# Simultaneous Bi-directional Communications and Data Forwarding using a Single ZigBee Data Stream

Zicheng Chi, Yan Li, Zhichuan Huang, Hongyu Sun, and Ting Zhu
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Email: {zicheng1, liy1, zhihu1, hongyu, zt}@umbc.edu

***Abstract*— With the exponentially increasing number of Internet of Things (IoT) devices and the huge volume of data generated by these devices, there is a pressing need to investigate a more efficient communication method in both frequency and time domains at the edge of the IoT networks. In this paper, we present Amphista, a novel cross-layer design for IoT communication and data forwarding that can more efficiently utilize the ever increasingly crowded 2.4 GHz spectrum near the gateway. Specifically, by using a single ZigBee data stream, Amphista enables a ZigBee device to send out two different pieces of information to both the WiFi gateway and another ZigBee device. We further leverage this unique feature and design a novel forwarding protocol that can simultaneously forward uplink (e.g., collecting sensing data) and downlink (e.g., disseminating software updates) data by using a single ZigBee data stream. Our extensive experimental results show that Amphista significantly improves throughput (by up to 400x) and reduces the latency.**

## I. Introduction

The number of Internet-of-Things (IoT) devices will grow exponentially to reach 26 billion by 2020 [1] and 1 trillion by 2025 [2]. Each person will touch or use 300 to 500 "smart" devices every day by 2032 [2]. These devices will also generate huge amount of wireless traffic. Based on the Cisco Global Cloud Index [3], the data created by these devices will reach 42.3 ZB (i.e., $4.23\times10^{22}$ bytes) per month and will be 49 times higher than total data center traffic by 2019. Therefore, there is a pressing need for conducting the computing at the edge of the network instead of in the cloud.

In edge computing, it is extremely important to efficiently collect the huge amount of data generated by these densely deployed IoT devices at the edge (e.g., gateway) of the network. This is because most of these IoT devices are using the industrial, scientific, and medical (ISM) band, the evergrowing number of IoT devices and the huge amount of data generated by these devices will cause the ISM 2.4 GHz band extremely crowded. This issue is becoming worse at the gateway side because all the data from heterogeneous IoT networks needs to be sent to the gateway through the overlapped wireless channels. For example, in Figure 1, a WiFi high quality (HQ) video camera is uploading real-time surveillance video to the gateway using WiFi channel 6, which is overlapped with ZigBee channels 16 to 19. To avoid WiFi packets colliding with ZigBee packets at the gateway, traditional approaches adopt either carrier-sense multiple access (CSMA) or time-division multiple access (TDMA). These approaches can effectively reduce the packet collisions when the number of IoT devices is small.

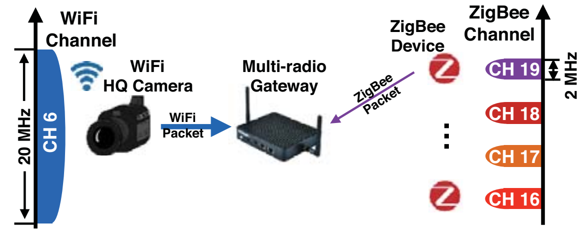



Fig. 1: **Limitations of CSMA (or TDMA) -based approaches.** The transmission of ZigBee packets in the 2 MHz ZigBee channel 19 to the gateway will block the WiFi packets in WiFi channel 6. Therefore, it i) introduces the delay to WiFi traffic; and ii) reduces the spectrum utilization in the 20 MHz WiFi Channel 6.

However, with the exponentially increasing number of IoT devices and huge volume of data generated by these devices, these approaches may cause inefficient communication in both time and frequency domains. In the time domain, only one device is able to send the packets to the gateway at any given time. For example, if a ZigBee device is sending packets to the gateway, the WiFi HQ video camera needs to wait. This will introduce a significant latency and an interruption to the real-time WiFi video traffic, especially when the number of ZigBee devices increases; ii) in the frequency domain, the transmission from a narrow-band ZigBee device will prevent the wide-band WiFi device's transmission. Therefore, the spectrum utilization is extremely low. For example, in order to avoid interference, when a ZigBee device is sending packets to the gateway using a 2 MHz channel (e.g., channel 19), the WiFi HQ video camera can not use the whole 20 MHz WiFi channel 6 that is overlapped with ZigBee's channel 19. One may argue that the WiFi HQ video camera can use another WiFi channel. However, all the WiFi channels are overlapped with ZigBee channels. When the number of IoT devices exponentially increases, it is not possible to find a clear and designated channel that can only be used by WiFi devices.

To address this limitation, we propose a novel design across physical and network layers. Our proposed approach – Amphista enables WiFi and ZigBee devices simultaneously transmit their packets to the gateway.

At the physical layer, Amphista embeds the ZigBee to WiFi (Z2W) data into ZigBee to ZigBee (Z2Z) communication by smartly modulating ZigBee packets' transmission power. The

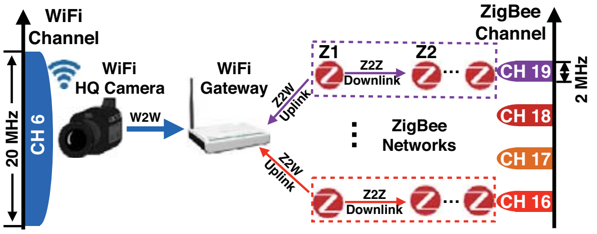



Fig. 2: **Advantages of Amphista.** When ZigBee devices uploading the sensing data to the gateway using Z2W communication which embeds the sensing data into ZigBee's transmission power, these devices can also use the same stream of normal ZigBee packets for disseminating software updates or control messages among themselves within the same ZigBee channel. Overall, Amphista only needs a single WiFi radio and supports the following concurrent communications and data forwarding: 1) one WiFi high quality (HQ) video camera uploading video; 2) four groups of ZigBee devices uploading sensing data to the same gateway; and 3) four groups of ZigBee devices disseminating software updates or control messages among themselves.

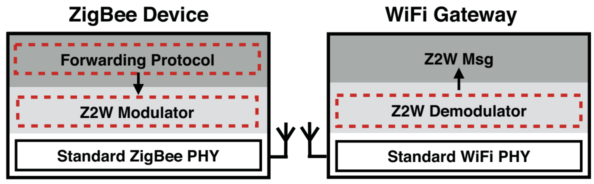



Fig. 3: **System Architecture of Amphista.** Our design is highlighted in red dashed boxes.

Z2W data can be detected by the channel state information on WiFi gateway along with WiFi to WiFi (W2W) transmission. As shown in Figure 2, the unique feature of Amphista is that when a ZigBee device ($Z1$) is conducting Z2W communication with the gateway, the same stream of ZigBee packets can be leveraged for forwarding the data from $Z1$ to another ZigBee device ($Z2$). By doing this, Amphista enables a ZigBee device to send out two different pieces of information to both the WiFi gateway and another ZigBee devices using a single ZigBee data stream that coexists with WiFi to WiFi communication. To summarize, Amphista supports three types of simultaneous communications: i) WiFi to WiFi (W2W), ii) ZigBee to ZigBee (Z2Z), and iii) Z2W communications at the same time within the same channel. Therefore, Amphista can provide much higher spectrum utilization than CSMA and TDMA methods.

At the network layer, we further leverage the unique physical layer communication feature and design a novel data forwarding protocol that can *simultaneously* provide **uplink** (i.e., collecting IoT sensing data) and **downlink** (i.e., disseminating software updates or control messages) **data forwarding** using a *single* ZigBee data stream. Overall, Amphista can support 4 independent ZigBee uplink and downlink data streams that are simultaneously coexisting with the WiFi traffic. Therefore, the spectrum utilization at the gateway is significantly increased. In summary, our main contributions are as follows:

- We developed a novel *simultaneous* communication method that enables three types of simultaneous communications: W2W, Z2Z, and Z2W. Different from existing approaches, our method enables the heterogeneous IoT devices to transmit their packets to the gateway at the same time and within the overlapped channel. It can significantly increase the spectrum utilization and reduce the number of retransmissions due to the packets collision between WiFi and ZigBee devices.
- We built an effective handshaking process that can minimize the cross-technology interference. Our evaluation results demonstrate that our approach introduces negligible interference to the original WiFi-to-WiFi and ZigBee-to-ZigBee communications.
- Different from existing cross-technology communication methods that only focus on PHY layer, we designed a distributed simultaneous uplink and downlink data forwarding scheme that uses the same stream of ZigBee packets to *simultaneously* i) upload sensing data to the gateway and ii) disseminate the software updates and control messages inside the ZigBee network. To the best of our knowledge, this is the first technology that can provide uplink and downlink data forwarding simultaneously. Our scheme only needs one-hop neighbors' information. Therefore, it is simple, symmetric, highly distributed, and scalable.
- We extensively evaluated our design under four different real-world settings. Amphista significantly improves throughput (by up to 400x) and reduces the latency. Moreover, Amphista's spectrum efficiency is 2.29 times higher than traditional CSMA and TDMA-based approaches.

## II. Design Overview and Challenges

Figure 3 shows the system architecture of Amphista, which provides the following two functions:

**I) ZigBee to WiFi (Z2W) Gateway Communication.** Since ZigBee and WiFi radios use fundamentally different physical layers, ZigBee cannot directly communicate with WiFi. The design challenge is how to enable the Z2W communication without i) changing the physical layers of both ZigBee and WiFi; and ii) affecting the W2W and Z2Z communications. To address this challenge, we leverage WiFi's fine-grained channel state information for decoding the embedded message from a ZigBee sender to the WiFi gateway. Specifically, after handshaking with WiFi gateway, the ZigBee device embeds Z2W message by modulating the optimized transmission power of ZigBee packets. When the WiFi gateway receives the WiFi packets that are overlapped with Z2W message embedded ZigBee packet, the gateway i) extracts the Z2W message; and ii) recovers the WiFi data by the native equalization scheme. Since the ZigBee packets can also be received by another ZigBee device, Amphista enables three types of simultaneous communications: i) Z2W, ii) W2W, and iii) Z2Z.

In our design, since the WiFi gateway to ZigBee communication is only used for sending out the control mes-

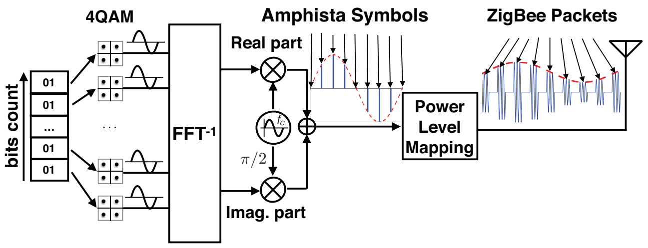



Fig. 4: 4QAM-OFDM on top of the channel state information

sages (e.g., acknowledgement), the required throughput can be relatively low. Therefore, we can use existing WiFi to ZigBee (W2Z) cross-technology communication techniques (e.g., WEBee [4]). However, WEBee cannot conduct the ZigBee to WiFi (Z2W) communication. Therefore, we propose our own unique Z2W design.

**II) Forwarding Protocol.** Our forwarding protocol leverages the simultaneous communication's unique feature – when a ZigBee device is conducting Z2W communication with the gateway by modulating the transmission power of ZigBee packets, the same stream of ZigBee packets can be used to forward the data to another ZigBee device. In this forwarding protocol, there are two main functions: i) neighbor maintenance, which maintains the throughput to each ZigBee node's 1-hop neighbors and to the WiFi gateway; ii) forwarding decision, which splits the data into three sending buffers. By doing this, we can minimize the total number of transmissions in ZigBee networks while maximizing the uploading throughput to the WiFi gateway.

## III. Z2W Communication

In this section, we introduce how to achieve Z2W communication without i) affecting Z2Z and W2W communications and ii) modifying physical layers of WiFi and ZigBee. Specifically, we design novel modulation and demodulation schemes on top of the standard WiFi and ZigBee physical layers.

### A. Z2W Modulation at ZigBee Sender Side

The objective of Amphista's physical layer design is to simultaneously enable Z2Z and Z2W communications. As an overview, our design for Z2W communication is to leverage the under-utilized power transmission capabilities in commodity ZigBee radios. Therefore, by modulating the transmission power levels of each ZigBee packet, we embed information for Z2W message in each ZigBee packet. We introduce the basic modulation schemes (PAM, MSK, and OFDM) for embedding WiFi information in the power levels of each ZigBee packet:

**Pulse-Amplitude Modulation (PAM)** is the simplest solution, which directly modulates the Z2W message on the power levels of ZigBee packets. Each power level is related to symbols that represent the binary data combinations. Though PAM achieves high throughput, the bit error rate (BER) increases significantly in noisy environments.

**OFDM** divides the spectrum into multiple subcarriers and modulates the data on each subcarrier. Since the available ZigBee's transmission power levels are limited, we use 4 Quadrature Amplitude Modulation (4QAM) for each subcarrier. As shown in Figure 4, the detailed implementation of 4QAM-OFDM is as follows: i) the two-bits combination symbols are first mapped into 4 sine waves; ii) the 4 sine waves are mapped and summed (using the IFFT) into an output signal; iii) the output signal is mapped into the power levels for each packet.

Compared to PAM, 4QAM-OFDM provides better spectrum efficiency while it can also work under noisy environment. The only issue is that the implementation of OFDM needs to calculate IFFT, which is difficult for ZigBee to calculate it in real-time given its limited computation and energy resources. To solve this problem, we precompute a lookup table to map the data bits and modulated power levels. The size of lookup table is $log_2 M \times S \times C$ bits, which is small and can be stored in ZigBee's memory. The $M$, $S$, and $C$ are the number of states used to represent data, the symbol rate, and the information capacity of data bits, respectively. In our implementation, the size of lookup table is around 64KB, which is much less than flash size of ZigBee devices (e.g., flash size of TelosB device is 1024KB).

### B. Z2W Demodulation at WiFi Gateway

At the WiFi gateway receiver side, the Z2W demodulator's main functionalities are i) extracting the embedded Z2W message and ii) recovering the original W2W message.

*1) Design of Z2W Message Extractor:* In this section, we explore how to extract the embedded signals in the received WiFi packets in a noisy environment. The embedded Z2W message can be extracted by measuring the amplitude of channel state information from demodulating the WiFi signal. A threshold filter removes channel state information that do not contain embedded messages. Thus, the values passed to the demodulators are values with embedded ZigBee message and channel interference. In the demodulation scheme, we first obtain the channel state information values defined as $x(t)$. We can then derive the quadrature symbols from the following equation:

$$\begin{aligned} b_i(t) &= x(t) \cdot \cos(f_c + \phi) \\ b_q(t) &= x(t) \cdot \sin(f_c + \phi) \end{aligned} \tag{1}$$

Where, $f_c$ is the Nyquist frequency that is inversely proportional to the length of the ZigBee packet. The demodulation algorithm maintains the phase state $\phi$. After the quadrature symbols are computed, we pass the symbols into the predetermined demodulation scheme. Each modulation (PAM and 4-QAM OFDM) scheme maps the symbols using lookup tables to their respective bits.

*2) Recovering the original WiFi Message:* While extracting Z2W messages, the WiFi receiver also recovers the WiFi messages. The recovering is possible because of the three techniques 1) interference cancellation, 2) equalization, and 3) bit error correcting techniques. First, from the received WiFi signal, we subtract out portions of interfering ZigBee signals. Then, we apply an equalization method on the remaining WiFi signals using a channel sensing technique. Finally, after the signals are decoded to bits, an error correcting code to the bits associated those equalized WiFi subcarriers that are overlapped with ZigBee channels.

**Interference cancellation:** Our interference cancellation functions by simultaneously transmitting data and commonly repeated wireless signals (e.g., beacons and headers) to recover original WiFi messages. By sensing the amount of interference, we are able to obtain ZigBee packets' interference. Then, we apply the interference cancellation technique using the following equation:

$$z(t) = x(n) - \sum_{n=1}^{N} \psi_n x\left(t - \tau_n\right) \tag{2}$$

Where, $z(t)$ is the recovered WiFi signal at time $t$, $x(n)$ is the received signal that contains noise. $\psi_n$ is the complex coefficient describing the interfering signal. $\tau_n$ is the delay respective to the transmission time of the interferer. To determine $\psi_n$, we use the standard least mean square (LMS) adaptive filter. We update the $\psi_n$ variable by using the following equation:

$$\psi_{n+1} = \psi_n + \mu \sum x^* \left(t - \tau_n\right) \cdot z(t) \tag{3}$$

Where, $\mu$ is the updating step size control factor. After the noise subtractions, we must correct the phase in the quadrature signals by using the frequency selective fading equalization. $x^*$ is related to ZigBee's Pseudo Noise (PN) signal. ZigBee uses a shared PN code to spread to a wider channel becoming more resistant to interference. This PN code is a standard constant array of random-like numbers that is multipled by the ZigBee transmitter and ZigBee and Amphista receivers. The intuition behind why LMS adaptive equalization functions is that because we know the shared ZigBee PN coding, we can remove ZigBee interference in the received WiFi packet.

**Frequency Selective Fading Equalization:** Because of human movements and objects that reflect RF signals, the WiFi channel experience strong frequency selective fades. Existing commodity WiFi devices must recover from faded signal due to channel interference such as fading and delays using an equalizer. Since the Z2W message in Amphista's simultaneous communication also causes distortions within specific frequency bands. The commodity WiFi receiver treats the interference from ZigBee as frequency selective fading. Therefore, the native equalizer is able to recover the original WiFi signal by using the pilot tones that are signals agreed upon by the transmitter and receiver.

To define this interference, the receiver computes the channel's frequency response $(H_n)$ for each pilot tone. The receiver is supposed to receive $\overrightarrow{X_w}$ but instead received $\overrightarrow{y}$. Computing the difference vector $\overrightarrow{h}$, the receiver applies correction to all the bits around pilot tone's frequency. We model the interference correction process as quadrature values due to the sine wave signals. The equalization scheme for the interfered carrier correction quadrature values are defined below:

$$i'_n = H_n \left[i_n \cos\left(\theta_n\right) + j_n \sin(\theta_n)\right] \tag{4}$$

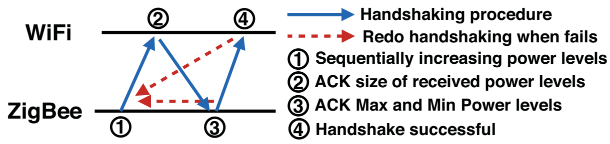



Fig. 5: Handshaking Process

$$j'_n = H_n \left[j_n \cos\left(\theta_n\right) - i_n \sin(\theta_n)\right] \tag{5}$$

Where, $i'_n$ and $j'_n$ are the correction quadrature values. $i_n$ and $j_n$ are the received pilot tone quadrature values, which are sent to the traditional WiFi OFDM demodulation systems, then the original WiFi bits are recovered.

**Error Correcting Codes:** After the WiFi bits are demodulated from each WiFi subcarrier, we note that the subcarriers associated with overlapped ZigBee channels may have a relative high probability of bit error. By appending Error Correcting Codes (ECC) to the data stream during concurrent communication, we can also increase the probability of reception. Because the corruption in the bit stream can be expected, as ZigBee packets are transmitted within a fixed frequency band, we can append extra ECC to non-affected bits. We utilize Reed-Solomon ECC as the scheme allows for variable matrix recovery sizes. We produce this matrix by encoding data chunks with a polynomial. For a message $m$ with a length of $i$, we define the error correct polynomial in the following equation:

$$m(x) = m_0 + m_1 x + m_{i-1} x^{i-1} \tag{6}$$

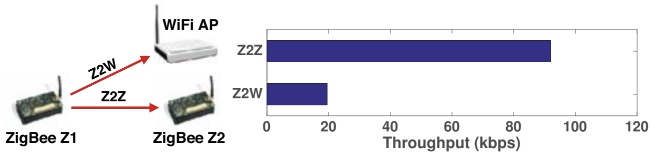
Reed-Solomon ECC depends on solving $n = i + 2s$ non-zero points, where $s$ is the maximum number of errors. Thus, $n$ is directly related to the number of appended ECC bits to the data stream. Intuitively, as the Z2W link quality or ZigBee signal strength increases, we increase the number of appended ECC bits. This solution is optimal because the number of appended ECC bits is directly related to the amount of ZigBee interference. From the experimental results, the number of appended ECC bits is always less than 1% of the total WiFi data.

### C. Handshake between ZigBee and WiFi

To coordinate between WiFi and ZigBee devices, we introduce a handshaking scheme to establish the Z2W communication. The goals of handshaking are: i) power range determination and ii) phase synchronization for various modulation schemes.

**Power Range Determination:** To ensure concurrent Z2W and W2W communications, the concurrent transmissions of WiFi and ZigBee must not saturate analog to digital converters' inputs of WiFi and ZigBee, but still remain above the sensitive levels. The power level handshaking scheme allows concurrent power level transmissions.

**Phase Synchronization:** 4QAM-OFDM is used to embed Z2W data in power levels that requires phase synchronization to discriminate between each symbol. In our design, we maintain phase states for each symbol and account for clock




(a) Experiment Setup (b) **Experiment Results**

Fig. 6: Concurrent communications of Z2Z and Z2W

drifts of the transmitter and receiver. During the handshaking protocol, the intervals between each power level measurements $\theta(t)$ provide timing information that defines phase $\phi(t)$.

$$\phi(t) = \theta_{i+1}(t) - \theta_i(t) \tag{7}$$

We perform the subtraction on all the handshaking values and obtain a rate of phase change.

The process of the handshaking is shown in Figure 5. Firstly, when the communication is establishing, the ZigBee transmitter provides the phase state by increasing the power levels sequentially so that modulation requiring quadrature states can be synchronized. Secondly, the WiFi receiver acknowledges ZigBee (by using WiFi to ZigBee technique such as WeBee [4]) the maximum and minimum power levels it received from the ZigBee transmitter to complete the handshake. Finally, if the sequence is not completely detected, the acknowledgement will contain the error message to invalidate and redo the handshake.

## IV. Forwarding Protocol

In previous sections, we show how ZigBee devices convey data to WiFi AP while concurrently communicating with other ZigBee devices. To demonstrate this scheme, we conduct an experiment with the setup shown in Figure 6(a) yielding results in Figure 6(b). ZigBee device $Z_1$ communicates to another ZigBee device $Z_2$ at 90kbps while concurrently to WiFi AP at 20kbps. Since different Z2Z and Z2W links suffer different fading and interference, the nodes have various packet reception rate (PRR). To utilize this unique feature of concurrent communications, we design a forwarding protocol for ZigBee networks.

### A. Forwarding Procedure

In this section, we present the Amphista forwarding protocol that enables efficient concurrent uplink and downlink communications across heterogeneous IoT devices. The design goal of this forwarding protocol is to maximize the throughput and minimize the delay in IoT networks. A state machine diagram running in the ZigBee device is shown in Figure 7. Specifically, a ZigBee device is in one of three states at any time: (i) maintenance, (ii) sending, and (iii) receiving. Transitions between the states are triggered by events. In the rest of this section, we explain the operations in each state in details.

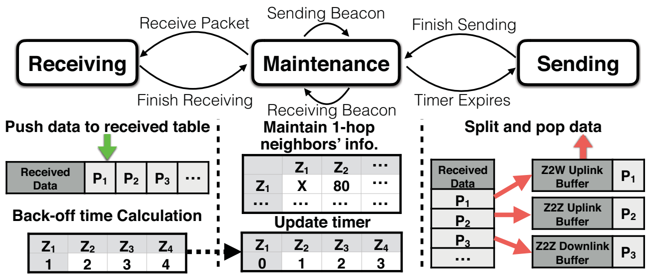


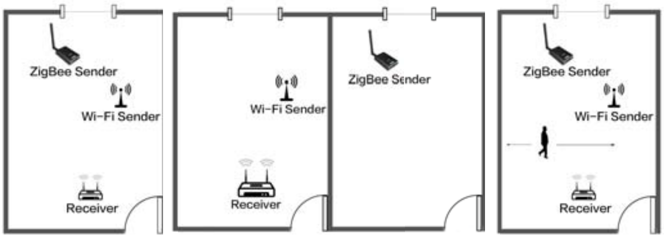

Fig. 7: State Machine Diagram of the forwarding protocol

*1) Maintenance State:* After a ZigBee device is turned on, the device enters the maintenance state. The main purpose of maintenance state is to i) update the PRR of 1-hop Z2Z ($p_{i,j}$) and Z2W ($q_i$) communications; ii) calculate $EDW$; and iii) update the status table of received packets to decide whether to forward them or not. To update the PRR, each ZigBee device will periodically send out a probe message. Another task in maintenance state is to update the timer that is set in receiving state to minimize the redundant transmissions. Once the timer of any packet expires, the device enters the sending state to decide how to send the packets through Z2Z and Z2W communications. Additionally, the new uplink and downlink requests will also be updated in maintenance state.

*2) Receiving State:* When a ZigBee device receives a data packet, the device enters the receiving state and adds a new entry in the received data table for the received data. The received data table contains the status of each received data. For each entry of the table, it contains the data ID, data destination and the back-off time to forward the data. Since the data is broadcasting in the ZigBee network, if a device receives a new data, there will be other ZigBee devices that can also receive the data. To minimize the redundant transmissions, each device will set a back-off time to forward the received data. The detailed design of back-off time will be discussed in Section IV-A4. If the device receives redundant uplink data, which means that another device with shorter back-off time uploads the uplink data. Then, the back-off time of the data will be updated as infinity so that the data will not be forwarded anymore.

*3) Sending State:* When a device has data packets in the received data table, it enters the sending state. In this state, it selects packets to send through Z2W uplink, Z2Z uplink and downlink to minimize the overall number of transmissions in the network. The key idea is to apply concurrent Z2Z downlink and Z2W uplink communication first because its aggregated throughput is higher, and apply concurrent Z2Z and Z2W uplink communication when there are extra uplink packets after concurrent Z2Z downlink and Z2W uplink communication. Specifically, the number of packets to send through Z2W uplink ($P_{AP}$), Z2Z uplink ($P_U$) and downlink ($P_D$) can be calculated as:

$$P_{AP} = \begin{cases} N_U, & \text{if } \frac{N_U}{N_D} \leq \frac{U_{AP}}{D} \\ N_U - \frac{N_D U_{AP} U}{D(U_{AP}+U)}, & \text{otherwise} \end{cases} \tag{8}$$




(a) LoS (b) NLoS (c) w/ Human

Fig. 8: Three indoor experiment scenarios

$$P_U = \begin{cases} 0, & \text{if } \frac{N_U}{N_D} \leq \frac{U_{AP}}{D} \\ (N_U - \frac{N_D U_{AP}}{D})\frac{U}{U_{AP}+U}, & \text{otherwise} \end{cases} \quad (9)$$

$$P_D = \begin{cases} N_D, & \text{if } \frac{N_U}{N_D} \geq \frac{U_{AP}}{D} \\ \frac{N_U D}{U_{AP}}, & \text{otherwise} \end{cases} \quad (10)$$

Where $N_U$ and $N_D$ are the number of uplink and downlink packets that are ready to send based on the packet table. $U_{AP}$, $U$ and $D$ are the Z2W uplink throughput, minimum uplink Z2W throughput of device's neighbors and maximum downlink Z2Z throughput of device's neighbors. After sending all the packets, the device goes back to maintenance state.

*4) Back-off Timer Design:* As introduced in Section IV-A2, since the forwarding decision is made in a distributed manner, the back-off timer is applied to reduce both the redundant uplink and downlink transmissions, and guarantee the downlink packets will be sent out to all the ZigBee devices. At a specific time while the potential forwarder receives packets from its neighbors, every forwarder calculates the priority parameter ($PP$) according to i) expected transmission count metric ($ETX$ [5][1]) and ii) the data size of uplink and downlink. We can divide it into three conditions to calculate the $PP$:

• $\mathbf{P_D = 0, P_U > 0}$**:** In this case, each device $Z_i$ calculates all the $ETX$ between all neighbors and pick the maximum one as $PP$ (where $PP = \max_{j\in[1,n],j\neq i} ETX Z_i, Z_j$).

• $\mathbf{P_D > 0, P_U > 0}$**:** When the forwarder has both the uplink and downlink packets in the buffer, generally, it has higher priority to forward neighbor's packets because when performing the downlink transmission, the uplink data can be sent directly to WiFi through Z2W link to eliminate uplink Z2Z transmissions. In this condition, $PP = \sum p_{i,j} + q_i$.

• $\mathbf{P_D > 0, P_U = 0}$**:** In this case, the network system cannot utilize the Z2W link to improve the performance. However, the devices can still use Z2W link to exchange control messages in order to reduce the overhead.

When a ZigBee device calculates the priority parameter ($PP$), it updates its back-off timer based on the $PP$ value. Intuitively, the higher the $PP$ value, the smaller the back-off timer should be because we always select the forwarder which can reduce the overall number of transmissions.

## V. Experimental Evaluation

We extensively evaluated our Amphista system in an academic building, which has a lot of other WiFi access points that create interference.

### *A. Experimental Setup*

In our experiment, we implemented our design on USRP devices and off-the-shelf ZigBee devices TelosB. We deployed the system in following three different scenarios:

• **Line-of-Sight (LoS):** As shown in Figure 8(a), the sender and the receiver were within Line of Sight at distances of 0.5, 3, and 10 meters .

• **None-Line-of-Sight (NLoS):** As shown in Figure8(b), the sender and the receiver were in different rooms with distances of 4, 7, and 10 meters.

• **Human Interference:** As shown in Figure 8(c), a person was in the middle of the sender and the receiver.

• **Mobile Scenarios:** A person was walking with a device in the pocket or on the wrist.

### *B. Comparison with TDMA and CSMA*

To examine the efficiency of Amphista, we compared it with two traditional MAC schemes: CSMA and TDMA. We conducted an experiment by using a pair of WiFi devices and 4 pairs of ZigBee devices. Each pair consists of two devices communicating with each other. In CSMA and TDMA, we utilized a multi-radio gateway that is equipped with both ZigBee and WiFi radios. We use the following four metrics: i) **packet reception probability (PRR)**: the number of packets received divided by the total number of packets transmitted; ii) **power efficiency**: the amount of energy required to transmit a bit; iii) **spectrum efficiency**: the amount of bandwidth required to transmit a bit; and iv) **throughput**: total number of bits transmitted per second.

As shown in Figure 9, the PRR of Amphista is as high as CSMA or TDMA, however, Amphista only needs 2.59 dBm to transmit 1 kilo-bit data comparing to 5.36 dBm/kbit and 5.2 dBm/kbit of CSMA and TDMA, respectively (see Figure 10). Moreover, Amphista's spectrum efficiency is 2.29 times as high as the popular CSMA scheme (see Figure 11). Furthermore, as shown in Figure 12, Amphista's throughput is around **two times** as high as CSMA in different scenarios. This is because by using CSMA or TDMA, ZigBee and WiFi devices are competing for accessing the overlapped channel. When the ZigBee device is transmitting, the WiFi device will avoid collisions. Since the ZigBee device uses only 2 MHz bandwidth, compared with 20 MHz bandwidth of WiFi, the major part of the spectrum is wasted, which results the overall performance decreases. However, since Amphista enables concurrent W2W, Z2Z, and Z2W communications, ZigBee and WiFi can concurrently communicate to the gateway.

[1]A common metric used to find high throughput path in multi-hop wireless networks.

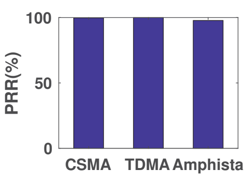



Fig. 9: Packet Reception Ratio(PRR)

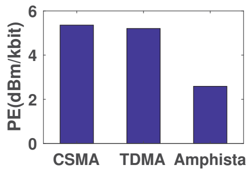



Fig. 10: Power Efficiency (PE)

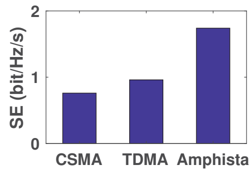



Fig. 11: Spectrum Efficiency (SE)

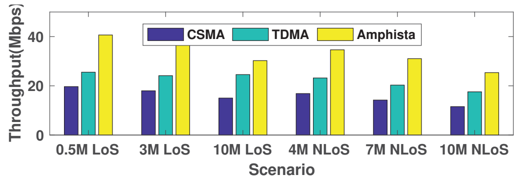
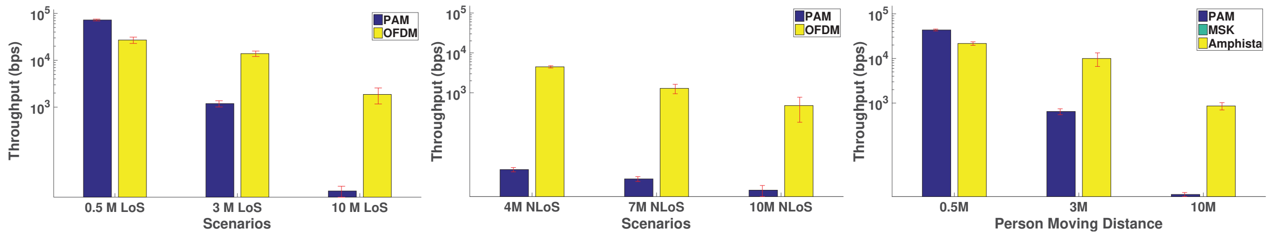



Fig. 12: Throughput

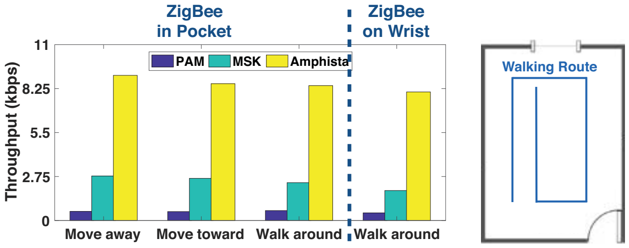



(a) LoS

(b) NLoS

(c) Human Interference

Fig. 13: Z2W Throughput




(a) Throughput

(b) Route

Fig. 14: The performance of ZigBee to WiFi communication in four different mobile scenarios. The throughput error ranged about 15-20% which means our modulation schemes remain robust and against fading caused by human movement.

### C. Z2W's Performance

We evaluated our Z2W's performance in both Line-of-Sight (LoS) and Non-Line-of-Sight scenarios by changing the distance from the sender to the receiver.

*1) Line-of-Sight:* To demonstrate Amphista's system performance of Z2W communication, we evaluated our system in a line-of-sight scenario. The result is shown in Figure 13(a). When the distance between the WiFi and ZigBee devices increase, the signal strength decreases causing loss of WiFi sampling fidelity. We conclude that although, PAM provides 10 times better throughput, as distance increases in real world, the throughput drops exponentially which is not usable. OFDM has a relative stable performance over distance. The highest throughput of OFDM is 27 kbps in a real world setting (0.5 meter LoS).

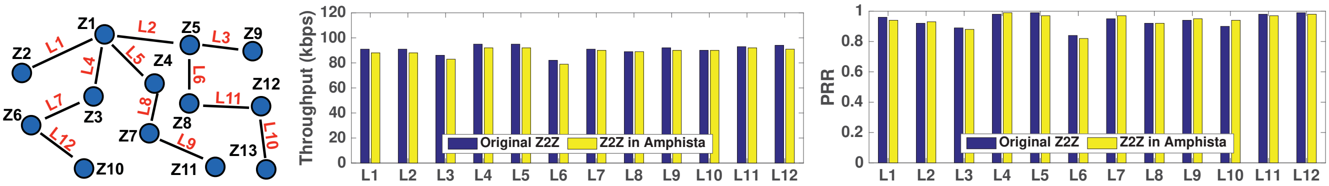
*2) Non-Line-of-Sight:* We also conducted experiments of Z2W communication in NLoS at different distances (see Figure 13(b)). In NLoS, RF signal experience diffraction, reflection, and increased fading contributing to increased multipath interference. Because of increased multipath interference, the transmission between WiFi and ZigBee contains more distortions lowering sampling fidelity. PAM cannot recover from the increased multipath interference and thus perform 100 times worse. The throughput of OFDM remains the same as expected. By embedding information in phase, OFDM modulation scheme provides robustness against multipath and sampling fidelity loss. The highest throughput of OFDM in NLoS is 4.5 kbps at 4 meters.

*3) Human Interference:* Human movement introduces Doppler effect, which causes frequency shifts and thus the received signal is distorted. At a short distance, PAM remains relatively unaffected due to the higher signal strength from the unaffected signals (see Figure 13(c)). As the distance increases, PAM decreases exponentially while OFDM decreases linearly. OFDM performs 7 times better in the medium distance and equivalently in long distance. Based on these results, we conclude that OFDM is robust under human interference.

*4) Mobile Scenarios:* In this experiment, we evaluated the throughput (results shown in Figure 14(a)) by attaching the ZigBee device to the human body in four different mobile scenarios: i) walking toward the WiFi receiver with the ZigBee device in a pocket; ii) walking away from the WiFi receiver with the ZigBee device in a pocket; iii) walking around the office with the ZigBee device in pocket; and iv) walking around the office with the ZigBee device on wrist. The route in scenarios iii) and iv) is shown in Figure 14(b). The results show that the throughput remained stable during each scenario. The throughput error ranged about 15-20%. The phase-based modulation remains robust against fading caused by the human movements.

| **Approach** | Amphista | FreeBee [6] | EMF [7] | C-mose [8] | ZigFi [9] |
|---|---|---|---|---|---|
| **Throughput** | 2,500bps | 14bps | 120bps | 215bps | 215.9bps |

TABLE I: Compared with the latest approaches, Amphista increases throughput by more than 400x, 46x, 26x, and 25x, respectively.




(a) Topology (b) Throughput (c) PRR

Fig. 15: Impact to ZigBee Network is negligible in terms of throughput and packet reception ratio.

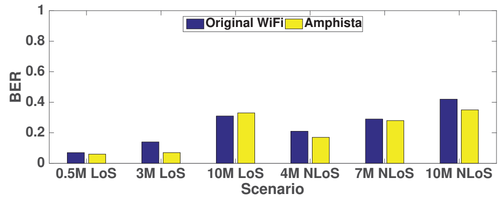



Fig. 16: Impact to WiFi to WiFi communication is negligible.

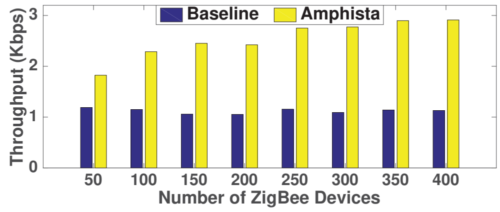



Fig. 17: Average Throughput (Uplink)

### D. Comparison with the Latest CTC Techniques

As shown in Table I, we compared the throughput of Amphista with the latest cross-technology communication (CTC) techniques under the same experimental setups. The results show that our design is up to 2 orders of magnitude better than the state-of-the-art approaches. This is because we i) adopt 4QAM-OFDM modulation scheme to increase the spectrum efficiency and ii) designed the handshaking protocol to minimize interference between WiFi and ZigBee devices.

### E. Impact to Z2Z Communication

To evaluate the impact of concurrent communication to the original ZigBee to ZigBee (Z2Z) traffic, we deployed thirteen ZigBee devices in a tree topology (shown in Figure 15(a)). The comparison results of throughput and PRR are shown in Figure 15(b) and Figure 15(c), respectively. Overall, impact of concurrent communication to the original Z2Z traffic is negligible. The largest difference occurs on link $L5$ (i.e., the link between device $Z1$ and $Z4$), where the Z2Z throughput is only reduced by 3 kbps (or 2%) when Amphista is enabled. This is because ZigBee uses direct sequence spread spectrum (DSSS) to improve protection against interference and noise.

### F. Impact to W2W Communication

We evaluated the impact of Amphista's concurrent communication to the original WiFi to WiFi (W2W) communication. Figure 16 shows the BER of WiFi to WiFi communication with and without Amphista. We can observe that the BER are under 0.5% in both scenarios. Moreover, there is negligible difference between original W2W communication and Amphista interfered W2W communication. This result demonstrates the effectiveness of our Amphista design and

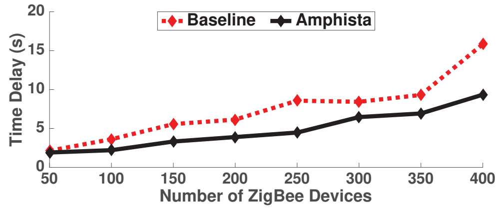



Fig. 18: Packet Delivery Delay (Downlink)

also explains why Amphista achieves higher power efficiency, spectrum efficiency, and throughput than CSMA and TDMA in Figures 9 to 12.

### G. Network Layer Evaluation

In this section, we evaluate the network layer design of Amphista. The system is deployed in a 33 $ft$ x 33 $ft$ square area. Each sensing node is deployed and randomly assigned to one of the four channels that are overlapped with the WiFi channel. The gateway is positioned in the center of the deployment field. When the system is on, each ZigBee device sends the data packets (with payload from 1 Byte to 10 Bytes) towards the gateway as uplink; while the gateway also disseminates the data (with a payload between 1KB to 3KB) from the gateway to all the ZigBee devices as downlink. The throughput of Z2W and Z2Z links follows the empirical results from Section V-C1. The **metrics** below are used to evaluate the network performance: i) **Packet delivery delay of downlink:** The aggregated throughput in the network divided by number of transmissions. ii) **Average uplink throughput:** The aggregated uplink throughput in the whole network divided by the number of uplink transmissions.

To verify the effectiveness of our forwarding protocol, we compare our design with the baseline that does not utilize the unique feature of concurrent Z2W and Z2Z communications. **Baseline:** For each ZigBee device in baseline, it has the same throughput of Z2W and Z2Z communications, however, it can only conduct either Z2W or Z2Z communication at a time.

We evaluate the scalability of the IoT network by increasing the number of nodes from 50 to 400. For the uplink throughput (shown in Figure 17), when the device number is small, our approach is similar to baseline. However, when the number increases, our approach shows an increasing trend while the baseline keeps the same. This is because when there are more nodes in the network, the conflict between uplink and downlink will be more severe for baseline approach but the Amphista approach's performance increases due to the bi-directional communications. The average uplink throughput for amphista is 173.6% higher than the baseline. Meanwhile, the time delay of downlink in our design is also 42.3%

less than the baseline (Figure 18) because less amount of retransmission is needed for Apmhista approach comparing with baseline approach.

## VI. Related Work

To utilize the coexistent features of different wireless technologies within the same frequency band, researchers have proposed different technologies [10]–[13], including the CTC techniques [7]–[9], [14]–[18] which enable direct communications between WiFi and ZigBee by modify PHY or link layers parameters. BlueBee [19] and $B^2W^2$ [20] achieve the BLE to ZigBee and BLE to WiFi communications, respectively. [21] and [22] manipulates the PHY layer symbols to communicate between WiFi and ZigBee by using customized radios. WEBee [4] enables WiFi to ZigBee communication by using WiFi signals to emulate ZigBee signals. However, WEBee cannot enable ZigBee to WiFi communication, which is one of the novelties of this paper. Therefore, WEBee is complimentary to our system.

Researchers have also proposed various techniques to improve the spectrum utilization and the performance of different wireless systems [23], [24]. Due to the increasingly crowded 2.4 GHz ISM band, significant amount of work has been conducted to improve its spectrum utilization [25]–[28]. To further improve the performance of wireless communication, researchers have proposed various interference mitigate techniques [29], [30] and collision avoidance solutions [31], [32].

Different from the above approaches that focus on physical layer design, our approach is a cross-layer design that explores the cross-technology communication's unique feature for simultaneous uplink and downlink data forwarding using a single ZigBee data stream.

## VII. Conclusion

To facilitate the edge computing with exponentially increasing number of IoT devices and the huge amount of data generated by these devices, we propose a novel design – Amphista, which can achieve simultaneous uplink and downlink communications and data forwarding with a single ZigBee data stream. Compared with existing approaches, Amphista significantly improves throughput and reduces the latency. By applying the noise cancellation and equalization techniques, our experiments also demonstrate that Amphista has a negligible impact on the original WiFi-to-WiFi communication. Our design is compliant with WiFi and ZigBee standards. It can be deployed on commodity ZigBee devices to achieve simultaneous uplink and downlink communications and data forwarding with negligible impact to the on-going WiFi traffic.

## Acknowledgment

This project is supported by NSF grants CNS-1652669, CNS-1824491, and CNS-1539047.

## References

[1] "http://www.gartner.com/newsroom/id/3598917."
[2] Thyaga Nandagopal, "Wireless Research at the NSF: Current Investments and Emerging Priorities," 2016.
[3] Cisco Systems, "Cisco Global Cloud Index: Forecast and Methodology, 2014-2019 White Paper."
[4] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *MobiCom*, 2017.
[5] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wirel. Netw.*, vol. 11, no. 4, pp. 419–434, Jul. 2005.
[6] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *MobiCom 2015*.
[7] 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 *INFOCOM*, 2016.
[8] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: Cross-technology communication with transparent morse coding," in *INFOCOM 2017*.
[9] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "Zigfi : Harnessing channel state information for cross-technology communication," in *INFOCOM'18*.
[10] 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 Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1379–1392, June 2014.
[11] T. Jin, G. Noubir, and B. Sheng, "Wizi-cloud: Application-transparent dual zigbee-wifi radios for low power internet access," in *INFOCOM, 2011*.
[12] K. Chebrolu and A. Dhekne, "Esense: Communication through energy sensing," in *MobiCom, 2009*.
[13] H. Sun, Z. Fang, Q. Liu, Z. Lu, and T. Zhu, "Enabling lte and wifi coexisting in 5ghz for efficient spectrum utilization," *J. Comput. Netw. Commun. 2017*.
[14] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *INFOCOM 2017*.
[15] X. Guo, X. Zheng, and Y. He, "Wizig: Cross-technology energy communication over a noisy channel," in *INFOCOM 2017*.
[16] Y. Chae, S. Wang, and S. M. Kim, "Exploiting wifi guard band for safeguarded zigbee," in *SenSys '18*.
[17] S. Wang, S. M. Kim, and T. He, "Symbol-level cross-technology communication via payload encoding," in *ICDCS '18*.
[18] W. Wang, T. Xie, X. Liu, and T. Zhu, "Ect: Exploiting cross-technology concurrent transmission for reducing packet delivery delay in iot networks," in *INFOCOM'18*.
[19] W. Jiang, Z. Yin, R. Liu, Z. Li, S. M. Kim, and T. He, "Bluebee: A 10,000x faster cross-technology communication via phy emulation," in *Sensys*, 2016.
[20] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2w2: N-way parallel communication for iot devices," in *Sensys, 2016*.
[21] Z. Chi, Y. Li, Y. Yao, and T. Zhu, "Pmc: Parallel multi-protocol communication to heterogeneous iot radios within a single wifi channel," in *ICNP '17*.
[22] Y. Li, Z. Chi, X. Liu, and T. Zhu, "Chiron: Concurrent high throughput communication for iot devices," in *MobiSys '18*.
[23] D. Halperin, S. Kandula, J. Padhye, P. Bahl, and D. Wetherall, "Augmenting data center networks with multi-gigabit wireless links," in *SIGCOMM, 2011*.
[24] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, "White space networking with wi-fi like connectivity," in *SIGCOMM, 2009*.
[25] S. Yun, D. Kim, and L. Qiu, "Fine-grained spectrum adaptation in wifi networks," in *MobiCom, 2013*.
[26] J. Zhang, H. Shen, K. Tan, R. Chandra, Y. Zhang, and Q. Zhang, "Frame retransmissions considered harmful: Improving spectrum efficiency using micro-acks," in *MobiCom, 2012*.
[27] S. Kumar, D. Cifuentes, S. Gollakota, and D. Katabi, "Bringing cross-layer mimo to today's wireless lans," in *SIGCOMM, 2013*.
[28] K. Chintalapudi, B. Radunovic, V. Balan, M. Buettener, S. Yerramalli, V. Navda, and R. Ramjee, "Wifi-nc: Wifi over narrow channels," in *NSDI, 2012*.
[29] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan, "Understanding and mitigating the impact of rf interference on 802.11 networks," in *SIGCOMM, 2007*.
[30] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding wifi localization," in *MobiSys, 2013*.
[31] S. Sen, R. R. Choudhury, and S. Nelakuditi, "Csma/cn: Carrier sense multiple access with collision notification," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, pp. 544–556, Apr. 2012.
[32] T. Nandagopal, T.-E. Kim, X. Gao, and V. Bharghavan, "Achieving mac layer fairness in wireless packet networks," in *MobiCom 2010*.