2015 3rd International Conference on Future Internet of Things and Cloud

Energy Consumption and Latency in BLE Devices under Mutual Interference: An Experimental Study

Jan Jaap Treurniet*, Chayan Sarkar†, R. Venkatesha Prasad†, Willem de Boer*
*Technolution B.V., The Netherlands
†Delft University of Technology, The Netherlands

***Abstract*—Bluetooth is a widely used technology for short range communications. Limited device density and frequency hopping based communication usually eliminates the chances of mutual interference among independent Bluetooth piconets. However, with the advent of Internet of Things, there is a sharp increase in Bluetooth-equipped devices, especially in wearable devices. This gives raise to multiple collocated piconets, thus increasing the mutual interference leading to performance degradation of Bluetooth communication. In this work, we study the latency and energy consumption by *Bluetooth low energy* (BLE) devices under the influence of mutual interference, which has not been studied in the literature. Based on the experimental results involving 32 BLE devices, we investigate the influence of mutual interference and develop models for energy consumption and latency. These models can be utilized in future BLE enabled devices.**

I. Introduction

Bluetooth low energy (BLE) is a technology for wireless personal area networks, which was added to the Bluetooth Core Specification 4.0 [1]. It is designed for applications where lowering energy consumption is paramount. It operates in the same 2.4 GHz band like the Bluetooth classic. This frequency is also shared by various other wireless technologies, e.g., Wifi, ZigBee, etc. Thus, coexisting communication using these technologies interfere with Bluetooth communication. A number of existing works has investigated the effects of these potential interfering communication technologies [2]. However, these technologies do not use frequency-hopping mechanisms like Bluetooth. As a result, the interference from other devices operating in the same band on Bluetooth is minimal.

Bluetooth devices communicate by forming piconets, where a device posing as master device controls the simultaneous communications within a piconet. As a result, there is no chance of collision among the devices within a piconet. However, mutual interference can occur amongst the devices belonging to different piconets. Small communication range and low device density within a close proximity reduces the chance of mutual interference. It is estimated that over 30 billion wireless-connected devices to be part of the Internet of Things (IoT) in 2020 [3], and it is assumed that a large proportion of these devices will be Bluetooth enabled. In fact, people are using more and more Bluetooth devices in their daily life, for example in smart phones, tablets, wireless headphones, smart watches, etc. Furthermore, future applications that can employ BLE for location-based services, for example, using iBeacon technology, in payment systems, in body area sensor networks [4], etc.

In crowded environments, such as public transport vehicles, a large number of Bluetooth enabled devices can coexists. These devices, owned by individuals will be operating independently (in different piconets), e.g., smart phones communicate with smart watches, headphones, automated journey tracker (devices within a vehicle), etc. Intuitively, we can expect high levels of interference amongst the devices that will influence the performance. Therefore, in this work we investigate the impact of interference from other BLE devices on energy consumption and latency of communication in a crowded environment.

Most of the Bluetooth enabled devices run on battery power including smartphones. Energy consumption by various applications is a major concern for smartphone users. Smartphone batteries have limited capacity, while wireless data transmission tends to be heavy in terms of energy consumption. A higher density of Bluetooth-enabled nodes, each wanting to communicate, influences energy consumption of all the devices. As a result, any application such as, localization, automated journey tracking, etc., will be less appealing for the users if the energy consumption by these applications are not kept absolute minimum. In this regard, analysis, simulations and experiments are needed to determine how much energy is required by a BLE device under various circumstances.

There are some studies that measure energy consumption of Bluetooth devices under the interference of WiFi and/or ZigBee and vice versa [5]. However, no previous work considered the influence of mutual interference from BLE devices on energy consumption and latency. In this article, we report our experiments with 32 BLE devices and our findings. Specifically, the main contributions of this work are twofold. Firstly, we study the energy consumption and latency of a BLE device for its various operations. We estimate the influence of simultaneous connection request by multiple devices on the energy consumption; and we also consider influence of interference due to parallel Bluetooth based communication (e.g. headphones, wristbands etc.) on the increased energy consumption. To the best of our knowledge, this is the first work that investigated energy consumption and latency of BLE devices under mutual interference. Secondly, we have developed models for energy consumption and latency for BLE, which could be used by application developers to make their applications energy efficient.

The rest of the article is organized as following. In the next section, we provide background information that is necessary to set the context of this work, i.e., we briefly discuss some of the properties of BLE and existing literature about energy consumption and latency by a Bluetooth device. In Section III,

978-1-4673-8103-1/15 $31.00 © 2015 IEEE
DOI 10.1109/FiCloud.2015.108

we describe the controlled experiments and their results that are conducted in simulation. Further, we describe the setup for measuring energy consumption by a BLE device, and measurement results. Based on the simulation and experimental results, we developed energy consumption and latency models and validated this using measurement data (Section IV). Finally, we conclude the article in Section V.

II. Background

To understand the context of the experiments, we need to understand some of the basic specifications and functionalities of the BLE protocol. Thus, we provide a brief overview of BLE. For further details, we refer to [6], [7], [8].

A. Basics of BLE

BLE was first introduced in 2010 with version 4.0 of the Bluetooth Core Specification [1]. BLE is not compatible with Bluetooth Classic (BR/EDR), but many devices, such as smartphones and tablets, are compatible with both BR/EDR and BLE. BLE is an ideal technology for typical Internet of Things (IoT) applications, where both the available resources and data to be transmitted, are limited. This means BLE is not designed to replace BR/EDR. Both protocols will be used, because both protocols have their own focusses. One of the common features for both BR/EDR and BLE is that the data communication takes place in infrastructure mode, i.e., a device – known as master – coordinates the communication with one or more devices (up to 7 slave devices). As a result, chances of collisions during data transmission and interference from other slave devices are highly insignificant. The special (differentiating) features of BLE are described in the following.

1) Low energy aspects: As the name suggests, the goal of Bluetooth low energy is to operate at low energy budget. To achieve this goal, a number of measures have been incorporated. Some of the important ones are: (i) the protocol is designed to operate asymmetrically, (ii) the parameters of the protocol, such as advertising and connection intervals, can be tuned specifically for an application to achieve the lowest energy consumption while still offering acceptable latencies, etc.

2) Physical and link layer: BLE uses the unlicensed 2.4 GHz ISM band as the Bluetooth classic. However, the band is divided into 40 channels of 2 MHz as compared to 80 channels in classic. Out of these 40 channels, 3 channels are used for advertising messages and the remaining 37 channels are used for data transmission using frequency-hopping scheme. The maximum bit rate is 1 Mbps, and the maximum transmission power is 10 mW. At the maximum transmission power, the BLE transmission range can be more than 100 m. During connection phase, the devices try to connect at constant intervals, which is called the *connection interval*. This value can be set anywhere between 7.5 ms and 4 s. A channel hopping scheme switches to a new channel at every connection interval. Since many other technologies, like WiFi and ZigBee also use this band, the channel hopping scheme mitigates interference from them.

3) Application layer: BLE devices can communicate using two topologies: the broadcast-observer topology and the connection topology. For our experiments, we consider only the connection topology as broadcast-observer is bound to use more energy during device discovery. We describe the different phases used during connection: device discovery, connection setup and data transfer.

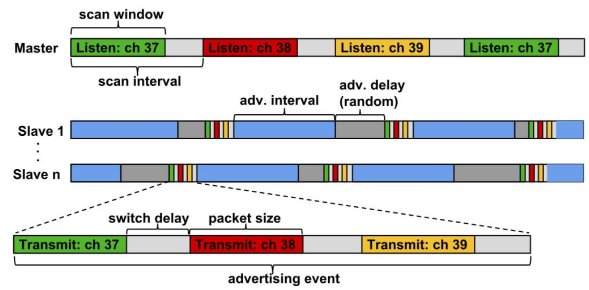


Fig. 1: Timeline of the BLE device discovery process.

Discovery and connection: Three channels are reserved for advertising, i.e., for device discovery. A timeline illustrating the discovery process is shown in Fig. 1. An advertising device transmits the advertisement packet three times on three different channels (Channels 37 to 39) within an advertisement event. The advertisement packet may contain information about the device name or supported services. A listening (master) device discovers devices by listening to these advertisements. After transmitting an advertisement packet, the device listens for a short period (switch delay) in which the master device sends a connection setup request. Two successive advertisement events are separated by a predefined advertisement interval. To avoid repeated collision among simultaneous advertising devises a random advertisement delay is added to the advertisement interval. The length of a scan interval (for the duration a listening device listens on a particular advertising channel) is at least equal to the sum of the lengths of an advertising event and an advertisement interval. The listening device changes the scanning channel after each scanning interval.

Connected phase: During a connection, one slave device (mostly but not necessarily) functions as a Generic Attribute Profile (GATT) server and offers services containing characteristics to other device(s). These characteristics can be used to transfer data packets containing a maximum of 20B of payload, initiated by either the GATT server or client (depending on the characteristic).

B. Related work

We briefly discuss some of the highly relevant works on energy consumption and interference for Bluetooth.

Liu *et al.* [9] developed a model for the analysis of device discovery in BLE in terms of latency and validated this with a simulation. This is done for a situation with only one advertising device. Later, the authors developed a model and did extensive energy measurements for the BLE device discovery process while varying advertising and scanning parameters [10], but they still did not take interference into account. Chong *et al.* [11] developed a model for throughput and energy consumption for ZigBee Network under the presence of Bluetooth Classic interference and validated the model with a simulation. Stranne *et al.* [12] performed experiments to validate the model for the influence of mutual interference on the throughput of Bluetooth classic from [13]. Howitt [14] developed a model for interference between independent Bluetooth Classic connections and verified the model for the case with one interferer by an experiment. Goldenbaum *et al.* [15] performed a study of the general trade-off between energy consumption and robustness in multi-antenna sensor networks with interference but do not perform any experiments. Gomez *et al.* [16] developed a model for BLE throughput based on the bit-error rate. The model is verified using a simulation. Kindt *et al.* [17] presented a very extensive energy model covering all operating modes of BLE and verified it using an experiment. Again, interference is not taken into account. Siekkinen *et al.* [18] measured energy consumption for BLE and ZigBee.

Summarizing the existing work, we conclude that energy consumption of Bluetooth Classic devices under mutual interference has been measured, as well as the energy consumption of BLE under WiFi interference. No prior investigations considered the influence of mutual interference from BLE devices on energy consumption and latency.

III. Simulations and Experiments

The device discovery process for BLE, as described in Section II-A3, is fairly simple and we want to explore this for a large number of devices. Thus, we explored the device discovery latency for a varying number of devices using simulation. As wireless data transmission is affected by many physical and environmental parameters, it is impossible to be sure about its behavior without a real-life validation. For this reason, we also performed extensive experiments with BLE devices.

A. Simulation details

The simulations are conducted in MATLAB. First, a list of transmitting intervals is generated for every advertising device. Overlapping intervals between devices are removed from the list of intervals. For the scanning device, a list of scanning intervals is generated. The transmitting intervals that are contained in a scanning interval are then filtered out. From the resulting intervals, we compute two important metrics: average latency and reception rate. The average latency is the time from the start of the first scan window until the moment the first packet from a certain device is received. The reception rate is the percentage of transmitted packets that is actually received by the scanning device.

The implementation assumes that there is one scanning device and a number of peripheral devices that are all advertising with the same interval and packet size. If two packets collide on the same channel, both packets are discarded. If a packet does not collide and is within a scanning interval, it is always assumed to be received i.e., we assume ideal channel conditions.

B. Simulation results

We ran the simulations using the settings as listed in Table I. In Figure 2, we can see how the discovery latency increases with the number of simultaneously advertising devices, and the discovery latency is less with smaller advertising interval. However, with over 40 devices, the shortest advertising interval no longer results in the fastest average discovery. This is because, with smaller advertising intervals, devices advertise too frequently. This results in a lot of collision at the scanning device due to simultaneous advertisement packets. As a result, the average discovery latency increases. Figure 3 shows the reception rate for various advertising intervals for varying number of nodes. The lower the advertising interval, the reception rate decreases at a faster rate with increasing number of devices.

Variable	Value
Advertising interval	20,60,100,200ms
Packet length	376μs
Switch delay	150μs
Scan interval	100ms
Scan window	100ms
Number of devices	1 to 100 (in steps of 5)
Number of runs	10000
Simulation length	5 s

TABLE I: Parameters used in simulation runs.

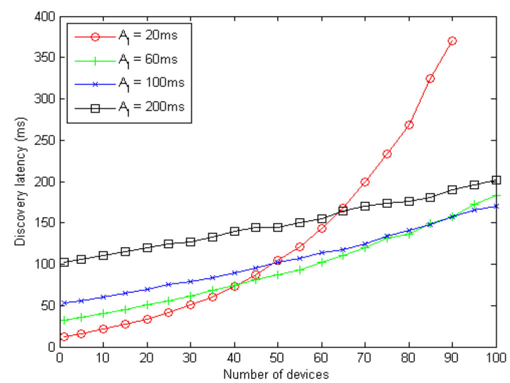


Fig. 2: Average latency in device discovery for various advertising intervals (A_I) and various number of device.

C. Experimental setup

The experiments were performed using BLE modules from Bluegiga [19]. These modules combine 8051 microcontroller with a Bluetooth transceiver. They are available in the form of a relatively inexpensive USB stick and as a development

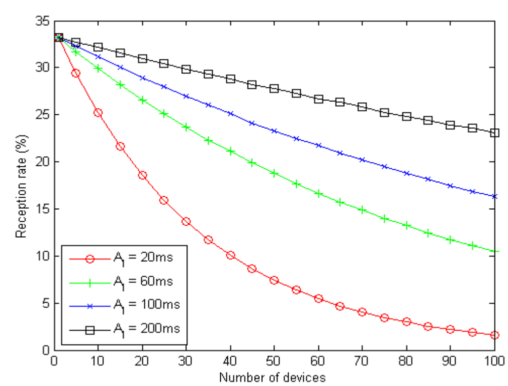


Fig. 3: Average reception rate for various advertising intervals (A_I) and various number of device.

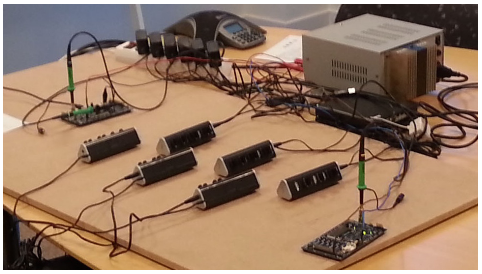


Fig. 4: Experimental setup.

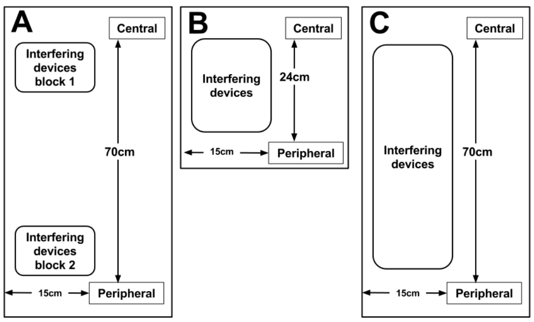


Fig. 5: Device layouts used in the experiments.

kit allowing for easy energy measurements. The modules can be programmed using BGscript, which is a module-specific scripting language.

There are two 'devices under test': a central (master) and a peripheral device (slave). Both are powered via a power supply and connected to an oscilloscope for power measurement. In addition, the central device is connected to a Matlab application via USB. It is also connected via a GPIO pin to a separate oscilloscope channel for sending trigger pulses.

1) Interference generating devices: The interference generating devices work autonomously and are only connected to a power supply. By default, all the devices are in advertising mode. A control application is used to switch them to data transfer mode when needed. In data transfer mode, the devices transmit at the maximum achievable throughput, which is around 100 kbps [17], [20]. We used up to 15 pairs[1] of interference generating devices.

2) Energy measurement: The BLE113 development board from Bluegiga is equipped with a current measurement circuit. This circuit consists of a shunt resistor and an instrumentation amplifier, which measures the current flowing to the module and outputs this as a voltage. This voltage is measured and the current can be calculated from this voltage using the expression $I = \frac{3.3-V_0}{30}$, where I is the current flowing to the module and V_0 the measured voltage. The module is powered via a $3.3V$ LDO, which means its voltage is constant at $3.3V$. Energy consumption is calculated by trapezoidal numerical integration over the current measurement multiplied by the voltage.

[1]One pair consists of one central and one peripheral device.

3) Device layouts: For our experiments, we have used three different device layouts (as shown in Fig. 5). The layouts vary based on the locations of the interfering devices and the distance between the 'devices under test'. The devices marked as central and peripheral are under test and their energy consumption is measured.

- In **Layout A**, all pairs of devices communicate over an approximately equal distance. This means all received signals will have around the same signal strength. There have been two cases - all the central devices (i.e., 15 devices) are placed in Device-block 1 and all the peripheral devices (i.e., another 15 devices) are placed in Block 2 and vice-versa. Please note that a peripheral device always communicate with a predefined central device.
- In **Layout B**, the interfering pairs communicate over a very small distance and all devices are very close to each other. This represents a situation where multiple interfering devices are communicating over a short distance, while situated between two devices communicating over a longer distance. An example of a similar situation is a smartphone communicating with a smartwatch, while another smartphone is communicating with a beacon in a vehicle.
- **Layout C** is similar to Layout B, but with the devices spread out over a larger area.

4) Channel limitation: During the experiments, we only used channels 1-8 for data transmissions instead of all 37 channels (used by the Bluetooth 4). This way, we increased the influence of interference for the measurements and we made sure that it can have a significant impact on latency and energy consumption without needing an excessive amount of interfering devices. The number of advertising channels is not limited though, i.e., all the three advertising channels are used.

5) Measurement procedure: For determining the energy consumption of different operations, a measurement sequence was generated containing eight phases (Fig. 6). The sequence is started by a command from the Matlab script to the central device via USB. The central node controls the rest of the sequence and sends trigger pulses to the oscilloscope. After the measurement sequence, the oscilloscope data is sent to the Matlab script, which splits the current measurement data using the trigger pulse data.

- When no measurement is active, sleep mode of the central node is disabled, to be able to receive the start command. When the sequence is started, sleep cycle is enabled to minimize the current consumption during measurement.
- In the **Discover** phase, the central device is listening for advertisement messages from the peripheral devices. This phase ends when the first advertisement packet is received.
- In the **Connect** phase, the central device waits for another advertisement message, and sends a connection request immediately after receiving this. To check if the connection is really established, a read request is sent to the peripheral. When a response to this request is received, the connect phase ends. When no message is received after 6 connection intervals, the connection is considered to be lost. The probable reason is that the connection request packet may have faced a collision and thus was not received by the peripheral. In this case, a new connection request is sent.

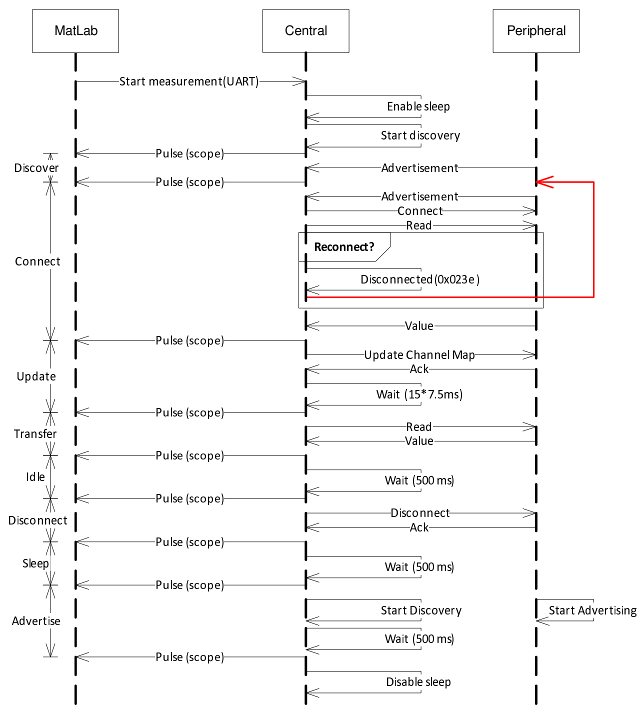


Fig. 6: Schematic view of the measurement sequence.

- In the **Update** phase, the connection parameters are updated to use only channels 1-8 as described in Section III-C4.
- In the **Transfer** phase, a read request is sent by the central device to the peripheral device. After receiving a response from the peripheral device this phase ends.
- In the **Idle** phase, the connection is kept active for 500ms without transmitting any data.
- In the **Disconnect** phase, a disconnect request is sent to the peripheral device. The phase ends when the disconnection is acknowledged by the peripheral. However, there is no way to report if the disconnection request times out. When this happens no trigger is sent and the result of the complete measurement cycle is discarded.
- In the **Sleep** phase, both devices are kept in sleep mode to perform the current calibration as explained in Section III-D1.
- In the **Advertising** phase, the central device is continuously scanning, and the peripheral device is continuously advertising.

D. Implementation issues

During the implementation of the software for the measurement devices and the execution of the experiments, we encountered some issues. We capture them here and also describe our solutions.

1) Current measurement: To measure the Bluetooth module's energy consumption, we measure its current consumption as explained in Section III-C2. This current varies from $0.9\mu A$ in Power Mode 2 (which is the lowest possible sleep mode) to $27.0mA$ (when the radio is receiving) [21]. We decided to assume that the current consumption in sleep mode is $0.9\mu A$, as specified by the manufacturer. We put the module in sleep mode for $500ms$ and calculate the parameter V_C for the current formula in Section III-C2 using the expression $V_C = V_M + 30 \cdot 9 \cdot 10^{-7}$, where V_M is the mean measured voltage during the sleep period.

2) Measurement method: We tried to collect the energy measurement data directly via USB streaming mode, controlled by MATLAB, which was informed by the Bluegiga module over UART. However, the timing was not precise enough. We solved this by adding an extra measurement channel to send the triggers over GPIO and analyze the data after the measurement was completed.

3) Testing environment: As the 2.4 GHz ISM band is used by other wireless protocols, the most important being WiFi, we had to find a suitable location to perform the experiments. We used a spectrum analyzer to analyze the interference present in a normal office environment. We tried performing the experiments inside an EMC testing cage, which blocks all signals from outside. However, during the experiments, we observed that a very small change in the setup, like moving the setup a few centimeters, resulted in completely different set of measurements. We expect this to be caused by the fact that the cage was not equipped with proper damping material. Therefore, all kinds of reflections of the signals could occur and attenuate or amplify each other. Eventually, we performed the measurements in a meeting room in a corner of the building that was covered by only one WiFi AP, which was turned off for during measurements. We regularly checked if there is any external interference during the measurements.

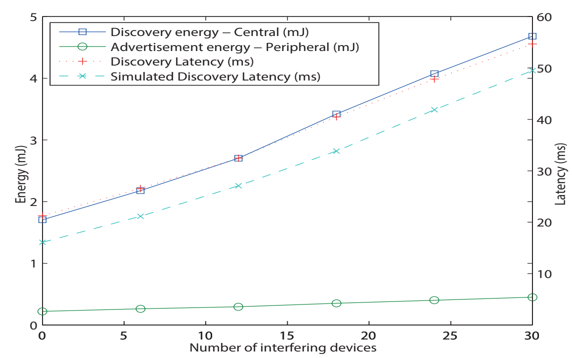


Fig. 9: Measurements for device discovery procedure.

E. Experiment results

In this section, we present the measurement results. We experimented with 32 BLE devices, where all the measurements are done for only one pair or the 'devices under test'. The remaining 15 pairs are used to create interference.

1) Detailed measurements: Fig. 7 and 8 shows the current consumption from a single measurement cycle for two configurations. In Fig. 7, it can be seen that interfering devices in advertising mode cause a longer discovery and connect process, while Fig. 8 shows that a lot of interfering devices in transferring mode can dramatically increase the time needed for data transfer.

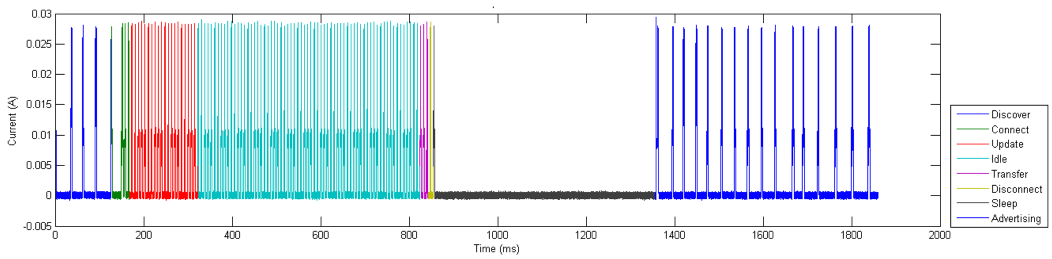


Fig. 7: Current consumption by the peripheral devices at the presence of 24 advertising devices.

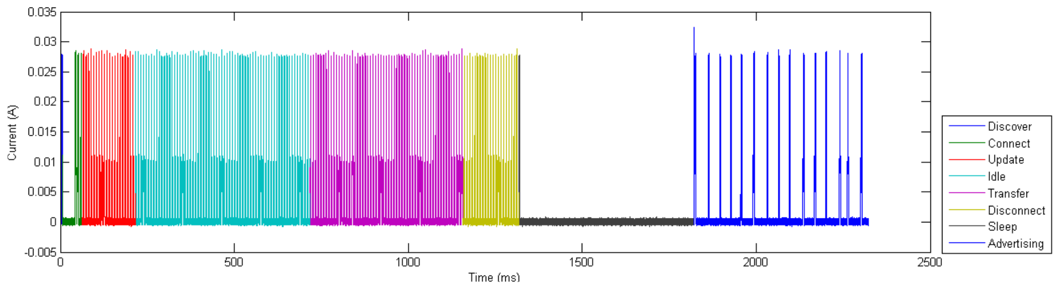


Fig. 8: Current consumption by the peripheral devices at the presence of 30 data transferring devices.

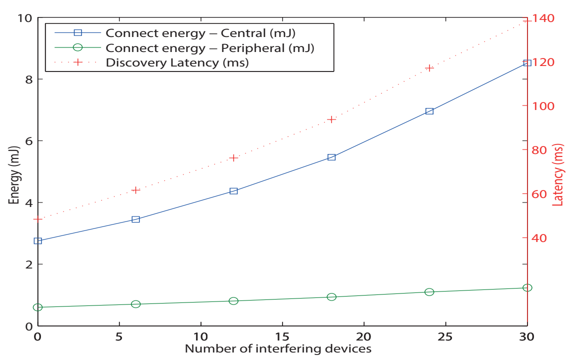


Fig. 10: Measurements for device connect.

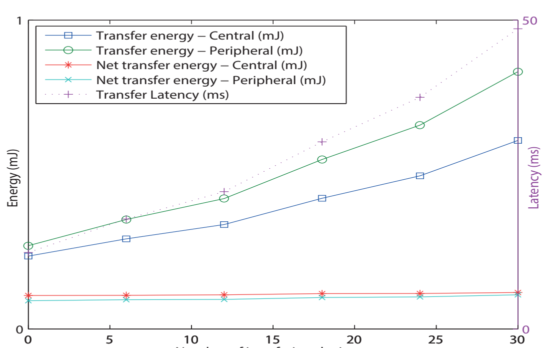


Fig. 11: Measurements for data transmission.

2) Interference measurements: We measured the influence of an increasing number of interfering devices on the latency and energy consumption for the various layouts as shown in Fig. 5. As the results for different layouts are similar, here, we present the results for layout C only due to paucity of space. Fig. 9 shows the latency and energy consumption measurements for device discovery under interference from advertising devices. We see that the maximum increase in energy consumption for discovery is 2.2x when 30 interfering devices were present.

We have also measured the latency in device discovery using simulation. Here, we compared the simulation and experimental results only for the same number of devices. When comparing them, we see that the shape of the graph is similar, but there is a constant difference. The difference is due to the fact that the simulation does not consider reconnection.

Fig. 10 shows the latency and energy consumption measurements for the connection setup operation under interference from advertising devices. Similar to the discovery results, we see a maximum energy consumption increase of 2.2x when number of interfering devices increase from 0 to 30.

Fig. 11 shows the measurement results during data transfer under interference. In this graph, the net transfer energy is the amount of energy spent during the transfer of a data packet minus the energy spent to keep the connection idle for the same period. In other words, the amount of energy spent on the transfer of the data packet, assuming that a connection would have been active anyway. The net transfer energy is calculated as follows:

$$E_{net} = E_{transfer} - L_{transfer} \cdot \frac{E_{idle}}{L_{idle}},$$

where E_{net} is the net transfer energy, $E_{transfer}$ is the measured transfer energy, $L_{transfer}$ is the transfer latency, E_{idle} is the idle energy and L_{idle} is the time for which the idle

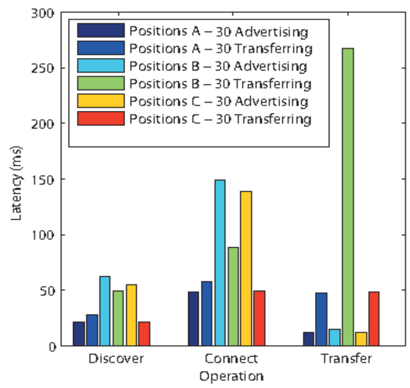
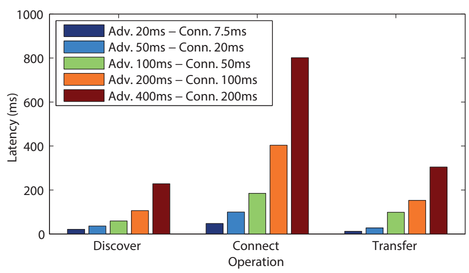
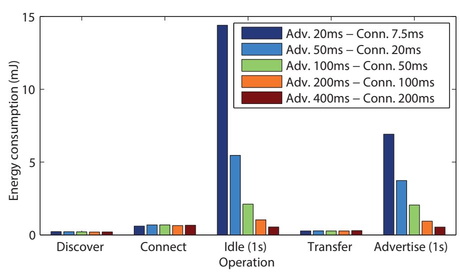


Fig. 12: Discover, connect and transfer latencies for various configurations.



(a)



(b)

Fig. 13: Measurements for various advertising and connection intervals.

energy is measured. From these graphs, we see that the net energy consumption increases only by 1.3x when number of interfering devices increase from 0 to 30.

3) Latency comparison: Fig. 12 shows latencies for the maximum number of interfering devices for all device layouts. This shows that the influence of interfering devices in advertising mode on the devices transferring (and vice versa) is very small. The largest latency increase for transferring devices is 3.9x from 0 to 30 devices in Layout C.

4) Connection and advertising intervals: To make a choice between keeping a connection alive continuously and connecting again for every time data has to be transferred, we experimented with longer advertising and connection intervals. These measurements were done without any interfering devices. Fig. 13a shows advertising, connect and transfer latency for different intervals, whereas energy consumption by a peripheral device is shown in Fig. 13b. From these measurements, we can conclude that longer advertising intervals result in lower energy consumption but also higher latency.

From the results, we can conclude that the influence of interfering devices that are advertising on already connected devices that are transferring data is very small. Moreover, the influence of connected devices that are transferring on the devices that are discovering is very small. The maximum increase of energy consumption and latency is about 3x.

IV. Model development

Based on the results, we developed models for the energy consumption and latency of BLE devices. Here, we describe only the relevant models that are required for the proposed system. In Table II, we list the functions, parameters, and symbols that are used in the model.

TABLE II: Parameters used in the energy model.

Symbol	Description	Unit
	Inputs	
I_{adv}	Advertisement interval	ms
I_{conn}	Connection interval	ms
N_{pkt}	Number of packets	#
N_{adv}	Number of advertising devices	#
	Model parameters	
E_{pkt}	Energy requirement for 1 data packet	mJ
$T_{proc,disc}$	Processing time for device discovery	ms
	Outputs	
$T_{discover}(I_{adv}, N_{adv})$	Discovery latency	ms
$T_{connect}(I_{adv}, I_{conn}, N_{adv})$	Connect latency	ms
$T_{transfer}(I_{conn}, N_{pkt})$	Transfer latency	ms
$E_{transfer}(N_{pkt})$	Transfer energy (per packet)	mJ

A. Advertising

The discovery latency for a device depends on the advertisement interval (I_{adv}) and the number of advertising devices (N_{adv}). We assume that the advertising interval is equal for all devices. The discovery delay (in ms) is given by,

$$T_{discover}(I_{adv}, N_{adv}) = (0.5 \cdot I_{adv} + T_{proc,disc}) \cdot \exp^{\frac{2N_{adv}}{3(0.5 \cdot I_{adv} + T_{proc,disc})}} . \quad (1)$$

From the simulation results, we found $T_{proc,disc} = 9.7$ms. We model the connection latency (in ms) as the discovery time plus three times the connection interval. This is the connection latency including the check whether the connection really succeeds as explained in Section III-C5. The connection latency is given by,

$$T_{connect}(I_{adv}, I_{conn}, N_{adv}) = T_{discover}(I_{adv}, N_{adv}) + 3 \cdot I_{conn}. \quad (2)$$

B. Data transfer

We model the transfer latency for a packet with the maximum supported payload of 20B and is given by,

$$T_{transfer}(I_{conn}, N_{pkt}) = \frac{3}{2} \cdot I_{conn} \cdot N_{pkt}. \quad (3)$$

Similarly, we model the transfer energy (in mJ) for a packet as,

$$E_{transfer}(N_{pkt}) = E_{pkt} \cdot N_{pkt}, \tag{4}$$

where $E_{pkt} = 0.27mJ$ (from the experiment).

These models are developed and validated based on the experimental data. Applications developers, using BLE, can use the energy model to predict the energy consumption of an application before implementing it.

C. Model evaluation

To evaluate the accuracy of our model, we used it to calculate part of the values that resulted from the experiments. Table III shows the experiment values with the corresponding model outcomes. From these results we can see that the model is in general pretty accurate. For the connection energy and latency, we see that the experiment values are higher than the model values in situations with interfering devices. With regard to energy, this is caused by the fact that we do not consider the interfering devices in the model. In case of connection latency, this is caused by the fact that reconnection (as described in Section III-C5) are not accounted for in the model.

TABLE III: Comparison between model and experiments in terms of energy consumption and latency for various operations of bluetooth devices. The number of interfering devices are varied from 0 to 30.

Interf. (#)	Intervals (ms)		Conn. en. (mJ)		Transf. en. (mJ)		Adv. en. (mJ/s)		Discov. lat. (ms)		Conn. lat. (ms)		Transf. lat. (ms)	
Adv.	Conn.	Adv.	Mod.	Exp.	Mod.	Exp.	Mod.	Exp.	Mod.	Exp.	Mod.	Exp.	Mod.	Exp.
0	7.5	20	0.60	0.60	0.27	0.27	7.33	6.91	20.4	21.3	42.9	48.0	11.3	12.4
6	7.5	20	0.60	0.70	0.27	0.28	7.33	6.96	25.0	26.6	47.5	61.6	11.3	12.7
12	7.5	20	0.60	0.81	0.27	0.27	7.33	6.90	30.6	32.5	53.1	76.2	11.3	12.4
18	7.5	20	0.60	0.93	0.27	0.28	7.33	6.96	37.5	40.5	57.5	93.7	11.3	12.8
24	7.5	20	0.60	1.10	0.27	0.28	7.33	6.93	45.9	47.8	68.4	117.1	11.3	12.6
30	7.5	20	0.60	1.23	0.27	0.27	7.33	6.95	56.2	54.7	78.7	138.4	11.3	12.3

V. Conclusion

With the expectation that the number of Bluetooth enabled device are going to increase manifold in the near future, the mutual interference among independent Bluetooth piconets can affect the performance of the communication. Thus, we experimentally studied the influence of mutual interference on the energy consumption and latency in BLE devices. We performed our experiments under mutual interference by up to 30 devices. Our experiments show that the advertising interval plays an important role in energy consumption and latency. Additionally, idle operation consumes a significant amount of energy. Thus, it is suitable to disconnect and re-establish the connection periodically, rather than keeping a continuous connection in case the data transfer between the central and the peripheral device is not continuous. We have developed models for energy consumption and latency, which can be used by other application developers. In general, the Bluetooth protocol is very robust and the performance does not degrade too much unless there are hundreds of devices within a close proximity.

Acknowledgment

This work is partially supported by an EU FP7 project, called iCore (http://www.iot-icore.eu/).

References

[1] Bluetooth SIG, "Bluetooth specification version 4.0," *Bluetooth SIG*, 2010.

[2] J.-S. Lee, Y.-W. Su, and C.-C. Shen, "A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi," in *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*, pp. 46–51, IEEE, 2007.

[3] ABIresearch, "Over 30 Billion Wireless-connected Devices to Be Part of the IoE in 2020." https://www.abiresearch.com/market-research/product/1016390-over-30-billion-wireless-connected-devices/, 2013. Accessed: 2015-03-01.

[4] C. Guo, R. Venkatesha Prasad, and M. Jacobsson, "Packet forwarding with minimum energy consumption in body area sensor networks," in *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, pp. 1–6, IEEE, 2010.

[5] W. Yuan, *Coexistence of IEEE 802.11 b/g WLANs and IEEE 802.15. 4 WSNs: Modeling and Protocol Enhancements*. TU Delft, Delft University of Technology, 2011.

[6] Bluetooth SIG, "Bluetooth specification version 4.1," *Bluetooth SIG*, 2013.

[7] R. Heydon, *Bluetooth Low Energy: The Developer's Handbook*. Pearson Always Learning, Prentice Hall, 2012.

[8] K. Townsend, C. Cufí, and R. Davidson, *Getting Started with Bluetooth Low Energy: Tools and Techniques for Low-Power Networking*. O'Reilly Media, Incorporated, 2014.

[9] J. Liu, C. Chen, and Y. Ma, "Modeling and performance analysis of device discovery in bluetooth low energy networks," in *Global Communications Conference (GLOBECOM), 2012 IEEE*, pp. 1538–1543, IEEE, 2012.

[10] J. Liu, C. Chen, Y. Ma, and Y. Xu, "Energy analysis of device discovery for bluetooth low energy," in *Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th*, pp. 1–5, IEEE, 2013.

[11] J. W. Chong, H. Y. Hwang, C. Y. Jung, and D. K. Sung, "Analysis of throughput and energy consumption in a zigbee network under the presence of bluetooth interference," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, pp. 4749–4753, IEEE, 2007.

[12] A. Stranne, O. Edfors, and B.-A. Molin, "Experimental verification of an analytical interference model for bluetooth networks," in *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, pp. 1–5, IEEE, 2006.

[13] A. Stranne, O. Edfors, and B.-A. Molin, "Energy-based interference analysis of heterogeneous packet radio networks," *Communications, IEEE Transactions on*, vol. 54, pp. 1299–1309, July 2006.

[14] I. Howitt, "Mutual interference between independent bluetooth piconets," *Vehicular Technology, IEEE Transactions on*, vol. 52, no. 3, pp. 708–718, 2003.

[15] M. Goldenbaum and S. Stanczak, "On multiantenna sensor networks with interference: Energy consumption vs. robustness," in *Smart Antennas (WSA), 2012 International ITG Workshop on*, pp. 125–132, IEEE, 2012.

[16] C. Gomez, I. Demirkol, and J. Paradells, "Modeling the maximum throughput of bluetooth low energy in an error-prone link," *Communications Letters, IEEE*, vol. 15, no. 11, pp. 1187–1189, 2011.

[17] P. Kindt, D. Yunge, R. Diemer, and S. Chakraborty, "Precise energy modeling for the bluetooth low energy protocol," *arXiv preprint arXiv:1403.2919*, 2014.

[18] M. Siekkinen, M. Hiienkari, J. Nurminen, and J. Nieminen, "How low energy is bluetooth low energy? comparative measurements with zigbee/802.15.4," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE*, pp. 232–237, April 2012.

[19] Bluegiga, "BLE113 Development Kit 2.0 Datasheet." https://www.bluegiga.com/, 2013. Accessed: 2014-10-10.

[20] Bluegiga, "Knowledgebase: Throughput with bluetooth smart technology." https://bluegiga.zendesk.com/, 2013. Accessed: 2014-10-13.

[21] Bluegiga, "Knowledgebase: Ble module low power and sleep modes." https://bluegiga.zendesk.com/, 2013. Accessed: 2014-10-14.