Sensors Lab Conference

Gabriele Paris
Physic department
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

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

[**ThroughputEvaluation**] 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.

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.
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.

A. Only TSCH MAC layer

The basic setup for the following analysis is a root (coordinator) node that sends packets to a leaf node at a rate of 1 packet of 4 bytes per second.
Note that the root node has no event routine related to data reception. In the same way, the leaf node has no code which permits data transmission.

Both nodes are configured to measure the energy consumption by the Energest[1] module available in Contiki-ng[2]. The energy consumption is measured referring to the following formula:

$$E_{tot} = \sum_{c \in comp}^{N_c} E_c = \sum_{c \in comp}^{N_c} I_c \cdot V_{cc} \cdot t \quad (1)$$

Where V_{cc} is the supply voltage, fixed as a constant at the value $3.3V$.
And the single current consumption is obtained from the table I[**EnergyConsumption**].
To perform the first test only the TSCH MAC layer is enabled

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
CURRENT DRAW FOR THE TX AND RX STATES USING THE CSMA PROTOCOL AND THE CURRENT DRAWN FOR TX AND RX TIME SLOTS USING TSCH AS MAC PROTOCOL, BOTH WHEN USING THE CC2538 RADIO.[**EnergyConsumption**]

the rest of the stack is unused, including the security link-layer. Physically speaking in this first test there is no need to measure the energy consumption variation given the distance, therefore the physical setup of the experiment consists of two Zolertia Remote RevB boards placed at a fixed distance of approximately 10 cm.
At first, we have measured the energy consumption relative to the leaf node, which is only listening for data. In figure 1 is possible to see the energy consumption during a period of around 27 minutes. What is possible to notice is that most of the energy is used in deep sleep mode.
Then leaf node in this configuration requires only to listen, is therefore understandable that the majority of the time is spent in the lowest energy configuration, hence is important to notice that as referenced in table I the deep state mode is the less energy consuming component, the board is prone to switch in this state as soon as possible to save battery.
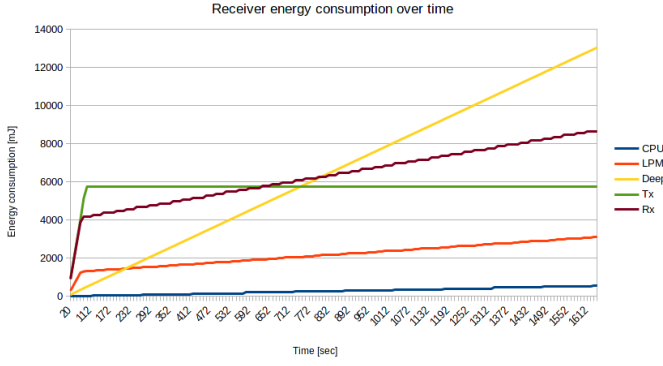
---

[1] https://github.com/contiki-ng/contiki-ng/wiki/Documentation:-Energest
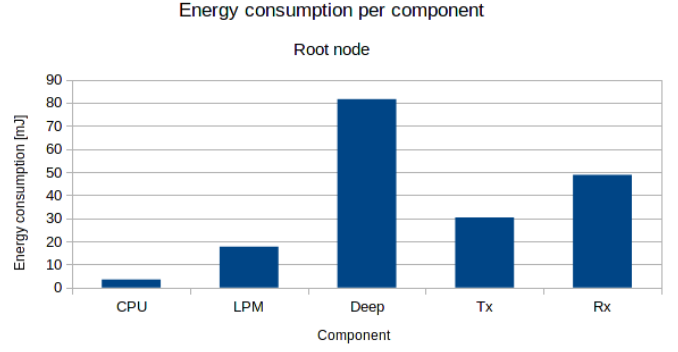[2] https://github.com/contiki-ng

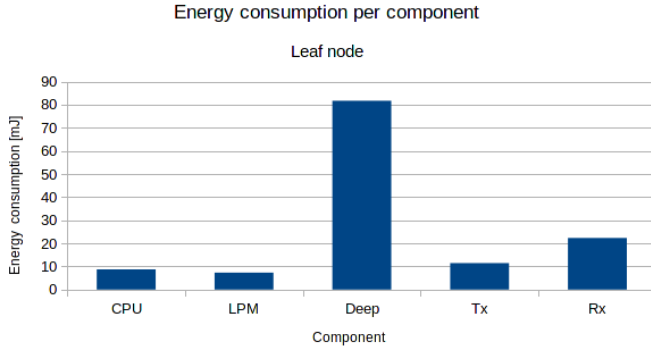Fig. 1. Zolertia leaf node energy consumption over time.

Fig. 2. Zolertia leaf node average energy consumption per component.

In figure 2 is shown the average energy consumption per component.

Then we have performed the same experiment regarding the root coordinator node, in the same configuration for the leaf node we have measured the energy consumption over time 3 and per component 4.
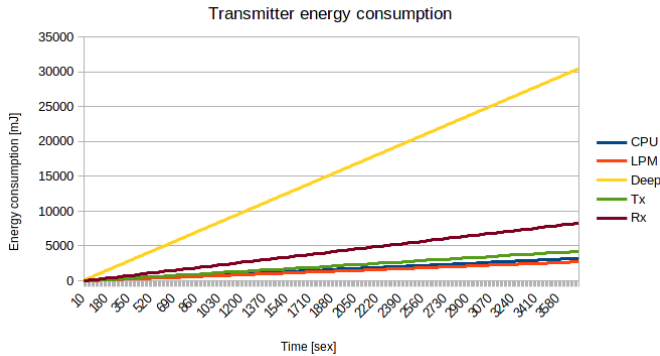

Fig. 3. Zolertia root node energy consumption over time.

The first thing that is possible to notice is that the energy consumption of the transmitter continues to rise, due to the program on the root node itself, that as explained in the section III is required to send data to the leaf node.

Fig. 4. Zolertia root node average energy consumption per component.

As before analyzing the breakdown of the energy consumption shows us that the state of deep sleep is the favorite for the component. What is interesting to notice is the conspicuous increase of energy consumption in the root node in comparison to the leaf node, in the table II have been reported a comparison between the two nodes average energy consumption per component.

Board	CPU	LPM	Deep	Tx	Rx
Leaf	8,738 mJ	7,337 mJ	81,697 mJ	11,430 mJ	22,326 mJ
Root	3,503 mJ	17,716 mJ	81,553 mJ	30,352 mJ	48,792 mJ

TABLE II
ENERGY CONSUMPTION PER COMPONENT PER NODE ON AVERAGE.

As the last measurement, the boards have been programmed to run the same code, this receives and sends messages with an interval of around 8 seconds.
The boards have been running for a half an hour, the total consumption has been recorded and shown in table III.

Board	CPU	LPM	Deep	Tx	Rx
Leaf	2127 mJ	68768 mJ	18858 mJ	227471 mJ	220308 mJ
Root	2127 mJ	68009 mJ	18662 mJ	225109 mJ	217921 mj

TABLE III
ROOT AND LEAF NODE RUNNING THE SAME PROGRAM, POWER COMPARISON PER COMPONENT.

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 similar to the previous one. Two boards 10cm apart from each other 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.
As before is reported the time cumulative energy consumption regarding leaf and root node.
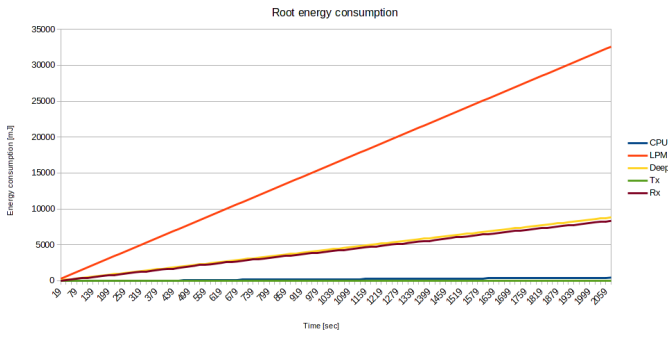
Fig. 5. Zolertia root node energy consumption over timem full 6TiSCH stack.
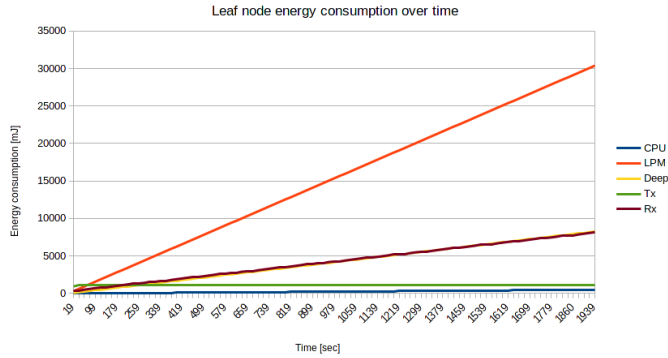

Fig. 6. Zolertia leaf node energy consumption over timem full 6TiSCH stack.

After a time analysis we provide a breakdown of the single components energy consumption. The plotted data is available
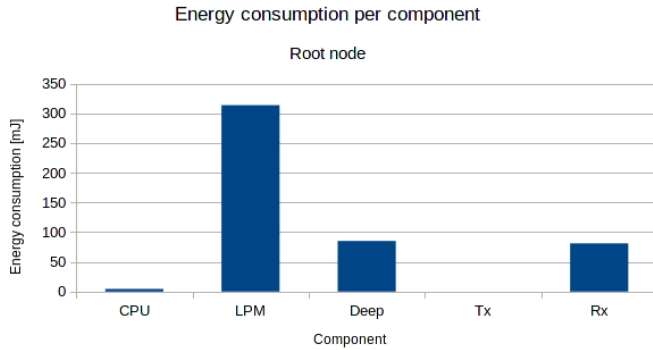

Fig. 7. Zolertia root node average energy consumption per component.

at the table IV.

Board	CPU	LPM	Deep	Tx	Rx
Leaf	4,739 mJ	313,51 mJ	85,145 mJ	2,135 mJ	80,812 mJ
Root	4,417 mJ	313,708 mJ	85,145 mJ	0 mJ	81,106 mJ

TABLE IV
ENERGY CONSUMPTION PER COMPONENT PER NODE ON AVERAGE.

Note that the 0 energy consumption on transmission is not due to absence of transmission on the root node, but by the
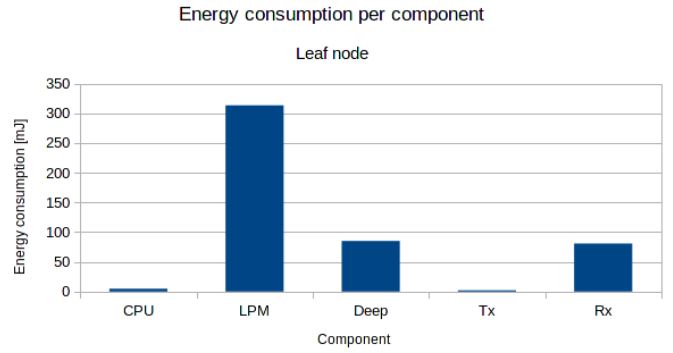

Fig. 8. Zolertia leaf node average energy consumption per component.

fact that very few packets are transmitted hence the number is too small to be reported.

C. Conclusions

In the previous subsections, we have tested the energy consumption per component at first by only using the TSCH MAC layer, and then by enabling the full 6TiSCH stack.
The following table reports the total energy consumption of leaf and root note in the two scenarios.

Board	Leaf	Root	
Mac only	131,518 mJ	181,916 mJ	
Full stack	486,341 mJ	484,376 mJ	
	269,79%	166,26%	**Increase**

TABLE V
TOTAL AVERAGE ENERGY CONSUMPTION OF LEAF AND ROOT NOTE IN THE TWO SCENARIOS.

As it is possible to see from the table V the energy consumption of the two nodes following the adding of the full stack increase drastically, this is normal and predictable by the fact that adding more layers on top of the single MAC add more computational requirements to the system, so a major time spent in less energy-saving states.

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 seperation and
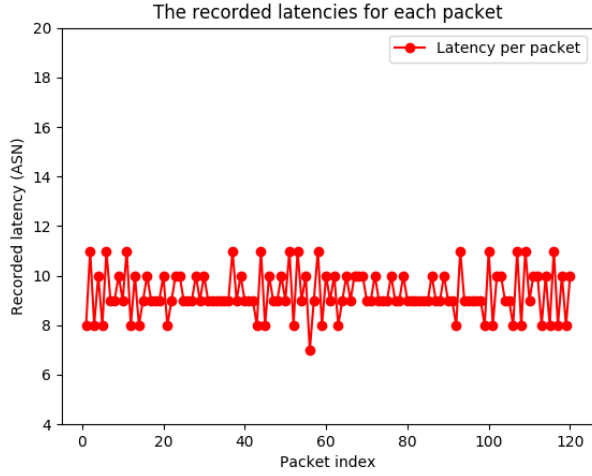
Fig. 9. Latencies for packets sent between two nodes at 1 meter seperation.
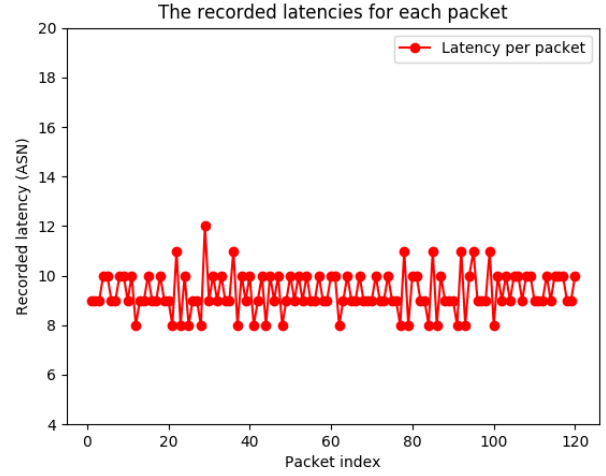

Fig. 10. Latencies for packets sent between two nodes at 5 meter seperation.

nodes with 5 meters seperation. 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 seperation 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 seperation, 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 9. 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 11, 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 rougly 9.33 slots (but this could be within margin of error - we should look at retransmission rates to verify V).
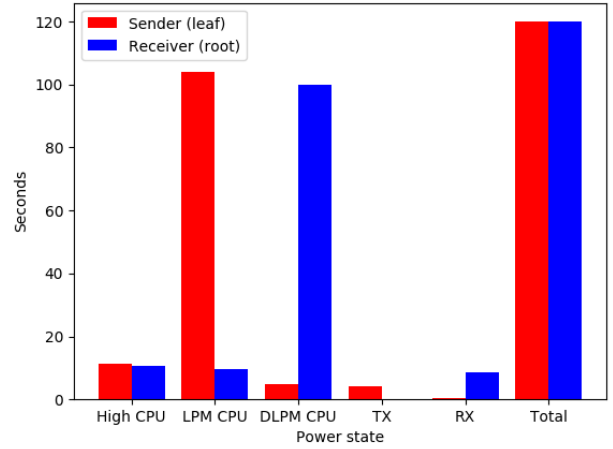

Fig. 11. The time a node spends in each power state at 1 meter seperation.

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] 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.
[2] 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.
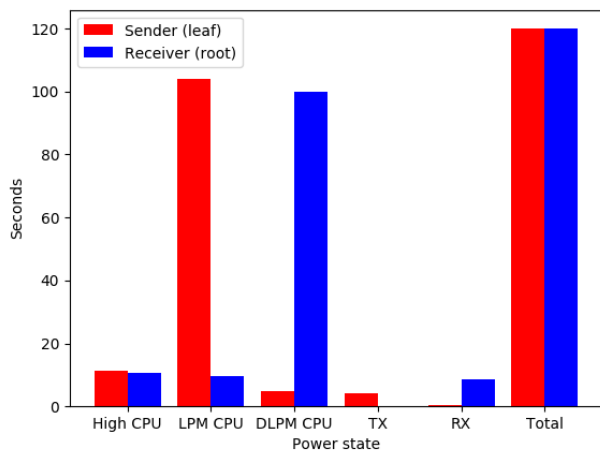
Fig. 12. The time a node spends in each power state at 5 meters seperation.