

EE-IoT: An Energy-Efficient IoT Communication Scheme for WLANs

Hossein Pirayesh, Pedram Kheirkhah Sangdeh and Huacheng Zeng

Department of Electrical and Computer Engineering, University of Louisville

Email: {hossein.pirayesh, pedram.kheirkhahsangdeh, huacheng.zeng}@louisville.edu

Abstract—While Narrow-Band Internet of Things (NB-IoT) has been standardized by 3GPP to provide wireless Internet access for IoT devices, this service is expected to come with a monthly fee (e.g., \$1 or \$2 per month per device). As the number of IoT devices tends to be large, the service charge will impose a considerable financial burden on the end users. In this paper, we propose an Energy-Efficient IoT (EE-IoT) communication scheme by taking advantage of the existing WiFi infrastructure that is widely available in home, office, campus, and city environments. EE-IoT will not only avoid monthly service charge for the end users but also maintain a low power consumption for IoT devices. The key component of EE-IoT is an asymmetric physical (PHY) design, which enables an OFDM-based broadband AP to communicate with multiple QAM-based narrowband IoT devices at a low sampling rate (250 kbps) in both uplink and downlink. The trick in our design is that, instead of using the same carrier frequency as the AP, each IoT device tunes its carrier frequency to a particular subcarrier of the AP's OFDM signal, making it possible to encode/decode the data on that subcarrier at a low sampling rate. Based on this new PHY, we propose a MAC protocol to enable EE-IoT in WLANs. We have built a prototype of EE-IoT on a USRP2 wireless testbed and evaluated its performance in an office building environment. Experimental results show that an AP can serve 24 IoT devices simultaneously and each IoT device can achieve more than 187 kbps in the downlink and more than 125 kbps in the uplink.

Index Terms—Internet of Things, energy efficiency, Wi-Fi, wireless communication, wireless local area network

I. INTRODUCTION

The Internet of Things (IoT) is the network of physical devices with sensing, communicating, and actuating capabilities to integrate physical world into computer-based systems, resulting in efficiency improvements, economic benefits, and reduced human exertions. As IoT devices are typically battery-powered and limited by their physical size, energy-efficient wireless connection is a key element for them to communicate over the Internet. Narrowband IoT (NB-IoT), which is a Low Power Wide Area Network (LPWAN) radio technology standard [1], has been developed by 3GPP to enable a wide range of cellular services for IoT devices.

While NB-IoT has been standardized as a part of LTE, there are two concerns about its commercial applications. First, similar to cellular services for mobile phones, NB-IoT services will not come free. Users have to pay monthly fee to enjoy the NB-IoT services (e.g., \$1 or \$2 per month per device). Although this fee is not much compared to one's mobile phone bill, it easily becomes significant if one has many IoT devices on demand for Internet service. The monthly charge of NB-

IoT services imposes a severe financial burden on the end users. Second, cellular networks are already very crowded. Serving extra billions of IoT devices may result in traffic congestion in cellular networks, especially considering the fact that the licensed spectrum bands suitable for energy-efficient IoT communications (below 6 GHz) are limited.

In this paper, we study IoT communications in wireless local area networks (WLANs). This study is motivated by the following observations. First, WiFi is the dominant Internet service provider in indoor environments. It also has a large outdoor coverage in urban and suburban areas. By upgrading WiFi access point's (AP's) air interface, the existing WiFi infrastructure can be leveraged to provide wireless Internet service for a large portion of IoT devices and avoid the monthly fee. Second, WiFi has demonstrated its success as an Internet provider for mobile devices. It is estimated that by 2020, WiFi will carry 38.1 exabytes traffic per month, continuing to exceed the monthly traffic in cellular networks (30.6 exabytes). As expected, enabling IoT communications in WLANs will dramatically offload the cellular IoT traffic, thereby mitigating the traffic congestion in cellular networks. Given its potential, a successful design of practical IoT communication scheme for WLANs will not only alleviate the above two concerns on NB-IoT services but also boost the prosperity and evolution of the IoT ecosystem.

To design a practical IoT communication scheme for WLANs, the challenge lies in preserving the energy efficiency of IoT devices. As most IoT devices are battery-powered and expected to work for many years without battery replacement, it is critical to minimize their power consumption for wireless communications. Simply embedding WiFi client chipset into IoT devices is not a plausible solution as it consumes too much energy for communications. To address this challenge, we propose an energy-efficient IoT (EE-IoT) communication scheme for WLANs. The key component of our EE-IoT scheme is an asymmetric physical (PHY) design, which enables seamless uplink and downlink data transmissions between OFDM-based broadband WLAN AP and multiple QAM-based (non-OFDM) narrowband IoT devices. The asymmetric PHY is designed based on intrinsic properties of OFDM modulation and frequency mixer. In this asymmetric PHY, the AP preserves its legacy hardware architecture to transmit/receive OFDM-modulated broadband signals. But for each IoT device, instead of receiving/transmitting the broadband OFDM-modulated signal from/to the AP, it only receives/transmits narrowband sig-

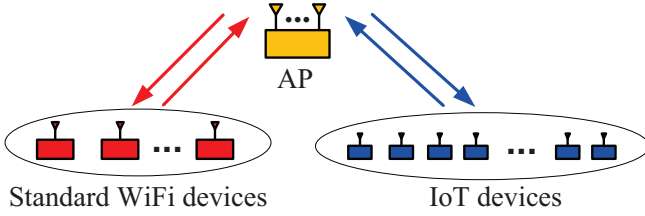


Fig. 1: A WLAN with standard WiFi devices and IoT devices.

nal on a single subcarrier by tuning its carrier frequency to that subcarrier. By doing so, the IoT devices can use a much lower sampling rate (250 ksps) for signal transmission/reception and do not require computation-intensive FFT/IFFT operations in their baseband signal processing, thereby leading to a significant reduction in their hardware complexity and power consumption. Based on the asymmetric PHY, we propose a semi-centralized MAC protocol for EE-IoT.

We have built a prototype of EE-IoT on a GNURadio-USRP2 wireless testbed and have evaluated its performance in an office building wireless environment. Experimental results show that EE-IoT can serve 24 IoT devices simultaneously in a 802.11-based OFDM frame in both uplink and downlink, and each IoT device can achieve more than 187 kbps in downlink and more than 125 kbps in uplink.

II. PROBLEM STATEMENT

Network Setting: We consider a WLAN as shown in Fig. 1, where an AP serves both standard WiFi devices and IoT devices. To serve two types of devices, the AP uses time division multiplexing. That is, the AP serves two types of devices in different time slots. Thus, there is no interference or interaction between WiFi devices and IoT devices. In this paper, we focus on the wireless communications between the AP and the IoT devices.

Design Objective: In such a WLAN, our objective is to design an energy-efficient scheme to enable uplink and downlink communications between the AP and the IoT devices. We note that the term “energy-efficient” is to emphasize the low power consumption for the IoT devices, which will be achieved through reducing their sampling rate and baseband signal processing complexity. Since the AP typically has sufficient power supply, its energy consumption is not considered in our design. In addition, we aim to preserve the AP’s hardware architecture as much as possible when extending its service from standard WiFi devices to IoT devices.

III. MATHEMATICAL FOUNDATION OF EE-IoT

In WLANs, the AP uses OFDM modulation for data transmission. Denote N as the number of FFT/IFFT points, which is also the number of subcarriers (e.g., $N = 64$ in 802.11ac). Denote $[X(0), X(1), \dots, X(N-1)]$ as the frequency-domain data sequence of an OFDM symbol. Then, the time-domain data sequence of the OFDM symbol, which we denote as

$[x(0), x(1), \dots, x(N-1)]$, can be obtained through IFFT operation as follows:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \cdot e^{j \frac{2\pi}{N} nk}. \quad (1)$$

For notational convenience, we reorganize the frequency-domain sequence $[X(0), X(1), \dots, X(N-1)]$ by defining $X_s(k)$ as follows:

$$X_s(k) = \begin{cases} X(k), & 0 \leq k \leq N/2; \\ X(k+N), & -N/2+1 \leq k \leq -1. \end{cases} \quad (2)$$

Then, the IFFT operation in (1) can be rewritten as follows:

$$x(n) = \frac{1}{N} \sum_{k=-N/2+1}^{N/2} X_s(k) \cdot e^{j \frac{2\pi}{N} nk}. \quad (3)$$

At the AP, the time-domain discrete data sequence $[x(0), x(1), \dots, x(N-1)]$ is then converted to continuous signal by digital-to-analog converter (DAC). The resulting baseband waveform, which we denote as $x_b(t)$, can be written as:

$$x_b(t) = A \sum_{k=-N/2+1}^{N/2} X_s(k) \cdot e^{j2\pi k \Delta f t}, \quad (4)$$

where Δf is OFDM subcarrier spacing bandwidth (e.g., $\Delta f = 312.5$ kHz in 802.11ac) and A is a constant that denotes signal amplitude.

Then, the baseband waveform is up-converted to radio frequency (RF) signal with carrier frequency f_c . The resulting radio signal can be written as:

$$x_r(t) = A \sum_{k=-N/2+1}^{N/2} X_s(k) \cdot e^{j2\pi k \Delta f t} \cdot e^{j2\pi f_c t}, \quad (5)$$

which can be rewritten as:

$$x_r(t) = A \sum_{k=-N/2+1}^{N/2} X_s(k) \cdot e^{j2\pi (f_c + k \Delta f) t}. \quad (6)$$

Suppose that the AP uses a single subcarrier, say subcarrier k , in the OFDM symbol to send data to an IoT device. That is, the AP puts payload on subcarrier k and puts zeros to other subcarriers. Then, the transmitted radio signal at the AP can be written as:

$$x_r(t) = A \cdot X_s(k) \cdot e^{j2\pi (f_c + k \Delta f) t}. \quad (7)$$

Assume that the signal on the subcarrier experiences flat fading from the AP to the IoT device, which is true in practice as a single subcarrier is narrowband (312.5 kHz). Then, the received radio signal at the IoT device can be written as:

$$y_r(t) = B \cdot X_s(k) \cdot e^{j2\pi (f_c + k \Delta f) t}, \quad (8)$$

where B is a constant complex number that characterizes path loss and flat fading coefficient.

Equation (8) indicates that, if the IoT device wants to decode the signal on subcarrier k , it does not require OFDM demodu-

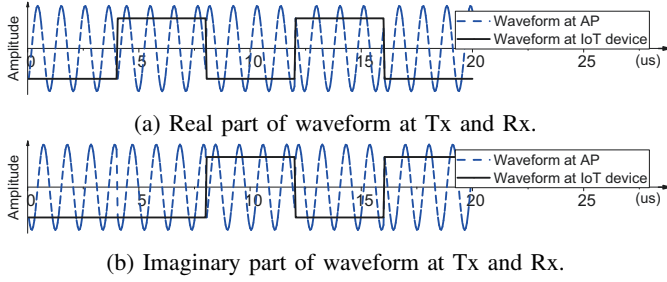


Fig. 2: The baseband signal waveform at OFDM AP versus the baseband signal waveform at non-OFDM IoT device.

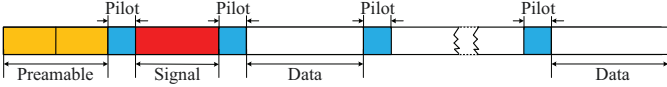


Fig. 3: Physical-layer frame format.

lation. Instead, it can use center/carrier frequency $f_c + k\Delta f$ to down-convert the radio signal. The down-converted baseband signal, which we denote as $y(t)$, can be written as:

$$\begin{aligned} y(t) &= B \cdot X_s(k) \cdot e^{j2\pi(f_c + k\Delta f)t} \cdot e^{-j[2\pi(f_c + k\Delta f)t - \phi]}, \\ &= Be^{j\phi} \cdot X_s(k), \end{aligned} \quad (9)$$

where ϕ is the phase offset between radio carrier signal and the local clock signal generated by IoT device's oscillator.

The result in (9) has the following two implications. First, the received baseband signal at the IoT device is a constant over the time duration of an OFDM symbol. This means that the required sampling rate at the IoT device is the OFDM symbol rate. For instance, Fig. 2 shows the baseband waveforms at the AP and the IoT device when a single subcarrier is used for data transmission. It is evident to see that 250 kps sampling rate is sufficient for the IoT device to decode the signal on the subcarrier. Second, $Be^{j\phi}$ in (9) represents the compound channel effect on baseband signal, which is a complex constant in block fading channel. Hence, it is easy for the IoT device to decode the signal of interest if the reference signals are embedded on the transmitter side. These two observations lay the mathematical foundation for our design of an asymmetric PHY for EE-IoT.

IV. PHY DESIGN FOR EE-IoT: DOWNLINK

In this section, we design a practical PHY to enable downlink data transmission from an OFDM-based broadband AP to multiple (non-OFDM) narrowband IoT devices. In what follows, we first present our proposed frame format for data transmission and then present the PHY design for single-user case. Finally, we extend our PHY design to multi-user case.

A. Frame Format

Fig. 3 depicts our proposed frame format. We elaborate each field of the frame as follows:

- **Preamble field:** The preamble field is designed for synchronization and channel estimation. It consists of two

identical Zadoff-Chu sequences of M_p symbols (e.g., $M_p = 12$ in our experiment).

- **Signal field:** The signal field is used to define the modulation and coding scheme (MCS) used in the data field as well as the total length of the frame. The MCS type of the symbols in this field is fixed (e.g., BPSK and 1/2 coding rate). The number of symbols in this field, which we denote as M_s , can be flexibly defined.
- **Pilot field:** The pilot field is one reference symbol, which is used to correct phase offset for signal detection at the receiver.
- **Data field:** The data field is used to carry payloads. The number of symbols in this field, denoted as M_d , can be user-defined (e.g., $M_d = 50$ in our experiment).

B. PHY Design: Single-User Case

Fig. 4 presents our downlink PHY design for an OFDM-based AP to serve a single IoT device. As shown in Fig. 4, The AP preserves its legacy architecture. Specifically, the AP does not require hardware modification to serve IoT device. The only manipulation is that the AP uses a single subcarrier (say subcarrier k) for data transmission and leaves other subcarriers unused. This is easy to be done through upper-layer control.

We now focus on the PHY design for the IoT device. Fig. 4 shows its PHY modules, which we elaborate as follows:

Local Carrier Frequency: To decode the signal from the AP, the IoT device tunes its local carrier frequency to $f_c + k\Delta f$, where f_c is the carrier frequency used at the AP and k is the index of the subcarrier used for payload. Note that the index of subcarrier is reordered in (2) and thus k is in the range of $[-N/2 + 1, N/2]$.

Bandwidth of LPF: For the down-converted signal from the frequency mixer, the IoT device uses a low pass filter (LPF) to suppress the noise. Configuration of its bandwidth is a trade-off problem. On the one hand, a large bandwidth will cause less signal distortion but bring more noise. On the other hand, a small bandwidth will cause more signal distortion but filter out more noise. Therefore, the optimal bandwidth of the LPF is determined by the SNR. In our experiment, we set the bandwidth of the LPF to $10 \times \Delta f = 3.125$ MHz.

Sampling Rate of ADC: On the transmitter side, the AP sends one QAM symbol on a single subcarrier in one OFDM symbol. Thus, the symbol rate at the IoT device is equal to the OFDM symbol rate at the AP. For a legacy AP, its sampling rate is 20 Msps and each OFDM symbol has 80 samples (64 samples in data part and 16 samples in cyclic prefix part). Hence, the QAM symbol rate at the IoT device is 250 kps (20 Msps divided by 80). Therefore, the required sampling rate of the ADC is 250 kps.

Phase Shift Compensation: The analytical results in Section III show that, by using an appropriate local carrier frequency, the IoT device can perfectly decode the signal on a subcarrier in the OFDM symbol from the AP. However, the analytical study in Section III did not consider the cyclic prefix (CP) in the OFDM symbols. In practice, the CP is attached at the beginning of each OFDM symbol, which is

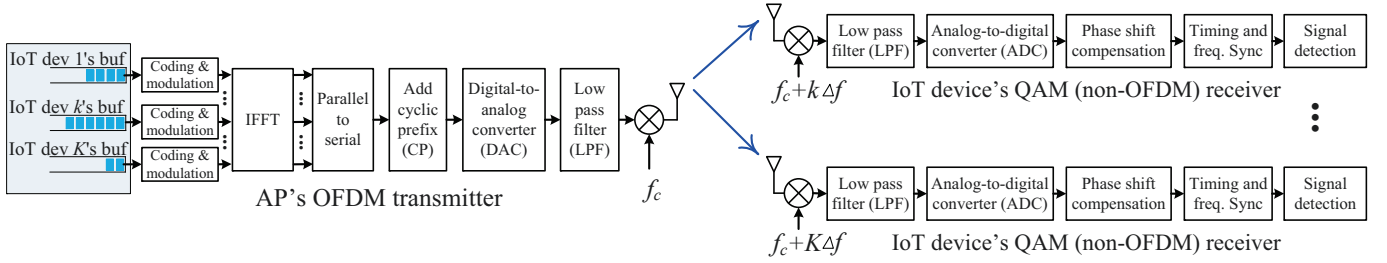


Fig. 4: Downlink PHY design for data transmission from a broadband AP to K narrowband IoT devices.

$16 \times 0.05 = 0.8 \mu\text{s}$ in 802.11 WLANs. As the time period of the signal on subcarrier k is $0.05 \times 64/k = 3.2/k \mu\text{s}$, the phase shift caused by the CP is $2\pi \times \frac{0.8}{3.2/k} = 2\pi \times \frac{k}{4}$ radians. To decode the signal at the IoT device, the phase shift caused by CP must be compensated for each OFDM symbol.

For the phase shift compensation module in Fig. 4, denote $y_{\text{in}}(n)$ as its input data symbol sequence; denote $y_{\text{out}}(n)$ as its output data symbol sequence. Then, the operation of this module can be written as:

$$y_{\text{out}}(n) = y_{\text{in}}(n) \cdot e^{j2\pi \frac{nk}{4}}, \quad (10)$$

where n is the time-domain symbol index, and k is the index of the subcarrier that is used for the IoT device. Note that the initial phase can be arbitrarily selected as it will be tackled by the signal detection module.

Timing and Frequency Synchronization: The purpose of timing synchronization is to search for the bursty signal frames in the received signal stream at the IoT device. To reduce the computational complexity, we propose a three-step strategy for timing synchronization, which combines energy detection (low complexity), auto-correlation of the preamble signal (coarse search), and cross-correlation of the preamble signal (fine search). Specifically, in step 1, we detect the energy of the received signal. If the detected energy is below a pre-defined threshold, then the search procedure stops with a false output; otherwise, we go to step 2. In step 2, we auto-correlate the received signal stream with a distance M_p , with the aim of identifying the two identical pieces of Zadoff-Chu signals in the preamble. If the auto-correlation value is smaller than a pre-defined threshold, the search procedure stops with a false output; otherwise, we identify a small search area and go to step 3. In step 3, we cross-correlate the received signal with a local copy of the preamble in the identified small search area. If the cross-correlation value is smaller than a pre-defined threshold, then the search procedure stops with a false output; otherwise, a signal frame is successfully found at the position with the maximum cross-correlation value.

Once a signal frame is found, we then conduct frequency synchronization. The purpose of frequency synchronization is to estimate the frequency offset and compensate for it. We take advantage of the two identical pieces of Zadoff-Chu signals in the preamble to estimate the frequency offset. Mathematically, the phase offset per symbol caused by the frequency offset,

which we denote as θ , can be written as:

$$\theta = \frac{1}{M_p} \cdot \angle \left(\sum_{n=P}^{P+M_p-1} y(n)y(n+M_p)^* \right), \quad (11)$$

where $y(n)$ is the received baseband signal stream, P is the beginning position of a frame (from timing synchronization), $(\cdot)^*$ is conjugate operator, $\angle(\cdot)$ is the angle of a complex number. After the frequency offset estimation, we can compensate for the frequency offset for the sampled signal. Note that QAM signal detection is not as susceptible to frequency offset as OFDM signal detection and, therefore, the frequency synchronization accuracy at IoT device is not that demanding as that in OFDM systems.

Signal Detection for Single-User Case: For the wireless channel between the AP and the IoT device, the delay spread is much less than the time duration of an OFDM symbol. Moreover, the bandwidth of the IoT device's LPF is much larger than the signal bandwidth (Δf). Therefore, the compound channel from the transmitter to the receiver can be modeled as a flat fading channel.

To detect the signals that have experienced flat fading channel, we take advantage of the distributed pilot signals (see Fig. 3). We can first use the pilot signals to estimate the channel coefficient and then use the calculated channel coefficient to equalize the channel for signal detection.

C. Downlink PHY Design: Multi-User Case

Number of IoT Devices (K): The previous design is focused on the case where the AP uses a single subcarrier to serve one IoT device. We now extend the downlink PHY design to the case where the AP uses K subcarriers to serve K IoT devices. Suppose that the OFDM modulation at the AP has 64 subcarriers and 48 of them are used for possible data transmission. Fig. 5 illustrates the transmit signal spectrum at the AP when $K = 24$.

Inter-Subcarrier Interference: For an IoT device, it intends to receive the signal on a particular subcarrier. The signals on the other subcarriers constitute inter-subcarrier interference for this IoT device. As shown in Fig. 5, the signals on all the subcarriers overlap in the spectral domain. How to manage the inter-subcarrier interference is a critical problem for an IoT device to decode its desired signal.

A natural approach to address the inter-subcarrier problem is by reducing the spectrum utilization. In other words, by

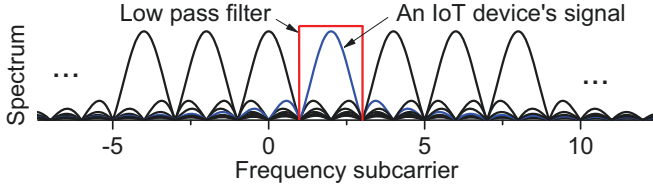


Fig. 5: Transmit signal spectrum at the AP when 24 subcarriers are used for downlink transmission.

decreasing the number of data-carrying subcarriers at the AP (i.e., K), the inter-subcarrier interference at the IoT devices will be alleviated. However, the decrease of K will result in a low spectrum efficiency. A trade-off between spectrum efficiency and each IoT device's throughput will be studied using experimental results in Section VIII.

Inter-Symbol Interference: For a given K , each IoT device uses a low pass filter with bandwidth $24\Delta f/K$ to suppress inter-subcarrier interference, as illustrated in Fig. 5. As the bandwidth of the filter is smaller than that of the desired signal, the use of such a filter will inevitably cause inter-symbol interference. Although there are many sophisticated techniques to combat inter-symbol interference (e.g., Viterbi sequence detector [2]), these techniques pursue the optimal performance and thus have a high computational complexity. As IoT devices are limited by their computational capability, those sophisticated techniques may not be suited for IoT devices. In light of this, we propose a low-complexity approach to combat inter-symbol interference by taking advantage of the Zadoff-Chu sequences in the preamble.

Signal Detection for Multi-User Case: Our signal detection approach consists of two steps: channel estimation and channel equalization. We first introduce the mathematical modeling and then present the proposed channel estimation and equalization methods.

Mathematical Modeling: The compound channel consists of over-the-air channel and the RF circuit response. While the delay spread of the over-the-air channel is very small compared to the time duration of an OFDM symbol, the delay spread of the LPF at the IoT devices is significant. Thus, we model the channel as a multi-tap channel with tap coefficients $[h_{-L}, \dots, h_{-1}, h_0, h_1, \dots, h_L]$.

With a bit abuse of notation, we denote $\{x(n)\}$ as the transmit signal on that subcarrier at the AP; and denote $\{y(n)\}$ as the received signal at the IoT device. Then, the transfer function can be written as:

$$y(n) = \sum_{l=-L}^L h_l \cdot x(n-l) + w(n), \quad (12)$$

where $w(n)$ is the combination of noise and residual inter-subcarrier interference.

Channel Estimation: To estimate the channel tap coefficients, we take advantage of the two identical Zadoff-Chu sequences in the frame preamble. Denote $[z(0), z(1), \dots, z(M_p)]$ as the Zadoff-Chu sequence in the preamble. Denote

$[z_i(0), z_i(1), \dots, z_i(M_p)]$ as a cyclically shifted version of this Zadoff-Chu sequence. That is,

$$z_i(n) = z((n-i)\%M_p), \quad 0 \leq n \leq M_p, \quad (13)$$

where $\%$ is modulus operator.

Denote $[y(0), y(1), \dots, y(2M_p-1)]$ as the received signal sequence in the frame preamble. With a local copy of the Zadoff-Chu sequence, the estimated channel tap coefficients, which we denote as $\hat{h}_l$, can be written as:

$$\hat{h}_l = \frac{1}{M_p} \sum_{n=0}^{M_p-1} y(n+l+M_p/2) \cdot z_{M_p/2}(n)^*, \quad -L \leq l \leq L. \quad (14)$$

For this channel estimation method, we have the following result: If the noise and the inter-subcarrier interference are negligible and $L \leq M_p/4$, then the channel tap coefficients can be perfectly estimated, i.e., $\hat{h}_l = h_l$ for $-L \leq l \leq L$. This result stems from the nice property that the auto-correlation of a Zadoff-Chu sequence with a cyclically shifted version of itself is zero.

Channel Equalization: After channel estimation, we then equalize the channel for signal detection. Note that, while there are many signal detection methods pursuing the optimal detection performance [2], the proposed equalization method aims to preserve the low computational complexity of IoT devices by leveraging the observed channel characteristics for approximation.

Based on (12), we have:

$$\begin{aligned} \hat{x}(n) &= \frac{1}{h_0} \left(y(n) - \sum_{l=-L \dots L, l \neq 0}^{l \neq 0} h_l \cdot \hat{x}(n-l) - w(n) \right) \\ &\stackrel{(a)}{\approx} \frac{1}{h_0} \left(y(n) - \sum_{l=-L \dots L, l \neq 0}^{l \neq 0} \frac{h_l}{h_0} \cdot y(n-l) - w(n) \right) \\ &\stackrel{(b)}{\approx} \frac{1}{h_0} \left(y(n) - \sum_{l=-L \dots L, l \neq 0}^{l \neq 0} \frac{h_l}{h_0} \cdot y(n-l) \right), \end{aligned} \quad (15)$$

where (a) follows from our observation that $|h_l| \ll |h_0|$ for $-L \leq l \leq L$ and $l \neq 0$, and (b) follows from that we ignore the noise.

After channel equalization, we then use the pilot signals in the frame to estimate the phase offset for each segment of data symbols and compensate for the phase offset correspondingly.

Computational Complexity: The proposed signal detection method has a linear computational complexity with the length of the frame. Specifically, its computational complexity is $O(LF)$, where F is the number of symbols in the frame.

V. PHY DESIGN FOR EE-IoT: UPLINK

In this section, we present our uplink PHY design for the data transmission from multiple (K) IoT devices to an AP. From the signal processing perspective, the uplink PHY design is a converse of the downlink PHY design. However, the challenge in the uplink PHY design is different from that in the downlink PHY design. In the uplink, if the QAM-modulated

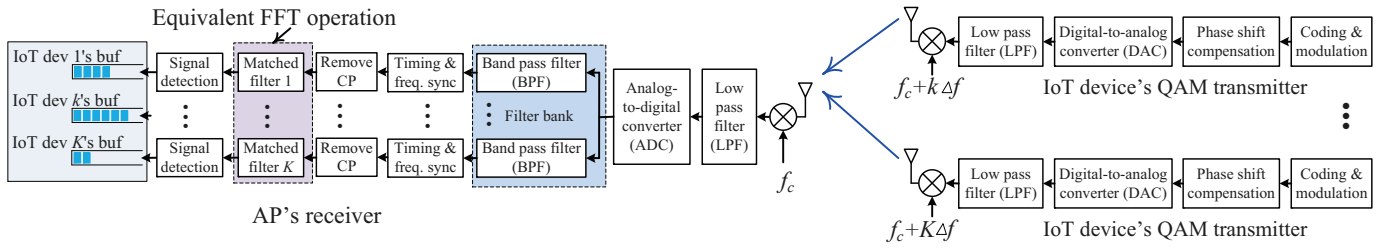


Fig. 6: Uplink PHY design for data transmission from K narrow-band IoT devices to a broadband AP.

signals from IoT devices are perfectly synchronized in both time and frequency domains, the AP can decode the QAM signals on its subcarriers as if the signals were from a single OFDM transmitter. Hence, the real challenge in the uplink PHY design is the timing and frequency synchronization among the signals from different IoT devices. In what follows, we present a PHY design to enable uplink data transmission between an AP and a set of asynchronous IoT devices.

A. Transmitter PHY for IoT devices

In the uplink data transmission, we use the same frame format as shown in Fig. 3. The two identical Zadoff-Chu sequences in the preamble will be used for synchronization and channel estimation, and the pilot signals will be used for phase offset correction. Fig. 6 depicts the uplink PHY design for the IoT devices, which we elaborate as follows.

Phase Shift Compensation: As we showed in the downlink PHY design, the CP in an OFDM symbol will introduce an extra delay for the signal on each subcarrier. This extra delay causes a phase shift for the signal on each subcarrier. For subcarrier k , the phase shift is $2\pi\frac{k}{4}$ radians, as we explained previously. Such a phase shift should be pre-compensated in order for the AP to decode the signal. Hence, the baseband signal processing module “Phase shift compensation” is designed for this purpose. Denote $x_{\text{in}}(n)$ and $x_{\text{out}}(n)$ as its input and output signal streams, respectively. Then, the function of this module can be written as:

$$x_{\text{out}}(n) = x_{\text{in}}(n) \cdot e^{-j2\pi\frac{nk}{4}}, \quad (16)$$

where n is the time-domain symbol index and k is the index of the subcarrier that is used for the IoT device.

Settling Time of DAC: While the sampling rate of the DAC in the IoT device is 250 kps, there is a requirement for the settling time of the DAC to maintain the rectangular shape of its output waveform. In this design, the settling time of the DAC should be less than the time duration of the CP, which is $0.8 \mu\text{s}$. Note that such a requirement is very mild and can be met by many low-end DACs on the market.

Low Pass Filter: Similar to the LPF design in the downlink, the LPF design in the uplink is also a trade-off between signal distortion and noise suppression. In the uplink, the IoT device serves as a transmitter, where the noise is less significant compared to a receiver. Therefore, we set the bandwidth of the LPF to $15 \times \Delta f \approx 5 \text{ MHz}$.

B. Receiver PHY for AP

As the IoT devices are driven by independent clock sources, their transmit signals are asynchronous when arriving at the AP. How to address the synchronization problem is the challenging task in the design of AP's receiver PHY. It is worth pointing out that the synchronization problem here is different from that in multi-user MIMO in OFDM communications. This is because the IoT devices only have low-complexity transceivers that work at low clock rate (250 kHz). Sophisticated MAC protocols (e.g., Timing Advance [3]) cannot be applied to IoT devices to achieve timing and frequency synchronization on the transmitter side. Hence, the synchronization challenge has to be tackled at the PHY layer on the receiver side (on AP side).

To address the synchronization problem, we borrow the idea of filter bank from the SC-FDMA uplink in LTE networks, where a LTE base station decodes signals from multiple asynchronous user equipments (UEs) [4]. Specifically, as shown in Fig. 6, the AP first uses a bank of bandpass filters to separate the signals from different IoT devices. With the bandpass filters, the AP can estimate and correct the synchronization errors independently for the signal from each IoT device. A shortcoming of this method is that perfect signal separation is not possible even with ideal brick-wall bandpass filters due to the frequency leakage among the adjacent subcarriers. A natural approach to addressing this shortcoming is to decrease the number of data-carrying subcarriers (i.e., K). As illustrated in Fig. 5, decreasing K can significantly reduce the inter-subcarrier interference. The impact of K on the performance of each IoT device in the uplink will be investigated using experimental results in Section VIII. In what follows, we outline the key modules in each of the AP's signal paths.

Bandpass Filter: The number of bandpass filters that are used at the AP is equal to the number of IoT devices (i.e., K). For each bandpass filter, we set its normalized center frequency to $k/32$ and set its normalized bandwidth to $K/48$, where k is the index of the subcarrier used by the target IoT device. Note that we use the normalized frequency and bandwidth because the filters are applied in the digital domain at the AP.

Timing and Frequency Synchronization: The timing and frequency synchronization will be done in the same way as that in the downlink. Specifically, for timing synchronization, we exploit the auto-correlation property of the two identical Zadoff-Chu sequences in the preamble for coarse timing syn-

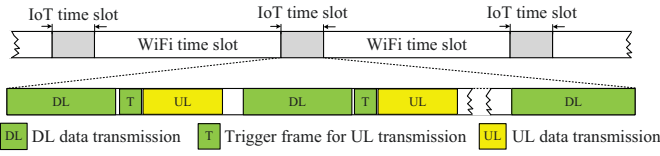


Fig. 7: Resource allocation in the IoT communication protocol.

chronization, and then exploit their cross-correlation property for fine timing synchronization. For frequency synchronization, we autocorrelate the two identical Zadoff-Chu sequences in the preamble to estimate the carrier frequency offset, and then compensate for the frequency offset in the digital domain.

Matched Filters (FFT): The bank of matched filters is equivalent to FFT operation at the AP. Specifically, the matched filter on path k can be written as:

$$Y_k = \sum_{n=0}^{N-1} y_k(n) e^{-j \frac{2\pi}{N} nk}, \quad (17)$$

where Y_k is the output data of the matched filter and $y_k(n)$, $0 \leq n \leq N-1$, is the input data sequence of the matched filter. It is easy to see that the matched filters are actually the same as FFT operation in math. The only difference is that different matched filters have different input data sequences for signal isolation.

Signal Detection: The signal processing module in the uplink is similar to its counterpart in the downlink. The main purpose of this module is to cancel inter-symbol interference and equalize the channel for signal recovery. The signal detection method that was proposed for the downlink multi-user case can be directly used here for the uplink signal detection. Furthermore, since the AP is not limited by power consumption and computational capability, more advanced signal detection methods (e.g., soft-decision Viterbi decoder [2]) can be employed to improve the detection performance.

VI. MAC PROTOCOL DESIGN FOR EE-IoT

In this section, we outline our proposed MAC protocol for the communications between the AP and IoT devices in the WLAN as shown in Fig. 1.

Protocol Overview: As shown in Fig. 1, the AP needs to serve both standard WiFi and IoT devices. To do so, we propose a time division multiplexing scheme as shown in Fig. 7. The AP periodically reserves a time slot for the communications between itself and the IoT devices. During the WiFi time slot, the IoT devices can switch to sleep mode to reduce their power consumption. During the IoT time slot, the AP can silence the standard WiFi devices by broadcasting a network allocation vector (NAV) packet. The duration of an IoT time slot can be either fixed or adaptively set, depending on the system requirement.

Channel Assignment: In an IoT time slot, our PHY design can support K (e.g., $K = 16$ or $K = 24$) parallel independent channels for uplink and downlink data transmissions between AP and IoT devices. For a new or wake-up IoT device, it first

listens to each of the K channels and selects the one with least traffic as its initial channel. After the selection, it will stick to this channel unless the AP assigns it to another channel. On the AP side, it maintains a list of active IoT devices. With the global information, it can perform an optimization procedure to adjust the channel assignment so as to improve the channel efficiency.

Downlink Transmission: The proposed MAC protocol is a semi-centralized protocol, where the AP is the controller for resource allocation. As such, it has full degree of freedom for downlink transmission scheduling on the K channels. In the downlink channels, the AP can periodically broadcast beacon frames that contain all the information about the network. The AP can also inform the IoT devices of its decision for channel re-assignment.

Uplink Transmission: As there are K parallel channels that can be used for uplink transmission, it is important to coordinate the IoT devices on those channels for uplink transmission. Thus, we have designed a special frame (called trigger frame for uplink transmission) for this purpose, as shown in Fig. 7. Specifically, an IoT device keeps listening its channel for downlink data transmission; it performs possible uplink data transmission only if it receives a *trigger frame* from the AP. For each individual channel, CSMA/CA is used to control the channel access among the IoT devices on this channel.

Uplink Power Control: A power control mechanism has been implemented for uplink data transmission. For each IoT device, it estimates the signal strength of the downlink trigger frame, based on which it adjusts its transmit power for uplink data transmission. By doing so, the AP will receive relatively similar signal power from the IoT devices on different channels. This mechanism improves the performance of AP's signal detection.

VII. IMPLEMENTATION

To evaluate the practicality and performance of EE-IoT in real-world wireless environments, we have prototyped the proposed PHY design and MAC protocol on a wireless testbed that consists of USRP2 and GNU-Radio software package.

Frame Parameters: The frame format in Fig. 3 is used for data transmission in both uplink and downlink, with $M_p = 12$, $M_s = 50$ and $M_d = 50$. The total number of symbols in a frame is set to 1044. On the AP side, one symbol actually refers to one OFDM symbol in legacy 802.11 standard. Specifically, each OFDM symbol has 64 subcarriers, where 48 of them may be used for data transmission. The length of CP is 16 samples. The length of one OFDM symbol is 80 samples and its time duration is 4 μ s.

Prototype of AP: We have built an AP using a USRP2 device and a laptop. We have implemented the proposed AP's PHY in Figures 4 and 6 in GNU-Radio on the laptop, which will control the USRP2 device to work in the way as designed. The maximum transmit power of the AP is set to 20 dBm. The sampling rate is set to 20 Msps.

Prototype of IoT devices: We have built three IoT devices using three independent USRP2 devices and laptops. We have

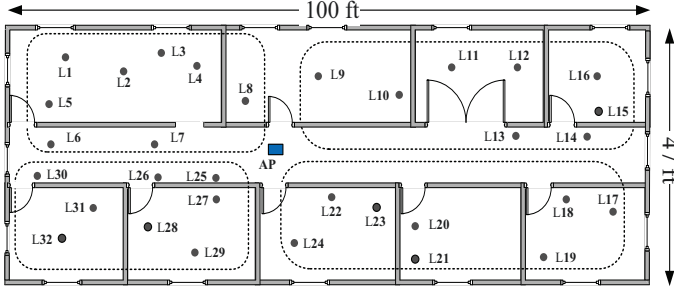


Fig. 8: The floor plan for EE-IoT evaluation.

implemented their PHY in Figures 4 and 6. The maximum transmit power of the AP is set to 0 dBm. While the symbol rate is 250 kHz, we use $4\times$ oversampling rate and therefore set the sampling rate to 1 Msps.

We also implemented a special device using USRP2 that can mimic up to 21 IoT devices. This device is used only for test purpose to emulate the inter-subcarrier interference in the uplink. Its performance will not be measured.

Prototype of MAC Protocol: We have implemented simplified version of the proposed MAC protocol, including downlink and uplink data transmission as well as uplink power control, with a fixed set of IoT devices. Time-sharing with standard WiFi devices, channel assignment at the AP, and uplink CSMA/CA among the IoT devices are not considered in our implementation.

VIII. PERFORMANCE EVALUATION

In this section, we evaluate the performance of EE-IoT.

A. Experimental Setup and Performance Metrics

Experimental Setup: We measure the performance of EE-IoT in an office building as shown in Fig. 8. The AP is placed at the spot marked “AP”. The four IoT devices (three IoT devices and one special device to mimic multiple IoT devices in the uplink) are placed at 4 out of the 32 locations. Particularly, these four IoT devices are always placed in four different areas marked by dashed boxes. The purpose of this setting is to more authentically emulate the real network scenarios.

Performance Metrics: We use two performance metrics to assess the performance of EE-IoT. The first one is error vector magnitude (EVM), which is widely used in WiFi device tests. EVM quantifies the normalized error magnitude between the measured constellation and the ideal constellation. Mathematically, it can be written as: $\text{EVM (dB)} = 10 \log_{10} \left(\frac{\mathbb{E}(|x - \hat{x}|^2)}{\mathbb{E}(|x|^2)} \right)$, where x is the original signal at the transmitter and $\hat{x}$ is the estimated signal at the receiver.

The second performance metric that we use is an IoT device’s data rate. Different from EVM, which will be directly measured from the experimental results, the data rate will be estimated based on the modulation and coding scheme (MCS) table specified in 802.11 standard as shown in Table I. Specifically, for an IoT device, its uplink and downlink data rate is estimated by: $r = \frac{1}{2} \times 250 \times \gamma(\text{EVM})$ kbps, where $\frac{1}{2}$

TABLE I: EVM specification in IEEE 802.11ac standards [5].

EVM (dB)	(inf -5)	[-5 -10]	[-10 -13]	[-13 -16]	[-16 -19]	[-19 -22]	[-22 -25]	[-25 -27]	[-27 -30]	[-30 -32]	[-32 -inf]
Modulation	N/A	BPSK	QPSK	QPSK	16QAM	16QAM	64QAM	64QAM	64QAM	256QAM	256QAM
Coding rate	N/A	1/2	1/2	3/4	1/2	3/4	2/3	3/4	5/6	3/4	5/6
$\gamma(\text{EVM})$	0	0.5	1	1.5	2	3	4	4.5	5	6	20/3

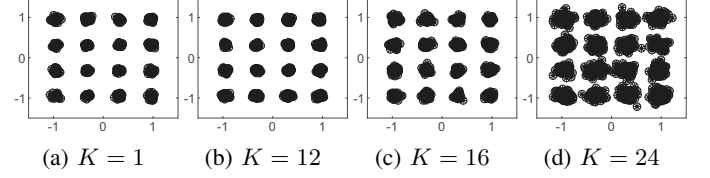


Fig. 9: The constellation of the decoded signal in the downlink.

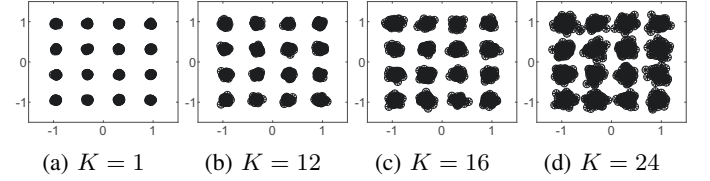


Fig. 10: The constellation of the decoded signal in the uplink.

means that one half time for downlink transmission and the other half for uplink transmission, 250 is the symbol rate (in kbps), $\gamma(\text{EVM})$ is the average number of bits carried by one symbol and its values are given in Table I.

B. A Case Study

We use a case study to show the details of downlink and uplink data transmission for the IoT device placed at Location 1 in Fig. 8.

Downlink: Fig. 9 shows the constellation of the decoded signals at the IoT device when 16QAM is used. It is evident to see that the 16QAM can be successfully demodulated by this IoT device. Specifically, the measured EVM is -27.0 dB when $K = 1$, -26.9 dB when $K = 12$, -23.5 dB when $K = 16$, and -18.0 dB when $K = 24$.

Uplink: Fig. 10 shows the constellation of the decoded signals (from the IoT device at Location 1) at the AP when 16QAM is used. We can see that 16QAM can be successfully demodulated at the AP. Specifically, the measured EVM is -32.1 dB when $K = 1$, -25.4 dB when $K = 12$, -21.6 dB when $K = 16$, and -17.6 dB when $K = 24$.

C. Complete Experimental Results

We now present the measured experimental results for one IoT device when it is placed at each of the 32 locations.

Downlink: Fig. 11 shows the data rate of the IoT device in the downlink. The results show that the achievable downlink data rate for this IoT device is greater than 500 kbps when $K \leq 12$, greater than 375 kbps when $K = 16$, and greater than 187 kbps when $K = 24$.

Uplink: Fig. 12 shows the data rate for the target IoT device in the uplink. The results show that the achievable uplink data rate for this IoT device is greater than 500 kbps when $K = 1$,

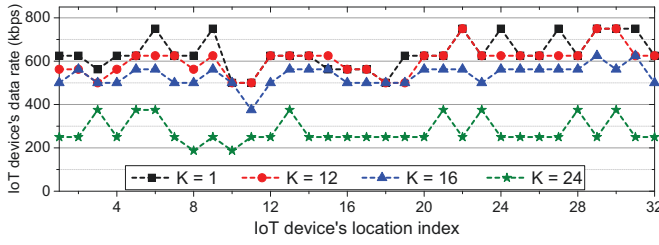


Fig. 11: Measured data rate for one IoT device in the downlink.

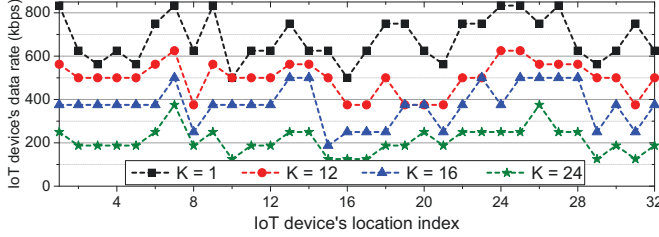


Fig. 12: Measured data rate for one IoT device in the uplink.

greater than 375 kbps when $K = 12$, greater than 187.5 kbps when $K = 12$, and greater than 125 kbps when $K = 24$.

IX. RELATED WORK

NB-IoT in Cellular Networks: As our society evolves to smart era, NB-IoT has attracted tremendous research efforts in both industry and academia. While there are many research results of NB-IoT (see, e.g., [1], [6]–[8]), most of them are limited to cellular networks. Our work focuses on IoT communications in WiFi networks, and thus differs from NB-IoT essentially.

NarrowBand WiFi Communications: Recently, there are some pioneering research efforts from the industry to explore the feasibility of narrowband WiFi communications. In [9], Bluetooth Low Energy (BLE) was studied in 802.11ax WLANs to support IoT applications. In [10], an overlay narrow-band IoT communication approach was studied in 802.11ax WLANs. In [11], narrow-band IoT communications in WiFi networks were studied and evaluated using simulation from MAC layer protocol perspective. However, these results remain in conceptual discussion and theoretical exploration without considering practical issues in real implementation. Our work differs from these efforts significantly.

Cross-Technology Communications: Another research line in relevance to this work is WiFi and ZigBee cross-technology communications [12], [13]. However, these efforts aim to enable cross communications between different types of wireless devices without hardware modification. Furthermore, the existing results can enable only one-way communication (from WiFi transmitter to ZigBee receiver). This work differs from this research line fundamentally.

X. CONCLUSION

In this paper, we proposed EE-IoT, an energy-efficient IoT communication scheme for WLANs. Compared to its

counterpart (NB-IoT), EE-IoT takes advantage of the widely existing WiFi infrastructure to provide wireless Internet access for IoT devices, and thus will not incur additional service fee to the end users. The key component of EE-IoT is an asymmetric PHY design, which enables an OFDM-based broadband AP to communicate with multiple (non-OFDM) narrowband IoT devices. In this asymmetric PHY, instead of using the same carrier frequency as the AP, an IoT device aligns its carrier frequency to a particular subcarrier of the AP's OFDM signals. Such a carrier frequency setting makes it possible for the IoT device to transmit/receive signal on a single subcarrier at a low sampling rate (250 kbps). We have evaluated the performance of EE-IoT in an office building environment. Experimental results show that an AP can serve 24 IoT devices simultaneously and each IoT device can achieve more than 187 kbps in the downlink and more than 125 kbps in the uplink.

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for their valuable comments and feedback. This project was partially supported by the NSF under Grant CNS-1717840.

REFERENCES

- [1] 3GPP TR 45.820, "Cellular system support for ultra low complexity and low throughput Internet of Things," V2.1.0, August 2015.
- [2] T. S. Rappaport, *Wireless communications: Principles and practice*, vol. 2. Prentice Hall PTR New Jersey, 1996.
- [3] J.-J. Van de Beek, P. O. Borjesson, M.-L. Boucheret, D. Landstrom, J. M. Arenas, P. Odling, C. Ostberg, M. Wahlqvist, and S. K. Wilson, "A time and frequency synchronization scheme for multiuser OFDM," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 11, pp. 1900–1914, 1999.
- [4] M. Morelli, C.-C. J. Kuo, and M.-O. Pun, "Synchronization techniques for orthogonal frequency division multiple access (OFDMA): A tutorial review," *Proceedings of the IEEE*, vol. 95, no. 7, pp. 1394–1427, 2007.
- [5] IEEE 802.11ac, "IEEE standard for information technology local and metropolitan area networks part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput," *IEEE Standards 802.11ac*, 2014.
- [6] R. Ratasuk, N. Mangalvedhe, Y. Zhang, M. Robert, and J.-P. Koskinen, "Overview of narrowband IoT in LTE Rel-13," in *IEEE Conference on Standards for Communications and Networking (CSCN)*, pp. 1–7, 2016.
- [7] A. Hoglund, X. Lin, O. Liberg, A. Behravan, E. A. Yavuz, M. Van Der Zee, Y. Sui, T. Tirronen, A. Ratilainen, and D. Eriksson, "Overview of 3GPP release 14 enhanced NB-IoT," *IEEE Network*, vol. 31, no. 6, pp. 16–22, 2017.
- [8] S.-M. Oh and J. Shin, "An efficient small data transmission scheme in the 3GPP NB-IoT system," *IEEE Communications Letters*, vol. 21, no. 3, pp. 660–663, 2017.
- [9] L. R. Wilhelmsson, M. M. Lopez, and D. Sundman, "NB-WiFi: IEEE 802.11 and Bluetooth low energy combined for efficient support of IoT," in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2017.
- [10] N. Butt, R. Di Taranto, D. Sundman, and L. Wilhelmsson, "On the feasibility to overlay a narrowband IoT signal in IEEE 802.11," in *IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–7, 2017.
- [11] Y. Wang, L. F. Del Carpio, D. Sundman, D. Peddireddy, and A. Larmo, "MAC layer design and evaluation of a narrowband Wi-Fi system," in *IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–6, 2017.
- [12] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *Proceedings of ACM MobiCom*, pp. 2–14, 2017.
- [13] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *Proceedings of ACM MobiCom*, pp. 317–330, 2015.