DTU Library

AptBLE: Adaptive PHY Mode based on K-means Algorithm in Bluetooth5

Luo, Xuan; Zang, Mingyuan; Yan, Ying; Dittmann, Lars

Link back to DTU Orbit

AptBLE: Adaptive PHY Mode based on K-means Algorithm in Bluetooth5

Xuan Luo*, Mingyuan Zang†, Ying Yan‡ and Lars Dittmann§

Department of Photonics Engineering, Technical University of Denmark

Kgs. Lyngby, Denmark

Email: *s192270@student.dtu.dk, †minza@dtu.dk, ‡yiya@fotonik.dtu.dk, §ladit@fotonik.dtu.dk

Abstract—**Since Bluetooth5 standard released in 2016, its usage in commercial electronic products has been increased rapidly and substantially. Comparing to BLE 4.2, Bluetooth5 supports three PHY modes, respectively 2M, 1M and Coded PHY mode, providing a higher throughput and a wider range. Whereas there is a trade-off between its throughput and coverage. When the connection is established, the PHY mode is commonly pre-configured and fixed. This rigid design limits the flexibility in offering dynamic throughput and coverage. Therefore, we propose a method termed AptBLE, that switches the PHY mode in Bluetooth5 adaptively by considering the Received Signal Strength Indicator (RSSI) level. Specifically, we optimise the RSSI threshold for different PHY modes using the K-means clustering algorithm. Moreover, based on AptBLE, we further enable the Data Length Extension (DLE) feature and term the improved method as AptBLEM. We implement AptBLE (M) on the boards and test in indoor environment. The experimental results show that, AptBLE is more flexible, robust and outperforms the original fixed PHY mode in terms of throughput and transmission range. Furthermore, AptBLEM can triple the throughput than AptBLE, with a maximum throughput value in 1035Kbps and 42m range in indoor environment.**

Index Terms—**Bluetooth5, Internet of Things, machine learning algorithm, RSSI, indoor measurement, adaptive algorithm**

I. Introduction

Bluetooth technology is designed for short-range wireless communications among Internet of Things (IoT) devices. Compared with other varieties of technologies (e.g. WiFi, LoRa, ZigBee) developed for IoT scenario, Bluetooth is widely used in low-data-rate, end-to-end connections with its compelling advantage in low energy (LE) consumption. It has been prevalent in applications such as the home automation service to connect the appliances and the digital health service to connect the wearable devices such as smart watches and hearing aids. With such diverse application scenarios, it is predicted by ABI Research [1] that the Bluetooth market will show a tripled growth to 1.6 billion shipments of Bluetooth devices.

Bluetooth5 (BT5) is an enhanced version of Bluetooth technology released by the Bluetooth Special Interest Group (SIG) in 2016. It significantly improves the capacity and coverage of Bluetooth to provide fast and further data transmissions and stable connections. Specifically, evolved from the Bluetooth 4.2, Bluetooth5 successfully doubles the transmission speed

and quadruples the coverage range. [2] One of the core features to support such new enhancements is the new modulation schemes in the Physical Layer (PHY). Inheriting the PHY options of LE 1M in Bluetooth 4.2, Bluetooth5 adds new PHY options: LE 1M coded and LE 2M to improve the data rate or coverage. In LE 2M, a doubled symbol rate of 2 Mega symbols per second (Msym/s) is enabled to support a 2MHz bandwidth for data transmission and thereby a double transmission speed.

Such newly added PHY options provide the support for applications like wireless headphones to transmit high-resolution audio. Despite this advantage, it won't be the best choice to keep the highest transmission speed all the time. Considering BLE as a low power technology, the high transmission speed will lead to a high power consumption and affect the battery life. In the light of the mobility scenario with Bluetooth, the highest transmission speed would result in a short range, and transmission quality might be affected with a moving device. The stable connection with a high transmission speed is particularly hard to be kept in indoor scenario. The walls, doors, and furniture would diminish the wireless signal strength, which would be reflected in terms of RSSI.

Adaptation is considered as one solution to balance the trade-off between transmission speed and coverage. It can help maximize PHY resource utilization by adaptively switching between the PHY options to adjust the modulation scheme. In this way, a stable transmission and relatively high-level throughput can be ensured, especially when the surrounding signal connectivity conditions are changing or when devices are in a moving state. Similar solution has been applied in IEEE 802.11 to achieve data rate adaptation based on previous transmission status as discussed in paper [3]. For Bluetooth technology, several studies have also been conducted on performance evaluation of PHY transfer mode and some of them apply the data rate adaptation method to optimize resource utilization. Researchers in [2] [4] measure the single pair throughput performance in indoor environment and compare with different PHY transfer modes. Paper [5] measures multi-node throughput and delay in an office environment. The work in [6] analyzes the trade-off between throughput and power efficiency in Bluetooth5. While both of them test in a single PHY mode each time and cannot dynamically change the data rate. Authors in [7] propose a method using RSSI values as the condition to change the PHY transfer mode, while they simply assume a RSSI threshold and only prove the PHY

mode adaptation is possible. Meanwhile, a data rate adaptation method based on Packet Reception Ratio (PRR) is studied in [8] to reduce the power consumption.

To achieve the adaptation design, thresholds need to be defined for dynamical switching between PHY options. The thresholds can be decided by clustering method to obtain an adaptable result under the dynamic environment. As a classic unsupervised Machine Learning method, clustering is suitable to be applied in the mobility scenarios in Bluetooth technology and has been studied in several pieces of work. K-means is a common-used algorithm, featuring fast convergence and easy implementation among the clustering methods. Paper [9] develops K-means clustering to increase the accuracy for indoor floor estimation and localization. The authors in [10] compare the performances of K-means clustering with k-nearest neighbors (kNN) in RSS-based indoor location sensing service. The work done in paper [11] develops and evaluates K-means algorithm as semi-supervised learning method for scalable BLE-based indoor detection on occupancy pattern. In spite of the extensive research works, K-means clustering hasn't been applied in adaptation process in Bluetooth technology.

In this paper, we propose an adaptation method to dynamically switch the PHY transfer modes accordingly based on the comparison of the detected RSSI value and the pre-defined RSSI threshold. The goal is to achieve seamless data transmission with high throughput. The optimal RSSI threshold values are obtained with K-means clustering algorithm. The proposed method is implemented on the board and indoor experiments are conducted in real case to compare and evaluate the performances of the adaptation method and single fixed PHY transfer modes.

The contributions of this paper are as follows:

- We propose AptBLE(M) method, which adaptively selects the appropriate PHY mode without being terminated and re-connection. AptBLE can efficiently improve transmission throughput and furthermore, AptBLEM can even triple the throughput than AptBLE with the same flexibility and robustness.
- We apply the K-means clustering algorithm to obtain optimal RSSI thresholds for our AptBLE, which hasn't been studied in the existing works to the best of our knowledge.
- We design a detailed and practical indoor experimental scenario to collect the data in different distances and build a dataset to train our K-means classification algorithm.
- We evaluate the proposed AptBLE(M) performances on the board in indoor places with different transmission distances. The obtained throughput results outperform the ones with original single PHY transfer mode.

This paper is organized as follows. In section II we introduce the features in BT5 and in section III we elaborate how to get optimal RSSI thresholds for AptBLE. Section III presents the implementation on board. In section IV we introduce the experimental setup, the scenario design and evaluation of the experimental results. Section V concludes the paper.

TABLE I
SUMMARIES OF PHY FEATURE IN BT5

PHY	Data rate	RSSI Sensitivity	Range	DLE
LE 1M	1Mbps	-95dBm	1x	Supported
LE 2M	2Mbps	-90dBm	0.8x	Supported
LE Coded S=2	500Kbps	-103dBm	2x	Supported
LE Coded S=8	125Kbps	-103dBm	4x	Supported

II. BLUETOOTH 5.0 FEATURES

A. New PHY Choices

BT5 is the latest version of the Bluetooth wireless communication standard. Compared with its preceding versions, one of the significant improvements is three PHY options namely LE 2M, LE 1M and LE Coded [12]. LE 1M is the PHY mode used in Bluetooth 4.0 while LE 2M and LE Coded are new choices used in BT5. LE 2M mode doubles the symbol rate that the LE 1M PHY uses to 2Msym/s. The LE Coded PHY quadruples the range without increasing transmission power by using Forward Error Correction (FEC) coding method. Specifically, two coding schemes are used, respectively S = 2, in which two symbols represent one bit, thereby reaching a 500Kbps bit rate and S = 8, in which eight symbols represent one bit, thereby reaching a 125Kbps bit rate in theory. The FEC technique in the Coded mode allows data to be correctly decoded at a lower RSSI values. Thus it enlarges the transmission distance. Whereas Coded mode sacrifices the data rate due to adding redundancy in error correction. Hence, when choosing a PHY mode in BT5, there is a trade-off between data rate and transmission range. Table I shows the summaries of PHY features in BT5.

B. Data Length Extension

Data Length Extension (DLE) has been first introduced in Bluetooth 4.2 Core Specification. It is also been applied in BT5, concretely, DLE is a feature added to the Link Layer which enlarges the Data Channel Protocol Data Unit (PDU) Payload field from the default 27 bytes to up to 251 bytes, as shown in Fig. 1. Each packet in Link Layer has fourteen bytes overhead, respectively one byte Preamble, four bytes Access Address, two bytes PDU Header, four bytes Message Integrity Check (MIC) for encryption, and three bytes Cyclic Redundancy Check (CRC) [13].

Although DLE feature has already proposed in BT4.2, many existing experimental papers haven't enabled the function,
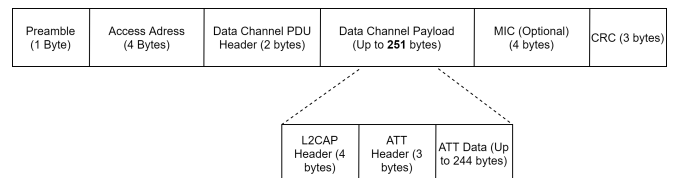

Fig. 1. Bluetooth 4.2/5.0 Link Layer Packet Structure.

causing a slower data rates comparing to theoretical values. For instance, paper [2] and [7] only used the default 27 bytes without DLE enabled in experiment. To enable the DLE, the board should support the DLE function. In our experiment, we use two nRF 52840-DK boards, where the data length up to 251 bytes can be set in the scripts. Moreover, in order to pair the central and peripheral device that both support DLE, a DLE request will be sent from the TX side and the RX will send back a DLE response in initialisation process [14]. In our work, AptBLEM enables the DLE option and can triple the data rate when transmission range is within 20 metres. The detailed experimental results will be discussed in section IV.

III. Algorithm and Implementation

A. K-means Clustering Algorithm

In a BT5 end-to-end connection between two devices, the wireless connection can become unstable due to RF interference from other wireless technology (such as WiFi) or the blockage from indoor environment like walls or doors. In general, the further the distance is between the devices, the weaker the signals would be, and the lower the RSSI value would be. In our work, in order to decide the RSSI threshold for adaptation method, the idea of clustering is introduced. We collect the RSSI values by setting the two devices with the distance of 3m, 6m, 9m, 15m, and 30m in between (depicted as blue point 1-5 in Fig. 4) as training set. This dataset contains 7543 RSSI values in total. Since the wireless links are unstable leading to variations of RSSI values, sampling of every 100 values is done to reduce the variations and prepare for clustering. The dots plotted in Fig. 2 are the collected RSSI data to be clustered.

For the clustering algorithm, K-means clustering model is trained in this paper. The K-means algorithm belongs to the type of unsupervised learning, which can be used to learn from a set of unlabeled data. It has a relatively easy implementation and efficient partitioning process which makes it suitable for BT5 on-board implementation compared with other existing clustering algorithms. The clustering process of K-means can be described as assigning the data points to each cluster and obtaining the cluster centroids via iterations. The object function for dataset $\mathbf{x}_1, ..., \mathbf{x}_n$ with the number of N data points [15] can be written as:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2, \tag{1}$$

where K is the number of clusters. The vector $\boldsymbol{\mu}_k$ is the mean value of data points in cluster k, which is usually called as the centroid of the cluster. With the Eq. (1), the training process can be regarded as finding the cluster centroids $\boldsymbol{\mu}_k$ to minimize the object function J. The whole process includes the following steps:

- initialize $\boldsymbol{\mu}_k$ by randomly selecting the values,
- cluster each data point by identifying the closest $\boldsymbol{\mu}_k$ to the data point,
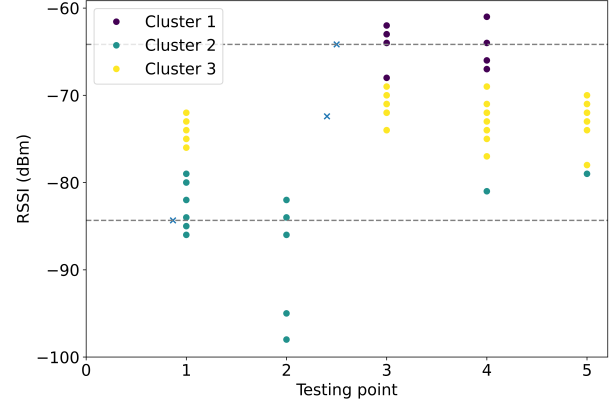
Fig. 2. K-means clustering results.

- compute the mean values of all data points partitioned to each clusters and update the $\boldsymbol{\mu}_k$ with these new values,
- iterate until the $\boldsymbol{\mu}_k$ values converge.

In our work, to identify the two RSSI threshold for the three PHY transfer modes, the dataset is trained to make three clusters to represent the three potential modes that the data points may belong to. The clustering results are marked with three colors and the converged centroids are marked as blue crosses in Fig. 2. Based on this result, considering the variations of the RSSI values in wireless transmission, the centroids of cluster 1 and 2 are selected and the corresponding RSSI values (the values pointed by the dash lines in Fig. 2) are the identified thresholds.

B. Adaptive PHY Switch On The Boards

In this subsection, the specific on-board implementation is introduced. The firmware used in this work is based on the official *ble_app_att_mtu_throughput* example in nRF Software Development Kit (SDK) [16]. The selected SDK version is v17.0.0, which can be supported on nRF58240-DK. Besides, the chipset is programmed with S140 soft device, which provides full and flexible APIs for building BLE in nRF52 board series on chip solutions [17].

The RSSI threshold value is learnt from the implemented K-means algorithm that is introduced in aforementioned subsection. The on-board implementation is demonstrated in the

Fig. 3. Testbed setup.

Fig. 4. Indoor experiment sites.

following pseudo code. When rssi_values > -64 for ten times, PHY switches to 2M mode. When -83 < rssi_values < -64 for ten times, PHY switches to 1M mode. When rssi_values < -83 for ten times, PHY switches to Coded mode. In the experiment, we assume the RSSI counter for switching the PHY is ten. Therefore, the PHY mod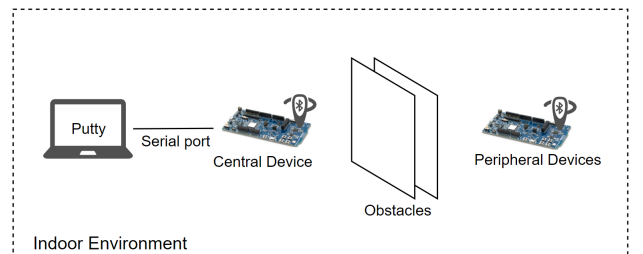e is updated only if the RSSI counter counts ten times, which avoids frequent switching. This implementation updates the PHY when the case BLE_GAP_EVT_RSSI_CHANGED is executed, therefore comparing to the implementation in [7], there is no need to terminate the whole transmission process so that to save time for re-scanning and re-connection and thereby achieve a seamless transmission.

In order to calculate the throughput, we put a time stamp on each single packet, when the packet is sent and received and sum altogether, denoted as T_{trans}. Meanwhile, we use a byte counter incrementing all transceived bytes, denoted as N_{bytes}. The throughput is denoted as $\delta_{throughput}$, which can be calculated as follows:

$$\delta_{throughput} = \frac{N_{bytes} \times 8\text{bits}}{T_{trans}} \quad (2)$$

IV. EXPERIMENTAL EVALUATION

A. Experimental Setup

To collect RSSI data for training and testing the implemented K-means algorithm, we build a testbed with two nRF52840-DK boards and one laptop installed with the Putty client to collect log data through the serial port. The testbed set up is shown in Fig. 3 and the configuration of board is listed in table II. Both central and peripheral devices use the LE 1M Coded mode as initial PHY, since it has a widest sensitivity range (-103 dBm in nRF52840), which can establish a connection with lower RSSI. In BT5, there are a total of 40 radio channels, each of which are 2MHz wide and are divided as advertising channels and data channels. We use the default setting, channel 37, 38 and 39 as advertising channels and the rest as data channels [12]. BT5 allows data length of

---

Algorithm 1: On-board adaptive PHY mode selection

Connection Establishment
parameters_initialisation (listed in tab. II);
advertising_start();
scan_start();
//write as a case in function ble_event_handler
while *case BLE_GAP_EVT_RSSI_CHANGED is*
 executed **do**
  if *rssi_value > RSSI_threshold_2M* **then**
    rssi_counter += 1;
    if *rssi_counter == 9* **then**
      rssi_counter = 0;
      PHY_transceiver = 2M_PHY;
    end

  else if *rssi_value < RSSI_threshold_2M &&*
  rssi_value > RSSI_threshold_1M **then**
    rssi_counter += 1;
    if *rssi_counter == 9* **then**
      rssi_counter = 0;
      PHY_transceiver = 1M_PHY;
    end

  else
    rssi_counter += 1;
    if *rssi_counter == 9* **then**
      rssi_counter = 0;
      PHY_transceiver = Coded_PHY;
    end
  end
end

each packet on the channel to be 27-255 octets. In the initial experiment, the data length is set in default as 27 bytes. In our AptBLEM, the DLE is enabled and data length is set in 251 bytes. Various data length influences the throughput due to the different payload in data transmission. The specific difference will be illustrated in the following subsection.

Fig. 4 illustrates the indoor experimental sites and the scenario. The red dot is referred as the central device and the blue dot is referred as peripheral device. In the experiment, we measure the single-link throughput performance with different distances (1m, 10m, 20m, 30m, 35m, 40m, 42m) and obstacles (doors or walls) by placing the peripheral board in different sites.

In order to get a optimal RSSI threshold value, we first set the boards on LE 1M Coded mode and collect a data set used for K-means clustering algorithm, including RSSI values, throughput, type of transfer mode and distance. We collect 1MB data in each experimental site and further concatenate all data altogether, which is used as a training set. Afterwards, we input the optimal RSSI threshold value that is learned from K-means algorithm and test the transmission from the same locations. Fig. 5 shows a specific procedure in dynamically updating PHY mode on the boards during the measurement. When the connection is established successfully, the host receives and collects the log data through serial port by the PUTTY client.

B. Experimental Results

Fig. 6 depicts the indoor experimental results. The performance evaluation is based on the transmission throughput with each type of PHY transfer mode deployed. In general, by deploying the proposed AptBLE or AptBLEM mode, the data transmission can be guaranteed a relatively higher throughput compared with only by deploying a single mode. As the distance values in between the devices increase, the performance of such enhancement may vary.

Performance evaluation of AptBLE: AptBLE has a more robust performance and is able to transmit in a longer distance, approximately 42 metres according to our tests. At the first 20 metres when the distance is short, AptBLE has a comparable throughput performance to LE 2M mode, where the RSSI value fluctuated around -60 dBm. As the distance goes further, the RSSI drops and AptBLE changes the PHY mode dynamically based on the preset threshold, which then performs better than LE 2M, LE 1M and LE 1M Coded mode. Such advancement is conspicuous when the devices are separated by 35 metres. The 2M mode fails to maintain the connection while the LE 1M and LE 1M Coded can still support the transmission with better coverage performance. The proposed AptBLE presents its superiority with a 67% higher throughput than LE 1M mode. When the transmission distance goes further than 40 metres, both the central device with LE 2M and LE 1M mode cannot connect to the peripheral device. While the AptBLE can still advertise, scan in LE 1M Coded mode then adjusts the PHY according to the RSSI values. By doing this, AptBLE can enable the throughput to be more than 1.5 times higher than the LE Coded mode to 21Kbps.

Performance evaluation of AptBLEM: In the proposed Apt-BLEM method, the DLE is enabled and the data length of a payload is maximized to 251 bytes. From the results, the throughput performance is improved because enabling DLE increases the payload in one single link layer packet. Specifically, when the devices are separated by 1 metre with no obstacle in between, LE 2M mode shows the best performance among the three basic modes with the throughput as 340Kbps. However, the proposed AptBLEM reaches 1035Kbps, which triples the throughput of LE 2M. When the distance is extended, the throughput decreases but can still remain around 366Kbps at the distance of 20 metres. When it comes to a long stretch, the AptBLEM can present outstanding performance especially when the three basic modes manifest their coverage limitations. By dynamically switching the PHY transfer modes and with DLE enabled, the AptBLEM approximately increases the throughput of LE 1M by 125% while still maintains around 12Kbps throughput at the distance of 42 metres when other modes hardly keep their connections.

Table III shows a summary of the experimental results. Our proposed method AptBLE can reaches the highest throughput overall and transmits with a maximum range. When DLE is on in the AptBLEM method, the maximum value of throughput can even be tripled than AptBLE or LE 2M mode and also transmit a maximum range.

TABLE II
BOARD CONFIGURATION

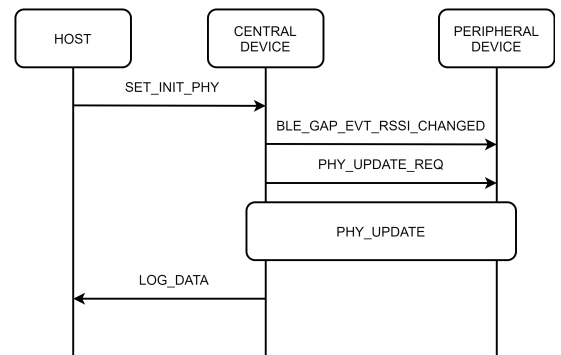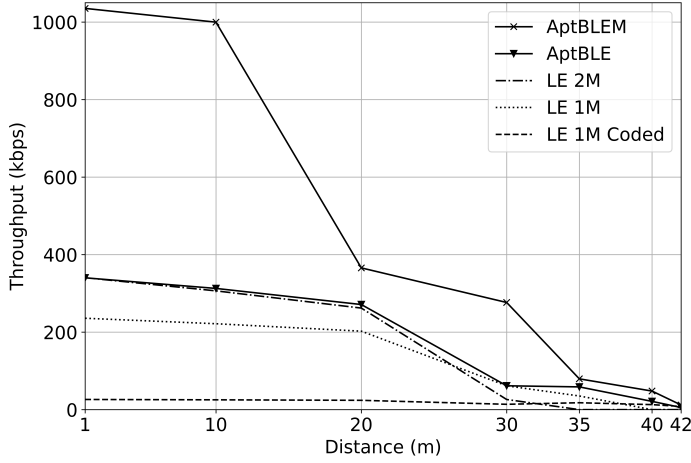| Parameter	Type/Value
ATT MTU size	247
Data Length	27 (251 is used in AptBLEM)
Connection interval	7.5ms
Initial PHY	Coded
TX Power	+8dBm
RSSI_counter	0
RSSI_threshold_2M	-64dBm
RSSI_threshold_1M	-83dBm

Fig. 5. PHY update procedure.

Fig. 6. Throughout comparison with respect to PHY transfer modes.

TABLE III
SUMMARY OF EXPERIMENTAL RESULTS

PHY	Max Throughput	Max Distance	DLE
LE 1M	235.78Kbps	39m	off
LE 2M	340.35Kbps	35m	off
LE Coded S=8	26.2Kbps	42m	off
AptBLE	**340.11Kbps**	**42m**	**off**
AptBLEM	**1035Kbps**	**42m**	**on**

V. CONCLUSION

In this paper, we propose two approaches, AptBLE and AptBLEM, to adaptively switch the PHY transfer modes in BT5 based on the RSSI level. The aim is to mitigate the trade-off between throughput and range, that is to transmit the data in relatively high throughput and reduce the effect from distance changing. We collect the real-case RSSI dataset with using the nRF52840-DK boards and train the K-means clustering algorithm to identify the RSSI thresholds for the PHY mode transfer. AptBLEM is developed based on AptBLE with DLE enabled to further improve the throughput performance. The proposed methods are implemented on the boards and evaluated in indoor environment. The results present that our algorithms can efficiently increase the transmission throughput within a short distance, and remain the robustness and relatively high throughput within a long distance, compared with the throughput performances when the PHY is set on a single mode of LE 1M/LE 1M Coded/LE 2M.

The future work can consist of:

- developing other types of clustering methods to compare their performance enhancements and computation resource consumption,
- measuring the power consumption of each mode and research on how the proposed methods may help reduce the total power consumption during the transmissions.

REFERENCES

[1] ABI Research, "Bluetooth low energy market set to triple by 2023, reaching 1.6 billion device shipments," https://www.abiresearch.com/press/bluetooth-low-energy-market-set-triple-2023-reaching-16-billion-device-shipments/.
[2] J. Yin, Z. Yang, H. Cao, T. Liu, Z. Zhou, and C. Wu, "A survey on bluetooth 5.0 and mesh: New milestones of iot," *ACM Transactions on Sensor Networks (TOSN)*, vol. 15, no. 3, pp. 1–29, 2019.
[3] J. Kim, S. Kim, S. Choi, and D. Qiao, "Cara: Collision-aware rate adaptation for ieee 802.11 wlans," in *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, 2006, pp. 1–11.
[4] P. Di Marco, P. Skillermark, A. Larmo, P. Arvidson, and R. Chirikov, "Performance evaluation of the data transfer modes in bluetooth 5," *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 92–97, 2017.
[5] B. Badihi, F. Ghavimi, and R. Jäntti, "On the system-level performance evaluation of bluetooth 5 in iot: Open office case study," in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2019, pp. 485–489.
[6] P. Bulić, G. Kojek, and A. Biasizzo, "Data transmission efficiency in bluetooth low energy versions," *Sensors*, vol. 19, no. 17, p. 3746, 2019.
[7] C. Zhang and Y. Yan, "Experimental performance evaluation of bluetooth5 for in-building networks," in *2020 11th International Conference on Network of the Future (NoF)*. IEEE, 2020, pp. 115–119.
[8] E. Park, M. Lee, and S. Bahk, "Adaptable: Data rate and transmission power adaptation for bluetooth low energy," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.
[9] A. Razavi, M. Valkama, and E. Lohan, "K-means fingerprint clustering for low-complexity floor estimation in indoor mobile localization," in *2015 IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–7.
[10] B. Altintas and T. Serif, "Improving rss-based indoor positioning algorithm via k-means clustering," in *17th European Wireless 2011 - Sustainable Wireless Technologies*, 2011, pp. 1–5.
[11] Z. D. Tekler, R. Low, B. Gunay, R. K. Andersen, and L. Blessing, "A scalable bluetooth low energy approach to identify occupancy patterns and profiles in office spaces," *Building and Environment*, vol. 171, p. 106681, 2020.
[12] M. Woolley, "Bluetooth core specification v5." Bluetooth, 2019.
[13] M. Collotta, G. Pau, T. Talty, and O. K. Tonguz, "Bluetooth 5: A concrete step forward toward the iot," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 125–131, 2018.
[14] Zach Michel, "Maximizing ble throughput part 3: Data length extension (dle)," https://punchthrough.com/maximizing-ble-throughput-part-3-data-length-extension-dle-2/.
[15] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
[16] Nordic Semiconductor, "Software development kit for the nrf52 series and nrf51 series socs," https://www.nordicsemi.com/Software-and-tools/Software/nRF5-SDK/.
[17] Nordic Semi, "S140 soft device specification," https://infocenter.nordicsemi.com/topic/struct_nrf52/struct/s140.html.