

WiCop: Engineering WiFi Temporal White-Spaces for Safe Operations of Wireless Personal Area Networks in Medical Applications

Yufei Wang*, *Student Member, IEEE*, Qixin Wang*, *Member, IEEE*, Guanbo Zheng†, *Student Member, IEEE*, Zheng Zeng‡, Rong Zheng§, *Senior Member, IEEE*, Qian Zhang¶, *Fellow, IEEE*

Abstract—ZigBee and other wireless technologies operating in the (2.4GHz) ISM band are being applied in *Wireless Personal Area Networks (WPAN)* for many medical applications. However, these low duty cycle, low power, and low data rate medical WPANs suffer from WiFi co-channel interferences. WiFi interference can lead to longer latency and higher packet losses in WPANs, which can be particularly harmful to safety-critical applications with stringent temporal requirements, such as *ElectroCardioGraphy (ECG)*. This paper exploits the *Clear Channel Assessment (CCA)* mechanism in WiFi devices and proposes a novel policing framework, **WiCop**, that can effectively control the temporal white-spaces between WiFi transmissions. Such temporal white-spaces can be utilized for delivering low duty cycle WPAN traffic. We have implemented and validated WiCop on *SORA*, a software defined radio platform. Experimental results show that with the assistance of the proposed WiCop policing schemes, the packet reception rate of a ZigBee-based WPAN can increase by up to 116% in the presence of a heavy WiFi interferer. A case study on the medical application of WPAN ECG monitoring demonstrates that WiCop can bound ECG signal distortion within 2% even under heavy WiFi interference. An analytical framework is devised to model the CCA behavior of WiFi interferers and the performance of WPANs under WiFi interference with or without WiCop protection. The analytical results are corroborated by experiments.

Keywords—software defined radio, coexistence

I. INTRODUCTION

Wireless Personal Area Networks (WPAN) plays a key role in future e-health [1]. For example, one important WPAN application is multi-parameter monitoring, where multiple vital signs of a patient are monitored continuously. These vital signs are sampled by the sensors mounted on the patient, and displayed on a central monitor. Traditionally, sensors are wirely connected to the central monitor. Wirely connections limit the mobility of patients, and if sensors fall off due to patients' movements, or if people trip over wires, accidents may happen. To mitigate these problems, WPANs are proposed to connect the many sensors, monitors, and other medical devices wirelessly. In wireless multi-parameter monitoring, the

sensors and the monitor form a single-hop wireless network with the monitor acting as a base-station and sensors as clients.

WPANs can be built upon various candidate wireless technologies operating in different *Radio Frequency (RF)* bands. For example, the IEEE is now considering traditional *Wireless Medical Telemetry Service (WMTS)* band, *Industrial Scientific and Medical (ISM)* 2.4GHz band, *Ultra Wide Bandwidth (UWB)* band etc. Among these RF bands, the 2.4GHz ISM band is the most attractive due to its license-free nature, and consequently a wide range of available devices and vendors. Among the technologies in the 2.4GHz ISM band, ZigBee, Bluetooth, and even part of the IEEE 802.15.6 standard (that overlaps with the functionalities of WPAN) suit WPANs the best due to their low power consumption, low radiation, and low cost [1]. However, all of them may suffer from coexistence interference from WiFi (aka IEEE 802.11) networks, which also run on the same ISM 2.4GHz band. In fact, due to the low power nature of other main-stream 2.4GHz ISM band technologies (ZigBee, Bluetooth, IEEE 802.15.6 2.4GHz standard etc.), and the nowadays ubiquitous presence of WiFi networks, WiFi stands out as the major threat to 2.4GHz ISM band WPAN coexistence reliability [2][3][4][5][6][7][8][9].

For instance, Liang et al. [3] experimentally shows the performance degradation of a ZigBee link under WiFi interference. In their experiments, the *Packet Reception Rate (PRR)* of Zigbee drops below 20% when the ZigBee receiver is 15ft away from an IEEE 802.11g interferer. This indicates that WiFi interference poses a significant threat to the performance of ZigBee-based WPANs.

Though the coexistence interference may not be a major concern for low duty-cycle non-critical applications such as body temperature monitoring [10], it is not the case for WPAN applications with stringent requirements on packet delivery ratio and/or latency. One example is *Electrocardiography (ECG)* monitoring [11]. The IEEE 1073 [12] standard mandates that each ECG sample be delivered within 500ms [11]. A sample delivered after its 500ms deadline is considered lost, which means a fault happens.

To deal with the WPAN-WiFi coexistence challenge, three categories of solutions have been proposed. The first category of solutions aim to operate WPAN over RF channels sufficiently away from the active WiFi RF channels [10]. However, such solution does not deal with cases where the ISM band is fully occupied (e.g., when there are two active non-overlapping IEEE 802.11n RF channels in a same location). The second

* Dept. of Computing, The Hong Kong Polytechnic University

† Dept. of Electrical Engineering, University of Houston

‡ Dept. of Computer Science, University of Illinois at Urbana-Champaign

§ Dr. Rong Zheng is currently with Dept. of Computer Science, University of Houston; her major contribution to this paper was made when she was a visiting associate professor in Dept. of Computing, the Hong Kong Polytechnic University.

¶ Dept. of Computer Science and Engineering, The Hong Kong University of Science and Technology

Email: csqwang@comp.polyu.edu.hk

category of solutions modify the current WPAN or WiFi standards, adding intelligent coexistence schemes to make WPAN or WiFi devices more aware of each other [3][7]. However, the need to modify existing standards/implementations does not address cases where *Commercially-Off-The-Shelf* (COTS) devices are used, or cases where interferers are non-cooperative. The third category of solutions try to spatially separate WPANs from WiFi networks via careful configuration-time planning. However, this does not deal with the case where WiFi networks are not under the same administration domain as WPANs. Furthermore, unintended usage of mobile WiFi devices may still cause spurious outages in WPANs¹.

In this paper, we assume the medical WPAN has a centralized polling topology. The base station is a heavy-weight expensive medical device (such as multi-parameter monitor, surgical robot etc.) that can be equipped with software-defined radio. Under such assumptions, we propose WiCop, a novel policing framework different from the aforementioned three categories of solutions. WiCop addresses the WPAN-WiFi coexistence problem by effectively controlling the temporal white-spaces (gaps) between consecutive WiFi transmissions. Though temporal white-spaces are abundant in light to medium loaded WiFi networks [3], they are scarce in heavy loaded WiFi networks and tend to be irregular. Our approach “engineers” the intervals and lengths of WiFi temporal white-spaces, and utilizes them to deliver low duty cycle medical WPAN traffic with minimum impacts on WiFi. WiCop exploits the *Clear Channel Assessment* (CCA) mechanisms in the WiFi standard. Two policing schemes are proposed: i) Fake-PHY-Header and ii) DSSS-Nulling. We have implemented and validated WiCop on SORA, a software defined radio platform. Experiments show that under WiFi interference, WiCop can improve WPAN packet delivery rates by up to 116%.

The rest of this paper is organized as follows. Section II briefly introduces WiFi (IEEE 802.11) standard. Section III presents a case study showing the significance of WiFi co-channel interference on WPAN, using ECG monitoring as the medical application background. Section IV proposes the WiCop policing framework to engineer WiFi interference traffic’s temporal white-spaces for WPAN communications. Section V analyzes the performance of different policing strategies. Section VI evaluates our WiCop framework through experiments. Section VII discusses related work. Section VIII concludes the paper.

II. BACKGROUND

Before delving into the details of WiCop, we shall first inspect the common features of all the WiFi subtype standards that are critical to our WiCop strategies.

Common Packet Formats: Due to backward compatibility considerations, all subtypes of WiFi running in 2.4GHz ISM band recognize the IEEE 802.11 1Mbps packet format, which is one of the basic data rates of 802.11b.

Viewing from the *Physical Layer* (PHY), we can abstract an IEEE 802.11 1Mbps packet as four consecutive segments (see

Fig. 1): preamble, *Start Frame Delimiter* (SFD), PHY header, and PHY payload².

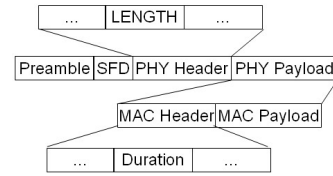


Fig. 1. IEEE 802.11 1Mbps PHY packet format.

The preamble is for receiver carrier acquisition, made up of 128 consecutive ‘1’s.

SFD is a 16-bit field indicating the subsequent PHY header.

The 48-bit PHY header contains several fields that carry control/management information. What is important is the LENGTH field, a 16-bit unsigned integer indicating the number of microseconds required to transmit the PHY payload. This implies a maximum of $2^{16} = 65535\mu s$ can be reserved for PHY payload.

The PHY payload usually consists of MAC header and MAC payload. These two parts have variable length. For example, an *Ready To Send* (RTS) packet has a 160-bit MAC header and has no MAC payload. The RTS packet has a *Duration* field in MAC header to claim a sequence of WiFi transmissions, lasting up to $32767\mu s$.

Common Receiver Diagram: Due to backward compatibility considerations, all subtypes of WiFi should have a compatible receiver to decode 802.11 1Mbps DSSS signal. The receiver diagram is shown by Fig. 2. First, *RX Filter* retrieves chips from raw samples. Second, *slicer* detects bit timing by picking the max energy. Third, *demode* retrieves one bit from every 11 chips. Fourth, *decode* is responsible for searching and processing preamble, PHY header and MAC header.

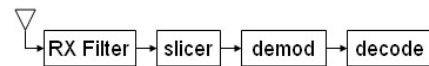


Fig. 2. diagram on receiving and decoding 802.11 1Mbps DSSS signal

Clear Channel Assessment (CCA): All subtypes of WiFi carry out *Carrier Sense Multiple Access* (CSMA) MAC protocol. According to CSMA, an IEEE 802.11 node shall always listen to the wireless medium before transmission. Only when the wireless medium is idle will the node start transmitting. The procedure, assessing the medium status (idle or busy), is called *Clear Channel Assessment* (CCA). CCA is performed in a slotted system. In every slot, WiFi PHY should report CCA status to WiFi MAC.

There are three types of CCA: *Energy Detection* (ED) only, *Carrier Sense* (CS) only, and ED+CS (the combination of ED and CS). ED-only CCA measures the wireless medium spectral power level; if it is greater than a threshold, the wireless

¹Repeated probe requests have been reported on certain WiFi devices when they are not associated with particular APs.

²which correspond to *Physical Layer Convergence Protocol* (PLCP) SYNC bits, SFD, PLCP header, and *MAC Protocol Data Unit* (MPDU) respectively according to the standard jargon [13].

medium is considered busy. CS-only CCA tries to capture WiFi preambles; if a preamble is successfully captured, the wireless medium is considered busy. Usually, CS-only CCA also looks into the content of the PHY header immediately following the captured preamble (if there is one) to provide more accurate CCA evaluations. ED+CS CCA does both. In practice, most WiFi devices support CS-only CCA or ED+CS CCA [14][13].

III. A CASE STUDY ON ECG MONITORING

In this section, we study the performance of a ZigBee WPAN for (emulated) ECG monitoring under WiFi interference, so as to empirically show the necessity of addressing the WPAN-WiFi coexistence problem.

A. Experiment Setup

Fig. 3 shows the layout of the experiment. The emulated ECG monitoring WPAN consists of one base station and one emulated ECG sensor, implemented using two TMote Sky nodes (aka *motes*, a well-known ZigBee device) [15]. In Fig. 3, the base station is denoted by *Mote-B*, and the emulated ECG sensor is denoted by *Mote-C*; the distance between *Mote-B* and *Mote-C* is d_2 . The transmission power of *Mote-B* and *Mote-C* is set to the maximum: 0dBm. *Host-Z* is a laptop connected with *Mote-B* through USB for data collection. *Host-I* is the WiFi interferer, implemented by a Linux laptop with Intel Pro/Wireless 3945ABG WiFi chip (the associated WiFi Access Point is unnecessary to appear in Fig 3). *Host-I* sends 802.11g packets (to WiFi Access Point), using *iperf 2.0.4*. *iperf* generates UDP packets at constant rate (in our case, 27Mbps). Other *iperf* parameters use the default values. The transmission power of *Host-I* is 30mW, a typical value adopted in practice [5]. The distance from *Host-I* to *Mote-B* and *Mote-C* are both d_1 .

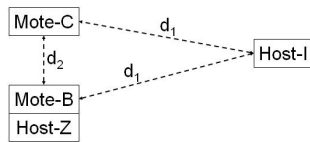


Fig. 3. Experiment Layout

Mote-C emulates an ECG sensor by sending out pre-recorded ECG samples (for narrative simplicity, we will refer to *Mote-C* simply as the “ECG sensor” in the following). Upon reception of ECG samples from the ECG sensor, the ECG base station reconstructs the ECG signal. The sampling rate of ECG signal is 250Hz, a typical value for ECG monitoring [16]; and each sample is 8-bit. The ECG sensor (*Mote-C*) sends the base station (*Mote-B*) one packet every 100ms. Hence each packet contains $250\text{Hz} \times 100\text{ms} = 25$ new ECG samples, which we call an *ECG sample chunk*. In addition, to increase reliability, the ECG sensor (*Mote-C*) buffers the two immediate previous ECG sample chunks, which are sent together with the new chunk in the same packet. Therefore, each packet contains 3 ECG sample chunks, i.e., $25 \times 3 = 75$ ECG samples; and every ECG sample is transmitted 3 times. At the typical ZigBee raw

bit rate of 250kbps, the transmission time cost of each packet is less than 4ms.

B. Performance Metric

To evaluate the performance of ECG monitoring under WiFi interference, we consider three metrics. The first metric is *Packet Reception Rate* (PRR), defined as the probability that a packet is successfully received.

Let $T_{polling}$ denote the ECG packet transmission period ($T_{polling} = 100\text{ms}$ in our case study). As mentioned before, ECG samples are only transmitted in the grouping of ECG sample chunks; and each ECG sample chunk is retransmitted $N_{re} = 3$ times within $T_{polling} \times N_{re} = 300\text{ms}$ (which is within the typical ECG sample delivery deadline [11]). An ECG sample chunk is lost iff it fails all its N_{re} retransmissions. A chunk loss is defined as a failure.

With the definition of failure, we introduce the second metric, *Mean Time To Failure* (MTTF), which is the expected duration between two ECG sample chunk losses. MTTF is given by (see Section V-F for detail):

$$MTTF = T_{polling} / (PER^{N_{re}}), \quad (1)$$

where $PER \stackrel{def}{=} 1 - PRR$.

The third metric is *Mean Time To Recovery* (MTTR), which is the expected duration of failures. MTTR is equal to (see Section V-F for the derivation):

$$MTTR = T_{polling} / PRR. \quad (2)$$

C. Experiment Results and Observations

With the layout in Fig. 3, we let *Host-I* transmit at an application layer rate of 27Mbps to the WiFi AP to emulate WiFi interference.

We set d_2 to 4ft. As the distance from *Host-I* to *Mote-B* (i.e., d_1) changes from 12ft to 4ft, the PRR decreases from 98% to 67% (see Fig. 4). At 67% PRR, the MTTF is around 2.8s. In other words, on average every 2.8s, an ECG sample chunk may be lost, which is a serious problem. The MTTR performance shows a similar trend. As the distance from *Host-I* to *Host-B* changes from 12ft to 4ft, MTTR increases 15% (see Fig. 5).

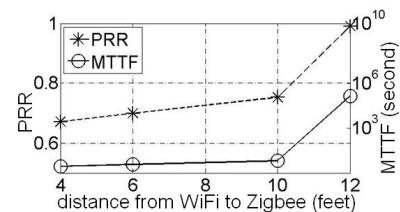


Fig. 4. PRR and MTTF of ECG monitoring WPAN under 802.11g interference

IV. POLICING FRAMEWORK

A. Framework Overview

The case study in Section III identifies WiFi interference as an eminent threat to WPAN reliability. This is consistent with the conclusions of the literature on 2.4GHz ISM band WPAN

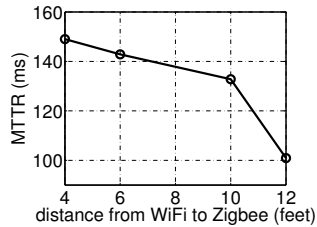


Fig. 5. MTTR of WPAN under 802.11g interference

coexistence [3] [7] [5]. In fact, due to the low power nature of other main-stream 2.4GHz ISM band technologies (ZigBee, Bluetooth, IEEE 802.15.6 etc.), and the ubiquitous presence of WiFi networks, WiFi stands out as the major threat to 2.4GHz ISM band WPAN reliability. This motivates us to devise a policing framework, called “WiCop”, to curb the WiFi threat.

As mentioned in Section I, we take into consideration the following three requirements when designing WiCop. First, WiCop shall refrain WiFi devices from transmitting at proper time, leaving temporal white-spaces for WPAN to communicate. Second, WiCop shall require no changes to COTS WiFi devices, nor COTS WPAN devices. Third, to allow cross layer design, and to achieve high adaptability, WiCop policing node shall reside upon *Programmable Wireless Interface*, such as *Software Defined Radio (SDR)*.

Furthermore, we assume that the medical WPAN adopts a centralized topology: with one base-station and multiple wireless clients. The base-station is an expensive/heavy-weight node. It has high computational power and can afford programmable wireless interfaces, such as SDR/WiCop. In contrast, the clients are cheap/light-weight WPAN devices, such as COTS ZigBee electrodes. Such assumption fits many medical WPAN applications, such as ECG monitoring, robotic surgery etc., where the expensive/heavy-weight ECG monitor or surgical robot can also serve as the base-station and run SDR/WiCop, while the patient only wears cheap/light-weight COTS ZigBee sensors.

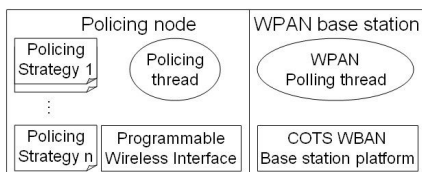


Fig. 6. WiCop policing framework architecture: the policing node and the WPAN base station can reside in a same host, or two separate but synchronized hosts

The WiCop framework architecture is illustrated in Fig. 6. The architecture involves two core entities: the policing node and the WPAN base station. The centerpiece of the policing node is the *WiCop policing thread* running upon a programmable wireless interface, e.g., SDR platform. The centerpiece of the WPAN base station is the *WPAN polling thread* running upon a COTS WPAN base station platform, e.g., TMote Sky [15]. This polling thread periodically polls remote WPAN client electrode(s)/actuator(s) for data/actuation. As mentioned in Section III, we call the corresponding period

the *WPAN polling period*, denoted as $T_{polling}$.

The policing node and the WPAN base station shall reside in a same host, or two well synchronized hosts. At the beginning of each WPAN polling period, the policing thread would first load a specific policing strategy, which will be further explained in Section IV-B. When the policing strategy is active, the policing thread triggers the WPAN polling thread to start polling the WPAN (for this specific WPAN polling period).

We call the temporal interval for a WPAN base station to finish one round of polling the *WPAN active interval*. Each WPAN polling period consists of one WPAN active interval, and one *WPAN idle interval*. Usually, the WPAN polling period is much longer than the WPAN active interval, leaving enough idle time for WiFi or other coexisting wireless schemes.

With all the above concepts in mind, we now proceed to propose various policing strategies.

B. WiCop Policing Strategies

The basic idea of our proposed WiCop policing strategies is to exploit the WiFi *Clear Channel Assessment (CCA)* mechanisms: by sending engineered WiFi compliant signals, we can properly control WiFi transmissions.

1) Strategy I: Fake-PHY-Header:

Policing Signal: According to WiFi CCA specifications, when another WiFi device detects the preamble/SFD and decodes the subsequent PHY header (See Fig. 1), it will refrain from transmitting for a number of microseconds depending on the received LENGTH field in PHY header. Therefore, the LENGTH field plays the role of reserving wireless medium access for its WiFi packet; and we can use LENGTH field to reserve wireless medium for WPAN transmission.

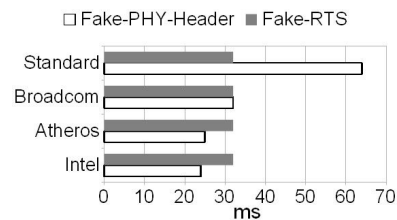


Fig. 7. Maximum duration a WiFi device mutes upon receiving a Fake PHY Header policing packet and a Fake RTS policing packet (please see Section IV-B3) respectively

As the LENGTH field is a 16-bit unsigned integer, in theory, a maximum of $65535\mu s$ can be reserved for the corresponding WiFi packet. However, our measurements show that the actual maximum duration that can be reserved is vendor dependent, as shown in Fig. 7. Fortunately, Fig. 7 also show all WiFi devices from major vendors can mute for at least 24ms. This is enough for reserving temporal white-spaces for typical WPAN communications. For example, in ECG WPAN monitoring, with each WPAN packet containing 75 8-bit samples, a WPAN only needs no more than 4ms to send a packet from the ECG sensor to the base station.

MAC Protocol: We propose to exploit the aforementioned LENGTH field to administrate coexisting WiFi transmissions. To do this, the WiCop policing node and the WPAN base station must coordinate in accessing the wireless medium, as explained by Fig. 8(a).

According to Fig. 8(a), each WPAN polling period starts with the policing node broadcasting a so called *Fake-PHY-Header policing signal*: a fake WiFi packet with only preamble, SFD and PHY header. Although this fake WiFi packet does not have PHY payload segment, the LENGTH field of its PHY header claims a packet duration equivalent to the temporal length of the WPAN active interval (hence “fake”). Immediately following this Fake-PHY-Header policing signal, the WPAN active interval starts, during which the WPAN base station polls its client(s).

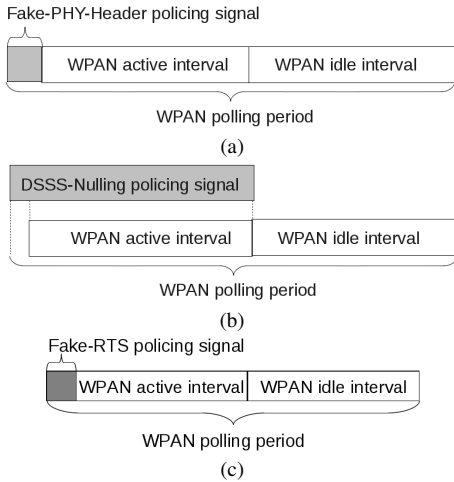


Fig. 8. Temporal domain schemes: (a) Fake-PHY-Header policing; (b) DSSS-Nulling policing; (c) Fake-RTS policing

On hearing the Fake-PHY-Header policing signal, a WiFi interferer will remain silent for the following WPAN active interval, creating a temporal white-space for WPAN to communicate.

2) Strategy II: DSSS-Nulling:

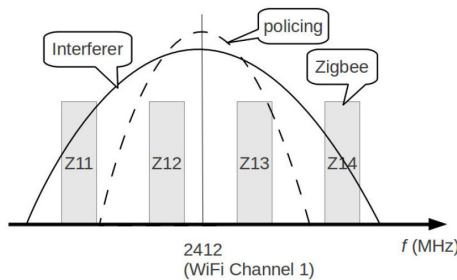


Fig. 9. Power Spectral Density (PSD) of interferer, policing, and ZigBee

Policing Signal: It is well-known that continuously sending repeated WiFi preambles can jam other WiFi devices’ transmissions [14][17]. Since WiFi preamble is a DSSS modulated signal, we call the continuous sending of repeated WiFi preamble “DSSS-Jamming”. We intend to use DSSS-Jamming as

another means of policing. However, DSSS-Jamming not only jams WiFi devices, it also jams co-channel WPAN devices. To solve this problem, we shape the DSSS-Jamming signal with a band-pass filter to generate the desired policing signal. We call the resulting policing signal *DSSS-Nulling policing signal* (i.e., the sides of the DSSS-Jamming signal spectrum are “nulled” to create spaces for WPAN signals), and the corresponding policing scheme the *DSSS-Nulling* policing.

Fig. 9 compares the *Power Spectral Density* (PSD) of DSSS-Nulling signal, WiFi signal, and ZigBee signal. When a DSSS-Nulling signal is present, a WiFi device thinks the carrier is busy and backs off. In contrast, as DSSS-Nulling signal does not occupy ZigBee channel Z11 and Z14, ZigBee communications are still possible.

In our prototype implementation, the band-pass filter to reshape DSSS-Jamming signal is realized via a baseband raised cosine *Finite Impulse Response* (FIR) filter, which results in a DSSS-Nulling signal bandwidth of 8MHz (in comparison, WiFi signal bandwidth is 22MHz). MATLAB simulations show that the side lobe of this filter is -55dB (Fig. 10). In other words, we reduce the interference power to WPANs by 55dB.

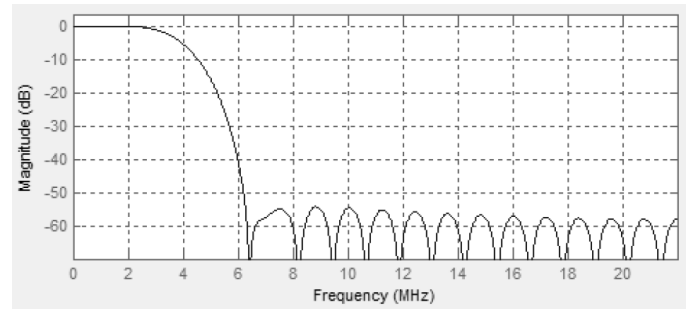


Fig. 10. Frequency response of the FIR that reshapes DSSS-Jamming signal into DSSS-Nulling signal (baseband equivalent spectrum)

Alternatively, one can use other forms of noise signal (e.g., simply a sine wave) in the WiFi band to jam/pollce WiFi transmission. However, as DSSS-Nulling signal carries repeated WiFi preamble information (though distorted by the band-pass filter), it can more effectively jam WiFi devices that support CS-only or ED+CS CCA. According to Tanenbaum and Wetherall [18], DSSS-Nulling signal can use at least 20dB less power than any other forms of noise in jamming an ED+CS CCA WiFi device.

MAC Protocol: Like the Fake-PHY-Header policing, DSSS-Nulling policing assumes that the WPAN runs centralized polling and the policing node resides on the same host as (or is synchronized to) the WPAN base station. But instead of *preceding* each WPAN active interval, the DSSS-Nulling policing signal persists throughout the WPAN active interval as shown by Fig. 8(b).

3) Strategy III: Fake-RTS: ³

Policing Signal: Similar to Fake-PHY-Header, we can extend the policing strategy to MAC layer. Instead of using a fake LENGTH field of PHY header, we transmit an IEEE 802.11 1Mbps *Request-To-Send* (RTS) frame [13]. Similar to the LENGTH field of PHY header, the RTS frame has a “Duration” field to claim that a sequence of WiFi packet-exchange is starting, which will last up to $32767\mu\text{s}$. Most COTS WiFi devices respect RTS frames (see Fig. 7). On receiving such an RTS frame, these WiFi devices will remain silent for the claimed duration. However, like Fake-PHY-Header policing, the RTS claim is fake: no subsequent WiFi packet-exchange will actually happen. The WiCop policing node will instead use the claimed duration as the WPAN active interval. We henceforth call this fake RTS frame the *Fake-RTS policing signal*, and this policing strategy *Fake-RTS* policing.

MAC Protocol: Similar to Fake-PHY-Header, the temporal view of Fake-RTS policing is shown in Fig. 8(c).

C. Qualitative Comparisons of Policing Strategies

The key differences among the aforementioned three policing strategies are summarized as follows

Policing Strategy	Fake-PHY-Header	DSSS-Nulling	Fake-RTS
CCA Compatibility	CS-Only, ED+CS	CS-Only, ED+CS, ED-Only	CS-Only, ED+CS
Success Rate	High	Highest	High
Temporal-Spectral Overhead	Lowest	Large	Low
Platform Requirement	high	high	low

Clearly, every policing strategy has its pros and cons.

CCA Compatibility: DSSS-Nulling is the most general. It works with all WiFi devices, no matter they support CS-Only, ED-Only, or ED+CS CCA. In contrast, Fake-PHY-Header and Fake-RTS policing both requires the interfering WiFi devices support CS based CCA. Fortunately, most main-stream WiFi adaptors nowadays support CS based CCA [14] [3], and hence ensure Fake-PHY-Header and Fake-RTS’s viability.

Success Rate: All three policing strategies have high success rate in suppressing interfering WiFi transmissions (see Section VI) when wireless channel quality is good.

Under poor wireless channel quality, however, DSSS-Nulling has the highest success rate in suppressing interfering WiFi transmissions. This is because DSSS-Nulling policing retransmits IEEE 802.11 1Mbps preambles throughout the WPAN active interval. The retransmissions enhance reception. In contrast, Fake-PHY-Header and Fake-RTS have no retransmission mechanisms to improve reception.

³It is brought to our attention recently that Hou et al. [6] is in fact the first to propose the Fake-RTS policing strategy (in the form of fake CTS to be exact), though we proposed the strategy independently. Nevertheless, we are the first to implement this strategy on an SDR platform; and by exploiting the flexibility of SDR, we integrate this strategy as one of the runtime alternatives in a holistic framework. We are also the first to compare this strategy with other strategies in the context of medical applications.

Temporal-Spectral Overhead: We define overhead ratio with

$$\rho = \frac{\text{Time-Spectrum Overhead}}{\text{Time-Spectrum Reserved for WPAN}},$$

and the ratios of each policing strategies are defined as follows.

In each WPAN polling period, there only needs to be one Fake-PHY-Header broadcast, which occupies 22MHz of spectrum (the standard WiFi PHY preamble/header spectrum bandwidth) and 0.2ms ⁴. Such a broadcast allows 4 ZigBee channels to communicate throughout one WPAN active interval. Therefore, the overhead ratio of Fake-PHY-Header policing is

$$\rho_{fph} = \frac{22 \times 0.2}{4B_z \times T_{act}} = \frac{1.1}{B_z T_{act}}, \quad (3)$$

where B_z (MHz) is the bandwidth of a Zigbee channel, and T_{act} (ms) is the length of WPAN active interval.

Similarly, the overhead ratio of Fake-RTS policing is

$$\rho_{fr} = \frac{22 \times 0.4}{4B_z \times T_{act}} = \frac{2.2}{B_z T_{act}}, \quad (4)$$

based on the fact that a fake RTS packet takes 0.4ms ⁵.

Suppose the effective DSSS-Nulling policing signal needs 8MHz of spectrum ⁶; and must persist throughout the WPAN active interval. This implies a DSSS-Nulling policing signal can only help reserve two Zigbee channels throughout the WPAN active interval. Therefore, the overhead ratio of DSSS-Nulling policing is

$$\rho_{dn} = \frac{8 \times T_{act}}{2B_z \times T_{act}} = \frac{4}{B_z}. \quad (5)$$

As T_{act} is usually $4\text{ms} \sim 40\text{ms}$, Eq. (3), (4), and (5) imply Fake-PHY-Header and Fake-RTS incur much lower overhead ratio than DSSS-Nulling, given that the policing is successful.

The overhead ratio of Fake-RTS policing is a little higher than that of Fake-PHY-Header, as Fake-RTS frame contains a MAC header in addition to the PHY header.

Platform Requirement: Both Fake-PHY-Header and DSSS-Nulling requires SDR platform; while Fake-RTS only requires commercial WiFi adaptor with soft MAC function [6].

D. Impact to WiFi

WiCop does little harm to WiFi transmission due to the following reasons.

First, WiCop carries out ED CCA (see Section II) before transmitting policing signals. This guarantees WiCop policing signal does not preempt existing WiFi transmissions⁷. Furthermore, both the Fake-PHY-Header and the DSSS-Nulling policing signal follow WiFi preamble/header formats. Therefore, from WiFi devices’ perspective, a WiCop policing node behaves just like another WiFi device.

⁴The more exact duration of a Fake-PHY-Header policing frame is 0.192ms , assuming IEEE 802.11 1Mbps DSSS modulation and long preamble [13].

⁵The more exact duration of a Fake RTS policing signal is 0.352ms , assuming IEEE 802.11 1Mbps DSSS modulation and long preamble [13].

⁶Note that the best bandwidth of DSSS-Nulling signal is out of the scope of this paper.

⁷As a WiFi packet typically lasts less than 1ms [13], the incurred backoff of WiCop policing signal has little impact on WPAN performance, as the typical medical WPAN polling period is $\geq 100\text{ms}$.

Second, medical WPAN traffic is typically of low duty-cycle and low workload [19]. For example, the WPAN polling period for ECG monitoring is typically 100ms; and during this 100ms, only 5ms is for WPAN traffic (and under WiCop policing). The remaining 95ms interval can be used for WiFi communications.

V. PERFORMANCE ANALYSIS

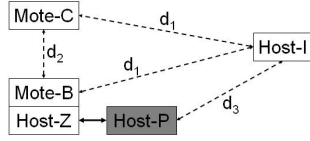


Fig. 11. Experiment Layout with Policing Node

In this section, we analyze the performance of ZigBee network under WiFi interference with or without policing strategies. For consistency, the similar layout and setup in Section III-A are adopted, with the inclusion of one policing node. The new layout with policing node is depicted in Fig. 11, with the shaded *Host-P* as the policing node.

We assume WiFi interferer performs CS-CCA, so WiFi interferer cannot detect ZigBee transmissions. Similar assumption has been made in [7]. For ease of analysis, the packet inter-arrival time and packet duration of WiFi interferer are assumed to be constant $1/\lambda$ and $1/\mu$ respectively, where $1/\lambda > 1/\mu$. More sophisticated stochastic models can be devised under general distributions but omitted in the paper, as the objective of the analysis is just to gain insights on average performance.

A polling based MAC protocol is adopted in ZigBee networks with a polling interval $T_{polling}$. At the beginning of each polling interval, the WPAN base station broadcasts a beacon containing a transmission schedule (for guaranteed access). Upon receiving this beacon, clients upload their respective data one by one in a batch. We suppose the downlink (from base station to clients) is free of error since the WPAN base station usually has larger transmit power, while the uplink (from clients to the WPAN base station) is susceptible to WiFi interference. We denote the duration of transmitting a ZigBee (uplink) packet as T_{pkt} . According to our configurations, $T_{pkt} > 1/\lambda > 1/\mu$, which is a common scenario in practice.

In the analysis, policing signals are encoded according to 802.11b 1Mbps DSSS mode. Moreover, ED-CCA is used before channel access. Thus, we assume policing signals do not preempt existing WiFi transmissions.

The rest of this section is organized as follows. First, we give the PRR of a ZigBee WPAN under WiFi interference without policing. Second, we inspect how the preamble of policing signal delays WiFi transmissions. Next, we analyze the PRR of ZigBee WPAN under WiFi interference with the three policing strategies respectively. Finally, the analytical form for WPAN's MTTF and MTTR is derived.

A. PRR with No Policing

The PRR of ZigBee WPAN under WiFi interference can be mainly attributed to two factors: the *Bit Error Rate* (BER)

under WiFi interference, and the number of ZigBee bits interfered. For simplicity, BER in absence of WiFi interference is assumed to be 0.

Since the WiFi transmission bandwidth (denoted as B_w) is much larger than the bandwidth of ZigBee (denoted as B_z), a WiFi interferer can be viewed as a white noise source in the pass band of ZigBee [8][2]. Let $P_{tx}^z, P_{tx}^w, P_{rx}^z, P_{rx}^w$ be the transmitted signal power and received signal power of the ZigBee transmitter and WiFi interferer (the received signal power from WiFi corresponds to the energy in the pass band of ZigBee) respectively. Let distance from the ZigBee WPAN base-station and the ZigBee client be d_2 and the distance from the WiFi interferer to the ZigBee base-station/client be d_1 (See Fig. 11). The BER can be modeled by [19]

$$BER_z = \frac{8}{15} \frac{1}{16} \sum_{k=2}^{16} (-1)^k \binom{16}{k} e^{20 \times SINR \times (\frac{1}{k} - 1)}, \quad (6)$$

where the *SINR* is *Signal Interference Noise Ratio* and $SINR \approx P_{rx}^z / P_{rx}^w$ (ignoring noise).

For typical indoor environment⁸, the large-scale path loss α along a distance of d can be modeled as (suggested by IEEE 802.15.4 Standard [19])

$$\alpha(d)(dB) = 40.2 + 20 \log_{10} d. \quad (7)$$

Note the model of Eq. (7) may be very optimistic in many scenarios (as it may imply LOS); however, we still adopt this model as it is widely used as the basis of performance analysis [20][19][5][8][9].

Therefore, with $\alpha(d)$, we have $P_{rx}^z = P_{tx}^z / 10^{\alpha(d_2)/10}$, and $P_{rx}^w = \frac{B_z}{B_w} P_{tx}^w 10^{\alpha(d_1)/10}$.

Once we get the value of BER_z , we can calculate the PRR of ZigBee under WiFi interference with

$$PRR_{np} = (1 - BER_z)^{n_{col}}, \quad (8)$$

where n_{col} is the average number of ‘‘corrupted’’ bits, which can be regarded as

$$n_{col} = \frac{\lambda T_{pkt}}{\mu T_{bit}}, \quad (9)$$

where T_{bit} is bit duration of ZigBee.

B. WiFi Interferer Random Backoff during Preamble of Policing Signals

All the policing signals consist of preamble(s). As mentioned before, a Fake-PHY-Header (or a Fake-RTS) policing signal starts with a preamble; while a DSSS-Nulling policing signal is made of repeated preambles. In this sub-section, we study how WiFi interferer behave during the preamble of policing signals.

In this sub-section, we assume WiFi interferer always has backlogged packets during the whole period of the policing signal transmission. This makes our analysis pessimistic on the WPAN side. Before the transmission starts, WiFi interferer follows a *random backoff* procedure [13]. This procedure, performed according to a temporally slotted system, where

⁸We suppose the propagation distance $d < 8m$

each slot is called a *Random Backoff slot* (RB-slot) and of duration $\tau_{slot} = 20\mu s$, is described as follows.

In each RB-slot, a WiFi device carries out a CCA based *Random Backoff Counter Decrement Decision Logic* (RBCDDL), which returns “yes” or ”no”. When a WiFi transmitter has a packet to transmit, it first initializes its random backoff counter n_b to $n_{b0} = 1 + cw$, where cw is an integer drawn according to uniform distribution over interval $[0, CW]$ (typically $CW = 7$) [13]. The decrement of n_b depends on the per-RB-slot RBCDDL decision: decrement by 1 on “yes”, and remain unchanged on “no”.

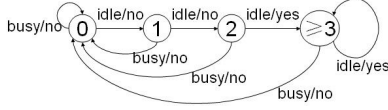


Fig. 12. Markov Chain on RBCDDL Behavior. “0” is the initial state.

The behavior of RBCDDL follows the *discrete time Markov chain* (simplified as “Markov chain” in the following) of Fig. 12. The input (“idle”, “busy”) is the results of CCA during the corresponding RB-slot. *Let us focus on the duration when WiCop policing preamble exists on the wireless medium.* Because WiCop policing preamble is a DSSS scrambled pseudo white-noise and its duration is much longer than an RB-slot duration τ_{slot} , we can reasonably assume the probability that CCA reports “busy” in an RB-slot to be a constant P_{cca} (we will drive P_{cca} in the end of this subsection). Therefore, the probability that RBCDDL reports x times of “yes” during n_a continuous RB-slots is

$$q(x, n_a) = \binom{n_a}{x} P_{yes}^x (1 - P_{yes})^{n_a - x}, \quad (10)$$

where P_{yes} is the probability that RBCDDL reports “yes” in one RB-slot. By analysing the Markov chain of Fig. 12, we have

$$P_{yes} = (1 - P_{cca})^3. \quad (11)$$

Now let us derive P_{cca} . Above all, we need briefly introduce the CCA mechanism [13].

WiFi PHY layer measures and reports CCA every RB-slot τ_{slot} . Typically, for a 802.11 1Mbps DSSS compatible WiFi receiver, $\tau_{slot} = 20\mu s$.

Conceptually, we shall regard the WiFi receiver carries out CCA and RF demodulation in parallel. The CCA works according to the automaton A_{cca} described in Fig. 13. A_{cca} has two states: “rx_idle” and “rx_busy”. Whenever the RF demodulation circuit acquires a WiFi packet’s preamble and successfully demodulates the subsequent SFD, a “SFD detected” event is triggered. The RF demodulation circuit then goes on to demodulate the WiFi packet. When the packet demodulation is fully completed or aborted due to check sum errors, a “WiFi packet reception ended” event is triggered. Correspondingly automaton A_{cca} is switched between the “rx_idle” and “rx_busy” states.

When A_{cca} is in “rx_idle”, in every RB-slot (each lasts for $\tau_{slot} = 20\mu s$), if the demodulator circuit decodes 8 consecutive bits of ‘1’s in the first $15\mu s$ (which corresponds to 15 bit-time

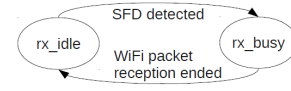


Fig. 13. CCA Automaton A_{cca} . The initial state is “rx_idle”.

of demodulation), a “busy” CCA decision is made; otherwise an “idle” CCA decision is made.

When A_{cca} is in “rx_busy”, however, in every RB-slot, a “busy” CCA decision is always made.

Therefore, when a WiCop policing node is broadcasting preamble (which consists of continuous bits of ‘1’s), the probability that a WiFi interferer reports CCA “busy” in an RB-slot is

$$\begin{aligned} P_{cca} &= \sum_{k_1=8}^{14} 2(1 - BER_w)^{k_1} BER_w \\ &+ \sum_{k_2=8}^{13} (14 - k_2) BER_w^2 (1 - BER_w)^{k_2} \\ &+ (1 - BER_w)^{15}, \end{aligned} \quad (12)$$

where BER_w is the bit error rate for the WiFi interferer’s demodulation. According to [19][13],

$$BER_w = Q\left(11 \times \frac{2P_{rx}^p}{N_0 B_w}\right)^{\frac{1}{2}}, \quad (13)$$

where $N_0/2$ (W/Hz) is the noise power spectral density [21], and P_{rx}^p is the received policing signal power. The calculation of P_{rx}^p is similar to that of P_{rx}^z (see Section V-A).

C. PRR with Fake-PHY-Header Policing

To determine the PRR with Fake-PHY-Header policing, we first derive P_{fph} , the probability that the WiFi interferer successfully decodes the Fake-PHY-Header policing frame.

Let t_0 be the time instance when the WiCop policing node starts transmitting a Fake-PHY-Header policing frame. Because the policing node carries out CCA before transmitting, we can assume at t_0 the WiFi interferer is not transmitting. On the other hand, as Section V-B, we still pessimistically assume the WiFi interferer is always backlogged during the whole period of the Fake-PHY-Header policing frame transmission. Hence at t_0 , the WiFi interferer has a positive random backoff counter value $n_{b0} = x$, where x is uniformly distributed over $\{1, 2, \dots, CW + 1\}$.

To successfully decode the Fake-PHY-Header policing frame, the WiFi interferer must first maintain its random backoff counter n_b above 0 in the first 6 RB-slots (which corresponds to the first 120 bits of the Fake-PHY-Header policing frame preamble [13]) after t_0 . This probability is $[1 - \sum_{x=1}^6 \frac{q(x,6)}{CW+1}]$. Then the WiFi interferer must correctly decode (just getting “yes” decisions from RBCDDL is no longer enough) the remaining 72 bits of the Fake-PHY-Header (the last 8 bits of preamble, plus 16-bit SFD, plus 48-bit PHY header), this corresponds to a probability of $(1 - BER_w)^{72}$. Therefore, the probability that a WiFi interferer successfully decodes the Fake-PHY-Header policing frame is

$$P_{fph} = [1 - \sum_{x=1}^6 \frac{q(x,6)}{CW+1}](1 - BER_w)^{72}. \quad (14)$$

This implies that the PRR of ZigBee under WiFi interference with Fake-PHY-Header policing is

$$PRR_{fph} = P_{fph} + (1 - P_{fph})PRR_{np}. \quad (15)$$

D. PRR with Fake-RTS Policing

Similarly, to decode a Fake-RTS policing frame, the WiFi interferer needs to decode an extra 160 bit MAC header (See Section II), compared to Fake-PHY-Header policing frame. Therefore, the success probability to detect and decode the Fake-RTS policing frame is $P_{fr} = P_{fph}(1 - BER_w)^{160}$. Thus, the PRR of ZigBee under WiFi interference and Fake-RTS policing is given by

$$PRR_{fr} = P_{fr} + (1 - P_{fr})PRR_{np}. \quad (16)$$

E. PRR with DSSS-Nulling Policing

The effect of DSSS-Nulling on the WiFi interferer is different from the other policing strategies in two aspects. First, DSSS-Nulling is transmitted persistently along with the ZigBee transmission. Second, the DSSS-Nulling policing signal is band-pass filtered.

Let us inspect how the repeated preamble (persistently along ZigBee transmissions) delays the WiFi transmission.

First, because each WPAN polling period ends with a long WPAN idle interval for WiFi interferer to transmit, we can assume the WiFi interferer's backlog by the beginning of the next WPAN polling period is very low (depleted or nearly depleted). Under the assumption of constant WiFi inter-arrival time $1/\lambda$, WiFi packet duration $1/\mu$, and ZigBee packet duration $T_{pkt} > 1/\lambda > 1/\mu$, we can pessimistically assume during each WPAN polling period, throughout the transmission duration of the κ th ($\kappa = 1, 2, \dots$) ZigBee packet, the WiFi interferer has at the most $N_c = \lceil \lambda \kappa T_{pkt} \rceil$ packets to transmit.

We further pessimistically assume that to transmit each of the N_c WiFi interferer packets, the random backoff counter is always initialized to $n_{b0} = 1$, the minimum possible value (hence the most intense interference threat to ZigBee); and that each WiFi interferer packet transmission collides with $N_B = \lceil \frac{1}{\mu T_{bit}} \rceil$ bits of the ZigBee packet, where T_{bit} is the duration of a ZigBee bit.

With the above pessimistic assumptions, we obtain a lower bound of PRR of ZigBee WPAN under WiFi interference with DSSS-Nulling policing signal as

$$PRR_{dn} \geq 1 - \sum_{x=1}^{N_c} (q(x, N_s)(1 - (1 - BER_z)^{xN_B})), \quad (17)$$

where $N_s = \lceil T_{pkt}/\tau_{slot} \rceil$; and BER_z is the bit error rate of ZigBee under WiFi interference (see Eq. (6)).

Another factor about the performance of DSSS-Nulling policing is the band pass filter used to shape DSSS-Nulling policing signal. Now let us prove that the impact from the band pass filter is minor.

When a WiFi interferer is receiving DSSS-Nulling signal, the reduced bandwidth of the policing signal (due to band pass filter) only affects the output of the RX filter (See Fig. 2).

Thus, we first derive the output of the RX filter upon receiving a DSSS-Nulling policing signal.

For normal WiFi signal, we define the Fourier transform of the chip signal is $kG_c(f)$, where $k = \pm 1$. The transfer function of a perfect RX filter is $H_{opt}(f) = G_c^*(f)exp(-j2\pi fT_c)$, where $G_c^*(f)$ is the complex conjugate of $G_c(f)$, and T_c is the chip duration [21]. The Fourier transform of the RX filter output is

$$\begin{aligned} G_o^{normal}(f) &= H_{opt}(f)kG_c(f) \\ &= k|G_c(f)|^2exp(-j2\pi fT_c). \end{aligned} \quad (18)$$

Then, the output of the RX filter at time $t = T_c$ is

$$\begin{aligned} g_o^{normal}(T_c) &= \int_{-\infty}^{\infty} G_o^{normal}(f)exp(j2\pi fT_c)df \\ &= k \int_{-\infty}^{\infty} |G_c(f)|^2df \\ &= kE_c, \end{aligned} \quad (19)$$

where E_c is also known as the chip energy.

For DSSS-Nulling signal, we denote the Fourier transform of DSSS-Nulling policing chip signal as $kG_c(f)H_x(f)$, where $H_x(f)$ is the transfer function of the band-pass filter. The Fourier transform of the RX filter output is thus,

$$\begin{aligned} G_o(f) &= H_{opt}(f)kG_c(f)H_x(f) \\ &= k|G_c(f)|^2H_x(f)exp(-j2\pi fT_c). \end{aligned} \quad (20)$$

Then, the output of RX filter at time $t = T_c$ is,

$$\begin{aligned} g_o(T_c) &= \int_{-\infty}^{\infty} G_o(f)exp(j2\pi fT_c)df \\ &= k \int_{-\infty}^{\infty} |G_c(f)|^2H_x(f)df \end{aligned} \quad (21)$$

We suppose $H_x(f)$ is an ideal rectangular filter, such that

$$H_x(f) = \begin{cases} A & -f_x \leq f \leq f_x < f_{cut} \\ 0 & \text{otherwise} \end{cases}, \quad (22)$$

where A is a constant, f_x is the cut off frequency of $H_x(f)$, and f_{cut} is the cut off frequency of H_{opt} . Therefore, the key observation is that the band pass filter only reduces the chip energy at the output of the RX filter by a constant factor A_x ($0 < A_x < 1$) such that $g_o(T_c) = kA_xE_c$.

To counter the negative effect of A_x , we can properly tune A , such that $A_x = 1$. In our analysis, we suppose $A_x = 1$. This implies that at the output of the RX filter, a WiFi receiver can not differentiate a DSSS-Nulling signal from a regular 802.11 frame. Therefore, we still use BER_w in (13) to denote the BER for WiFi interferer to decode DSSS-Nulling signal.

F. MTTF and MTTR of WPAN

With the above ZigBee packet reception rates PRR at hand, we can calculate the WPAN performance metric of MTTF and MTTR (see Section III-B for their definitions).

According to the description of Section III-B, assuming i.i.d. ZigBee packet losses, Markov chain of Fig. 14 describes the state of a ZigBee client after each of its uplink packet transmission. In this Markov chain, each state is labeled by a number, which is the current number of continuous ZigBee packet transmission failures of the ZigBee client (i.e., start

from current time and look back, how many ZigBee packet transmissions have continuously failed; note each transmission success resets this number to 0).

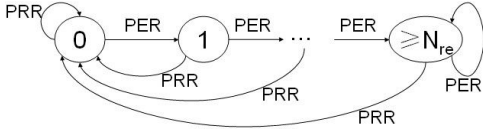


Fig. 14. Markov chain of WPAN state: each state indicates the current number of continuous ZigBee uplink packet transmission failures; initial state is “0”.

According to the description of Section III-B, a WPAN failure is defined as the loss of a data chunk after its N_{re} ZigBee uplink packet (re)transmissions. Therefore, a WPAN failure happens every time the Markov chain of Fig. 14 enters state “ $\geq N_{re}$ ”. The Markov chain takes one input every WPAN polling period $T_{polling}$, therefore, the WPAN’s *Mean Time To Failure* (MTTF) is

$$MTTF = \frac{T_{polling}}{\pi_{N_{re}}} = \frac{T_{polling}}{PER^{N_{re}}}, \quad (23)$$

where $\pi_{N_{re}}$ is the stable probability of state “ $\geq N_{re}$ ” in Fig. 14’s Markov Chain.

To obtain *Mean Time To Recover* (MTTR), we define $P_f(k)$ as the probability that a WPAN failure lasts $kT_{polling}$ ($k = 1, 2, \dots$) since it starts. This probability can be represented by $P_f(k) = PRR \times PER^{k-1}$. With $P_f(k)$, we can calculate MTTR by

$$MTTR = \sum_{k=1}^{\infty} P_f(k)kT_{polling} = \frac{T_{polling}}{PRR}. \quad (24)$$

VI. EXPERIMENTS

We implemented the WiCop policing node upon *Microsoft Research Software Radio* (SORA) [22] platform (for interested readers, a video demo is available on YouTube [23]). The SORA platform consists of the following hardware: a desktop computer (denoted as *Host-P* in Fig. 11), a *Radio Control Board* (RCB), and a third-party radio daughter board. The radio daughter board used is USRP XCVR2450. The SORA platform software mainly consists of various modulation-demodulation modules, drivers, and the corresponding development tools. For WiCop, we implemented the aforementioned policing strategies upon SORA Soft WiFi driver v1.0 (simplified as “*SORA driver*” in the following).

A. Effects on WiFi Temporal White-Spaces

We first illustrate the impact of WiCop on WiFi temporal white-spaces. The experiment set up reuses that of Section III-A and Fig. 11. *Host-I* is the WiFi interferer, which keeps sending traffic to WiFi AP at an application data rate of 10Mbps. Three feet from *Host-I* lies *Host-P*, the WiCop policing node. *Host-P* is wired/synchronized to the WPAN

base station *Mote-B* (via *Host-Z*)⁹. The WPAN polling period is 10ms, and the WPAN active interval is less than 5ms. To protect the WPAN, the policing node broadcasts policing signals every 10ms, claiming a WPAN active interval of 5ms. This affects the WiFi interference traffic, which is recorded by *Host-M*, the host of WiFi AP (the WiFi interference traffic destination).

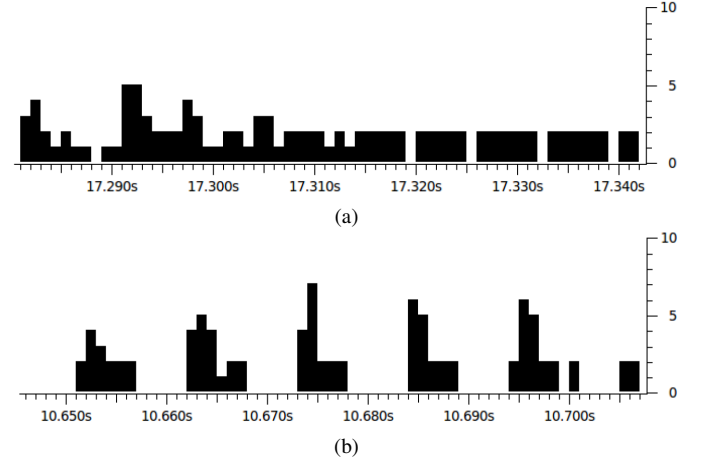


Fig. 15. (a) WiFi interference traffic when there is no policing; (b) WiFi interference traffic when there is policing. The X axis is time (unit: second); the Y axis is the number of WiFi interference traffic packets received in each 1ms time slot. In case of (b), WiCop sends a Fake-PHY-Header policing packet every 10ms to claim 5ms of WPAN active interval.

Fig. 15 shows two typical WiFi traffic traces (collected with sniffer), one generated under no WiCop policing, and the other generated under WiCop policing (the specific policing strategy used in this example is Fake-PHY-Header).

Under no policing, there are few WiFi temporal white-spaces wide enough to allow the 5ms WPAN active intervals (see Fig. 15(a)). In contrast, under policing, WiFi temporal white-spaces of no less than 5ms wide emerge every 10ms, enough to allow periodical WPAN communications.

We next illustrate the effectiveness of Fake-PHY-Header, DSSS-Nulling, and Fake-RTS policing. Fig. 16 compares the distributions of WiFi temporal white-space lengths under these three policing strategies. For each policing strategy, we run the aforementioned experiment for 25s, with a WPAN polling period of 25ms and WPAN active interval of 5ms. If policing is successful for every WPAN polling period, $25s/25ms = 1000$ WiFi temporal white-spaces of length $\geq 5ms$ should be created. From Fig. 16, we see that all three policing strategies result in more than 650 such temporal white-spaces. DSSS-Nulling is the most effective, creating more than 850 whitespaces with interval no less than 5ms. Note Fig. 16 also shows that there are a large number of WiFi temporal white-spaces of length less than 2.5ms. This occurs when WPAN is in its idle intervals and the WiFi interferer is transmitting continuously. When

⁹In the experiment, we wire *Host-P* to *Host-Z* via high-bandwidth Ethernet. Although ideally, *Host-P* and *Host-Z* should be a same node, so that the policing signal transmissions is immediately followed by the WPAN base station polling, we find using the high-bandwidth Ethernet to synchronize *Host-P* and *Host-Z* is also feasible. Besides, this puts our evaluations more pessimistic on the WiCop side (hence more convincing), as WiCop is using less than perfect devices.

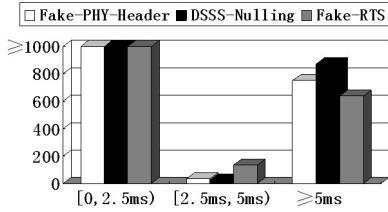


Fig. 16. Histogram showing WiFi temporal white-space distribution under Fake-PHY-Header policing (white bar), DSSS-Nulling policing (black bar), and Fake-RTS policing (grey bar) respectively. The X axis is the range of the lengths of WiFi temporal white-spaces (granularity: 2.5ms); the Y axis is the number of such WiFi temporal white-spaces encountered throughout the 25s experiment trial. Y axis is truncated at 1000 to save page space: temporal white-spaces in the 0 ~ 2.5ms range are mostly those between consecutively transmitted WiFi packets. WiCop sends a policing packet every 25ms to claim 5ms of WPAN active interval. Therefore, this graph basically shows the success rate of Fake-PHY-Header, DSSS-Nulling and Fake-RTS respectively.

WiFi is transmitting continuously, WiFi standard requires a short temporal white-space (less than 2.5ms) between every two consecutive WiFi packets.

It is also of interest to see how WiFi transmissions are negatively affected by WiCop. Fig. 17 shows the goodput of TCP and UDP connections over WiFi when there is policing. The WPAN polling period is 25ms. As the claimed length of WPAN active interval increases, the goodput decreases. However, when the claimed WPAN active interval is 5ms, the decreases in TCP/UDP goodput are both moderate. This shows that our policing strategies enable the coexistence of WiFi and WPAN.

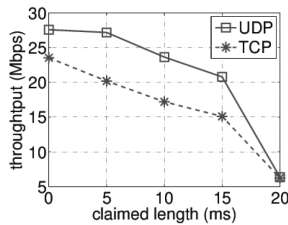


Fig. 17. WiFi goodput degradation under WiCop policing (Without loss of generality, we use Fake-PHY-Header policing strategy in this example). X axis is the claimed length of WPAN active interval; Y axis is the throughput of WiFi interference traffic. WPAN polling period is 25ms.

B. Effects on WPAN Performance

Now, we are in the position to evaluate the effects of WiCop on WPAN performance.

We reuse the experiment set up of Section III-A and the layout in Fig. 11, and deploy it in a typical indoor environment.

The WPAN is a centralized ZigBee WPAN, which runs a WPAN polling period of 100ms, and a WPAN active interval of 5ms. Both the WPAN base station and WPAN client transmits at 0dBm over a mutual distance of $d_2 = 4$ ft.

The WiFi interferer (*Host-I*) runs IEEE 802.11g and transmits at power level of 30dBm. Its distances to the WPAN base station (*Mote-B*), WPAN client (*Mote-C*), and WiCop policing node (*Host-P*) are set to 6ft, 6ft, and 3ft, respectively. The (application layer) data rate of the WiFi interferer is set to

5Mbps and 15Mbps respectively. For each of the data rate, four experiment trials are carried out, respectively corresponds to no policing, Fake-PHY-Header policing, DSSS-Nulling policing, and Fake-RTS policing. Each trial lasts 600s.

The results are summarized by Fig. 18, 19, and 20, respectively plotting the PRR, MTTF, and MTTR of the WPAN. Each of these figures also plots the theoretical predictions.

The setup of theoretical calculations is summarized as follows. First, the calculations use the same layout as the experiment. Second, as we use iperf to generate WiFi interference in experiment, we suppose the WiFi packet inter-arrival time and packet duration are constant in theoretical calculation. Thus, we use Eq. (8), (15), (16), and (17) to calculate PRR. Third, the parameters about PHY/MAC of ZigBee or WiFi strictly follow IEEE 802.15.4 or 802.11 standard. Last, all the other parameters in calculation use the same value of the parameters in experiment.

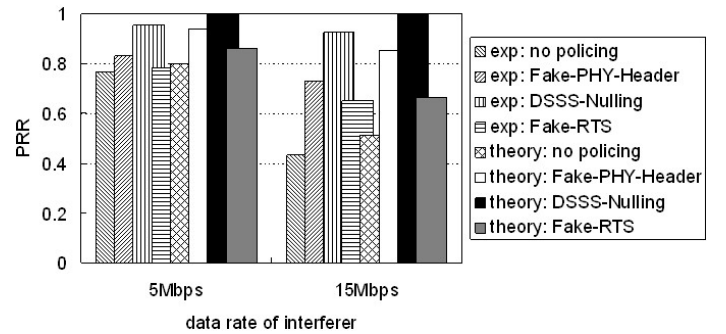


Fig. 18. WPAN PRR under different WiFi interferer data rates. Both experimental (“exp”) results and theoretical (“theory”) predictions are plotted.

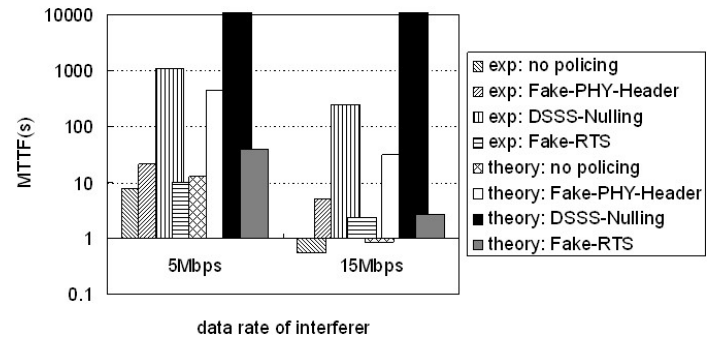


Fig. 19. WPAN MTTF under different WiFi interferer data rates. Both experimental (“exp”) results and theoretical (“theory”) predictions are plotted. As theoretical values of MTTF with DSSS-Nulling policing under 5 and 15Mbps interference are 1×10^{11} and 3.7×10^9 (seconds) respectively, we truncate Y axis at 10^4 .

These figures, no matter through experimental results or theoretical predictions, lead to a number of observations. First, under heavy WiFi interference (e.g., when the WiFi interferer’s data rate is 15Mbps), the WPAN PRR degrades significantly if there is no policing. Second, DSSS-Nulling policing performs better than Fake-PHY-Header and Fake-RTS policing in maintaining WPAN PRR under heavy WiFi interference. This is because DSSS-Nulling policing signal continuously repeats throughout the WPAN active interval; while Fake-PHY-Header (or Fake-RTS) policing signal is just broadcasted once, right

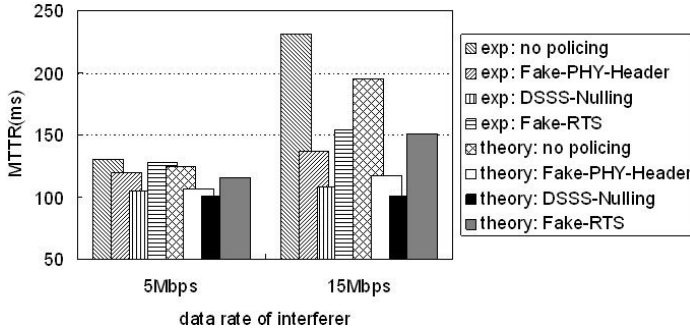


Fig. 20. WPAN MTTR under different WiFi interferer data rates. Both experimental (“exp”) results and theoretical (“theory”) predictions are plotted.

before each WPAN active interval. Third, WiCop can significantly improve WPAN performance under WiFi interference. For example, under heavy WiFi interference (15Mbps trials), experimentally, DSSS-Nulling policing can improve PRR by 116% (from 0.43 to 0.93), improve MTTF from 0.5s to 245.6s, and decrease MTTR from 232ms to 108ms. Fourth, the metric obtained by theoretical calculation is more optimistic than the same metric obtained by experiment (under the same data rate and with/without the same policing strategy). The reason is: there are other ‘hidden’ WiFi interferers around experimental environment; SORA does not have enough big power and enough good signal quality to suppress these ‘hidden’ WiFi interferers; ‘hidden’ interferers also degrade the signal quality of policing signal.

Given that the analytical framework includes several simplifying approximations, the gap between theoretical predictions and experimental results is expected. We define the PRR gap between the theoretical one (PRR_{the}) and the experimental one (PRR_{exp}), under a data rate of interferer \mathcal{R} ($\mathcal{R} = 5, 15$ Mbps) and a policing scheme \mathcal{P} (may be No Policing, Fake-PHY-Header, DSSS-Nulling, or Fake-RTS), as $\gamma_{pr}(\mathcal{R}, \mathcal{P}) = |PRR_{the}(\mathcal{R}, \mathcal{P}) - PRR_{exp}(\mathcal{R}, \mathcal{P})| / PRR_{the}(\mathcal{R}, \mathcal{P}) \times 100\%$. We find that for all the combinations of \mathcal{R} and \mathcal{P} , $\gamma_{pr}(\mathcal{R}, \mathcal{P})$ ranges from 2% to 16%.

C. A Case Study on ECG Signal Distortion

In this section, we utilize real-world ECG traces from the public medical database of PhysioNet [16] to evaluate the distortion of ECG signal.

The “gold standard” of measuring ECG signal distortion is the subjective metric of *Mean Opinion Score* (MOS) [24]: mean score given by medical professionals by comparing the original ECG trace and the reconstructed ECG trace.

Unfortunately, obtaining subjective metrics like MOS incur overwhelming workload. As a result, several objective metrics have been proposed in literature. Among these objective metrics, *Wavelet based Weighted Percentage Root mean square Difference* (WWPRD) is one of the best for two reasons. First, it quantifies the significance of ECG signal components in frequency domain. Second, it can be mapped to MOS in some range. Therefore, in our experiments, we choose WWPRD as the distortion metric.

According to Al-Fahoum et al. [24], the way to calculate WWPRD is as follows.

First, we use *Cohen-Daubechies-Feauveau* (CDF) 9/7 *Wavelet Transform* (WT)[25][24] to obtain the sub-band coefficients of the original signal and the reconstructed signal respectively. Let the coefficients of the j th sub-band of original signal be $\{c_{j,1}, c_{j,2}, \dots, c_{j,n_j}\}$, where $j = 0, 1, 2, 3, 4, 5$. Denote the coefficients of the j th sub-band of reconstructed signal as $\{\tilde{c}_{j,1}, \tilde{c}_{j,2}, \dots, \tilde{c}_{j,n_j}\}$. The *Wavelet Percentage Root mean square Difference* (WPRD) of the j th sub-band is given by

$$WPRD_j = \sqrt{\frac{\sum_{i=1}^{n_j} (c_{j,i} - \tilde{c}_{j,i})^2}{\sum_{i=1}^{n_j} c_{j,i}^2}},$$

where $c_{j,i}$ is the i th coefficient of the j th sub-band of original signal, $\tilde{c}_{j,i}$ is the i th coefficient of the j th sub-band of reconstructed signal. Last, we calculate WWPRD by

$$WWPRD = \sum_{j=0}^5 w_j \times WPRD_j,$$

where w_j is the weights of the j th sub-band. The weights are 6/27, 9/27, 7/27, 3/27, 1/27, and 1/27, respectively [24].

Clearly, the smaller value of WWPRD, the less the distortion of the received signal.

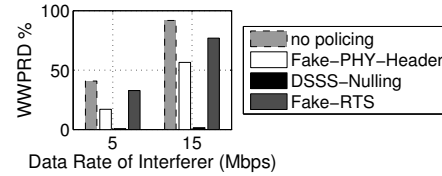


Fig. 21. WWPRD of ECG signal under different WiFi interferer data rates

In our evaluation, we overlay the real-world ECG traces from PhysioNet[16] onto the packet reception traces in Section VI-B. That is, for the experiments in Section VI-B, we emulate the ECG sensor (i.e., the WPAN client *Mote-C* in Fig. 11) readings by reading from PhysioNet ECG traces. Fig. 21 shows the WWPRD of the ECG traces received at the WPAN base station (i.e., *Mote-B* in Fig. 11). From this figure, we can make two observations: First, the WWPRD under no policing is at least 40%, which significantly exceeds the empirical acceptable limit of 15% [25]. Therefore, WiFi interference indeed distorts ECG signal. Second, policing strategies can reduce the distortion. For example, with DSSS-Nulling policing, the WWPRD is less than 2% even under heavy WiFi interference (when WiFi interferer data rate is 15Mbps).

VII. RELATED WORK

In this section, we provide a brief overview of related work pertaining to WiCop in the area of 1) WPAN (due to definition gray areas, sometimes these WPANs are also referred to as WBANs, for consistency, in the following, we still use the term “WPAN”) and WiFi co-existence, 2) Denial of Service attack (DoS) to WLANs, 3) experimental evaluation in real medical settings.

Coexistence: It is widely accepted that WiFi can severely interfere ZigBee communication [7][8][4]. Recently, many researchers found that ZigBee transmitters might impact WiFi performance under certain conditions [26][3][27]. Most of these works use packet loss rates to measure the performance of WPAN. However, in our work, applying ZigBee to delay sensitive applications, we also consider application level performance metrics, such as MTTf and MTTR.

Some researchers give analytical framework to evaluate the performance of ZigBee network under WiFi interference. Shin et al. [8] conducted numerical analysis and simulations to evaluate the PER of ZigBee communication under the interference of WiFi. Zhang et al. [28] analyzed the collision probability of WiFi and Zigbee, under two assumptions. One is that WiFi uses ED-CCA; the other is that inter-arrival time of WiFi packets is exponentially distributed. Our analytical framework gives another solution to calculate the corruption probability when collision occurs, by considering the impact from WiFi packet duration (this impact was also revealed by the experiment of Liang [3]). Further, our work is the only one comparing theoretical result and experimental result (for our best knowledge).

Some researchers propose to passively exploit the temporal or spectral white-spaces in WiFi transmission to enable coexistence of WiFi and other wireless schemes. Huang et al. [7] designed a MAC protocol to detect and use the idle time slice (temporal white-spaces) in WiFi sessions. Liang [3] proposed a mechanism to detect and estimate the temporal white-spaces in WiFi transmission and designed an MAC protocol to utilize temporal white-spaces of different lengths. Arkoulis [29] proposed a simple and efficient method to detect a single operational frequency channel that guarantees satisfactory communication. However, in some cases, white-spaces in time and frequency domain may not exist or are insufficient. WiCop, in contrast, proactively enforces temporal white-spaces on demand to support WPAN traffic.

DoS: Researchers have studied different methods to jam WiFi, such as beacon loss jamming [14], de-authentication [30], ACK corruption [31], etc.. All these works exploit the defects of current IEEE 802.11 standards. However, our work aims to provide co-existence between WLANs and WPANs. Thus, malicious attacking methods, such as jamming beacon and fake death packet, are not considered.

Experimental Evaluation in Medical Environments: Some researchers deployed wireless monitoring network in real medical units [10] [32] However, few of these works considers the interference from other wireless technologies.

Garudadri [33] applied Compressed Sensing to ECG. This approach uses the redundancy in periodic ECG trace, to mitigate distortion under high packet losses. This approach is orthogonal to WiCop and can be used in conjunction with WiCop to further improve the robustness of ECG monitoring.

Finally, it should be noted that WiCop is a general mechanism to regulate temporal white-spaces in WiFi transmissions. Though we have demonstrated its effectiveness with ZigBee-based WPANs, it can be utilized to protect WPANs based on

other wireless technologies operating in the ISM bands.

Compared to the conference version of this paper [34], we added more experiments and theoretical analysis.

VIII. CONCLUSION

To address the WPAN-WiFi coexistence challenge, we exploit WiFi's CCA mechanisms to propose WiCop policing framework, which can effectively engineer the temporal white-spaces of WiFi transmissions, reserving enough resource for WPAN communications without significantly affecting WiFi performance. To evaluate the performance of WiCop, we propose an analysis framework, giving closed-form formulae on the PRR of different policing strategies. To validate the theoretical prediction, we implemented WiCop on SORA, a software defined radio platform. Experiments show that with the assistance of the proposed WiCop policing strategies, even under heavy WiFi interference, the PRR of a ZigBee-based WPAN can increase by up to 116%. Another case study on the medical application of WPAN ECG monitoring shows WiCop can bound ECG signal distortion within 2% even under heavy WiFi interference.

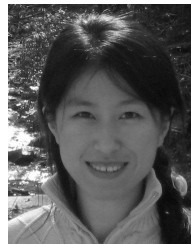
REFERENCES

- [1] M. Patel *et al.*, "Applications, challenges, and prospective in emerging body area networking technologies," *Wireless Communications, IEEE*, vol. 17, no. 1, pp. 80–88, Feb 2010.
- [2] Y. Wang *et al.*, "Evaluating the ieee 802.15.6 2.4ghz wban proposal on medical multi-parameter monitoring under wifi/bluetooth interference," *IJEHMC*, vol. 2, no. 3, pp. 48–62, 2011.
- [3] C.-J. M. Liang *et al.*, "Surviving wi-fi interference in low power zigbee networks," in *Proc. ACM SenSys*, 2010, pp. 309–322.
- [4] R. de Francisco *et al.*, "Coexistence of wban and wlan in medical environments," in *Proc. IEEE VTC*, sept. 2009, pp. 1–5.
- [5] N. Golmie *et al.*, "Performance analysis of low rate wireless technologies for medical applications," *Computer Communications*, vol. 28, no. 10, pp. 1266–1275, 2005.
- [6] J. Hou *et al.*, "Minimizing 802.11 interference on zigbee medical sensors," in *Proc. ICST BODYNETS*, 2009, pp. 5:1–5:8.
- [7] J. Huang *et al.*, "Beyond co-existence: Exploiting wifi white space for zigbee performance assurance," in *Proc. ICNP*, oct. 2010, pp. 305–314.
- [8] S. Shin *et al.*, "Lecture notes in computer science:packet error rate analysis of ieee 802.15.4 under ieee 802.11b interference," *Wired/Wireless Internet Communications*, vol. 3510, pp. 618–618, 2005.
- [9] N. Golmie *et al.*, "Interference evaluation of bluetooth and ieee 802.11b systems," *Wireless Networks*, vol. 9, pp. 201–211, 2003.
- [10] O. Chipara *et al.*, "Reliable clinical monitoring using wireless sensor networks: experiences in a step-down hospital unit," in *Proc. ACM SenSys*, 2010, pp. 155–168.
- [11] N. Chevrollier and N. Golmie, "On the use of wireless network technologies in healthcare environments," in *Proc. ASWN*, Jun. 2005, pp. 147–152.
- [12] *IEEE Standard 1073*, 1998.
- [13] *IEEE Standard 802.11*, 2007.
- [14] R. Gummedi *et al.*, "Understanding and mitigating the impact of rf interference on 802.11 networks," in *Proc. ACM SIGCOMM*, 2007, pp. 385–396.
- [15] J. Yick *et al.*, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [16] *PhysioNet*. <http://www.physionet.org>.
- [17] C. Wullems *et al.*, "A trivial denial of service attack on ieee 802.11 direct sequence spread spectrum wireless lans," in *Wireless Telecommunications Symposium, 2004*, may 2004, pp. 129–136.
- [18] A. S. Tanenbaum *et al.*, *Computer Networks*, 5th ed. Prentice Hall PTR, 2010.
- [19] *IEEE Standard 802.15.4*, 2003.
- [20] A. Goldsmith, *Wireless Communications*, 1st ed. Cambridge University Press, 2005, pp. 27–63.
- [21] S. Haykin, *Communications Systems*, 3rd ed. Wiley, 1994.

- [22] K. Tan *et al.*, "Sora: high-performance software radio using general-purpose multi-core processors," *Communications of the ACM*, vol. 54, no. 5, Jan. 2011.
- [23] *WiCop Demo*. <http://www.youtube.com/watch?v=xVy5FtTNzw8>.
- [24] A. Al-Fahoum, "Quality assessment of ecg compression techniques using a wavelet-based diagnostic measure," *IEEE TITB*, vol. 10, no. 1, pp. 182–191, Jan 2006.
- [25] M. S. Manikandan *et al.*, "Wavelet energy based diagnostic distortion measure for ecg," *Biomedical Signal Processing and Control*, vol. 2, no. 2, pp. 80–96, 2007.
- [26] S. Pollin *et al.*, "Harmful coexistence between 802.15.4 and 802.11: A measurement-based study," in *CrownCom*, May 2008, pp. 1–6.
- [27] J.-H. Hauer *et al.*, "Experimental study of the impact of wlan interference on ieee 802.15.4 body area networks," in *Lecture Notes in Computer Science*, vol. 5432, 2009, pp. 17–32.
- [28] X. Zhang *et al.*, "Enabling coexistence of heterogeneous wireless systems: case for zigbee and wifi," in *Proc. AM MobiHoc*, 2011, pp. 6:1–6:11.
- [29] S. Arkoulis *et al.*, "Cognitive radio-aided wireless sensor networks for emergency response," *Measurement Science and Technology*, vol. 21, Dec 2010.
- [30] J. Bellardo *et al.*, "802.11 denial-of-service attacks: real vulnerabilities and practical solutions," in *Proc. USENIX Security Symposium*, vol. 12, 1994.
- [31] D. J. Thunte *et al.*, "Intelligent jamming in wireless networks with applications to 802.11 b and other networks," in *IEEE MILCOM*, 2006.
- [32] J. Ko *et al.*, "Medisn: medical emergency detection in sensor networks," in *Proc. ACM SenSys*, 2008, pp. 361–362.
- [33] H. Garudadri *et al.*, "Artifacts mitigation in ambulatory ecg telemetry," in *IEEE Healthcom*, July 2010, pp. 338–344.
- [34] Y. Wang *et al.*, "Wicop: Engineering wifi temporal white-spaces for safe operations of wireless body area networks in medical applications," in *IEEE RTSS*, Nov 2011, pp. 170–179.



Guanbo Zheng received the B.E. degree in communication engineering from Northeastern University, Shenyang, China in 2004, and the M.E. degree in telecommunication engineering from Inha University, Incheon, Korea in 2008. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Houston, TX. He is a student member of the IEEE.



Zheng Zeng received the B.E. degree from the Dept. of Computer Science and Technology, Tsinghua University, Beijing, China, in 2005; and the PhD degree from the Dept. of Computer Science, University of Illinois at Urbana-Champaign in 2011. She is currently working in Apple Inc.



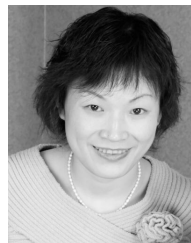
Yufei Wang received his BS in Electronics and Information System and MS in Communication and Information System, both from Nankai University, Tianjin, China. He is currently working toward a PhD degree in the Department of Computing, the Hong Kong Polytechnic University in Hong Kong since 2009. His research interests include real-time/embedded system and networking, wireless coexistence, wireless monitoring, and medical CPS. He is a student member of the IEEE and a member of IEEE Communication Society.



Rong Zheng (S'03-M'04-SM'10) received her Ph.D. degree from Dept. of Computer Science, University of Illinois at Urbana-Champaign and earned her M.E. and B.E. in Electrical Engineering from Tsinghua University, P.R. China. She is on the faculty of the Department of Computer Science, University of Houston since 2004, currently an associate professor.



Qixin Wang received the B.E. and M.E. degrees from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 1999 and 2001, respectively, and the Ph.D. degree from the Department of Computer Science, University of Illinois at Urbana-Champaign in 2008. He is currently an Assistant Professor in the Department of Computing at the Hong Kong Polytechnic University. Dr. Wang is a member of the IEEE and the ACM.



Qian Zhang received the BS, MS, and PhD degrees from Wuhan University, China, in 1994, 1996, and 1999, respectively, all in computer science. She joined the Hong Kong University of Science and Technology in 2005 and is now a professor at the Department of Computer Science and Engineering.

Dr. Zhang is a Fellow of IEEE.