On the Coexistence of 802.11 and 802.15.4 Networks with Delay Constraints

Wei Zhang, Mahima A. Suresh, Yuhan Zhou, Raghavan S. Veera, Radu Stoleru
Computer Science Department and Electrical Engineering Department, Texas A&M University
{wzhang, agumbe, stoleru}@cse.tamu.edu, {raghavan,yuhan83}@tamu.edu

***Abstract*—Coexisting 802.11 and 802.15.4 single-cell wireless networks are experiencing significant performance degradation due to the aggressive nature of 802.11 (when compared to 802.15.4) and to their different traffic characteristics [1]. *Providing tight delay guarantees* to certain 802.15.4 applications (e.g., health monitoring with unsaturated, periodic traffic) is becoming increasingly infeasible, in the presence of coexisting WiFi with bursty and bandwidth-hungry traffic. *Optimizing the performance* of these coexisting networks (e.g., WiFi throughput maximization, while satisfying 802.15.4 deadlines) has been a challenging task, primarily due to the lack of: i) analytical models that take into consideration realistic network traffic conditions; and ii) accurate simulators for coexistence. In this paper, we address the aforementioned research challenges by modeling the transmission buffers of wireless devices as $M/G/1$ queues, and employ queuing theory and Markov Chain models to derive, for the first time, closed form solutions for throughput and delay in 802.11/802.15.4 coexisting networks. Using our proposed models, this paper presents a novel approach for joint MAC protocol tuning, that maximizes 802.11 throughput while satisfying delay constraints of 802.15.4. We validate our proposed solutions and models through new 802.11/802.15.4 coexistence capabilities in the ns-3 simulator (important for the research community).**

I. Introduction

In recent years, the massive increase in the number of wireless network deployments has led to the spectral crowding of the unlicensed 2.4GHz ISM band. WiFi network traffic is mostly contributed by mainstream/entertainment applications that exploit the high data rate provided at the 802.11 MAC layer. Wireless Sensor Networks (WSN) traffic on the other hand, has been concentrated towards industrial/monitoring applications due to throughput limitations imposed by the IEEE 802.15.4 standard. The increased proliferation of both types of wireless networks demands the study of their behavior in coexisting environments. Additionally, with the advent of newer standards (e.g., the energy modulation based 802.11n, ac) as well as long range 802.15.4 radios (e.g. the Freescale ZigBee range extender), the single-cell wireless coexistence is expected to become pervasive in the future.

Coexistence issues have been reported and have shown the significant impact of IEEE 802.11 and 802.15.4 networks on each others' performance [2] [3], resulting in undesired throughput degradation and a significant increase in delays. On one hand, longer transmission durations of 802.15.4 devices (because of lower data rate) result in throughput degradation of 802.11 networks, negatively influencing WiFi users' experience. 802.15.4 devices, on the other hand, suffer due to the non-aggressive nature of the MAC protocol they employ (when compared to 802.11). Consequently, satisfying *delay constraints*, which are a necessity for WSN applications like health monitoring in hospitals and environmental monitoring for hazardous substances [4], becomes rather challenging. This single-cell coexistence problem to optimize the throughput of 802.11 while ensuring application delay constraints for 802.15.4 motivates the need for an accurate model for the coexistence networks with realistic traffic patterns.

Recently [1] proposed the first analytical model for single-cell coexistence of 802.11 DCF and 802.15.4 (as BoX-MAC [5] for simplicity) networks. Though it demonstrated promising results, the analysis has two major drawbacks: i) the Markov Chain model made improper assumptions on some probabilities (discussed in Section III), which hurt its accuracy; and ii) it assumed saturated traffic, leaving unsaturated/bursty traffic conditions as an open research problem. We address these problems by developing a new Markov Chain model which not only relaxes the strong assumptions aforementioned, but also introduces the probabilities for the transmission buffers of 802.11/BoX-MAC being empty to account for unsaturated traffic. We model the buffers as $M/G/1$ queues to accurately compute the buffer empty probabilities. We then derive closed form solutions for normalized throughput and total packet delivery delay. It is worthwhile to emphasize that this paper investigates an important, difficult research problem (i.e. coexistence with unsaturated traffic) that is part of a much more complicated problem, namely performance optimization of coexisting wireless networks with duty-cycling techniques such as 802.11 PSM [6] and BMAC LPL [7]. We intend to provide a stepping stone towards modeling/analysis of their duty-cycling behavior under coexistence.

The contributions of this paper are as follows: 1) it presents a joint MAC protocol tuning method (computationally facile), that maximizes WiFi throughput while satisfying 802.15.4 packet delivery delay constraints; 2) it proposes a new accurate Markov Chain model to account for unsaturated traffic; 3) it proposes a $M/G/1$ queueing model that can accurately predict buffer empty probabilities; 4) it presents the first analysis and closed form expressions for throughput and packet delivery delay; 5) it presents the first ns-3 based coexistence simulator; and 6) it validates the accuracy of the model and MAC protocols tuning method through extensive simulations.

This paper is organized as follows. Section II presents the state of art. Section III-C introduces new models for coexistence and the queueing analysis. Section IV describes a joint protocol tuning method. In Section VI we propose our ns-3 simulator and evaluate the accuracy of proposed models

978-1-4673-8590-9/15/$31.00 ©2015 IEEE

and joint protocol tuning method. Section VII concludes and presents ideas for future work.

II. State of the Art

In the recent past, there has been significant research on Markov Chain models for CSMA based networks. [8] is the seminal work to model 802.11 DCF as a 2-D Markov Chain. [9] and [10] improve [8] by considering the retransmission limits and the freezing of the backoff counter. Moreover, various studies have extended [8] to address the unsaturated traffic [11] [12] [13]. More precisely, some works introduce to the Markov Chain the probability of buffer being empty, which is then estimated by simple probabilistic methods [13]; while others not only bring in the aforementioned probability, but also employ queueing theory ($M/G/1$, $M/G/1/K$ etc.) to compute it more rigorously [11] [12]. However, these methods have only been applied to 802.11, whereas, we tackle a more challenging problem, that of coexistence analysis.

There have been approaches proposed to mitigate the problems in coexisting 802.11 and 802.15.4 networks. One approach introduced the concept of orthogonal channel assignment for WiFi and WSN devices [14], which is rendered ineffective in high density deployments. Other approaches were based on exploiting WiFi and WSN signal properties [15]. Most of these approaches, although provide promising results, are limited to non-single-cell coexistence (i.e., WiFi can harm ZigBee, but not vice versa, due to the power level asymmetry) and hence are not applicable to single-cell scenario.

Recently, an analytical model for coexistence was proposed [16]. Using existing Markov Chain models for 802.11 [8] and 802.15.4 [17], a combined model was formulated, and mathematical expressions for aggregate throughput were derived. The model was evaluated using a newly built Monte Carlo based coexistence simulator. [1] improves upon [16] by predicting channel busy probabilities more accurately and proposing two contention window tuning methods to achieve QoS and fairness. However, improper assumptions of some key probabilities attenuate the fidelity of these models.

This paper investigates a different problem, that of 802.11 throughput maximization while meeting the 802.15.4 packet delivery delay guarantees. For this, we first change the Markov Chains in [1] significantly to enhance the accuracy, and then, to model the unsaturated traffic we introduce the buffer empty probabilities, which are predicted by a $M/G/1$ queueing model. We validate the accuracy of our proposed models and MAC protocols tuning method through extensive simulations using our novel extensions for coexistence to the well known ns-3 simulation framework.

III. Model for 802.11 and 802.15.4 Coexistence with Unsaturated Traffic

In this section, we first of all briefly describe the BoX-MAC standard and some key assumptions of our Markov Chain (MC) model. Then we introduce the MC model in detail (it is worth noting that, this MC model, which, is quite different from the ones in [16] and [1], significantly improves

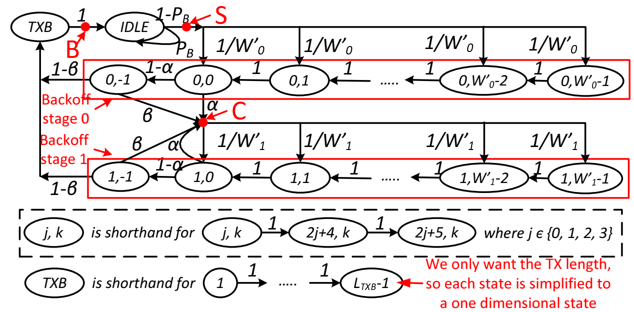

Fig. 1: Markov Chain for the BoX-MAC protocol

the accuracy). Finally we cover the $M/G/1$ queueing model and derive the expressions of throughput and delay.

A. *Preliminaries and Assumptions*

We consider BoX-MAC in this paper because it is the MAC protocol of TinyOS, and yet a simplified version of 802.15.4. Generally, 802.15.4 has several double sized contention window (CW) backoff stages up to a maximal one (e.g. CW=8, 16, 32 etc.), while BoX-MAC only uses two such backoff stages (shown as CW=W'_0, W'_1 in Figure 1). Same as 802.15.4, BoX-MAC employs the double-channel-sensing (DCS) mechanism, i.e. a device transmit a packet only if the channel stays idle for two continuous time slots.

We assume that all devices employ *energy based modulation*, and are within a single wireless cell. The same type of devices are homogeneous, i.e. their traffic pattern are Poisson with equal λ , and they transmit *packets of equal size* L (i.e. λ_i=λ_j and L_i=L_j if nodes i and j are of same type). For simplicity, we also consider an *ideal channel* (i.e. no shadowing, fading and capture effect), implying communication fails only due to collisions. Moreover, we make two fundamental, yet widely used assumptions on the probabilities of a reference device: *a) the probability of a transmission attempt is constant and independent of the attempts of other devices; and b) the collision probability (conditioning on an attempt) is constant and independent of the number of collisions experienced. These assumptions were proven to be accurate [8]- [13].*

B. *Markov Chain models for coexisting BoX-MAC and 802.11*

1) BoX-MAC: The MC for BoX-MAC is shown in Figure 1, where each state is two dimensional $(s(t), c(t))$. Specifically, $s(t)$ is a stochastic process representing the current backoff stage (i.e. two backoff stages as 0 and 1); while $c(t)$ is a process representing the current backoff counter in the corresponding stage ($0 \leq c(t) \leq W'_0$ or W'_1). Notably, in the dashed-line box in Figure 1, since BoX-MAC has a time slot 3 times longer than 802.11 (i.e. 802.11 time slot is our baseline), to account for the difference in slot sizes, for each state (j, k) (*not including the states for double-sensing, i.e.* $(0, -1)$ *and* $(1, -1)$*, because double-sensing is generally fast, and we assume each such state takes one baseline time slot*) in the BoX-MAC MC, we add two dummy states to it (i.e. $(2j{+}4, k)$, and $(2j{+}5, k)$). Thus the node entering state (j, k) will transit to state $(2j{+}4, k)$, and then to $(2j{+}5, k)$ with probability 1, thereby accounting for an entire BoX-MAC slot. *For clear presentation, we omit the two dummy states henceforth.*

In the BoX-MAC protocol, a device starts by choosing a random number from 0 to $W_0'-1$, and then begin with the backoff procedure by decrementing the backoff counter. In the Markov Chain, this behavior corresponds to starting from state $(0,k)$, where k is a random number between 0 and $W_0'-1$, and then move to the $(0,k-1)$ and so on so forth, until it reaches states $(j,0)$ and $(j,-1)$, where the double-channel-sensing is performed. If the channel is sensed busy (with probability $\alpha+(1-\alpha)\beta$, where α and β are the probabilities that the node finds the channel busy for the first and the second time, respectively), the device transitions to state $(1,k)$ where k is a random number between 0 and $W_1'-1$. If the channel is sensed idle in both states $(j,0)$ and $(j,-1)$, the packet is transmitted. Transmission is represented by TXB, which includes L_{TXB} tandem states with all transition probabilities are equal to one (L_{TXB} is the duration of a BoX-MAC transmission). Since two-dimension for TXB is meaningless, each state is simplified to one dimensional. Unlike the saturated traffic scenario, for this unsaturated case, the probability that the transmission queue being empty is not zero, and we denote by P_B such probability. Thus after a transmission, a device enters $IDLE$ state to check the status of its queue, i.e. wherever there is no packet in the queue (w.p. P_B), it stays in the $IDLE$; otherwise, it begins a new backoff process immediately. *Note that since β is conditioned on the success of the first channel sensing, it differs from [1] where $\beta=\alpha$ is improperly assumed.*

The next step is deriving the expression for the probability that a BoX-MAC device attempts to sense the channel for the first time, i.e. ϕ_B. Notably, ϕ_B is the another difference from [1] where the probability that a BoX-MAC node attempts to send (i.e. τ_B) is derived. *The reason we use ϕ_B here is that it gives us opportunity to model the details of the coexistence, lacking of which bring inaccuracies.* Moreover, since ϕ_B is a conditional probability given that the reference node is not transmitting, the expression for it is $\phi_B=\sum_{i=0}^{1} b'_{i,0}/(\sum_{i=0}^{1}\sum_{j=0}^{W_i'} b'_{i,j}+b'_{IDLE})$, where $b'_{i,j}$ (and b'_{IDLE}) represents the stationary probability of state (i,j) (and state $IDLE$) in the MC. The denominator of the equation serves as the condition for ϕ_B because it includes all states that the reference node is not transmitting. Then by similar method in [8], we can derive an elegant form of ϕ_B as

$$\phi_B=\frac{2(1-P_B)}{3((3+(1-\alpha)\beta+\alpha)(1-P_B)+2(1-\alpha)(1-\beta))} \quad (1)$$

As we can see, Equation (1) needs three unknown probabilities, i.e. α, β and P_B. For the derivations of the former two, we will leverage a MC for the channel (shown later), while for P_B, a queueing analysis will be used.

2) 802.11: The Markov Chain for 802.11 is shown in Figure 2. Unlike the MC for BoX-MAC and the one in [1], each state in this one is three dimensional, i.e. $(s(t),c(t),p(t))$. The first two stochastic processes have same meaning as BoX-MAC (the backoff behavior is also similar expect there are m stages instead of 2 for 802.11, and for conciseness the details are not elaborated here), while the third one $p(t)$ represents the channel status in the previous time slot (i.e. $p(t)=1$ or 0, corresponds to busy or idle). Generally, the purpose of

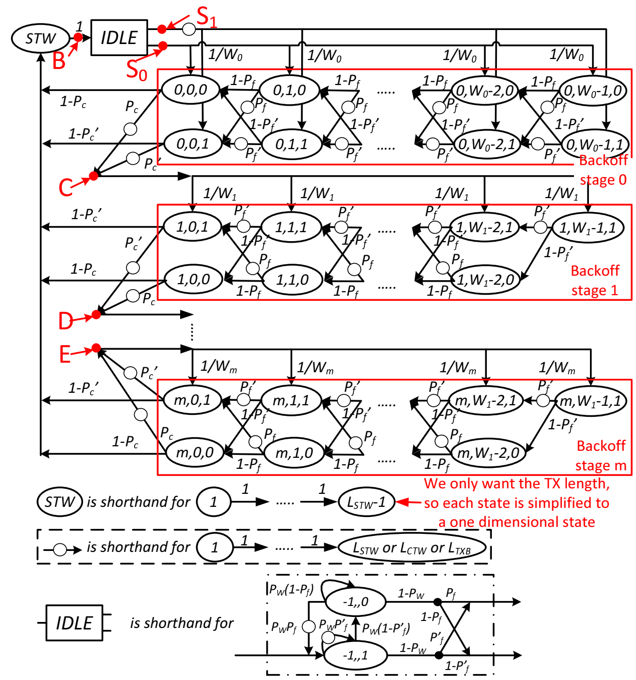

Fig. 2: Markov Chain describing the 802.11 protocol.

$p(t)$ is to describe the event that the channel being idle for two continuous baseline time slots, which is critical for BoX-MAC devices because they can transmit data only if the double-channel-sensing is passed. Notably, for each backoff stage in Figure 2 (i.e. each solid line box), the bigger circles represent the time instances that the channel is idle, while each smaller circle denotes an ongoing transmission by other devices (either BoX-MAC or WiFi or both, see the dashed line box in Figure 2). The probability P_f is the transition probability from state $(i,j,0)$ to a channel busy state (i.e. at least one other node is sending). Since $(i,j,0)$ means that its previous time slot is idle, which implies that there are two continuous idle time slots, BoX-MAC devices have chance to transmit. However, from state $(i,j,1)$, since there is single idle slot for the channel, BoX-MAC cannot seize the channel, thus the corresponding P_f' is not equal to P_f. Note that for states where the reference device attempts to transmit, i.e. states $(1{\sim}m,0,0{\sim}1)$, the corresponding transition probabilities are denoted as P_c and P_c' (i.e. collision probability), and since both P_c and P_f imply the same transition probability, we have $P_c=P_f$ (similarly, $P_c'=P_f'$).

The state $IDLE$ is a shorthand for several states (shown in the dot-dash line box in Figure 2) according to the behavior of 802.11. When the transmission queue is empty (with probability P_W), the device has to wait in $(-1,,0)$ or $(-1,,1)$ state. Note that here we denote by $s(t)=-1$ the queue being empty (abused a little bit), and as before $p(t)=0$ or 1 depends on the channel status of the last time slot. Since the transitions between $(-1,,0)$ and $(-1,,1)$ are fairly simple (e.g. the self-loop for $(-1,,0)$ happens only if the queue is empty and no other nodes is transmitting, i.e. $P_W(1-P_f)$), the elaboration is omitted here. When the queue is not empty (w.p. $1-P_W$) and if no one is using the channel (w.p. $1-P_f$ or $1-P_f'$), the

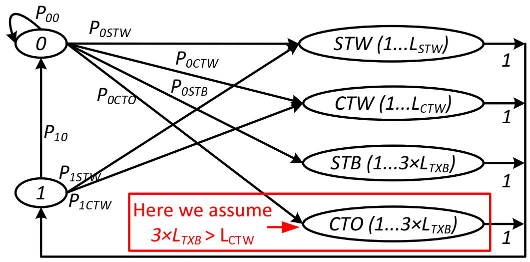

Fig. 3: Markov Chain describing the channel.

device will begin with the backoff process from $(0,j,0)$; if the channel is being used (w.p. P_f or P_f'), the device simply wait until the transmission finishes and begin to backoff from $(0,j,1)$. Notably for backoff stages other than 0, there is only one input (i.e. to $(1\cdots m,j,1)$) because from their (i.e. stages 1 to m) perspectives the channel is always busy in the previous time slot due to the collided transmission.

Before deriving P_f and P_f', we first of all discuss the probability that a 802.11 device attempts to transmit, i.e. τ_W, which is needed in the expressions of P_f and P_f'. Similar to ϕ_B we discussed before, τ_W is also a probability conditioned on the channel being idle, thus we have

$$\tau_W=\frac{\sum_{i=0}^{m}\sum_{k=0}^{1}b_{i,0,k}}{\sum_{i=0}^{m}\sum_{j=0}^{W_i}\sum_{k=0}^{1}b_{i,j,k}+b_{IDLE}} \tag{2}$$

where $b_{i,j,k}$ (and b_{IDLE}) represents the stationary probability of state (i,j,k) (and state $IDLE$) in the MC.

Unfortunately, for such a complex MC, there is no simple close form expression for τ_W, we therefore resort to a more general method, namely the transition matrix method. Since each transition probability for the 802.11 MC is known (symbols such as P_f, P_c are also considered known because they will be replaced by initial guesses of number when using numerical method), its transition matrix $\mathbf{T}$ can be built. And for a MC being stationary, $\boldsymbol{\pi}\mathbf{T}=\boldsymbol{\pi}$ must hold, where $\boldsymbol{\pi}$ is the row vector stores the stationary probabilities of all states. Then by using the condition that $\sum\pi_i=1$, each element in $\boldsymbol{\pi}$ (i.e. π_i) can be obtained. Thus the value of τ_W can be computed.

Having obtained τ_W, we are ready to discuss P_f and P_f'. For P_f, since it is conditioned on $p(t)=0$ (two continuous idle time slots), thus BoX-MAC can transmit, then we have

$$P_f=1-(1-\phi_B)^{N_B}(1-\tau_W)^{N_W-1}=P_c \tag{3}$$

where N_B and N_W are the number of BoX-MAC and WiFi devices, respectively. While for P_f', since the previous time slot is busy, BoX-MAC has no chance to transmit, thus

$$P_f'=1-(1-\tau_W)^{N_W-1}=P_c' \tag{4}$$

As mentioned before, α and β can be derived using a Markov Chain for the channel, which is discussed next.

3) Channel: The Markov Chain describing the channel is shown in Figure 3. This MC is simple and only one dimensional. In particular, state 0 represents a currently idle channel with the previous time slot being idle, while state 1 represents a currently idle channel with the previous time slot being busy. States STW, CTW, STB and CTO are similar to STW in the MC of 802.11, and they respectively denote a successful WiFi transmission, a collision WiFi transmission, a successful BoX-MAC transmission and all other cases. Notably, we assume that the occupation of a BoX-MAC packet in the channel is longer than that of WiFi, thus the length for CTO is the length of a BoX-MAC packet. Same as the cases in the 802.11 MC, BoX-MAC can only send data when the channel is idle for two continuous time slots (i.e. state 0), thus from 0 there are four transitions to STW, CTW, STB and CTO respectively. Whereas from 1, there are only two transitions to STW and CTW because BoX-MAC has no chance to transmit in this case. According to the meaning of these transitions, we first of all define $A_0=(1-\phi_B)^{N_B}$ (i.e. no BoX-MAC attempts to send), $A_1=N_B\phi_B(1-\phi_B)^{N_B-1}$(i.e. only one BoX-MAC attempts to send), $B_0=(1-\tau_W)^{N_W}$ (i.e. no WiFi attempts to send) and $B_1=N_W\tau_W(1-\tau_W)^{N_W-1}$ (i.e. only one WiFi attempts to send), and then we easily have $P_{00}=A_0\cdot B_0$, $P_{0STW}=A_0\cdot B_1$, $P_{0CTW}=A_0\cdot(1-B_0-B_1)$, $P_{0STB}=A_1\cdot B_0$ and $P_{0CTO}=1-A_0-P_{0STB}$ for state 0, and $P_{10}=B_0$, $P_{1STW}=B_1$ and $P_{1CTW}=1-B_0-P_{1STW}$ for state 1. By solving this simple MC (using the same method as in [8]), and denoting $P_{0STW}+P_{0CTW}+P_{0STB}+P_{0CTO}$ by P_{0X} as well as $P_{1STW}+P_{1CTW}$ by P_{1X} for simplicity, we obtain the stationary probabilities of state 0 and 1 as $b_0=\frac{1-P_{00}}{P_{10}(1+P_{0X})+(1-P_{00})(1+P_{1X})}$ and $b_1=\frac{P_{10}}{P_{10}(1+P_{0X})+(1-P_{00})(1+P_{1X})}$.

Then by the definition of α, i.e. the probability that the channel being busy, then $\alpha=1-$P(channel idle), thus

$$\alpha=1-(b_0+b_1) \tag{5}$$

The derivation of β is little tricky because it is a probability of channel being busy in current time slot conditioned on it was idle in last time slot, namely $P(b|i)$. What we do is utilizing the law of total probability i.e. $P(b|i)=P(b,i)/P(i)=(P(i_0)P(b|i_0)+P(i_1)P(b|i_1))/P(i)$, which gives us

$$\beta=\frac{b_0(1-A_0\cdot B_0)+b_1(1-B_0)}{b_0+b_1} \tag{6}$$

So far, we have eight unknowns (i.e. ϕ_B, α, β, τ_W, P_f, P_f', P_B, P_W) and six Equations (1)-(6). We move to queueing theory to model the transmission buffers of BoX-MAC and 802.11 as two $M/G/1$ queues to find the remaining two expressions for P_B and P_W. Note that in this paper we use the terms buffer and queue interchangeably.

C. $M/G/1$ queue models for BoX-MAC and 802.11

We first define λ_B and μ_B as the arrival rate and the service rate for the queue of a BoX-MAC node, and λ_W and μ_W as the same for a 802.11 node. For $M/G/1$ queue, we have $P_B=1-\rho_B$ and $P_W=1-\rho_W$, where ρ_B and ρ_W are the traffic intensities (defined as $\rho_B=\lambda_B/\mu_B$ and $\rho_W=\lambda_W/\mu_W$), and since the arrival process is Poisson, we only need to get the expressions for μ_B and μ_W.

Since service processes are based on the CSMA behavior, which is generally distributed, there is no simple expressions for them. However, as mentioned before, what we really need is the service rate, i.e. the mean of the service time. In the following subsections, we first of all introduce the concept of probability generating function (PGF) and how it is related to mean and variance. Then by using PGF, we derive expressions for the mean (i.e. μ_B and μ_W) and variance (i.e. σ_B^2 and σ_W^2). Notably the variance is needed in the computation of delay.

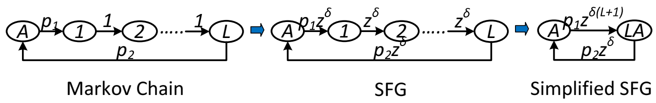

Fig. 4: Signal Flow Graph of a Markov Chain.

1) Probability generating function (PGF): PGF is a mathematical tool to describe a discrete random variable (r.v.). Assume X is a discrete r.v. taking values in $\{0, 1, ...\}$, then its PGF is defined as: $G(z){=}\sum_{x=0}^{\infty} p(x)z^x$, where p is the probability mass function of X. PGF has two important properties, i.e. the expectation of X is given by $\mathrm{E}\left(X\right){=}\frac{dG(z)}{dz}|_{z=1}{=}G'(1)$, while the variance of X is given by $\mathrm{Var}(X){=}G''(1){+}G'(1){-}(G'(1))^2$, where $G''(1){=}\frac{d^2G(z)}{dz^2}|_{z=1}$.

Moreover, as shown in [12], the PGF for any given two different states in a MC is the transfer function (TF) of the corresponding signal flow graph (SFG), which can be obtained by taking Z-transform for each transition probability (along with the time spent in the transition) in the MC. Then we can apply the well-known Mason formula to the SFG to derive the PGF. Figure 4 shows a MC and its SFG, where the SFG is very similar to the MC, except that each transition probability is replaced by its Z-transform (i.e. $p{\cdot}z^{\delta}$, where p is the transition probability and δ is the duration of a baseline time slot). Furthermore, by the property of Z-transform, the SFG can be simplified by combining all states with transition probability 1. For example, the states 1 to L can be simplified as a single state LA, and the PGF between A and L (without the loop) becomes $p_1z^{(L+1)\delta}$. *Note that $z^{(L+1)\delta}$ here actually means the overall transition between A and L uses $L{+}1$ time slots, which is accurate*. Now, by the Mason formula, we can obtain the overall PGF from A to L as $G_{AL}(z){=}\frac{p_1z^{(L+1)\delta}}{1-p_1z^{(L+1)\delta}p_2z^{\delta}}$.

2) BoX-MAC MAC service time characteristics: Let T_B be the r.v. representing the MAC service time of BoX-MAC. As before, to compute $\mathrm{E}(T_B)$, we need to obtain the PGF of T_B, i.e. $BG(z){=}\mathrm{E}(z^{T_B})$ by using the SFG of the BoX-MAC MC (Figure 1). The SFG is generated by simply adding z to each transition probability in Figure 1 (to make the presentation concise, a figure for the SFG is omitted here). *It is worth noting that since T_B is the MAC service time, the PGF of T_B is actually the transfer function between point S and B in Figure 1*. Since the entire SFG is a little complex, we first of all derive the PGF between point S and state $(0,0)$ (using the Mason formula) as $BG_{0,0}(z){=}\frac{z^{3\delta}}{W_0'}\frac{1-z^{3\delta\cdot W_0'}}{1-z^{3\delta}}$, where 3δ represents a BoX-MAC time slot which is 3 times longer than the baseline (802.11) one. Then since the PGF between $(0,0)$ and $(-1,0)$ is simply $(1{-}\alpha)z^{\delta}$, the PGF between S and $(0,-1)$ is $BG_{0,-1}(z){=}BG_{0,0}(z){\cdot}(1{-}\alpha)z^{\delta}$. Using similar method (note that due to the self loop of C, the derivation is a little harder), we can have the PGF between point C and $(1,-1)$ as $BG_{1,-1}(z)$. Finally by combining all the PGF's of different parts, the overall PGF (between S and B) can be expressed as $BG(z){=}BG_{0,0}(z)(1{-}\alpha)z^{\delta}(1{-}\beta)z^{3\delta L_{\mathrm{TXB}}}{+}BG_{0,0}(z)(\alpha z^{\delta}{+}(1{-}\alpha)\beta z^{2\delta})BG_{1,-1}(z)(1{-}\beta)z^{3\delta L_{\mathrm{TXB}}}$.

Then by the property of PGF, we have $1/\mu_B{=}\mathrm{E}(T_B){=}BG'(1)$, thus

$$P_B{=}1{-}\lambda_B{\cdot}BG'(1) \tag{7}$$

3) 802.11 MAC service time characteristics: We define T_W as the r.v. denoting the MAC service time of 802.11. Similar to the analysis for BoX-MAC, here the PGF (i.e. $WG(z){=}\mathrm{E}(z^{T_W})$) of T_W is needed to obtain $E(T_W)$. The same method as before is used to convert the MC for 802.11 (in Figure 2) to its corresponding SFG. To get the expression for the PGF between point S_0 (or S_1) and B, we also employ the divide-and-conquer strategy, i.e. first expressing the TF's from S_0 (or S_1) to C, then C to D and so on, and finally combining them to generate the overall PGF.

However, due to the increased complexity of the SFG, the derivation for the PGF between S_0 (or S_1) and C is more difficult than that for BoX-MAC. We first of all define the PGFs from S_0 to two states with same backoff counter (e.g. S_0 to $(0,j,0)$ and to $(0,j,1)$) as a row vector $\mathbf{WG0_{0,j}}(z)$.Then the PGF from S_0 to $(0,W_0{-}1,0)$ and to $(0,W_0{-}1,1)$ (for demonstration purpose) is $\mathbf{WG0_{0,W_0-1}}(z){=}\left[z/W_0,0\right]$. From $\mathbf{WG0_{0,W_0-1}}(z)$, $\mathbf{WG0_{0,W_0-2}}(z)$ is also obtained, and so on so forth, until $\mathbf{WG0_{0,0}}(z)$ is derived. Specifically, based on the transitions between $\mathbf{WG0_{0,j}}(z)$ and $\mathbf{WG0_{0,j-1}}(z)$, we define a matrix which describes the TF between them as $\mathbf{Q0}{=}\begin{bmatrix}(1{-}P_f)z & a(z)\\(1{-}P_f')z & b(z)\end{bmatrix}$, where $a(z){=}A_0C_1z^{\delta L_{\mathrm{STW}}}{+}A_0(1{-}C_0{-}C_1)z^{\delta L_{\mathrm{CTW}}}{+}(1{-}A_0{-}A_1C_0)z^{3\delta L_{\mathrm{TXB}}}$ and $b(z){=}P_f'z^{\delta L_{\mathrm{STW}}}{+}(P_f'{-}C_1)z^{\delta L_{\mathrm{CTW}}}$, where A_0, A_1 are defined in Section III-B3, and C_0 and C_1 are defined as $(1{-}\tau_W)^{N_W-1}$ and $(N_W{-}1)\tau_W(1{-}\tau_W)^{N_W-2}$, respectively. Although the expressions above look tedious, they are simple Z-transforms of all different transition paths between $\mathbf{WG0_{0,j}}(z)$ and $\mathbf{WG0_{0,j-1}}(z)$. With $\mathbf{Q0}$ (note that $\mathbf{I}{-}\mathbf{Q0}$ is invertible), we note that $\mathbf{WG0_{0,0}}(z){=}\mathbf{WG0_{0,W_0-1}}(z){\cdot}(\mathbf{I}{+}\mathbf{Q0}{+}{\cdots}{+}\mathbf{Q0^{W_0-1}}){=}[z/W_0,0]{\cdot}\frac{\mathbf{I-Q0^{W_0}}}{\mathbf{I-Q0}}$. Since the PGF (denoted by $\mathbf{TG}(z)$) between $[(0,0,0),(0,0,1)]$ and point C is $\mathbf{TG}(z){=}\left[a(z){-}A_0C_1z^{\delta L_{\mathrm{STW}}}, b(z){-}P_f'z^{\delta L_{\mathrm{STW}}}\right]^T$ (note $P_c{=}P_f$ and $P_c'{=}P_f'$), we define the PGF for backoff stage 0 (represented by $WG0_0(z)$) as the PGF between S_0 and C. We then have $WG0_0(z){=}\mathbf{WG0_{0,0}}(z){\cdot}\mathbf{TG}(z)$. By similar method, the PGF for backoff stages 1 to $m{-}1$ can be derived. For example, the PGF for stage 1 (i.e. between C and D) is $WG0_1(z){=}\mathbf{WG0_{1,0}}(z){\cdot}\mathbf{TG}(z)$, where $\mathbf{WG0_{1,0}}(z)$ is the PGF between C and $[(1,0,0),(1,0,1)]$ and $\mathbf{TG}(z)$ is the PGF from $[(1,0,0),(1,0,1)]$ to D. Note that $\mathbf{TG}(z)$ does not change with backoff stage.

Same as BoX-MAC, the PGF for stage m is a little tricky because of the self-loop. The derivation, however, is still trivial by using the Mason formula. The corresponding result is as

$$\mathbf{WG0_{m,0}}(z){=}\left[\frac{\mathbf{WG0'_{m,0}}^{(1)}(z)}{1{-}\mathbf{WG0'_{m,0}}^{(1)}(z){\cdot}\mathbf{TG}^{(1)}(z)},\frac{\mathbf{WG0'_{m,0}}^{(2)}(z)}{1{-}\mathbf{WG0'_{m,0}}^{(2)}(z){\cdot}\mathbf{TG}^{(2)}(z)}\right]$$

where the superscript (i) represent the ith element in a vector, and $\mathbf{WG0'_{m,0}}$ is the forward PGF (i.e. without the self-loop) from point E to $[(m,0,0),(m,0,1)]$.

Having had the PGF's for all stages, the final PGF from S_0 to B can be derived using Mason formula as below

$$WG0(z)=\mathbf{WG0_{0,0}}(z)\cdot\left[(1-P_f)z^{\delta L_{\text{STW}}},(1-P_f')z^{\delta L_{\text{STW}}}\right]^T$$
$$+\sum_{i=1}^{m}\left(\mathbf{WG0_{i,0}}(z)\cdot\left[(1-P_f)z^{\delta L_{\text{STW}}},(1-P_f')z^{\delta L_{\text{STW}}}\right]^T\cdot\prod_{j=0}^{i-1}WG0_j(z)\right)$$

By similar derivation as above, the PGF from S_1 to B (i.e. $WG1(z)$) is obtained. Then by the property of PGF, we have

$$P_W=1-\lambda_W\cdot\left(P_0\cdot WG0'(1)+P_1\cdot WG1'(1)\right) \quad (8)$$

where $P_0=(b_{IDLE,0}\cdot(1-P_f)+b_{IDLE,1}\cdot(1-P_f'))/b_{IDLE}$ and $P_1=(b_{IDLE,0}\cdot P_f+b_{IDLE,1}\cdot P_f')/b_{IDLE}$, which denote the probabilities that the outputs of $IDLE$ state being S_0 and S_1, respectively.

Now, by employing numerical methods to solve Equations (1)-(8), all eight unknown probabilities are obtained. Several performance metrics can be derived from these probabilities. In this paper, we are interested in the aggregate throughput and the total delay, i.e., the time between a packet entering the queue until it is transmitted successfully.

D. Throughput Analysis for Coexisting BoX-MAC and 802.11

With the help of the Markov Chain model for the channel shown in Section III-B3, it is straightforward to derive the expressions for the throughput of BoX-MAC and 802.11.

1) Throughput for BoX-MAC: From the perspective of the channel, the throughput of BoX-MAC is simply the time spent in a successful BoX-MAC transmission, thus we have

$$S_B=b_0P_{0STB}3Lp_{\text{TXB}}$$

where b_0 is the stationary probability for state 0 in the channel MC (Section III-B3), P_{0STB} is the transition probability for a BoX-MAC transmission being succeed, and $3Lp_{\text{TXB}}$ is packet payload size of BoX-MAC in baseline time slot.

2) Throughput for 802.11: Similar to BoX-MAC, we can express the 802.11 throughput as

$$S_W=(b_0P_{0STW}+b_1P_{1STW})Lp_{\text{STW}} \quad (9)$$

where b_1 is the stationary probability for state 1 in the channel MC (Section III-B3), P_{0STW} is the transition probability for an 802.11 transmission being succeed, and Lp_{STW} is packet payload size of 802.11 in baseline time slot.

E. Delay Analysis for Coexisting BoX-MAC and 802.11

By Little's Law, delay $D=L/\lambda$, where L is the steady state queue length and λ is the arrival rate. Consequently, to compute the delay we need to obtain the steady state queue length of the BoX-MAC and 802.11. For a $M/G/1$ queue, the expression for the average queue length is $L=\rho+\frac{\lambda^2(\sigma^2+1/\mu^2)}{2(1-\rho)}$ [18], where σ^2 is the variance of the service time, and $\rho=\lambda/\mu$.

Since we obtain the PGF's of the MAC service time of BoX-MAC and 802.11 in Section III-C2 and III-C3, the corresponding variances are $\sigma_B^2=\text{Var}(T_B)=BG''(1)+BG'(1)-(BG'(1))^2$ and $\sigma_W^2=\text{Var}(T_W)=WG''(1)+WG'(1)-(WG'(1))^2$. Then the expressions for the queue length are $L_B=\rho+\frac{\lambda_B^2(\sigma_B^2+1/\mu_B^2)}{2(1-\rho_B)}$ and $L_W=\rho+\frac{\lambda_W^2(\sigma_W^2+1/\mu_W^2)}{2(1-\rho_W)}$, respectively.

With average MAC service time, we can simply write the expressions for delay (by Little's Law) as $D_W=L_W/\lambda_W$ and,

$$D_B=L_B/\lambda_B \quad (10)$$

IV. Joint Protocol Tuning for Coexisting 802.11 and 802.15.4 with Delay Constraints

In this section we present a protocol tuning method that maximizes 802.11 throughput while satisfying delay constraints of 802.15.4, by varying the CW size of individual nodes. CW affects throughput since it directly controls the transmission probability (i.e. aggressiveness) [19]. As previously demonstrated [16] [1], the congested CW size of BoX-MAC (W_1') and the minimum CW size of 802.11 (W_0) have a significant impact on the achievable throughput in coexisting networks, while the initial CW size of BoX-MAC (W_0') and the maximum CW size of 802.11 (W_m) do not. Thus, we only consider tuning W_1' and W_0, and fix the other two. Our tuning method also adjusts the WiFi data arrival rate λ_W since it is the only way to tune the throughput under unsaturated condition.

Since we want to ensure that the performance (i.e. throughput) of WiFi devices be maximized under the delay constraint of BoX-MAC, we formulate this as a non-linear optimization problem,

$$\begin{aligned}&\text{argmax} && S_W(W_0,W_1',\lambda_W)\\ &\text{subject to} && D_B\leq\text{Delay Threshold},\\ &&& \text{Equations}(1)-(8),\\ &&& W_1',W_0,L_B,L_W\geq 0\end{aligned}$$

where the throughput of 802.11 S_W (see Equation (9)) is the objective function, while the delay threshold of BoX-MAC (see Equation (10) for D_B) and Equations (1)-(8) serve as the main constraints. More importantly, the CWs and queue lengths (i.e. L_W, L_B) for all nodes must be greater than or equal to 0, so that the queue of each node is stable (i.e. arrival rate $<$ service rate). By solving this problem numerically (e.g. *fmincon* toolbox in MATLAB), we obtain the optimal CW sizes for BoX-MAC and optimal CW's and arrival rates λ_W for WiFi. This tuning method is centralized, and due to the homogeneity of devices of same type, the optimal parameters for same type are equal. Thus this method does not exhibit fairness issues for the same type of devices. We note that ensuring fairness between BoX-MAC and 802.11 is not appropriate because of the very large difference in their maximal flow rates.

V. Validation of Model for Coexistence

We implemented our coexistence simulator based on the well-known ns-3 simulator. The spectrum PHY module in ns-3 is used as the common operating channel for both 802.11 and BoX-MAC devices. The implementation involved significant modifications at different layers of the network architecture of WiFi and LrWPAN modules of ns-3, to handle homogeneous and heterogeneous collisions along with support for unsaturated traffic conditions. We compare the throughput and delay obtained from our analytical model with those obtained from the ns-3 simulator.

Extensive simulations were performed under varying configuration parameters in order to characterize the effects of CW size tuning on aggregate throughput and total delay. The parameters we vary are: number of nodes, the minimum contention window size W_0 of 802.11, the congested contention window size W_1' of BoX-MAC and the corresponding

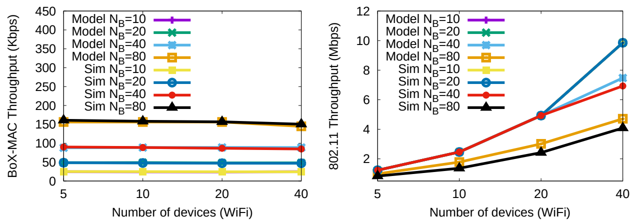

Fig. 5: Throughput of model and simulator vs # devices.

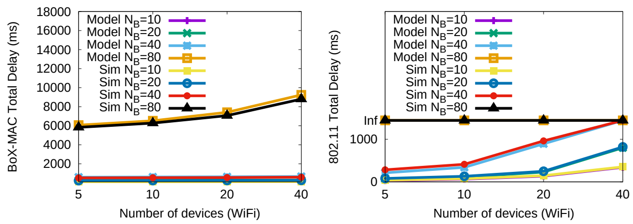

Fig. 6: Total delay of model and simulator vs # devices.

per-node offered load (data arrival rate) λ_W and λ_B. The default values of these parameters are: W_0=16, W_m=1024, P_W=1500, λ_W=20 for WiFi, and W_0'=310, W_1'=70, P_B=48, λ_B=4 for BoX-MAC.

Typically, the time spent for the model to numerically converge is less than one minute on a fast PC (with i7 4790K CPU and 16GB ram), while solving the optimization problem takes a little longer than that. The simulation time for all tests is 10000s, which, on average, takes more than ten hours on the same machine.

A. *Effects of number of devices*

Analyzing the coexistence of WiFi and WSN by varying the number of devices in a real implementation is tedious and time consuming. The same can be done by merely varying the number of devices in the analytical model and simulator. Figure 5 and Figure 6 depict the effect on throughput and delay when the number of WiFi (i.e., $\{5, 10, 20, 40\}$) and BoX-MAC (i.e., $\{10, 20, 40, 80\}$) devices are varied, while setting other parameters to their defaults. We surprisingly observe that the results for the simulator and analysis match each other closely. Specifically, when the network density is small (e.g. 20 BoX-MAC vs 10 WiFi), the throughput of both types increase linearly with the increase of the number of devices due to unsaturated queueing condition. For a crowded network (e.g. 80 BoX-MAC vs 40 WiFi), the BoX-MAC reach near-saturated condition (we have chosen λ_B=4 on purpose to demonstrate), and the 802.11 devices reach saturated condition. Thus, we observe that the throughput of BoX-MAC still increase linearly, however, the one for 802.11 increases slowly due to saturation. The delay exhibits similar behavior as the throughput. Especially when the number of devices becomes large, the delay of BoX-MAC becomes large due to near-saturated condition, while the queue for 802.11 tends to infinity (denoted by "Inf" in the figure) due to saturation.

B. *Effects of contention window sizes*

To analyze the impact of contention window size on throughput and delay, we consider a scenario with 10 802.11 and 20 BoX-MAC devices. All default parameters are used

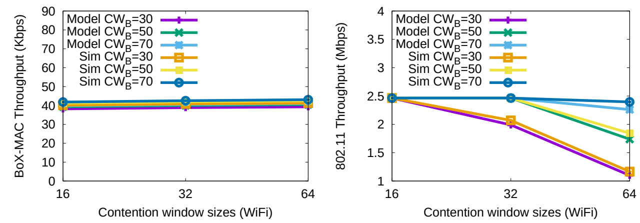

Fig. 7: Throughput of model and simulator vs CW sizes.

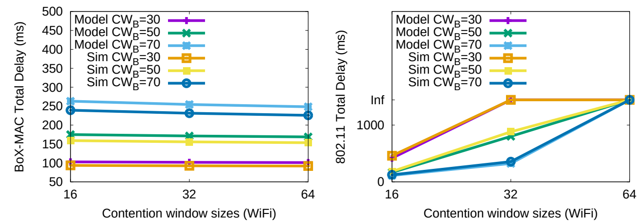

Fig. 8: Total delay of model and simulator vs CW sizes.

expect the CW size, which are selected within $\{30, 50, 70\}$ and $\{16, 32, 64\}$ for BoX-MAC and 802.11, respectively. The results are depicted in Figure 7 and Figure 8 for throughput and delay, respectively. Remarkably, the results obtained from the model agree with the simulation quite well. In the case of throughput, since the CW's for BoX-MAC always lead to unsaturated condition, its throughput does not vary much. While for 802.11, since a smaller CW (such as W_0=16) implies higher aggressiveness, the queue is unsaturated, and the throughput of 802.11 equals to its input. However, when the CW of 802.11 increases, due to the decrease of the aggressiveness, its queue becomes sensitive to the aggressiveness of BoX-MAC devices. For example, if CW of BoX-MAC equals 70, W_0=64 lead to saturation, but if CW of BoX-MAC equals 30 or 50, W_0=32 tends towards saturation. For the delay analysis, since bigger CW results in longer MAC service time, the delay is higher for bigger CW. As mentioned before, the queue of 802.11 becomes saturated for some cases, hence the corresponding delay is infinite.

C. *Effects of per-node offered load*

Due to the significant impact of traffic arrival rate, i.e per-node offered load, on the network performance, we study its effect on the throughput. As earlier, we consider a scenario with 40 WiFi, 40 BoX-MAC devices and default parameters. We vary the per-node offered load in $\{2, 4, 6\}$ packets per second for Box-MAC and $\{1, 10, 20, 30\}$ packets per second for WiFi. The results are depicted in Figure 9 and Figure 10 for throughput and total delay, respectively. Amazingly, the simulation results are in close agreement with those obtained from the model. In particular, when the offered load of WiFi increases, until saturation, the throughput of both BoX-MAC and WiFi increase linearly. Then, their throughput gradually and slowly decreases due to the increased collision rate. Similar results can be observed for the delay. As the queue becomes saturated, the delay tends toward infinity.

VI. Joint Protocol Tuning Evaluation

We evaluate the tuning method proposed in Section IV using our ns-3 simulator. We demonstrate that our approach guar-

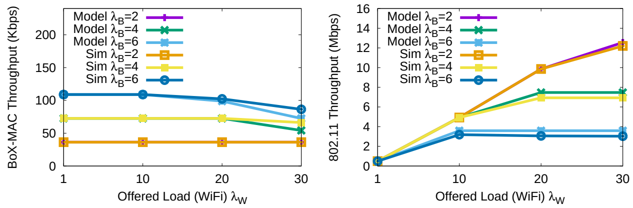

Fig. 9: Throughput of model and simulator vs offered load.

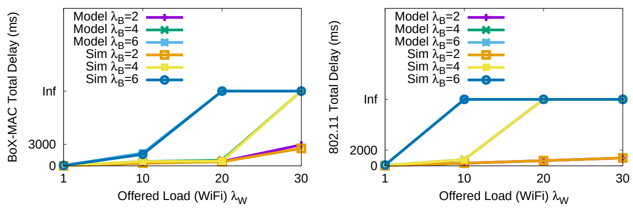

Fig. 10: Total delay of model and simulator vs offered load.

antees the delay constrains of BoX-MAC while maximizing 802.11 throughput. The evaluation of the tuning method is performed upon 20 BoX-MAC and 10 802.11 nodes, with different BoX-MAC delay constraints among $\{50, 100\}$ms.

We inspect three cases of the arrival rates λ_B for BoX-MAC from $\{2, 4, 6\}$, and obtain from the model the optimal CW sizes for BoX-MAC (i.e. W_1') and 802.11 (i.e. W_0) as well as the optimal arrival rates λ_W for 802.11. As shown in Table I, when the delay constraint for BoX-MAC is 100ms, the optimal CWs obtained from the tuning method are significantly different from their default values (i.e. W_0=16 and W_1'=70). And since λ_B is only 2, it is easy to satisfy the delay constraint, which offers WiFi a very good available throughput (which is =78). However when delay constraint becomes 50 ms, since it is harder to satisfy the constraint, the BoX-MAC becomes more aggressive by reducing its W_1', and WiFi increase the W_0 to further give space for BoX-MAC. Therefore the maximal throughput of WiFi λ_W decreases to 53. Similarly, when λ_B=4 or 6, the same trends of W_1', W_0 and λ_W can be observed in Table II and III. Especially, for an extreme case where λ_B=6 and delay constraint equals 50, the BoX-MAC becomes very aggressive (i.e. W_1'=3.1837), while WiFi reduce its aggressiveness by increasing its W_0 to 20.3774 and thus has a throughput of only 27.

In practice, this tuning method can be used on a dual radio powerful master device which first computes the optimal parameters and then informs both WiFi and BoX-MAC devices about the values obtained.

VII. Conclusions

To optimize the performance in single-cell coexisting 802.11 and 802.15.4 networks with realistic traffic patterns, we have presented the first analytical model for predicting throughput and total delay. Our analysis is anchored in solid theoretical results based on modeling coexisting 802.11 DCF and BoX-MAC as Markov Chains [1] and $M/G/1$ queueing theory. Additionally, we proposed the first protocol tuning method that maximizes 802.11 throughput while satisfying BoX-MAC delay constraints. Through extensive simulations,

TABLE I: Optimal parameters obtained under λ_B=2

Delay constraint	λ_W (802.11)	W_0 (802.11)	W_1' (BoX-MAC)
50 ms	53	8.1748	13.2673
100 ms	78	5.1457	21.3654

TABLE II: Optimal parameters obtained under λ_B=4

Delay constraint	λ_W (802.11)	W_0 (802.11)	W_1' (BoX-MAC)
50 ms	42	14.6472	6.7848
100 ms	60	9.2516	10.8520

TABLE III: Optimal parameters obtained under λ_B=6

Delay constraint	λ_W (802.11)	W_0 (802.11)	W_1' (BoX-MAC)
50 ms	27	20.3774	3.1837
100 ms	46	13.2743	7.2562

we demonstrate the correctness of our analysis and the effectiveness of the tuning method. As future work, we aim to extend our work to optimize the performance of coexisting 802.11 and 802.15.4 networks with duty-cycling devices.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant NO. (#1127449, #1145858).

References

[1] W. Zhang, M. A. Suresh, R. Stoleru, and H. Chenji, "On modeling the coexistence of 802.11 and 802.15.4 networks for performance tuning," *Wireless Comm., IEEE Trans.*, 2015.

[2] A. Sikora and V. Groza, "Coexistence of IEEE 802.15.4 with other systems in the 2.4 GHz-ISM-band," in *IMTC*, 2005.

[3] S. Pollin, B. H. I. Tan, C. Chun, and A. Bahai, "Harmful coexistence between 802.15.4 and 802.11: a measurement-based study," in *CrownCom*, 2008.

[4] P. Suriyachai, U. Roedig, and A. Scott, "A survey of mac protocols for mission-critical applications in wireless sensor networks," *Comm. Surveys & Tutorials, IEEE*, 2012.

[5] D. Moss and P. Levis, "BoX-MAC: Exploiting physical and link layer boundaries in low-power networking," *Tech. Report SING-08-00*, 2008.

[6] "IEEE standard for wireless LAN medium access control and physical layer specication, 802.11," 2007.

[7] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Sensys*, 2004.

[8] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *Selected Areas in Comm., IEEE Journal*, 2000.

[9] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement," in *INFOCOM*, 2002.

[10] Y. Xiao, "Performance analysis of priority schemes for IEEE 802.11 and IEEE 802.11e wireless LANs," *Wireless Comm., IEEE Trans.*, 2005.

[11] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *Networking, IEEE/ACM Trans.*, 2007.

[12] H. Zhai, Y. Kwon, and Y. Fang, "Performance analysis of IEEE 802.11 MAC protocols in wireless LANs," *Wireless comm. and mobile computing*, 2004.

[13] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "Unsaturated throughput analysis of ieee 802.11 in presence of non ideal transmission channel and capture effects," *Wireless Comm., IEEE Trans.*, 2008.

[14] R. Musaloiu-E and A. Terzis, "Minimizing the effect of WiFi interference in 802.15.4 wireless sensor networks," in *SenSys*, 2008.

[15] J. Huang, G. Xing, G. Zhou, and R. Zhou, "Beyond co-existence: Exploiting WiFi white space for ZigBee performance assurance," in *ICNP*, 2010.

[16] W. Zhang, M. A. Suresh, and R. Stoleru, "On modeling the coexistence of WiFi and wireless sensor networks," in *MASS*, 2013.

[17] S. Pollin, M. Ergen, S. Ergen, and B. Bougard, "Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer," *Wireless Comm., IEEE Trans.*, 2008.

[18] D. Gross, *Fundamentals of queueing theory*. John Wiley & Sons, 1998.

[19] D.-J. Deng, C.-H. Ke, H.-H. Chen, and Y.-M. Huang, "Contention window optimization for IEEE 802.11 DCF access control," *Wireless Comm., IEEE Trans.*, 2008.