Bluetooth Communications in the Internet of Things

- Marcio Jose de Menezes - 3109694

1. Introduction and experiment setup

The lab experiment consisted of several Bluetooth devices (Raspberry Pi) transmitting beacons and a sink device logging the transmitted beacons. In order to be able to evaluate the loss for different transmission rates, we have transmitted at different rates along the experiment. To be able to identify the different setups we have used along the experiment, we used 2 2-bytes fields of the beacon packet to write a code to identify the setup. The experiment goal is to evaluate the wireless channel limitations observing packet loss for different transmission rates. The log file has information about all packets received by the sink and therefore it is necessary to generate a data parser to extract the relevant information. For post-processing we have developed a python script which is responsible to generate statistical information about the conducted experiment as well as to generate some plots for evaluation.

2. Collected data and analysis

The post-processing script generates the file packetAnalysis.log along with a plot representing the network traffic, our node traffic and the time slot of each setup. The experiment has recorded 44640 packets over 6283.22s. Our group has configured 16 distinct setups with the following ids [11, 53003, 12, 13, 14, 15, 16, 17, 18, 20, 21, 23, 24, 25, 26, 27]. Unfortunately we had an issue with our device where the transmission rate became stuck at 1.28s for setup ids [11, 53003, 12, 13, 14, 15, 16, 17, 18]. Setup id 53003 was set by mistake and likely our device misbehavior occurred due to garbage being written on the configuration registers. The work around to fix this problem is to reset the Bluetooth interface, what can be accomplished with hciconfig hci0 down and then hciconfig hci0 up. Fortunately it is still possible to extract some interesting information from the logged data, the setup ids 11, 18, 21 and 24 won't be considered as we interrupted transmitting during these setups. Table 1 presents statistical data regarding the experiment.

Setup Id	TX rate	Sampled time	RXd Packets	Estimated Loss
53003	1.28s	200s	137	11.6129%
12	1.28s	255s	170	14.5729%
13	1.28s	202s	137	12.7389%
14	1.28s	291s	184	18.9427%
15	1.28s	250s	158	18.5567%
16	1.28s	233s	145	20.3297%
17	1.28s	597s	395	15.0538%
20	0.10s	88s	683	11.8710%
23	0.10s	90s	752	14.3508%
25	0.25s	124s	399	18.4049%
27	0.10s / 5s	60s	12	0 %

Table 1: Experiment statistics according to setup

Analyzing the experiment statistics, we observed that for a given transmitting rate, there was often a discrepancy on the estimated packet loss. We attributed this discrepancy to the fact that the network traffic was changing dynamically as other groups started transmitting beacons. Image 1 display the network traffic over time, stacked together with the packet traffic originated by our node.

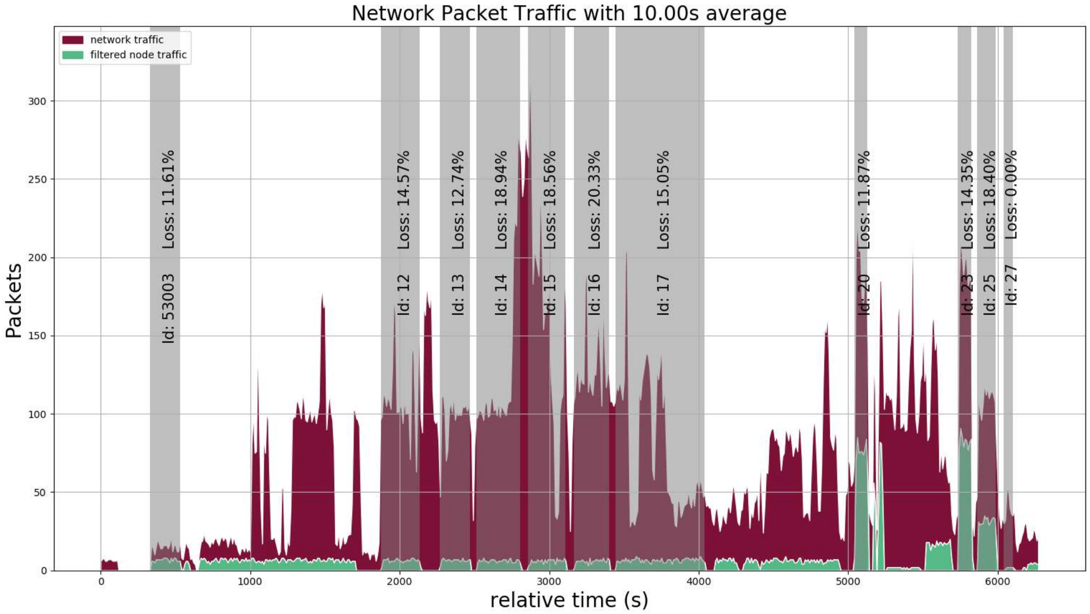

Image 1: Network traffic along with filtered node traffic

We can observe in Image 1 that for a given transmitting rate (e.g. 1.28s), the packet loss often increases whenever the network traffic grow. For instance, setup id 53003 recorded on the beginning of the experiment had loss around 11.6% when the average traffic on the network was below 10 packets/second. As the volume increased, setup ids 14 and 15, with same transmitting rate as 53003 experienced loss around 18%.

Setup 27 was configured with minimum transmitting rate of 100ms and maximum 5s, we can observe that the device only transmitted at 5s and there were no lost packets. Also the network traffic observed during this setup was very modest if compared to other time slices. However this setup only ran for 60s and therefore we cannot say much because the amount of collected data is not enough for any conclusion.

In some instances the behavior is also not exactly as expected, for example, setup 25 ran with transmission rate of 250ms and on better network conditions than setup 20 which had tx rate of 100ms and higher network packet traffic and yet, setup 20 presented lower loss (11.87%) in contrast to (18.4%) experienced by setup 25. Naturally we expected the opposite behavior, since higher network packet volume increases the probability of transmitting devices to interfere between themselves. Nevertheless, this adverse behavior could still be explained by the fact that the experiment was conducted in a dynamic environment where people could move freely and modify the channel conditions.

In conclusion, we can state based on the recorded data that as network volume grows, the packet loss normally grows, this growth due to greater probability of collision is smooth, what is expected since BLE features adaptive frequency hopping over its 37 channels, making this protocol very robust in respect to interference due channel collision.