Elsts, A., Fafoutis, X., Piechocki, R., & Craddock, I. (2017). Adaptive channel selection in IEEE 802.15.4 TSCH networks. In *Global Internet of Things Summit (GIoTS), 2017: Proceedings of a meeting held 6-9 June 2017, Geneva, Switzerland* (pp. 207-212). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/GIOTS.2017.8016246

Adaptive Channel Selection in IEEE 802.15.4 TSCH Networks

Atis Elsts, Xenofon Fafoutis, Robert Piechocki, and Ian Craddock
Faculty of Engineering, University of Bristol

***Abstract*—Time-Slotted Channel Hopping (TSCH) from the IEEE 802.15.4-2015 standard uses channel hopping to combat interference and frequency-selective fading. Due to the pseudo-random nature of TSCH standard channel hopping process, the energy usage and end-to-end delay achieved in statically configured TSCH networks are suboptimal when compared to using a dynamically updated set of channels. We investigate and compare the advantages of several different adaptive channel selection metrics and methods under the presence of external, frequency-specific interference. In our experiments, PRR-based channel quality assessment with downstream-driven channel selection shows the best results. It is able to reliably distinguish between heavily-interfered and lightly-interfered channels, reduces the number of packet retransmissions up to 2.7 times, achieves up to 22 % lower average radio-on time, and shows close-to 100 % PDR even under heavy interference.**

I. Introduction

The Time-Slotted Channel Hopping (TSCH) protocol, specified in the IEEE 802.15.4-2015 standard [1], is a low-power TDMA MAC protocol for the wireless Internet of Things (IoT). TSCH makes use of pseudorandom channel hopping to combat external interference and frequency-selective multipath fading. The core observation here is that several major causes of bad performance, such as WiFi interference or deep fading are very unlikely to equally affect all of IEEE 802.15.4 channels. Since a retransmission of an unacknowledged packet in TSCH with high probability takes place on a different channel, having just a few low-quality channels do not render the whole network unusable. Nevertheless, the performance of a TSCH network can be improved by selectively avoiding low-quality channels [2] and therefore reducing the number of packet retransmissions. This channel selection should be continuously updated during the runtime of the network based on up-to-date channel quality metrics, as the wireless medium is notoriously dynamic.

In a TSCH network, a node uses three inputs to decide which channel to use for a particular transmission: the network's absolute sequence number (ASN), the scheduled channel offset, and the pseudorandom frequency hopping-sequence (HS), stored as a lookup table:

$$channel = HS[\ (ASN + \text{channelOffset})\ \%\ \|HS\|\ ] \quad (1)$$

The standard TSCH can be extended to avoid transmissions on low-quality channels: either the hopping sequence itself may be changed on the network nodes, or the nodes can pseudorandomly remap the set of low-quality channels to a different set of channels every time a transmission is done. This opportunity

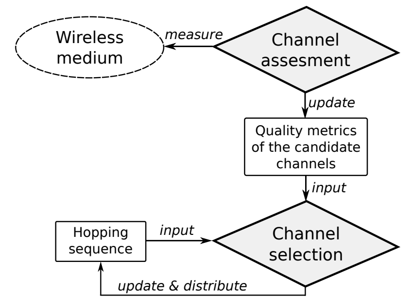


Fig. 1: Overview of the adaptive channel selection process.

was recognized by the authors of the Bluetooth standard, who included Adaptive Frequency Hopping (AFH) technique in its specification [3]. In contrast, only a few have tried applying similar techniques to TSCH networks. The existing work [4] [5] [6] shows that *Adaptive Channel Selection* (ACS) for TSCH reduces the number of retransmissions, but the authors did not quantify the energy usage of their methods.

In this paper we quantify the effect of ACS on energy usage (radio duty cycle), end-to-end packet delivery rate (PDR), and link-layer packet error rate (PER). We implement a complete ACS system (Fig. 1) for TSCH, consisting of a *channel-assessment* module and a *channel-selection* module. Our contributions are:

- we investigate a packet reception rate (PRR) based channel quality metric for channel assessment (Section II) in TSCH, and show that in some situations it outperforms the frequently used [4] [5] RSSI-based metric for clear-channel assessment in TSCH;
- we present analytic methodology for optimal parameter selection for our channel assessment metrics (Section II-C);
- we generalize the previously proposed TSCH adaptive channel selection algorithms [4] [5] [6] to two main options: upstream- and downstream-driven selection (Section III);
- we evaluate PDR, PER and radio duty cycle requirements of these two channel selection algorithms and these two assessment metrics, *i.e.*, RSSI and PRR (Section IV).

The results are evaluated using a network simulator, and validated in a testbed with Texas Instruments CC2650 system-on-chip based sensor nodes. The results show that downstream-driven selection using the PRR metric achieves the best results in most of our scenarios; it operates with negligible control-plane overhead while achieving up to 2.7 times lower packet error rate and up to 22 % lower average radio duty cycle compared to an unmodified version of TSCH, as well as close-to 100 % PDR in all test scenarios.

II. Channel assessment

A. Background

We investigate two well-known channel quality assessment metrics: packet reception rate (PRR), measured using existing data traffic, and the wireless medium noise levels during periods when no transmissions are expected, measured through periodic RSSI sampling.

The PRR metric can be estimated on the transmitter nodes through comparing the number of transmitted packets with the number of acknowledged packets. It is a passive metric, as it does not require additional traffic or measurement effort. It can be evaluated for each neighbor individually, therefore is suitable for networks with very localized interference; beside interference, it can also detect frequency-selective deep fading.

The noise-level RSSI metric requires additional energy to perform the periodic sampling, and cannot be computed for each neighbor separately. The minimal time required for a RSSI measurement is $128\,\mu$s according to the IEEE 802.15.4 standard; however, radio wakeup and frequency calibration takes additional, hardware-specific amount of time. For example, the time required to turn the radio on an perform a RSSI measurement is at least $320\,\mu$s on CC2420. However, this is still an order-of-magnitude shorter than the time required to transmit a packet. Also, RSSI measurements can be done on all channels, including currently unused channels.

B. Approximating the quality of a channel

We select the Exponentially-Weighted Moving Average (EWMA) filter to incorporate history information in the RSSI and PRR metrics. An EWMA channel quality estimator is defined as follows:

$$p_{n+1} = (1-\alpha)p_n + \alpha Y_n \ , \tag{2}$$

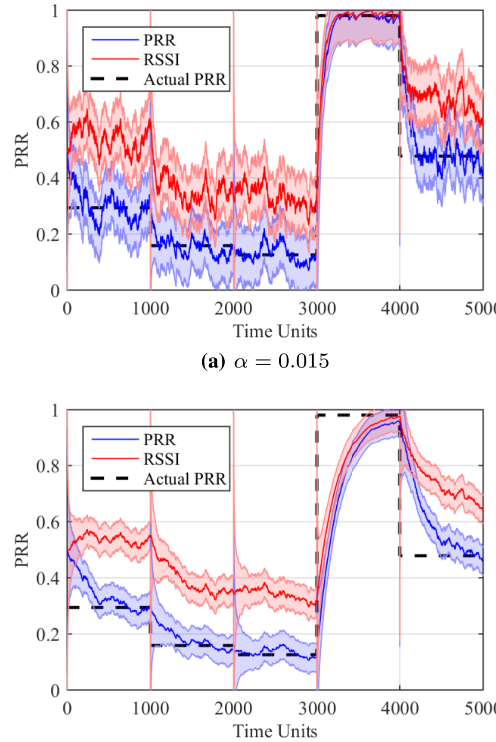
where $Y_n \in \{0, 1\}$ depending on whether the n-th transmission was successful or the n-th RSSI was below a certain threshold, respectively. This filter is well-suited for embedded devices, as it requires only $O(1)$ memory, and its sensitivity can be tuned with its α parameter (Fig. 2).

Assuming that the external interference follows a Poisson process, the inter-arrival time between interfering packets is exponentially distributed with a rate λ. The actual collision probability, P_c is given by [7]:

$$P_c = e^{-(\tau+\tau_i)\lambda} \ , \tag{3}$$

where τ and τ_i is the time the node and the interferer occupy the channel respectively. Estimating the quality of the channel using the PRR metric would eventually converge to P_c. The RSSI metric, on the other hand, samples the channel for a shorter period of time, τ_{cca}. This is equivalent to attempting to transmit shorter packets that occupy the channel for τ_{cca}. Therefore, using the RSSI metric, the channel quality is now approximated by:

$$P_{cca} = e^{-(\tau_{cca}+\tau_i)\lambda} \ . \tag{4}$$



(a) $\alpha = 0.015$

(b) $\alpha = 0.004$

Fig. 2: Examples of the EWMA estimator for different channel assessment metrics for two different α values. The shaded areas correspond to 90% confidence intervals.

We next use Monte Carlo simulation to visualize the effectiveness of the two channel assessment metrics, as well as the effect of the filter parameter α. In the simulation the node samples the channel every single unit of time, whilst the interference level (λ) changes every $N = 1000$ units of time. In addition, the simulation assumes the maximum IEEE 802.15.4 frame size, *i.e.* $\tau = \tau_i = 3.87$ ms, and that the duration of an RSSI measurement is $\tau_{cca} = 0.3$ ms. Fig. 2a and Fig. 2b plot the estimated channel quality of the PRR and RSSI method against the actual PRR of the channel, for $\alpha = 0.015$ and $\alpha = 0.004$ respectively. The simulations demonstrate that the effect of the α parameter is twofold. On one hand, a higher value of α allows the estimator to quickly adapt to changes in the interference pattern. On the other hand, a smaller value of α allows the estimator to converge closer to the actual PRR value. Moreover, both figures demonstrate that the output of the RSSI-based channel quality estimator is highly correlated with the actual PRR.

C. Optimization of the α coefficient

As it can be seen in Fig. 2b, the EWMA has two phases of operation: the stabilization phase and the stable phase. In the former, the estimator adapts to a recent change in the channel quality. This initial approximation error follows an exponential decay with a rate of α.

$$\epsilon_t = Ae^{-\alpha t} \ . \tag{5}$$

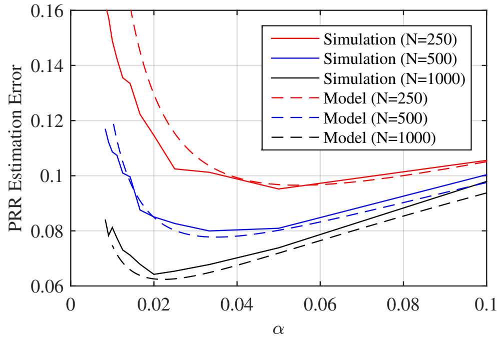


Fig. 3: The accuracy of PRR estimation for various α parameters.

Let us consider that the stabilization phase lasts for $T_{1/4}$ time units, that is until the error reaches a quarter of its initial value.

$$T_{1/4} = \frac{2\ln 2}{\alpha} \ . \tag{6}$$

The average error during the stabilization phase can be then approximated by:

$$\bar{\epsilon} = \frac{1}{T_{1/4}} \int_0^{T_{1/4}} A e^{-\alpha t} dt = \frac{A}{\alpha}(1 - e^{-2\ln 2}) \ . \tag{7}$$

During the stable phase, we model the error as the standard error of a Bernoulli distribution, considering the standard deviation in the worst case scenario.

$$\epsilon_s = A \frac{1}{2\sqrt{T_{1/4}}} \tag{8}$$

The overall error can now be approximated by:

$$E = \frac{T_{1/4}}{N}\bar{\epsilon} + \frac{N - T_{1/4}}{N}\epsilon_s = \frac{A}{N}\Big(k \ T_{1/4} + \frac{N - T_{1/4}}{2\sqrt{T_{1/4}}}\Big) \ , \tag{9}$$

where $k = \frac{(1-e^{-2\ln 2})}{2\ln 2} = 0.541$. Fig. 3 plots the error model against Monte Carlo simulations for $A = 0.67$ and for various periods of change, N, for the interference pattern. We can observe that the model is particularly good at predicting the optimum α coefficient for the EWMA. Moreover, the results suggest that this optimum configuration depends on the dynamics of the interference pattern - the more frequently the interference pattern changes, the higher the α value needs to be, so that the system quickly adapts these frequent changes.

We can now minimize the error E, obtained by Eq. 9, with respect to $T_{1/4}$, and obtain the optimum α value using Eq. 6.

$$\arg\min_{\alpha \in (0,1)} E(\alpha) = \arg\min_{\alpha \in (0,1)} \frac{A}{N}\Big(k \ T_{1/4}(\alpha) + \frac{N - T_{1/4(\alpha)}}{2\sqrt{T_{1/4}(\alpha)}}\Big) \tag{10}$$

The optimum α is obtained by Eq. 6 using the $T_{1/4}(\alpha)$ that solves the following equation:

$$-4 \ k \ T_{1/4}(\alpha)^{\frac{3}{2}} + T_{1/4}(\alpha) + N = 0 \ , \tag{11}$$

where $N > 0$, and $A \neq 0$. The analytic expression of the solution of Eq. 11 is provided in the Appendix. Fig. 4 plots the optimum α parameter for various values of N.

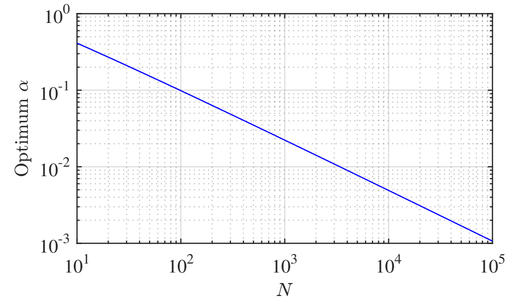


Fig. 4: The dependence of the optimum EWMA α parameter on the period of change of the interference pattern.

III. Channel selection

A. Background and related work

In TSCH networks, enhanced beacon (EB) messages are periodically transmitted for network maintenance purposes by all nodes capable of data forwarding. The IEEE 802.15.4 standard allows to distribute hopping sequence information inside these EB packets, and we take advantage of this feature. We assume a single-parent network (*i.e.*, a tree), where each non-root node accepts EB packets exclusively from a single upstream node, and direct sibling-to-sibling communication never takes place. Under this assumption, whenever the upstream node lists its current hopping sequence in an EB packet, all directly reachable downstream nodes that receive this EB can use it to update their own hopping sequences. This allows to *replace the channel hopping sequence* in the whole TSCH network, or in some of its branches.

Another method for ACS in TSCH is *channel blacklisting*. Here, the hopping sequence as such is not changed; instead, every time a packets is to be transmitted or received on a specific channel, the blacklist is checked, and if the channel is in the blacklist, it is pseudorandomly replaced with another, non-blacklisted candidate channel. Unlike the hopping sequence replacement technique, blacklisting can be initiated both by upstream and downstream nodes.

Channel selection is also often done in single-channel networks (for example, ZigBee standardizes this). However, selecting one sufficiently good channel out of n is an easier problem than selecting k channels out of n, in particular because the latter may require k times more energy for channel quality measurements.

From the existing methods (Table I), ATSCH [4] is an extension of TSCH that uses RSSI sampling for quality assessment and applies upstream-driven hopping sequence replacement together with both-ends-driven blacklisting. In ATSCH, the set of non-blacklisted channels on a link is defined as the intersection of the channels blacklisted on the upstream and downstream nodes. However, ATSCH reserves several slots in each slotframe for channel sampling, significantly reducing the network capacity. Moreover, it reserves an additional slot for control traffic, furthermore reducing the capacity and incurring potentially large costs for idle listening.

ETSCH [5] gives better overall results, as it improves the assessment accuracy of ATSCH due to more frequent RSSI

TABLE I: Comparison of several ACS methods

Name	Channel assessment	Detection of localized interference	Regional adaptations	Channel assessment overhead	Channel selection overhead	Impact on network capacity
This paper						
Upstream-driven	RSSI-based	$+$	$+$	Frequent RSSI sampling (on upstream only)	Larger EB packets	None
Downstream-driven	PRR or RSSI-based	$+$	$+$	PRR: lost packets; RSSI: frequent channel sampling	Notification messages	Small (notification traffic only)
Existing work						
ATSCH [4]	RSSI-based	$+$	$+$	Infrequent RSSI sampling	Notification messages, larger EB packets, and idle listening in reserved slots	Large: three reserved slots per slotframe
ETSCH [5]	RSSI-based	$-$	$-$	Frequent RSSI sampling (on gateway only)	Larger EB packets	None
Li *et al.* [6]	PRR-based	$+$	$+$	Lost packets	Larger data packets	Shorter payload in data packets
Autonomous blacklisting [8]	ETX-based	$+$	$+$	Lost packets	None	Dynamic: zero to a large number of skipped slots

sampling and avoids the capacity loss. However, it does not tackle the problem of localized interference, as it does not give any "voting rights" to downstream nodes at all. Neither ATSCH nor ETSCH use PRR as their quality metric, therefore may fail to detect localized interference affecting only one end of the link, and cannot help with frequency-selective fading.

Li *et al.* [6] implement downstream (*i.e.*, transmitter) driven PRR-based selection where the list of channels is included in data packets sent to the upstream node. They use a moving average of PRR as the channel quality metric, extended with setting the metric to zero in case of a CCA failure or consecutive Tx failures. However, [6] lacks some details necessary for an implementation of their solution, for example, how the list of "good" channnels is selected. Additionally, they require that the Rx node remains aware of the Tx packet rate, which is not a realistic assumption when, for example, adaptive of event-driven sensing is used.

Autonomous channel blacklisting [8] is a zero-coordination method proposed for CSMA-CA multichannel networks. Here, locally blacklisted channels are not used for transmissions; instead, the node willing to transmit awaits for a non-blacklisted channel to become active. This method increases reliability and energy efficiency at the expense of delay, but is not well-suited for the typical TSCH network as it may significantly reduce the network capacity.

In all of these methods, a predefined set of potentially active channels is kept on all nodes. We call this set *candidate channels*.

B. Upstream-driven selection

In this method, the upstream node periodically samples RSSI updates the qualities of all candidate channels using Eq. 2. The input value $Y_n = 0$ iff the idle RSSI value is above a user-defined "noise floor" threshold ($-95\,$dBm by default in our implementation), and $Y_n = 1$ otherwise. PRR-based channel assessment is not a perspective method here, as in typical TSCH applications upstream nodes typically do not send frequent periodic unicast packets to downstream nodes.

To detect external interference, the RSSI sampling should be done when the TSCH network itself is not transmitting any packets. In our implementation, it is done between the start of a TSCH timeslot and the Tx offset, defined by the IEEE 802.15.4 standard to be $2120\,\mu$s later.

Upon detecting that a channel in the current hopping sequence is busy, *i.e.*, the channel quality estimator p_n falls below a user-defined quality threshold THR (with value between 0.0 and 1.0), the node searches for a replacement in the candidate channel set. The channel with the best quality metric is selected from the candidate channels that are: (1) above the minimal quality threshold; (2) not already in the current hopping sequence; (3) not recently detected as busy (in our implementation — in the last 5 minutes). The condition (3) is meant to reduce network churn in case of rapid channel quality fluctuations.

If such a channel is found, the current hopping sequence is updated, and several EB packets with the new sequence are generated in rapid succession to update the downstream nodes.

C. Downstream-driven selection

Here, each downstream node keeps track of two blacklists: the *local* blacklist, containing up-to-date information about the locally detected busy channels, and the *shared* blacklist, containing the last version of the local blacklist received and acknowledged by the upstream node. Only the shared blacklist is used for packet receptions on the downstream node, as the transmitting upstream node is not necessarily aware of the downstream node's local blacklist.

Downstream node operation. At the start of a scheduled transmission or reception slot, the node first selects the channel c to use in the slot following the usual procedure described in the IEEE 802.15.4 standard (Eq. 1), using the current hopping sequence, the current ASN, and the scheduled channel offset as the input parameters. Subsequently, the local and the shared blacklists $LocalBL$ and $SharedBL$ are checked:

- If $c \notin LocalBL$ and $c \notin SharedBL$, the channel c is used.
- If $c \in LocalBL$, but $c \notin SharedBL$, the slot is skipped, as this means a temporary inconsistency between the nodes.
- Otherwise, c is pseudorandomly replaced with another channel from the candidate channel pool.

If the RSSI-based metric is used, the channel qualities are updated as described in Section III-B. For the PRR-based metric, they are instead updated after the end of transmissions. The reception of an ACK from the upstream node is used to decide whether the Tx has been successful. A new value of channel quality estimator p_n is calculated using Eq. 2, with $Y_n = 1$ in case of success, $Y_n = 0$ otherwise. If after the update p_n falls below the quality threshold THR, the channel is added to the local blacklist. Additionally, each time a channel is replaced or skipped for a transmission, its quality is slightly increased in order to eventually unblacklist the channel. (However, a channel is always kept in the local blacklist for at least 5 minutes in our implementation to avoid network churn.) This is done iff $p_n < THR$ using a modified Eq. 2, with $Y_n = 1$, but with reduced $\alpha/2$ as the coefficient.

Whenever the local blacklist on the downstream node is updated, a high-priority notification message containing the new blacklist is inserted at the *front* of the node's message queue, unless such a message is already scheduled. When this message is successfully acknowledged by the upstream node, the shared blacklist is updated on the downstream node.

Upstream node operation. The upstream node selects channels for transmissions and receptions in a similar way: initially, the IEEE 802.15.4 standard procedure is applied to calculate a channel c. If c is not in the shared blacklist, it is used; otherwise, it is pseudorandomly replaced with another candidate channel, using the same random number generator as on the downstream node. The node does not perform channel quality assessment on downstream links on its own, it relies on notification messages from downstream nodes.

IV. Evaluation

A. Setup

To evaluate the methods, we try them out in several different scenarios with various levels of interference. We use a 6-channel large candidate channel set with IEEE 802.15.4 channels $11, 14, 17, 20, 23, 26$, and use $HS = 14, 17, 20, 23$ as the initial hopping sequence.

In most of the experiments, we generate light interference on *all* of these candidate channels, leading to 20 % expected packet error rate (PER) even on the "good" channels. Additionally, extra interference is generated on three of the candidate channels; the amount of this interference is varied in the different experiments, leading to 20 — 100 % expected PER on these extra-interfered channels. The numbers of the extra-interfered channels are randomly changed every 10 minutes. The only exception is a test case with no interference at all, designed purely to show the overhead of the ACS process.

TABLE II: Parameters used in the evaluation.

Method	N (per channel)	$\alpha(N)$
RSSI-based upstream	1000	0.028
RSSI-based downstream	400	0.068
PRR-based (with $PRR = 66\,\%$)	225	0.140

The methods are evaluated for 30 min in star topology with one upstream node and four downstream nodes. Application's data rate is fixed at one packet per second per node, with 120-byte PHY layer size. Slotframe period is 0.5 sec; each slotframe has a single shared slot for broadcast traffic and one contention-free slot for each downstream→upstream link; consequently, the schedule capacity is 2 unicast packets per second per link, *i.e.*, if no retransmissions were required, only 50 % of the network's maximal capacity would be used. This configuration and the 10 min change period implies the values of N, the number of channel samples (Table II). Using N as input, we select $\alpha(N)$ using Eq. 11.

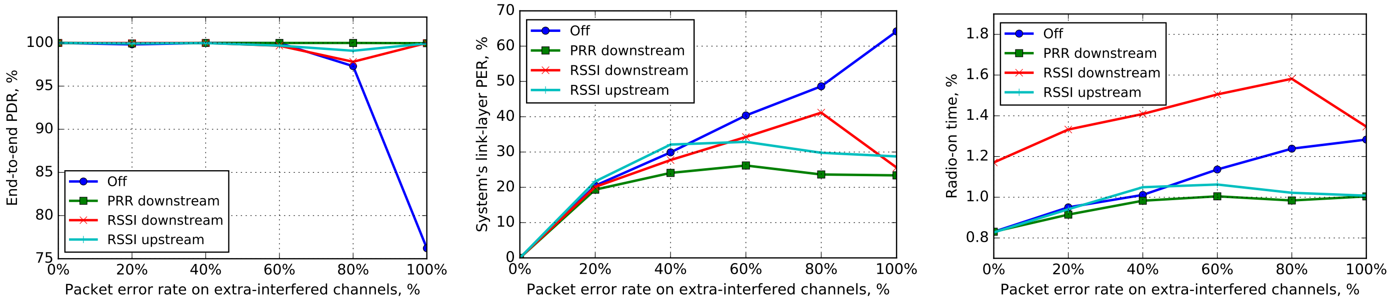
The problem here essentially is to distinguish between heavily-interfered and lightly-interfered channels. To facilitate this, we also select different threshold THR values for the different scenarios, setting the "is busy" threshold to the midpoint of the expected quality the "good" channels and the expected quality of the "bad" channels.

B. Simulation

We use the Cooja network simulator. The proposed ACS methods are compared with a baseline, unmodified Contiki implementation of TSCH [9] (Fig. 5). The simulator is configured to randomly generate interference packets, each 800 μs long, following the Poisson distribution; the average frequency of these packets are channel and experiment dependent.

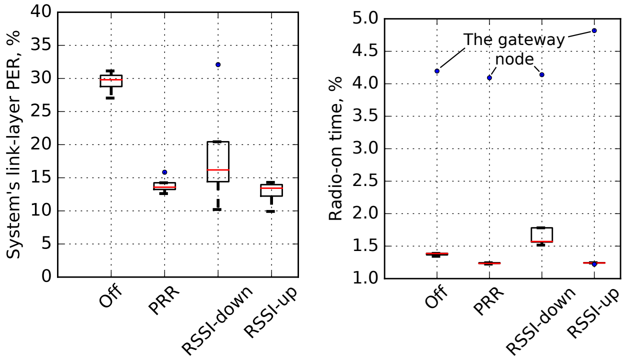
Baseline (selection off). As expected, packet error rate (PER) in this baseline configuration increases linearly with the interference PER in the extra-interfered channels. In high interference levels PDR starts declining (Fig. 5a), as the schedule runs out of capacity to deliver all retransmissions. In contrast, all adaptive methods always give close-to-100 % PDR, as they are able to exploit the less-interfered channels.

PRR downstream. This shows the best results on average. The radio duty cycle is only slightly better than what the upstream RSSI method achieves, but, since the PRR estimator has some additional qualitative benefits (*e.g.*, it measures the target value, not just its correlate; does not require the user to fine-tune the "noise floor" threshold; is finer grained; and can recognize frequency-selective fading), there are even more reasons to prefer it.



(a) End-to-end packet delivery rate **(b)** Link-layer packet error rate **(c)** Radio-on time

Fig. 5: The performance of the different ACS methods in simulations.



(a) Link-layer packet error rate **(b)** Radio-on time

Fig. 6: The performance of the different ACS methods in the testbed. PDR for all methods >99.8 %.

RSSI upstream. This method shows solid performance; it is particularly recommended if the upstream node is not energy constrained. However, in the 80 % PER case it is not able to deliver >99.9 % PDR unlike the PRR method.

RSSI downstream. This shows the worst results of the adaptive approaches, likely because this method has to work with the least amount of information. In particular, it never shows better radio duty cycle than the baseline TSCH, although it does reduce the number of retransmissions.

The two best of the adaptive methods do not incur noticeable costs in 0 % and 20 % PER experiments, *i.e.*, in the experiments with no differences in channel qualities. On the other side of spectrum, in the 100 % experiment the PRR methods has to retransmit 2.7 times fewer packets and has 22 % lower radio duty cycle.

C. Testbed

We use a WiFi (802.11b) router to generate interference on a random set of IEEE 802.15.4 channels. Network topology and other parameters are the same as in the simulator, except that we use Texas Instruments CC2650-based hardware nodes, and noise floor set to −85 dBm. We run just a single experiment to validate the conclusions from the simulations; the nodes are expected to have around 50 % PER in interfered channels.

The results are given in Fig. 6. Data from individual nodes are used to construct each box-plot. Here the RSSI-upstream method requires 8 % less retransmissions than the PRR method on the average (and 2.3 times less than the baseline). Still, the latter is the most efficient one in terms of duty cycle, although the difference is less pronounced than is simulations (as the CC2650 radio spends more time in idle listening): just 10.0 % improvement on downstream nodes on average.

V. Conclusions

By using simulations and testbed experiments with CC2650-SoC devices, we have compared multiple methods for adaptive channel selection in IEEE 802.15.4 TSCH networks under WiFi interference: upstream and downstream driven, RSSI and PRR based.

The results from the evaluation setup show that under heavy interference the standard TSCH does not have enough scheduled slots to reliably deliver all packets. In contrast, the adaptive approaches make the system more reliable as long as a few channels are free from interference; in particular, the PRR approach shows close to 100 % PDR in all scenarios.

Acknowledgments

This work was performed under the SPHERE IRC funded by the UK EPSRC, Grant EP/K031910/1.

References

[1] "IEEE Standard for Local and metropolitan area networks—Part 15.4," IEEE Std 802.15.42015, 2015.

[2] B. Kerkez, T. Watteyne, M. Magliocco, S. Glaser, and K. Pister, "Feasibility analysis of controller design for adaptive channel hopping," in *the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*, 2009.

[3] "Bluetooth Specification Version 4.2," https://www.bluetooth.org/en-us/specification/adopted-specifications, 2014, Bluetooth SIG.

[4] P. Du and G. Roussos, "Adaptive time slotted channel hopping for wireless sensor networks," in *4th Computer Science and Electronic Engineering Conference (CEEC)*. IEEE, 2012, pp. 29–34.

[5] R. Tavakoli, M. Nabi, T. Basten, and K. Goossens, "Enhanced Time-Slotted Channel Hopping in WSNs using Non-Intrusive Channel-Quality Estimation," in *IEEE MASS*, 2015, pp. 217–225.

[6] P. Li, T. Vermeulen, H. Liy, and S. Pollin, "An adaptive channel selection scheme for reliable TSCH-based communication," in *IEEE ISWCS*, 2015, pp. 511–515.

[7] H. Lieske, F. Beer, G. Kilian, J. Robert, and A. Heuberger, "Characterisation of Channel Usage in ISM/SRD Bands," in *European Telemetry and Test Conference*, 2014, pp. 32–38.

[8] A. Elsts, "Source node selection to increase the reliability of sensor networks for building automation," in *EWSN*, 2016, pp. 125–136.

[9] S. Duquennoy, A. Elsts, B. A. Nahas, and G. Oikonomou, "TSCH and 6TiSCH for Contiki: Challenges, Design and Evaluation," in *IEEE DCOSS*, 2017.