Bringing Internet Connectivity to Heterogeneous IoT Devices via Artificial Doppler Effects

***Abstract*—Recent years have seen major innovations in developing energy-efficient wireless technologies such as Bluetooth Low Energy (BLE) for Internet of Things (IoT). Despite demonstrating significant benefits in providing low power transmission and massive connectivity, hardly any of these technologies have made it to directly connect to the Internet. Recent advances demonstrate the viability of direct communication among heterogeneous IoT devices with incompatible physical (PHY) layers. These techniques, however, require modifications in transmission power or time, which may affect the media access control (MAC) layer behaviors in legacy networks. In this paper, we argue that the frequency domain can serve as a free side channel with minimal interruptions to legacy networks. To this end, we propose DopplerFi, a communication framework that enables a two-way communication channel between BLE and Wi-Fi by injecting artificial Doppler shifts, which can be decoded by sensing the patterns in the Gaussian frequency shift keying (GFSK) demodulator and Channel State Information (CSI). The artificial Doppler shifts can be compensated by the inherent frequency synchronization module and thus have negligible impacts on legacy communications. Our evaluation using commercial off-the-shelf (COTS) BLE chips and 802.11-compliant testbeds have demonstrated that DopplerFi can achieve throughput up to 6.5 Kbps at the cost of merely less than 0.8% throughput loss.**

I. Introduction

The wide deployments of Wi-Fi and cellular network infrastructures have provided ubiquitous and transparent access to the Internet for today's laptops, tablets, and smartphones. The coming wave of Internet of Things (IoT), however, does not enjoy the same level of ubiquity in Internet access. This is because these tiny, low-end IoT devices tend to adopt low-power wireless technologies instead of power-intensive Wi-Fi or cellular technologies [1]. As a mature technology, Bluetooth Low Energy (BLE) is widely adopted in today's wearables and smart objects, and is envisioned to continue its dominance in the market in near future [2]. These BLE-based IoT devices connect to the Internet through a gateway, which can be a smartphone or an access point equipped with both BLE and Wi-Fi/Ethernet interfaces. Although a gateway can bring Internet connectivity to IoT devices using protocols that are incompatible to Wi-Fi, it induces a large amount of traffic overhead and requires additional gateway deployments, which are not scalable to massive IoT connectivity [3].

Recent research efforts have been devoted to enabling direct communications between heterogeneous IoT protocols and Wi-Fi. These pioneering designs enable cross-technology communications by embedding bits into transmission power levels [3], [4] or packet transmission time shifts/patterns [5]–[8]. Despite that these innovations are transparent to legacy links in that they do not modify frame format or introduce extra packets, they still affect the media access control (MAC) layer behaviors in existing networks. Transmission power modifications induce changes in interference and communication ranges. Perturbing packet transmission time, though merely inducing negligible latency at the application layer, may disturb contention behaviors.

Instead of relying on the amplitude and time dimensions that easily affect MAC behaviors, we argue that a more controllable and nonintrusive way is to exploit the frequency domain. Our observation is that today's communication technologies are robust to carrier frequency perturbations. Such a capability lies in the need to combat against carrier frequency offset (CFO) caused by the Doppler effect and hardware impairments. The current designs of BLE and Wi-Fi can tolerate and compensate up to 150 KHz CFO, which is much higher than the inherent CFO. It leaves enough redundancy to encode bits in the carrier frequency. In addition, such an amount of CFO has little impact on adjacent channels due to the guard band protection.

In this paper, we propose `DopplerFi`, which aims to enable two-way cross-technology communications between Wi-Fi and BLE by subtly shifting the sender's carrier frequency. DopplerFi is inspired by the Doppler effects caused by movements in legacy networks. It intentionally injects artificial Doppler shifts in legacy packets. These artificial Doppler shifts hide themselves in inherent carrier frequency perturbations and are transparent to MAC and upper layers. Artificial Doppler shifts are injected into the transmitted packets, which can be easily realized by the carrier frequency calibration capability of BLE and Wi-Fi radios. Injecting a controllable amount of frequency shift induces minimal impact on existing transmissions as legacy PHYs have already been designed to eliminate the impact of CFO.

Translating the above idea into a practical system, however, entails a variety of challenges. Although BLE and Wi-Fi receivers are able to detect CFO from PHY-compatible packets, there is no PHY module that can directly identify frequency shifts from incompatible packets. The first hurdle comes from the fact that BLE adopts much narrower channel compared to today's Wi-Fi. BLE uses 1 MHz or 2 MHz channels while today's OFDM-based Wi-Fi (e.g., IEEE 802.11ac/n/g/a) channels are 20 MHz wide. Even if Wi-Fi packets are shifted by different amounts of frequency (in an order of KHz), one overlapped BLE channel would still be overwhelmingly filled by Wi-Fi signals in the frequency domain. Thus, BLE cannot detect the frequency shifts in Wi-Fi packets by simply looking at the frequency domain. To overcome this hurdle, our fundamental insight is that all Wi-Fi packets prepend the

same preamble whose frequency patterns can be recognized at a granularity of one subcarrier. The bandwidth of one subcarrier in Wi-Fi is 312.5 KHz, and thus the shifts in such a bandwidth can be captured by BLE radios. To extract the preamble frequency patterns using standard BLE chips, we cannot directly analyze the frequency domain as there is no module in BLE such as fast Fourier transform (FFT) that can obtain the frequency domain signals. Our observation is that the preamble frequency patterns can be reflected in the output of the standard Gaussian frequency shift keying (GFSK) demodulator in BLE. As such, we extract the output bits of GFSK demodulator from BLE's PHY, and then decode the frequency shifts injected into the Wi-Fi packets.

Another challenge stems from decoding BLE frequency shifts using the standard PHY of Wi-Fi receivers. Although there is a module in Wi-Fi reception pipeline to estimate and compensate CFO, it requires a Wi-Fi preamble which is not possessed by a standard BLE packet. Instead, we extract frequency shifts in BLE packets by analyzing the channel state information (CSI), which can be extracted from commercial off-the-shelf (COTS) Wi-Fi cards. Since one BLE channel overlaps with multiple adjacent subcarriers in Wi-Fi, different amounts of frequency shifts in BLE packets can be differentiated by analyzing the frequency correlations among adjacent CSI values.

We implement DopplerFi on TI CC2400 BLE devices [9], and WARP [10]. Evaluation results of validated DopplerFi in creating a reliable two-way free side channel between BLE and Wi-Fi under a wide range of scenarios. DopplerFi achieves throughput up to 6.5 Kbps in an interference-free environment and 1.59 Kbps in a crowded environment with 20+ Wi-Fi access points (APs). On the other hand, DopplerFi induces merely 0.8% and 0.3% throughput loss on legacy Wi-Fi and BLE links, respectively.

The contributions of this paper are summarized as follows.

- We provide a comprehensive study toward creating a free side channel between heterogeneous IoT devices and Wi-Fi devices in the frequency. The frequency domain side channel imposes minimal impact on MAC behaviors in legacy networks.
- Our solution requires no hardware or PHY changes and decodes cross-technology bits by extracting patterns from the inherent GFSK and CSI readings that are readily available in the standard PHYs.
- We present BLE and Wi-Fi prototype implementations on TI CC2400 BLE devices and WARP, and validate the performance under various environments.

The remainder of this paper is structured as follows. We begin in Section II with an introduction of carrier frequency shifts and our motivation. We elaborate the detailed system design in Section III, followed by our system implementation and evaluation in Section IV. Related work is reviewed in Section V. Section VI concludes this work.

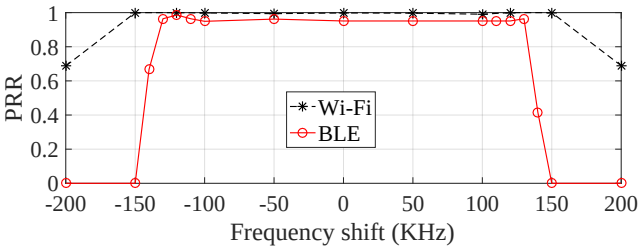


Fig. 1. Impact of frequency shift on legacy transmissions.

II. Exploiting Carrier Frequency Shifts

In this section, we explore the feasibility of creating a side channel between Wi-Fi and BLE in the frequency domain. Through this exploration, we observe that the frequency-domain side channel is a practical and suitable design in that it elegantly integrates with legacy networks with negligible impact.

In a typical wireless communication system, signals are unconverted to a higher frequency carrier before transmission. Receivers are expected to tune their radios to the exact same carrier frequency for reception. However, due to the impairments of the transmitter and the receiver oscillators, there is a frequency offset, i.e., CFO, between the carrier frequencies of a sender and a receiver. The CFO is normally within a limited range of 500 Hz [11]. Additionally, the movements of the sender or receiver result in Doppler effect, which changes the signal frequency captured at the receiver.

A standard OFDM-based Wi-Fi receiver estimates and compensates CFO using training symbols in the preamble. It computes the phase shift in the received long training symbols to obtain the corresponding CFO, and then compensate the offset to eliminate its impact on the following data symbols. For a 20 MHz Wi-Fi link, the maximum recoverable CFO is 156.25 KHz.

Departure from Wi-Fi, BLE employs frequency hopping techniques with the carrier modulation using GFSK. BLE is much more robust to CFO errors. The maximum tolerable frequency offset error is 150 KHz for a 2 MHz BLE link [12]. Some BLE chips incorporate the feature of crystal drift compensation to match the central frequency of the transmitted signal. With such a feature, the tolerable CFO can be relatex to around 250 KHz [9].

Therefore, we observe that there exists a sufficient amount of redundancy that allows legacy radios to freely modulate their carrier frequencies. To validate the feasibility of embedding bits into frequency offsets, we conduct an experiment using WARP as Wi-Fi radios and TI CC2400 [9] as BLE radios to study the impact of frequency shift injection on legacy transmissions. Two WARP nodes is configured to transmit and receive 96-Byte packets at various data rates conforming to IEEE 802.11 PHY/MAC, while CC2400 nodes adopt the Bluetooth Core Specification Version 4.0 [12]. We gradually tune the carrier frequency of senders and measure the packet reception ratio (PRR), which is defined to be the ratio of the number of correctly decoded packets to the total number of

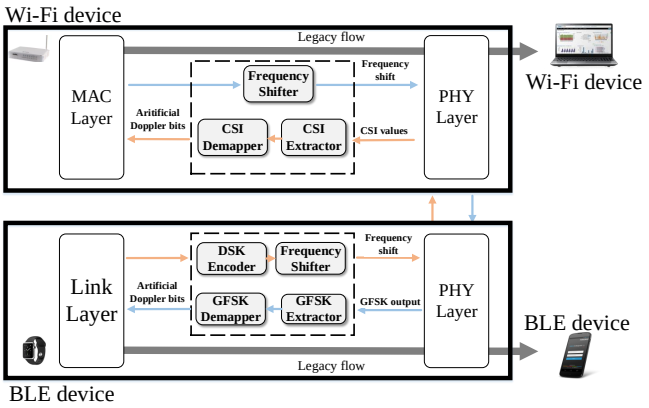


Fig. 2. Overview of DopplerFi.

transmitted packets. Fig. 1 sketches the results, which clearly show that tuning the carrier frequency within a proper range (-150 KHz - 150 KHz for BLE and -130 KHz - 130 KHz for Wi-Fi) has negligibly small impacts on legacy transmissions. Additionally, we also test the interference to adjacent channels and observe that the frequency shift within 200 KHz does not affect the performance of other links due to the 2 MHz guard band or channel interval in both Wi-Fi and BLE. These observations imply that with careful design, we can perturb the carrier frequency to create a free channel between Wi-Fi and BLE.

III. DopplerFi Design

This section describes an overview of DopplerFi, followed by design specifics in each component. Since Bluetooth will continue its dominance in the smart IoT market [2], we use the detailed design of cross-technology communications between BLE and Wi-Fi to demonstrate the idea of DopplerFi.

A. Design Overview

DopplerFi facilitates BLE-enabled IoT devices to directly communicate with Wi-Fi devices without strings. Simply put, DopplerFi runs a lightweight signal processing block sitting between PHY and MAC. It can be supported by existing Wi-Fi and BLE radios and protocols. DopplerFi does not modify the MAC-related configurations such as transmission power or time, and thus is transparent to upper layers.

Architecturally, DopplerFi is similar to a single frame-level control function such as rate adaptation or power control, in that it merely tunes one parameter, i.e., carrier frequency, in PHY configurations. Departure from rate adaptation and power control functions, DopplerFi works in a cross-layer fashion to pass messages between PHY and upper layers. In the sender mode, DopplerFi tunes the carrier frequency by CFO calibration in PHY. In the receiver mode, DopplerFi extracts CSI values or GFSK outputs from PHY interfaces.

Fig. 2 gives an overview of the DopplerFi architecture in Wi-Fi and BLE transceivers. DopplerFi extends legacy Wi-Fi and BLE by adding the following components.

- In the BLE transceiver, DopplerFi contains four modules: the *DSK (Doppler Shift Keying) encoder* and the *frequency shifter* that work in the sender mode, and the *GFSK extractor* and the *GFSK demapper* that work in the receiver mode. The DSK encoder reads the frequency hopping pseudo random sequence in BLE and determines when to embed bits into carrier frequency. The frequency shifter converts bits into carrier frequency shifts. As such, DopplerFi modulates side-channel bits into the carrier frequency of transmitted BLE packets. The GFSK extractor extracts output bits from the inherent GFSK demodulator in the BLE PHY. The GFSK demapper demodulates frequency shifts based on the patterns in the GFSK output bit sequence.
- In the Wi-Fi transceiver, DopplerFi contains three modules: the *frequency shifter* that work in the sender mode, and the *CSI extractor* and the *CSI demapper* that work in the receiver mode. Analogous to the BLE transceiver, the Wi-Fi transceiver subtly varies its carrier frequency according to the frequency shifter. In contrast to GFSK-based demodulation in BLE, the Wi-Fi transceiver extracts embedded side-channel bits by extracting CSI using the CSI extractor. The CSI values are fed to the CSI demapper to demodulate the embedded bits by analyzing the variance in the CSI readings caused by BLE frequency shifts.

The components in DopplerFi enable bidirectional communications between Wi-Fi and BLE while retaining ongoing legacy BLE and Wi-Fi transmissions.

B. Embedding Artificial Doppler Shifts

Wi-Fi sends packets in fixed channels with pre-defined carrier frequencies. Although BLE utilizes frequency-hopping spread spectrum (FHSS) technology to change the transmission channel between adjacent packets, the carrier frequency of each channel is fixed. DopplerFi establishes a cross-technology channel by embedding symbols within these carrier frequencies. To shift carrier frequency without affect Wi-Fi and BLE legacy transmissions, the following requirements should be satisfied. First, the amount of frequency shift should not exceed the limit of recoverable CFO. Additionally, the injected frequency shifts should be incorporated with the frequency hopping in BLE without any modification to the FHSS mechanism. We refer the frequency shift satisfying the above requirements as *artificial Doppler shift*, implying that it has minimal impact on existing systems just like Doppler shifts caused by normal movements.

To meet the above requirements, DopplerFi carefully selects different amounts of frequency shifts for BLE and Wi-Fi respectively, and employs a special companion referred to as the DSK encoder to modulate BLE carrier frequencies in conjunction with FHSS.

In standard Wi-Fi PHY, the build-in CFO compensation module can correct frequency offset up to 156.25 KHz, which is half of the subcarrier interval. To deliver as many side channel bits as possible in each packet, one might use multiple levels of frequency shifts for modulation. Unfortunately, since BLE adopts 1 MHz or 2 MHz bandwidth while Wi-Fi adopts

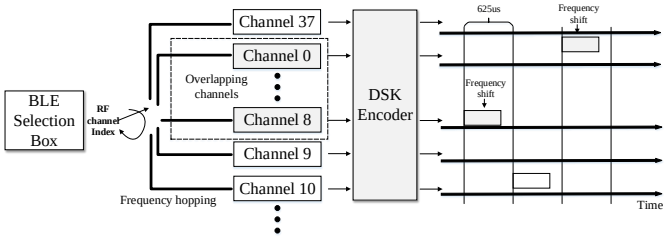


Fig. 3. DSK encoder.

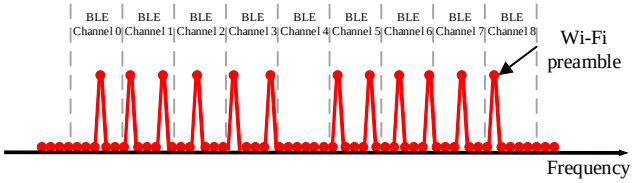


Fig. 4. Different patterns of Wi-Fi STF received by BLE radios in different overlapping channels.

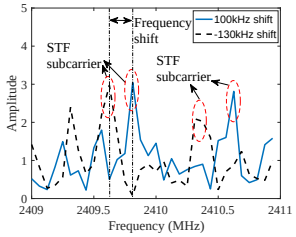


Fig. 5. Wi-Fi preamble data captured by a BLE radio. The BLE radio stays in BLE's Channel 3 to receive Wi-Fi packets by a node in Wi-Fi's Channel 1 in the 2.4 GHz band.

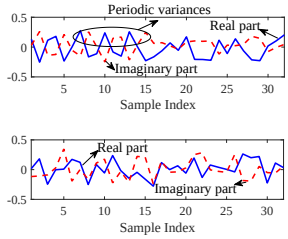


Fig. 6. I/Q data of Wi-Fi packets obtained by a BLE radio. Carrier frequency of Wi-Fi packets are shifted by -130 KHz and 100 KHz in the figures from top to bottom.

20 MHz bandwidth, BLE cannot directly measure Wi-Fi's frequency shift as a BLE channel is always overwhelmingly fulfilled by Wi-Fi signals regardless of Wi-Fi's frequency shifts. In addition, BLE is a single-carrier system that employs a completely different PHY compared to the Wi-Fi PHY. BLE and Wi-Fi cannot directly measure each other's frequency offset using CFO estimators. Considering all these difficulties, we select one level of frequency shift at the opposite directions to enable 1 bit in one packet. Particularly, BLE packets are shifted with $\pm 100(130)$ KHz (asymmetric shifts are adjusted according to the overlapping channel) while Wi-Fi packets are shifted with ± 80 KHz, which ensures enough space for artificial Doppler shift demodulation while ensuring CFO recovery in legacy packet reception even in the presence of inherent CFO and Doppler effect.

The FCC rules enforce BLE to conduct FHSS by transmitting packets in each of the 40 channels with equal probability. Since the aggregated bandwidth of BLE is several times wider than Wi-Fi channels, it is with a certain probability that a BLE packet jumps outside a Wi-Fi channel. To tackle this issue, DopplerFi employs the DSK encoder to guarantee bits are continuously modulated on the overlapped frequencies. Before packet transmission, a selection box in BLE PHY is employed to select the transmission channel according to a pseudo random sequence. The DSK encoder reads channel index chosen by the selection box and assigns artificial Doppler bits to packets on the channels that overlap with the target Wi-Fi channel while skipping packets on other channels to ensure all bits can be captured by the Wi-Fi receiver.

As illustrated in Fig. 3, a BLE sender wants to send "10" to a Wi-Fi receiver. Meanwhile, it needs to transmit to a BLE receiver. In the first time slot, the selection box chooses channel 7, which is an overlapping channel with Wi-Fi. Thus, the DSK encoder injects an artificial Doppler shift of 80 KHz to the carrier frequency. In the second time slot, BLE sends a packet on a non-overlapping channel (Channel 9), so the DSK buffers the second bit "0". In the thirds time slot, the DSK detects that the packet is transmitted on an overlapping channel (Channel 0), it reads the buffer and pick the oldest bit as the artificial Doppler bit to shift the carrier frequency.

C. Extracting Artificial Doppler Shifts From GFSK Demodulator

Here we introduce a mechanism that allows the BLE receiver to recover the artificial Doppler bits embedded in Wi-Fi packets. To estimate the frequency shift in an incompatible packet sent in a $10\times$ wider channel, we entail the following hurdles: i) BLE devices employ the GFSK demodulation that cannot be used to decode OFDM symbols in Wi-Fi, and ii) there is no built-in frequency offset estimator that can detect the frequency shift in Wi-Fi packets.

To overcome the above hurdles, our insight is that the preamble prepending to each Wi-Fi packet has distinct structures, whose frequency changes can be reflected in the standard GFSK demodulator in BLE. The short training field (STF) in Wi-Fi preamble consists of 12 subcarriers with unit magnitude and the interval between adjacent non-zero subcarriers is 4 [13]. Without loss of generality, we use Channel 1 (with central frequency at 2412 MHz) in IEEE 802.11 standard to illustrate. Fig. 4 illustrates the STF frequency patterns received by BLE radios in 9 overlapping channels. As the subcarrier spacing in Wi-Fi is 312.5 KHz and the channel bandwidth in BLE is 2MHz, 6 or 7 subcarriers fall within one BLE channel. BLE receivers in different channels observe different frequency patterns of the Wi-Fi STF. For example, the BLE channel 3 (frequency at 2410MHz) and channel 5 overlap with two non-zero subcarriers while channel 4 overlaps with none. In the DopplerFi system, we select BLE channels that overlap with non-zero subcarriers to receive artificial Doppler bits.

The above STF patterns enable BLE receivers to differentiate different artificial Doppler shifts in Wi-Fi packets. Fig. 5 illustrates two Wi-Fi preambles with different artificial Doppler shifts sampled by a BLE radio in the frequency domain. Although the BLE radio samples at merely 1/10 clock rate compared to the Wi-Fi radio, the captured samples can still reveal the frequency shifts in the non-zero subcarriers. Such frequency shifts are also reflected in the time domain. As depicted in Fig. 6, though seemingly irregular, the Wi-Fi preamble samples captured by the BLE radio show distinct

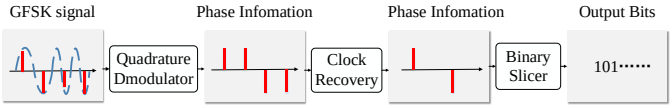


Fig. 7. BLE GFSK demodulator.

periodic patterns in the time domain.

Our goal is to extract the above patterns using the data obtainable from standard BLE modules. In particular, we exploit the GFSK demodulator, which is the core signal processing module in BLE receivers. As illustrated in Fig. 7, the GFSK demodulator is comprised of three components: quadrature demodulator, clock recovery and binary slicer. The quadrature demodulator is used to demodulate FM, FSK, GFSK signals. Mathematically, the quadrature demodulator computes the argument of product of the signal and the conjugate delayed signal as

$$\phi[n] = \arg(x[n]\bar{x}[n-1]), n = 0, 1, ..., \tag{1}$$

where $x[n]$ is the baseband sample captured by the receiver. Normally, a GFSK signal $x[n]$ can be written as

$$x[n] = \exp(2\pi f/f_s n), \tag{2}$$

where f is the frequency of the signal and f_s is the sampling rate of the receiver. Thus, we have $\phi[n] = 2\pi f/f_s$, which reflects the frequency of the signal. Then, $\phi[n]$ is sent to the clock recovery block, which is a discrete-time error-tracking synchronizer that corrects the timing error of the incoming signal. Finally, the corrected phases are put into the binary slicer, where a positive phase is demodulated as "1" and a negative phase is demodulated as "0".

Fig. 8 shows the output of the quadrature demodulator when the BLE receiver samples Wi-Fi preamble symbols with different artificial Doppler shifts. We observe that frequency shifts in Wi-Fi lead to bias in the quadrature demodulator: positive (negative) phase dominates the output corresponding to the negative (positive) frequency shift. Hence, we can also observe biases in the final output bits, which are obtainable in BLE radios.

To extract the output bits that correspond to Wi-Fi preamble symbols, DopplerFi takes the following two steps.

Step 1: GFSK extraction. First, DopplerFi employs a GFSK extractor to obtain the output bits that correspond to a Wi-Fi preamble. Once a BLE node associates with a Wi-Fi node and is in the receiver mode, it checks the quantized received signal strength indication (RSSI) samples to detect the start of a Wi-Fi packet. If the RSSI is higher than a threshold while the delayed RSSI is lower than the threshold, we regard that the start of a Wi-Fi packet is successfully detected. In particular, the GFSK extractor continuously checks the following condition.

$$\text{RSSI}[n-1] < \text{RSSI}_{\text{Threshold}} < \text{RSSI}[n], \tag{3}$$

where the threshold $\text{RSSI}_{\text{Threshold}}$ in our experiment set to 0.02. After we detect the start of a Wi-Fi frame, we stack the first 16 bits output by the binary slicer.

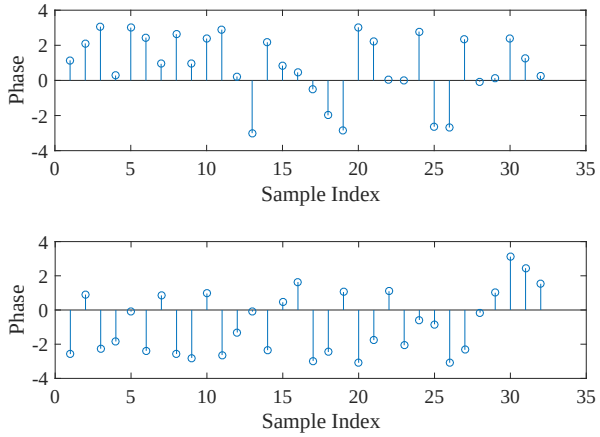


Fig. 8. BLE quadrature demodulator output. The BLE radio captures Wi-Fi preamble symbols with -130 KHz and 100 KHz shifts in the figures from top to bottom.

Step 2: GFSK demapping. Next, DopplerFi determines the artificial Doppler bit from the bit sequence obtained by the GFSK extractor. The rule is to check the bias lying in the bit sequence, as described below.

$$\begin{cases} \sum_{n=0}^{n=15} o[n] > \eta & \Rightarrow 0 \\ \sum_{n=0}^{n=15} o[n] < -\eta & \Rightarrow 1 \end{cases}, \tag{4}$$

where the decision gate η is set to 8 in our experiments.

D. Extracting Artificial Doppler Shifts From CSI

Now we present how a Wi-Fi receiver demodulates the aritificial Doppler bits in BLE packets. Since the BLE packets is transmitted in much narrower channels that overlap with only several subcarriers, we utilize CSI, which is accessible from commercial Wi-Fi cards, to estimate frequency shifts in BLE packets. To this end, we have the following observations. First, CSI values in several Wi-Fi subcarriers will be affected by BLE packets, resulting in energy spikes or deep drops, as illustrated in the top figure of Fig. 9. Second, the time duration of a BLE packet is usually longer than the duration of a Wi-Fi packet, and thus affects CSI values of multiple Wi-Fi packets. The spectrogram in Fig. 9 shows that a BLE packet overlaps with three consecutive Wi-Fi packets.

Based on these observations, DopplerFi demodulates artificial Doppler shifts in BLE packets using CSI as follows.

Step 1: CSI extraction. DopplerFi first employs the CSI Extractor to identify the CSI values that are hit by BLE packets. As adjacent subcarriers experience very similar multipath and have similar wavelengths, the CSI values across adjacent subcarriers vary smoothly except for BLE affected ones. Therefore, we locate the BLE affected CSI values with the following equation:

$$D[k] = |\text{CSI}[k] - \text{CSI}[k-1]|, \tag{5}$$

where $\text{CSI}[k]$ indicates the CSI value on the kth subcarrier. If the maximun value in $D[k]$ is higher than a threshold, the CSI Extractor identifies the corresponding subcarriers as the ones hit by a BLE packet and logs the CSI values.

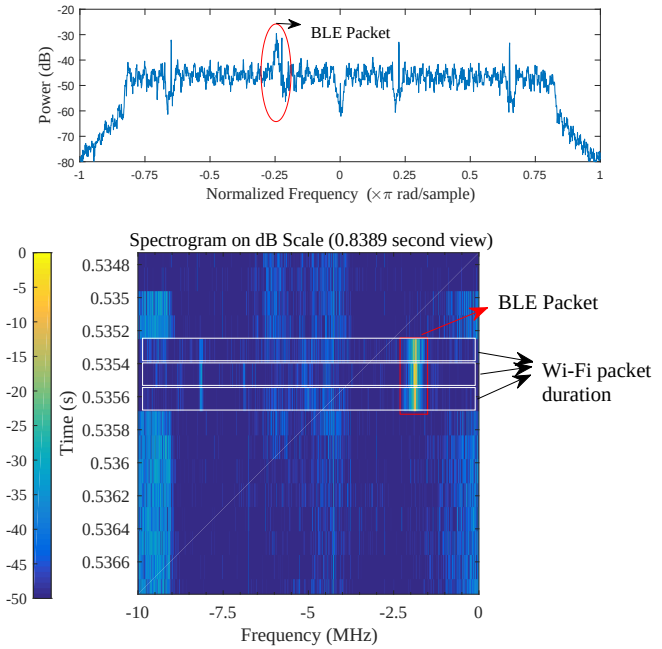


Fig. 9. Spectrum of a Wi-Fi channel captured by WARP. A BLE packet hits overlapped subcarriers of multiple Wi-Fi packets.

Step 2: CSI Demapping. Then, DopplerFi demodulates the artificial DopplerFi bits according to the logged CSI values using the CSI demapper module. Normally, there are multiple CSI vectors corresponding to one BLE packet. Therefore, after fetching the subset of the extracted CSI values, we first pick up For each CSI vector, we locate the subcarrier with the maximum variance as follows.

$$I = \arg\max_{k} |\text{CSI}[k] - \overline{\text{CSI}}|, \tag{6}$$

where $\overline{\text{CSI}}$ is the mean of CSI values on non-zero subcarriers. If there are multiple CSI vectors within one BLE's transmission time slot, we average all the indices $\{I\}$. Analogous to the GFSK demapper, the index or averaged index is compared with the standard BLE frequency to determine the bias caused by artificial Doppler shifts. Bias to lower (higher) frequency subcarriers is interpreted as "0" ("1").

E. Integration with Legacy Networks

By far we have elaborated all design components of DopplerFi. Now we discuss some practical issues when integrating DopplerFi with legacy Wi-Fi and BLE networks.

BLE and Wi-Fi link establishment. The association or pairing mechanisms in BLE and Wi-Fi are incompatible and cannot be directly applied to DopplerFi to initiate a cross-technology channel between a Wi-Fi node and a BLE node. A BLE node cannot simply scan all channels to identify the channel adopted by the target Wi-Fi node, as the BLE node cannot decode Wi-Fi beacons or data packets. To tackle this issue, we allow the BLE node to concurrently send one probe request to each Wi-Fi channel while following the mandatory FHSS sequences. Once the DopplerFi-enabled Wi-Fi node receives the probe request, it sends back a probe response to the BLE node. Then, the association between the BLE and Wi-Fi nodes is established.

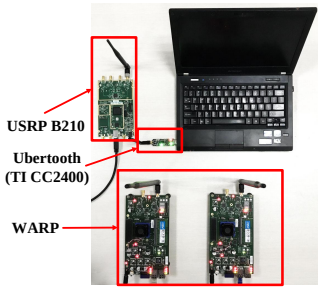


Fig. 10. DopplerFi prototype based on TI CC2400, USRP, and WARP.

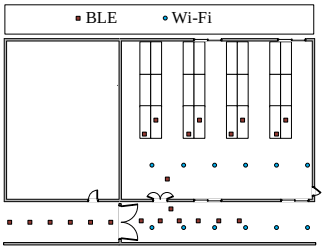


Fig. 11. Testbed topology.

DopplerFi MAC. DopplerFi nodes only piggyback artificial Doppler bits when they have legacy traffic. Thus, the MAC behaviors of DopplerFi completely conforms to channel access of legacy Wi-Fi and BLE nodes. When legacy nodes do not have enough data traffic, DopplerFi utilizes mandatory control frames such as beacons to piggyback artificial Doppler bits.

Inherent CFO and Doppler effect. The inherent CFO is caused by impairments in oscillators, and varies over time. The CFO is within a limited range of 400 Hz [11], which is much smaller than the artificial Doppler shift. Recent developments in elements and circuits make CFO even smaller. Experiments show that this amount of CFO cannot be recognized by DopplerFi and does not affect the demodulation performance. Doppler effects caused by indoor movements are merely several Herts or tens of Herts, and thus are negligible to DopplerFi.

Impact on adjacent channel. Adjacent Wi-Fi channels are separated by an interval termed as guard band, which is a 2 MHz gap to avoid adjacent channel interference. DopplerFi induces frequency shifts that less than 5% of the guard band width. Such an amount of frequency shift has been considered in the Wi-Fi design and cause no or negligible cross-channel interference. BLE transmits data using a single carrier with $\pm$250 KHz shifts in one channel. Since the BLE channel spacing is 2 MHz, the gap between two adjacent BLE channels is 1.5 MHz (taking into account the frequency shifts caused by the GFSK modulation), which can tolerate ten times of the frequency shifts induced by DopplerFi.

IV. Evaluation

A. Implementation and Experimental Setup

We implement DopplerFi on three platforms: WARP, Ubertooth, and USRP, as shown in Fig. 10. WARP nodes run the open source 802.11 reference design that conforms to standard Wi-Fi PHY/MAC. Ubertooth is a BLE adapter equipped with a LPC175x ARM Cortex-M3 MCU operating speed up to 120MHz and a TI CC2400 transceiver in 2.4 GHz [14]. USRP nodes are integrated with GFSK-based BLE PHY, which enables precise exploration of intermediate results in BLE PHY. We inject artificial Doppler shifts by dynamically

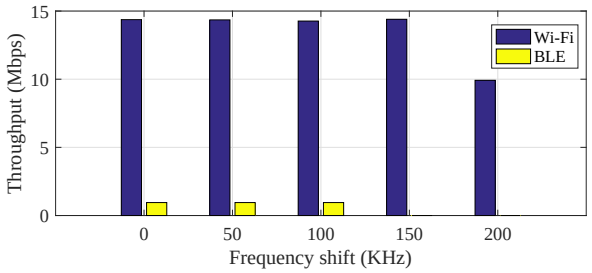
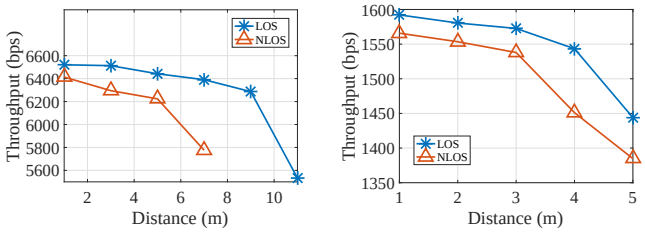
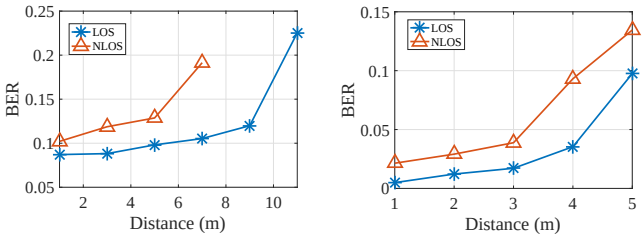


Fig. 12. Throughput of legacy transmissions embedded with DopplerFi.



(a) Wi-Fi to BLE. (b) BLE to Wi-Fi.

Fig. 13. Throughput at various distances in the ideal environment.



(a) Wi-Fi to BLE. (b) BLE to Wi-Fi.

Fig. 14. BER vs. communication distance.

changing the frequency offset in these devices. We modify the MDMCTRL register in TI CC2400 to change the frequency offset of Ubertooth. We modify the 802.11 reference design to change the frequency offset in WARP.

We conduct experiments in a typical indoor office environment as shown in Fig. 11. We deploy Wi-Fi and BLE nodes at different locations in an office or a hallway, which are separated by a stone wall. The BLE and Wi-Fi links are tested under both line-of-sight (LOS) and non-line-of-sight (NLOS) scenarios. There are 20+ Wi-Fi APs in proximity. 26 persons sit in the office area, while a few persons were walking during the experiments. The channel access of nodes is configured to conform to standard 802.11 MAC and BLE link layer. Unless otherwise specified, Wi-Fi nodes are configured to operate on channel 1 centered at 2.412 GHz.

B. Impact on Legacy Transmissions

The essential motivation of DopplerFi is that there is sufficient redundancy in carrier frequencies in that injecting a controllable amount of frequency shift has negligible impacts on legacy transmissions. This set of experiment evaluates the transparency of DopplerFi. We set different amounts of frequency shift to modulate artificial Doppler bits in Wi-Fi/BLE senders, which send files to the corresponding legacy receivers.

Fig. 12 illustrates the throughputs of legacy Wi-Fi and BLE links embedded with different amounts of artificial Doppler shift. We observe that when we inject no more than 150 KHz frequency shifts, the legacy Wi-Fi link suffer a negligible amount of throughput loss (less than 0.8%). The legacy BLE link can tolerate up to 100 KHz frequency shift with throughput loss less than 0.3%. DopplerFi injects 100 KHz and 80 KHz frequency shifts into Wi-Fi and BLE senders, and thus have negligible impacts on legacy transmissions.

C. Interference-Free Environment

To test the performance limit of DopplerFi, we configure WARP and USRP nodes to continuously send packets in a clean channel with minimal interference. This set of experiments is conducted at midnights, during which there are only few active links in the environments. According to FCC's regulations, the packet interval in BLE is fixed to be 625 μs. The length of Wi-Fi packet is set to 100 μs with packet interval of 40 μs.

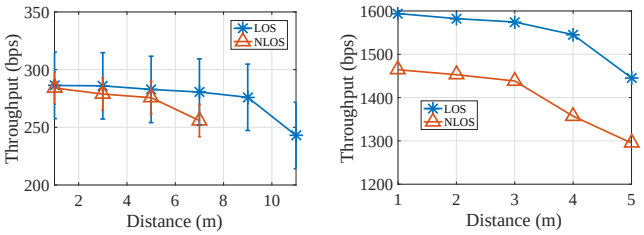
Fig. 13 shows the throughput of DopplerFi under the ideal environment. We observe that the maximum achievable throughputs for Wi-Fi to BLE (W2B) and BLE to Wi-Fi (B2W) are over 6.5 Kbps and 1.59 Kbps, respectively. For LOS links, W2B and B2W retain at least 6.3 Kbps and 1.54 Kbps throughputs within the transmission ranges of 9 m and 4 m, respectively. The maximum achievable throughput is comparable to the state of the arts [3], [5], [6].

D. Real Environment

We evaluate the performance of DopplerFi in real environments, where there are 20+ APs contending channel with DopplerFi links. We thoroughly study the BER and throughput of DopplerFi transmissions under a wide range of scenarios.

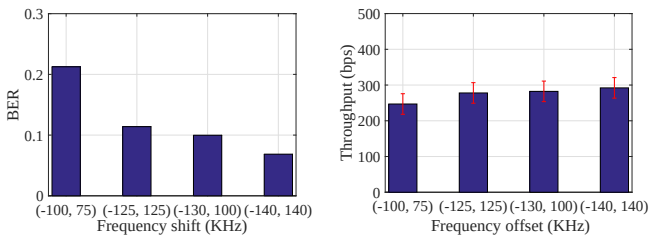
Impact of communication distance. We first investigate the performance of DopplerFi in both LoS and NLoS scenarios with different communication distances. Fig. 14(a) shows the BER of W2B links in both LOS and NLOS scenarios. The BER of W2B link in LOS scenarios grows linearly from 8.71% to 11.99% when the distance increases from 1 m to 9 m, while the BER increases sharply when the distance reaches 11 m. The BER under NLoS also has a similar trend, while reaching such a critical point at 5 m. The BER in NLoS scenarios is about 1.17× of the BER in LOS scenarios.

Fig. 14(b) depicts the BER of B2W links at various communication distances. Compared to W2B links, B2W links achieve lower BER at close distances within 4 m while suffering higher BER when the communication distance exceeds 4 m. The reason is that CSI demodulator in Wi-Fi can precisely recover the frequency shifts in BLE only when the magnitude of BLE samples is large enough. The BER gap in B2W between NLOS and LOS scenarios is up to 4.37×, which is larger than that in W2B. This is because the received signal strength drops significantly through wall in NLOS scenarios.



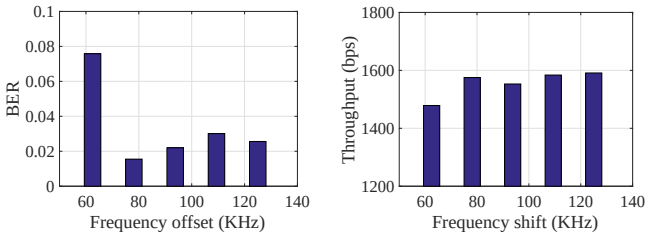
(a) Wi-Fi to BLE. (b) BLE to Wi-Fi.

Fig. 15. Throughput vs. communication distance.



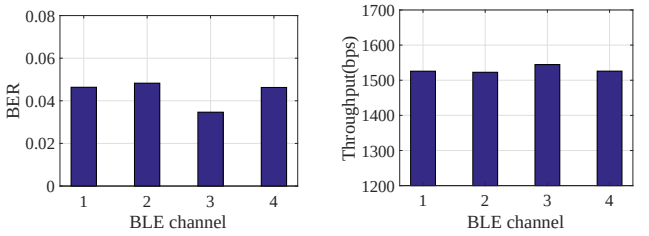
(a) Wi-Fi to BLE. (b) Wi-Fi to BLE.

Fig. 16. Impact of frequency shift on Wi-Fi to BLE links.



(a) BLE to Wi-Fi. (b) BLE to Wi-Fi.

Fig. 17. Impact of frequency shift on BLE to Wi-Fi links.



(a) BER. (b) Throughput.

Fig. 18. Impact of BLE channel.

Throughput performance at various distances is shown in Fig. 15. We observe that throughput in the real environment is much lower compared to the ideal environment without interference. This is because DopplerFi senders need to contend channel with 20+ APs and their associated clients, and thus transmission opportunities are much lower. Compared with B2W, W2B can transmit at a longer distance while yielding lower throughput. This is because BLE is less affected by surrounding Wi-Fi links.

Impact of frequency shift. Previous experiments use fixed artificial Doppler shift to embed bits. Now we study the impact of frequency shift on the performance of DopplerFi. Fig. 16 shows the impact of frequency on W2B links. The frequency offset in this experiment is shifted asymmetrically as the Wi-Fi preamble pattern received by BLE is asymmetric. We select 4 pairs of frequency shifts as depicted. As expected, BER decreases with the increment of frequency shift. We observe that with frequency shift no less than $\pm$100 KHz, W2B links yield BER as low as 0.1 and throughput higher than 250 bps. Recall that frequency shift no more than 150 KHz induces merely less than 0.8% throughput loss to legacy Wi-Fi links. Thus, DopplerFi achieves a good balance by setting frequency shift in Wi-Fi senders within a range of 100-130 KHz.

The impact of frequency shift on B2W links are plotted in Fig. 17(a), in which the BER of B2W links keep at a low level when the frequency shift is no less than 80 KHz. This is because the subcarrier spacing in Wi-Fi is 312.5 KHz, and the frequency difference must reach at least half of a subcarrier width to result in different patterns in CSI.

Impact of overlapping channel. Recall that BLE nodes adopt FHSS to transmit in different channels following a pseudo-random sequence. DopplerFi employs the DSK encoder to modulate DopplerFi bits in the overlapped channel. This set of experiments investigates the decoding ability of the CSI extractor when the Wi-Fi channel overlaps with different BLE channels. Regularly, there are 7 BLE channels overlapping with one Wi-Fi channel. Due to asymmetry, there are totally 4 different overlapping patterns.

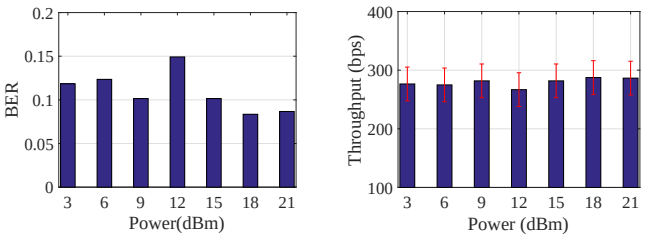
The Fig. 18 shows the BER and throughput performance under the 4 different overlapping patterns. The Wi-Fi receiver collect samples in channel 1. The BLE channels 1-4 indicate the channels from 2406 Mhz to 2412 MHz. In all overlapping cases, DopplerFi achieves BER of less than 5% and throughput larger than 1.5 KHz, which demonstrates that our demodulation scheme is robust in different overlapping patterns.

Impact of Tx power. Finally, we evaluate the impact of Wi-Fi's Tx power on DopplerFi. We use the same setting as in the distance experiments. At each location, we vary the transmit power from 3 dBm to 21 dBm by setting the power parameter in the 802.11 reference design of WARP.

Fig. 19(a) compares the BER under various Tx power. DopplerFi experiences 11.84% BER with Tx power of 3 dBm, and the BER drops to 8.67% with Tx power of 21 dBm. With higher Tx power, DopplerFi receivers yield higher signal to noise ratio (SNR) and thus the GFSK patterns are more obvious. The Fig. 19(b) draws the throughput variance across different Tx power levels. The results show that although DopplerFi tends to achieve higher throughput with higher Tx power levels, the differences are marginal, which indicates that DopplerFi is robust even when the sender adjusts its Tx power using a power control mechanism.

V. Related Work

The design of DopplerFi is inspired by CFO related side channels [15], [16], which exploit CFO-induced phase rotation in OFDM systems to facilitate different functionalities. Carpool [15] injects extra phase shift in each OFDM symbol to



(a) BER. (b) Throughput.

Fig. 19. Impact of Wi-Fi TX power.

create a free side channel in existing OFDM PHY structure to carry symbol-specific parity check bits. In contrast, PriLA [16] injects excessive CFO that varies across OFDM symbols to prevent Wi-Fi eavesdroppers from correctly decoding packets. The CFO pattern is generated from a CSI key shared by the sender and the receiver to make the CFO recoverable at the receiver. Different from these proposals, DopplerFi studies a CFO-based side channel for cross-technology communications. Technically, DopplerFi departures from existing CFO-based side channels in that existing works utilize the inherent CFO estimation block in OFDM PHY for demodulation, while DopplerFi makes it possible to demodulate CFO changes using incompatible PHY.

DopplerFi belongs to the category of cross-technology communication. The coexistence of different wireless technologies on the same 2.4 GHz band initiates the designs of enhancing performance using heterogeneous radios. To save the energy of Wi-Fi radios, ZiFi [17] leverages ZigBee radios to identify the existence of Wi-Fi networks through interference signatures caused by Wi-Fi beacons. WizSync [18] utilizes the periodic Wi-Fi beacons to calibrate the clocks of ZigBee nodes. Direction communications between incompatible PHYs are studied in [3]–[8], [19]–[21]. Esense [19] encodes bits into Wi-Fi packet lengths to create a side channel from a Wi-Fi sender to ZigBee receivers. HoWiES [20] extends this idea to allow the combination of multiple Wi-Fi packets for encoding. GSense [21] prepends a customized preamble to legacy packets to establish a common channel for heterogeneous nodes of different clock rates. Recent advances [3]–[8] explore more transparent approaches to embed side-channel bits without modifying the data of legacy packets. FreeBee [5] modulates the time interval between beacons to enable cross-technology communication. This idea has been extended to data packets [6], [7]. EMF [8] embeds multiple flows by shifting packets or flipping packet order. WiZig [4] senses packet energy to establish a cross-technology channel that is robust to noise. B2W2 [3] tunes the transmission power of BLE packets to send bits to Wi-Fi concurrently with BLE and Wi-Fi transmissions. These studies establish cross-technology channels by making changes in packet transmission time or power, which may change contention behaviors or the interference/transmission range. Differently, our goal is to explore the frequency dimension that has minimal impacts on MAC behaviors.

VI. Conclusion

This paper introduces DopplerFi, a cross-technology communication framework that aims to introduce minimal disturbance to legacy networks. By exploiting the redundancy in carrier frequency shifts, DopplerFi establishes a free side channel without modifying transmission power or time of legacy packets. While our current implementation of DopplerFi demonstrates the bidirectional communications between Wi-Fi and BLE, we believe the framework can be extended to support other wireless technologies.

References

[1] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, and P. Dutta, "The internet of things has a gateway problem," in *Proc. ACM HotMobile*, 2015, pp. 27–32.
[2] "ABI research forecasts." [Online]. Available: https://www.abiresearch.com/press/abi-research-forecasts-wi-fi-bluetooth-802154-nfc-/
[3] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2W2: N-way concurrent communication for IoT devices," in *Proc. ACM SenSys*, 2016, pp. 245–258.
[4] X. Guo, Z. Xiaolong, and Y. He, "Wizig: Cross-technology energy communication over a noisy channel," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
[5] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *Proc. ACM MobiCom*, 2015, pp. 317–330.
[6] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: Cross-technology communication with transparent morse coding," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
[7] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
[8] Z. Chi, Z. Huang, Y. Yao, T. Xie, H. Sun, and T. Zhu, "EMF: Embedding multiple flows of information in existing traffic for concurrent communication among heterogeneous iot devices," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
[9] "CC2400: 2.4 GHz low-power RF transceiver." [Online]. Available: http://www.ti.com/lit/ds/symlink/cc2400.pdf
[10] "WARP project." [Online]. Available: http://warpproject.org
[11] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang, "Fine-grained channel access in wireless LAN," in *Proc. ACM SIGCOMM*, vol. 40, no. 4, 2010, pp. 147–158.
[12] "Bluetooth core specification version 4.0," *Bluetooth SIG*, 2010.
[13] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Standard 802.11*, 2010.
[14] "Project ubertooth." [Online]. Available: http://ubertooth.sourceforge.net
[15] W. Wang, Y. Chen, Q. Zhang, K. Wu, and J. Zhang, "Less transmissions, more throughput: Bringing carpool to public WLANs," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1168–1181, 2016.
[16] W. Wang, Y. Chen, and Q. Zhang, "Privacy-preserving location authentication in wi-fi networks using fine-grained physical layer signatures," *IEEE Trans. Wireless Comm.*, vol. 15, no. 2, pp. 1218–1225, 2016.
[17] R. Zhou, Y. Xiong, G. Xing, L. Sun, and J. Ma, "ZiFi: wireless LAN discovery via ZigBee interference signatures," in *Proc. ACM MobiCom*, 2010, pp. 49–60.
[18] T. Hao, R. Zhou, G. Xing, M. W. Mutka, and J. Chen, "Wizsync: Exploiting Wi-Fi infrastructure for clock synchronization in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 6, pp. 1379–1392, 2014.
[19] K. Chebrolu and A. Dhekne, "Esense: Communication through energy sensing," in *Proc. ACM MobiCom*, 2009, pp. 85–96.
[20] Y. Zhang and Q. Li, "Howies: A holistic approach to ZigBee assisted WiFi energy savings in mobile devices," in *Proc. IEEE INFOCOM*, 2013, pp. 1366–1374.
[21] X. Zhang and K. G. Shin, "Gap sense: Lightweight coordination of heterogeneous wireless devices," in *Proc. IEEE INFOCOM*, 2013, pp. 3094–3101.