Experimental evaluation of IEEE 802.15.4 TSCH on a 6TiSCH network
Garcia Algora, Carlos Manuel; Vitalio, Alfonso Reguera; Steenhaut, Kris

Published in:
XVII Symposium of Electrical Engineering (SIE 2017)

Publication date:
2017

Document Version:
Final published version

Citation for published version (APA):
Experimental evaluation of IEEE 802.15.4 TSCH on a 6TiSCH network

Carlos M. García Algora*, Vitalio Alfonso Reguera†, Kris Steenhaut *
*Department of Electronics and Informatics, Vrije Universiteit Brussel
{cgarciaa, ksteenha}@etrouvub.be
†Department of Telecommunications and Electronics, Universidad Central "Marta Abreu" de Las Villas
{cgalgora, vitalio}@uclv.edu.cu

Abstract

The Time-Slotted Channel Hopping (TSCH) mode for medium access control (MAC) included in the standard IEEE 802.15.4 has been designed as the multichannel MAC protocol for Low-power and lossy networks (LLNs), a key component of the Internet of Things (IoT). Its flexibility makes the TSCH mode a very promising candidate for the future of the MAC layer in LLNs. As such, its performance under different conditions must be assessed, so that accurate guidelines for its application can be drawn. In this paper, we present an experimental evaluation of the protocol based on simulations on ContikiOS Java Simulator (COOJA). The experimental setup is based on the 6TiSCH architecture for LLNs.

Keywords

TSCH; LLN; Evaluation; COOJA; Multichannel MAC; Wireless Networks

I. INTRODUCTION

The realization of the Internet of Things relies heavily on Low-power and lossy networks (LLNs), which are composed of several low-end devices interconnected through low-power half-duplex transceivers using batteries as their main power source. In addition, LLNs are intended to be deployed as ad hoc self-configurable networks that last for several months without direct human intervention. To provide coverage of large areas with low-power radios, LLNs must rely on multi-hop communications. Because of these characteristics, energy efficiency and auto-adaptability are major concerns in LLNs.

The expanding range of applications of LLNs include environmental monitoring, industrial control, security and surveillance, health monitoring, smart homes/buildings/cities, precision agriculture and many others. These applications create a wide spectrum of requirements for LLNs from high-throughput time-critical surveillance systems to low-throughput precision agriculture scenarios. Moreover, according to the physical specifications (PHY) of the standard IEEE 802.15.4 [1], most LLNs operate in the 2.4 GHz band, in which they have to coexist with other systems composed of devices with higher radio output power and traffic demands. Hence, LLNs must be robust against interference and noise.

Since the radio interface is the component that consumes the most energy in LLN nodes, they use an energy-saving technique, called Radio Duty Cycling (RDC) [2], that turns the radio off for more than 90% of the time and only turns it on to transmit packets or to check for pending packets intended for it. Besides RDC, a considerable amount of energy can be saved by using medium access control (MAC) protocols that reduce the energy consumption of the network while keeping a good performance in terms of throughput and delay.

The characteristics of LLNs and the requirements of its applications have imposed the need for new protocols specific for LLNs. Protocols designed for wireless ad hoc networks and general wireless networks
are not suitable for LLNs because most of them assume the existence of multiple radio interfaces, that the radios do not perform RDC, and/or the existence of full-duplex transceivers at the nodes. The plethora of MAC protocols proposed for LLNs ranges from single-channel asynchronous protocols to multichannel time-slotted protocols, including almost any combination in between.

Multichannel communication offers attractive potential advantages in wireless networks. Simultaneous transmissions on orthogonal channels allowed by the multichannel operation can potentially improve the network performance, e.g. in terms of throughput and delay. Hence, the evolution of MAC protocols for LLNs has led to the inclusion in 2012 of the Time-Slotted Channel Hopping (TSCH) MAC protocol as the amendment IEEE 802.15.4e to the standard for MAC and physical layers of Wireless Personal Area Networks (WPANs), which include LLNs as one of its standardizing targets. The amendment is included in the latest version of the standard published in 2015 [1]. Moreover, the 6TiSCH architecture [3], proposed by the IETF as the standard-based protocol stack for future LLNs, is based on TSCH at the MAC layer and IEEE 802.15.4 PHY at the physical layer.

In order to apply and configure TSCH properly, it must be evaluated and its performance under different conditions must be assessed. To this aim, some relevant research work has been presented in the past years [4]–[10]. The studies in [4] and [5] propose theoretical models based on Markov chains, which are validated through simulations and testbed experiments. The results show the validity of the models but they are only applicable to the shared links of TSCH.

The analysis of a TSCH network published in [6] shows the instability of the links with nodes that are more than 5 meters away from each other, as well as the negative effect of IEEE 802.11 (WiFi) systems on the performance of IEEE 802.15.4 networks. Juc et al [8] evaluate the performance of TSCH in terms of delay, but PDR and energy efficiency are not evaluated. In [7], a comprehensive evaluation of TSCH is oriented to office environments in the presence of WiFi networks. A simulation-based performance comparison of TSCH and EM-MAC, a contention-based protocol, is presented in [10]; but more care should be given to the inclusion of confidence intervals.

In this paper, we propose an experimental evaluation of TSCH based on extensive simulations under different traffic loads for a medium-sized LLN. Hence, the contributions of this paper are:

- The performance of TSCH under different traffic conditions is evaluated through extensive simulations.
- The experimental setup follows the 6TiSCH architecture proposed by the IETF, including the User Datagram Protocol (UDP) at the network layer, IPv6 addressing and the IPv6 Routing Protocol for LLNs (RPL) [11] at the network layer, 6LoWPAN [12] as the adaptation sublayer for IPv6 in LLNs, TSCH at the MAC layer, and IEEE 802.15.4 PHY at the physical layer. The performance analysis of TSCH is assessed in terms of packet delivery ratio (PDR), end-to-end delay and energy consumption.

II. IEEE 802.15.4 TSCH

In TSCH, time is divided into slots, each of them long enough to transmit a maximum-length frame (127 bytes) and receive the (optional) corresponding acknowledgement (ACK) frame. Depending on the implementation, this time varies from 10 to 15 milliseconds. Slots are grouped into slotframes that repeat over time. Slots can be shared or dedicated. Communication on shared slots follows a slotted Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) strategy. Dedicated slots establish an unidirectional link between two nodes and only the source of that link can transmit during that slot. More than one node can be allocated as receiver of a dedicated slot, so that link-layer broadcast packets can be sent using only one transmission.

Each slot in a slotframe is identified by a combination of slot offset (inside the slotframe), an absolute slot number (ASN) and channel offset. The ASN is the number of slots that have occurred since the initialization of the network and it is incremented after every time slot. The channel offset is an integer constant that is used to generate a channel hopping pattern using equation 1.

\[
 f_{ch} = F\left(\left(ASN + ch_{offset}\right) \mod N_{ch}\right) \tag{1}
\]
For example, let’s assume that the channel hopping sequence is composed of 16 channels numbered from 11 to 26. In the schedule matrix pictured in Figure 1, during the $k^{th}$ cycle of the slotframe, the physical frequency used during the time slot with $slot_{Offset} = 3$ and $channel_{Offset} = 1$ would be, using equation 1:

$$f_{ch} = F\left(\left(\left(4 + 1\right) \mod 16\right)\right)$$

$$f_{ch} = F(4) = 11 + 5 = 16$$

Analogously, the channel used for the next occurrence of the slot, i.e. during the $(k + 1)^{th}$ cycle of the slotframe, would be $f_{ch} = X$, and so on and so forth. This way a channel hopping pattern is created only by incrementing the ASN while keeping constant values of slot offset and channel offset for each slot inside a slotframe.

Multiple slotframes can coexist in the same network. For example, independent slotframes can be used for different kinds of traffic. When multiple slotframes are used, they are combined into a single schedule and conflicts between active slots of different slotframes are resolved by assigning a priority to each individual slotframe.

A network is created when a coordinator starts sending Enhanced Beacons (EB) containing all the information needed to synchronize with the network. Nodes join the network by listening on all the available channels, one at a time, until they hear an EB from the coordinator or from other node that has successfully join the network. Once a node joins the TSCH network, it can start sending its own EBs. To keep the tight synchronization needed by TSCH, nodes can piggyback synchronization information on any packet sent at MAC level.

The standard IEEE 802.15.4 defines how to manage the schedule but it does not define how to build it. Therefore, scheduling protocols oriented to build the TSCH have been proposed. The work in [13]–[15] stands out as promising alternatives for the future. Nevertheless, SF0 is still too basic for the functionalities needed in LLNs, SF1 is oriented to centralized scheduling, which reduces the flexibility of the network. On the other hand, Orchestra creates TSCH schedules in a decentralized way, adapted to the traffic requirements of the nodes and using information from RPL. This approach places Orchestra among the leading scheduling protocols for 6TiSCH networks.

### A. Orchestra scheduling

Orchestra [15] autonomously creates and maintains TSCH schedules at the nodes using information from the RPL tree. The schedule is composed of multiple slotframes, each one dedicated to a type of traffic. For example, in Figure 2 the application traffic schedule is dedicated to application level communication, RPL signaling is be exchanged following the RPL schedule, and the TSCH beacons schedule is intended for Enhanced Beacon (EB) transmission, which allows nodes to join the network and achieve synchronization.
The Orchestra specification defines four types of slots. They range from common shared slots (CS), yielding a behavior similar to slotted ALOHA, to sender-based dedicated slots (SBD) that can potentially create contention-free medium access. An Orchestra schedule includes:

- One slot common to all nodes for RPL signaling via broadcast or unicast messages.
- One broadcast slot for TSCH EBs from each RPL parent to its children.
- A dedicated transmission slot from each node to its RPL preferred parent
- A dedicated slot from each RPL parent to its children.

If dedicated slots for a node cannot be allocated, shared slots can be used instead.

Time synchronization is achieved making use of the RPL tree, in order to guarantee a loop-free synchronisation structure. Different from TSCH, Orchestra uses periodic TSCH and RPL beacons for this purpose. The slotframe length can be tuned to trade off throughput, network latency and power consumption. Orchestra is implemented in ContikiOS and is available in Contiki’s GitHub repository.

### III. Experimental Setup

Simulations are performed using the ContikiOS Java (Cooja) simulator using Zolertia Z1 devices as network nodes to be simulated. For each configuration, 50 simulations runs with different random seeds were performed to provide statistical validity to the results. The protocol stack used on simulation is based on the 6TiSCH architecture. The only exception would be the use of Orchestra for the creation of the TSCH schedule instead of using SF0 or SF1, as suggested by the 6TiSCH specification.

<table>
<thead>
<tr>
<th>Network stack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
</tr>
<tr>
<td>Transport</td>
</tr>
<tr>
<td>Network</td>
</tr>
<tr>
<td>MAC</td>
</tr>
<tr>
<td>Physical</td>
</tr>
</tbody>
</table>

The protocol stack used in all the experiments is shown in Table I. In all the experiments, 22 sensor nodes send information about the network performance to a sink using UDP packets. The metrics used to assess the performance of TSCH are packet delivery ratio (PDR), delay and energy efficiency. The delay refers to the end-to-end delay at application layer. For the energy efficiency, two metrics are used:

- RDC% On: Calculated as the percentage of time the node stays awake on the total simulated time.
- RDC% Tx + Rx: Calculated as the percentage of time the node is transmitting or receiving on the total simulated time.

Table II shows the values used for the parameters of the network setup. The traffic load of the network is regulated using the packet generation rate (PGR) of the nodes.

---

1ETX stands for expected transmission count, defined as the average number of radio transmissions needed to successfully send a packet through a given route.
TABLE II
SIMULATION PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radio Output Power</td>
<td>0 dBm</td>
</tr>
<tr>
<td>Radius of Transmission Area</td>
<td>35 m</td>
</tr>
<tr>
<td>Radius of Interference Area</td>
<td>35 m</td>
</tr>
<tr>
<td>Packet Size</td>
<td>125 bytes</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>5400 s (1h30m)</td>
</tr>
</tbody>
</table>

IV. RESULTS AND DISCUSSION

When the traffic load is incremented in a communication network, it is to be expected that the higher the traffic load, the lower the PDR. As Figure 3 shows, that principle does not always stand in an LLN running TSCH and Orchestra. Though the difference is small, the graph in 3a shows that for $PGR \leq 2\text{ pkts/min/node}$ the PDR increases until it reaches its maximum at that point in the graph.

![Figure 3](image1)

(a) Packet delivery ratio (3a), RDC percentage with radio transmitting or receiving (3b), RDC percentage with radio on (3c).

![Figure 4](image2)

(a) Average delay (4a), Cumulative Density Function of delay (4b), RDC percentage with radio on (3c).

The reason for this behavior of the protocol is given by its synchronization mechanism. According to [1] and [15], TSCH and Orchestra use data and ACK frames for piggy-backing synchronization information. When traffic load (expressed here in terms of PGR) is low, data frames are not exchanged often enough to keep the network tightly synchronized. Therefore, packets are lost because of nodes that are not correctly synchronized with their parents.

The explanation above is confirmed by the energy efficiency graphs shown in Figure 3. In Figure 3b it can be corroborated that the percentage of time spent transmitting or receiving increases proportionally to the traffic load, which is the expected result as more packets need to be exchanged. However, Figure 3c shows that the percentage of time spent with the radio on (i.e. transmitting, receiving or idle listening) is higher for low traffic loads. Being a time-slotted protocol, the percentage of time spent awake should
increase proportionally to the traffic load, plus a baseline consumption caused by waking up during reception slots even when no packet is going to be sent. Hence, the performance degradation in terms of energy efficiency for low traffic loads can only be explained by the influence of the synchronization mechanism of TSCH/Orchestra.

One of the main advantages of time-slotted MAC protocols is that they provide bounded delay as long as the traffic load does not exceed the capacity of the network. Figure 4a shows that the performance of TSCH/Orchestra in terms of average delay of application data packets increases only slightly for $PGR \leq 2$ pkts/min/node, which is consistent with the expected behavior of a TDMA-based protocol. As expected, the performance suffers a noticeable degradation when the network need to handle more traffic than its capacity allows at $PGR \geq 4$ pkts/min/node.

An interesting result shown in this graph is that the average delay is slightly higher for $PGR = 0.25$ pkts/min/node than for $PGR = 0.5$ pkts/min/node. This result confirms that the low frequency of data exchange between neighbors degrades the synchronization in the network, and therefore packets need to wait for the children to re-synchronize with their parents. This results is more noticeable in Figure 4b, which shows the cumulative density function of the delay at application level. Notice that the worst performance occurs when $PGR = 0.25$ pkts/min/node (magenta-colored line in Figure 4b). Moreover, the best performance in terms of delay is for $PGR = 2$ pkts/min/node (green-colored line in Figure 4b), which confirms the previous explanation.

V. Conclusions

The TSCH mode of the IEEE 802.15.4 standard for LLNs shows a stable performance under average traffic conditions in terms of PDR, energy efficiency and delay. Nevertheless, the synchronization mechanism in TSCH causes performance degradation when packets are not sent often enough. A potential solution could be to use periodic packet transmissions at the MAC layer, when no data packets have been sent for a certain amount of time. These “keep alive” messages will increase the protocol overhead but they could improve the delay experienced by application data packets, and could even improve the energy efficiency of TSCH.

The deployment of LLNs using TSCH must take into account the traffic demands of the network and configure the scheduler in accordance. In this case, the default configuration of Orchestra does not provide enough capacity to handle PGR values higher than 2 pkts/min/node, which is not too high for LLN applications such as Body Area Networks (BANs) and surveillance systems. New scheduling algorithms for TSCH oriented to work with the 6TiSCH architecture are required.

Acknowledgment

The authors would like to thank the Belgian Development Cooperation through VLIR-UOS for supporting this work under the ICT Network Project between VLIR (Vlaamse Inter Universitaire Raad, Flemish Interuniversity Council, Belgium) and Five Cuban universities.

References


