Modeling WiFi Traffic for White Space Prediction in Wireless Sensor Networks

Indika S. A. Dhanapala[1,*], Ramona Marfievici[1], Sameera Palipana[1], Piyush Agrawal[2], Dirk Pesch[1]
[1]Nimbus Center for Embedded Systems Research, Cork Institute of Technology, Cork, Ireland
[2]United Technologies Research Center, Cork, Ireland
[*]Contact Author: I.S.A.Dhanapala@mycit.ie

***Abstract*—Cross Technology Interference (CTI) is a prevalent phenomenon in the 2.4 GHz unlicensed spectrum causing packet losses and increased channel contention. In particular, WiFi interference is a severe problem for low-power wireless networks causing a significant degradation of the overall performance. We propose here a proactive approach based on WiFi interference modeling for accurately predicting transmission opportunities for low-power wireless networks. We leverage statistical analysis of real-world WiFi traces to learn aggregated traffic characteristics in terms of Inter-Arrival Time (IAT) that, once captured into a specific 2nd order Markov Modulated Poisson Process (MMPP(2)) model, enable accurate estimation of interference. We further use a hidden Markov model (HMM) for channel occupancy prediction. We evaluated the performance of: $i)$ the MMPP(2) traffic model w.r.t. real-world traces and an existing Pareto model for accurately characterizing the WiFi traffic and, $ii)$ compared the HMM based white space prediction to random channel access . We report encouraging results for using interference modeling for white space prediction.**

I. Introduction

Wireless technologies operating in unlicensed radio spectrum, such as the 2.4 GHz ISM band, suffer from Cross Technology Interference (CTI). The interference occurs due to the broadcast nature of wireless transmissions of co-located devices whose radios are based on different technologies such as WiFi (IEEE802.11), Bluetooth (IEEE802.15.1) or IEEE802.15.4 and who cannot coordinate their transmissions. CTI creates packet losses, increases channel contention which increases delay, and ultimately under-utilizes the scarce frequency spectrum [1], [2]. The coexistence between IEEE802.11 and low-power IEEE802.15.4 devices represents a challenge for several reasons: $i)$ IEEE802.11 is a higher bandwidth technology, $ii)$ WiFi channels have a bandwidth of 22 MHz and can therefore interfere with multiple IEEE802.15.4 channels simultaneously (see Fig. 1); this aggregated WiFi interference reduces white spaces and transmission opportunities for WSNs and degrades their reliability and lifetime and, $iii)$ IEEE802.11 supports high-throughput transmissions that create rapid variations in time and frequency domain hard to detect by the typical IEEE802.15.4 node [2].

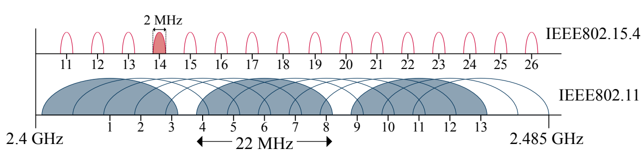

Fig. 1: Overlapping WiFi and IEEE802.15.4 channels.

Related work. Existing solutions rely on reactive or proactive approaches to address CTI between WiFi and WSNs. Unfortunately, reactive approaches [2]–[4] only deal with CTI when it occurs, thus negatively affecting performance. On the other hand, proactive approaches try to predict white spaces, by modeling channel occupancy [5]–[7] or emulating interference caused by WiFi and Bluetooth [8]. A two-state semi-Markov model for channel occupancy is defined in [6], [7] and noise measurements are used to measure the duration of the *Free* and *Busy* instants, and compute the CDF of *Free* and *Busy* periods. Based on the longest *Busy* period, MAC protocols' parameters are derived such that the application requirements are met. Jamlab [8] models and regenerates WiFi interference patterns, considering both saturated (always *Busy*) and unsaturated traffic scenarios. A Markov chain model is used for saturated traffic and a probability mass function of empirical data for the non-saturated one. The work from [5] is closely related to ours. Nevertheless, the proposed channel modeling technique, Pareto, is a heavy-tailed distribution which usually captures long-range dependent traffic behavior refuted in the long run by our real-world WiFi traces.

Contribution. In this paper, we present a proactive time domain technique to address CTI for WSNs based on: $i)$ an accurate WiFi traffic model to capture aggregated WiFi interference, and $ii)$ a model-based technique to predict white spaces for IEEE802.15.4 transmissions. The approach is motivated by our findings when we assessed the statistical properties of real-world WiFi traces at different time scales: traffic exhibits self-similarity [9]. We exploit this to better infer traffic properties.

We selected a 2^{nd} order Markov Modulated Poisson Process (MMPP(2)) to model the WiFi packet Inter-Arrival Times (IAT) and a Hidden Markov Model (HMM) for predicting the white spaces. We use an MMPP(2) model because we need: to model WiFi traffic for both saturated and unsaturated channel characteristics and, to capture the self-similarity of WiFi traffic with only a few model parameters that are easily measurable. We learn WiFi aggregated traffic characteristics in terms of packet IAT from real-world WiFi traces and use those to calibrate the parameters of the MMPP(2), which provides us with an accurate model of aggregated WiFi interference. We then use the HMM for channel occupancy prediction, which is trained using the real-world WiFi traces and the MMPP(2) model. This provides us with a prediction when the channel is interference free. We have evaluated the performance of the MMPP(2) traffic model with respect to the traces we collected and an existing Pareto model [5] and compared our HMM based channel access with 0.5-persistent random access technique. Our analysis shows we can successfully predict the presence of white spaces. We present a detailed description of our approach in the next section, and its performance evaluation in Section III. We conclude and outlook in Section IV.

II. Approach

Traffic statistical properties. Key statistical metrics providing insights on network traffic are: *mean* (μ), *standard deviation* (σ) and, *coefficient of variation* (C). Self–similarity is quantified through the *Hurst* parameter (scalar $H \in (0{,}1)$). A value of 0.5 or less indicates the absence of self–similarity, and the closer H is to 1 the greater the self–similarity. Calculating H is not straightforward: firstly, it can only be estimated, secondly, although there are several methods to estimate it, the results are often conflicting. We used Peng, Periodogram and Box–Periodogram methods as they can produce a good estimate for a sample size as low as 7000 [10]. The median of the estimated H is used as it is more stable than the mean even when one of the estimators fails.

Overview. Fig. 2 illustrates our approach for predicting white spaces for WSNs. To achieve our goal, we must start from assessing quantitatively the characteristics of the WiFi traffic in the operating environment. To this end, we collect a set of WiFi traces of length x s and, in case of overlapping WiFi channels, we merge them to replicate the WiFi interference as seen by WSNs. Then, we extract the packet IAT distribution of the aggregated trace and characterize it in terms of mean (M_1), C and H. All the aforementioned statistical parameters are exploited to tune an MMPP(2) model for estimating the aggregated WiFi traffic for y s. Then, we develop an HMM with two states, *Free* and *Busy*, which is key for predicting transmission opportunities at run-time. The initial state probabilities are determined using the steady state probabilities of the MMPP(2) model, while the training is done using a set of WiFi traces of length z s.

Determining WiFi traffic characteristics. Our approach is based on an MMPP(2) model that: i) assumes traffic exhibits self-similarity and, ii) uses empirical data for estimating its parameters. Therefore, we start by acquiring raw real-world WiFi traces from the operating environment and then process them in three steps: *1) aggregating*: aggregate WiFi traces of multiple overlapping WiFi channels as seen by WSNs, i.e., concatenating and ordering raw traces by their timestamps; *2) traffic characteristics extraction*: the IAT distribution and its corresponding statistics, i.e., M_1, C and H, are determined from the aggregated WiFi traces; *3) verifying self-similarity property*: check if $H \in (0.5{,}1)$ for the aggregated trace.

Estimating WiFi traffic. We now describe how the above processing is exploited for building estimates of aggregated WiFi interference considering the profile of the environment. To this end we use an MMPP(2) model. An MMPP(2) process

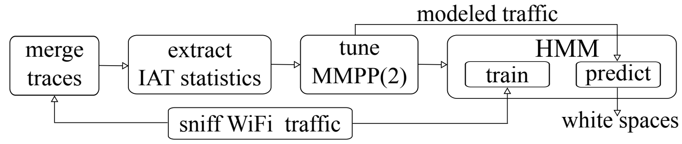

Fig. 2: The system model.

can be approximated by a second-order *hyperexponential distribution* [11], with parameters μ_1, μ_2-mean packet arrival rates and p-probability at which traffic is generated at a mean rate of μ_1, for fitting empirical packet IAT distribution to the model. We exploit the output of the *traffic characteristics extraction*, M_1 and C, to automatically compute the parameters p, μ_1 and μ_2 through the *balanced means* method [12]; then derive the parameters of the MMPP(2) model. We omit all the formulas due to space limitations. When $\frac{1}{\sqrt{2}} \leq C \leq 1$, a second-order *Coxian distribution* [12] must be used prior to the second-order *hyperexponential distribution*.

Predicting white spaces. Next, we exploit the output of the previous stage, the estimated aggregated interference, along with real-world WiFi traces, to train an HMM that enables us to predict the wireless channel state. The HMM parameters are: *1)* hidden states $\boldsymbol{S} = \{s_1, s_2\}$: the regimes of the wireless channel, *Free* and *Busy*; *2)* initial state probabilities $\boldsymbol{\pi}$: the steady state probabilities of the MMPP(2); *3)* observations $\boldsymbol{O} = \{o_1, o_2\}$: values of the mean packet IAT of the MMPP(2) modeled trace, *IAT_small* and *IAT_large*. We define a threshold computed as the average over all samples' means for the training phase, and as a moving average of the mean WiFi packet IAT for the prediction phase motivated by the need to dynamically adapt the HMM to the operating environment; *4)* state transition probability matrix $\boldsymbol{A}$: models the evolution of the wireless channel as transitions among the set of unobserved states; *5)* observation probability matrix $\boldsymbol{B}$. The parameters A and B are initialized using uniformly distributed probabilities matrices and recomputed using the *Baum-Welch* algorithm.

III. Performance Evaluation

We validate our approach by: i) conducting a statistical comparison between empirical data traces we collected in indoor deployments, traces from the MMPP(2), and traces from a state-of-the-art Pareto model, ii) comparing our predictions from the HMM with a 0.5-persistent random access method.

WiFi traces and their collection. We considered two environments, OFFICE and HOME. While the OFFICE has bursty traffic, it exhibits less self-similarity than the one in HOME; confirmed by the values of H: 0.5 for OFFICE and 0.7 for HOME. We used 802.11n-compliant WiFi dongles to sniff the traffic. As we are interested in exploiting IAT distributions, we recorded traces of timestamped traffic. Then, a WiFi aggregated interference trace was obtained by merging traces of overlapping WiFi channels and, finally, the IAT distribution was extracted.

A useful model should accurately model traffic on any channel. In the OFFICE we collected traces from WiFi channels 1-4, ensuring each WSN overlapped channel, i.e., 11-14, is affected by a different number of WiFi channels (Fig. 1). Since

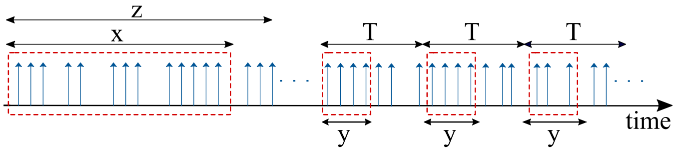

Fig. 3: Approach timeline: x is the MMPP(2) training duration, y is the length of the modeled traffic, z is the HMM training duration, while T corresponds to the WSN traffic data rate.

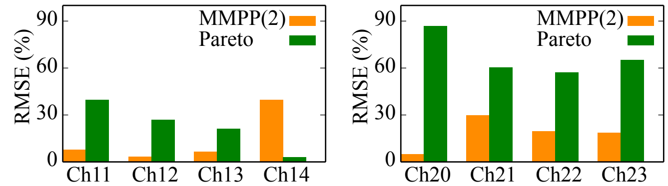

Fig. 4: Comparison of RMSE: OFFICE (left) and HOME (right).

HOME did not exhibit that much WiFi traffic as OFFICE, we sniffed WiFi channels 7 to 13 overlapping with WSN channels 20 to 23, each being affected by four WiFi channels. Moreover, to induce diversity in traffic channels in HOME we generated a continuous trace of video-streaming on WiFi channel 11 which only overlaps with WSN channels 21, 22, 23. The WiFi traces were collected during the day, from 10:30AM to 12:30PM.

System calibration. We start the evaluation by calibrating the training duration x and modelling duration y of MMPP(2), and the training duration z of the HMM, defined as in Fig. 3. Our logs showed that for $x \leq 1$ s, the statistics from the training traces can not correctly capture the WiFi behavior. A larger x, instead, is equivalent with an increase in the length of the training trace, prohibited in resource-constrained settings like the ones we target. Because we are interested in predicting the short-term behaviour of WiFi traffic, y should be as small as possible. Moreover, in order to accurately model WiFi traffic, the MMPP(2) must generate traffic from both its states at least once. This leads to define a lower bound for y denoted as $y_{lb} = \frac{1}{r_1} + \frac{1}{r_2}$, where $\frac{1}{r_i}$ is the duration of a state in the MMPP(2) model. Therefore, during the calibration process, we used $y = k \times y_{lb}$, $k \in \mathbb{Z}^+$, to find its optimum value. As our goal is to reduce the RMSE between the modeled traffic and the testing set, we first played with different values for x and y to understand their impact on the RMSE. Next, we fixed y at $k = 1$ in OFFICE and $k = 2$ in HOME, and vary x as an exponential function from 60 s to 40 minutes. All evaluations show that a value of 1 for k and 300 s for x and 2 for k and 500 s for x provide a minimum RMSE in HOME and OFFICE, respectively. The performance of the HMM depends on the size z of the training set. Our goal is to choose z to maximize hit rate and minimize the FDR. For this we used the precision-recall curve analysis looking at the impact of z on the variation of the two metrics. All evaluations show that a value of 300 s for OFFICE and 960 s for HOME satisfy the aforementioned criteria. Since we want to avoid the dependency between parameter z and the WSN channel, we average the hit rate and FDR over all channels.

MMPP(2) model validation. To evaluate the performance of our approach, we trained the MMPP(2) model with traces from different environments and channels. We compare on the basis of RMSE statistics computed on the generated trace from the MMPP(2) model and a state-of-the-art Pareto model [5] versus an unseen trace from the collected ones. Both models were trained with the same trace and Pareto's scaling parameter was set to the maximum packet duration of a WSN packet, i.e., 4.256 ms, representing the minimum white space duration. Fig. 4 shows the results from our comparison in terms of percentage RMSE. The accuracy of the estimation in HOME is lower than in OFFICE for both models, as interference decreases. However, this is not true for MMPP(2) on channel 14 in OFFICE and 20 in HOME. The accuracy of the estimation of MMPP(2) is always higher than the Pareto model. The difference is more marked on channel 11 in OFFICE and channel 20 in HOME, Fig. 4 showing a difference of 6.6 ms and 244.5 ms, respectively. However, channel 14 in OFFICE shows a different behavior, Pareto delivering best performance on this channel even if there is more traffic. Table I shows that on channel 14 there is a higher traffic, higher number of samples and lower mean IAT 10.7 ms, than on the other channels, while H has the lowest value 0.51, which translates to a less self-similarity. The characteristics of the traffic on channel 14 are more favorable for Pareto which captures the mean IAT than for MMPP(2) which requires all the three parameters for traffic modeling. Moreover, on channel 20, which is not overlapped by the video-streaming traffic channel, the mean of IAT is almost double, 141.5 ms, than on other channels, 67–82 ms, see Table II. Pareto, a heavy-tailed distribution, fails to capture the unsaturated traffic on this channel, having an RMSE of 259 ms. On the other hand, MMPP(2) model performs the best, i.e., RMSE of 14.5 ms out of all HOME channels. All results confirm the versatility of the MMPP(2) for modeling both unsaturated and saturated traffic.

HMM model evaluation. We consider four metrics to assess the performance of our approach: i) *hit rate*, ii) *false discovery rate (FDR)*, iii) *precision*, computed as $1 - FDR$, and, iv) *F1 score*. All metrics are derived from the elements of the confusion matrix. Hit rate accounts for the ability of the model to predict the channel is *Free* when it is really *Free*, while FDR provides a direct assessment of the probability

TABLE I: Traffic characteristics in OFFICE and HOME.

OFFICE				HOME			
WSN Ch	mean (ms)	CV	Hurst	WSN Ch	mean (ms)	CV	Hurst
11	18.6	0.80	0.54	**20**	141.5	0.90	0.63
12	18.4	0.80	0.52	**21**	67.1	0.92	0.69
13	18.2	0.81	0.55	**22**	76.0	0.91	0.70
14	10.7	0.87	0.51	**23**	82.2	0.92	0.71

TABLE II: Comparison of absolute RMSE.

OFFICE			HOME		
WSN	RMSE (ms)		WSN	RMSE (ms)	
Ch	MMPP(2)	Pareto	Ch	MMPP(2)	Pareto
11	1.6	8.1	**20**	14.5	259.0
12	0.7	5.2	**21**	34.3	69.5
13	1.4	4.4	**22**	21.4	62.9
14	4.2	0.3	**23**	20.6	71.7

TABLE III: Performance metrics of 0.5-persistent random access method and proposed HMM approach in two environments: a) OFFICE: $x = 300$ s, $y = y_{lb}$ and $z = 300$ s, b) HOME: $x = 500$ s, $y = 2 \times y_{lb}$ and $z = 960$ s.

OFFICE									**HOME**							
WSN ch	**FDR (%)**	**Hit rate (%)**	**F1 score (%)**	**TP**	**FP**	**FN**	**TN**	**Method**	**WSN ch**	**FDR (%)**	**Hit rate (%)**	**F1 score (%)**	**TP**	**FP**	**FN**	**TN**
11	22.2	50.6	61.3	543	155	530	152	random	20	2.7	48.2	64.5	248	7	266	7
	21.6	52.2	62.7	560	154	513	153	HMM		2.8	61.1	75.0	314	9	200	5
12	23.1	49.6	60.3	530	159	538	153	random	21	3.2	53.3	68.6	269	9	237	13
	23.0	90.8	83.4	970	289	98	23	HMM		3.7	56.1	70.9	284	11	222	11
13	0.0	50.2	66.8	692	0	687	1	random	22	3.8	50.4	66.1	256	10	252	10
	0.1	100	100	1379	1	0	0	HMM		4.3	56.7	71.2	288	13	220	7
14	36.1	49.8	56.0	431	243	434	272	random	23	4.8	51.0	66.4	260	13	250	5
	37.5	71.1	66.5	615	369	250	146	HMM		4.0	55.9	70.6	285	12	225	6

at which the model incorrectly identifies the channel as *Free* when it is *Busy*. In fact, FDR is a measure of the packet loss rate of the WSN when the prediction mechanism is being used. F1 score is defined as the harmonic mean between hit rate and precision metrics, balancing the two. HMM training requires a set of observations and initial state probabilities. The latter are obtained from the MMPP(2) model and are constituted by the steady state probabilities, while the former are obtained from the sniffed z s length WiFi traffic. Additionally, this step extracts samples of y s length every T s. Next, the mean IAT of each sample is computed and compared with the predefined threshold, as defined in Section II, in order to obtain the training observations sequence. Then, both the state transition and observation probability matrix are initialized for the data set using uniformly distributed probabilities matrices and recomputed using the *Baum-Welch* algorithm. Based on these values, the calibrated x, y and z values, and considering a WSN data rate $T = 5$ s, the HMM model is used for predicting white spaces every T s. We evaluated the performance of our approach using the four metrics and by comparing it with a 0.5–persistent random access method with no retransmissions as Pareto can not be used in conjunction with an HMM. The quantitative results in the leftmost part of Table III show that for all combinations of environments and channels our approach performs as good as the 0.5-persistent access method. Moreover, our approach consistently performs better in OFFICE than HOME, apart from channel 13. The reason for this behavior is that in OFFICE the traffic exhibits heavy bursts which reduces the availability of white spaces compared to HOME, while increasing the probability of incorrectly discovering channel is *Free*. This behavior is evident from the False Positives (FPs). On the other hand, a single FP is observed on channel 13, and our approach always predicts this channel is *Free*, succeeding every time except once. Moreover, looking at the hit rate metric, our approach performs much better in correctly predicting the channel is *Free* than the random method. Differences are more marked in OFFICE and can be as high as 50%, except channel 11. We conjecture this is an effect of the HMM state transition matrix, although this aspect requires a finer-grained investigation. Finally, the F1 score confirms our approach is more efficient in identifying available white spaces in both environments.

IV. Conclusions and Future Work

While most existing solutions addressing CTI focus on detecting and classifying interference, less work has been done on interference modeling specially for mitigation. In this work, we proposed a solution for accurately modeling WiFi aggregated traffic and predicting the presence of interference on the channel. We validated our WiFi traffic model against real-traces and a state-of-the-art Pareto model, and evaluated the prediction mechanism against 0.5-persistent random access method, for both saturated and unsaturated traffic. Results show that we are better in both settings.

Whether our approach shows the same performance at different times/locations with distinct traffic characteristics, and if it improves with an automatic system calibration, are already on our research agenda. Finally, a practical use of our approach would be its integration with both WiFi and WSN technologies which is possible with the emergence of new chips with collocated radios.

Acknowledgements. We thank Renato Lo Cigno from University of Trento, Italy, for his insightful comments. This work has been funded by the Irish Research Council in collaboration with United Technologies Research Center, Cork, Ireland.

References

[1] A. Hithnawi *et al.*, "Understanding the Impact of Cross Technology Interference on IEEE 802.15.4," in *WiNTECH*, 2014.

[2] C. A. Boano and K. Roemer, "External Radio Interference," in *Radio Link Quality Estimation in Low-Power Wireless Networks*. Springer International Publishing, 2013, pp. 21–63.

[3] V. Iyer *et al.*, "Chrysso - A Multi-channel Approach to Mitigate External Interference," in *SECON*, 2011.

[4] B. Kusy *et al.*, "Radio Diversity for Reliable Communication in WSNs," in *IPSN*, 2011.

[5] J. Huang *et al.*, "Beyond Co-existence: Exploiting WiFi White Space for Zigbee Performance Assurance," in *ICNP*, 2010.

[6] C. A. Boano *et al.*, "Jag: Reliable and Predictable Wireless Agreement under External Radio Interference," in *RTSS*, 2012.

[7] ——, "Making Sensornet MAC Protocols Robust Against Interference," in *EWSN*, 2010.

[8] ——, "JamLab: Augmenting Sensornet Testbeds with Realistic and Controlled Interference Generation," in *IPSN*, 2011.

[9] L. Muscariello *et al.*, "Markov Models of Internet Traffic and a New Hierarchical (MMPP) Model," *Computer Communications*, vol. 28, no. 16, 2005.

[10] W. Rea *et al.*, "Estimators for Long Range Dependence: An Empirical Study," *arXiv preprint arXiv:0901.0762*, 2009.

[11] A. Nogueira *et al.*, "Fitting Algorithms for MMPP ATM Traffic Models," *Proc. of the Broadband Access Conference*, 1999.

[12] I. Adan and J. Resing, *Queueing Systems*. TU/e, 2015.