# Performance and Trade-offs of the new PHY Modes of BLE 5

(Paper published at the 1st Workshop on Pervasive Systems in the IoT era (PERSIST-IoT). July 2, 2019)

Michael Spörk
michael.spoerk@tugraz.at
Graz University of Technology
Graz, Austria

Carlo Alberto Boano
cboano@tugraz.at
Graz University of Technology
Graz, Austria

Kay Römer
roemer@tugraz.at
Graz University of Technology
Graz, Austria

## ABSTRACT

With the release of Bluetooth Low Energy (BLE) version 5, the Bluetooth Special Interest Group introduced three additional physical (PHY) modes for BLE communication. These PHY modes enable an application to either double its throughput, or significantly improve its reliability, making BLE applicable to an even wider range of application domains. Unfortunately, no experimental study has yet investigated the actual performance of BLE 5's PHY modes in BLE connections or shown their trade-offs between energy efficiency, reliability, and throughput. Thus, how to use BLE 5's PHY modes to achieve specific application requirements is still an open question.

To fill this gap, we experimentally study the performance of all four BLE 5 PHY modes in BLE connections and observe that it is, indeed, possible to double BLE's throughput or to increase BLE's reliability by using the new PHY modes. Furthermore, we provide guidelines using our measurements on how to select the most suitable PHY mode based on specific application requirements.

## CCS CONCEPTS

• **Networks** → **Network performance analysis**; • **Computer systems organization** → *Embedded systems.*

## KEYWORDS

Bluetooth Low Energy, BLE 5, Throughput, Reliability

## 1 INTRODUCTION

Bluetooth Low Energy (BLE) has become a pervasive wireless technology to connect constrained and low-power devices to the Internet of Things (IoT). BLE does not only provide an energy-efficent and reliable way of communication; its wide adoption in almost all consumer electronic devices, like smartphones and tablets, makes it the technology of choice for a wide range of application domains, such as smart health [6], smart cities [2], and smart homes [14].

In order to support an even wider range of applications, the Bluetooth Special Interest Group released version 5 of the Bluetooth specification, so called BLE 5, in June 2016 [4]. Besides a longer advertising packet length and increased coexistence capabilities, BLE 5 introduces three new physical (PHY) modes. One of these, the 2M PHY, promises to double BLE's throughput. The other two modes, the Coded S2 PHY and the Codes S8 PHY, promise to increase BLE's communication reliability [3].

Although BLE 5 devices have been available since 2017 [9] and even BLE version 5.1 (enabling a more advanced localization using multiple antennas) has been released in 2019, no experimental study has, to the best of our knowledge, investigated if the new PHY modes of BLE 5 actually perform as advertised when used in BLE connections. Furthermore, how to use the different PHY modes to sustain specific application requirements, such as a certain power consumption, communication reliability, or throughput, has not been studied in detail and still remains an open question.

**Contributions.** In this paper, we fill this gap and experimentally study the performance of BLE 5 and its new PHY modes. Our investigation allows to understand: (i) whether the BLE 5 PHYs deliver on their promises, and (ii) how to select the best PHY for a given application. To this end, we perform the first comprehensive experimental study of all four BLE 5 PHY modes used in a BLE connection and answer the following questions:

- Does the 2M PHY really allow to double the throughput?
- Do the Coded S2 PHY and the Coded S8 PHY really increase the relibility of a BLE connection?
- How does the chosen PHY mode affect the overall power consumption of a BLE device?

Based on these measurements, we show the trade-off between energy efficiency, reliability, and throughput for each PHY mode and provide guidelines on how to select the most suitable PHY for a given application. For this purpose, we derive the effective throughput and effective power consumption of all four PHY modes for BLE connections with different link quality and investigate:

- Which PHY provides the maximum effective throughput?
- Which PHY minimizes the effective power consumption?

The remainder of this paper is structured as follows: Sect. 2 provides the necessary technical background on the four PHY modes of BLE 5 and connection-based BLE. Sect. 3 describes our experimental setup and evaluates the (i) power consumption, (ii) data throughput, (iii) packet reception rate, and (iv) robustness to Wi-Fi interference of a BLE 5 connection using the four PHY modes. Sect. 4 provides guidelines on choosing the best PHY mode for specific application requirements. Sect. 5 lists related work and Sect. 6 summarizes our findings and discusses future work.

## 2 BLE 5 PRIMER

In this section, we provide the necessary technical details on the different PHY modes of BLE 5 (Sect. 2.1) and discuss how they are used in BLE connections for bi-directional data exchange (Sect. 2.2).



*Persist-IoT'19, July 02, 2019, Catania, Italy*



ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

| Preamble (1 or 2 bytes) | Access Address (4 bytes) | PDU (2 to 257 bytes) | CRC (3 bytes) |
|---|---|---|---|

**(a) BLE link layer packet format for the 1M and 2M PHY mode.**

| Preamble (10 bytes) | Access Address (4 bytes) | CI (2 bytes) | TERM1 (3 bytes) | PDU (2 to 257 bytes) | CRC (3 bytes) | TERM2 (3 bytes) |
|---|---|---|---|---|---|---|
| | FEC block 1 | | | FEC block 2 | | |

**(b) BLE link layer packet format for the Coded S2 and S8 PHY mode.**

**Figure 1: Overview of the BLE link layer packet structure.**

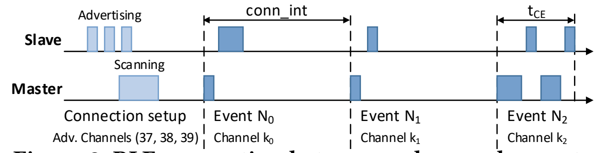



**Figure 2: BLE connection between a slave and a master.**

## 2.1 BLE 5 PHY Modes

With the publication of the BLE 5 specification [4] in June 2016, three additional physical (PHY) modes, the 2M, the Coded S2, and the Coded S8 PHY mode, were introduced. While the 2M PHY, where the M stands for Megasymbols/s (Msym/s), promises twice the data rate compared to the existing 1M PHY mode, the two Coded PHY modes are meant to increase the communication reliability of BLE devices [3]. Hence, the four PHY modes provide application developers with additional possibilities to fine-tune BLE's performance to individual application requirements, such as energy-efficiency, throughput, and reliability.

To achieve these different characteristics, the PHY modes use different error correction/detection, modulation, and coding schemes. Besides, all PHYs use Gaussian Frequency Shift Keying on the 40 BLE radio channels located in the 2.4 GHz ISM band.

*2.1.1 1M PHY.* This is the original mode of BLE and was the sole mode for all BLE communication with a version number below 5. Therefore, this is the only backwards-compatible mode that can be used with BLE devices not supporting BLE 5. In this mode, the modulation scheme supports a physical modulation of 1 Msym/s, meaning that transmitting a single bit of payload takes 1 $\mu$s. All packet data is not coded and, therefore, has no error correction.

Fig. 1a shows the link-layer packet format of the 1M PHY mode. The preamble is 1-byte long and is followed by 4 bytes containing the access address. After the Protocol Data Unit (PDU), which has a variable length between 2 and 257 bytes, the packet ends with a 3-byte CRC checksum, which is used to check for packet corruption.

*2.1.2 2M PHY.* In case an application needs to sustain a high throughput, it may use the 2M PHY mode of BLE. In contrast to the other three PHYs, this mode uses a physical modulation of 2 Msym/s, resulting in 0.5 $\mu$s of air time for a single payload bit. Similar to the 1M PHY, data sent with the 2M PHY is not coded and has no error correction. The 2M PHY link-layer packet format is similar to the format of the 1M PHY shown in Fig. 1a, however a 2-byte preamble is used. Hence, the new 2M PHY promises twice the throughput at the cost of a lower reliability for poor link qualities.

*2.1.3 Coded S2 PHY.* When a more reliable communication is needed (*e.g.*, due to a long communication distance or the presence of co-located radio interference), the Coded S2 PHY mode may be used. This mode uses a physical modulation of 1 Msym/s, but makes use of forward error correction (FEC) with a symbol coding of 2 (S2), leading to an increased robustness. A single data bit encoded with S2 coding takes 2 $\mu$s on the air, resulting in a data rate of 500 kb/s.

The link layer packet format of this mode is shown in Fig. 1b. As this figure shows, the packet is split into three parts: a preamble, a FEC block 1, and a FEC block 2. The preamble is 80 $\mu$s long and is sent without coding. The FEC block 1 contains the access address as well as the coding indicator (CI) and ends with the first termination field (TERM1). The FEC block 2 contains the PDU and a 3-byte CRC checksum and is terminated by a second termination field (TERM2). According to the BLE specification [4], even when a packet is sent with the Coded S2 PHY, the FEC block 1 always uses S8 Coding. Compared to the 1M PHY, this PHY improves BLE's reliability, at the cost of less throughput and an increased power consumption.

*2.1.4 Coded S8 PHY.* The Coded S8 PHY uses an even more robust coding and error correction scheme than the Coded S2 PHY. This PHY also transmits with a physical modulation of 1 Msym/s, but uses an FEC with a symbol coding of 8 (S8) for the whole packet. Using the Coded S8 PHY, a single data bit takes 8 $\mu$s on the air. Fig. 1b shows the link-layer packet format of the Coded S8 PHY mode. Using the Coded S8 PHY, all packet fields are sent with a coding of 8, resulting in a data rate of 125 kb/s for the whole packet. The Coded S8 PHY promises to improve BLE's reliability for poor link qualities even further, at the cost of a lower throughput and increased power consumption compared to the other PHYs.

## 2.2 BLE Connections

BLE supports two modes of communication: a *connection-less* and a *connection-based* mode. In the connection-less mode, a device is either broadcasting short data packets on the three BLE advertisement channels (37, 38, and 39) or scanning for such broadcast messages. However, if two devices need to bidirectionally exchange data packets, they need to use connection-less primitives to establish a BLE connection. In the connection-based mode, one device acts as a master and the other as a slave; communication takes place during *connection events* ($N_0$ ... $N_i$), as shown in Fig. 2.

The time between the start of two consecutive connection events is defined by the *connection interval* (*conn_int*). Every connection event starts with a link-layer packet from the master, to which the slave responds. In case master and slave have no additional data to send, the connection event ends after this mandatory exchange of keep-alive messages. If, however, more data needs to be transmitted, master and slave keep exchanging link-layer packets until all data is successfully sent or the *maximum connection event length* ($t_{CE}$) is reached. The last link-layer packet during a connection event is always sent by the slave, after which both devices disable their radio and resume communication at the next connection event.

Fig. 2 shows an example where, after the connection setup using connection-less primitives, the master starts connection event $N_0$ by sending a short keep-alive packet to the slave. The slave has data to send and therefore responds with a link-layer packet (which is longer than the keep-alive packet from the master) carrying the data. During connection event $N_1$, both devices have no data to transmit and therefore only exchange the mandatory keep-alive packets. In connection event $N_2$, however, the master has data to send and therefore starts the connection event by sending a link-layer packet carrying data. Because the master has additional data to send, it waits for the slave's response before sending a second

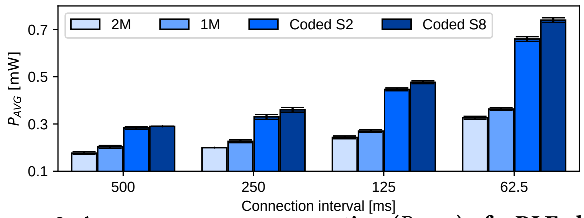



**Figure 3: Average power consumption ($P_{AVG}$) of a BLE slave for different connection intervals and PHY modes when using a fixed PDU length of 253 bytes.**

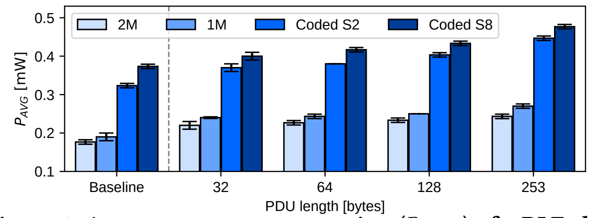



**Figure 4: Average power consumption ($P_{AVG}$) of a BLE slave for different PDU lengths and PHY modes when using a fixed connection interval of 125 ms.**

link-layer packet containing the rest of the data. The slave responds with a second link-layer packet, ending the connection event.

In the connection-based BLE mode, the link layer autonomously handles packet acknowledgment (ACK) and flow control using a 1-bit ACK field and a 1-bit sequence number in every link-layer packet header. If a link-layer packet was not successfully sent, it is automatically retransmitted. To further ensure reliable communication, BLE connections use adaptive frequency hopping (AFH). Using AFH, one of the enabled data channels in the *data channel map* is selected at the start of every connection event and is used by master and slave to exchange all packets during the event.

## 3 MEASURING THE PERFORMANCE OF BLE 5

To evaluate the performance of the four different PHY modes of BLE 5, we perform an experimental study (Sect. 3.1) and measure the power consumption (Sect. 3.2), the maximum achievable throughput (Sect. 3.3), the link-layer packet reception rate (Sect. 3.4), and the robustness under Wi-Fi interference (Sect. 3.5) of a BLE connection.

### 3.1 Experimental Setup

We perform our experiments on a testbed located in a vacant University lab (6x10 meters). The measurements were taken on a BLE connection between a BLE master and slave for all four PHY modes.

BLE master and slave have direct line of sight and use a transmission power of 0 dBm in all of our experiments.

**BLE master.** We use an nRF52840 DK device from Nordic Semiconductor [10] as a BLE master for all of our measurements. The master scans for a BLE slave and initiates a BLE connection. After the connection has been established, the master subscribes to a custom Generic Attribute Profile (GATT) attribute on the slave.

**BLE slave.** We use another nRF52840 DK device as a BLE slave that advertises its presence and waits for a BLE connection to be initiated by the BLE master. Once the master has successfully subscribed to the custom GATT attribute, the slave periodically notifies the master with a GATT notification. In our experiments, we are able to vary the length of the GATT notification in bytes and the time between two consecutive notifications.

**Controlling the BLE PHY mode.** Both master and slave run the Zephyr operating system [19]. Zephyr provides a standard-compliant BLE stack allowing access to the inner workings of the BLE link layer. This way, we have fine-grained control over the connection settings and the PHY mode of the BLE connection.

### 3.2 Power Consumption

To evaluate the impact of the different PHY modes on the power consumption of a system, we use the experimental setup described in Sect. 3.1 and measure the power consumption of a BLE slave in different configurations. We focus on the power consumption of the slave, as it usually operates on a constrained energy budget, such as a coin cell battery, while the BLE master usually has a continuous power supply. The power consumption of a master sustaining a single BLE connection is comparable to the slave's consumption, but increases with every connection that needs to be maintained.

Once the master has established a BLE connection and has subscribed to the custom GATT attribute, the slave periodically sends a notification of configurable length to the master every 1000 ms. In this set of experiments, master and slave have a distance of approximately 1 meter and direct line of sight. We disable all debugging and application logging features on the slave and use the Monsoon Power Monitor [8] to measure the system's power consumption. We vary the used PHY mode, connection interval, and PDU length and measure the average power consumption ($P_{AVG}$) of the slave for every settings over 120 seconds. We repeat the measurements for each configuration five times to ensure statistical significance.

Fig. 3 shows $P_{AVG}$ for different PHY modes and connection intervals when using a fixed PDU length of 253 bytes. First, we can clearly observe that $P_{AVG}$ increases when using a lower connection interval, as the BLE radio is more active (see Sect. 2.2). This matches our expectations as well as the models and measurements shown in [17]. Second, we can see a significant difference in power consumption when different PHY modes are used. As expected, the 2M PHY mode results in the lowest $P_{AVG}$, as it has the lowest radio duty cycle due to its fast data rate. The Coded S8 PHY, however, leads to the highest power consumption, because of its higher radio duty cycle caused by the overhead of the employed coding scheme. Compared to the legacy 1M PHY, the 2M PHY consumes approximately 8% less power in our experiments. The Coded S2 and S8 PHY consume approximately 61% and 70% more power compared to the 1M PHY for all four connection intervals, respectively.

Fig. 4 shows $P_{AVG}$ of the slave for different PHY modes and PDU lengths when using a fixed connection interval of 125 ms. The values labeled `Baseline` shows $P_{AVG}$ when the BLE connection is alive, no data is transmitted by the application, therefore showing the power consumed for maintaining the BLE connection (*i.e.*, to exchange only keep-alive packets). Similar to the data shown in Fig. 3, the 2M PHY mode is the most energy efficient, while the Coded S8 PHY mode results in the highest power consumption.

Comparing Fig. 3 and Fig. 4 we see that the used connection interval has a high impact on $P_{AVG}$. For example, a BLE slave using the 2M PHY mode consumes approximately 85% more power when using a connection interval of 62.5 ms instead of 500 ms. Using the Coded S8 PHY, a slave consumes even 155% more power when using

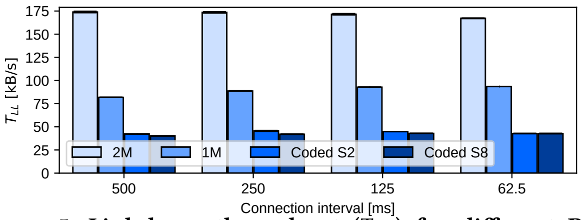



**Figure 5: Link-layer throughput ($T_{LL}$) for different PHY modes and connection intervals measured at the master.** *Note that the standard deviation is at most ±0.4% in our tests and, therefore, not clearly visible in the figure.*

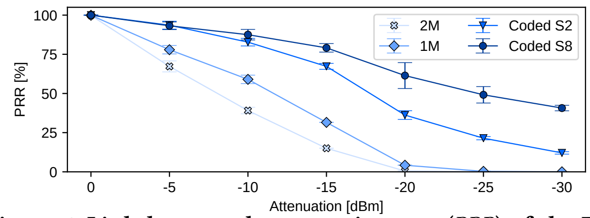



**Figure 6: Link-layer packet reception rate (*PRR*) of the BLE connection for different PHY modes and attenuation values.**

a connection interval of 62.5 ms instead of 500 ms. The used PDU length, however, only slightly increases the power consumption, leading to 20% more consumption when sending a PDU length of 253 bytes instead of 32 bytes for the Coded S8 PHY mode. We also see that the power used for maintaining the BLE connection (shown as `Baseline` in Fig. 4) accounts for a significant portion, between 72% and 91%, of the overall power consumption of the system.

## 3.3 Throughput

Next, we measure the maximum achievable throughtput of the different PHY modes of BLE 5. We use the experimental setup discussed in Sect. 3.1 and keep BLE master and slave at an approximate distance of 1 meter with direct line of sight. Similar to Sect. 3.2, the master initiates a BLE connection and subscribes to the custom GATT attribute on the slave. However, for these measurements the slave sends a new notification every 5 ms with a PDU length of 253 bytes, filling the outgoing transmission buffer of the slave.

The master application measures the time it takes to receive 10000 subsequent GATT notifications from the slave and uses the measured time to calculate the link-layer throughput ($T_{LL}$) of the BLE connection. Every throughput measurement consists of 10 test runs, where each run consists of 10 throughput measurements (over 10000 GATT notifications) per PHY mode and connection interval.

To achieve the maximum possible throughput, we configure the BLE devices to use the maximum number of link-layer transmission and reception buffers (18 and 19, respectively) and increase the L2CAP buffer and fragment count to 50. This maximizes the connection event length $t_{CE}$ of the BLE connection (see Sect. 2.2) and hence the number of packets sent during a connection event.

Fig. 5 shows the $T_{LL}$ for different PHY modes and connection intervals measured at the master. As expected, the 2M PHY, having a physical modulation of 2 Msym/s, provides the highest throughput of all PHYs, while the Coded S8 PHY has the lowest $T_{LL}$ in our experiments. According to our measurements, the 2M PHY provides between 178% and 212% of the 1M PHY mode's throughput, therefore keeping its promise of doubling its throughput [3]. Contrary to our expectations in Sect. 2.1, the Coded S8 PHY mode provides almost 50% of the throughput of the 1M PHY mode. This, however, can be explained by the behavior of the BLE stack of Zephyr [19] on the nRF52840 platform. Whenever the transmission buffers of the BLE link layer get filled, the logic in the link layer implementation always uses the Coded S2 PHY, even if the Coded S8 PHY was chosen by the developer. Therefore, although we configure master and slave to use the Coded S8 PHY, they autonomously switch to the Coded S2 PHY mode when the BLE buffers are filled.

Another observation from Fig. 5 is that the used connection interval has no significant impact on $T_{LL}$ in our experiment, because we configure the BLE buffers on master and slave so that multiple notification packets can be transmitted in a single connection event.

## 3.4 Packet Reception Rate

To estimate the differences in communication reliability of the PHY modes of BLE 5, we measure the link-layer packet reception rate (PRR) for different link qualities. We use the experimental setup described in Sect. 3.1, but change the distance between the master and slave to approximately 10 meters with direct line of sight. In order to accurately and repeatably measure the PRR for different link qualities, we connect a programmable attenuation device [7] and an external 2.4 GHz antenna to the antenna connector of the slave. This programmable attenuator allows us to have fine-grained control over the antenna attenuation on the slave, which we use to lower the link quality of the BLE connection in 5 dBm steps.

The master initiates a BLE connection with a connection interval of 25 ms and subscribes to the custom GATT attribute of the slave, similar to the experiments described above. The BLE slave tries to send a notification with a PDU length of 253 bytes every 25 ms. If a notification is pending (*i.e.*, it has not been successfully sent yet), no new notification is added by the slave, leading to a maximum of one notification sent per connection event.

We insert log ouputs into the link layer implementation of the BLE master to accurately measure the link-layer PRR of the BLE connection. The master's link layer logs the number of link-layer transmissions and retransmissions, which we use to calculate the PRR. We also limit the data channel map of the BLE connection to the BLE channels 12 to 19, as these channels are not interfered by any co-located radio technology; this minimizes the effects of RF noise on our measurements. We measure the PRR for every PHY and attenuation setting over more than 3000 connection events per setting and repeat each test 10 times.

Fig. 6 shows the PRR of the four PHY modes of BLE 5 for different attenuation values. The x-axis of Fig. 6 shows the effective attenuation of the antenna of the slave (the +3 dBm of the external 2.4 GHz antenna minus the configured attenuation on the programmable attenuator). As expected, the 2M PHY mode has the lowest PRR out of the four available PHY modes. For example, while the Coded S2 and S8 PHYs provide a PRR of 67% and 80% for an attenuation of -15 dBm, the 2M PHY is only able to sustain a 15% PRR. The 1M PHY is able to sustain a PRR of 32% for an attenuation of -15 dBm. Hence, the Coded S2 PHY and the Coded S8 PHY increase the link-layer PRR of BLE connections for poor link qualities, and therefore BLE's reliability, due to their employed coding schemes.

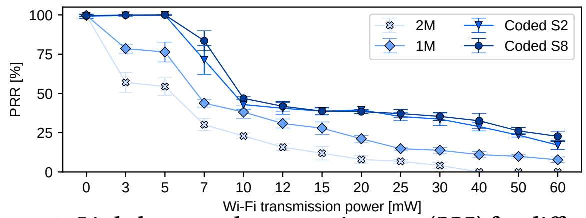



**Figure 7: Link-layer packet reception rate (*PRR*) for different PHY modes and interfering Wi-Fi transmission power.** *Note the non-linear x-axis scale showing the Wi-Fi transmission power.*

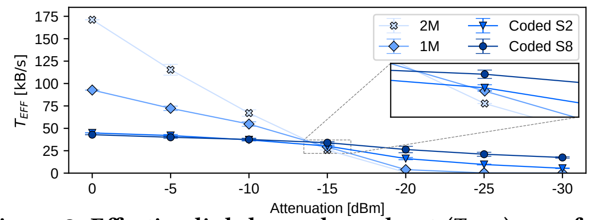



**Figure 8: Effective link-layer throughput ($T_{EFF}$) as a function of the PHY mode and the attenuation of the BLE slave antenna when using a connection interval of 125 ms.**

## 3.5 Robustness to Interference

We finally experimentally evaluate the robustness under Wi-Fi interference of the four PHY modes of BLE 5. To this end, we measure the link-layer packet reception rate (PRR) of a BLE connection in the presence of co-located Wi-Fi interference. We make use of the experimental setup from Sect. 3.1 and introduce a Raspberry Pi 3 (RPi3) [12], which we use to generate repeatable Wi-Fi interference. For this experiment, we place the BLE master and slave at a distance of approximately 3 meters with direct line of sight. The RPi3 used for Wi-Fi jamming is placed at a distance of 1 meter to the slave and 4 meters to the master. To create Wi-Fi interference, we use `JamLab-NG` [15] and let the RPi3 generate IEEE 802.11b packets on Wi-Fi channel 6 using its on-board Broadcom `bcm43438` radio. We let the RPi3 send a 1500-byte long packet every 10 milliseconds.

Similar to the previous experiments, the master initiates a BLE connection with the slave and subscribes to the slave's custom GATT attribute. The BLE connection is configured to use a connection interval of 125 ms and uses only the BLE data channels 12 to 19 that all overlap with the Wi-Fi channel 6 where interference in generated. The slave sends a GATT notification with a PDU length of 253 bytes in the same way as described in Sect. 3.4. We use the log output of the master's link layer and count the link-layer transmissions and retransmissions, which we use to calculate the PRR of the BLE connection. We measure the PRR for 5 minutes (resulting in over 2400 values per test) for every PHY and Wi-Fi transmission power and repeat the experiment 10 times for each setting.

Fig. 7 shows the average PRR of the BLE connection for different PHY modes and Wi-Fi transmission power settings. Our measurements show that, as expected, the Coded S8 PHY mode provides the highest PRR and thus the highest reliability under interference. The data also show that the Coded S2 and S8 PHY increase the link budget by 5dBm under Wi-Fi interference. While the Coded S2 and S8 PHYs are able to sustain almost 100% PRR, the 2M PHY only provides a PRR of 54% for a Wi-Fi transmission power of 5mW.

# 4 CHOOSING THE MOST SUITABLE PHY

The measurements in Sect. 3 show that the used BLE 5 PHY mode significantly influences the energy efficiency, throughput, and reliability of a BLE connection. Based on our measurements, we investigate how to maximize the effective throughput (Sect. 4.1) and minimize the effective power consumption (Sect. 4.2) by selecting the most suitable PHY mode (Sect. 4.3).

## 4.1 Maximizing Throughput

In this section, we answer the question: *Which PHY mode of BLE 5 provides the maximum effective data throughput?* To this end, we calculate the effective link-layer throughput of the four PHY modes of BLE 5 for different link qualities of the used BLE connection.

We define the effective link-layer throughput ($T_{EFF}$) as the number of link-layer bytes per second that are successfully sent over a BLE connection and calculate $T_{EFF}$ as

$$T_{EFF} = PRR \cdot T_{LL}, \tag{1}$$

where $PRR$ is the measured link-layer packet reception rate for a given PHY mode and antenna attenuation (shown in Sect. 3.4), whereas $T_{LL}$ is the maximum achievable link-layer throughput of a given PHY mode and connection interval (shown in Sect. 3.3).

Fig. 8 shows $T_{EFF}$ for different BLE antenna attenuations when using a connection interval of 125 ms. The data suggest that the best PHY mode to sustain a maximum effective throughput depends on the link quality of the BLE connection (indicated by the BLE antenna attenuation). In case only a few link-layer data packets are corrupted and thus need to be retransmitted, the 2M PHY mode provides the highest $T_{EFF}$. If packets are frequently corrupted, because of a poor link quality of the underlying BLE connection, the Coded S8 PHY is able to recover most corrupted packets and hence achieves the highest effective throughput. Using the 1M or the Coded S2 PHY mode always leads to a suboptimal $T_{EFF}$.

Results for connection intervals of 62.5 ms, 250 ms, and 500 ms show similar behavior, but are omitted due to space constraints.

## 4.2 Minimizing Power Consumption

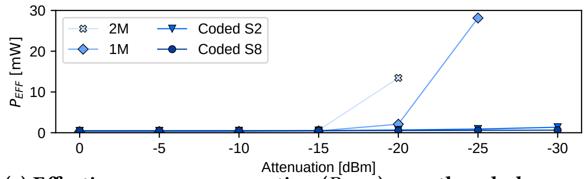
In this section, we answer the question: *Which PHY mode of BLE 5 minimizes the effective power consumption?* To this end, we calculate the effective power consumption of a slave using the four PHY modes of BLE 5 for different link qualities of the BLE connection.

We define the effective power consumption ($P_{EFF}$) as the overall power consumption of a slave periodically transmitting application data, accounting for the additional power consumption introduced by packet retransmissions. $P_{EFF}$ is calculated as

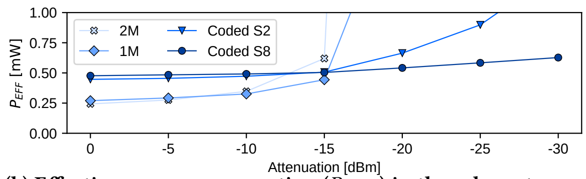
$$P_{EFF} = P_M + \frac{P_{DATA}}{PRR}, \tag{2}$$

where $P_M$ is the overall power consumption of a slave for maintaining the BLE connection, *i.e.*, to only exchange the mandatory keep-alive packets. The $P_M$ value for a given connection inverval and PHY mode is shown in Fig. 4 labeled as `Baseline`. *PRR* is the link-layer packet reception rate for a given PHY mode and antenna attenuation (shown in Sect. 3.4). Finally, $P_{DATA}$ is the power consumed for transmitting the actual application data over the BLE connection. By measuring the average power consumption while transmitting data ($P_{AVG}$) and the power consumed for maintaining the BLE connection ($P_M$), $P_{DATA}$ can be calculated as

$$P_{DATA} = P_{AVG} - P_M. \tag{3}$$




**(a) Effective power consumption ($P_{EFF}$) over the whole range.**




**(b) Effective power consumption ($P_{EFF}$) in the relevant range.**

**Figure 9: Effective power consumption ($P_{EFF}$) as a function of the used PHY and antenna attenuation of a BLE slave using a connection interval of 125 ms.** *Note that (a) shows $P_{EFF}$ over its whole range, while (b) shows the range relevant for analysis.*

Fig. 9 shows the effective power consumption ($P_{EFF}$) of a BLE slave when using a connection interval of 125 ms and a PDU length of 253 bytes. Similar to the analysis in Sect. 3.3, the data in Fig. 9 suggest that the most energy efficient PHY mode mainly depends on the link quality of the BLE connection. In case only a few link-layer packets are corrupted and therefore retransmitted, due to a good link quality, the 2M PHY mode provides the most energy-efficient communication. When packets are frequently corrupted, the Coded S8 PHY leads to a lower power consumption, as most corrupted packets can be recovered and do not need to be retransmitted. In contrast to Sect. 3.3, we can see that during a small transition area (an attenuation between -10 dBm and -15 dBm) the 1M PHY mode slightly outperforms the other PHYs in our experiments.

Similar results for $P_{EFF}$ can be observed for a connection interval of 250 ms and 500 ms, but are omitted due to space constraints.

### 4.3 PHY Mode Selection

The previous sections suggest that choosing the most suitable PHY mode to achieve a maximum throughput or a minimum power consumption requires information about the link quality or the number of link-layer retransmissions of the underlying BLE connection. Unfortunately, off-the-shelf BLE devices do not provide developers with this information per default. However, a developer may use the approach presented by Spörk et al. [18] to statically measure the link quality of a BLE connection during device deployment and manually select the most suitable PHY mode for an application.

Furthermore, an application may use the approach in [18] to dynamically measure the link quality and adapt the used PHY mode at runtime. To this end, a device can use the standardized *L2CAP PHY Update Procedure* of BLE 5 to change the used PHY mode [4].

## 5 RELATED WORK

Several studies have experimentally investigated the key performance metrics of BLE. While most of these studies have focused on connection-less BLE [14], a few evaluated the connection-based mode [16, 17]. The latter compare the performance of BLE to IEEE 802.15.4 and conclude that BLE is, indeed, more energy efficient at the cost of a shorter communication range. However, these works used BLE v4.0 [16] and v4.1 [17] for their measurements.

A few works [1, 5, 11, 13] investigate BLE 5 and its different PHY modes. Ray and Agarwal [13] describe the capabilities of BLE 5, including its PHY modes, but only theoretically discuss BLE 5's potential in the IoT. Others experimentally investigate BLE 5's different PHY modes for connection-less BLE communication [1, 5, 11]. These works conclude that the used PHY has an effect on the power consumption and throughput when used in BLE advertising.

In this paper, to the best of our knowledge, we provide the first experimental study that investigates the performance of all four PHY modes of BLE 5 when using connection-based BLE. Furthermore, we are the first to provide guidelines for selecting the best PHY mode to achieve specific application requirements.

## 6 CONCLUSIONS AND FUTURE WORK

In this paper, we provide the first experimental performance analysis of BLE 5's new PHY modes in BLE connections. We highlight the trade-offs of each PHY mode and show how the used PHY mode affects the energy efficiency, communication reliability, and throughput of a connection-based BLE application. We further provide guidelines showing how to select the most suitable PHY mode to sustain specific application requirements, such as a minimum effective power draw or a maximum effective throughput.

Our results can be used to improve the performance of existing BLE applications. Furthermore, BLE applications may use our results and guidelines to dynamically adapt the used PHY at runtime.

## ACKNOWLEDGMENTS

This work has been performed within the LEAD project "Dependable Internet of Things in Adverse Environments" funded by Graz University of Technology.

## REFERENCES

[1] B. Al Nahas and S. Duquennoy and O. Landsiedel. 2019. Concurrent Transmissions for Multi-Hop Bluetooth 5. In *Proc. of the $16^{th}$ EWSN Conf.*
[2] A. K. Bhattacharjee et al. 2017. Extending Bluetooth Low Energy PANs to Smart City Scenarios. In *Proc. of the SMARTCOMP Conf.*
[3] Bluetooth SIG. 2016. Bluetooth 5 Brochure: 5 is the Future . https://www.bluetooth.com/~/media/files/marketing/bluetooth5-brochure-web-144ppi.ashx.
[4] Bluetooth SIG. 2018. Bluetooth Core Specification v5.0.
[5] M. Collotta et al. 2018. Bluetooth 5: A concrete step forward toward the IoT. *IEEE Communications Magazine* 7 (2018).
[6] M. Ghamari et al. 2016. A Survey on Wireless Body Area Networks for eHealthcare Systems in Residential Environments. *Sensors* 16. Issue 6.
[7] Mini-Circuits. 2017. RCDAT-8000-90 - Programmable Attenuator.
[8] Monsoon Solutions Inc. 2015. Monsoon Power Monitor.
[9] Nordic Semiconductor. 2017. Bluetooth 5 in Samsung Smartphone. https://blog.nordicsemi.com/getconnected/bluetooth-5-in-samsung-smartphone.
[10] Nordic Semiconductors. 2018. nRF52840 Specifications.
[11] G. Pau et al. 2017. Bluetooth 5 Energy Management through a Fuzzy-PSO Solution for Mobile Devices of Internet of Things. *Energies* 7 (2017).
[12] Raspberry Pi Foundation. 2018. RASPBERRY PI 3 MODEL B - Raspberry Pi.
[13] P. P. Ray and S. Agarwal. 2016. Bluetooth 5 and Internet of Things: Potential and Architecture. In *Proc. of the 2016 Int. IEEE SCOPES Conf.*
[14] T. Renzler, M. Spörk, C.A. Boano, and K. Römer. 2018. Improving the Efficiency and Responsiveness of Smart Objects Using Adaptive BLE Device Discovery. In *Proc. of the 4th ACM MobiHoc SMARTOBJECTS Workshop.*
[15] M. Schuß et al. 2019. JamLab-NG: Benchmarking Low-Power Wireless Protocols under Controllable and Repeatable Wi-Fi Interference. In *Proc. of the EWSN Conf.*
[16] M. Siekkinen et al. 2012. How Low Energy is Bluetooth Low Energy? Comparative Measurements with ZigBee/802.15.4. In *Proc. of the IEEE WCNCW Workshop.*
[17] M. Spörk et al. 2017. BLEach: Exploiting the Full Potential of IPv6 over BLE in Constrained Embedded IoT Devices. In *Proc. of the $15^{th}$ ACM SenSys Conf.*
[18] M. Spörk, C.A. Boano, and K. Römer. 2019. Increasing the Timeliness of Bluetooth Low Energy in Noisy RF Environments. In *Proc. of the $16^{th}$ EWSN Conf.*
[19] The Zephyr Project. 2018. Zephyr OS: An RTOS for IoT.