HMM-driven Smart White-space-aware Frame Control Protocol for Coexistence of ZigBee and WiFi

Jie Yuan, Ted Ward, Sam Honarvar, Tingting Chen and Johnson Thomas
Department of Computer Science
Oklahoma State University, Stillwater, OK, USA
{jyu, ted.ward, honarva, tingtic, jpt}@cs.okstate.edu

***Abstract*—ZigBee has been used more and more extensively in various applications such as wireless patient monitoring, herds monitoring, smart control in home networking and game remote controllers, etc. Most of these applications are performance-sensitive so the throughput and packet delivery ratio should be guaranteed for ZigBee system to work properly. However, since both WiFi and ZigBee are operated in unlicensed ISM spectrum, the interferences from WiFi hotspots make the coexistence of ZigBee and WiFi a big challenge. WiFi traffic contains quite a lot of white spaces between frame clusters in the time domain which could be taken advantage of to improve the performance of the systems with the coexistence of ZigBee and WiFi. Most existing mechanisms dealing with the coexistence of heterogeneous wireless systems neglects this important fact. In this paper, we propose a novel approach that ensures high performance of ZigBee in spite of the presence of strong interference from WiFi, and at the same time keep the WiFi performance almost unaffected. Our approach is to learn a Hidden Markov Model (HMM) based on traces of WiFi whitespaces in the current network. With such a HMM model we can accurately characterize the dynamic distribution of the durations of white spaces in different times. Based on the HMM model of white spaces and the analysis of system performance we develop a novel ZigBee frame control protocol called *HMM-driven Smart White-space-aware Frame Control Protocol* which can allow ZigBee networks to coexist with WiFi networks with desired link throughput and packet delivery ratio. Some initial experimental results have shown the effectiveness of our protocol.**

Keywords*—ZigBee; WiFi; Frame Clusters; White Spaces; HMM ; Frame Control Protocol

I. INTRODUCTION

The ISM spectrum is becoming increasingly popular in wireless networks such as ZigBee and WiFi. However, the coexistence of such heterogeneous wireless systems is still an open challenge. ZigBee devices are subject to the interferences from WiFi signals. As the number of WiFi hotspots and devices steadily increases over time, ZigBee and WiFi devices are more likely to be located in the same place which makes the coexistence problem even more challenging and critical. The traditional approach to solve the coexistence problem is to assign orthogonal channels to ZigBee and WiFi devices, i.e. let the two kinds of devices use non-overlapping channels. However, as the number of WiFi devices increases dramatically nowadays, the 3 non-overlapping 802.11 channels (1, 6, and 11) in the 2.4GHz band are heavily occupied by existing WiFi access points and devices. These 3 non-overlapping 802.11 channels overlap with 12 out of 16 802.15.4 channels in the 2.4GHz band. Fig. 1 [1] shows the overlaps between 802.11 and 802.15.4 channels. The four 802.15.4 channels (15, 20, 25 and 26) that don't overlap with 802.11 channels 1, 6 and 11 are heavily used by ZigBee devices to avoid the interference from WiFi and are getting more and more congested, especially in a highly busy environment. This incurs significant decrease in the efficiency of spectrum usage. If we could manage to make WiFi and ZigBee work properly in the same channel or overlapping channels, we can make full use of all the 16 802.15.4 channels in the 2.4GHz band, thus significantly increase the efficiency of spectrum usage and the potential throughput of ZigBee links.

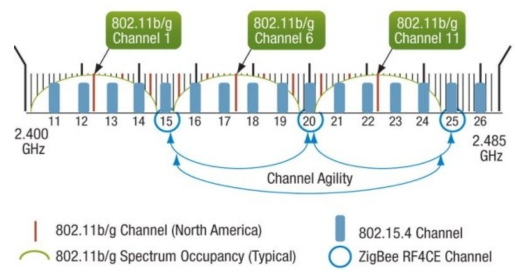

Fig. 1. Overlaps between 802.11 and 802.15.4 channels

The existing CSMA-CA approach to the coexistence problem of ZigBee and WiFi neglects the prominent channel white spaces and thus is not satisfactory. CSMA-CA allows a WiFi device to conduct the clear channel assessment by carrier-sensing 802.11-modulated signals. ZigBee signals are invisible to WiFi transmitters and the WiFi transmitters will not defer their transmissions even in presence of ongoing ZigBee transmissions. ZigBee devices have a power that is typically 20dB lower than that of WiFi devices. So the ZigBee signals can barely be sensed by WiFi devices while WiFi signals can easily be sensed by ZigBee devices. Even if the in-air ZigBee packets can be sensed by WiFi devices, a ZigBee transceiver has a 16 times longer response time and is thus often preempted by WiFi when it switches from sensing to transmission, or from transmission to reception mode. In a word, WiFi devices cannot effectively detect ZigBee signals and thus often corrupt in-air ZigBee packets.

In [2], a protocol called WISE was proposed to deal with the co-existence of WiFi and ZigBee. WISE takes advantage of channel white spaces (i.e. the inter-arrival times of WiFi frame clusters) and works better than other protocols that neglect white

This work is supported in part by NSF grant No. CNS-0709329.

spaces such as B-MAC [3] and OppTx [4]. However, WISE neglects the dynamic distribution characteristic of white spaces. Also WISE is very sensitive to the size of sliding window and fails to fit white spaces distribution well when the window size is larger than their empirical optimal size.

In this paper, we propose a novel approach that takes advantage of the channel white spaces and thus could further reduce the effects of the interferences between WiFi and ZigBee devices. Our approach has several improvements over WISE such as more accurate description of white spaces distributions by capturing its dynamic characteristic and greater flexibility to the size of the sliding window. Our major contributions are summarized as follows.

- By observing the data traces of WiFi traffic we reveal that there are strong correlations between neighboring white spaces. Our experiments show HMM fits the white spaces distributions better than Pareto and thus the prediction power of HMM is better than that of Pareto.
- We propose a novel frame control protocol called *HMM-driven Smart White-space-aware Frame Control Protocol* which suits the dynamic distribution characteristic of white spaces.
- We implement our frame control protocol in TinyOS and test it in a testbed to show its effectiveness.

II. Related Work

The coexistence of heterogeneous wireless systems in unlicensed ISM bands is a hot topic. The 802.15.4 specifications [5] use the Adaptive Frequency Hopping (AFH) for the coexistence of WiFi and Bluetooth devices. The Enhanced Adaptive Frequency Hopping (EAFH) [6] was proposed to further improve AFH. The OppTx protocol proposed by Srinivasan, etc. [4] exploits the correlations in packet delivery and loss to set transmission backoff delay. Xinyu Zhang, etc [7] proposed the Cooperative Busy Tone mechanism which designates a separate ZigBee node as a signaler that emits the busy tone to prevent WiFi preemption. All these approaches above neglect the probabilistic feature of channel white spaces between WiFi frames. Stefan Geirhofer etc. [8] modeled the busy and idle durations of 802.11b channels with semi-Markov model. However, they didn't propose a frame control protocol for practical use. Also, their semi-HMM has only one state for the event "channel is idle", which fails to capture the dynamic distribution characteristic of white spaces. The WISE frame control protocol proposed in [2] exploits the channel white spaces between frame clusters. WISE adopts the Pareto model for WiFi white spaces based on the fact that the arrival process of WiFi frame clusters has the feature of self-similarity. However they neglected the dynamic characteristics of the probabilistic distribution of the white spaces. They assumed that the white spaces within a sliding window follow an i.i.d Pareto distribution. WISE works well when the window size is less than 100ms because within such a small time interval the probabilistic distribution is relatively stable. In the next Section we will show that there are strong correlations between neighboring white spaces. Our approach captures the correlations between white spaces within a sliding window and adjusts the sub-frame size according to the dynamic distributions of white spaces.

III. HMM Modeling of WiFi White Spaces

In this section, we observe the WiFi whitespace distributions and model them using Hidden Markov Models to capture the dynamic characteristics.

The channel utilization ratio in a WiFi network is usually quite low. Fig. 2(a) is the state trace of a real-life WiFi channel within about one second in a heterogeneous system with two ZigBee motes, one WiFi AP and two laptops. Fig. 2(b) is the close-up of the interval between timestamp 625ms and 700ms. A 0 value in the vertical axis means idle channel and 1 means busy channel. These traffic data was obtained by the driver of CC2420 radios and the motes' microcontrollers. Whenever the electric potential of CCA pin goes high and no start-of-frame signal received, a white space is detected. As shown in Fig. 2, the network traffic has considerable amount of white spaces between frame clusters. We do not consider the extremely short inter-frame intervals as white spaces.

In Fig. 2, we can observe that for a short white space there are usually a bunch of white spaces close to it also with short durations, and vice versa. This characteristic of the white spaces durations inspired us to come up with a new model of dynamic distribution for channel white spaces.

We argue that the durations of the white spaces Δt_1, Δt_2, Δt_3, ... , Δt_N follow a Hidden Markov process. This means the current white space is determined by the previous k white spaces in the case of k-order Markov process. In this paper we use 1-st order Hidden Markov Model, i.e. k=1. Compared to the Pareto model used in [2], HMM has the advantage that it captures the characteristic of dynamic distributions of white spaces and gets rid of the assumption of WISE that all white spaces within a sliding window follow an i.i.d Pareto distribution. Fig. 3 shows our HMM where a_{ij} means transition probabilities and $b_m(o_n)$ means the probability that the state m emits an observation o_n. It also shows how the HMM generates an observation sequence $(o_1,o_2,...,o_7)$.

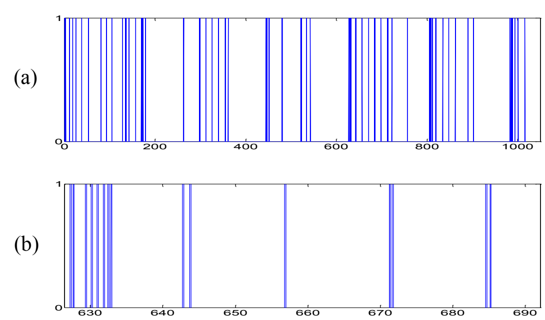

Fig. 2. State trace of a WiFi channel

In an HMM, there are a set of states $\{S_1, S_2, S_3, ... , S_K\}$. Each state has an initial state probability P_{0m} $(1 \leq m \leq K)$ and a probabilistic distribution of its observations. We use $\{\Theta_1, \Theta_2, \Theta_3, ... , \Theta_K\}$ to describe the parameters of the states' distributions, e.g. mean and covariance matrix for Gaussian. The transition probability matrix M where $M_{ij}=Pr(S(t+1)=S_j|S(t)=S_i)$ determines how a state could transit to another state.

In our application, the durations of white spaces $(\Delta t_1, \Delta t_2, \Delta t_3, ..., \Delta t_N)$ serve as the observation sequence $(o_1,o_2,o_3,...,o_N)$. We use Viterbi Algorithm to infer the state sequence associated with the given observation sequence. We use the Gaussian model (GMM is also among the alternatives) as the observation distribution for each state in our protocol due to its analytical tractability and powerful capability. With pre-obtained data traces of the traffic in the current network, we can learn all the parameters of this HMM model.

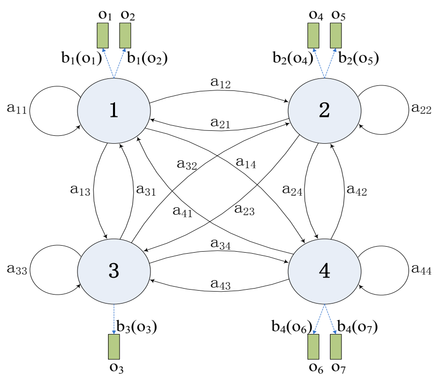

Fig. 3. 4-state ergodic HMM. In this example, the corresponding state sequence for the observation sequence $(o_1,o_2,...,o_7)$ is (1,1,3,2,2,4,4).

In practice, we can associate each white space with a hidden state, or optionally we can associate a group of white spaces (say 10 consecutive white spaces) with a hidden state. The former strategy should get better result but will result in heavier computation load. With the latter strategy we are actually abstracting over the 10 consecutive white spaces so we'll lose some detailed information of the Markov process. In our implementation, we used the former one for better accuracy.

Assume that the *i*th white space has just passed, i.e. at time step t_i. We want to know the duration of the $(i+1)$th white space. The durations of all the previous white spaces (i.e. all the observations up to *i*) $(o_1, o_2, o_3, ..., o_i) = (\Delta t_1, \Delta t_2, \Delta t_3, ..., \Delta t_i)$ are already known. We can infer from the trained HMM model the current state and the probability distribution of the next state based on the maximum likelihood estimation. Let $P_{i+1}(m)$ be the estimated probability that the next state is S_m $(1 \le m \le K)$, i.e. $P_{i+1}(m) = P(S(i+1) = S_m)$, and let random variable x be the duration of white spaces. Then

$$\Pr(x > t) = \sum_{m=1}^{K} [P_{i+1}(m) \cdot \Pr(x > t \mid S_m)] \tag{1}$$

is the probability distribution of the duration of the next white space. $P_{i+1}(m)$ describes which hidden state the next white space Δt_{i+1} would be in, and this can be inferred by the Viterbi Algorithm. Note we use the subscripts to denote the distinct states from the finite states set $\{S_1, S_2, S_3, ..., S_K\}$, and use the indices in the parenthesis to denote the states at a particular time step. $\Pr(x > t \mid S_m)$ is the conditional probability of $x > t$ given that the hidden state is S_m. Since the HMM is already trained and the observation distribution with parameters Gaussian mean and sigma within each state including S_m is known, thus

$$\begin{aligned}\Pr(x > t \mid S_m) &= 1 - \Pr(x \le t \mid S_m) \\ &= \frac{1}{2}\left(1 - erf\left(\frac{t-\mu_m}{\sqrt{2}\cdot\sigma_m}\right)\right)\end{aligned} \tag{2}$$

where *erf* denotes the error function.

IV. Smart White Space-aware Frame Control Protocol

To avoid the collision with WiFi signals, the simplest strategy is to decrease the packet size of ZigBee. However, this will greatly reduce the system throughput. So we need to get an optimal tradeoff between the overall throughput and the packet delivery ratio. Our guideline for this optimal tradeoff is trying to finish the ZigBee transmission within the current white space before the arrival of the next WiFi frame cluster. Since the average lifetime of a white space is limited, we need to divide a ZigBee frame into sub-frames and give an ID for each frame session. The following content in this section shows how to optimize this sub-frame size λ as well as how our protocol works.

Based on the HMM White Spaces model, we can get the following conditional collision probability for a given frame size λ:

$$C(\lambda,\delta) = 1 - \sum_{m=1}^{K} [P_{i+1}(m) \cdot \Pr(x > \delta + \frac{\lambda}{R} \mid S_m, \delta)] \tag{3}$$

where R is the channel rate of ZigBee, δ is the age of white space when a frame is ready for transmission, random variable x is the duration of white spaces and $P_{i+1}(m)$ denotes the probability that the upcoming white space has a hidden state S_m. This is quite straightforward: we need to finish the transmission of a frame within the remaining lifetime of a white space.

Our goal of frame adaptation is to maximize the transmission efficiency while limit the collision probability. Protocol header has a fixed size, so the transmission efficiency is a monotonic increasing function of the sub-frame size λ. Given a collision probability threshold T, our optimization problem can be formulated as follows:

$$Maximize \quad \lambda \tag{4}$$

$$Subject \quad to \quad : \begin{cases} C(\lambda,\delta) < T & (5) \\ \lambda \le M & (6) \end{cases}$$

where M is the maximum allowed ZigBee frame size.

Let λ_0 be the root of the equation $C(\lambda,\delta) - T = 0$, then the optimal sub-frame size $\lambda_{opt} = \min(\lambda_0, M)$. The objective function is

$$f(\lambda) = C(\lambda,\delta) - T = \frac{1}{2} - T + \frac{1}{2}\sum_{m=1}^{K} P_{i+1}(m) \cdot erf\left(\frac{\delta + \frac{\lambda}{R} - \mu_m}{\sqrt{2}\cdot\sigma_m}\right) \tag{7}$$

In our implementation, we used bisection method to find the numerical solution of the root of the objective function $f(\lambda)$.

To realize the optimization method described above, our *Frame Control Protocol* has a training and prediction process to learn the white spaces distribution and to determine the sub-frame size respectively. We keep a sliding window of durations of white spaces collected in the past 1 second. Fig. 4 shows the sliding window mechanism of the training and prediction process. In each round of training and prediction we use the current window of white spaces to train an HMM model and calculate the optimal sub-frame sizes for the next L white spaces. When L white spaces have passed we begin a new round. In the next round we use an updated sliding window to retrain an HMM model and repeat the same process as the previous round. In Fig. 4 each interval in the time axis represents a white space, and in the time axis we just list all the white spaces in temporal order and ignore the busy times because we are dealing with white spaces. The value of L is determined by the number of white spaces in the training data. We set it to be one third of the number of training samples.

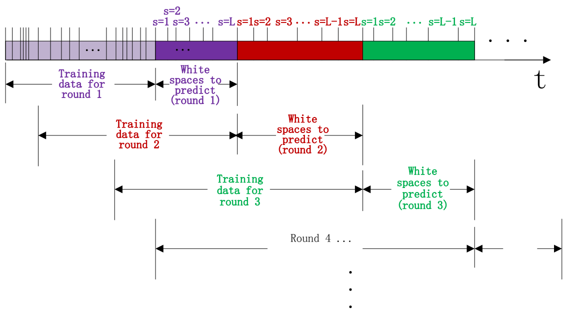

Fig. 4. Schema of the training and prediction process

V. EXPERIMENTS

In this section we show the results of 2 sets of experiments. We first compare Pareto and HMM in the power of predicting the upcoming white spaces, and then show experimental results from a real-life WiFi network with heavy traffic using our approach.

In the first experiment, we use 4-state HMM and use 3-component GMM as the observation distribution function. Both the training and test data are obtained from a WiFi network. Fig. 5 shows one of the sequences of white spaces.

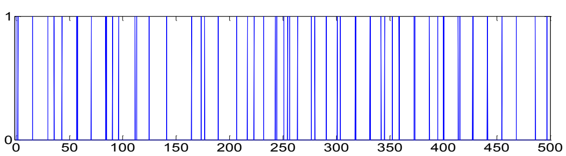

Fig. 5. Inter frame-cluster white spaces

For both Pareto and HMM, we first use the training data to train the model parameters. And then use the models to predict the duration of the upcoming white spaces. For any test white space sequence (Δt_1, Δt_2, Δt_3, ... , Δt_N), we use the 2 models to predict Δt_i respectively based on the previous white spaces (Δt_1, Δt_2, Δt_3, ..., Δt_{i-1}) for all i = 2, 3, ..., N. Although we can predict the probabilistic distribution of the upcoming white space Δt_i , we only use the expectation value in this performance evaluation. And then calculate the average prediction error over the (N-1) predictions.

TABLE I. PREDICTION ERROR OF THE UPCOMING WHITE SPACES

Model	**Average Prediction Error (ms)**			
	Run 1	***Run 2***	***Run 3***	***Average***
Pareto	7.4902	7.4902	7.4902	7.4902
HMM	5.7296	5.6017	5.1583	5.4965

From the table we can see that HMM is more accurate in predicting the upcoming white spaces based on the information about the previous white spaces.

In the second experiment, we use 4-state HMM and use Gaussian model as the observation distribution function. Two laptops connected to the access point are used to generate high-intensity WiFi traffic, and two ZigBee motes equipped with CC2420 radios communicate with each other. They work in overlapping channels and the interference from WiFi is quite strong. Fig. 6 shows the result of our experiment. The curve marked by "+" is for the CSMA-based B-MAC and the curve marked by green dots is for our protocol. We can see significant performance improvement over B-MAC with our protocol.

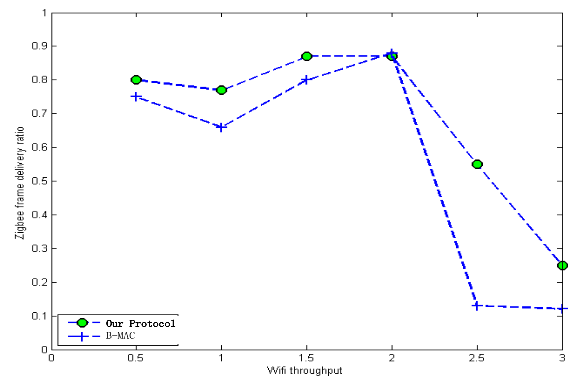

Fig. 6. ZigBee packet delivery ratio vs WiFi throughput (Mbps)

VI. CONCLUSION

In this paper we propose a *HMM-driven Smart White-space-aware Frame Control Protocol* for ZigBee that deals with interference from WiFi. Our experimental results show that exploiting the WiFi channel white spaces is very useful for the coexistence of ZigBee and WiFi. Compared to WISE proposed in [2] our approach captures the dynamic distribution characteristic of white spaces and thus models the white spaces distribution more accurately, which gives us a stronger power of prediction on upcoming white spaces. Our experimental results show that our protocol works effectively for ZigBee networks co-existing with WiFi networks where there exist strong interferences from WiFi devices.

REFERENCES

[1] ZigBee Alliance. ZigBee RF4CE Standard, 2009.

[2] J. Huang, G. Xing, G. Zhou, R. Zhou. Beyond Co-existence: Exploiting WiFi White Space for ZigBee Performance Assurance. In ICNP, 2010.

[3] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In ACM SenSys, 2007.

[4] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, and P. Levis. The betafactor: measuring wireless link burstiness. In ACM SenSys, 2008.

[5] 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.

[6] A. C.-C. Hsu, D. S. L. Wei, C.-C. J. Kuo, etc. Enhanced Adaptive Frequency Hopping for Wireless Personal Area Networks in a Coexistence Environment. In IEEE Globecom 2007.

[7] X. Zhang, K. G. Shin. Enabling Coexistence of Heterogeneous Wireless Systems: Case for ZigBee and WiFi. In ACM Mobihoc, 2008.

[8] S. Geirhofer, L. Tong, and B. Sadler. A Measurement-Based Model for Dynamic Spectrum Access in WLAN Channels : In Proc. IEEE MILCOM 2006.