Beyond Co-existence: Exploiting WiFi White Space for ZigBee Performance Assurance

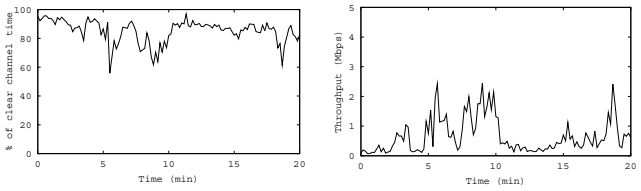
Jun Huang[1]; Guoliang Xing[1]; Gang Zhou[2]; Ruogu Zhou[1]
[1]Michigan State University, USA; [2]College of William and Mary, USA;
email:{huangjun, glxing, zhouruog}@msu.edu, gzhou@cs.wm.edu

***Abstract*—Recent years have witnessed the increasing adoption of ZigBee technology for *performance-sensitive* applications such as wireless patient monitoring in hospitals. However, operating in unlicensed ISM bands, ZigBee devices often yield unpredictable throughput and packet delivery ratio due to the interference from ever increasing WiFi hotspots in 2.4 GHz band. Our empirical results show that, although WiFi traffic contains abundant *white space*, the existing coexistence mechanisms such as CSMA are surprisingly inadequate for exploiting it. In this paper, we propose a novel approach that enables ZigBee links to achieve *assured* performance in the presence of heavy WiFi interference. First, based on statistical analysis of real-life network traces, we present a Pareto model to accurately characterize the white space in WiFi traffic. Second, we analytically model the performance of a ZigBee link in the presence of WiFi interference. Third, based on the white space model and our analysis, we develop a new ZigBee frame control protocol called WISE, which can achieve desired trade-offs between link throughput and delivery ratio. Our extensive experiments on a testbed of 802.11 netbooks and 802.15.4 TelosB motes show that, in the presence of heavy WiFi interference, WISE achieves 4x and 2x performance gains over B-MAC and a recent reliable transmission protocol, respectively, while only incurring 10.9% and 39.5% of their overhead.**

I. Introduction

Recent years have witnessed the increasing adoption of ZigBee technology for low-cost, low-power personal-area wireless networks. In particular, numerous cheap commercial off-the-shelf (COTS) ZigBee devices are being deployed for a range of *performance-sensitive* applications, such as wireless patient monitoring in hospitals and home networking for wireless headsets and game remote controllers. These applications impose stringent requirements for the underlying networks including high throughput and packet delivery ratio. For instance, wireless ECG sensors for patient monitoring must reliably report cardiac rhythm data at desired rates for real-time diagnosis. Similarly, wireless headsets and game remote controllers should achieve required bandwidth and delivery ratio for satisfactory user experience.

However, several major challenges must be addressed when ZigBee technology is applied in performance-sensitive scenarios. Operating in unlicensed ISM bands, ZigBee devices must compete for the spectrum resources with other RF devices. In particular, due to the proliferation of WiFi hotspots in 2.4 GHz band, ZigBee and WiFi devices are increasingly located in the same environment leading to interference between each other [8]. Several approaches have been proposed to promote the coexistence of ZigBee and WiFi. The most popular approach is to assign orthogonal channels to ZigBee and WiFi devices.



(a) Clear channel time. (b) Aggregated throughput.

Fig. 1. Channel utilization trace of a WiFi network consisting of 2 APs and 18 active users.

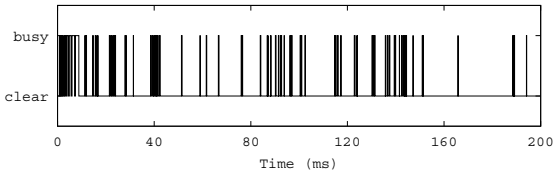


Fig. 2. WiFi channel state trace.

However, such a frequency domain solution is often infeasible as the 2.4 GHz spectrum is populated by ever increasing number of WiFi devices. Existing 802.11 b/g/n access points heavily occupy three orthogonal channels in the 2.4 GHz band, which overlap with 12 out of total 16 channels defined in 802.15.4 - the PHY/MAC specification of ZigBee. As a result, to completely avoid the interference from WiFi, ZigBee networks can only work on four channels, which significantly limits the efficiency of spectrum usage.

In this paper, we argue that there exist abundant opportunities for ZigBee and WiFi to coexist in the *same* or *overlapping* channels. Fig. 1 shows the channel utilization trace captured in a real-life 802.11-based network [2] consisting of 2 APs and 18 active users. It is clear that the channel is free in most of time. Although WiFi traffic surges from 5th to 10th minute, the channel is still free in more than 60% of time. Fig. 2 shows a typical trace of channel usage of the same WiFi network. We can see that the network traffic is highly bursty leaving significant amount of *white spaces* between 802.11 frames.

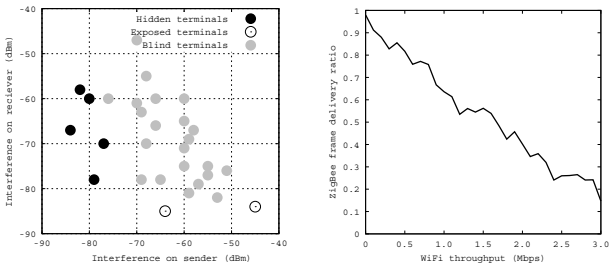
Unfortunately, our empirical results based on 802.11 netbooks and 802.15.4 TelosB motes show that the existing CSMA mechanisms are surprisingly inadequate for exploiting the prominent channel white space to enable WiFi and ZigBee coexistence. First, commodity WiFi NICs typically conduct clear channel assessment (CCA) by carrier sensing 802.11-modulated signals [8]. As a result, WiFi transmitters cannot detect ZigBee signals and hence do not defer their transmissions even when there exist ongoing ZigBee transmissions.

Moreover, even if this issue is addressed (e.g., by adopting energy-based CCA), there is still a large region in which ZigBee transmitters can sense WiFi transmitters but not vice versa because the transmit power of WiFi is much higher than that of ZigBee.

In this paper, we propose a novel approach to deal the interference between ZigBee and WiFi. Our major contributions are summarized as follows.

1) Based on an empirical study of ZigBee and WiFi coexistence, we reveal that WiFi nodes are often *blind terminals* of ZigBee nodes due to inadequate carrier sensing mechanism of 802.11 and transmit power asymmetry between ZigBee and WiFi. A WiFi blind terminal fails to detect ZigBee signals and hence can easily corrupt ongoing ZigBee packet reception, which is a major cause of poor ZigBee performance in coexisting environments.
2) We conduct extensive statistical analysis of data traces captured in real-life WiFi networks. We show that, in a channel shared by a group of 802.11 devices, WiFi frames are highly clustered and the arrival process of clusters has the feature of self-similarity. We then present a Pareto model that accurately characterizes the white space between WiFi frame clusters.
3) We propose an analytical framework that models the performance of a ZigBee link in the presence of WiFi blind terminals. Based on the white space model, we derive the expected frame collision probability and channel utilization ratio. The results will help a network designer analyze and predict the performance of ZigBee links coexisting with WiFi networks.
4) We develop a novel ZigBee frame control protocol called WISE, which can achieve desired trade-offs between link throughput and delivery ratio. WISE predicts the length of white space in WiFi traffic based on the Pareto model, and intelligently adapts frame size to maximize the throughput efficiency while achieving assured packet delivery ratio.
5) We implement WISE in TinyOS 2.x and evaluate its performance through extensive experiments using 802.11 netbooks and 802.15.4 TelosB motes. Our results demonstrate significant advantages of WISE over B-MAC - the default MAC protocol in TinyOS, and OppTx - a state-of-the-art protocol designed to utilize opportune conditions of bursty links. When there exists heavy WiFi interference, WISE achieves performance gains of 4x and 2x over B-MAC and OppTx, while only incurs 19.5% and 42.5% of their overhead.

The rest of the paper is organized as follows. Section II reviews related work. Section III discusses the blind terminal problem. Section IV presents a white space model. In Section V, we model the performance of a ZigBee link in the presence of WiFi blind terminals. Section VI presents the the frame control protocol WISE. We offer experimental results in Section VII and conclude the paper in Section VIII.



(a) Distribution of hidden terminals, exposed terminals and blind terminals. (b) ZigBee packet delivery ratio vs WiFi throughput.

Fig. 3. The blind terminal problem.

II. Related Work

Traffic modeling [13] [12] [3] is a fundamental problem in the Internet community. The self-similarity of Internet traffic has also been observed [12] [3]. Different from existing studies on Internet traffic modeling, we focus on characterizing the white space in link-level traffic of WiFi channel shared by a group of users. The most similar work to our study is [6]. Compared to our work, the empirical white space model proposed in [6] is built for specific applications, such as FTP, VOIP and Skype. However, in real scenarios with diverse applications, the traffic is highly bursty at a wide range of time scales, which is not considered in [6]. In this paper, we build the white space model based on real traffic traces, and examine the modelability of white space in different time scales.

The coexistence of heterogenous devices is a critical issue in unlicensed ISM bands. In [9], Adaptive Frequency Hopping (AFH) is proposed for Bluetooth and WiFi coexistence. AFH is further improved in [7] by sensing and predicting the WiFi behavior using the model proposed in [6]. However, these approaches are designed for frequency hopping systems, and require the support of cognitive radios for spectrum sensing. Several recent studies have been conducted to mitigate the bursty interference on low power 802.15.4 links. Srinivasan et al. [16] proposed an opportune transmission (OppTx) protocol to improve the performance of bursty 802.15.4 links. OppTx measures and quantifies the correlations in packet delivery and loss, and use them to set transmission backoff delay. However, OppTx is oblivious to the probabilistic feature of white space, and hence cannot explicitly utilize the white space in WiFi channel. Several error detection and recovery methods [18] [5] [10] [4] are proposed to utilize partial packets to improve the link reliability. These approaches typically work at the MAC layer. The frame control protocol proposed in this paper operates transparently to the MAC layer, hence can be integrated with these approaches. In our earlier work [19], system called ZiFi was developed to utilize ZigBee radio to detect the existence of WiFi hotspots based on the unique interference signatures of WiFi. However, mitigating the interference of WiFi for ZigBee devices is not addressed.

III. The Blind Terminal Problem

As shown in Fig. 1 and 2, WiFi traffic contains abundant *white space*. We ask a key question: does the existing WiFi

and ZigBee MAC layers allow the white space to be efficiently utilized? The answer to this question is crucial to the feasibility of coexisting ZigBee and WiFi networks within the same frequency domain. We show experimentally that CSMA is surprisingly ineffective in utilizing the white space. In particular, ZigBee networks coexisting with WiFi networks often suffer from significant interference due to the *blind terminal* problem. We now illustrate the blind terminal problem by a case study.

A. A Empirical Study of ZigBee and WiFi Coexistence

We deploy two TelosB motes equipped with 802.15.4 compliant CC2420 radios in an office. Both motes run the CSMA-based B-MAC [14]. ZigBee sender broadcasts at a fixed rate. A Linux netbook equipped with 802.11 compliant Intel Atheros 928x NIC serves as the interferer and is placed at different locations in the same office. We vary the position of WiFi interferer and measure the performance change of ZigBee link. 802.15.4 adopts multiple retransmissions to achieve reliable packet delivery under interference. However, this incurs extra energy consumption. To avoid the complication of retransmissions on the analysis of our results, we intentionally diabled them for ZigBee link. We note that this does not affect the conclusion of this study. The WiFi node runs a traffic generator [1] that generates a combination of UDP and TCP flows at a preset rate. Such a setting allows us to analyze the impact of interference at different traffic rates. We record a) the WiFi signal power measured at ZigBee sender and receiver from the received signal strength indicator (RSSI), b) the sending rate of ZigBee, c) the receiving rate of ZigBee, and d) the sending rate of WiFi. In addition, we can calculate the ZigBee packet delivery ratio (PDR) using b) and c).

Based on the experiment results, we can classify the role of WiFi node as hidden terminal, exposed terminal, or *blind terminal* depending on how it interferes ZigBee sender and receiver. Table I shows the condition of each role. Fig. 3(a) shows the distribution of three terminals in the space of interfering powers to the ZigBee sender (X axis) and receiver (Y axis). Each data point (x, y) in Fig. 3(a) corresponds to a different location of WiFi node whose signal strength is measured as x and y dBm by the ZigBee sender and receiver, respectively. For instance, the point (-81dBm, -59dBm) represents a hidden terminal because the experiment results measured satisfy: a) the sending rate of ZigBee does not change, and b) the PDR of ZigBee link drops. In such a scenario, the ZigBee sender cannot sense the transmissions of WiFi (due to the weak signal power of -81 dBm) while the receiver is strongly interfered (with power of -59 dBm). Similarly, we can identify (-45dBm, -84dBm) as an exposed terminal, since the sending rate of ZigBee decreases while the PDR remains high. Hidden and exposed terminals have been well studied before. However, our results also indicate the existence of blind terminals (grey points in Fig. 3(a)) where both the sending and receiving rates of ZigBee decrease.

To further study the blind terminal problem, we select one blind terminal position, and vary the traffic rate of WiFi to examine its impact on ZigBee link performance. Fig. 3(b) shows that the PDR of ZigBee link drops with the increasing traffic rate of WiFi. In addition, we observe that the sending rate of WiFi strictly follows the rate we set in the traffic generator. This result shows that the WiFi sender fails to sense the transmissions of ZigBee.

TABLE I
THE ROLE OF WIFI INTERFERER.

Hidden terminal	The WiFi node is located within the interference range of ZigBee receiver, but outside the CCA range of ZigBee sender.
Exposed terminal	The WiFi node is located within the CCA range of ZigBee sender, but outside the interference range of ZigBee receiver.
Blind terminal	The WiFi node is located within both the CCA range of ZigBee sender and the interference range of ZigBee receiver.

B. Analysis of Results

The performance degradation of ZigBee in the presence of blind WiFi terminals is mainly caused by the following two reasons.

The heterogeneous PHY layer. Commodity WiFi NICs typically conduct CCA by carrier sensing, i.e., declare channel busy only when valid 802.11-modulated signal is detected [8]. As a result, WiFi transmitters cannot sense ZigBee signals and hence do not defer their transmissions even when there exist ongoing ZigBee packet transmissions. Therefore, WiFi signals can easily corrupt the ongoing reception of ZigBee packets. Although the WiFi's blindness to ZigBee transmitters can be alleviated by adopting different carrier sensing mechanisms (e.g., energy-based CCA), unfortunately, off-the-shelf WiFi drivers do not provide such an option. Even such an option is available, adopting it for the existing WiFi deployments poses a major management challenge.

Power asymmetry. The second cause of the blind terminals is that the transmit power of 802.11 devices is much higher than ZigBee. In particular, the maximum transmit powers of WiFi and ZigBee are 14 and 0 dBm, respectively. Therefore, even if WiFi MAC layer adopted energy-based CCA, there is still a large region in which ZigBee transmitters can sense WiFi transmitters but not vice versa. As a result, the traditional approaches to dealing with interference, such as RTS/CTS exchanges, cannot effectively handle the blind terminal problem.

In this paper, we propose a novel approach to deal with the blind terminal problem, which allows coexisting ZigBee links to efficiently use WiFi white space. First, we model the WiFi white space based on data traces captured in real-life WiFi networks (see Section IV). We then analyze the performance of ZigBee under heavy WiFi interference (see Section V). Finally, in Section VI, we propose WISE - a ZigBee frame control protocol that can achieve efficient channel utilization based on the white space model.

IV. MODELING WHITE SPACE IN REAL-LIFE WIFI NETWORKS

In this section, we study how to model the temporal white space of WiFi networks. The white space model will be used

to control the frame transmissions of ZigBee in presence of WiFi blind terminals. We first conduct extensive statistical analysis on data traces captured in real-life WiFi networks. We show that, in a channel shared by a group of 802.11 devices, the arrival process of aggregate WiFi frame clusters has the feature of self-similarity. We then study in what time scale the temporal white space of WiFi is modelable. Finally we present a Pareto model that accurately characterizes the white space.

A. WiFi Frame Clustering

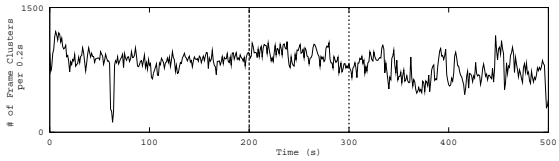
As shown in Fig. 2, the arrival of WiFi frames is highly bursty and clustered. We observe that frames are clustered together with short intervals typically less than 1 ms, while the idle periods between clusters are significantly longer. The short frame intervals are attributed to the MAC layer contention mechanism of 802.11, in which senders back off for a short random time before each transmission.

According to 802.15.4 [9], the protocol header of ZigBee frame is 17 Bytes, which are transmitted at a rate of 250 Kbps. Thus the packet-in-air time of ZigBee is at least 544 μs. After accounting for the software overhead (e.g., the delay introduced by CPU and radio interaction), the minimum packet transmission time of ZigBee approaches the maximum backoff window size of 802.11. Therefore, it is very difficult for ZigBee senders to utilize the short WiFi frame inter-arrival times for packet transmission. In the following, we will only focus on modeling the arrival process of WiFi *frame clusters* where each cluster may include multiple frames spaced by intervals less than 1 ms. We define the interval between frame clusters as *inter-cluster space* while the interval between the frames within the same cluster as *intra-cluster space*. Moreover, white space hereafter refers to inter-cluster space unless otherwise indicated.

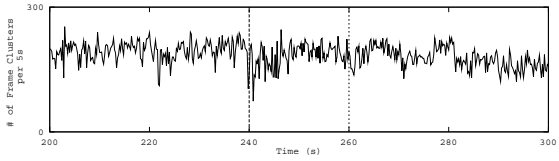
B. Self-Similarity of WiFi Frame Clusters

We plot the scaling behavior of the frame cluster arrival process in Fig.4. The data is captured in OSDI 2006 [2], which contains 150 consecutive minutes of monitored WLAN traffic in a channel shared by 2 APs and 18 active users. Fig.4 shows the number of arrived frame clusters for three time units: 5s, 1s and 0.2s. The plots show similar variance at all time scales. This time-scale invariant feature suggests that the arrival process of WiFi frame clusters is self-similar. In the following, we introduce the backgroud on self-similarity, and use statistical and graphical tools to formally test the feature of self-similarity for 802.11 frame clusters. The results will enable us to model the distribution of WiFi white space.

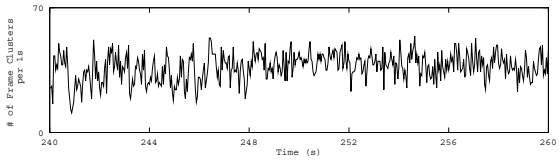
Let $X = (X_t : t = 0, ..., N)$ be a *covariance stationary stochastic* process with mean μ, and variance σ^2. Define $X^{(m)} = \{X_k^{(m)}, k \in [0, \frac{N}{m}]\}$ to be the aggregated covariance stationary time series, obtained by averaging the original series over blocks of size m. Then X is *H-self-similar* if it has the same autocorrelation function $r(k) = \mathsf{E}[(X_t - \mu)(X_{t+k} - \mu)]/\sigma^2$ as the series $X^{(m)}$ for all m [3]. This means that the variances of the series are self-similar for all m, except for the change in scale. The degree of self-similarity is expressed by the $Hurst$ parameter H, which describes the speed of decay of the series' autocorrelation function. For self-similar series, $1/2 < H < 1$. As $H \to 1$, the degree of self-similarity increases. We now use statistic and graphical tools to formally test the feature of self-similarity. These tools are described in [17], and widely used in traffic analysis literature [12] [3].



(a) # of frame clusters per 5s. The data in window (200s, 300s) is shown in (b) using the time unit of 1s.



(b) # of frame clusters per second. The data in window (240s, 260s) is shown in (c) using the time unit of 0.2s.



(c) # of frame clusters per 0.2s.

Fig. 4. Self-similarity of 802.11 frame cluster arrival process.

Rescaled range statistics (R/S method) : The R-S method is based on the fact that for a self-similar time series $X = (X_t : t = 0, ..., N)$, the $rescaled\ range$, R/S of series $X^{(m)}$ grows according to a power law with exponent H as a function of $\frac{N}{m}$. Thus for a given time series, the log-log plot of R/S against $\frac{N}{m}$ has a slop which is an estimate of the $Hurst$ parameter H. Fig.5(a) gives the R-S plot for the data trace used in Fig.4. The result shows that the asymptotic slope of R/S plot is clearly between 0.5 and 1 (lower and upper dotted lines respectively), which suggests that the WiFi frame cluster arrival is self-similar.

Periodogram-based analysis : The periodogram is an estimate of the spectral density of a given time series. For a self-similar time series, its spectral density obeys a power-law near the origin. Therefore, in a log-log plot of the power sepctrum, peridogram should be proportional to the frequency. The $Hurst$ parameter H of the time series can be estimated by $\beta = 1 - 2H$, where β is the periodogram slope. Fig.5(b) shows periodogram plot of the same trace used in Fig. 4. The slope of the regression line is $\beta = -0.75$, yielding an estimate of H as 0.87, which indicates the self-similar nature of the trace.

C. Pareto Model of WiFi White Space

As discussed in Section IV-B, in a channel shared by a group of 802.11 devices, the arrival process of WiFi frame clusters has the feature of self-similarity. According to [17], the self-similarity is a feature of arrival process with heavy-tailed or power law distributed inter-arrival time. Since the

Pareto process is one of the most widely adopted power law distributions, we chose Pareto model to fit the arrival process of WiFi frame clusters. In the following, we first give the Pareto model and then discuss the goodness-of-fit of it with respect to real WiFi data traces.

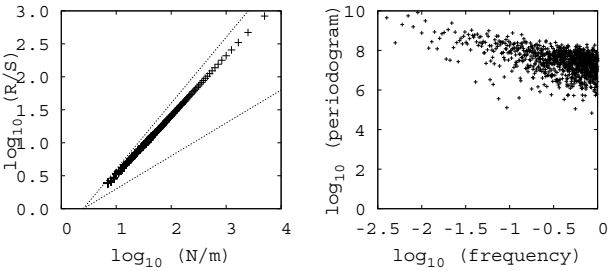
We assume the inter-arrival time of frame clusters within time window T fits Pareto model. That is, the distribution of white spaces follows i.i.d Pareto distribution, which satisfies

$$\Pr\{x > t\} = \begin{cases} (\frac{\alpha}{t})^{\beta}, & t > \alpha \\ 1, & otherwise \end{cases} \quad (1)$$

where α and β are the scale and shape of Pareto model respectively. According to the observation in Section IV-A, we set α to 1 ms. In other words, our model only accounts for the inter-cluster space that is longer than 1 ms, because shorter white spaces cannot be used by ZigBee links. In Pareto model, β is given by $\frac{\lambda}{\lambda-\alpha}$, where λ is the average inter-arrival time of frame clusters.

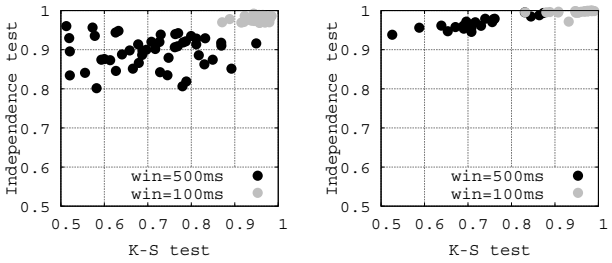
We use Kolmogorov-Smirnov Test (K-S test) of 0.95 significance level to evaluate the goodness-of-fit of the Pareto model. K-S test is a widely adopted tool to test the goodness-of-fit. We divide the time into equal sized windows. For each window, a Pareto distribution is fitted by maximum likelihood estimation. K-S test is then applied for each window to test the goodness-of-fit for the estimated Pareto distribution. If a significance level of 0.95 is used, then $0.95k$ out of total k windows should pass the test, if the white space perfectly follows the Pareto distribution. In the Pareto model, we also assume that the inter-arrival times of WiFi frame clusters are independent of each other. To test this assumption, we compute the one lag autocorrelation for each window. For a time series of n samples generated from an uncorrelated white noise process, the probability that the magnitude of the autocorrelation exceeds $1.96/\sqrt{n}$ is 0.05. Thus we compare the autocorrelation results with $1.96/\sqrt{n}$, and expect that 95% windows will give an autocorrelation value smaller than $1.96/\sqrt{n}$, thus pass the independence test.

Fig. 6 gives the results of goodness-of-fit test on the Pareto model. We conduct K-S test on two data traces which are captured in OSDI2006 [2] and SigCOMM2008 [15]. The OSDI and SigCOMM traces includes a group of trace files, which are different in the captured date and time, the monitoring channel, and the position of the traffic sniffer. For each file we check the goodness-of-fit with different window sizes. However, only the results of 100ms and 500ms are shown due to space limitation. Each trace file corresponds to two points in the figure. The x-value of the point is the percentage of windows in the corresponding data trace file that pass the K-S test, when the window size is set to a specific value. The y-value is the percentage of windows that pass the independence test. Thus we expect that, if points are clustered at the top right corner, then the fitness of the Pareto model is good. We observe that the modelability of the frame cluster arrival process varies with time scale. At a small time scale of 100ms, the arrival process can be well characterized by the Pareto model.



(a) R-S-Plot. (b) Periodogram plot.

Fig. 5. Statistical test of self-similar.



(a) OSDI trace. (b) SigCOMM trace.

Fig. 6. Goodness-of-fit tests of Pareto model for real-life WiFi traces.

V. Modeling ZigBee Link Performance

In this section, we study the impact of blind WiFi terminals on ZigBee link performance. Based on the Pareto white space model, we will derive the expected frame collision probability. The result will help network designers predict the performance of ZigBee networks in the presence of WiFi interference. Moreover, it provides foundation for optimizing the link behavior to deal with such interference (Section VI).

According to Section III, we define a WiFi blind terminal for a specific ZigBee link as follows: 1) Blind terminal has a carrier-sense based MAC layer, and is located within the CCA range of ZigBee sender, i.e., ZigBee sender will defer if the blind terminal is transmitting; and 2) Blind terminal is located within the interference range of ZigBee receiver, i.e., a ZigBee frame will be colliding with the transmitting frames from the blind terminal. We list the notation used in our analysis in Tab. II. We assume that the channel is shared by a set of k blind terminals $\mathcal{B} = \{\mathcal{B}_i, i = 1...k\}$. The channel condition of a ZigBee sender is modeled by $< \alpha, \beta, u, \omega >$, where ω is the percentage of white space, u is the channel utilization ratio of $\mathcal{B}$. α and β are parameters of the Pareto model given in Eq. (1), which characterize the distribution of the white space in the channel.

We now derive the probability of collision between ZigBee and WiFi frames. Our analysis accounts for the ZigBee carrier sensing model and the white space distribution in WiFi traffic. The main objective of our analysis is to characterize the expected performance of a ZigBee link solely based on the *transmitter's view* of channel condition. The accuracy of our analysis can be easily improved by accounting for the channel condition on the receiver. For instance, a frame may be successfully received by the receiver even when it collides

with other frames due to the capture effect [11]. Therefore, we can derive frame delivery ratio based on the probability of frame collision and the signal-to-noise ratio of receiver. However, we argue that such results are not practical because obtaining receiver channel condition requires significant messaging overhead and is not supported by the existing ZigBee MAC layers.

The CSMA of ZigBee will conduct CCA before each transmission and perform exponential backoff if the channel is busy. It will force a frame transmission if the maximum number of CCA tries is reached. A forced transmission will cause the ZigBee frame to collide with the in-air blind terminal frame. In our analysis, we ignore the possibility of forced transmissions because it occurs rarely. Therefore, we will slightly underestimate the overall collision probability. We analyze the following two cases: a) the collision probability when ZigBee transmits a frame in intra-cluster space, denoted by $\mathcal{C}_a(\tau)$, and b) the collision probability when ZigBee transmits a frame in inter-cluster space, denoted by $\mathcal{C}_b(\tau)$, where τ is the ZigBee frame in-air time.

For a randomly arrived ZigBee frame, the probability that it starts to transmit between two WiFi frames within the same frame cluster is the fraction of frame interval in the total clear channel time. It is given by $p_a = \frac{1-u-\omega}{1-u}$.
As discussed earlier, when the inter-arrival time of frames is shorter than 1 ms, the in-air time of a ZigBee frame of reasonable size will always be longer than the WiFi frame intervals, which will cause a collision between ZigBee and WiFi frames. So we have $\mathcal{C}_a(\tau) = 1$.

We now focus on the case where ZigBee sender transmits a frame in inter-cluster space. For a randomly arrived ZigBee frame, the probability that it starts to transmit in white space is the fraction of white space in the total clear channel time, which is given by $p_b = \frac{\omega}{1-u}$.
In this case, the collision probability depends on the *lifetime* and *age* of the white space upon the start of ZigBee transmission, where the *lifetime* is the time interval between two WiFi frame clusters, and the *age* is the time interval between the start of the white space to the start of the ZigBee transmission. Since the distribution of white space age is affected by the backoff process of ZigBee, we now consider two cases: a) ZigBee transmits frame after backoff, and b) ZigBee transmits frame without backoff.

The CSMA of ZigBee will perform backoff if the channel is busy upon the arrival of frame. We assume that the backoff will always align the start of ZigBee transmission with the start of white space. In this case collision occurs only when the white space lifetime is shorter than ZigBee's frame in-air time. Note that here we will underestimate the collision probability since we ignore the actual age of white space when the ZigBee senses a clear channel. However, the inaccuracy caused by this assumption is not significant due to the short backoff interval of ZigBee MACs. Denote the expected collision probability by $\mathcal{C}_b^0(\tau)$, it satisfies

$$\mathcal{C}_b^0(\tau) > \mathcal{F}(\frac{\tau}{D}) = 1 - (\frac{\alpha}{\tau})^{-\beta} \tag{2}$$

TABLE II
NOTATION USED IN WHITE SPACE MODELING.

τ	packet size of ZigBee.
H	header size of ZigBee.
M	maximum packet size of ZigBee.
D	data rate of ZigBee.
λ	average white space lifetime.
ω	fraction of channel time that is white space.
u	channel utilization ratio of WiFi.

where $\mathcal{F}$ is the CDF of Pareto distribution.

When a ZigBee frame arrives within white space, ZigBee will send the frame without backoff. In this case the white space age is uniformly distributed over the entire white space lifetime. Given that the white space lifetime is ℓ, collision occurs if the frame arrives $\ell - \tau$ later than the start of the white space. Its probability is given by $\min\{\frac{\tau}{\ell}, 1\}$. We consider an arrival process of k frame clusters $X = \{X_1, ..., X_k\}$, and denote the set of white spaces by $L = (\ell_1, ..., \ell_k)$, where ℓ_i is the time interval between X_{i-1} and X_i. The arrival time of ZigBee frame is uniformally distributed over L. The probability that the ZigBee frame falls into the ℓ_i is given by $\ell_i / \sum_{j=1}^{k} \ell_j$. Denote the expected collision probability by $\mathcal{C}_b^1(\tau)$, it is given by

$$\mathcal{C}_b^1(\tau, k) = \sum_{i=1}^{k} \left(\frac{\min(\frac{\tau}{D}, \ell_i)}{\ell_i} \times \frac{\ell_i}{\sum_{j=1}^{k} \ell_j} \right) \tag{3}$$

As the number of frame clusters ranges from 1 to ∞, we have

$$\mathcal{C}_b^1(\tau) = \frac{\int_0^\infty \min(\frac{\tau}{D}, \ell) f_P(\ell) \mathrm{d}\ell}{\int_0^\infty \ell f(\ell) \mathrm{d}\ell} = 1 - \frac{1}{\beta}(\frac{\alpha D}{\tau})^{\beta-1} \tag{4}$$

where $f(\cdot)$ is the PDF of Pareto distribution. The probability that the channel is busy upon the arrival of ZigBee frame is the fraction of WiFi transmission time, which is given by WiFi channel utilization u. Therefore the expected collision probability of frame transmission in white space is given by $\mathcal{C}_b(\tau) = u\mathcal{C}_b^0(\tau) + (1-u)\mathcal{C}_b^1(\tau)$. According to Eq. (2) and Eq. (4), we have

$$\mathcal{C}_b(\tau) > 1 - \left(\frac{\alpha D u}{\tau} + \frac{1-u}{\beta} \right) (\frac{\alpha D}{\tau})^{\beta-1} \tag{5}$$

Note that for Pareto model, $\beta > 1$. Putting all together, the overall expected collision probability $\mathcal{C}(\tau)$ is given by

$$\mathcal{C}(\tau) = p_a \mathcal{C}_a(\tau) + p_b \mathcal{C}_b(\tau) \tag{6}$$

Since β is given by $\beta = \frac{\lambda}{\lambda - \alpha}$, and $\frac{\omega}{1-u} < 1$. Taking it into Eq. (6), the overall expected collision probability satisfies:

$$\mathcal{C}(\tau) > 1 - \left(1 + (\frac{\alpha D}{\tau} - 1)u \right) \left(\frac{\alpha D}{\tau} \right)^{\frac{\alpha}{\lambda - \alpha}} \tag{7}$$

VI. WISE: White Space-Aware Frame Adaptation

As discussed in previous sections, ZigBee suffers poor PDR when its channel is shared by a closely deployed WiFi network, due to the collision with packets from blind WiFi terminals. A straightforward method to improve PDR is to use small packet size for ZigBee. However, considering the protocol overhead introduced by PHY and MAC layer headers,

the throughput efficiency achieved by small packets is very low. It is therefore desirable to achieve a balance between PDR and efficiency. To this end, we propose a frame control protocol called *WhIte Space-aware framE adaptation* (WISE) for ZigBee networks. WISE predicts the length of white space in WiFi traffic based on the Pareto model, and intelligently adapts frame size to maximize the throughput efficiency.

A. Overview of WISE

The design objective of WISE is to maximize the throughput efficiency of ZigBee while bounding the packet collision probability under user requirement. WISE consists of the following two components that reside between PHY and MAC layers. The *white space modeling* component builds the Pareto model based on maximum likelihood estimation. The *frame adaptation* component computes the size of frame that maximizes the throughput efficiency while limiting the collision probability within the user given bound.

When WISE gets a frame from the MAC layer, it may split the frame into *sub-frames* and the size of each sub-frame is determined by predicting the remaining lifetime of the white space using the Pareto model. WISE maintains a *session* for transmitting all sub-frames of a MAC frame. Each sub-frame carries a session ID and delimiters. This is necessary because a) the white space may be too short to accommodate the transmission of entire frame when the channel is heavily loaded with WiFi traffic; and b) the integrity of the MAC frame needs to be protected, so that the receiver can process the frame correctly. Such a design allows WISE to operate transparently to the MAC layer and the modification to the MAC layer is kept the minimum. The receiver will assemble all sub-frames within the same session into a integral MAC frame and pass to the MAC layer. In the following, we discuss the optimization of sub-frame size and the design of WISE.

B. Optimizing Sub-Frame Size

As blind WiFi terminals cannot sense the signal of ZigBee, collisions will occur if a ZigBee frame cannot finish its transmission before the arrival of the next WiFi frame cluster. Therefore, to reduce collision probability, the transmission time of ZigBee should be shorter than the remaining lifetime of the current WiFi white space. Let ρ be the white space age when a frame is ready for transmission. As the lifetime of white space follows the Pareto model defined in Eq. (1), we have the following conditional collision probability $\mathcal{C}(\tau, \rho)$ for a given frame size τ

$$\mathcal{C}(\tau, \rho) = \Pr\{t < \rho + \frac{\tau}{D} \mid \rho\} = 1 - (\frac{\rho}{\frac{\tau}{D} + \rho})^{\beta} \tag{8}$$

where D is the channel rate of ZigBee, α and β are the scale and shape of the Pareto model of white space.

The goal of frame adaptation of WISE is to maximize the efficiency of transmission while limiting the collision probability under user requirement. Since the size of protocol header is fixed, the transmission efficiency is a monotonic increasing function of the sub-frame size. Given a specific collision probability threshold T, the optimization problem of WISE frame adaptation can be formulated as follows:

$$\begin{aligned} \text{Maximize} \quad & \tau & (9)\\ \text{Subject to} \quad & \mathcal{C}(\tau, \rho) < T & (10)\\ & \tau \leq M & (11) \end{aligned}$$

where M is the maximum frame size of ZigBee. Solving the problem, we obtain the optimal sub-frame size:

$$\tau = \text{Min}\ \{\rho \times \gamma, M\} \tag{12}$$

where γ is given by

$$\gamma = D \times \left((1 - T)^{-\frac{\lambda - \alpha}{\lambda}} - 1\right) \tag{13}$$

where λ is the mean in the Pareto model of the white space.

C. Frame Session Management

When a frame is passed to WISE by the MAC protocol, WISE computes the sizes of sub-frames and then starts a session to transmit them. The receiver maintains the states of sub-frames in a session in order to keep the integrity of the original MAC frame. Each sub-frame is composed of a 1-byte WISE header and payload. The WISE header includes 1-bit *start session delimiter*, 1-bit *end session delimiter* and 6-bit *session ID*. We now discuss how a session is managed by WISE in details.

1) Session ID assignment: Each sub-frame in a session carries the same session ID. The session ID is assigned by the sender by randomly generating a number between 0 and 63, as WISE header uses 6-bit ID to identify each session. Note that the receiver identifies whether it is the destination of a sub-frame solely by the session ID, because a sub-frame is only part of MAC frame and hence may not include the MAC address. Two sessions on different nodes within the communication range of each other may accidentally choose the same session ID and start at the same time. However, such a possibility is low and its impact on the performance of WISE is neglectable.

2) Session initialization and sub-frame transmission: WISE sender initiates the session by transmitting a *session registration frame* (SRF), which is identified by setting 1 in the *start session delimiter* bit. The SRF must protect the integrity of the MAC layer header of the frame, so that the receiver can conduct frame pre-processing correctly. Due to the criticality of SRF, we carefully control the collision probability of SRF as follows. When the frame size derived from Eq. (12) is smaller than the sum of the PHY header, WISE header and MAC header, the transmission will be deferred by a random backoff. The sender will repeat this process until it can transmit the entire MAC layer header within one sub-frame.

When a SRF is received, WISE receiver will conduct MAC layer pre-processing on the MAC header, such as address recognition etc. If the receiver is the destination, it records the session ID in a *session table* and allocates buffer for the session. At the same time, a session lifetime timer is initiated. The session will be forcibly terminated upon the timeout, so

that the receiver will not wait too long when the last sub-frame of the session is lost. In this case, the received partial packet will be assembled and submitted to the MAC layer.

D. Implementation

WISE has been implemented in TinyOS on both TelosB motes equipped with 802.15.4 compliant CC2420 radios. We now discuss the details of implementation of several components of WISE.

1) White space sampling: We implement the channel modeling algorithm in the driver of CC2420. CC2420 radio exposes CCA and start of frame delimiter (SFD) pins to the microcontroller. When a signal above the CCA threshold is detected, the CCA pin goes low to indicate the busy channel, otherwise it goes high. Whenever there is a change of the pin state, a signal is triggered to interrupt the microcontroller. The SFD pin indicates the start of a decodable packet. It interrupts the microcontroller when a SFD (0xA7 in 802.15.4) is detected. WISE treats all undecodable signals as blind terminal interference. We note that an undecodable signal may be attributed to a 802.15.4 interferer. We will discuss how to deal with this issue in Section VI-E. To sample the white space, WISE captures all the interruptions on CCA and SFD pins. Whenever the CCA pin goes low but the SFD pin remains unchanged, an arrival of undecodable signal is detected.

2) White space modeling: WISE periodically samples the channel and measures the interval between two undecodable signals in order to build the white space model. According to Section IV, if the length of the interval is longer than 1 ms, it is considered as a sample of white space. WISE keeps a moving window for collected samples, and uses the maximum likelihood estimation to derive a Pareto model. The size of the moving window is set to 100 ms as it is shown in Section IV that the distribution of inter-frame spaces fits the Pareto model only if the time scale is shorter than 100 ms. The channel sampling frequency is a tunable parameter. In our experiments, we observe that a maximum sampling frequency of 200 Hz is high enough to ensure the accuracy of Pareto model. Since the sampling window is 100 ms, at most 20 samples need to be stored. We will evaluate the impact of the sampling frequency on modeling accuracy in Section VII.

3) Sub-frame Adaptation: To reduce the computational overhead, we adopt a discrete approach to optimizing the size of WISE sub-frames. According to Eq. (12), for a given collision probability bound, the optimal size of sub-frame depends on ρ and γ, where ρ is the white space age upon the start of channel assessment, and γ is a function of the scale of Pareto model α and the average lifetime of the white space λ. Since we set α to 1 ms, γ only depends on λ. For the purpose of computational efficiency, we discretize the time into slots of 1 ms. For a given collision bound, γ is calculated offline for each integer value of average white space lifetime λ. The results are stored in γ-table, which can be looked up online by λ. In our experiments, we observed that the impact of blind terminal on ZigBee performance is neglectable when the average lifetime of the white space is longer than 20 ms. Therefore, the storage cost of γ-table is small.

E. Impact of ZigBee Interference

As discussed earlier, WISE derives the white space Pareto model by sampling intervals between undecodable signals. However, the undecodable signal may be attributed to a 802.15.4 source that is located within the CCA range, but outside the communication range. Therefore, the 802.15.4 signal may introduce errors in the estimation of Pareto model that is originally derived for 802.11 traffic. We now discuss two solutions to this issue. However, evaluating these solutions is left for future work.

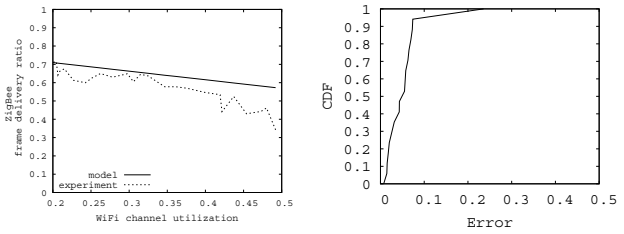
WiFi detection using PHY features. A ZigBee node may detect WiFi signals by capturing specific PHY signatures of the signal. As a result, WISE will only use samples of detected WiFi signals to build the white space model. This approach is feasible if the PHY specification of WiFi is known to the ZigBee detector. In [6], a feature based approach to 802.11 signal detection is proposed to search the preamble and SFD field of 802.11 packets. Feature-based signal detection has high accuracy, but may require nontrivial support from the radio hardware.

WiFi detection using MAC features. The MAC features of 802.11 and 802.15.4 are significantly different. The difference can be utilized by ZigBee radio to distinguish 802.11 and undecodable 802.15.4 signals. For example, the frame transmission times of 802.11 and 802.15.4 follow distinct distributions, due to the significant differences in channel rate and frame size. In addition, the interval between two back-to-back 802.11g frames is much shorter than that of 802.15.4. This is because the minimum contention window of 802.11g is 32, with a time unit of 9 μs while it is 8 for 802.15.4 with a time unit of 320 μs. Therefore, captured signals are attributed to 802.11 source with higher probability, if the intervals between them are shorter than $32 \times 9 = 288\mu$s.

VII. Evaluation

In this section, we present the evaluation results of WISE. We implemented WISE in TinyOS 2.x on TelosB motes equipped with 802.15.4 radios. We employ CSMA-based B-MAC [14] (without low power listening), which is the default MAC in TinyOS. Our implementation of WISE did not require any change in B-MAC's implementation. Unless otherwise indicated, the transmit power is set to -7 dBm, which assures good delivery performance for each ZigBee link in our setup. We use ASUS Eee netbooks equipped with 802.11 compliant Intel Atheros 928x NICs as WiFi interferers. D-ITG [1] is used to generate WiFi traffic at different rates. As a high-fidelity Internet traffic generator, D-ITG is capable of generating simultaneous flows from different protocols. Empirical results showed that D-ITG can reproduce realistic traffic patterns under a wide range of network settings [1].

Our evaluation focuses on three performance metrics: frame delivery ratio, throughput, and throughput overhead. Throughput is measured as the total number of bytes of *impactive*



(a) Experiment results vs model. (b) CDF of error.

Fig. 7. Evaluation of the performance mdoel.

payloads delivered in one second. Throughput overhead is defined as $\frac{N_t - N_d}{N_d}$, where N_t is the total number of bytes transmitted per second, and N_d is the throughput. Thus the throughput overhead quantifies the additional bytes transmitted by the sender to deliver one byte of impactive payload. We compare WISE to two baseline protocols: 1) B-MAC without WISE and 2) the opportune transmission (OppTx) protocol proposed in [16]. OppTx is a state-of-the-art low-power sensor network protocol designed to mitigate the impact of interference. It significantly improves the throughput of bursty links by transmitting back-to-back packets and controlling the backoff delay when a failure occurs [16]. For a fair comparison, the backoff delay of OppTx is always tuned to the optimal value. In contrast to WISE, OppTx is oblivious to WiFi traffic and hence cannot explicitly utilize white space in WiFi channels.

A. Accuracy of the Performance Model

In this section, we study the accuracy of the performance model proposed in Section V. We deploy two TelosB motes in an indoor environment and ensure that the PDR of the ZigBee link is above 95% without WiFi interference. To introduce blind terminal interference, we intentionally place a WiFi interferer close to the ZigBee sender and receiver. The average interference powers on ZigBee sender and receiver are -60 dBm and -66 dBm, respectively. The WiFi node generates traffic of UDP and TCP flows. We vary the traffic rate of WiFi to evaluate the impact on frame delivery ratio of ZigBee link. To measure the channel utilization of WiFi, the ZigBee sender samples the CCA pin at a frequency of 100Hz. The experiment is conducted for 10 mins. The channel utilization of WiFi is given by the portion of number of '0' samples, which indicates a busy channel. The results in Fig. 7 show that our model matches the experiment results closely. In particular, 90% of the errors are smaller than 0.1.

B. Impact of Sampling Frequency

WISE needs to periodically sample the channel for deriving the Pareto model of white space, which may pose considerable overhead for low-power 802.15.4 devices. We now study the impact of sampling frequency. The experimental setting is the same as in Section VII-A. The WiFi node generates traffic of UDP and TCP flows at 2.3 Mbps. The sampling frequency of WISE is varied from 1 to 20 samples/100ms while collision probability bound is varied from 0.1 to 0.4.

Fig. 8 shows the impact of sampling frequency on the frame delivery ratio of WISE. Since WISE uses the white space model to control the collision probability, a higher frame delivery ratio implies a better modeling accuracy. We observe that the link performance meets the given collision bounds, and the frame delivery ratio grows with the sampling frequency. However, sampling frequency only shows small impact on frame delivery ratio. For instance, when sampling frequency is increased from 1 to 20 samples/100 ms, the frame delivery ratio of WISE (under 0.4 collision bound) only changes from 0.605 to 0.717 with a difference of 0.112. Under collision bound of 0.1, the difference is only 0.046. This result implies that WISE can achieve high modeling accuracy with extremely low sampling overhead. This feature is particularly desirable for ZigBee devices due to their resource limitation.

C. Impact of Collision Probability Bound

The collision probability bound of WISE is a user specified parameter. As discussed in Section VI, WISE conducts frame adaptation to maximize the transmission efficiency, and uses the Pareto model to limit the collision probability within the given bound. In this experiment, we use the same setting with Section VII-A while varying the collision bound of WISE. Broadcast traffic is adopted to exclude the impact of retransmissions. ZigBee sender generates a traffic rate of 100 packets/s with 50-byte payload for each packet. Fig. 9 shows the sending rate and frame delivery ratio under different collision bounds. We observe that the sending rate of WISE increases with collision bound because WISE accesses the channel more aggressively. However, under a lower collision bound, WISE achieves a higher frame delivery ratio, which indicates better reliability and lower throughput overhead. Such configurability allows users to tune the performance of WISE for different trade-offs between delivery ratio and throughput.

D. Performance Comparison

We now compare the performance of WISE to that of B-MAC and OppTx under different levels of WiFi interference. The traffic rate of ZigBee is 4Kbps. Fig. 10 shows the frame delivery ratio for broadcast traffic. We observe that the frame delivery ratio of B-MAC and OppTX drops linearly with the increase of WiFi data rate, while WISE significantly outperforms these two protocols. Since WISE uses the white space model to control the collision probability of each transmission, the delivery performance remains relatively stable despite the increasing data rate of WiFi. We observe that when WiFi data rate is 3Mbps, the performance gains of WISE over B-MAC and OppTx are 4x and 2x, respectively.

We now evaluate the performance of WISE for unicast traffic. The maximum retransmission tries is set to 3, which is the default setting in 802.15.4. Fig. 11 shows the impact of WiFi data rate on the frame delivery ratio of ZigBee link. The result is similar as the case of broadcast traffic. Despite the high traffic load of WiFi blind terminal, WISE with collision bound of 0.1 constantly achieves a frame delivery ratio above

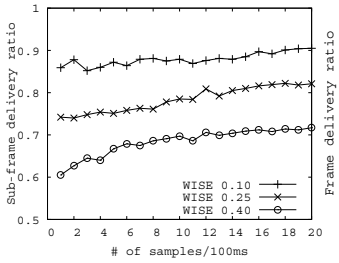


Fig. 8. Sampling frequency.

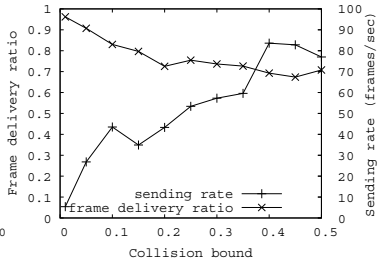


Fig. 9. Collision bound.

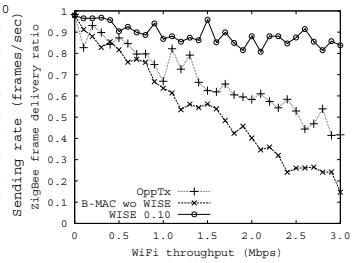


Fig. 10. Brdcast frame delivery ratio.

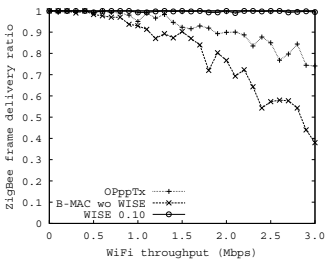
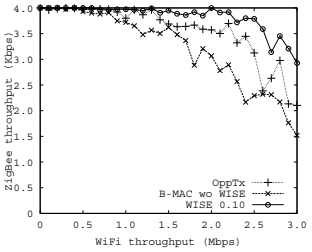
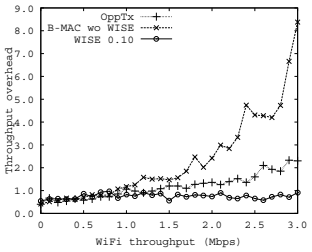


Fig. 11. Unicast Frame delivery ratio.



(a) Throughput.



(b) Throughput overhead.

Fig. 12. ZigBee throughput vs WiFi throughput.

98%. Fig. 12(a) shows the throughput of OppTx, B-MAC, and WISE with collision bound of 0.1. It can be seen that WISE performs consistently better than the other two protocols.

We observe from Fig. 11 that, although the frame delivery ratio is always higher than 95%, the throughput of WISE begins to drop when the traffic load of WiFi exceeds 2Mbps. We note that the reason of the throughput drop for WISE is different from the other two protocols. For B-MAC and OppTx, the throughput drop is mainly caused by frame loss because they failed to predict the interference of future WiFi transmissions. In contrast, WISE maintains constant loss rate (as required by the collision probability bound) while decreases the sending rate. This result suggests that, when the channel is heavily loaded by WiFi traffic, WISE will gracefully lower the sending rate to avoid frequent retransmissions, which leads to significantly lower overhead. This phenomenon is illustrated more clearly in Fig. 12 that shows the impact of WiFi data rate on the throughput overhead of ZigBee. Due to channel sampling, the overhead of WISE is slightly higher than that of other protocols when WiFi traffic load is low. However, when WiFi traffic load is high, WISE achieves significantly lower overhead than other protocols. Specifically, when the throughput of WiFi is 3.0 Mbps, the throughput overhead of WISE is only 0.65, which is 10.9% and 39.5% of that of B-MAC and OppTx, respectively.

VIII. CONCLUSION

In this paper, we propose a novel approach that enables ZigBee links to achieve assured performance in the presence of heavy WiFi interference. Based on statistical analysis of real-life network traces, we present a Pareto model to accurately characterize the white space in WiFi traffic. We also analytically model the performance of a ZigBee link in the presence of WiFi interference. Finally, we develop WISE - a new ZigBee frame control protocol, which allows ZigBee networks co-existing with WiFi to achieve desired link throughput and delivery ratio. Our extensive experiments on a testbed of 802.11 netbooks and 802.15.4 TelosB motes show that WISE achieves 4x and 2x performance gains over B-MAC and a recent reliable transmission protocol, respectively, while only incurs 10.9% and 39.5% of their overhead.

IX. ACKNOWLEDGEMENT

This work is supported, in part, by the National Science Foundation under grant CNS 0916576.

REFERENCES

[1] D-itg distributed internet traffic generator. http://www.grid.unina.it/software/ITG/.
[2] R. Chandra, R. Mahajan, V. Padmanabhan, and M. Zhang. Crawdad data set microsoft/osdi2006 (v. 2007-05-23), 2007.
[3] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: evidence and possible causes. *IEEE/ACM Trans. Netw.*, 1997.
[4] H. Dubois-Ferrière, D. Estrin, and M. Vetterli. Packet combining in sensor networks. In *ACM SenSys*, 2005.
[5] R. K. Ganti, P. Jayachandran, H. Luo, and T. F. Abdelzaher. Datalink streaming in wireless sensor networks. In *SenSys*, 2006.
[6] S. Geirhofer, L. Tong, and B. Sadler. Dynamic spectrum access in wlan channels: Empirical model and its stochastic analysis. In *ACM TAPAS*, 2006.
[7] S. Geirhofer, L. Tong, and B. Sadler. Cognitive medium access: Constraining interference based on experimental models. *IEEE Journal on Selected Areas in Communications*, 2009.
[8] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan. Understanding and mitigating the impact of rf interference on 802.11 networks. In *ACM SigCOMM*, 2007.
[9] IEEE. Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (lr-wpans). *IEEE Standard 802.15.4*, 2003.
[10] K. Jamieson and H. Balakrishnan. Ppr: Partial packet recovery for wireless networks. In *ACM SigCOMM*, 2007.
[11] J. H. Kim and J. K. Lee. Capture Effects of Wireless CSMA/CA Protocols in Rayleigh and Shadow Fading Channels. *IEEE Transactions on Vehicular Technology*, 1999.
[12] W. E. Leland, W. Willinger, M. S. Taqqu, and D. V. Wilson. On the self-similar nature of ethernet traffic. *ACM SIGCOMM Comput. Commun. Rev.*, 1995.
[13] V. Paxson and S. Floyd. Wide-area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 1995.
[14] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *ACM SenSys*, 2007.
[15] A. Schulman, D. Levin, and N. Spring. Crawdad data set umd/sigcomm2008 (v. 2009-03-02), 2009.
[16] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, and P. Levis. The beta-factor: measuring wireless link burstiness. In *ACM SenSys*, 2008.
[17] W. Wei. Time series analysis. *Addison-Wesley*, 1990.
[18] Y. Wu, G. Zhou, and J. A. Stankovic. Acr: Active collision recovery in dense wireless sensor networks. In *INFOCOM*, 2010.
[19] R. Zhou, Y. Xiong, G. Xing, L. Sun, and J. Ma. Zifi: Wireless lan discovery via zigbee interference signatures. In *ACM MobiCom*, 2010.