

Development of Collision Alleviating DCF Protocol with Efficient Backoff Algorithm for Wireless Ad hoc Networks

Tatineni Madhavi · Gottapu Sasi Bhushana Rao

Published online: 1 November 2014
© Springer Science+Business Media New York 2014

Abstract Ad hoc networks have been proposed for emergency communication wherein the required infrastructure is unavailable. However, a major concern in Ad hoc networks is collisions. Even in infrastructure based wireless networks, when the number of contending nodes is high, more number of frame collisions occur which leads to drastic reduction in network performance. In all IEEE 802.11 based wireless and Ad hoc networks, the backoff algorithm dynamically controls the contention window of the nodes experiencing collisions. Even though several algorithms such as Binary Exponential Backoff, Double Increment Double Decrement backoff, Exponential Increase Exponential Decrease backoff, Hybrid Backoff, Binary Negative Exponential Backoff etc. have been proposed in the literature to enhance the performance of IEEE 802.11 Distributed Coordination Function (DCF) protocol, most of them have not been developed for real- traffic scenarios. Also the packet collision rate is high using these algorithms. So, in this paper, a new Contention Window based Multiplicative Increase Decrease Backoff (CWMIDB) algorithm is proposed for the DCF protocol to alleviate the number of collisions. Furthermore, the packet transmission procedure of the DCF protocol is modified to avoid channel capture effect and this is represented with a Markov chain model. A simple mathematical model is developed for transmission probability considering the non-saturated traffic and channel errors. Results show that the proposed CWMIDB algorithm provides superior quality-of-service parameters over existing backoff algorithms.

Keywords IEEE 802.11 · DCF · Backoff algorithm · Markov chain · Throughput

T. Madhavi (✉)
Department of ECE, GIT, GITAM University, Visakhapatnam, India
e-mail: m_tatineni@yahoo.com

G. Sasi Bhushana Rao
Department of ECE, AU Engineering College, Andhra University, Visakhapatnam, India

1 Introduction

The DCF is the generic protocol used in all IEEE 802.11 standards that employ Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) method using binary exponential backoff algorithm. Readers can refer [1] for data transmission procedures under basic and RTS/CTS mechanisms used in DCF protocol. For supporting the real time applications, throughput and minimum End-to-End delay are the two main challenging issues in the design of IEEE 802.11 WLAN protocols. To evaluate and enhance the performance of IEEE 802.11 DCF protocol, several packet transmission procedures have been suggested in the literature and these are represented with Markov chain models. However, these models cannot accurately predict the performance of the network and they suffer with high packet collisions resulting in degradation of throughput and End-to-End delay, particularly under congested environments. Many researchers concentrated on the performance evaluation of DCF assuming ideal channel conditions. Bianchi was the first to derive a model that incorporates the exponential backoff process inherent in 802.11 as a two dimensional Markov chain to analyze the saturated throughput of 802.11 [2,3]. To increase the accuracy of the results, busy medium conditions are considered in the model [4]. Here, the throughput is analyzed under saturated traffic conditions. Daneshgaran et al. [5] presented a Markov model to analyze the throughput considering transmission errors and capture effects over Rayleigh fading channels. Hung et al. [6] considered the effect of hidden nodes to evaluate the performance of IEEE 802.11 DCF. Malone et al. [7] extended the Bianchi's model to evaluate the throughput in non-saturated conditions considering different traffic arrival rates. However, these models cannot evaluate the performance of DCF accurately. In our previous research, we have proposed an exact Markov chain model to accurately predict the performance of the DCF protocol [8]. To alleviate the collisions and avoid channel capture effect, a post backoff stage is introduced to provide inter packet backoff (IPB) delay between successive packet transmissions. The modified model is named as Collision Alleviating DCF (CAD) protocol and showed that the performance of the proposed model is better than the existing models.

Due to absence of infrastructure, a number of problems arise in Ad hoc networks, such as bandwidth constraints, security, mobility, hidden node and exposed node problems, reliability and routing mechanisms [9]. Packet loss occurs in Ad hoc networks due to frequent link failures because of the mobility of nodes. Due to sharing of wireless bandwidth among Ad hoc nodes, medium access control may rely on physical carrier sensing multiple access mechanism with collision avoidance to determine the idle channel, such as in the IEEE 802.11 DCF [10]. In wireless networks, the channel is non-uniformly shared which leads to hidden terminal problem in Ad hoc networks.

Since communication through Ad hoc networks plays a major role in emergencies, it is very important to design an efficient MAC protocol for wireless Ad hoc networks. To improve the performance of nodes in Ad hoc as well as in congested wireless networks, designing of a backoff algorithm is crucial. Several backoff algorithms have been proposed in the literature for IEEE 802.11 based networks. The Binary Exponential Backoff (BEB) algorithm is used in IEEE 802.11 Wireless Local Area Network (WLAN) standard. Algorithms such as Double Increment Double Decrement (DIDD) [11], Exponential Increase Exponential Decrease (EIED) [12], Hybrid backoff (HB) algorithm [13], Binary Negative Exponential Backoff (BNEB) algorithm [14], Contending Stations Backoff Algorithm (CSBA) [15], etc. have been proposed in the literature for the IEEE 802.11 DCF protocol to reduce the number of collisions and to improve system performance. A backoff algorithm was proposed in which the size of the contention win-

dow is adjusted according the number of stations [16]. Authors have proposed an Adaptive minimum Contention Window Binary Exponential Backoff (AWBEB) algorithm in which the size of contention window for the next packet depends on the backoff stage from which the previous packet was transmitted successfully [17]. The history of the past trials of transmission is considered in History Based Adaptive Backoff (HBAB) algorithm [18]. Size Based Adaptive (SBA) backoff algorithm has been developed to provide Quality-of-Service (QoS) requirements in mobile Ad hoc networks [19]. However, these algorithms (except BEB algorithm) have not been designed for finite load conditions.

In binary exponential backoff algorithm, the node increases its contention window (CW) exponentially when the current packet transmission is unsuccessful. So, the node has to sense the channel for a long time to transmit data. Any other node which senses the channel at the same time with a lower contention window size gets access to the channel. The other node may be in the coverage area of the current transmitting node or may be a hidden node. The node which is in the coverage area captures the channel and transmits data. As hidden nodes cannot sense the transmission of other nodes, they can start their transmissions with lower contention window size which leads to packet collisions. In this case, the actual node keeps on doubling its contention window size leading to wastage of time and energy. Due to the above-mentioned reasons, the throughput of the wireless network decreases and the average delay required for packet delivery increases. Hence, the current backoff algorithm is inefficient for congested wireless and Ad hoc networks. To solve these problems, a new Contention Window based Multiplicative Increase Decrease Backoff (CWMIDB) algorithm is proposed in this paper to reduce the number of collisions. The CAD protocol which is developed in our previous research is used to avoid channel capture effect [8]. So, the CAD protocol with CWMIDB algorithm is considered here to reduce the collisions and to enhance QoS parameters such as throughput and End-to-End delay in wireless networks under finite load conditions.

2 CWMIDB Algorithm

The purpose of using CWMIDB algorithm in IEEE 802.11 is to minimize collisions during contention between multiple nodes and also in the presence of hidden nodes. This is a very simple backoff algorithm in which the size of the contention window increases and decreases rather than increasing exponentially.

In CWMIDB algorithm, whenever a node wishes to transmit data, it senses for a carrier on the channel. If the channel is idle for a period of time equal to distributed interframe space (DIFS), the node transmits. If the frame is correctly received, the receiving node sends an acknowledgment (ACK) frame after another fixed Short interframe space (SIFS) period. If the channel is sensed busy for a period of DIFS, the node generates a random backoff interval before transmitting, to minimize the probability of collision with packets being transmitted by other nodes. Initially, the backoff time is uniformly chosen in the range $(0, CW_{min} - 1)$ where CW_{min} is the minimum contention window size. The backoff time counter is decremented as long as the channel is sensed idle, "frozen" when a transmission is detected on the channel, and reactivated when the channel is sensed idle again. The node transmits when the backoff time reaches zero. If two nodes have the same backoff value at any given instant, they transmit at the same time and eventually, a collision occurs. When a node detects a failed transmission (it does not receive the ACK of a frame), it reschedules the backoff procedure.

2.1 Algorithm

- Step 1: Initialize Contention window CW to its minimum value CW_{min} and the maximum value of CW to its maximum value CW_{max} according to the specifications of the IEEE 802.11 standard.
- Step 2: Sense the channel for a period of DIFS. If the channel is idle, transmit the data frame and go to step 5. Otherwise go to step 3.
- Step 3: Set the backoff counter (BOC) between 0 and $(CW_{min} - 1)$.
- Step 4: Decrement the BOC by 1 if the channel is idle. If BOC reaches zero, transmit the data frame.
- Step 5: If acknowledgment received, go to step 1 to transmit the next data frame. Else go to step 6.
- Step 6: If the acknowledgement is not received select the BOC between 0 and $(W_i - 1)$ if $CW < CW_{max}$. Else $CW = CW_{max}$. Here $W_i = 2^{i/2}W$ for even values of i and $W_i = 4X2^{(i-1)/2}W$ for odd values of i where i is the retransmission attempt and $W = CW_{min}$. Go to Step 4.

2.2 Markov Chain Model

The state transition diagram of the Markov chain model for the CAD protocol is shown in Fig. 1 [8]. In this model, to avoid channel capture and reduce contention among nodes, after successful transmission of a packet at any backoff stage, the node waits for a random backoff interval to access the channel again. Under saturated conditions, the node selects this interval between $(0, CW_{min} - 1)$ at post backoff stage. In unsaturated conditions, where the packet arrival follows the Poisson's process, the node waits in idle state $(-1, 0)$ at post backoff stage until the next packet arrives in its queue. When the packet arrives at the node's buffer and the channel is idle, from $(-1, 0)$ state, it goes to state $(0, 0)$ and transmits the packet. When the channel is busy, it selects the CW between 0 and $(CW_{min} - 1)$. The selection of the contention window in the next stage depends on the backoff algorithm as explained in Sect. 2.1. The packet can be transmitted at any backoff stage when its backoff counter is zero.

Consider the number of contending nodes as fixed, defined as n . Let $b(t)$ and $s(t)$ be the stochastic process representing the backoff counter and the backoff stage $(0, \dots, m)$ respectively for a given node at slot time t , where m is the maximum backoff stage. In order to consider the non-saturated traffic, we define q as the probability of having at least one packet in the node's buffer to transmit. Similar to [3], the key approximation in the proposed model is that, at each transmission attempt, and regardless of the number of retransmissions suffered, each packet collides with constant and independent probability P_{col} . Also, it is assumed that transmission errors due to imperfect channel can occur with probability P_e and the channel is busy with probability P_b . The collision and transmission error probabilities are assumed to be statistically independent [15]. Here, the state of each node is described by $\{i, k\}$, where i indicates the backoff stage $(0, \dots, m)$ and k indicates the backoff delay. The backoff delay takes the values $(0, 1, \dots, W_i - 1)$ where $W_i = 2^i CW_{min}$. W, W_0 and CW_{min} can be interchangeable. The contention window will be increased either due to packet collisions or due to transmission errors since a node cannot distinguish a packet collision from a transmission error. The probability of successful transmission is therefore equal to $(1 - P_e)(1 - P_{col})$, from which the equivalent probability of failed transmission P_{eq} can be expressed as $P_{eq} = P_e + P_{col} - P_e P_{col}$. The state transition diagram of the Markov chain model shown in the Fig. 1 has the following transition probabilities [8]:

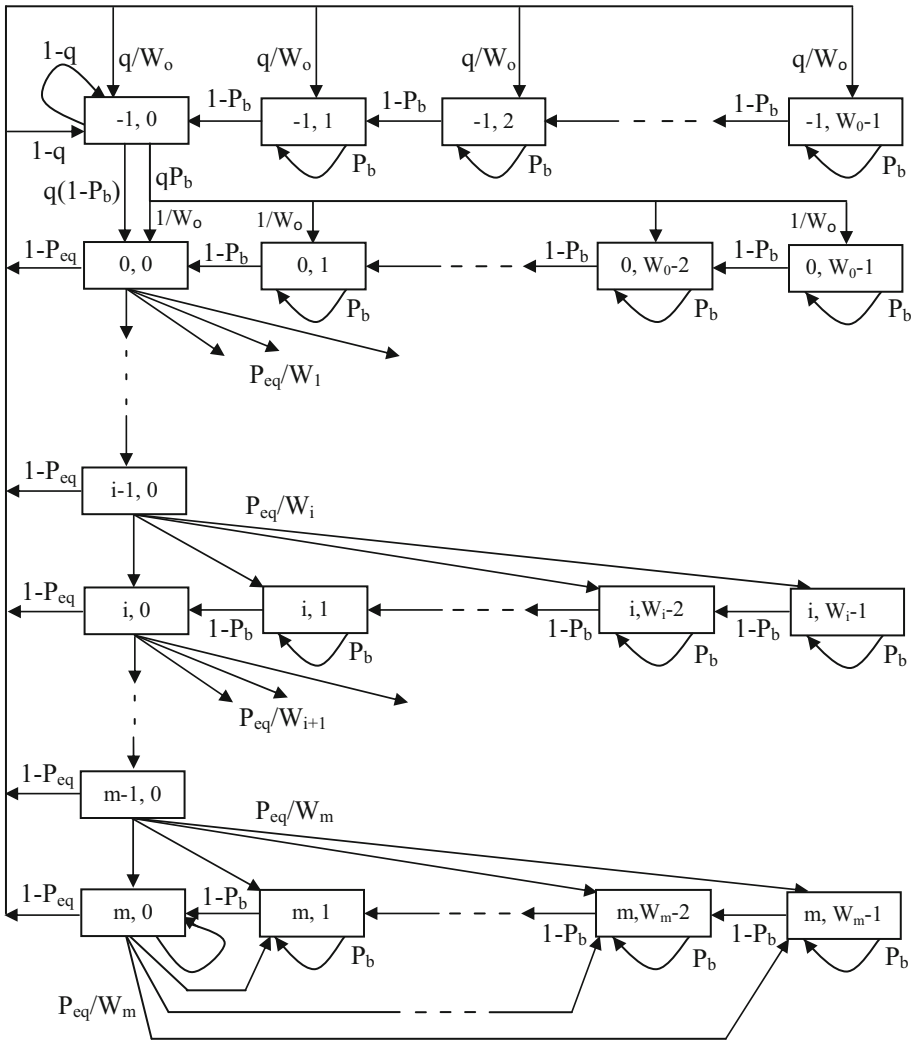


Fig. 1 State transition diagram of the Markov chain model for the CAD protocol

2.2.1 Backoff State Transitions

(i) The backoff counter decrements when the node senses the channel to be idle.

$$P \{(i, k) | (i, k + 1)\} = 1 - P_b \quad k \in (0, W_i - 2), \quad i \in (0, m) \quad (1)$$

(ii) The backoff counter freezes when the node senses that the channel is busy.

$$P \{(i, k) | (i, k)\} = P_b \quad k \in (1, W_i - 1), \quad i \in (0, m) \quad (2)$$

(iii) After each successful transmission, the node with a packet in queue goes to post-backoff stage.

$$P \{(-1, k) | (i, 0)\} = \frac{(1 - P_{cq}) q}{W_0} \quad k \in (0, W_0 - 1) \quad (3)$$

- (iv) After an unsuccessful transmission at stage $i-1$, the node reschedules a backoff delay in the next stage.

$$P \{(i, k) \mid (i - 1, 0)\} = \frac{P_{eq}}{W_i} \quad k \in (0, W_i - 1), \quad i \in (1, m) \tag{4}$$

- (v) When the transmission is unsuccessful in all the stages, the node reaches the last stage of backoff procedure and remains in that stage until the packet transmission is successful.

$$P \{(m, k) \mid (m, 0)\} = \frac{P_{eq}}{W_m} \quad k \in (0, W_m - 1) \tag{5}$$

2.2.2 Post-Backoff State Transitions

- (i) After each successful transmission, the node goes to idle state $(-1, 0)$ when the queue is empty and waits in that state until the new packet arrives in the queue.

$$\begin{aligned} P \{(-1, 0) \mid (i, 0)\} &= (1 - P_{eq})(1 - q) \quad i \in (0, m) \\ P \{(-1, 0) \mid (-1, 0)\} &= 1 - q \end{aligned} \tag{6}$$

- (ii) The node with a packet for transmission, goes to $(0, 0)$ state when the channel is free and then transmits the packet.

$$P \{(0, 0) \mid (-1, 0)\} = q(1 - P_b) \tag{7}$$

- (iii) The node with a packet for transmission, selects a backoff stage when the channel is busy.

$$P \{(0, k) \mid (-1, 0)\} = \frac{qP_b}{W_0} \quad k \in (0, W_0 - 1) \tag{8}$$

- (iv) In the post-backoff stage, the backoff counter decrements when the node senses the channel to be idle and freezes when the channel is busy.

$$\begin{aligned} P \{(-1, k) \mid (-1, k + 1)\} &= (1 - P_b) \quad k \in (0, W_0 - 2) \\ P \{(-1, k) \mid (-1, k)\} &= P_b \quad k \in (1, W_0 - 1) \end{aligned} \tag{9}$$

The probability that a node occupies a given state $\{i, k\}$ at any discrete time slot is $b_{i,k} = \lim_{t \rightarrow \infty} P\{s(t) = i, b(t) = k\}$ where i, k are integers and $-1 \leq i \leq m, 0 \leq k \leq W_i - 1$.

The following relations are valid in steady state:

$$b_{i,0} = P_{eq}^i b_{0,0} \quad i \in (1, m - 1) \tag{10}$$

In the above Eq., P_{eq} is the equivalent probability of failed transmission.

When the packet transmission is unsuccessful at any backoff stage $i - 1$, the node reschedules a backoff delay in the next stage i . The backoff counter freezes at (i, k) state when the node senses that the channel is busy. So, the steady state probability of (i, k) state $b_{i,k}$ is given by the following Eq. (11).

$$b_{i,k} = \frac{P_{eq}^i}{1 - P_b} \frac{W_i - k}{W_i} b_{0,0} \quad i \in (1, m - 1), \quad k \in (1, W_i - 1) \tag{11}$$

Based on the Eq. (10) and considering that the node keeps iterating in the m^{th} stage until the packet transmission is successful, the steady state probabilities of $(m, 0)$ and $(m - 1, 0)$ states $b_{m,0}$ and $b_{m-1,0}$ are given by

$$b_{m,0} = \frac{P_{eq}^m}{1 - P_{eq}} b_{0,0} \tag{12}$$

$$b_{m-1,0} = P_{eq}^{m-1} b_{0,0} \tag{13}$$

Similarly, the steady state probability of (m, k) state $b_{m,k}$ is given by

$$b_{m,k} = \frac{W_m - k}{W_m} \frac{P_{eq}^m}{(1 - P_{eq})(1 - P_b)} b_{0,0} \tag{14}$$

In the above Markov chain model, after successful transmission at any backoff stage, the node directly enters the state $(-1, 0)$ when there are no packets to be transmitted and keeps iterating in that state until the arrival of new packet. The stationary probability to be in state $b_{-1,0}$ can be evaluated as

$$b_{-1,0} = \frac{1 - q}{q} b_{0,0} \tag{15}$$

After successful transmission at any backoff stage, when the node is ready to transmit the next packet, the node enters the state $(-1, k)$ to provide a backoff delay between these packets to avoid channel capture. So the stationary probability to be in state $b_{-1,k}$ obtained is

$$b_{-1,k} = P_b b_{-1,k} + q (1 - P_{eq}) \sum_{i=0}^m b_{i,0} \frac{W_0 - k}{W_0} \tag{16}$$

Where,

$$\sum_{i=0}^m b_{i,0} = \frac{1}{1 - P_{eq}} b_{0,0} \tag{17}$$

Now, substituting Eq. (17) in to the Eq. (16), $b_{-1,k}$ becomes

$$b_{-1,k} = q \left(\frac{W_0 - k}{W_0} \right) \frac{1}{1 - P_b} b_{0,0} \quad k \in (1, W_0 - 1) \tag{18}$$

When the node has a packet for transmission at idle state and when the channel is busy, it selects backoff counter between 0 and W_0 . The stationary probability that the node to be in state $b_{0,k}$ is

$$b_{0,k} = \frac{P_b (1 - q)}{1 - P_b} \frac{W_0 - k}{W_0} b_{0,0} \quad k \in (1, W_0 - 1) \tag{19}$$

2.3 Derivation for Transmission Probability, τ

After each unsuccessful packet transmission, when the backoff period increases linearly or exponentially such as in BEB algorithm, the packet delay increases. On the other hand, when the backoff period decreases for each unsuccessful transmission, such as in BNEB algorithm, throughput also decreases. An efficient backoff algorithm must enhance network throughput and reduce End-to-End delay. In order to satisfy these characteristics, the backoff procedure should be modified. In CW-MIB algorithm, when collision occurs at any backoff stage, the node selects its contention window size depending on the retransmission attempt, i . For even values of i , the size of the contention window W_i becomes $2^{i/2} W$ and for odd values of i , W_i becomes $4X2^{(i-1)/2} W$.

Let L be the variable and $L = 0, 1, 2, 3$ and so on.

$$\begin{cases} W_{2L} = 2^L W \\ W_{2L+1} = 4X2^L W \end{cases} \tag{20}$$

In other words,

For even values of i , $W_{2L} = 2^L W$; $1 < L < (m - 1)/2$ and
 For odd values of i , $W_{2L+1} = 4X2^L W$; $0 < L < (m - 3)/2$

Therefore, if i is even, $W_i = 2^L W$. If i is odd, $W_i = 4X2^L W$.

According to probability conservation relation, total probability is equal to one. Therefore,

$$\sum_{i=-1}^m b_{i,0} + \sum_{i=-1}^m \sum_{k=1}^{W_i-1} b_{i,k} = 1 \tag{21}$$

The above Eq. can be expanded as

$$\begin{aligned} & \left(b_{-1,0} + \sum_0^m b_{i,0} \right) + \left(\sum_{k=1}^{W_0-1} b_{-1,k} \right) + \left(\sum_{k=1}^{W_m-1} b_{m,k} \right) + \left(\sum_{i=1}^{m-1} \sum_{k=1}^{W_i-1} b_{i,k} \right) \\ & + \left(\sum_{k=1}^{W_0-1} b_{0,k} \right) = 1 \end{aligned} \tag{22}$$

The only difference between BEB and CWMIDB algorithms is in selection of contention window at each backoff stage and hence the value of $b_{i,k}$ is different for even and odd values of i in CWMIDB algorithm. In the above Eq. (22), the derivation of all the terms except term 4 is the same for BEB and CWMIDB algorithms. Terms 1, 2, 3 and 5 can be expanded as in CAD protocol [8].

The term 4 must be derived separately for even and odd values of i . For $1 \leq i \leq m - 1$, and even values of i , the limits are 2 and $m - 1$ and for odd values of i , the limits are 1 and $m - 2$.

Therefore,

For $i = \text{even}$, $2L = 2$. Then $L = 1$; $2L = m - 1$; $L = (m - 1)/2$

For $i = \text{odd}$, $2L + 1 = 1$. Then $L = 0$; $2L + 1 = m - 2$; $L = (m - 3)/2$

Now, the term 4 can be written as

$$\sum_{i=1}^{m-1} \sum_{k=1}^{W_i-1} b_{i,k} = \sum_{k=1}^{W_i-1} \sum_{L=1}^{\frac{m-1}{2}} b_{2L,k} + \sum_{k=1}^{W_i-1} \sum_{L=0}^{\frac{m-3}{2}} b_{2L+1,k} \tag{23}$$

Now, after simplification, the term 4 becomes

$$\begin{aligned} \sum_{i=1}^{m-1} \sum_{k=1}^{W_i-1} b_{i,k} &= \frac{b_{0,0} P_{eq}^2}{2(1 - P_b)} \left(\frac{2W \left(1 - (\sqrt{2} P_{eq})^{m-1} \right)}{1 - 2P_{eq}^2} - \frac{1 - P_{eq}^{m-1}}{1 - P_{eq}^2} \right) \\ &+ \frac{b_{0,0} P_{eq}}{2(1 - P_b)} \left(\frac{4W \left(1 - (\sqrt{2} P_{eq})^{m-1} \right)}{1 - 2P_{eq}^2} - \frac{1 - P_{eq}^{m-1}}{1 - P_{eq}^2} \right) \end{aligned} \tag{24}$$

Now, substituting the equations for all the terms in to the Eq. (22), the Eq. for $b_{0,0}$ is as follows.

$$b_{0,0} = \left\{ \frac{1-q}{q} + \frac{1}{1-P_{eq}} + q \left(\frac{W-1}{2} \right) \frac{1}{1-P_b} + \left(\frac{4X2^{\frac{m}{2}}W-1}{2} \right) \frac{P_{eq}^m}{(1-P_{eq})(1-P_b)} + \frac{P_b(1-q)W-1}{(1-P_b)2} + \alpha \right\}^{-1} \tag{25}$$

Where

$$\alpha = \frac{P_{eq}^2}{2(1-P_b)} \left(\frac{2W \left(1 - (\sqrt{2}P_{eq})^{m-1} \right)}{1-2P_{eq}^2} - \frac{1-P_{eq}^{m-1}}{1-P_{eq}^2} \right) + \frac{P_{eq}}{2(1-P_b)} \left(\frac{4W \left(1 - (\sqrt{2}P_{eq})^{m-1} \right)}{1-2P_{eq}^2} - \frac{1-P_{eq}^{m-1}}{1-P_{eq}^2} \right)$$

The probability that a node transmits a packet in a randomly chosen slot time τ can be expressed as

$$\tau = \sum_{i=0}^m b_{i,0} = \frac{1}{1-P_{eq}} b_{0,0} \tag{26}$$

By substituting Eq. (25) in to Eq. (26), τ becomes

$$\tau = \frac{2q(1-P_b) \left(1 - 2P_{eq}^2 \right) \left(1 - P_{eq}^2 \right)}{2(1-P_b) \left(1 - 2P_{eq}^2 \right) \left(1 - P_{eq}^2 \right) \left[1 - P_{eq}(1-q) \right] + q(W-1)(1-P_{eq}) \left(1 - P_{eq}^2 \right) \left(1 - 2P_{eq}^2 \right) \left[q + P_b(1-q) \right] + q \left[4X2^{\frac{m}{2}}W-1 \right] P_{eq}^m \left(1 - 2P_{eq}^2 \right) \left(1 - P_{eq}^2 \right) + 2P_{eq}Wq(1-P_{eq}) \left(1 - P_{eq}^2 \right) \left[P_{eq} \left(1 - (\sqrt{2}P_{eq})^{m-1} \right) + 2 \left(1 - (\sqrt{2}P_{eq})^{m-1} \right) \right] - qP_{eq}(1-P_{eq}) \left(1 - 2P_{eq}^2 \right) \left[P_{eq} \left(1 - P_{eq}^{m-1} \right) + \left(1 - P_{eq}^{m-1} \right) \right]} \tag{27}$$

In Eq. (27), under saturated traffic conditions ($q \rightarrow 1$) when $m = 0$, i.e., when no exponential backoff is considered and assuming $P_b = P_{eq} = 0$, the transmission probability τ reduces to $2/(W+1)$. This is similar to the Eq. for the constant backoff window problem. The relation between τ , P_b and P_{col} can be found in [8]. A unique solution can be obtained on solving the equations of τ (27), P_b , P_{eq} and P_{col} using numerical methods. In the above equations, W , W_0 and CW_{min} are interchangeable.

3 Throughput Analysis

Throughput is defined as the average rate of successful data delivery over a communication channel. It is the ratio of the effective payload to the time required to transmit the payload successfully. Here, the effective payload is the size of the data within the frame or packet

excluding the MAC and physical layer headers. The payload information including the MAC header is transmitted with the same data rate. The actual data rate depends on the modulation technique used. The PHY header and the control frames (RTS, CTS and ACK) are transmitted with lower transmission rates. In this section, the performance of IEEE 802.11 protocol using CWMIDB algorithm is analyzed in terms of throughput and this is done in three different situations. First, the saturation throughput analysis is done as a function of the number of nodes under ideal channel conditions. The analysis is then extended by considering the packet errors due to transmission through Rayleigh fading channel. Finally, the throughput analysis is carried out considering the non-saturated traffic conditions. The performance of the proposed CWMIDB algorithm is compared with the legacy DCF (BEB algorithm) and various existing backoff algorithms such as HB and DIDD. CWMIDB algorithm is also compared with the CAD protocol using BEB algorithm. Even though the algorithm is suited for any IEEE 802.11 family, performance analysis is done using the network parameters of IEEE 802.11b protocol since it is easier to compare with the existing models. The performance analysis can be applied to both the access mechanisms (basic and RTS/CTS). In the analysis of throughput, the minimum value of the contention window CW_{min} plays a key role. This value depends on the protocol. The values of CW_{min} for IEEE 802.11b and IEEE 802.11a are 32 and 15 respectively.

3.1 Effect of Packet Collisions and Channel Errors on Packet Transmission

When the channel is erroneous, multipath fading occurs and hence the SNR (Signal-to-Noise Ratio) decreases, which causes packet errors. Therefore, factors that affect the error performance of wireless channels are to be considered when designing wireless network protocols. The error performance of wireless channels is usually modeled by capturing the statistical nature of the interaction among reflected radio waves. Statistical calculation for Bit Error Rate (BER), which is generally used to characterize channel errors at the physical layer, is a well-known practice. The BER for a communication channel is defined as the probability of a single bit being corrupted in a defined number of transmitted bits [20]. For example, a BER of 10^{-3} means that, 1 bit in every 1,000 bits would be corrupted on an average.

In this paper, it is assumed that degradation in network performance can be either due to packet collisions or transmission errors. In other words, packet transmission is unsuccessful when more than one node transmits the packets simultaneously, in which case, they collide with each other or the transmitted packets may be corrupted at the receiver. This assumption is due to the fact that a node cannot distinguish a packet collision from transmission errors. In both the cases, the transmitting node will not be able to receive an acknowledgement, and then it reschedules its backoff mechanism until the packet is transmitted successfully. The packet/frame error rate depends on the BER, the packet payload and the length of physical and MAC layer headers. In turn, the BER depends on the SNR, the modulation technique used and the coding scheme. The equations of frame error probability have been presented in [8] and [22].

3.2 Effect of Packet Arrival Rate on the Performance of the Network

Similar to [5], it is assumed that the packet arrival process at each node is Poisson distributed and λ is the rate at which the packets arrive at the node's buffer from the upper layers. Then the probability q that the queue contains at least one packet available at the end of a slot can be expressed as Eq. (28).

$$q = 1 - e^{-\lambda E[S_T]} \quad (28)$$

In the above Eq., the average length of slot time $E[S_T]$ when the channel errors are considered is given as

$$E[S_T] = P_{tr} P_s (1 - P_e) T_{success} + P_{tr} (1 - P_s) T_{collision} + (1 - P_{tr}) slottime + P_{tr} P_s P_e T_e \quad (29)$$

This time can be calculated by taking the successful transmission time $T_{success}$ with a probability of $P_s(1 - P_e)$, unsuccessful transmission time due to collision $T_{collision}$ with a probability of $(1 - P_s)$, unsuccessful transmission time T_e due to channel errors with a probability of $P_s P_e$ and idle slot time can be calculated with a probability of $(1 - P_{tr})$. Here, $T_{collision}$ and T_e are assumed to be the same. Here, P_{tr} and P_b can be interchanged. The equations for $T_{success}$ and $T_{collision}$ using basic and RTS/CTS access mechanisms are given in [8]. In the proposed model, under non-saturated traffic conditions, the four unknown parameters τ , p_{tr} , p_{col} and q are related to each other and these can be evaluated numerically. Here P_s is the successful packet transmission probability.

The equation for throughput when the channel errors are considered becomes

$$S = \frac{P_{tr} P_s (1 - P_e) E[P]}{P_{tr} P_s (1 - P_e) T_{success} + P_{tr} (1 - P_s) T_{collision} + (1 - P_{tr}) * slottime + P_{tr} P_s P_e T_e} \quad (30)$$

Where *Slottime* is the idle slot time and $E[P]$ is the average packet payload size.

4 End-to-End Delay Analysis

The End-to-End delay refers to the delay for a successfully transmitted packet. The End-to-End delay is the time duration that a packet lasts in the system since it has been handled by the MAC layer until an acknowledgment of its successful reception is achieved.

In this section, the average End-to-End delay, also called as packet delay, in single hop Ad hoc networks is analyzed. In Bianchi's model, the retry limit is not considered whereas in Chatzimisios' model, the retransmission limit is taken into account. But, the collision probability in these two models is almost the same for the given parameters. These two models exaggerate the collision probability as the freezing of the backoff counter when other transmissions are going on in the channel is not considered. Some researchers calculated the average packet delay for basic and RTS/CTS access modes considering retry limits, but the transmission errors were ignored. Therefore, these models do not predict the 802.11 frame delay in an accurate way. In this paper, an accurate delay analysis is made considering transmission errors and self loop probability of every backoff state. Most of the previous research was concentrated on throughput enhancement. In many cases, improvement is obtained at the expense of delay degradation. It is therefore necessary to improve the delay performance as well as throughput.

In the proposed Markov chain model, at any backoff stage, the backoff counter of a node freezes each time the interfering neighboring nodes start transmitting a packet. When the backoff timer resets to zero, the node starts transmitting the packet. At this time, the backoff timers of all interfering neighboring nodes are immediately frozen. The average delay for a single-hop network can be calculated as the product of the average number of slots required for a successful transmission and the average time interval between two consecutive time counter decrements. For multi-hop networks, this product is to be multiplied with the average number of hops [21]. In this paper, an accurate delay analysis for IEEE 802.11 DCF using

CWMIDB algorithm is obtained. A homogeneous network with n nodes is considered with packet arrival rate of λ packets/sec. To climax the effect of data rate on delay performance, the analysis is made for different data rates (1, 5.5 and 11 Mbps). Bianchi’s, Daneshgaran’s and Chatzimisios’ models overestimate the collision probability since neither of the models account for the probability that the backoff counter is frozen when there is activity in the channel. The average End-to-End delay calculation is not included in either Bianchi’s model or Daneshgaran’s model. The accurate delay analysis of IEEE 802.11 wireless network is done in Chatzimisios’ and Ziouva’s models but the analysis is limited to saturated conditions.

4.1 Mathematical Analysis of End-to-End Delay of CAD Protocol using CWMIDB Algorithm

A successful transmission may occur at one of the several backoff stages. The equation for average packet delay for successful transmission can be found in [4]. Let $E[N_c]$ be the number of collisions of a frame until its successful reception. This is given by the following equation:

$$E[N_c] = \frac{1}{P_s} - 1 \tag{31}$$

Each node spends some time before accessing the channel under busy channel conditions. This is called backoff time and is represented with $E[BD]$. When the transmitted frame collides, each node has to wait for some time before sensing the channel again. Let this time be T_o . Therefore mean frame delay is the sum of average time spent by the node during collisions and during successful transmission and can be calculated as

$$E[D] = E[N_c](E[BD] + T_{collision} + T_o) + (E[BD] + T_{success}) \tag{32}$$

T_o depends on the access method and equals to

$$T_o = \begin{cases} SIFS + ACK_timeout \\ SIFS + CTS_timeout \end{cases} \tag{33}$$

4.1.1 Calculation of $E[BD]$:

The average backoff delay depends on the value of its counter and the duration for which the counter freezes when the node detects transmissions from other nodes. Channel is busy when other nodes are transmitting and also during the time when the node A is transmitting. Now, number of times that the node is idle is given by

$$E[\psi] = \frac{1}{P_b} - 1 \tag{34}$$

When the counter of a node is at state $b_{i,k}$, the average number of backoff slots $E[X]$ required to reach state 0 without taking into account the time the counter is stopped is given by

$$E[X] = \left[\sum_{i=0}^m \sum_{k=1}^{W_i-1} kb_{i,k} \right] = \sum_{k=1}^{W_0-1} kb_{0,k} + \sum_{k=1}^{W_m-1} kb_{m,k} + \sum_{i=1}^{m-1} \sum_{k=1}^{W_i-1} kb_{i,k} \tag{35}$$

The above Eq. is valid for both saturated and non-saturated traffic conditions except that a term $(W_0 + 1)/2$ is to be added for saturated conditions. In non-saturated conditions, the node remains in idle state until the packet arrives at the node’s buffer. Therefore, the post

backoff stage delay need not be considered. In saturated traffic conditions, the node enters the post backoff stage to provide IPB delay, and hence the packet experiences this delay. But, to get the generalized Eq. for both situations, the average number of slots is calculated for backoff stages 0 to m and the average slots in the post backoff stage is added in saturation scenarios.

Based on equations of $b_{0,0}$, $b_{i,k}$ and $b_{m,k}$, the equation for $E[X]$ becomes (36)

$$E[X] = \frac{b_{0,0}}{6(1 - P_b)} \times \left[\begin{aligned} &P_b(1 - q)(W_0^2 - 1)(1 - P_{eq}) \left(1 - (2P_{eq})^2\right) \left(1 - P_{eq}^2\right) \\ &+ P_{eq}^m(2^{m+4}W_0^2 - 1) \left(1 - (2P_{eq})^2\right) \left(1 - P_{eq}^2\right) + 4P_{eq}^2W_0^2 \left(1 - (2P_{eq})^{m-1}\right) \\ &\left(1 - P_{eq}^2\right) (1 - P_{eq}) + 16P_{eq}W_0^2 \left(1 - (2P_{eq})^{m-1}\right) \left(1 - P_{eq}^2\right) (1 - P_{eq}) \\ &- \left[P_{eq}^2 \left(1 - (P_{eq})^{m-1}\right) + P_{eq} \left(1 - P_{eq}^{m-1}\right)\right] (1 - P_{eq}) \left(1 - (2P_{eq})^2\right) \end{aligned} \right] \frac{1}{(1 - P_{eq}) \left(1 - P_{eq}^2\right) \left(1 - (2P_{eq})^2\right)} \tag{36}$$

Let $E[N_{fon}]$ be the average number of times that the node detects transmissions from other nodes and is given by

$$E[N_{fon}] = \frac{E[X]}{\max(E[\psi], 1)} - 1 \tag{37}$$

Therefore, the average time $E[S]$ that the counter stopped while listening to other nodes transmissions is

$$E[S] = E[N_{fon}] (P_s T_{success} + (1 - P_s) T_{collision}) \tag{38}$$

Now, the average backoff delay $E[BD]$ is

$$E[BD] = E[X] + E[N_{fon}] (P_s T_{success} + (1 - P_s) T_{collision}) \tag{39}$$

By substituting the above equations into the equation of $E[D]$, the mean frame delay can be calculated.

The notations used in analytical analysis are listed in Table 1.

5 Results and Discussion

In this paper, a new CWMIDB algorithm is proposed for IEEE 802.11 DCF protocol. The throughput analysis for saturated and non-saturated traffic conditions under ideal and Rayleigh fading channel is carried out and it is compared with various existing backoff algorithms such as HB, BEB, DIDD etc. Most of the developed algorithms enhance the throughput at the expense of degradation in packet delay. The proposed CWMIDB algorithm greatly improves the system performance in terms of throughput as well as End-to-End delay.

Table 1 Notations used in analytical analysis

Notation	Description
P_b, P_{Tr}	Probability that the node at the backoff stage senses the channel busy
P_s	Probability of successful transmission
P_{col}	Probability of transmission collision
P_e	Probability of transmission failure due to imperfect channel conditions
P_{eq}	Equivalent probability of failed transmission
W_0, W, CW_{min}	Initial Contention window size
τ	The probability that a station transmits a packet in a randomly chosen slot time
m	Maximum backoff stage
n	Number of contending nodes
q	The probability of having at least one packet in the stations buffer to transmit
$E[S_t]$	Expected time per slot
$DIFS$	Distributed interframe space period
$SIFS$	Short interframe space period
δ	Propagation delay
T_{ACK}, T_{RTS} and T_{CTS}	Transmission time of ACK, RTS and CTS frames
T_{PACKET}	Transmission time of data including PHY and MAC headers
λ	Packet arrival rate
$E[P]$	Average packet payload size
$T_{success}$	Average time that the channel is sensed busy because of successful transmission
$T_{collision}$	Average time that the channel is sensed busy due to collisions
T_e	Average time that the channel is sensed busy due to transmission errors

5.1 Collision Probability

Figure 2 depicts how the collision probability depends on the number of nodes for different backoff algorithms. Compared to other models, the number of collisions reduces in the proposed model using CWMIDB algorithm. For a network with 25 nodes, the collision probability reduces by 8% using CWMIDB algorithm when compared to the proposed model using BEB algorithm (CAD protocol). This reduction is even more when compared to Chatzimisios', Ziouva's and Bianchi's models.

5.2 Saturation Throughput Analysis under Ideal Channel Conditions

The throughput is analyzed assuming saturated traffic load in ideal channel conditions i.e. the traffic at each node is saturated when $q \rightarrow 1$ and ideal channel conditions occur when $P_e = 0$. Figure 3 shows the resulting throughput using basic access and RTS/CTS mechanisms. The proposed CWMIDB algorithm is compared to HB, DIDD and BEB algorithms. CWMIDB algorithm is also compared with the proposed model with BEB algorithm. When the number

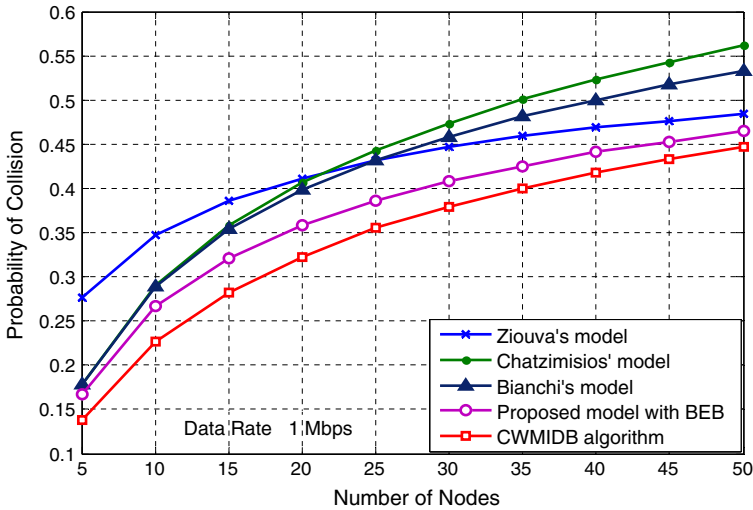


Fig. 2 Collision probability as a function of number of nodes

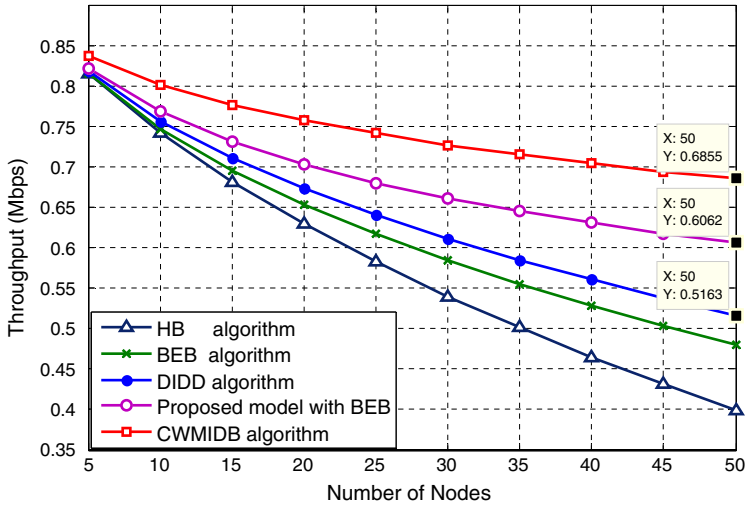
of nodes increases, more collisions occur and hence throughput decreases. For 25 contending nodes, saturation throughputs of 0.7411, 0.6801 and 0.6408 Mbps are obtained under basic access mechanism using CWMIDB, the proposed model with BEB and DIDD algorithms respectively. That is, the throughput of the CWMIDB algorithm is improved by 8.97% compared to the proposed model with BEB algorithm and 15.7% compared to DIDD algorithm. Compared to the HB and legacy BEB algorithms, this improvement is 27.14 and 20.2% correspondingly. Figure 3(b) portrays that RTS/CTS access mechanism achieves higher throughput compared to basic access mechanism due to minimum collisions and shorter collision time.

From the above analysis, it is clear that the CWMIDB algorithm consistently outperforms other backoff algorithms due to the presence of post backoff stage, backoff freezing states and the selection of contention window for each collision.

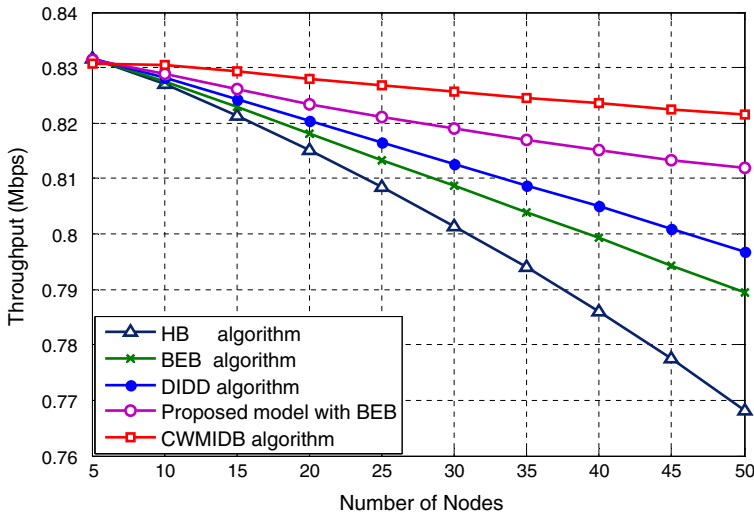
5.3 Non-Saturated Throughput Analysis Considering Transmission Errors

The HB and DIDD backoff algorithms were developed for saturated traffic and error-free channel conditions. So the throughput of CWMIDB algorithm under finite load and in erroneous channel conditions is now compared with Daneshgaran’s model (BEB algorithm) and with the developed CAD protocol using BEB algorithm. The data rate is taken as 1 Mbps and the size of MSDU is taken as 1,000 Bytes. Figure 4 depicts the variation of throughput against different packet arrival rates for different packet errors using basic access mechanism when $n = 50$. Independent of packet arrival rate, the CWMIDB algorithm achieves higher throughput. However, when λ increases above 40 Pkts/s, the saturation behavior occurs. When packets arrive at a rate of 100 Pkts/s, throughput improvement using CWMIDB algorithm is 8.6% higher compared to Daneshgaran’s model. As seen from the figure, the throughput performance of CWMIDB algorithm is better compared to that of the proposed model with BEB algorithm.

The throughput as a function of packet arrival rate using RTS/CTS mechanism is plotted in Fig. 5. When the packet arrival rate exceeds 30 Pkts/s, the throughput enters saturation



(a) Basic access

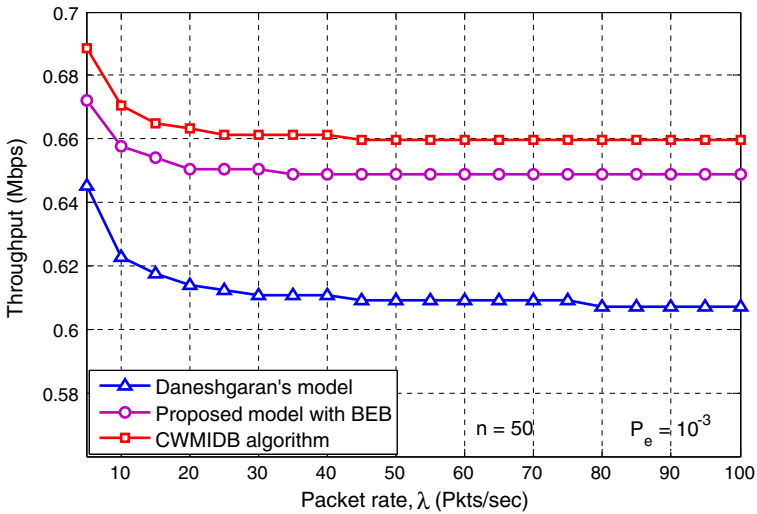


(b) RTS/CTS

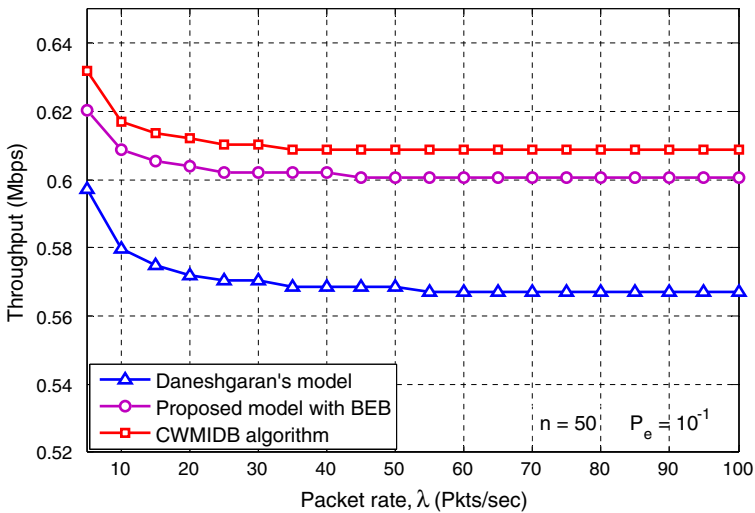
Fig. 3 Saturation throughput as a function of number of nodes

conditions. RTS/CTS mechanism is advantageous compared to basic access mechanism due to lesser packet collisions resulting in higher throughput.

Figure 6 illustrates that the throughput increases with packet length. This is because all the packets have same headers independent of the payload information. So, compared to larger payload information, the overhead is more for smaller payloads and hence the throughput is less. As the packet length continues to increase, the throughput increases at a slower rate. There is no significant change in throughput when the packet length is above 1,500 bytes and it saturates when the packet length is greater than 2,000 bytes. For a packet size of 1,000 bytes, throughputs of 0.7226, 0.7031 and 0.6912 Mbps are obtained using CWMIDB, the



(a) $P_e = 10^{-3}$



(b) $P_e = 10^{-1}$

Fig. 4 Non-saturation throughput as a function of λ for different packet errors using basic access mechanism

proposed model with BEB and Daneshgaran’s model respectively for $n = 25$. Therefore, the throughput of CWMIDB algorithm is 8% higher compared to that of Daneshgaran’s model. Even in congested environments, i.e. when $n = 50$, the performance of the proposed CWMIDB algorithm is good.

The accuracy of the proposed model can be observed under basic access mechanism. With basic access mechanism, the number of packet collisions increase which increases the freezing duration of other nodes.

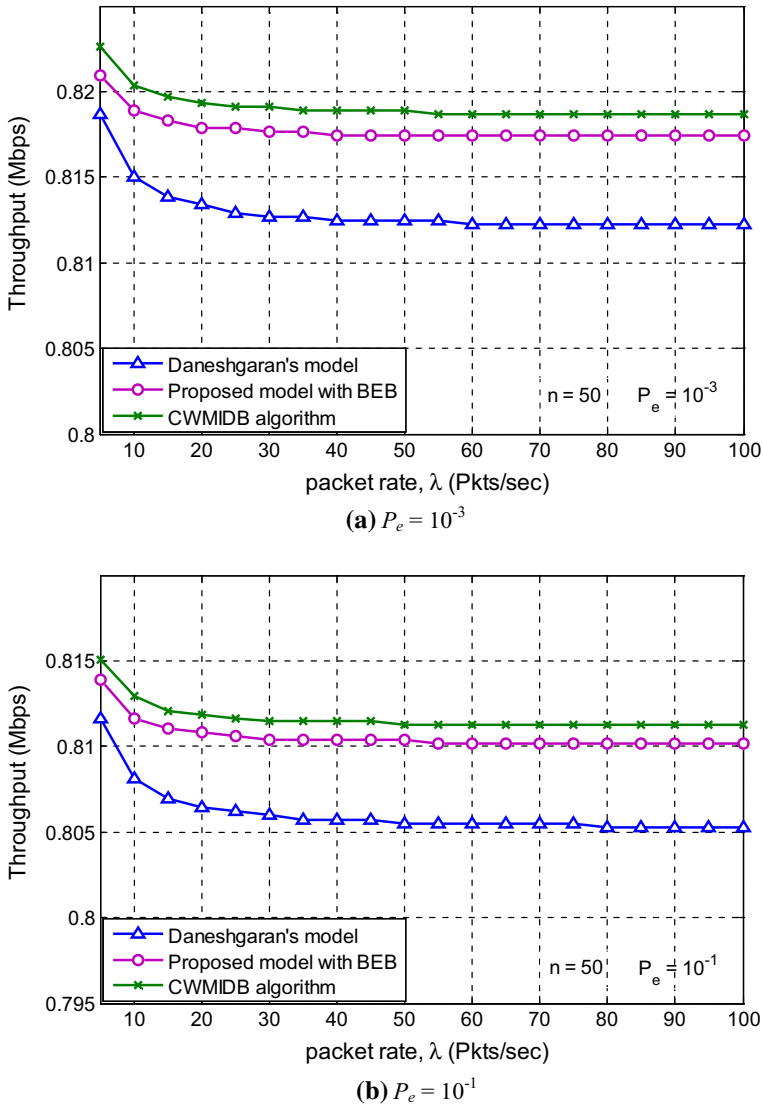
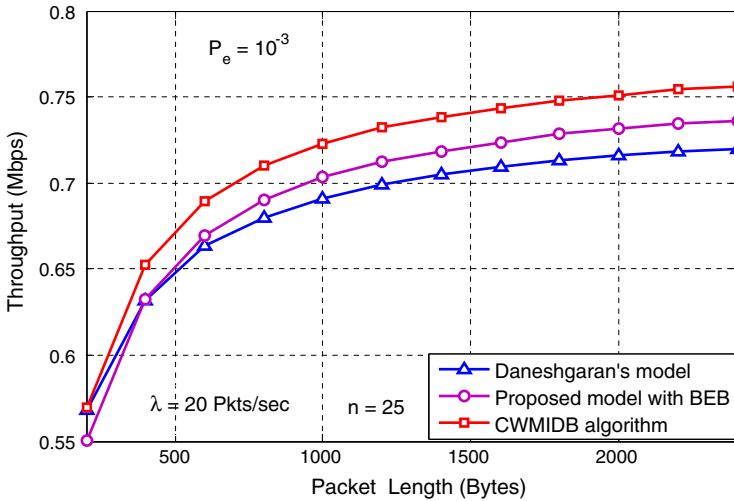


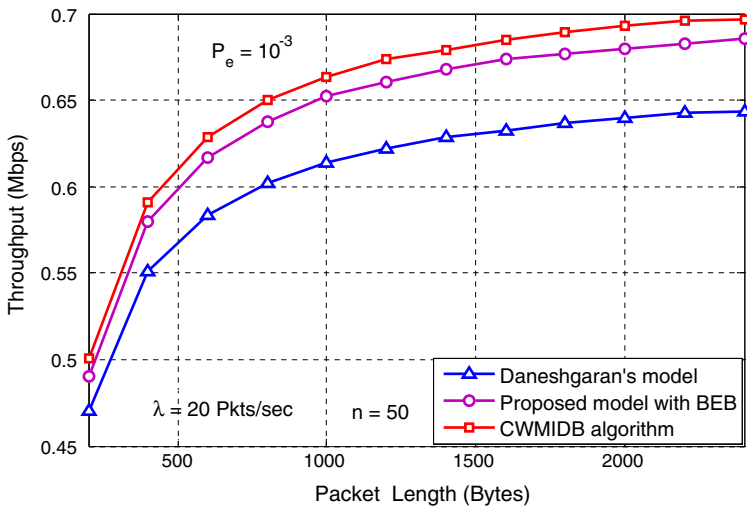
Fig. 5 Non-saturation throughput as a function of λ for different packet errors using RTS/CTS mechanism

5.4 End-to-End Delay Analysis in Error Prone Channel Conditions

Since the average End-to-End delay calculation is not included in Daneshgaran’s model, the delay performance of CWMIIDB algorithm is compared to that of the CAD protocol with BEB algorithm. The number of nodes versus End-to-End delay for three data rates (1, 5.5 and 11 Mbps) under the two access mechanisms is plotted in Fig. 7. Results show that the End-to-End delay greatly depends on data rate. When the data rate increases, packet transmission time reduces and hence the delay drops significantly. The delay performance achievable by the RTS/CTS mechanism is always higher compared to basic access mechanism due to shorter packet collision duration.



(a) $n = 25$



(b) $n = 50$

Fig. 6 Non-saturation throughput as a function of packet length for two network sizes using basic access mechanism

For a network with 25 nodes under basic access mechanism, compared to BEB algorithm, the delay using CWMIDB algorithm reduces by 62.1, 62.1 and 62.8% for data rates of 1, 5.5 and 11 Mbps respectively. This delay reduction is even more for large network sizes. Similarly, in a network with 25 nodes under RTS/CTS access scheme, using CWMIDB algorithm, the delay reduces by 61.2, 60.9 and 61.2% when the data is transmitted at 1, 5.5 and 11 Mbps respectively.

The dependency of End-to-End delay on packet arrival rate is also observed. From Fig. 8, it is observed that the delay required to transmit the packet is very less using CWMIDB algorithm compared to BEB algorithm. This reduction is 61.8% under basic access mecha-

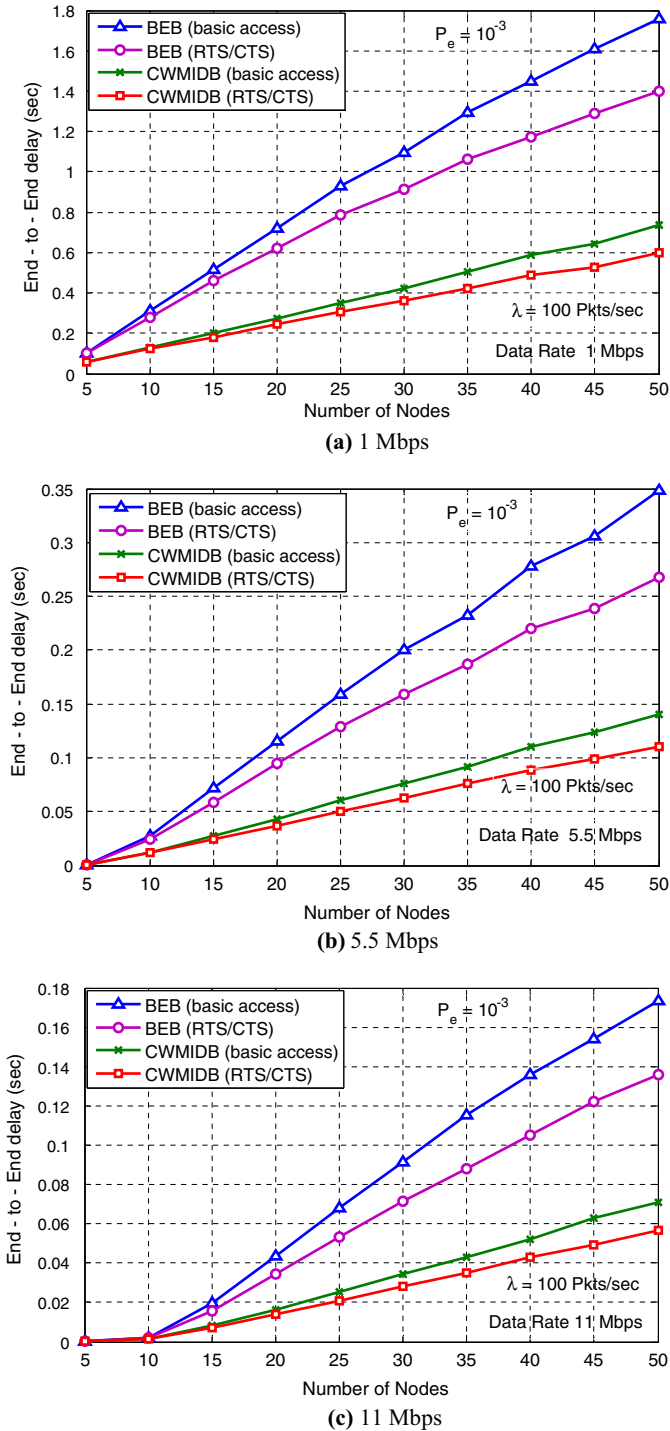


Fig. 7 Nodes versus End-to-End delay in erroneous channel for different data rates ($\lambda = 100$ Pkts/s)

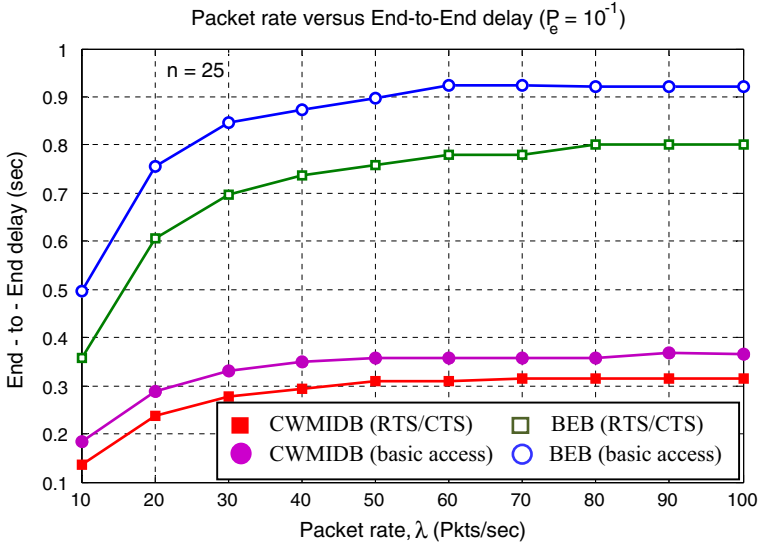


Fig. 8 Packet rate versus End-to-End delay in erroneous channel for different packet arrival rates

Table 2 Comparison of End-to-End delay for different packet lengths using basic access mechanism

Packet length (bytes)	End-to-End delay (s) - Basic access mechanism - λ = 50 Pkts/s					
	Nodes 10		Nodes 20		Nodes 30	
	BEB algorithm	CWMIDB algorithm	BEB algorithm	CWMIDB algorithm	BEB algorithm	CWMIDB algorithm
100	0.0003	0.0003	0.0415	0.1559	0.1126	0.0421
500	0.1020	0.0428	0.3413	0.1267	0.5605	0.2122
1,000	0.2745	0.1142	0.7055	0.2639	1.0956	0.4249
1,500	0.4407	0.1832	1.0498	0.4000	1.5916	0.6166
2,000	0.5855	0.2459	1.3760	0.5238	2.0870	0.8079
2,500	0.7234	0.3075	1.7017	0.6474	2.5818	0.9989

nism and 60.85 % under RTS/CTS mechanism for a packet arrival rate of 50Pkts/s when the contending nodes are 25 and $P_e = 10^{-1}$.

Finally, the delay variation with respect to packet length for three network sizes is observed and shown in Table 2 and Table 3. The delay increases with packet size because it takes a long time to transmit the large payload information. Also, the delay increases with number of nodes due to a higher number of collisions.

Table 2 shows the values of End-to-End delay for different packet sizes under basic access mechanism. In a network with 20 nodes, under basic access mechanism, to transmit a packet of 1,000 Bytes, the delay required using BEB algorithm is 0.7055 s. But using CWMIDB algorithm, only 0.2639 s is required to transmit the same data because this algorithm avoids channel capture effect and reduces the number of collisions. In other words, the delay is greatly reduced by 62.6%. Similarly for the same combination, the delay reduces by 61.5% under RTS/CTS mechanism. This can be observed from the Table 3. The parameters listed in Table 4 are used to analyze the performance of the CAD protocol with CWMIDB algorithm.

Table 3 Comparison of End-to-End delay for different packet lengths using RTS/CTS mechanism

Packet length (bytes)	End-to-End delay (s) - RTS/CTS mechanism - $\lambda = 50\text{Pkts/s}$					
	Nodes 10		Nodes 20		Nodes 30	
	BEB algorithm	CWMIDB algorithm	BEB algorithm	CWMIDB algorithm	BEB algorithm	CWMIDB algorithm
100	0.0037	0.0025	0.0751	0.0287	0.1558	0.0602
500	0.1015	0.0450	0.3115	0.1189	0.4897	0.1956
1,000	0.2452	0.1054	0.5960	0.2296	0.8851	0.3517
1,500	0.3821	0.1643	0.8813	0.3377	1.2878	0.5105
2,000	0.5145	0.2199	1.1375	0.4450	1.6618	0.6585
2,500	0.6390	0.2760	1.3933	0.5449	2.0353	0.8063

Table 4 Simulation parameters

Channel bit rate	1, 5.5 and 11 Mbps
PHY header	24 bytes
MAC header	28 bytes
RTS	20 bytes + PHY header
CTS	14 bytes + PHY header
ACK	14 bytes + PHY header
DIFS	50 μs
SIFS	10 μs
Slot time	20 μs
Propagation delay, δ	1 μs
CW_{\min} (slots)	31

6 Conclusions

In this paper, a new CWMIDB algorithm for WLANs is proposed. The Markov chain model developed for the CAD protocol is used as a base for developing the CWMIDB algorithm. The proposed algorithm can be implemented with a slight modification to the current BEB algorithm used in the legacy DCF protocol. The main feature of this algorithm is to avoid channel capture and improve the performance of the wireless network. In this algorithm, the size of the contention window will not be doubled immediately after a collision. Instead, it will be selected based on the number of collisions. The collision probability has been analysed for the proposed algorithm. The effects of the number of nodes, packet arrival rate, packet size, packet error rate and data rate on the performance of the network have been analyzed. Results show that the proposed CAD protocol with CWMIDB algorithm provides an edge over the existing backoff algorithms in terms of throughput and delay. It is also observed that throughput increases and the average delay decreases under RTS/CTS mechanism, as compared to basic access mechanism. The performance of the CWMIDB algorithm is very efficient, especially when basic access mechanism is employed in highly congested environments.

References

1. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. (1999). ANSI/IEEE Std. 802.11.
2. Bianchi, G. (1998). IEEE 802.11 saturation throughput analysis. *IEEE Communications Letters*, 2(12), 318–320.
3. Bianchi, G. (2000). Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3), 535–547.
4. Ziouva, E., & Antonakopoulos, T. (2002). CSMA/CA performance under high traffic conditions: Throughput and delay analysis. *Computer Communications*, 25(3), 313–321.
5. Daneshgaran, F., Laddomada, M., Mesiti, F., & Mondin, M. (2008). Unsaturated throughput analysis of IEEE 802.11 in presence of non ideal transmission channel and capture effects. *IEEE Transactions on Wireless Communications*, 7(4), 1276–1286.
6. Hung, F.-Y., & Marsic, I. (2007). Analysis of non-saturation and saturation performance of IEEE 802.11 DCF in the presence of hidden stations. In *Proceedings of 66th IEEE conference on vehicular technology*, pp. 230–234.
7. Malone, D., Duffy, K., & Leith, D. J. (2007). Modeling the 802.11 distributed coordination function in non-saturated heterogeneous conditions. *IEEE/ACM Transactions on Networking*, 15(1), 159–172.
8. Sasi Bhushana Rao, G., & Madhavi, T. (2013). Performance analysis of collision alleviating distributed coordination function protocol in congested wireless networks - a Markov chain analysis. *IET Networks*, 2(4), 204–213. doi:10.1049/iet-net.2012.0187.
9. Singal, T. L. (2010). *Wireless communications*. Tata McGraw Hill.
10. Al-Jubari, A. M., Othman, M., Ali, B. M., & Hamid, N. A. W. A. (2011). TCP performance in multi-hop wireless ad hoc networks: Challenges and solution. *EURASIP Journal on Wireless Communications and Networking*, 1, 1–25.
11. Natkaniec, M., & Pach, A. (2000). An analysis of modified backoff mechanism in IEEE 802.11 networks. In *Proceedings of the Polish-German teletraffic symposium (PGTS 2000)*, pp. 89–96.
12. Wu, H., Lin, Y., Cheng, S., Peng, Y., & Long, K. (2003). IEEE 802.11 distributed coordination function: Enhancement and analysis. *Journal of Computer Science & Technology*, 18(5), 607–614.
13. Xiangyu, P., Letian, J., & Guozhi, X. (2007). Performance analysis of hybrid backoff algorithm of wireless LAN. In *Proceedings of 5th international conference on WiCom*, pp. 1853–1856.
14. Choi, B. G., Bae, S. J., Lee, T. J., & Chung, M. Y. (2009). Performance analysis of binary negative-exponential backoff algorithm in IEEE 802.11a WLAN under erroneous channel condition. In *Proceedings of ICCSA*, pp. 237–249.
15. Sartthong, J., & Sittichivapak, S. (2012). Backoff algorithm optimization for IEEE802.11 wireless local area networks. In *Proceedings of ECTI-CON*, pp. 1–4.
16. Jang, K. W. (2004). A new backoff algorithm to guarantee quality of service over IEEE 802.11 wireless local area networks. In *Proceedings of WONS*, pp. 371–376.
17. Fan, J., Gao, F., Wang, W. S., & Dong, G. F. (2008). Performance analysis of an adaptive backoff scheme for ad hoc networks. In *Proceedings of IEEE international conference on CIT*, pp. 624–629.
18. Albalt, M., & Nasir, Q. (2009). Adaptive backoff algorithm for IEEE 802.11 MAC protocol. *International Journal of Communications, Network and System Sciences*, 2(4), 300–317.
19. Zhalehpoor, S., & Shahriar Shakhoseini, H. (2009). SBA backoff algorithm to enhance the quality of service in MANETs. In *Proceedings of international conference on signal acquisition and processing (ICSAP 2009)*, pp. 43–47.
20. Chatzimisios, P., Boucouvalas, A., & Vitsas, V. (2004). Performance analysis of IEEE 802.11 DCF in presence of transmission errors. In *Proceedings of IEEE international conference on communications* (7), pp. 3854–3858.
21. Babich, F., & Comisso, M. (2009). Throughput and delay analysis of 802.11-based wireless networks using smart and directional antennas. *IEEE Transactions on Communications*, 57(5), 1413–1423.
22. Mahasukhon, P., Hempel, M., Sharif, H., Zhou, T., Ci, S., & Chen, H.-H. (2007). BER analysis of 802.11b networks under mobility. In *Proceedings of IEEE international conference on communications (ICC '07)*, pp. 4722–4727.



Tatineni Madhavi received her M.Tech in Radar and Microwave Engineering in 2004 and Ph.D in 2013 from Andhra University, India. She is currently working as Associate Professor in GITAM University, Visakhapatnam, India. She has over 20 years of experience in Industry and Teaching. She has published several papers in International/National Journals/Conferences. Her main research interest includes modeling and performance analysis of wireless communications systems, Cellular/mobile communications and Wireless Sensor Networks.



Gottapu Sasi Bhushana Rao is presently working as a Professor and Head of the Department in Department of Electronics and Communication Engineering, Andhra University College of Engg., Visakhapatnam, India. He has over 28 years of experience in Industry, R&D and Teaching. He has published more than 290 research papers in various reputed International/National Journals/Conferences. He is a Senior Member of IEEE, FIETE, IEEE Com Society, IGU and International Global Navigation Satellite System (IGNSS), Australia. He is also the Chartered Engineer for IETE. His area of research includes GPS/INS Signal Processing, Cellular/mobile communications, RADAR, SONAR and Acoustic signal modeling. He Received “Dr. Survepalli Radhakrishnan Award for Best Academician” in the year 2008 and “Best Researcher Award” from Andhra University, Visakhapatnam, during 75th Convocation Platinum Jubilee Celebrations-2008. He has completed 5 research projects and 2 projects are under progress. He is also the Chief Editor of the Journal of Communication, Navigation and Signal Processing, Andhra University College of Engg., Editor of the Journal of Positioning (POS), POS, USA.