lasting for $1~\mu s$. To hop to the desired channel for performing ED and also get the ED measurements from hardware and performing the quality estimation evaluations, we assume this time to be more than twice as high, namely $T_{ED} = 300~\mu s$ (the value observed in our experiments is less). Therefore, a coordinator can perform two EDs in receiving timeslots and three EDs in transmitting and idle timeslots. Each ED will be done in the channel next to the channel for which the prior ED was done. Hence, by each 16 consecutive EDs, the whole channel space is scanned and the results can be used to assign quality factors to channels.

Eqn. 8 gives the sampling period of each channel in our technique.

$$T_{sampling}^{ETSCH} = \frac{macTsTimeslotLength \times 16}{Mean(\# \text{ of EDs per timeslot})} \quad (8)$$

Considering the least number of possible EDs per timeslot to be two, the maximum sampling period for each channel will be eight timeslots, which is independent of the slotframe size. Compared to Eqn. 2 which gives the sampling period of the ATSCH algorithm [6], the sampling rate of our proposed technique is S times greater than that in ATSCH. Thus, with a slotframe size of 12, ATSCH performs one ED per second in each channel, while our proposed technique performs at least 12 EDs in each channel during the same period. Furthermore, ATSCH has a throughput cost for the network because of the reserved timeslots, while our NICE technique imposes no throughput cost to the network. That is why we call our proposed channel quality estimation technique non-intrusive.

Results of the EDs are used to assign CQE values to each of the channels. To compute a stable estimate of the channel quality, as the ED measurements may fluctuate, we use an exponential smoothing technique [13]. This technique is also used by the ATSCH to compute CQE values. Every time a new ED is done in a channel, a new CQE is calculated for that channel using Eqn. 9.

$$CQE_\tau(ch) = \alpha~ED_\tau(ch) + (1 - \alpha)CQE_{\tau-1}(ch) \quad (9)$$

where $ED_\tau(ch)$ is the new ED sample of channel ch and $CQE_{\tau-1}(ch)$ is the last computed CQE for that channel. Coefficient α, with $0 \le \alpha \le 1$, is the smoothing coefficient that controls the effect of new ED samples on the CQE. By selecting small values of α, we can have more stable estimation of channel quality values.

V. EXPERIMENTS

A. Experimental Setup

To evaluate the performance of ETSCH, we implemented TSCH protocol on the Atmel ATMEGA256RFR2 Xplained Pro kit [14]. This kit includes an ATmega256RFR2 chip which integrates an AVR microcontroller and a 2.4 GHz RF transceiver. Our implementation follows default timings defined in the standard.

We use a mesh TSCH network with seven devices and one PAN coordinator. Motes are distributed in random places in a $10~m \times 10~m$ office workspace. The transmission power of the motes is 0 dBm. We ran the experiments using two different versions of our proposed mechanism; a full ETSCH version, and a version (ETSCH–) without the EBSL, which uses the basic hopping sequence list (all 16 channels) to transmit EBs. This allows us to investigate the impact of EBSL on the performance of the network.

Slotframes of size $S = 8$ are used in the experiments. The first timeslot is allocated to EB transmission by the coordinator and each of the other 7 timeslots is allocated to one of the devices to transmit a packet of 100 bytes. Each experiment lasts for 6000 slotframes, thus each mote broadcasts 6000 packets in an experiment. All nodes listen to all the timeslots for packet reception from other motes.

We also compare the performance of ATSCH [6] to ETSCH. Furthermore, we use an Advanced version of ATSCH (AATSCH) that performs a full channel space ED in each allocated timeslot instead of only one channel. Thus, AATSCH has an ED rate 16 times greater than that of ATSCH, but it imposes a higher throughput cost to the TSCH protocol, because there should be no transmission in all the channels on the allocated timeslots. As ATSCH and AATSCH need two more timeslots per slotframe to perform EDs, we use slotframes of size 10 for them.

For all of the mentioned mechanisms we set the exponential smoothing coefficient (α) to 0.1. Each mechanism uses a HSL of size 8 and updates it once every 160 EDs (10 EDs per channel). N_{BL} is set to 5 for all the experiments.
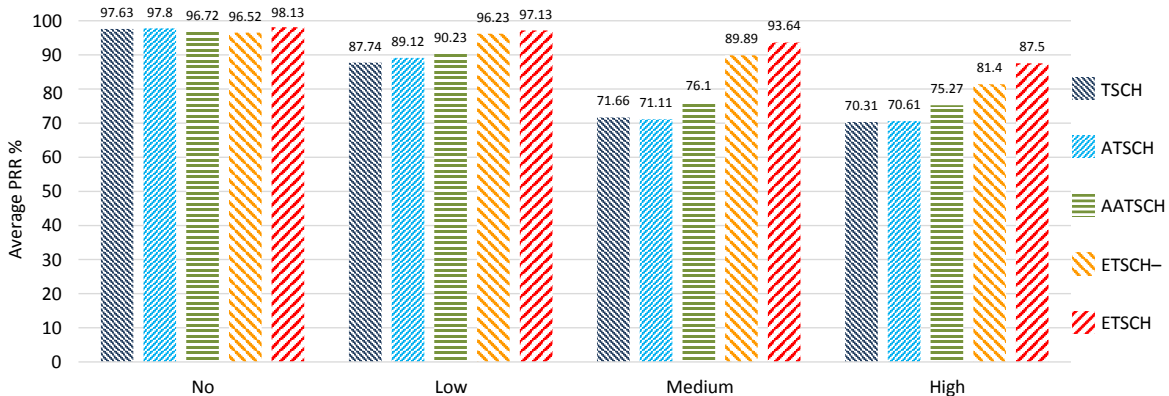
Figure 6. Average achieved PRR of different mechanisms for different interference scenarios

B. Interference Scenarios

Co-channel wireless interference is the main source of packet errors in urban networks. Whitelisting can reduce the negative effect of this interference on the network by using good quality channels. The level and also dynamism of the interference can affect the performance of the whitelisting technique. To test the performance of our proposed mechanism, we try different interference scenarios with different levels and dynamism of interference for the network under test. The level of interference from other sources in our experiment environment was negligible compared to the used transmission powers. Thus, we do not consider the effect of them in our experiments.

We exploit some controlled noise generators using Atmel motes. Each noise generator provides controlled interference by transmitting dummy packets on a pair of adjacent IEEE 802.15.4 channels at a transmit power of 4 dBm. Using this mechanism, a noise generator transmits a short packet on a channel and immediately hops to the paired adjacent channel. This process is done continuously to generate interference on both the paired adjacent channels. Furthermore, each noise generator can be programmed to hop to different pairs of adjacent channels within predefined periods and using a predefined sequence.

We consider four interference scenarios in our experiments; high, medium, low, and no interference scenarios. Table I provides a short description of the behavior of each scenario. In the no interference scenario, we run the experiments without any controlled noise generator to see the cost of periodic HSL changes on the performance of our mechanism. For the low interference scenario, one noise generator hops every 20 seconds on a different pair of adjacent channels to provide an interference with a low level and also a low dynamism. The medium interference scenario provides a higher level of interference (using three noise generators) with a low dynamism. In the high interference scenario, a high level and also a high dynamism of interference is generated.

Considering wireless IVNs as a case study, there are some

Table I
INTERFERENCE SCENARIOS

Scenario	Behavior of noise generator(s)
No	no controlled noise generator
Low	one noise generator, hops every 20 seconds
Medium	3 noise generators, hop every 20 seconds
High	3 noise generators, hop every 5 seconds

interference sources visible in any position along the urban roads. An in-vehicle network in a moving car constantly experiences interference from different sources (e.g., Wi-Fi networks). If each interference source be visible over a range of up to 50 meters, and this car moves with a speed of 36 km per hour, each noise source would be visible for 5 seconds. Our high interference scenario models this kind of interference when there are three interference sources visible at any time. In this scenario, the car moves into the range of a new interference sources (e.g., a Wi-Fi network) every 5 seconds. For the medium interference scenario, we consider lower mobility of IVNs which leads to increasing the visibility duration of each interference source. Thus, we have lower dynamism of interference in this scenario.

VI. PERFORMANCE ANALYSIS

We investigated various metrics to evaluate the performance of our proposed techniques. Packet Reception Ratio (PRR) is the percentage of packets that are successfully received at the receiver node over the total number of packets transmitted by the sender node. It reflects the quality of the links. Fig. 6 shows the average of achieved PRR of all links in the network for different mechanisms and interference scenarios. Both versions of ETSCH provide better PRR on average in comparison with other mechanisms when network experiences a dynamic interference. This is because it uses a highly adaptive quality estimation technique, which selects the best quality channels for hopping. As depicted in Fig. 6, ATSCH performs almost the same as basic TSCH when network conditions change. It shows that ATSCH cannot detect and follow the dynamic interference well. This leads to increasing packet losses when noisy channels are also
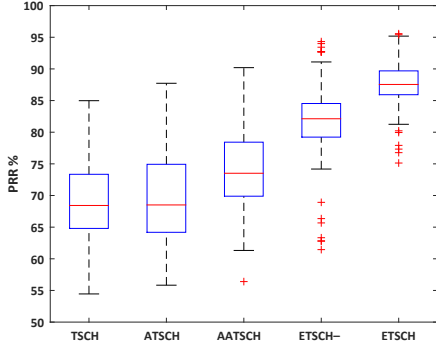
Figure 7. Distribution of average PRR of network links for different mechanisms in the high interference scenario.

selected to be used in the HSL. On the other hand, TSCH hops over all channels and thus its PRR has a direct relation with the number of noisy channels. The next observation of Fig. 6 is that AATSCH performs better than ATSCH in existence of interference. This is because the sampling rate of AATSCH is 16 times more than ATSCH, which leads to a better estimation of the channels' quality. Although AATSCH and ETSCH– have almost the same sampling rates, AATSCH has a lower PRR than that of ETSCH– in medium and high interference scenarios. This is because ETSCH spreads ED over a slotframe and therefore detects noisy channels better.

Fig. 6 shows that using EBSL to disseminate EBs improves the reliability of data transmission because it prevents HSL mismatch between the coordinator and nodes. In the no interference scenario, the effect of HSL mismatch is more observable. For AATSCH and ETSCH– in this scenario, there is a small reduction in the achieved PRR compared to that of ATSCH. This is because the rate of HSL updates in these two mechanisms is higher than that of ATSCH, which leads to more HSL mismatches is caused by EB losses. This is while ETSCH outperforms other mechanisms by reducing the rate of EB losses. In the other scenarios, more adaptive CQE leads to higher PRR of AATSCH and ETSCH– compared to ATSCH.

To better investigate the behavior of different mechanisms, Fig. 7 shows the distribution of average PRR of all links in the network for different mechanisms in the experiments with the high interference scenario. The figure shows that both versions of ETSCH provide better PRR than other mechanisms for all links of the network. Furthermore, the results show that using a secondary hopping sequence list to transmit EBs reduces HSL mismatch between coordinator and joined devices and thus improves the PRR of all network links with a lower standard deviation.

Length of burst packet losses is the number of consecutive packets lost over a link. This metric is used to report the distribution of packet losses over a link, since the PRR does not reflect it. Fig. 8 illustrates the distribution of maximum length of burst packet losses for the links between the coordinator and other devices. Fig. 8(a) shows the measured results for the no interference scenario. For AATSCH and ETSCH–

mechanisms, the maximum length of the burst packet losses increases compared to basic TSCH. The reason is that more frequent HSL updates in these two mechanisms lead to a higher chance of HSL mismatch in the network. ETSCH alleviates this negative effect and has the same distribution of this metric as ATSCH which performs HSL updates at a lower rate.

For the second interference scenario (Fig. 8(b)), basic TSCH has burst packet losses with a length of two for most of the links. This is because TSCH hops over all channels and there are only two adjacent channels with interference which cause packet losses. ATSCH also has almost the same results as TSCH because when the noise generator hops to a new pair of adjacent channels which are also in the HSL, ATSCH needs some time to detect these new noisy channels. This delay in updating HSL causes some packets to be lost. Accordingly, AATSCH and ETSCH can follow the dynamism of interference faster and thus they experience smaller burst packet losses in most of the links.

Fig. 8(c) shows that when the number of noise generators increases to three (6 noisy channels), TSCH and ATSCH experience a wider distribution of maximum length of burst packet losses. This is due to timeslot synchronization loss between nodes and the coordinator when five consecutive EB losses occur and nodes start to resynchronize themselves with the coordinator. Moreover, when there are more noisy channels, the burst packet losses increase and most of the links in the network experience a higher maximum length of burst packet losses. This is while AATSCH and ETSCH– decrease burst packet losses by selecting HSL more precisely. ETSCH even has lower and less deviation of the maximum length of burst packet losses, because it reduces EB losses which are the source of HSL mismatch between nodes.

For high interference scenario (Fig. 8(d)), all techniques perform better than the TSCH. Moreover, as it is expected, ETSCH decreases the maximum length of burst packet losses more than other mechanisms by keeping the network nodes synchronized using best channels to transmit EBs.

As our NICE technique only is used on the coordinator of the network and energy may not be a stringent constraint for the coordinator, energy consumption is not a crucial metric in our work. However, to have a comprehensive comparison between our ETSCH and other mechanisms, we analyse energy consumption. The expected transmission rate for successful packet transmission over a link is equal to $1/\text{PRR}$. This metric is used to consider the energy that is consumed for packet (re)transmissions to deliver all packets to their destinations. Thus, the energy consumption (E) of each mechanism is

$$E = \left[I_{ED} N_{ED} T_{ED} + \frac{1}{\text{PRR}} \left[(I_{Rx} N_{Rx} T_{Tx}) + (I_{Tx} N_{Tx} T_{Tx}) \right] \right] \times V_{cc} \tag{10}$$

where I_{ED}, I_{Rx}, and I_{Tx} stand for the radio transceiver current in energy detection, receive, and transmit modes, respectively. V_{cc} represents the operation voltage of the transceiver. T_{ED}
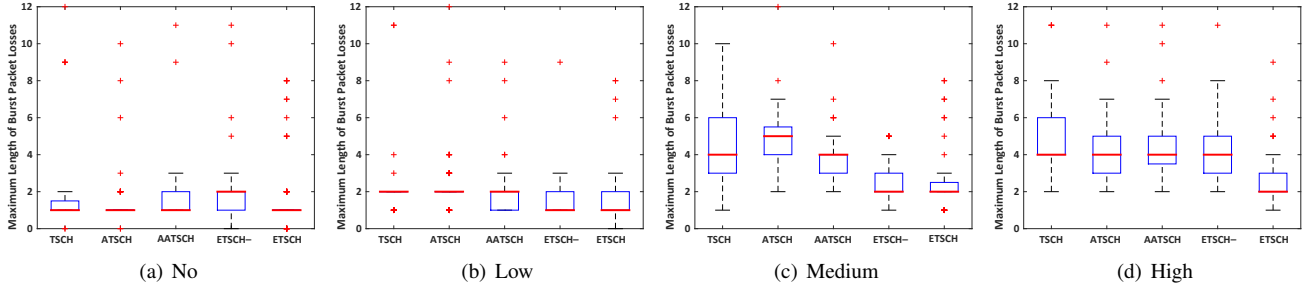
Figure 8. Maximum length of burst packet losses measured at coordinator for different interference scenarios.

and T_{Tx} represent the duration of one energy detection and a full packet transmission/reception, respectively. N_{ED}, N_{Rx}, and N_{Tx} reflect the number of each operation in the experiment duration. Based on the ATmega256RFR2 datasheet, $I_{ED} = I_{Rx} = 5$ mA, $I_{Tx} = 10$ mA, and V_{cc} is 3.3 V. Also T_{ED} is 128 μs and transmission time of each byte is $T_{Tx} = 32$ μs. As ETSCH is proposed for highly dynamic interference conditions, here we use the extracted PRR values for high interference scenario.

Considering the period of one slotframe in our experiments (8 timeslots), coordinator sends an EB with a length of 100 bytes and listens for 7 packets with the same size. With a maximum of 3 EDs per timeslot in NICE, the consumed energy by ETSCH in a slotframe will be $E = 593.77$ μJ at the coordinator. For TSCH that does not perform any EDs but has a lower PRR, this value will be equal to $E = 675.86$ μJ. ATSCH even has a more energy consumption than TSCH, because it performs some EDs but has a PRR same as TSCH. It is clear that ETSCH outperforms other mechanisms from the point of energy consumption when the consumed energy for packet retransmissions is taken into account. Furthermore, because of PRR improvement in ETSCH, the energy consumption of other devices in the network is reduced too.

VII. CONCLUSION

This paper proposed Enhanced Time-Slotted Channel Hopping (ETSCH), a mechanism on top of the TSCH protocol that uses a Non-Intrusive Channel-quality Estimation (NICE) technique and an Enhanced Beacon hopping Sequence List (EBSL) to improve the performance of TSCH protocol. NICE measures the spectrum energy in the idle part of timeslot offset when all the nodes of network are silent. The energy sampling results are used to assign qualities to wireless channels. These assigned qualities are used by coordinator to adaptively select the subset of best channels for hopping purposes. ETSCH uses a small secondary hopping sequence list (EBSL) that consists of best quality channels to disseminate periodic Enhanced Beacons (EBs). These EBs contain control information of the network such as the Hopping Sequence List (HSL). Only one field of EBSL is updated per period, thus the rate of EB losses in the network is reduced. Experimental results show that for high interference scenario, use of NICE and EBSL provides 24% higher packet

reception ratios and 50% shorter length of burst packet losses for ETSCH compared to TSCH protocol. The results also show that ETSCH outperforms the ATSCH technique while it imposes no bandwidth cost to the network.

REFERENCES

[1] "IEEE Standard - Part 15.4: Wireless MAC and PHY Specifications for Low Rate Wireless Personal Area Networks (WPANs)," *IEEE Std 802.15.4-2006*.

[2] "IEEE Standard - Part 11: Wireless LAN MAC and PHY Specifications," *IEEE Std 802.11-2007*.

[3] "IEEE Standard - Part 15.1: MAC and PHY Specifications for Wireless Personal Area Networks (WPANs)," *IEEE Std 802.15.1-2005*.

[4] "IEEE Standard -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)*.

[5] T. Watteyne, A. Mehta *et al.*, "Reliability Through Frequency Diversity: Why Channel Hopping Makes Sense," in *PE-WASUN*. ACM, 2009.

[6] P. Du and G. Roussos, "Adaptive time slotted channel hopping for wireless sensor networks," in *CEEC*, 2012.

[7] *HART Field Communication Protocol Specifications, Revision 7.1, HART Communication Foundation Std.*, 2008.

[8] *ISA, ISA100.11a:2009 Wireless systems for industrial automation: Process control and related applications, International Society of Automation Std.*, 2009.

[9] K. Pister and L. Doherty, "TSMP: Time synchronized mesh protocol," *IASTED Distributed Sensor Networks*, 2008.

[10] K. Srinivasan, M. A. Kazandjieva *et al.*, "The beta-factor: Measuring wireless link burstiness," in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*. ACM, 2008.

[11] S. Hanna and J. Sydor, "Distributed sensing of spectrum occupancy and interference in outdoor 2.4 GHz Wi-Fi networks," in *GLOBECOM*, 2012.

[12] S. Pollin, M. Ergen *et al.*, "Distributed cognitive coexistence of 802.15.4 with 802.11," in *Cognitive Radio Oriented Wireless Networks and Communications*, 2006.

[13] E. S. Gardner and D. G. Dannenbring, "Forecasting with exponential smoothing: Some guidelines for model selection," *Decision Sciences*, 1980.

[14] *ATmega256RFR2 Xplained Pro Evaluation Kit*. [Online]. Available: http://www.atmel.com/tools/ATMEGA256RFR2-XPRO.aspx