Poster: ContikiMAC, some Critical Issues with the CC2420 Radio

Marie Paule Uwase
Vrije Universiteit Brussel
Université Libre de Bruxelles
muwase@etro.vub.ac.be

Maite Bezunartea
Vrije Universiteit Brussel
mbezunar@etro.vub.ac.be

Jacques Tiberghien
Vrije Universiteit Brussel
Université Libre de Bruxelles
jgtiberg@vub.ac.be

Jean-Michel Dricot
Université Libre de Bruxelles
Jean-Michel.Dricot@ulb.ac.be

Kris Steenhaut
Vrije Universiteit Brussel
ksteenha@etro.vub.ac.be

Abstract

During experimental evaluations of RPL on a network with some 20 Zolertia Z1 motes running Contiki version 2.6, inconsistent Packet Delivery Ratios (PDR) and message latencies were frequently observed. Replacing ContikiMAC by NullRDC brought consistency, proving that the problem had to be found in the Radio Duty Cyling (RDC) protocol. Two implementation issues of ContikiMAC were found guilty. The first is the timing of the inter-packet interval related to the interval between two successive Clear Channel Assessment (CCA) probes. The second is the choice of the value of the CCA threshold. These issues will be analyzed in this poster, a satisfactory work-around has been implemented and some to be tested alternatives are proposed.

* Keywords ContikiMAC, Inter-packet interval, CCA threshold, IEEE 802.15.4, Zolertia Z1

1 ContikiMAC timing vs. IEEE 802.15.4

To test unicast transmission with ContikiMAC, a Z1 mote sent, on average once per second, a data packet to a second Z1 located a few meters away. All radio traffic was monitored by a Texas Instruments Packet Sniffer with a CC2531 dongle. The observed time between successive, unacknowledged, packets was 2565μs, the duration of one packet being 1760μs, implying that the interval between packets was 805μs, double of the 400μs specified in [1].

This excessive delay explains why long sequences of missing packets were observed in an otherwise flawless transmission link: when one of the CCAs detects a frame, everything works fine, but when the two CCAs fall just between two frames, the receiver goes back to sleep. If sender and receiver have identical clocks, either all frames are detected (PDR = 100%) or none is detected (PDR = 0%). When the clocks differ slightly, the sampling by CCAs shifts from one frame to the next one and, in between, packets are not detected (see Figure 1).

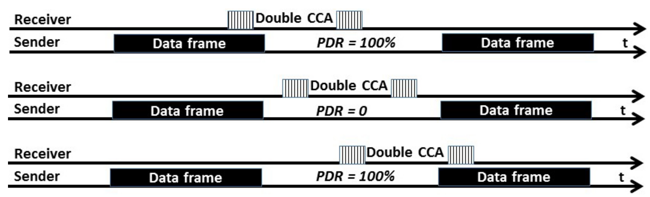

Figure 1. Effect of slightly different clocks on the PDR (the receiver clock is faster in this example).

The number of lost packets depends on the accuracy of the clocks. For instance, small differences between sender and receiver clocks cause long episodes of packet loss, and large differences cause short episodes. Repeating the tests with different pairs of motes confirmed that the length of the sequences of missing packets varied with the involved motes.

How can these inter-packet intervals be reduced? Several possibilities have been identified.

First, the CC2420 transmitter needs to be calibrated before it starts transmission and this takes 192μs but can be reduced to 128μs [2]. In the driver, the code to shorten calibration is included but commented out. Activating this code saves 64μs.

The default implementation of the CSMA MAC layer protocol offers an additional opportunity to reduce inter-packet intervals. The Carrier Sense functionality is implemented through 6 CCAs before the transmission of a new packet is started and through one additional CCA before any retransmission required by ContikiMAC. Skipping this last CCA by setting WITH_SEND_CCA to false reduces the inter packet interval by another 260μs and did not cause any observable problem in our experiments.

A frame is retransmitted if no ACK is received. When a correct frame is received, the CC2420 generates itself an ACK which is sent after 192μs. After the sender mote has sent a frame, it waits a time defined by the INTER_PACKET_INTERVAL parameter before checking for

International Conference on Embedded Wireless Systems and Networks (EWSN) 2016
15–17 February, Graz, Austria

ISBN: 978-0-9949886-0-7

the ACK. The default value for this parameter is 400μs, but its name is misleading, as this is only a part of the true inter-packet interval. We prefer to call it BEFORE_ACK_DETECT_WAIT_TIME and, considering the timing of receiving an ACK message, we assign it a value of at least 335μs or preferably (see further) 366μs (one clock tick on the Z1 is 30.5μs). Unfortunately, the minimum achievable inter-packet interval is 463μs. This with the shortest calibration delay, no CCA before sending each frame and only 335μs for the delay before probing for an ACK. This is well above the values specified in [1]. This implies that ContikiMAC as proposed cannot be implemented on IEEE 802.15.4 radio links. However, even with measured inter-packet intervals of 490μs we did not observe any sequences of missing frames. Increasing the interval between CCA probes would be the right solution, and is being explored, but this might jeopardize other protocols such as RPL, as the minimum packet size would also be increased and might exceed the size of DIO and DAO messages.

2 The CCA threshold

In Z1 motes, the default CCA threshold is -77dBm while the CC2420 receiver can correctly receive packets at -90dBm. How does this impact the operation of ContikiMAC?

As incoming packets are detected by means of the CCA, one would expect that no packets with a RSSI below the CCA threshold can be received. This was checked experimentally by varying the transmit power and the distance. Indeed, below the CCA threshold level the PDR drops dramatically, but remains close to 25% on a university campus and to 6% in the countryside far from any 2.4GHz sources. An inventory of lost and received packets confirmed by sniffer readings showed that, below the CCA threshold, long sequences of lost packets alternated with sequences of a few consecutive well received packets. We concluded from this that the receiver wakes up occasionally due to external unrelated radio sources, receives one of the packets that are being continuously transmitted, acknowledges it, and remains awake to receive the next packets, until the transmitter buffer is empty. The minimal RSSI for detecting incoming packets with ContikiMAC can be reduced by lowering the CCA threshold, but this visibly increases the losses due to collisions.

For signals levels close to the CCA threshold, the detection of an ACK can be jeopardized, causing packet duplication and useless radio traffic. To minimize the inter-packet interval, the detection of an ACK is done in two steps: first, at the time an ACK should arrive, ContikiMAC checks for any incoming signal. If none is detected it starts immediately the retransmission of the frame. Otherwise, after an additional delay, it checks if the received signal was indeed an ACK. If not, a collision status is reported. To detect an incoming signal, a Boolean OR function is applied to three status bits of the radio. The first one is the complement of the CCA, and becomes true when any signal above the threshold is being detected. The second becomes true when the start of an IEEE 802.15.4 frame has been detected, and the third when an entire packet has been loaded in the receiver FIFO buffer. Depending on the moment this OR function is evaluated, the ACK needs to have a RSSI higher than -77dBm or -90dBm to be detected (see Figure 2).

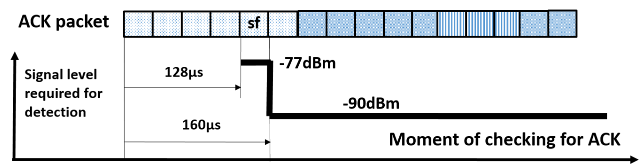

Figure 2. ACK detection levels.

This motivated us to choose 366μs rather than 335μs for the BEFORE_ACK_DETECT_WAIT_TIME when low energy levels are used. The effects of eliminating entirely the CCA check in the detection of an ACK are being tested.

3 Related Work

Very few papers report on experimental studies of the ContikiMAC protocol. From an energy point of view we established that it is better than X-MAC and LPP [3], but when broadcasts are required, as in the trickle timer used for RPL, some incompatibilities have been reported [4] and work-arounds proposed.

4 Conclusions

Our experiments demonstrated that with signals below the CCA threshold level, ContikiMAC becomes unreliable and that even at higher levels, careful adjustments of the default timing settings are required.

5 References

[1] Adam Dunkels. The contikimac radio duty cycling protocol. *SICS Technical Report T2011:13*, 2011.

[2] TexasInstruments. Datasheet: Smartrf cc2420 2.4 ghz ieee 802.15.4 zigbee-ready rf transceiver. *Obtained from www.ti.com/product/cc2420*, 2014.

[3] Marie Paule Uwase, Maite Bezunartea, Tanh Long Nguyen, Jacques Tiberghien, Kris Steenhaut, and Jean-Michel Dricot. Experimental evaluation of message latency and power usage in wsns. In *Proceedings of IEEE International Black Sea Conference on Communications and Networking*, pages 69–72. IEEE, May 2014.

[4] Milosh Stolikj, Thomas M. M. Meyfroyt, Pieter J. L. Cuijpers, and Johan J. Lukkien. Improving the performance of trickle-based data dissemination in low-power networks. In *Proceedings 12th European Conference on Wireless Sensor Networks*, pages 186–201, February 2015.