

# Sensors Lab Conference

Gabriele Paris

*Dept physic*

*University of Antwerp*

Antwerp, Belgium

gabriele.paris@student.uantwerpen.be

Pieter Hendriks

*Dept computer science*

*University of Antwerp*

Antwerp, Belgium

pieter.hendriks@student.uantwerpen.be

**Abstract**—In this study, we assess the performance of TSCH using the Contiki-NG operating system on Zolertia RE-mote (Rev. B) nodes. We'll show that, in two node networks, the leaf node consumes more power than the root node. Further, we'll assess the performance of the TSCH mode over various distances.

**Index Terms**—6TiSCH, 6TSCH, Contiki-NG, Energest, IEEE 802.15.4, Power Consumption, Range, Time Slotted Channel Hopping, TiSCH, TSCH, Zolertia

## I. INTRODUCTION

The IETF IPv6 over the TSCH mode of IEEE802.15.4e (6TiSCH) working group has standardized a set of protocols to enable low power industrial-grade IPv6 networks. 6TiSCH proposes a protocol stack rooted in the Time Slotted Channel Hopping (TSCH) mode of the IEEE802.15.4-2015 standard, supports multi-hop topologies with the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) routing protocol, and is IPv6-ready through 6LoWPAN [2].

Contiki-NG is an operating system for resource-constrained devices in the Internet of Things. Contiki-NG contains an RFC-compliant, low-power IPv6 communication stack, enabling Internet connectivity. The system runs on a variety of platforms based on energy-efficient architectures such as the ARM Cortex-M3/M4 and the Texas Instruments MSP430. The code footprint is on the order of a 100 kB, and the memory usage can be configured to be as low as 10 kB.

This paper serves as an initial look at the performance of TSCH on the Contiki-NG operating system. Specifically, we're going to look at the energy consumption by the TSCH network stack vs the 6TiSCH network stack. We'll also analyze the TSCH joining process. Finally, we'll see how much impact the distance between node has on the network.

## II. RELATED WORK

### A. Throughput Analysis

[5] In this paper, an analysis of the throughput performance of TSCH networks is made. They show which parameters of the TSCH networks (node count, hop count, ...) influence the throughput metric in what way. Our research focuses on different performance characteristics of the TSCH implementation and can be useful alongside these results.

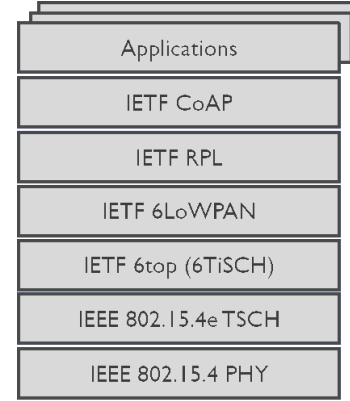


Fig. 1. 6TiSCH Protocol Stack [3].

## III. ANALYSING THE 6TiSCH ENERGY CONSUMPTION

In the first analysis, we compared the energy consumption during a certain time period of the entire 6TiSCH stack to when only enabling the TSCH MAC layer (without link-layer security) after network convergence. The full stack can be seen as in Figure 1.

For both analyses, we report on the consumption of the root and the leaf node separately. Following we remark differences in energy consumption between the root and the leaf node in the two different scenarios.

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 [6].

the physical setup of the experiment consists of two Zolertia Remote RevB boards placed at a fixed distance of approximately 20 cm as shown in Figure 2.

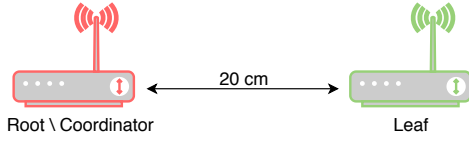


Fig. 2. Network topology for 6TiSCH energy consumption analysis.

The basic setup for the following analysis are a root (coordinator) node and a leaf node that exchange the same amount of data packets.

Both nodes are configured to measure the energy consumption by the Energest<sup>1</sup> module available in Contiki-ng<sup>2</sup>.

The Energest module can be used to implement lightweight, software-based energy estimation approach for resource-constrained IoT devices. By tracking the time various hardware states such as the radio is turned on, and by knowing the power consumption of the state, it is possible to estimate the energy consumption [1].

The energy consumption is measured referring to the following formula:

$$E_{tot} = \sum_{c \in state}^{N_s} E_s = \sum_{s \in state}^{N_s} I_s \cdot V_{cc} \cdot t \quad (1)$$

Where  $V_{cc}$  is the supply voltage, fixed as a constant at the value 3.3V,  $I_s$  is provided by the Table I column *Device profiling* [4], and  $t$  is measured using Energest.

All measurements are performed in a span of 15 minutes each.

State	CC2538 datasheet	Device profiling
CPU	20 mA	15.35mA
LPM	0.6 mA	9.59 mA
Deep LPM	0.0013 mA	2.58 mA
LISTEN	24 mA	28.32 mA
Rx	27 mA	30.14 mA
Tx	34 mA	31.12 mA

TABLE I

COMPARISON BETWEEN VALUES FROM THE DATASHEET AND N6705B DC POWER ANALYZER RADIO. [4]

#### A. Only TSCH MAC layer

In the first part of the experiment, we used only the stack till the MAC layer (TSCH) as shown in Figure 1.

We proceeded as explained in the setup to measure the energy consumption between leaf and root node.

Results are reported in Table II. What we can see is that data

Node	CPU	LPM	Deep	Tx	Rx
Leaf	3.330 mJ	314.385 mJ	85.143 mJ	1.132 mJ	993.527 mJ
Root	3.330 mJ	314.385 mJ	85.143 mJ	1.132 mJ	993.527 mJ

TABLE II

AVERAGE POWER CONSUMPTION COMPARISON BETWEEN ROOT AND LEAF NODE PER STATE, TSCH MAC ONLY.

from root and leaf nodes show no variation, and the energy consumed from both of them is the same.

#### B. Full stack

The next experiment is as cited in the section III introduction related to the energy consumption once the full 6TiSCH stack has been enabled (except for the security layer).

The setup of this experiment is the same to the previous one. Two boards 10cm apart from each other (2) are running the same source code.

In this scenario, we have a coordinator node and a leaf node. No messages are exchanged between the two if not for standard 6TiSCH service messages.

We proceeded as explained in the setup to measure the energy consumption between leaf and root node.

Results are reported in Table III.

Node	CPU	LPM	Deep	Tx	Rx
Leaf	5.00 mJ	313.341 mJ	85.143 mJ	80.121 mJ	98.362 mJ
Root	3.890 mJ	310.560 mJ	84.209 mJ	0 mJ	33.879 mJ

TABLE III

AVERAGE POWER CONSUMPTION PER STATE BETWEEN ROOT AND LEAF NODE, FULL 6TiSCH STACK.

#### C. Final comparison

As a final comparison, we compared the total energy usage between root and leaf nodes in the two configurations. What we measured has been reported as Table IV.

Node	MAC only	Full stack
Leaf	1.398 J	0.582 J
Root	1.398 J	0.433 J

TABLE IV

ENERGY COMPARISON BETWEEN LEAF AND ROOT NODE IN MAC ONLY AND FULL STACK CONFIGURATIONS.

### IV. ANALYSING RANGE CAPABILITIES

#### A. Approach

In order to properly analyse the range capabilities, we must first define a test structure that will be repeated at each range. The test in this case is sending 120 packets (1 per second, 52 data bytes each) over a specified range (varying between 1 and 100 meters) so we can show the performance characteristics of TSCH as the distance between nodes varies. The data payload size was chosen out of convenience, as each packet carries metrics from the sender and they happened to be 52 bytes in size. Varying this packet size will result in different performance metrics and so must be addressed V.

Due to time constraints, combined with other obligations, the full scope of the experiments in this section has not yet been carried out. A brief comparison is made (showing the methodology) between nodes with 1 meter separation and nodes with 5 meters separation. Further work must be done to complete this analysis V.

As of now, though, we can show that the TSCH functionality is essentially unchanged when we're looking at 1 vs 5 meters

<sup>1</sup><https://github.com/contiki-ng/contiki-ng/wiki/Documentation:-Energest>

<sup>2</sup><https://github.com/contiki-ng>

separation between the two nodes. The power requirements are within margin of error, there is no significant packet loss in either case. The longer distance has a slightly higher latency (potentially caused by more retransmissions, we'll need to look at this V).

For this analysis, we disabled the 6TiSCH stack (using only the TSCH mac layer) and ensured no security was enabled. No further changes to the Contiki-NG network stack were required.

## B. Results

1) *1 meter separation:* First, we'll compare the latencies. At 1 meter separation, we found, on average, 9.29 timeslots delay between the enqueueing of the packet on the sender's side and the arrival on the receiver's side ???. Presumably, a majority of this time is spent in the queue (or in retransmissions), but this will be verified later V. Over the distance of 1 meter, there is 100% packet arrival rate.

Regarding power use, then, we see that the leaf node consumes more power than the root node does. This is shown in ??, combined with the current draw for each power state as described in I. Overall, we find that the leaf node has used approximately 4.35 joules of energy whereas that value is only 2.58 joules for the root node. Since each packet carried a data payload of 52 bytes, we find the following power requirements: roughly  $89 \frac{mJ}{kbit}$  for the leaf node and  $53 \frac{mJ}{kbit}$  for the root node.

2) *5 meters separation:* When we increase the distance to 5 meters, not much changes. As the distance is still very small (and should be easily within range), this is to be expected. The packet delivery ratio is still at 100%, latency is slightly increased to roughly 9.33 slots (but this could be within margin of error - we should look at retransmission rates to verify V).

The energy consumption is essentially the same, they are definitely within margin of error for our current test setup and so will not be discussed further than they have been in IV-B1.

## V. FUTURE WORK

- Analyze the TSCH joining process
- Perform the experiments as required for IV
- Further discuss the current results in IV
- Further analyze the found flaws in IV

## REFERENCES

- [1] Adam Dunkels, <https://github.com/contiki-ng/contiki-ng/wiki>
- [2] X. Vilajosana, T. Watteyne, T. Chang, M. Vučinić, S. Duquennoy and P. Thubert, "IETF 6TiSCH: A Tutorial," in IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 595-615, Firstquarter 2020, doi: 10.1109/COMST.2019.2939407.
- [3] Karaagac, Abdulkadir & Haxhibeqiri, Jetmir & Moerman, Ingrid & Hoebeke, Jeroen. (2018). Time-critical communication in 6TiSCH networks. 161-166. 10.1109/WCNCW.2018.8368987.
- [4] Sabovic, Adnan & Delgado, Carmen & Bauwens, Jan & De Poorter, Eli & Famaey, Jeroen. (2019). Accurate Online Energy Consumption Estimation of IoT Devices Using Energest. 363-373. 10.1007/978-3-030-33506-9\_32.
- [5] Lee, S. B., Kim, E. J., & Lim, Y. (2018). Contiki-NG-based IEEE 802.15. 4 TSCH Throughput Evaluation. In Proceedings of the Korean Institute of Information and Commucation Sciences Conference (pp. 577-578). The Korea Institute of Information and Commucation Engineering.
- [6] García Algora, Carlos & Alfonso Reguera, Vitalio. (2018). Experimental evaluation of IEEE 802.15.4 TSCH on a 6TiSCH network. 39. 70-78.