Single Hop IEEE 802.11 DCF Analysis Revisited: Accurate Modeling of Channel Access Delay and Throughput for Saturated and Unsaturated Traffic Cases

Emad Felemban and Eylem Ekici

***Abstract*—Analytical models for IEEE 802.11 Distributed Coordination Function (DCF) play important roles in performance estimation, protocol optimization and admission control for wireless networks utilizing this MAC protocol. While a myriad of models for IEEE 802.11 DCF channel access delay and throughput exist, such models' accuracy is generally limited to rather narrow ranges of scenario parameters. Our investigations single out the inaccuracy in modeling the backoff process as the primary reason for the said deviations. In this paper, we introduce two highly accurate models for IEEE 802.11 DCF protocol in a single-hop setting under both saturated and unsaturated traffic loads. First, a crucial augmentation to the classical model proposed by Bianchi [1] is presented for the saturation load analysis to account for channel state during the backoff countdown process, resulting in a highly accurate estimation of collision probability, channel throughput and channel access delay. We then extend this accurate model to unsaturated traffic cases through an iterative approach which similarly results in highly accurate performance metric estimations for a wide range of parameters. Both models have been evaluated through simulations and in comparison with existing analytical models.**

***Index Terms*—Modeling, performance analysis, IEEE 802.11, DCF, saturation, non-saturation.**

I. INTRODUCTION

IEEE 802.11 protocol is widely used in today's wireless networks like hotspots and mesh networks. With increasing usage, tractable analytical models that estimate 802.11 protocol performance accurately gain more importance. These analytical models are frequently used for

- network performance estimation and network capacity calculation [1], [2],
- protocol optimization through contention window size adjustments [2], [3], and
- admission control schemes [4], [5].

Numerous attempts were made to analyze the performance of 802.11 Distributed Coordination Function (DCF) mechanism in the literature. Most of the existing models study the performance of DCF under saturation load conditions which assume that every node in the network always has a packet ready for transmission. A common problem in most existing saturation DCF analytical models is the inaccuracy in modeling the backoff process behavior of DCF, and more specifically, the backoff freezing due to channel activity. This shortcoming results in inaccurate estimations of performance metrics with different system parameters. This inaccuracy confines the usage of existing models to a very limited range of system parameters, especially for protocol optimization through contention window adjustments [2], [3].

Saturation load models for DCF have very limited practical usage since most of the applications do not operate in the saturation load region. Unsaturated load models, on the other hand, can be used in more practical applications. However, they usually have a tradeoff between complexity and accuracy due to the correlated behavior of the unsaturated nodes. In addition, they inherit the previously mentioned backoff freezing problem of saturation models. These limitations have negative impacts on the accuracy and performance of practical applications that utilize these models like admission control schemes [4], [5].

In this paper, we introduce two highly accurate models for IEEE 802.11 DCF protocol in singlehop setting under both saturated and unsaturated traffic loads. First, a crucial augmentation to the classical model proposed by Bianchi [1] is presented for the saturation load analysis. More specifically, the channel state during the backoff of a tagged node is modeled with a separate Markov chain for which the transition probabilities are estimated from the overall system model. Measures obtained from the channel state model also affect transition probabilities of the system model. Through an iterative approach, we obtain accurate descriptions of the overall system that yield in performance measures that hold for a wide range of protocol and user parameters.

This saturation model is then extended to model unsaturated load cases using an iterative algorithm to compute a binomial distribution for the number of contending nodes. This distribution is then used to compute unsaturated load performance metrics. The unsaturated load model similarly results in highly accurate performance metrics estimation, allowing it to be used in real-time applications. Both models have been evaluated through simulations and in comparison

Manuscript received July 8, 2010; revised December 24, 2010 and April 13, 2011; accepted June 2, 2011. The associate editor coordinating the review of this paper and approving it for publication was L. Libman.

E. Felemban is with the College of Computer and Information Systems, Umm Al Qura University Saudi Arabia (e-mail: eafelemban@uqu.edu.sa). This work has been done while Emad Felemban was at Ohio State University.

E. Ekici is with the Electrical and Computer Engineering Department, Ohio State University, USA (e-mail: ekici@ece.osu.edu).

Digital Object Identifier 10.1109/TWC.2011.072511.101227

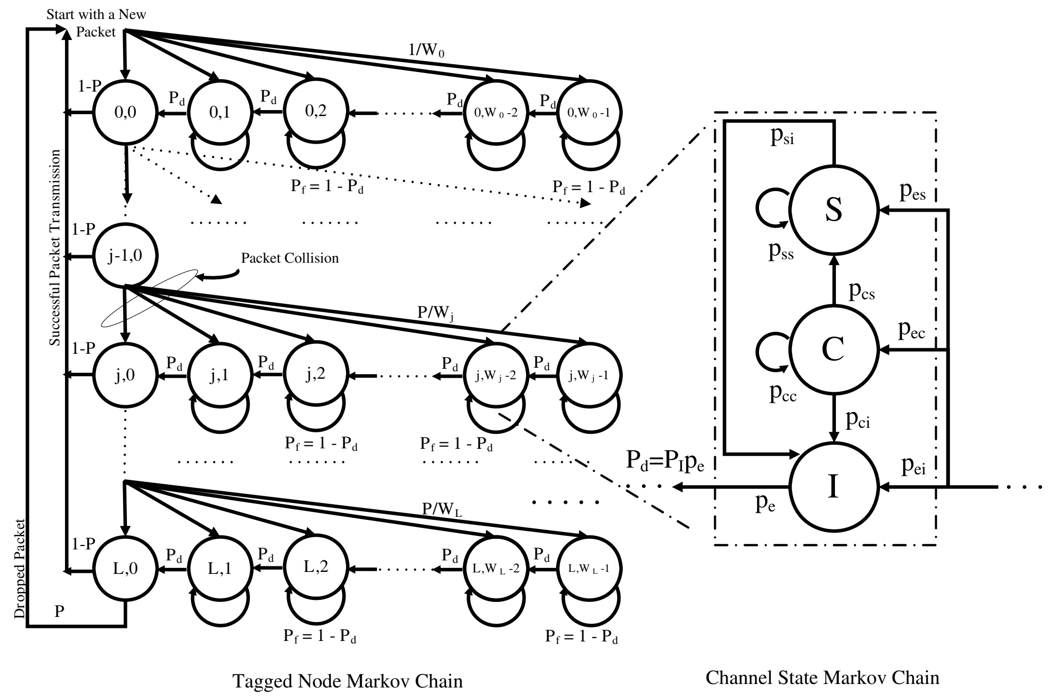


Fig. 1. System Markov chains.

with existing analytical models which confirm the improved accuracy for both models.

This paper is organized as follows: In Section II, previous approaches to 802.11 protocol analysis are reviewed. Then, the proposed saturation model is introduced in Section III. Calculation of performance metrics are introduced in Section IV. In Section V, the iterative algorithm used to model the unsaturated load is presented. Finally, Section VI concludes the paper.

II. Related Work

The performance of 802.11 DCF scheme has been studied extensively in the literature. Different approaches and simplifying assumptions were used for these analysis. The seminal paper of Bianchi [1] is the first paper to describe the binary exponential backoff mechanism of DCF protocol within a single node as a two dimensional Discrete Time Markov Chain. Bianchi's Markov chain, while providing a closed form solution, lacks two important features specified by IEEE 802.11b standard, which are retransmission limits and backoff counter freezing. Variations of Bianchi's Markov chain were proposed to describe the 802.11 protocol more accurately and adhere to the standard. Wu [6] and Chatzimisios [7] included the retransmission limit where a packet is dropped after reaching a maximum number of retransmissions. Zhang [8] and Xiao [9] improve the analysis by adding the backoff counter freezing probability to their models. Parallel to the Markovian modeling efforts, other analytical models predict the saturation performance of 802.11 without fully describing the detailed behavior of the binary exponential backoff [2] and [10].

In the literature, three main approaches were used to account for the non-saturation load assumption. Some Markovian models like Engelstad [11] and Malone [12] use additional states to model the behavior of a tagged node when it does not have any packets to send. Another approach used by Zhai [13] and Tickoo [14] scales the transmission probability of a saturation model by the probability of having at least one packet in the queue (i.e., the queue utilization). Another approach used to model the non-saturation performance estimates the number of active contending nodes (i.e. nodes that have a packet ready for transmission in their queue) implicitly in the model like in [15] and [16]. Both models assume a queue length of one packet. Garetto and Chiasserini [17] study the number of contending nodes using a multidimensional Markov chain which includes, in addition to the backoff process, the number of packets in the queue as well as the number of nodes in the network.

In all previous models, it is assumed that the probability of transmission is independent of the status of the previous time slot. Some proposed models like [18], [19] condition the events of the current time slot on the states of the previous time slots. Foh [18] adds more details to the Markov chain originally presented by Bianchi by accounting for the status of the previous time slot in the calculation of the transmission probability. In [19], a detailed analysis has been presented using a three state Markov chain that represents the states of the channel, however, this detailed analysis has not been used to estimate the transmission probability. Instead, it was used to compute the channel throughput based on the channel steady state probabilities.

In our proposed model, we carefully study the channel state during backoff using a separate Markov chain. Detailed modeling of the channel allows the model to be valid and accurate for a very wide range of protocol parameters. Based on this accurate saturation model, we propose an iterative algorithm to compute the non-saturation load performance.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

III. Saturation Load Analysis

Existing saturation models for DCF fail to accurately estimate the performance over a wide range of scenario parameters. Our investigations single out the inaccuracy in modeling the backoff process as the primary reason for this deviation. Specifically, the main problem lies in the estimation of accurate channel states while a node is decrementing its backoff counter.

In this section, we propose a saturation model for DCF considering a detailed channel state analysis based on the two dimensional Markov chain approach first proposed by Bianchi in [1]. First, the classical approach of Bianchi to model the saturation load of 802.11 DCF is presented in Section III-A for the sake of completeness. Then, our proposed detailed analysis of the channel state is presented in Section III-B.

The network model assumes N identical nodes in a fully-connected single-hop network. This setting implies that there are no hidden terminals. Each node has a backlogged queue with the same size packets. The channel is assumed to be error free and capture effect has not been used. As assumed in [1] and many others, we assume that the conditional collision probability is constant and independent of the backoff stage. This assumption was validated experimentally in [12] and shown to be a good approximation of the collision probability especially as the number of nodes increases.

A. Base Model

Since we assume that all nodes are identical, it is enough to analyze the behavior of one node to predict the behavior of the other $N-1$ nodes as well as the channel performance. We refer to this node as the ***Tagged Node***. Following Bianchi's approach in [1], the behavior of the binary exponential backoff process in a single node can be analyzed using two stochastic processes $s(t)$ and $b(t)$ that represent the backoff stage and the backoff counter values at time t, respectively. We refer to this two dimensional Markov chain as Tagged Node Markov Chain (TNMC) as shown in Figure 1. One major deviation from Bianchi's model is the inclusion of the freezing probability P_f which represents the probability that a node in backoff state freezes its backoff counter due to third party activity in the channel.

Proceeding with the traditional computation of the stationary probabilities of the Markov chain states like in [1], the probability τ that a station attempts to transmit a packet is the probability that the backoff counter reaches 0 in all backoff stages, which can be calculated as

$$\tau = \frac{1-P^{L+1}}{\left(\sum_{j=0}^{L}\left[1+\frac{1}{1-P_f}\sum_{k=1}^{W_j-1}\frac{W_j-k}{W_j}\right]P^j\right)(1-P)}, \quad (1)$$

where $L+1$ is the maximum number of retransmissions before dropping a packet or what is defined in IEEE 802.11 DCF standard [20] as ShortRetryLimit. The contention window size at backoff stage j, W_j, is defined as

$$W_j = \begin{cases} 2^j CW_{min} & \text{if } 0 \le j < m \\ 2^m CW_{min} & \text{if } m \le j \le L, \end{cases}$$

where CW_{min} is the minimum contention window size, CW_{max} is the maximum contention window size and $m = log_2\left(\frac{CW_{max}}{CW_{min}}\right)$.

Knowing τ, we can calculate the conditional collision probability P as

$$P = 1-(1-\tau)^{N-1}, \quad (2)$$

and the probability that a slot time is busy as $P_{\mathfrak{b}} = 1-(1-\tau)^N$. Finally, we can calculate the probability that the slot time contains the beginning of a successful packet transmission as $P_{\mathfrak{s}} = N\tau(1-\tau)^{N-1}$.

Note that the freezing probability P_f used to calculate τ in Equation (1) is not defined yet. In existing works, several assumptions have been made regarding the calculation of P_f. In [1] and [7], P_f was set to 0; and to P in [8] and [9]. We will numerically show that none of these assumptions yield an accurate estimation of the freezing probability P_f which affects the accuracy of predicting the protocol performance under different parameters settings.

We argue that a more detailed investigation of the channel states during backoff is extremely important in finding an accurate estimation of P_f. In the next section, our proposed detailed analysis of the channel state during back-off is presented to estimate the freezing probability P_f more accurately. With this estimation of P_f, Equations (1) and (2) can be solved numerically.

B. Calculation of Freezing Probability

A major shortcoming of existing models is not the Markov chain analysis but the estimation of an accurate transition probability associated with the decrementing/freezing of the backoff counters (i.e. P_d and P_f in Figure 1). In this section, we introduce our new proposed ***Channel State Markov Chain (CSMC)*** that accurately describes channel states while the tagged node is backing off, which is used to compute the transition probability $P_d = 1-P_f$.

Each backoff state in the TNMC shown is modeled as a three state CSMC as shown in Figure 1. States S, C, and I represent the state of the channel having a ***S***uccessful transmission, a ***C***ollision, or being ***I***dle while the tagged node is backing off. Hence, P_S, P_C, and P_I represent the steady state probability of the channel being in Successful, Collision or Idle state, respectively.

Referring to Figure 1, a backoff state can be entered from either a transmission state or from a previous backoff state. Note that the event of entering a backoff state is done at the edge of a slot time. At the same instance of time the tagged node enters a backoff state, the channel may become either busy due to a transmission, assuming that transmissions start at the beginning of a time slot, or stay idle for the entire duration of the time slot. Let us define p_{ei} as the probability that the tagged node enters a backoff state and finds the channel idle. Note that, from the tagged node's perspective, the channel is idle if none of the other $N-1$ nodes are transmitting. Then, p_{ei} can be calculated as $p_{ei} = (1-\tau)^{N-1}$.
Similarly, let us define p_{es} and p_{ec} as the probability that the tagged node enters a backoff state and finds the channel busy

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

with a successful transmission and a collision, respectively. We can calculate p_{es} and p_{ec}, respectively, as

$$\begin{aligned} p_{es} &= \binom{N-1}{1}\tau(1-\tau)^{N-2}, \\ p_{ec} &= 1 - p_{ei} - p_{es}. \end{aligned} \quad (3)$$

Once the tagged node enters the Idle state, either directly or coming from the states S or C in the same backoff state, it exits the backoff state with probability $p_e = 1$. Then, we can compute the probability of exiting the backoff state and thus decrementing the backoff counter as

$$P_d = p_e P_I, \quad (4)$$

and, hence, the freezing probability is computed as

$$P_f = 1 - P_d, \quad (5)$$

which can be used to calculate the transmission probability τ in Equation (1). To calculate the steady state probabilities P_I, P_S, and P_C, we need to compute the transition probabilities between the channel states. When the tagged node enters the backoff state and the channel has a successful transmission, the node that successfully transmitted a packet may send another successful packet if it selects the backoff counter of the next packet as zero since the other nodes have non-zero backoff counters. In that case, the channel stays in the S state with probability p_{ss}. On the other hand, if the successful node selects any other backoff counter value other than 0, then the channel state becomes idle with probability p_{si}. The transition probabilities p_{ss} and p_{si} are given as $p_{ss} = \frac{1}{W_0}$ and $p_{si} = 1 - p_{ss}$, respectively. Note that, the probability of transition from S to C is zero since after a successful transmission, only the node that transmits successfully has the full access to the channel in the next time slot if it selects the new backoff counter as zero.

The event of entering the backoff state when the channel is in the collision state C means that two or more nodes start transmitting during the current slot time. Since colliding packets cannot be decoded by third party nodes, those nodes cannot estimate the deferral Network Allocation Vector (NAV) timeout. Thus, IEEE 802.11 standard suggests that once a node receives an erroneous frame due to collisions, that node must defer channel access for a duration called Extended Inter Frame Space (EIFS). The EIFS interval begins when the physical layer indicates a medium IDLE condition at the end of the transmission of the erroneous frame. The value of EIFS is defined according to the standard [20] as $EIFS = SIFS + T_{CTS/ACK} + DIFS$ where $T_{CTS/ACK}$ is the time to send the CTS/ACK control frame depending on the type of the access mode. After the colliding transmissions end, this EIFS duration prevents other nodes from accessing the channel and allows the colliding nodes to directly access the channel in the following time slot if they select the new backoff counter value as zero. So, we can define the transition probability p_{ci} from the collision state C to the idle state I as the probability that no colliding node selects zero as the new backoff counter value after their transmission. Given that the number of colliding nodes is n, the probability that no colliding node selects zero as the new backoff counter is $(1 - \frac{1}{\overline{CW}})^n$. We can calculate the probability mass function of the number of colliding nodes as

$$Q(n) = \binom{N-1}{n}\tau^n(1-\tau)^{(N-n-1)}. \quad (6)$$

Then, the transition probability p_{ci} can be calculated as

$$p_{ci} = \sum_{n=2}^{N-1} Q(n)(1 - \frac{1}{\overline{CW}})^n, \quad (7)$$

where $\overline{CW}$ is the average backoff window size over all stages and can be calculated as

$$\overline{CW} = \sum_{i=0}^{L} \frac{(1-P)P^i W_i}{1 - P_{drop}}, \quad (8)$$

and P_{drop} is the probability that the packet is dropped after $L+1$ transmissions and can be computed as $P_{drop} = P^{L+1}$. Similarly, if only one node selects zero as the new backoff counter value after a collision occurs, the transition from the collision state to the successful transmission state happens with probability p_{cs}, which can be calculated as

$$p_{cs} = \sum_{n=2}^{N-1} Q(n)n(\frac{1}{\overline{CW}})(1 - \frac{1}{\overline{CW}})^{n-1} \quad (9)$$

Finally, the transition probability p_{cc} is calculated as $p_{cc} = 1 - p_{ci} - p_{cs}$.

Note that to keep our model tractable, we do not track the backoff stage of the other $N-1$ nodes in the network. Instead, we approximate their probabilities of selecting zero as a new backoff counter value by averaging backoff window over all backoff stages given as $\frac{1}{\overline{CW}}$.

Knowing all transition probabilities, transition matrix π of the CSMC can be written as

$$\pi = \begin{pmatrix} p_{ei} & p_{es} & p_{ec} \\ p_{si} & p_{ss} & 0 \\ p_{ci} & p_{cs} & p_{cc} \end{pmatrix} \quad (10)$$

To obtain the steady state probabilities for the channel state vector $A = [\ P_I\ P_S\ P_C\ ]$, we solve $\pi A = A$, using an iterative computation with an initial estimate of A.

C. Numerical Solution of the Saturation Model

To obtain the steady state values of the conditional collision probability P and the transmission probability τ, we need to solve the system of non-linear equations, including Equations (1) and (2) in the TNMC. Note that we need to compute P_f for the calculation of τ using Equation (1). We use an iterative process to evaluate the system behavior. First, an initial guess of $\tau^{(0)}$ is made and the conditional collision probability $P^{(0)}$ is calculated using Equation (2). Then, both $\tau^{(0)}$ and $P^{(0)}$ are fed into the CSMC to calculate the transition probabilities p_{ei}, p_{es}, p_{ec}, p_{si}, p_{ss}, p_{cs}, p_{cs}, and p_{ci}, which are used to calculate the steady state probabilities of the channel P_I, P_S and P_C. Knowing P_I, we can calculate the freezing probability $P_f^{(0)}$ using Equation (5). Then, $P_f^{(0)}$ and $P^{(0)}$ are in turn used in Equation (1) to calculate the updated transmission probability τ_{new}. Now, the updated transmission probability $\tau^{(1)}$ is computed using a relaxed fixed-point computation as $\tau^{(1)} = \alpha\tau^{(0)} + (1-\alpha)\tau_{new}$ where α is 0.5 in our calculations. The iterative process terminates once the difference between $\tau^{(i)}$ and $\tau^{(i-1)}$ is smaller that a predefined value ϵ.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

D. Convergence of Saturation Model Iterative Analysis

To prove the convergence of the iterative algorithm used in the saturation model, we need to show that this system has a unique solution. First, let us invert Equation (2) to represent τ as

$$\tau^*(P) = 1 - (1-P)^{\frac{1}{n-1}} \tag{11}$$

$\forall\ n > 1$. Note that $\tau^*(0) = 0$ and $\tau^*(1) = 1$ which means that τ^* is an increasing monotonic function. Furthermore, let us expand Equation (1) as shown in (12).

Solving for $P = 0$ results in

$$\tau(0, P_f) = \frac{2(1-P_f)}{(1-2P_f) + W_0}. \tag{13}$$

Note that we cannot directly solve $\tau(1, P_f)$ due to the terms $(1-P)$, $(1-P^{L+1})$ and $(1-P^{L-m+1})$ in the denominator of Equation (12). To solve this problem, we can further simplify Equation (12) as shown in (14).

Now $\tau(1, P_f)$ can be computed as

$$\begin{aligned}\tau(1,P_f) &= \frac{2(1-P_f)}{(1-2P_f) + W_0\frac{\sum_{j=0}^{m-1}(2)^j}{\sum_{j=0}^{L}1^j} + W_0(2)^m(\frac{\sum_{j=0}^{L-m+1}1^j}{\sum_{j=0}^{L}1^j})} \\ &= \frac{2(1-P_f)}{(1-2P_f) + W_0[\frac{\sum_{j=0}^{m-1}(2)^j + (2)^m\sum_{j=0}^{L-m+1}1^j}{\sum_{j=0}^{L}1^j}]}. \end{aligned} \tag{15}$$

Recall that from Equation (11), it is clear that τ^* is a monotonically increasing function. To prove solution uniqueness, it is enough to show that the other representation of τ (i.e. Equation (12)) is a monotonically decreasing function for all values of P_f. For that, we need to show that $\tau(0, P_f) > \tau(1, P_f)$. Since the inequality

$$\begin{aligned} W_0[\frac{\sum_{j=0}^{m-1}(2)^j + (2)^m\sum_{j=0}^{L-m+1}1^j}{\sum_{j=0}^{L}1^j}] &> W_0 \\ \frac{\sum_{j=0}^{m-1}(2)^j + (2)^m\sum_{j=0}^{L-m+1}1^j}{\sum_{j=0}^{L}1^j} &> 1 \end{aligned} \tag{16}$$

holds $\forall\ L,\ 1 < L < \infty$ and $\forall\ m,\ 1 < m < L$, it is easy to show that $\tau(0, P_f) > \tau(1, P_f)\ \forall\ P_f \in [0, 1]$ which in turn proves the convergence of the iterative algorithm of the saturation model.

IV. Computation of Performance Metrics

After calculating the steady state transmission probability τ, performance metrics like throughput and average channel access delay can be calculated. As defined in [1], the normalized channel throughput U can be calculated as

$$U = \frac{P_{\mathfrak{s}}T_p}{P_{\mathfrak{s}}T_s + (P_{\mathfrak{b}} - P_{\mathfrak{s}})T_c + (1-P_{\mathfrak{b}})T_i}, \tag{17}$$

where T_i is the duration of one idle time slot, T_s and T_c are the durations of the successful and collided transmissions respectively. T_p is the transmission duration of the payload without the additional headers of 802.11 and the physical layer. For the basic access mechanism of the DCF, T_s and T_c are calculated as

$$T_s = T_c = DIFS + T_h + T_p + SIFS + T_{ACK} \tag{18}$$

With RTS/CTS handshake, T_s and T_c are calculated, respectively, as

$$\begin{aligned} T_s &= DIFS + T_{RTS} + SIFS + T_{CTS} + SIFS + T_h + T_p \\ &\quad + SIFS + T_{ACK} \\ T_c &= DIFS + T_{RTS} + SIFS + T_{CTS} \end{aligned} \tag{19}$$

where T_h is the transmission duration of the MAC and physical layer headers and T_{RTS}, T_{CTS}, and T_{ACK} are the transmission durations of the RTS, CTS, and ACK frames. SIFS and DIFS are the guard times defined by the IEEE 802.11 standard. We assume that the propagation delay is negligible.

In addition to the throughput, we are interested in the calculation of the average channel access delay for packets that are transmitted successfully. The channel access delay is defined as the time from the packet becoming the head of the queue until the acknowledgment frame is received. Consider a successful transmission at the first attempt. The average channel access delay can be estimated as $T^{(0)} = T_s + \overline{W_0}\mathcal{F}$ where $\overline{W_j}$ is the average number of backoff slots selected at stage j and calculated as $\overline{W_j} = \frac{W_j - 1}{2}$, and $\mathcal{F}$ is the average slot duration the tagged node should wait before decrementing its backoff counter. Computation of $\mathcal{F}$ will be presented later in section IV-A. For any subsequent success at the $i + 1^{st}$ attempt preceded by i collisions, the channel access time $T^{(i)}$ is computed as follows:

$$T^{(i)} = T_s + iT_c + \sum_{j=0}^{i}\overline{W_j}\mathcal{F}. \tag{20}$$

The probability $P^{(i)}_{success}$ that a packet is successfully transmitted in the $i+1^{st}$ transmission attempt is given by $P^{(i)}_{success} = (1-P)P^i$. Conditioning the summations by the probability of successful transmission, the overall average channel access delay T for non-dropped packet is computed as

$$T = \frac{1}{1-P_{drop}}\sum_{i=0}^{L}P^{(i)}_{success}\left[T_s + iT_c + \left(\sum_{j=0}^{i}\overline{W_j}\mathcal{F}\right)\right]. \tag{21}$$

A. Average Slot Duration Calculation

To complete the computation of the average channel access delay presented in Equation (21), we need to calculate the average slot duration $\mathcal{F}$ that a node spends in each backoff state before decrementing its backoff counter. Note that the time spent at each backoff state depends on the channel state at the new time slot as well as the previous time slot. Let us define D_I, D_S, and D_C as the durations that the tagged node spends at a particular backoff state when the state of the channel at the entrance time is idle, having a successful transmission, and having a collision, respectively. When the tagged node enters a backoff state and the channel is idle at that time slot, the node waits for 1 time slot before decrementing the backoff counter and exiting from the current backoff state.

On the other hand, when the node enters the backoff state and finds the channel busy with a successful transmission, it should wait for the duration of the successful transmission

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

$$\tau(P,P_f) = \frac{2(1-P_f)(1-2P)(1-P^{(L+1)})}{(1-2P_f)(1-2P)(1-P^{(L+1)}) + W_0(1-P)(1-(2P)^m) + W_0(2P)^m(1-P^{(L-m+1)})(1-2P)} \quad (12)$$

$$\begin{aligned}\tau(P,P_f) &= \frac{2(1-P_f)}{(1-2P_f) + W_0\frac{(1-P)}{1-P^{(L+1)}}\frac{(1-(2P)^m)}{(1-2P)} + W_0(2P)^m(\frac{1-P^{(L-m+1)}}{1-P^{(L+1)}})} \\ &= \frac{2(1-P_f)}{(1-2P_f) + W_0\frac{\sum_{j=0}^{m-1}(2P)^j}{\sum_{j=0}^{L}P^j} + W_0(2P)^m(\frac{\sum_{j=0}^{L-m+1}P^j}{\sum_{j=0}^{L}P^j})}\end{aligned} \quad (14)$$

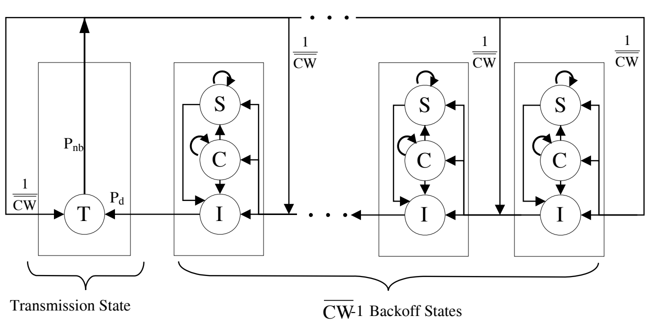


Fig. 2. Simplified Markov chain for delay calculation.

and any other successive successful transmissions plus one additional idle time slot before exiting the backoff state. Note that a node that transmits a packet successfully can access the channel and send another successful packet with probability $p_{ss} = 1/W_0$. The average number of successive successful transmissions can be estimated as $\frac{1}{1-p_{ss}}$. Thus, we have $D_I = 1$ and $D_S = \frac{1}{1-p_{ss}}T_s + D_I$.

Similarly, when the node enters a backoff state and finds the channel busy with a collision, the node waits for the duration of the collision and any other successive collisions or successful transmissions, plus one additional idle time slot before exiting the backoff state. Note that, on the average, there can be $\sum_{i=0}^{L} ip_{cc}^{i}$ successive collisions. Then, after the successive collisions, there is a successful transmission with probability $\frac{p_{cs}}{1-p_{cc}}$ or the node will go to idle state with probability $\frac{p_{ci}}{1-p_{cc}}$. Thus, D_C can be calculate as

$$D_C = \left(\sum_{i=0}^{L} ip_{cc}^{i}\right)T_c + \frac{p_{cs}}{1-p_{cc}}D_S + \frac{p_{ci}}{1-p_{cc}}D_I. \quad (22)$$

We need to find the probabilities for each of these events. To simplify the analysis, we use a reduced form of TNMC as shown in Figure 2. Consider the simplified Markov chain. The tagged node can enter any backoff state from either a previous backoff state or from the transmission state (except of the last backoff state which is accessible only from the transmission state). Let us consider the two cases in detail to calculate the average slot duration for each case:

a) Entering from a previous backoff state: The node enters a backoff state from a previous backoff state if and only if the previous time slot was idle. Recall that in Section III-B, we calculate the probability P_d that the backoff counter decrements its counter in Equation (4). Then, conditioning on decrementing the backoff counter from a previous backoff state, the average slot duration in this case can be calculated as

$$F_b = \frac{p_{ei}}{P_d}D_I + \frac{p_{es}}{P_d}D_S + \frac{p_{ec}}{P_d}D_C. \quad (23)$$

b) Entering from a transmission state: After finishing a transmission, either successful or collision, the node always selects a new backoff counter with probability $P_{nb} = 1$. However, with probability $1/\overline{CW}$ the node will remain in a transmission state and with probability $1 - 1/\overline{CW}$ the node will go to one of the backoff states. Then, the average slot duration case is given as

$$F_t = \left(1 - \frac{1}{\overline{CW}}\right)\left[\frac{p_{ei}}{p_{nb}}D_I + \frac{p_{es}}{p_{nb}}D_S + \frac{p_{ec}}{p_{nb}}D_C\right], \quad (24)$$

where $\overline{CW}$ is defined in Equation (8). Since a state is a backoff state with probability $1-\tau$, the average slot duration is calculated as

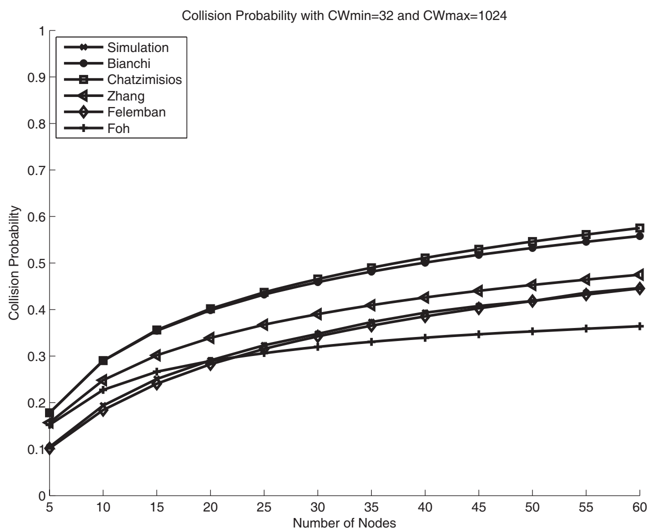
$$\mathcal{F} = (1-\tau)F_b + \tau F_t, \quad (25)$$

which can be used to calculate the average channel access delay in Equation (21).

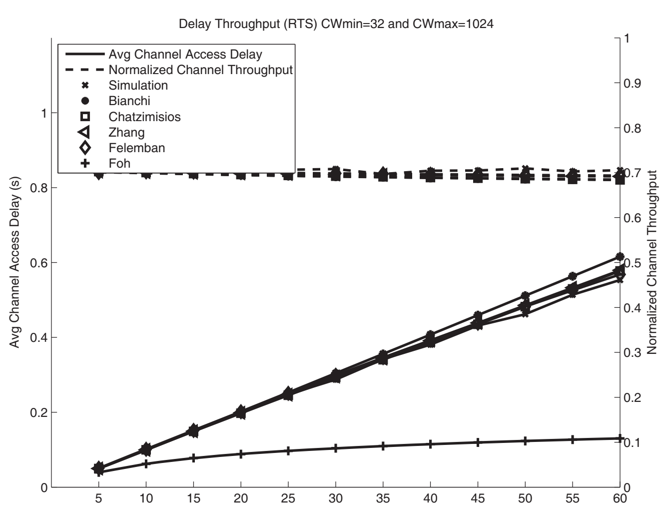
B. Numerical Validation

To validate the accuracy of the proposed saturation model, we compare the analytical results obtained from the analytical model against simulation results following the implementation of IEEE 802.11b standard in Qualnet [21] simulator. We have used the standard values for SIFS, DIFS and EIFS durations, as well as the durations of the RTS, CTS and ACK control frames. The payload from the application layer is 1024 bytes with 1 Mbps channel bit rate. We vary the number of contending nodes from 5 to 60 in increments of 5. We obtain the collision probability, the normalized throughout and the average channel access delay from simulations and using Equations (2), (17), and (21), respectively.

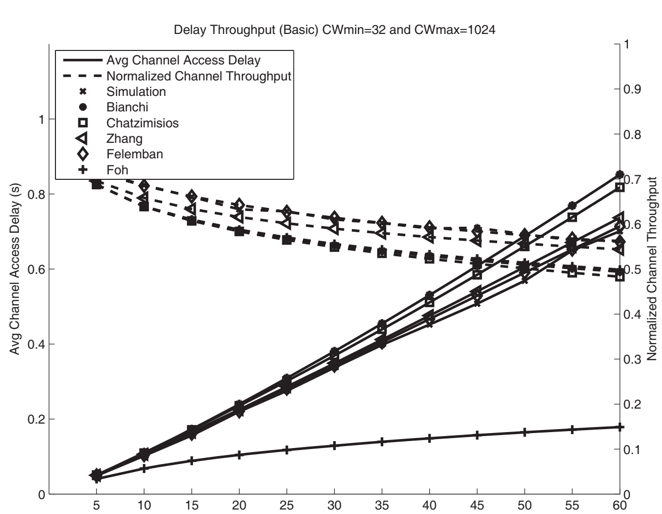
We have compared the performance of our proposed model against four other saturation models named after the last name of the first author. These models are Bianchi, Chatzimizios, Foh and Zhang. We have selected Bianchi's model [1] as it is the classical model and the first analytical model that analyzes the backoff operation of a single node using a two dimensional Markov Chain. Chatzimizios [7] uses a modified version of the Bianchi model that considers the retransmission limit and packet dropping. Foh model [18] accounts for the status of the previous time slot in calculating the transmission probability. Finally, the Zhang model [8] adds the probability of backoff



(a) Collision Probability



(b) Throughput and Channel Access Delay with RTS/CTS Mode



(c) Throughput and Channel Access Delay with Basic Mode

Fig. 3. Performance comparison with CW_{min}=32, CW_{max}=1024.

counter freezing during backoff. Note that the average channel access delay calculation is not included in the Bianchi model. We adopt the delay calculation presented in [22] for the Bianchi model. Note also that the model proposed by Foh in [18] calculates two collision probabilities based on the channel status of the previous time slot. For completeness, we have used $p_{foh} = (P_i)p_0 + (1 - P_i)p_1$ to combine the resulted collision probabilities into one collision probability metric where, according to [18], P_i is the probability that a particular period on the channel is idle, p_0 (p_1) is the probability that, from a station's point of view, at least one of the other stations transmit during a slot after an idle (a busy) period. Since the model presented in [18] did not include any delay calculation for the average service time, we calculate the delay for Foh model using the delay calculation presented in [22].

We ran extensive simulations using Qualnet simulator over a wide range of contention window sizes. Due to space limitation, we show in detail the numerical validation results for two contention window settings in Figures 3 and 4, which show the collision probability, throughput, and the average channel access delay for both RTS/CTS and Basic mode as the number of nodes increases.

We note that the contention window size affects the collision probability as shown in Figures 3(a) and 4(a). As the contention window sizes decrease, the collision probability increases because every time a collision occurs, the increase of the backoff range is limited. Second, although the Chatzimisios model adds the retransmission limit to the Bianchi model, their collision probability coincides with the Bianchi model for virtually all parameter settings. This suggests that the effect of ignoring the retransmission limit of collision probability computation is overshadowed by other effects.

Both Bianchi and Chatzimisios models overestimate the collision probability since neither model accounts for the probability that the backoff counter is frozen when there is activity in the channel. Referring to Equation (1), if P_f is set to 0, the transmission probability, and consequently the collision probability will increase. We also note that the Zhang model, which modifies the Bianchi model by adding the backoff counter freezing probability, does not accurately predict the collision probability, which suggests that the collision probability cannot be used as the backoff freezing probability. Our proposed model accurately predicts the collision probability for the contention window sizes, and in fact for a wide range of parameter settings, which is a result of the detailed analysis of the channel states during backoff. It is noteworthy that Foh model differentiates the collision probability based on the channel states just before transmission. Note also that the collision probability does not depend on the access scheme where we can see the same collision probability value for both RTS/CTS and basic access mode for the two different contention sizes.

Note that Foh model differentiates the collision probability according to the channel status in the previous time slot, i.e. busy or idle. The model then uses those collision probabilities as backoff freezing probabilities. Although this approach may accurately capture the behavior of the protocol's collision probability, it fails to capture the freezing probability in addition to the overall performance of 802.11 protocol as shown in

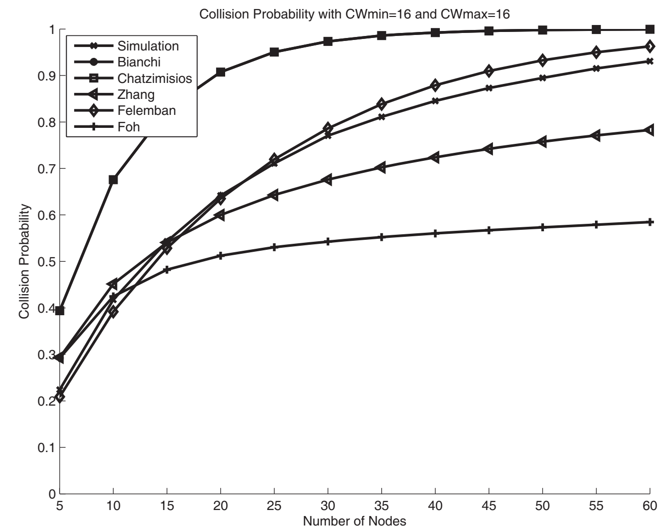
Figures 3(a) and 4(a) and later on the throughput and channel access delay results. Although our model does not differentiate the collision probability according to the channel status of the previous time slot, it accurately predicts the collision probability, which suggests that the main contribution to the accuracy is the correct modeling of the freezing probability.

Due to the space limitation, we have combined the normalized channel throughput and the average channel access delay charts in one graph for both access modes and contention window sizes as shown in Figures 3(b), 3(c), 4(b) and 4(c). We use the dashed lines to refer to the average channel access delay and the solid lines to refer to the normalized channel throughput for results of different models. We use the markers $\times$, $*$, $\square$, $\triangleleft$, $\diamond$ and $+$ to represent the data obtained from the simulation, Bianchi model, Chatzimisios model, Zhang model, our proposed model and Foh model, respectively.

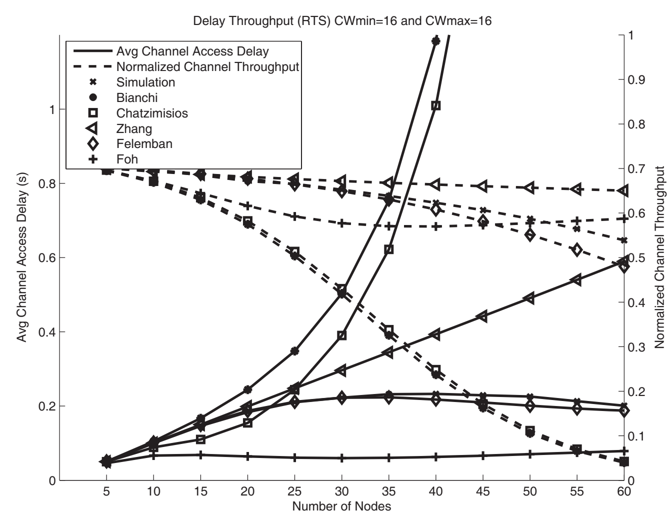
Figures 3(b) and 3(c) show the normalized channel throughput and the average channel access delay when the RTS/CTS mode and the basic mode are used with the standard contention window sizes (i.e. $CW_{min} = 32$ and $CW_{max} = 1024$). We notice that when using the basic access mode, the channel throughput decreases and the average channel access delay increases compared to RTS/CTS access mode. The predicted performance metrics from all models has minimal deviations from the simulation data when RTS/CTS access model is used.

With basic access mode, the accuracy of the proposed model become more noticeable. This is because with the basic access mode, the cost of packet collisions increases which affects the freezing duration for backing off nodes. The proposed model with the detailed channel states captures this increase in the freezing duration resulting in more accurate estimation of both the throughput and the average access delay.

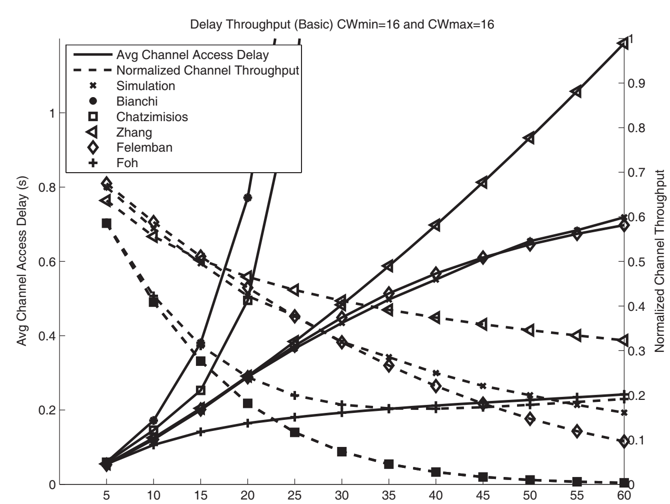
As the contention window sizes decrease, the accuracy of estimating the collision probability become more noticeable in estimating the channel throughput and the average channel access delay as shown in Figures 4(b) and 4(c). Both Bianchi and Chatzimisios models overestimate (underestimate) the channel access delay (channel throughput) due to the large collision probability estimations as shown in Figure 4(a). Also, Zhang model overestimates the channel access delay which is due to the overestimation of the average freezing time presented in their model.



(a) Collision Probability



(b) Throughput and Channel Access Delay with RTS/CTS Mode



(c) Throughput and Channel Access Delay with Basic Mode

Fig. 4. Performance comparison with CW_{min}=16, CW_{max}=16.

V. Extension to Non-Saturation Analysis

So far, our analysis targeted the saturation load where every node in the network always has a packet ready for transmission. However, from an application's point of view, saturated load is unlikely to be valid in real life applications due to queue overflow and infinite queuing time. To model DCF protocol under unsaturated load, existing non-saturation models modify the transmission probability τ of a single node by either adding more states to the TNMC or by scaling τ with the probability that the queue is empty.

We will numerically show that these approaches cannot accurately predict the performance of the network under unsaturated load. This is due to the assumption that individual nodes are independent, which is correct in the case of saturation

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

traffic but does not hold with unsaturated load. With saturation load, all nodes have packets ready for transmission, thus the number of contending nodes, which have non-empty queues, in the network is constant and known to all nodes. This advantage promotes saturation load case and makes it more appealing and easier for analysis compared to unsaturated load cases where the number of contending nodes is continuously changing based on the packet arrival rates.

Garetto and Chiasserini [17] studied this effect in detail. They found that the predicted distribution of the number of contending nodes using the simple traditional two dimensional Markov chain has a large deviation from the actual distribution obtained from the simulation. To solve this problem, Garetto proposed a Markov chain model that accounts for the number of contending nodes in the network and results in a highly accurate estimation of the performance metrics. However, since the number of the states in proposed analytical models strongly depends on the number of nodes and the buffer size, the scalability and the complexity of the model limit the practical usage of such an approach.

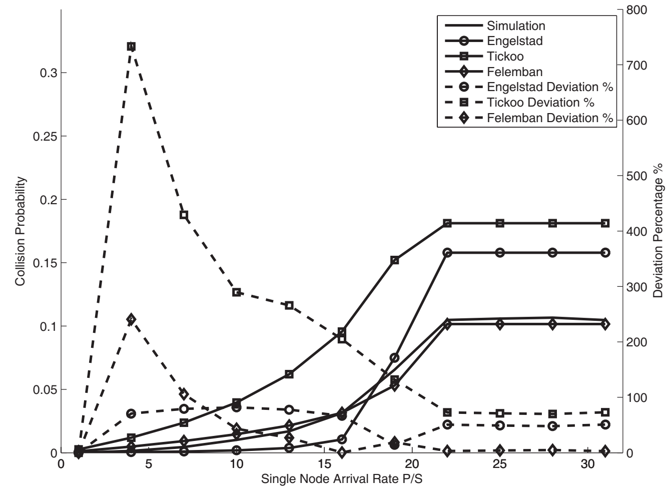
In this paper our aim is to develop a lower complexity model for non-saturation traffic that can possibly be utilized by run-time procedures to provide reasonably accurate predictions. Therefore, we have pursued an approximation model that models the dynamic nature of the non-saturation traffic as an interacting set of steady state saturated systems with different number of users with the help of our saturation model presented in Section III. We note that the accuracy of our approach increases as the empty/non-empty transition rates of node queues decrease, i.e., as the network allows for more time for the system to reach a steady state solution for a given number of active users. We show through extensive simulations that the proposed model accurately estimates the collision probability, throughput, and average channel access delay for a wide range of scenario parameters and traffic loads.

A. Non-Saturation Model

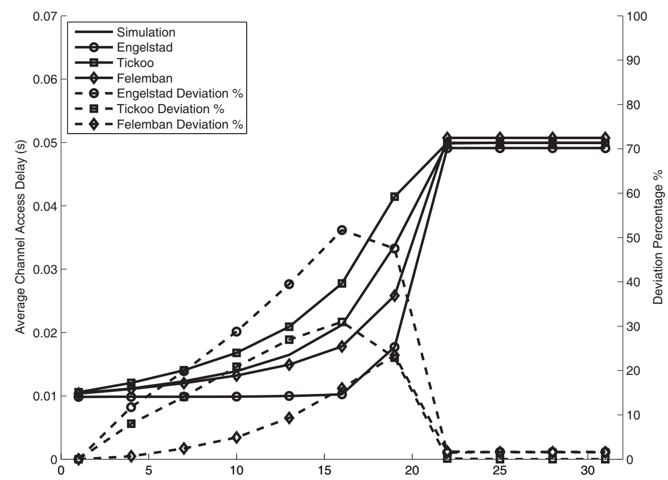
Recall that the saturation analysis means that all nodes always have packets to send, therefore, the number of nodes contending for the channel is always N. Since a node with an empty queue does not contend for the channel, the average number of contending nodes in the unsaturated load varies between 1 and N depending on the offered load to each node. To model the non-saturation load cases, we propose an iterative algorithm to compute the probability P_0 that a node does not have any packets to send using our proposed saturation model. For Poisson packet arrivals at rate λ, and approximating the service time distribution as exponential with mean $E[T]$ as presented in [23], P_0 is computed as

$$P_0 = 1 - \frac{\lambda}{\overline{\mu}}, \tag{26}$$

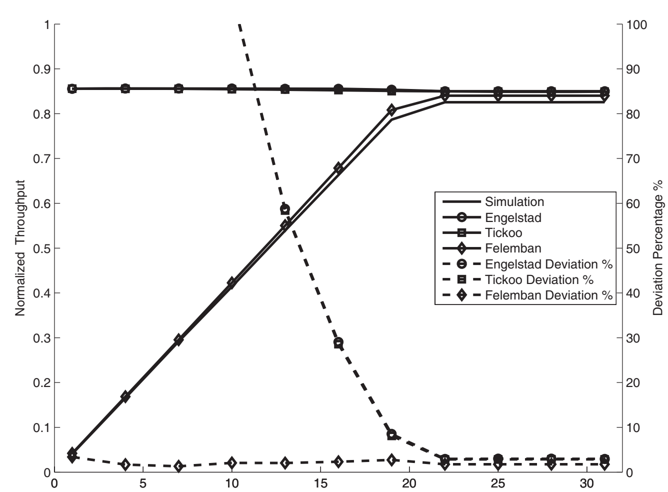
where λ is the packet arrival rate (P/slottime) and $\overline{\mu}$ is the average service rate given in Equation (29). Recall that, from Equations (21) and (2), the channel access delay for a successful packet is a function of the number of active nodes. Since the number of active nodes in non-saturation loads varies, we can estimate the probability mass function of



(a) Collision Probability



(b) Average Channel Access Delay



(c) Throughput

Fig. 5. Performance comparison and estimation deviation for non-saturation load with 5 nodes.

the number of active contending nodes i (i.e. nodes with at least one packet in their queues) using P_0 as

$$\mathfrak{P}_i = \binom{N}{i}(1-P_0)^i P_0^{(N-i)}. \quad (27)$$

The average channel access delay T_i given that i nodes are contending for the channel can be obtained from the average channel access delay of our proposed saturation model. Then, the expected channel access delay given that there is at least one active node can be calculated as

$$E[T] = \frac{\sum_{i=1}^{N} T_i \mathfrak{P}_i}{1-\mathfrak{P}_0}, \quad (28)$$

and the average service rate $\overline{\mu}$, in (P/slottime), is given as

$$\overline{\mu} = \frac{1}{E[T]}, \quad (29)$$

which we can use in Equation (26) to obtain a new estimation of P_0. We repeat this procedure until the new value of P_0 does not change from the last value, which shows that the system converges.

To obtain the non-saturation load performance metrics (i.e. the collision probability, channel throughput, and the channel access delay), we compute the PMF of the number of active contending nodes with the probability P_0 using Equation (27). Then, the collision probability, channel throughput and the channel access delay for the non-saturation traffic are computed, respectively as

$$\begin{aligned} P_{nonSat} &= \frac{\sum_{i=1}^{N} P_i \mathfrak{P}_i}{1-\mathfrak{P}_0} \\ U_{nonSat} &= \sum_{i=1}^{N} U_i \mathfrak{P}_i \\ T_{nonSat} &= \frac{\sum_{i=1}^{N} T_i \mathfrak{P}_i}{1-\mathfrak{P}_0} \end{aligned} \quad (30)$$

where P_i, U_i, and T_i are the collision probability, channel throughput, and channel access delay given that i nodes are contending for the channel which can be obtained from our proposed saturation model.

B. Numerical Validation

We validated the proposed non-saturation model against the simulation results using the same simulation settings used in the saturation case. In our simulations, we assume Poisson packet arrivals for all nodes. We vary the packet arrival rate to each node and run the simulations for 5, 10, 20 and 40 nodes using four different contention window sizes. Similar to the saturation load numerical validation, we obtain the conditional collision probability, the normalized throughput, and the average channel access delay. We compare our proposed model against two non-saturation models, i.e., Tickoo [14] and Engelstad [11]. Note that the Engelstad model is a model for IEEE 802.11e that we reconfigure by setting the number of classes to 1. Due to space limitation, we show the simulation results for the default contention window sizes (i.e. CW_{min} = 32 and CW_{max} = 1024) for 5 nodes in Figure 5 along with the deviation percentage of each model from the simulation results.

Figure 5(a) shows the collision probability estimated from the models and obtained from the simulation as the packet arrival rate increases. To show the accuracy of the models, the deviation percentage of each model is shown on the same graph as well. We use solid lines to represents the prediction values and dashed lines to represent the deviation from the simulation results. We observe that Tickoo model has the largest deviation reaching 800% at lower arrival rate mainly because it uses the same transmission probability as in [1] which does not account for retransmission limits or the freezing probability. With lower arrival rate, we note also that Engelstad model has lower deviation compared to our proposed model, however as the arrival rate increases beyond 7 P/S, the estimation error of the proposed model decreases to less than 5%. This shows that the proposed model can accurately estimates the collision probability for a wide range of arrival rates.

Figure 5(b) shows the average channel access delay comparison for 5 nodes using 32 and 1024 as the minimum and maximum contention window. Similar to Figure 5(a), we plot both data (Solid Lines) and deviation percentage (Dashed Lines). Engelstad model underestimates the delay by a noticeable factor which is due to the assumption of the independence between non-saturated nodes discussed earlier in this section. Tickoo model on the other hand, overestimates the delay, especially for low traffic loads, which is due to inaccurate estimation of freezing time in their delay calculations. Over all arrival rates, it is clear that the proposed model consistently outperforms other models in terms of accuracy mainly due to the detailed analysis of the backoff freezing states.

Figure 5(c) shows the non-saturation normalized throughput comparison. Explicit equations to calculate non-saturation channel throughput were not provided for either Engelstad or Tickoo models as they did not show any throughput performance for non-saturation case. As a result, we used the saturation throughput Equation(17) to draw the throughput for Engelstad and Tickoo models. In our proposed non-saturation model, however, by estimating the number of active nodes contending for the channel in the network, we obtain a deviation below 2% for throughput for all tested arrival rates as shown in Fig. 5(c).

Due to space limitation, detailed estimation comparisons are omitted for different scenario configurations. Instead, Fig. 6 shows the deviation percentage of the average channel access delay for all models for different scenario configurations. The x axes of the figure represents the configuration of the scenario as tuples (x, y, z) where x is the number of nodes, y represents the minimum contention window and z is the maximum contention window size. Engelstad model underestimates the delay by a noticeable factor which is due to the assumption of the independence between non-saturated nodes discussed earlier in this section. Tickoo model on the other hand, overestimates the delay, especially for low traffic loads, which is due to inaccurate estimation of freezing time in their delay calculations. Our proposed non-saturation model shows consistent accuracy of estimating the average channel access delay over a wide range of scenario parameters.

VI. Conclusion

In this paper, we have presented an analytical model for IEEE 802.11 DCF protocol that can accurately predict the

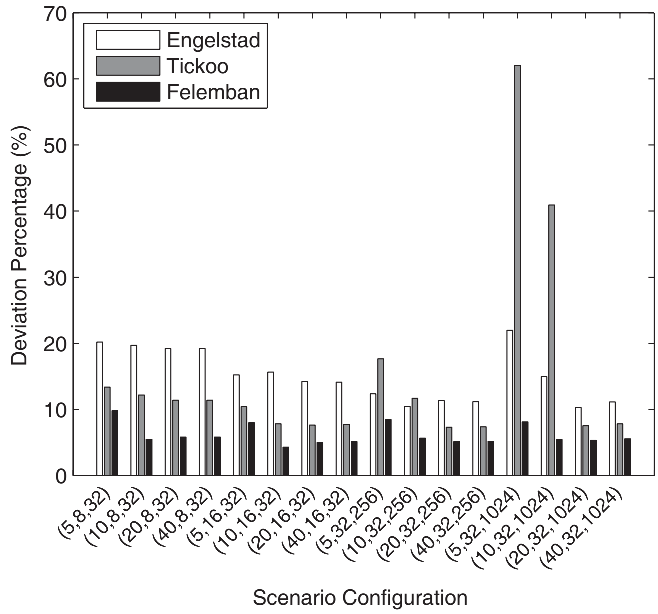


Fig. 6. Channel access delay deviation percentage for non-saturation scenarios.

performance for a wide range of scenario parameters in single-hop networks. We have started with accurately modeling the performance of DCF under the saturation assumption using a detailed analysis of the channel states during backoff countdown. Then, we extend the proposed saturation model to non-saturation cases. To validate the accuracy of our proposed models, we extensively tested the estimation performance of the models for a wide range of contention window sizes, number of nodes and traffic loads. As a future work we will extend this work to model IEEE 802.11e protocol which provides QoS differentiation. Also, it is of interest to study the queuing delay and heterogenous packet arrival rates in the case of non-saturation loads.

REFERENCES

[1] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, pp. 535–547, Mar. 2000.

[2] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve theoretical throughput limit," *IEEE/ACM Trans. Networking*, vol. 8, pp. 785–799, Dec. 2000.

[3] Y. Chetoui and N. Bouabdallah, "Adjustment mechanism for the IEEE 802.11 contention window: an efficient bandwidth sharing scheme," *Computer Commun.*, vol. 30, pp. 2686–2695, 2007.

[4] M. Ergen and P. Varaiya, "Throughput analysis and admission control for IEEE 802.11a," *Mobile Networks and Applications*, vol. 10, pp. 705–716, 2005.

[5] L. Lin, H. Fu, and W. Jia, "An efficient admission control for IEEE networks based on throughput analysis of (un)saturated channel," in *Proc. IEEE GLOBECOM*, vol. 5, pp. 3017–3021, Dec. 2005.

[6] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of reliable transports protocol over IEEE 802.11 wireless LAN: analysis and enhancement," in *Proc. IEEE INFOCOM*, vol. 2, pp. 599–607, June 2002.

[7] P. Chatzimisios, A. Boucouvalas, and V. Vitsas, "IEEE 802.11 packet delay—a finite retry limit analysis," in *Proc. IEEE GLOBECOM*, vol. 2, pp. 950–954, Dec. 2003.

[8] L. Zhang, Y. Shu, O. Yang, and G. Wang, "Study of medium access delay in IEEE 802.11 wireless networks," *IEICE Trans. Commun.*, vol. 89, pp. 1284–1293, Apr. 2006.

[9] Y. Xiao, "Performance analysis of priority schemes for IEEE 802.11 and IEEE 802.11e wireless LANs," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 1506–1515, July 2005.

[10] Y. Tay and K. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," *Wireless Networks*, vol. 7, pp. 159–171, 2001.

[11] P. Engelstad and O. Osterbo, "Analysis of the total delay of IEEE 802.11e EDCA and 802.11 DCF," in *Proc. IEEE International Conference on Communications*, vol. 2, pp. 552–559, June 2006.

[12] D. Malone, I. Dangerfield, and D. Leith, "Verification of common 802.11 MAC model assumptions," in *Passive and Active Measurement Conference PAM*, 2007.

[13] H. Zhai, Y. Kwon, and Y. Fang, "Performance analysis of IEEE 802.11 MAC protocols in wirless LANs," *Wireless Commun. and Mobile Comput.*, vol. 4, no. 0, pp. 917–931, 2004.

[14] O. Tickoo and B. Sikdar, "Modeling queueing and channel access delay in unsaturated IEEE 802.11 random access MAC based wireless networks," *IEEE/ACM Trans. Networking*, vol. 16, pp. 878–891, Aug. 2008.

[15] C. Foh, M. Zukerman, and J. Tantra, "A Markovian framework for performance evaluation of IEEE 802.11," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 1276–1285, Apr. 2007.

[16] E. Winands, T. Denteneer, J. Resing, and R. Rietman, "A finite-source feedback queueing network as a model for the IEEE 802.11 DCF," *Europ. Trans. Telecommun.*, vol. 16, no. 1, pp. 77–89, 2005.

[17] M. Garetto and C.-F. Chiasserini, "Performance analysis of the 802.11 distributed coorednation function under sporadic traffic," *Networking*, May 2005.

[18] C. Foh and J. Tantra, "Comments on IEEE 802.11 saturation throughput analysis with freezing of backoff counter," *IEEE Commun. Lett.*, vol. 9, pp. 130–132, Feb. 2005.

[19] C. Hu, H. Kim, and J. C. Hou, "Short-term nonuniform access in IEEE 802.11-compliant WLANs: a microscopic view and its impact," in *ACM MSWIM*, Oct. 2006.

[20] "Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specification," IEEE Std 802.11-2007.

[21] Qualnet, "http://www.scalable-networks.com."

[22] P. Chatzimisios, A. Boucouvalas, and V. Vitsas, "Packet delay analysis of IEEE 802.11 MAC protocol," *Electron. Lett.*, vol. 39, pp. 1358–1359, Sep. 2003.

[23] K. Medepalli and F. Tobagi, "System centric and user centric queueing models for IEEE 802.11 based wireless LANs," in *Proc. 2nd International Conference on Broadband Networks*, vol. 1, pp. 612–621, Oct. 2005.

Emad Felemban is an Assistant Professor at the Department of Computer Engineering, Umm al-Qura University (UQU), Makkah, Saudi Arabia. He received his master and PhD degrees from The Ohio State University in 2003 and 2009, respectively. His current research interests are in wireless sensor networks, performance analysis of wireless networks and QoS provisioning. He served as a TPC members for PIMRC 2010 and Globecom 2010.

Eylem Ekici (S'99-M'02-SM'11) received the BS and MS degrees in Computer Engineering from Bogazici University, Istanbul, Turkey, in 1997 and 1998, respectively, and the PhD degree in Electrical and Computer Engineering from the Georgia Institute of Technology, Atlanta, GA, in 2002. Currently, he is an Associate Professor with the Department of Electrical and Computer Engineering, The Ohio State University. His current research interests include wireless sensor networks, vehicular communication systems, cognitive radio networks, resource management, and analysis of network architectures and protocols. He is an Associate Editor of *Computer Networks Journal* (Elsevier) and *ACM Mobile Computing and Communications Review*, and IEEE/ACM TRANSACTIONS ON NETWORKING.