Virtual Backoff Algorithm: An Enhancement to 802.11 Medium-Access Control to Improve the Performance of MANETS

Bhuvaneswari.Rapuru[#1], Y.Neeraja[*2]
Narayana Engineering College,Gudur,India.
[#1] bhunasweetoo@gmail.com
[*2]neerajareddy_31@yahoo.co.in

***Abstract*—This paper presents a scheme, called the virtual back-off algorithm (VBA), which is based on the sequencing technique for efficient medium-access control. The proposed method min- imizes the number of collisions while reducing the delays that occur during the backoff periods. We present an analytical study on MAC-layer issues, which are very important when accessing a channel over wireless networks. The VBA scheme uses fair distrib- uted mechanisms to access a channel. We introduce a counter at each node to maintain the discipline of the nodes. The performance of the proposed method is evaluated under various conditions, and the obtained results are very promising. The enhanced pro- tocol improves the utilization of bandwidth by increasing the throughput up to 75%, and the amount of collisions is reduced to65% when compared with legacy protocols. The proposed scheme shows that the energy requirements are minimum due to the limitation on the number of transmissions.**

***Index Terms*—Channel access, distributed coordination function (DCF), medium-access control (MAC), performance evaluation.**

I. INTRODUCTION

THE main objective of medium-access control (MAC) is to provide effective sharing of the channel. The wireless MAC protocol should support various flows with fairness by providing effective channel access and bandwidth. However, this goal involves difficult and challenging tasks while imple- menting MAC protocols [1].

The IEEE 802.11 MAC protocol is the key protocol that is used for efficient channel access with time-bound and contention-free access in wireless networks [2]–[7]. The 802.11 carrier-sense multiple access with collision avoidance MAC protocol has two methods [5], [6] of channel access, namely, the distributed coordination function (DCF) and the point coor- dination function (PCF). The DCF provides distributed channel access, whereas the PCF provides centralized channel access using a coordinator. The PCF uses polling to provide channel access for the nodes.

Our focus is on DCF protocol and it is important to note that the performance of the wireless network will be influenced by the backoff methods of DCF protocol that are used during channel access. Slotted binary exponential (SBE) [3], [8] and binary exponential backoff (BEB) [1], [3], [9] are some of the backoff procedures that are used in the DCF protocol. In the SBE backoff method [3], [8], a number of slots are divided into idle DIFS periods, and in the beginning, these periods shouldbe used to transmit the data by any station. In BEB [1], [3], [9], when a station wants to transmit a data packet, then it must first carrier sense the medium. If the channel is idle for at least the DIFS duration, the node randomly picks a backoff counter value, which is randomly selected from a uniform distribution interval [0, CW], where CW is the integer within the range CW min $\leq$ CW $\leq$ CW max. If the medium is sensed to be busy, the node must defer its transmission until the medium is idle for at least the DIFS duration before selecting a backoff value. BEB aims to distribute the idle slots by giving random backoff values. The backoff value represents the number of idle slots a node has to wait before it can transmit the data. It is well noted that an efficient backoff algorithm would increase the performance of wireless networks.

There is a wide range of methods discussed in the literature to provide better channel access. One of the most popular methods is BEB, and it suffers from collisions [1], [3], [9]. The chosen range of the random backoff period is critical to the performance of the MAC protocol. If the range of the backoff period is too large, then much bandwidth will be wasted in the idle state. If the range of the backoff period is too small, then collisions are likely to happen, and it will lead to the wastage of bandwidth as well. Thus, it is an interesting problem of optimization. In this paper, we propose a procedure, called the virtual backoff algorithm (VBA), to minimize the number of collisions and to increase the throughput of the system. The proposed VBA adopts the sequencing technique, which is described in [10]–[12]. The VBA gives a solution to MAC by proposing an alternate method to backoff algorithms. We present two variants of the VBA, namely, the VBA with no counter-sharing information (VBA-NCS) and the VBA with counter sharing (VBA-CS). The basic principle of our method is to limit the number of transmissions of a node based on a sequence number.

Section II explains the preliminaries of 802.11 MAC and its mechanisms, followed by the discussion on the overview of the existing approaches. Section III presents the detailed analytical framework and the system model, and Section IV presents theorems and the proposed VBA scheme. Sections V and VI provide details of the performance analysis of the proposed method and the conclusions, respectively.

II. BACKGROUND

In this section, we present a brief review of the literature on the issues of 802.11 MAC. Baldwin *et al.* [13], [14] proposed a real-time MAC protocol for ad hoc wireless LANs. This protocol develops two methods, namely, called the transmission control (TC) procedure and the enhanced collision avoidance (ECA) procedure. TC is used for checking the deadlines of a packet after their backoff value expires. The packet trans- mission will be deferred if the deadline of the packet expires; otherwise, the ECA procedure is used. In ECA, the current backoff value will be compared with the backoff values of other stations. The station that has lowest value will gain channel access. The major drawback of this method is fixing and meeting the deadlines for the packet transmission. Another important observation is that if the deadline of a packet expires, then it is deferred from transmission after spending some time in the backoff procedure, and this may cause some unwanted delay for the station that is sending the packet. In addition, there is no guarantee for a station that this problem will be fixed in the next attempt.

Xiao *et al.* [15] proposed a DCF protocol, called backoff counter reservation and classifying stations (BCR-CS). In this scheme, every station should be in any of three states defined as idle, reserved, and contentious. The idle state refers to a station that is not ready to transmit any frame, whereas the reserved and contentious states refer to a station that is ready to transmit the frame. The main difference between these states is the announcement of backoff counter values, i.e., if a station successfully announces its backoff values to its neighbors, then it is in the reserved state; otherwise, it is in the contentious state. The authors presented two variants of the BCR-CS method by modifying the backoff procedures. The rationale behind this scheme is the announcement of backoff values. The success of BCR-CS greatly depends on the backoff counter information. As we know, there is always a chance for communication delay among the nodes while they transmit the backoff counter value, which obviously causes collisions in the network. Another limi- tation is to maintain the state information of various contending nodes.

Choi *et al.* [16] proposed a method, named early backoff announcement (EBA), which is a modification to the legacy DCF and aims to improve the throughput of the network. Announcing the backoff intervals before transmissions of nodes is the key idea in the EBA method. This algorithm required many updates for backoff periods due to the change in network conditions, and this method fails to reduce the number of collisions when the network size is large. This recent work on the backoff method is also fails to reduce the number of collisions and to increase the throughput. EBA may introduce same backoff intervals, and, further, it causes collisions. To increase the throughput of the system, the number of stations should properly be chosen to reduce the number of collisions, because the increase in the former increases the lat-ter. There exist a number of literature that discuss fairness issues while accessing a channel. One such literature is authored by Nitin *et al.* [8], who proposed distributed fair scheduling poli- cies for wireless networks. This provides a basis for fairly allo- cating bandwidth so that the throughput of the system increases. There are several proposals made in approximating the ideal time slots by developing various versions of backoff methods for MAC protocols [5], [8], [13]–[17]. Some methods [8], [18] give preference to throughput, and others treat fairness as an important factor. When throughput is considered to be important, it is improved at the expense of fairness, and vice versa. To evaluate the efficiency and robustness of the protocols being continually proposed, various metrics such as channel utilization and saturation throughput are introduced. The time taken by the channel for successful data transmission is re- ferred to as channel utilization. The throughput that can be achieved by the system at maximum under stable conditions is represented as saturation throughput. The amount of overhead introduced by the protocol for successful transmission and the speed with which the collisions are resolved determine the channel utilization and saturation throughput parameters. The studies on MAC show the problem of contention resolution during channel access. Sixty-eight percent of throughput is achieved when using the legacy DCF and other benchmark- ing algorithms such as ECA [13], [14], BCR-CS [15], and EBA [16]. Moreover, the collision rate is very high for these algorithms. Hence, we made an attempt to increase MAC efficiency, as well as to reduce the number of collisions by proposing the VBA algorithms, i.e., VBA-NCS and VBA-CS, based on the sequencing technique [10], [12]. The basic idea is to group the backoff time slots into groups of sizes of the sequence number. Every node relinquishes a time slot to other competing peers, i.e., they drop the RTS packet attempting to transmit in that time slot in each sequence group. The rationale behind this scheme is to provide a distributed mechanism, which utilizes the discrepancy of the sequence numbers of RTS packets among the neighboring nodes, to solve the po- tential contradictory medium access among the transmitting nodes. The use of RTS/CTS packets is optional. A request message will be used in the absence of RTS/CTS packets. The performance of the proposed VBA scheme is good when compared with the previous works such as the legacy DCF [5], [9], ECA [13], [14], BCR-CS [15], and EBA [16] methods. The advantage of the VBA scheme is the ability to lower communication overhead because the sequencing scheme does not require information sharing among the neighboring nodes. Hence, the VBA can be applied as an alternate method instead of using the legacy

DCF [5], [9], ECA [13], [14], BCR-CS [15], and EBA [16]. The performance of the method is estimated using the NS-2 simulator [19], and the results show promising performance of the VBA.

III. SYSTEM MODEL

The basic approach is to limit the total number of packet transmissions that are divided into groups for a particular node for some value K, which is defined as the sequence number. It is proposed to limit the number of packet transmissions of a node to $\leq K$ to ensure that each group of different nodes contains an equal number of transmissions for a time period T.

TABLE I
NOTATION

Notation	*Meaning*
P(t)	*Transmission Probability*
P(s)	*Successful Transmission Probability*
N	*Total number of nodes*
	Arrival rate
x	*No. of occurrences*
$f(x,$ □$)$	*Poisson function*
P(N=K)	*Probability function of K*
CW	*Control Window*

This approach is illustrated by Krishna *et al.* [10], [12].Let N be the total number of participating nodes, and let $p(t)$ be the probability that a node sends a frame during a time period. Let $p(s)$ be the successful transmission probability of a node, and it can only be estimated under the assumption that no other node sends the frame at the same time. Thus, $(1 - p(t))$ is the no transmission probability of other nodes during a time period (see Table I for notations). Now, $p(s)$ can be expressed as

$$p(s) = p(t) * (1 - p(t) \text{ - - - - - - - - - } (1)$$

If there are N nodes transmitting the packet, then the successful transmission probability is

$$p(s) = N * p(t) * (1 - p(t))^{N-1} \text{......................} (2)$$

The system is modeled using the Poisson distribution for estimating the occurrence of the number of events independent of time. The Poisson process is used to calculate the number of attempts made by a station (arrivals) during a time period.

The probability that there exist x occurrences when λ is the expected number of arrivals during a time period is given by

$$f(x,\lambda) = \frac{(\lambda)^x (e)^{-\lambda}}{x!} \text{..} (3)$$

In (3), λ is the rate of arrivals, i.e., the average number of arrivals per unit of time. If N_t is the number of occurrences before time t, then

$$P(Nt = k) = f(k;\lambda t) = \frac{e^{-\lambda t}(\lambda t)^k}{k!} \text{ ....} (4)$$

The success probability $P(s)$ of M nodes is

$$P(s) = M.Pt.(1 - Pt)^{M-1} \text{..........................} (5)$$

$$P(s) = M.\frac{e^{-\lambda t}(\lambda t)^k}{k!}.\left(1 - \frac{e^{-\lambda t}(\lambda t)^k}{k!}\right)^{M-1} \text{...............} (6)$$

The idle probability of N nodes is

$$P(i) = (1 - P(t))^N \text{...} (7)$$

A. Average Attempt Rate

The arrival rate λ depends on the size of the control window CW[CW_{min},CW_{max}], and it is estimated as

$$\lambda(t) = \sum_{i=1}^{M} \frac{1}{CWi(t)} \text{...} (8)$$

The attempt rate $\lambda(t)$ at a given time is dependent on the sequence number K.
Hence, the attempt rate is

$$\lambda(t) = \sum_{i=1}^{N} \frac{1}{Ki} \text{...} (9)$$

and the average attempt rate is

$$\lambda(t) = \frac{N}{K} \text{...} (10)$$

B. Service Time Estimation

Let t_s be the total service time of a packet when a node attempts to send information to a destination node; t_d be the time taken to send the RTS packet, which is equal to the DIFS; t_{s1} be the time taken to receive the CTS packet when an RTS packet is sent and is equal to the SIFS; t_{s2} be the time taken to send data on receipt of the CTS packet and is equal to the SIFS; t_{s3} be the time taken to receive the ACK packet when data are sent; and α, β, γ, ρ, and ε be constants.

The service time t_s for a node is calculated in the case of using the RTS/CTS mechanism as

$$ts - [[\sum_{t=1}^{K} td \times t] \times \alpha + [ts1 \times \beta + ts2 \times \gamma + ts3 \times \rho]] + \varepsilon$$

$\alpha, \beta, \gamma, \rho, \varepsilon > 0$; K is a positive integer. (11)

The service time t_s for a node is calculated in the case of without the RTS/CTS mechanism as

$$ts = [[\sum_{t=1}^{K} td \times t] \times \alpha + [ts1 \times \gamma + ts3 \times \rho]] + \varepsilon$$

$\alpha, \beta, \gamma, \rho, \varepsilon > 0$; K is a positive integer. (12)

It is understood that the service time depends on factors such as the attempt rate and the service rate. The service time can be improved if the attempt rate is controlled by achieving consensus among the nodes. The proposed VBA algorithm is designed to achieve consensus among the nodes by using the sequence number$K$. It is shown in (11) and (12) that the service time depends on the value of K and other parameters α, β, γ, ρ, and ε. Here, K denotes the number of times a node can be permitted to attempt to access a channel, whereas α, β, γ, and ρ are delay parameters. The introduction of delay parameters will help us estimate how many retransmissions of control and data packets are taking place, as well as to estimate delay variation (known as jitter) for various transmissions of the same frame. The objective of the proposed algorithm is to produce constant service time for all nodes of the wireless network.

IV. VIRTUAL BACKOFF ALGORITHM

The proposed VBA can be applied both with RTS/CTS and without RTS/CTS scenarios. We present two variants of the VBA, named VBA-NCS and VBA-CS. The performance of the VBA depends on the discipline of nodes on the ideal slot distribution. However, it is often that most of the nodes will try to access the channel during their idle slots, which will cause collisions, thereby reducing the performance of the system. The steps of the algorithm can be summarized, as shown in Algorithms 1 and 2.

Algorithm 1: Virtual Backoff With Counter Sharing (VBA-CS)

```
INPUT: STA_1, STA_2, STA_3, . . . , STA_n
//requests from various stations for channel access
Shared Variables:
for every i, 1 ≤ i ≤ n
counter[i] C {0, 1, 2, . . . ,N} initially 0, updated by stations
Sequence Number, K C {0, 1, 2, . . . ,N}, initially 0, will be
set to a positive integer
Procedure:
//Initialization
Set sequence numberK = m; //Fixing the Sequence Number
for (i = 1 to n)
counter[i] = 0;
for (i = 1 to n)
{
   while (channel access[i])
   if (counter[i]! = K)
    {
     if (channel == idle)
    {
       if (counter[i]<min(counter[i+1], counter[i+2], . . . ,
       counter[i + n]))
      access channel;
      else
    defer access;
    }
  counter[i] + +;
  }
 else
 }
 defer access;
}
```

Algorithm 2: Virtual Backoff With No Counter Sharing (VBA-NCS)

```
INPUT: STA_1, STA_2, STA_3 . . . , STA_n
//requests from various stations for channel access
Shared Variables:
for every i, 1 ≤ i ≤ n
counter[i] C {0, 1, 2, . . . ,N} initially 0, updated by stations
Sequence Number, K C {0, 1, 2, . . .N}, initially 0, will be
set
to a positive integer.
Procedure:
//Initialization
Set sequence numberK = m; //Fixing the Sequence
Number
for (i = 1 to n)
counter[i] = 0;
for (i = 1 to n)
{
  while (channelaccess[i])
  if (counter[i]! = K)
   {
    if (channel == idle)
    {
      access channel;
   else
    defer access;
   }
  counter[i] + +;
  }
 else
 defer access;
 }
```

The VBA will allow each node to access a channel a limited number of times, which is equal to a factor, i.e., the sequence number K. In other words, the number of attempts made by the node will be restricted to K. After distribution of idle slots, each node is entitled to access the channel after their slot time. The main problem arises when more than one node attempts to access the channel in the same slot time. If a node wants toaccess the channel and it senses that the channel is busy, then access of the channel is denied to that node. Otherwise, it can access the channel, provided that the counter of the node is less than the sequence number.

A. Complexity Analysis

The main complexity of the VBA is with counter update for nodes. Each node is expected to maintain a counter, and the maximum value of the counter is equal to K. Thus, when a node attempts to access the channel, then the counter value increases, and it is stopped when the counter value reaches K. This operation takes $O(1)$ complexity. However, the complexity of the algorithm increases when counter values are to be shared among the nodes. This requires $O(m)$ operations among the nodes and additional communication overhead. However, this overhead can easily be reduced by sending the counter information at the time of request for the channel by a node. In addition, the VBA can be implemented with sharing the counter information, in which case, each of the node transmissions is restricted to the value of the sequence number. This will reduce the computational complexity but will cause some collisions.

There are significantly fewer collisions when compared with existing methods. Thus, the VBA can be implemented in two ways—with or without sharing counter information among the nodes. We named these methods VBA-CS and VBA-NCS, and they are described in Algorithms 1 and 2. The key point of the algorithm is that a node can transmit packets in its slot time with maximum attempts equal to the sequence number K. The best-case complexity of the algorithm is $O(n)$ when no counter information is shared, and the worst-case complexity of the algorithm is $O(n \times m)$ when counter information is to be shared.

Lemma 1: The increase of the sequence number K increases the success probability and reduces the number of collisions.

Proof: Let p be the probability of first success of accessing a channel after a series of independent failure attempts. In addition, let x denote the number of failures $\{0, 1, 2, 3, 4, \ldots\}$ preceding the first success of independent attempts to access a channel.

Then, the probabilities of each attempt will be p, $(1-p)p$, $(1-p)P2Pp, \ldots$, and

$$E(X) = \sum xp(x) = 0.p + 1.(1-p)p + 2.(1-p)p \ldots$$

$$= pq(1+2q+3q2+\ldots\ldots)$$

$$= \frac{n(n-1)x^2}{2!} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(13)$$

However, in the VBA, a node is allowed to access the channel
until the number of attempts is equal to K.

Definition 1 (Sequence Window): A sequence window is a time period where a node is allowed to access a channel in its time slot.

It is seen that a sequence window will consist of successes and failures when a node attempts to access the channel. Thus, the sequence window will give the following probability distribution:

$P(x=r)=p(r$ success, 1 failures$) + p(r$ failures, 1 successes$)$

$$=prq + qrp \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (14)$$

It describes that a sequence window can have r number of successes before one failure and r number of failures before one success, i.e.,

$$E(X) = \sum_{i=1}^{n} rp(x == r)$$

$$= \sum_{i=1}^{n} r(prq + qrp) \ldots\ldots\ldots\ldots(15)$$

$$= p/q + q/p \ldots\ldots\ldots\ldots\ldots\ldots\ldots(16)$$

The two factors in (15), i.e., prq and qrp, represent the number of successful and unsuccessful attempts to access a channel and is called the sequence window. The length of the sequence window is less than or equal to K, and it is obvious that the proposed method should produce more number of successes than failures in a sequence window, i.e., the probability distribution function (pdf) prq should be high when compared with the pdf qrp.

Hence, the following equation will provide ideal conditions for getting more successful attempts and a few failure attempts:

$$prq = SW, \text{ where } SW = S + F$$

$$qrp = 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

S is a random variable that represents successes, and F is a random variable that represents failures.

The ideal case is $F=0$ and $S=K$, but $K=S+F=SW$, i.e.,

$$\Rightarrow prq = K$$

$$\Rightarrow prq = K = pq(1-p)-2 = p/q$$

$$\Rightarrow p = K/K + 1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (18)$$

Hence, the increase of the sequence number K increases the success probability p and reduces the number of collisions.

Definition 2 (Critical Factor): The critical factor, which is denoted γ, is defined as the expectation of the number of failures preceding the first success in a series of attempts to access a channel in a sequence window.

Lemma 2: The growth in the critical factor reduces the number of successes in accessing a channel.

Proof: Let P be the success probability in a series of attempts and x be the number of failures, i.e.,

$0, 1, 2, 3, 4 \ldots$

Then, the probabilities will be p, $(1-p)p$, $(1-p)2p \ldots$

The average number of failure attempts preceding the first success in a series of attempts to access a channel is called the critical factor γ, and it is given by

$$E(X) = \sum xp(x) = 0.p + 1.(1-p)p$$

$$+2.(1-p)2p \ldots\ldots = \gamma$$

$$\gamma = pq(1-q)-2$$

$$\gamma = \frac{(1-p)}{p} \ldots\ldots\ldots\ldots\ldots(19)$$

$$\gamma = \frac{q}{p} \ldots\ldots\ldots\ldots\ldots(20)$$

$$q = \frac{\gamma}{\gamma+1} \ldots\ldots\ldots\ldots\ldots(21)$$

$$p = \frac{\gamma}{\gamma+1} \ldots\ldots\ldots\ldots(22)$$

Hence, from (22), it is proved that the growth in the critical factor reduces the number of successes in accessing the channel.

V. PERFORMANCE ANALYSIS

In this section, we present the simulation results of the VBA. The simulation is performed using NS-2 [19]. The performance is measured by calculating the MAC efficiency and by estimating the number of collisions during transmission. The delay parameter is set based on the DIFS and SIFS times. We have set up a discrete event simulator using the features of NS-2 and investigated the performance of the proposed schemes.

We have estimated its MAC efficiency, collisions, and service time, as well as other delay parameters. We conducted our experiments with and without using RTS/CTS packets. All simulations are performed on a Linux platform using NS-2 [19] with a simulation time of 120 s. Our simulation setup include 250 wireless LAN nodes trying to access the channel with a bit rate of 11 kb/s. Other simulation parameters are listed in Table II .

TABLE II
SIMULATION PARAMETERS

Parameter	Value
DIFS	*50 µs*
SIFS	*10 µs*
MAC	*802.11DCF*
Routing	*AODV*
Network Type	*Single hop*
Number of contending	*2-250*
Network size	*250*
Consequence number, K	*50,100,150,200,250,300*
CW min	*31*
CW max	*1023*
Traffic type	*CBR*
PHY data rate	*1 Mbps*
PLCP preamble	*144 µs*
PLCP Header	*48 µs*
Frame size	*2304 octets*
ACK Timeout	*50µs*

The performance of the VBA is compared with existing standard algorithms, namely, legacy DCF, ECA, EBA, and BCR-NS. These experiments are performed with a 95% confidence level, and the confidence intervals are shown in the corresponding graphs. Fig. 1 shows the normalized MAC efficiency comparison of different nodes for these algorithms.

Our experiments show that the performance of the VBA is significantly improved when compared with other methods

.

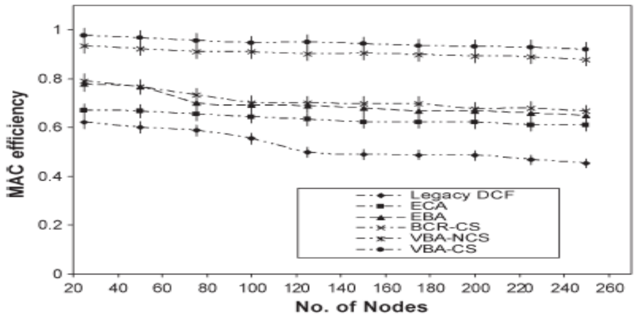

Fig. 1. VBA throughput comparison with legacy DCF and EBA.

It can be observed in Fig. 1 that both VBA algorithms produce better MAC utilization, i.e., more than 90%. Most of our experiments show that the throughput performance of the VBA scheme is improved up to 75% (average increase is 66%) when compared with the legacy DCF. In addition, the normalized MAC efficiency of the VBA scheme is improved up to 60% (an average increase of 41%) when compared with the ECA, EBA, and BCR-CS methods. We notice that the performance of the ECA, EBA and BCR-CS methods produces improvement over the legacy DCF, but when the number of nodes increases, the performance decreases. The advantage of the proposed scheme is it produces better MAC performance, although the counter information is not being shared among other stations, whereas in the case of the EBA and BCR-CS schemes, the performance depends on its backoff value announcement and sharing. It is observed that the performance of the VBA greatly depends on the sequence number *K*. The throughput of the system increases when the sequence number increases. The throughput performance of the proposed sequencing scheme for different numbers of nodes, when the sequence number increases, can be observed in Fig. 2. The optimal value for the sequence number depends on the size of the network, and it is observed that the increase of the sequence number gets saturated at one point by producing throughput close to 90%. It is observed that the sequence number saturates at 300 and above for 16 nodes, where it produces throughput between 89% and 95%

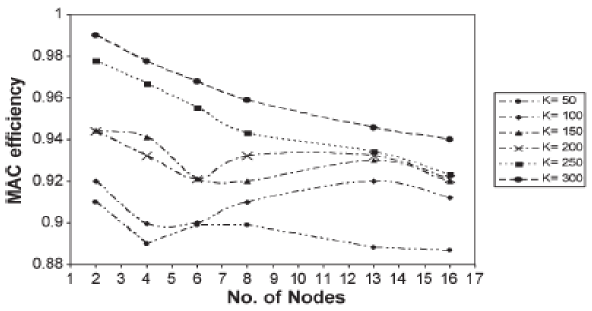

Fig. 2. VBA throughput comparison when the sequence number increases.

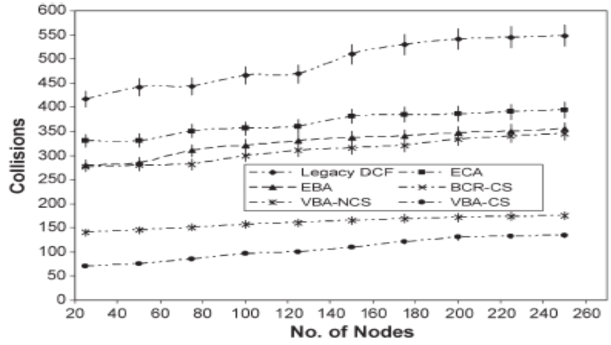

Fig. 3. Comparison of collisions for different numbers of nodes.

A. Comparison of Collisions for Different Numbers of Nodes

Fig. 3 shows the collision plot for the MAC 802.11 DCF, ECA, BCR-CS, EBA, and VBA methods for different sets of nodes. From the graph, it is observed that the number of collisions for the VBA is less than that of the MAC 802.11 DCF, as well as with EBA. It is observed that the VBA is able to reduce the number of collisions to 65% when compared with other approaches. The strength of the proposed method is significantly reducing the number of collisions and, therefore, increasing the quality of service for wireless networks.

B. Validation

The performance of the VBA is validated using the analytical models that are described in Sections III and IV. Fig. 4 shows the performance graph between the average number of successful packets (normalized) and the sequence number K. It can be observed in Fig. 4 that the simulation performance of the VBA is closer to the analytical performance of the algorithm. In addition, it is clear that, when the sequence number increases, he number of successes also increases. Hence, Lemma 1 is proved. It is to be noted that the deviation between the analytical and the simulation performance is between 2% and 7% and the average deviation is around 5%.

It is observed that, when the sequence number is large, the simulation performance of our method is closer to the analytical performance. The critical factor, which is defined in Section IV, is used to estimate the number of unsuccessful packets. It is observed from (22) that the value of the critical factor should be always as low as possible. Our algorithm is successful in maintaining a low critical factor, i.e., between 3% and 9%, against 1% to 2% of the analytical result. It is to be noted that our experiments show that the critical factor for the legacy DCF, ECA, BCR-CS, and EBA methods is between 12% and 47%, which is quite high when compared with the VBA.

The analytical curve shown in Fig. 5 has a few number of unsuccessful transmissions when the sequence number is increased. In addition, it can be seen in Fig. 5 that the simulation performance of the VBA is closer to the analytical performance. Hence, Lemma 2 is proved.

The service time is the time taken to deliver a packet. It is estimated when various nodes contend for channel access. In (11), the delay parameter α represents the number of times a node has to do the contention procedure before sending the data. The delay parameter β represents how many times a node has to send a request message for a single transmission of data. The delay parameter γ represents the number of times a packet is retransmitted, and the delay parameters ρ and ε represent the number of times a node sends an acknowledgement message and other control messages, respectively.

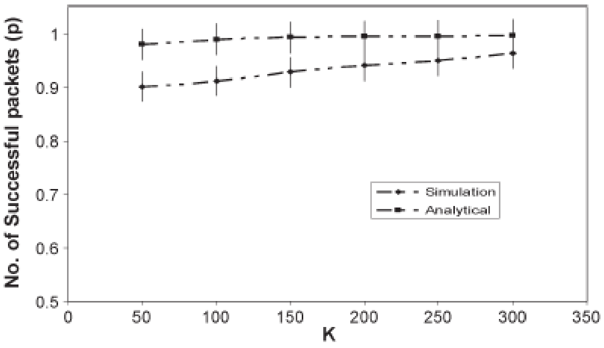

Fig. 4. Performance validation of the VBA with the analytical model when K increases.

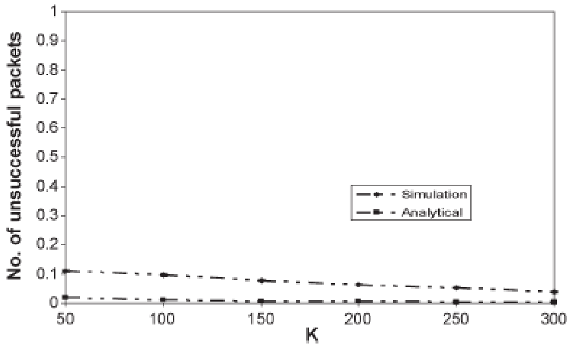

Fig. 5. Performance validation of the VBA with the analytical model when the critical factor varies.

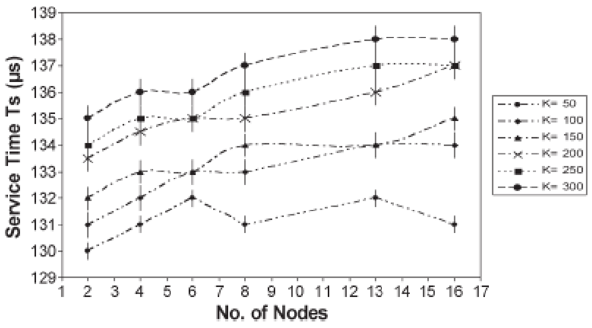

Fig. 6. Performance validation of the VBA with service time.

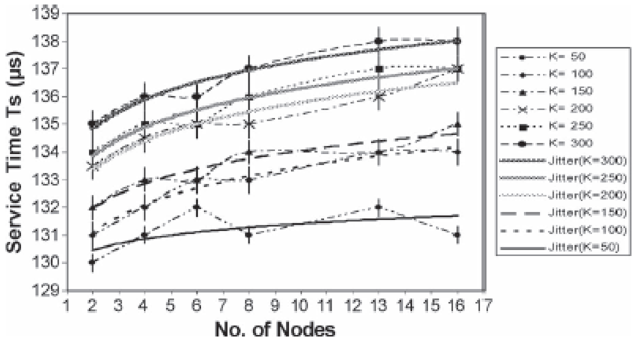

Fig. 7. Performance validation of the VBA with delay variation.

It is obvious that channel-access methods should be efficient by having less delay during contention and data-transmission periods so that the service time is minimal. Hence, it is important to devise methods that produce permissible tolerance for delay parameters α, β, γ, ρ, and ε. We analytically require 130 μs per transmission if the values of the delay parameters are $\alpha = 1$, $\beta = 1$, $\gamma = 0$, $\rho = 1$, and $\varepsilon = 0$. However, it is very difficult to achieve this as there may be many adverse affects on the system when a node attempts for channel access. Our aim is to produce the service time of the proposed algorithm closer to the ideal condition. Our experiment shows that the average service time per transmission is 134.1667 μs, which is closer to 130 μs (see Fig. 6).

In addition, the number of collisions is reduced using the VBA method when compared with the legacy DCF and EBA methods, which helps the VBA produce effective service time.

The variation in delay, which is known as jitter, is estimated for the VBA and is shown in Fig. 7. The delay variation is estimated with a 95% confidence level, and it is observed that the jitter is around 4 μs.

VI. CONCLUSION

TheMAC-layer protocol DCF has been enhanced by proposing a new method, known as the VBA, which is based on the sequencing technique. The performance of the MAC layer is good while using the VBA when compared with other methods like legacy DCF, ECA, BCR-CS, and EBA. We have observed a 65% reduction in the number of collisions, a 75% increase in throughput, and effective service time guarantees using the VBA scheme. To conclude, the proposed VBA method uses the concepts of sequencing. In the existing MAC layer, there is a certain amount of waste in the available bandwidth, which can be reduced by sequencing, whereby selected RTS/ request packets are dropped in a controlled way in the new enhanced protocol. The enhanced protocol, which has been designed and implemented in this paper, improves the utilization of bandwidth by increasing the throughput. The results are validated by the simulation results generated by using NS-2. An analytical framework has been presented in support of the proposed method. The fairness factor is good as the proposed method limits the number of attempts by grouping the backoff time slots into groups of sizes of the sequence number.

In the future, the VBA scheme can be extended to ad hoc networks such as wireless sensor networks, and the performance of sensor networks needs to be analyzed. Self-discipline of the nodes while accessing a channel is important and the performance of any algorithm depends on this agreement. Presenting the VBA with a learning approach can be a future work extending this work. In addition, detecting the malicious nodes is potentially another future work using the VBA scheme.

REFERENCES

[1] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple access modes and their throughput–delay characteristics," *IEEE Trans. Commun.*, vol. COM-23, no. 12, pp. 1400– 1416, Dec. 1975.

[2] F. Fitzek, A. Kopsel, A. Wolisz, M. Krishnam, and M. Reisslein, "Providing application-level QoS in 3G/4G wireless systems: A comprehensive framework based on multirate CDMA," *IEEE Wireless Commun.*, vol. 9, no. 2, pp. 42–47, Apr. 2002.

[3] W. Pattara-Atikom, P. KrishnaMurthy, and S. Banarjee, "Distributed mechanisms for quality of service (QoS) in wireless LANs," *IEEE Wireless Commun.*, vol. 10, no. 3, pp. 26–34, Jun. 2003.

[4] V. Marques, R. L. Aguiar, C. Garcia, J. I. Moreno, C. Beaujean E. Melin, and M. Liebsch, "An IP based QoS architecture for 4G operator scenarios," *IEEE Wireless Commun.*, vol. 10, no. 3, pp. 54–62, Jun. 2003.

[5] S. Mangold, S. Choi, G. R. Hiertz, O. Klein, and B. Walke, "Analysis of IEEE 802.11e for QoS support in wireless LANs," *IEEE WirelessCommun.*, vol. 10, no. 6, pp. 40–50, Dec. 2003.

[6] H. Zhu, M. Li, I. Chlamtac, and B. Prabhakaran, "A survey of quality of service in IEEE 802.11 networks," *IEEE Wireless Commun.*, vol. 11,no. 4, pp. 6–14, Aug. 2004.

[7] J. Schiller, *Mobile Communications*. London, U.K.: Pearson, 2001.

[8] N. Vaidya, S. Gupta, and P. Bahl, "Distributed fair scheduling in wireless LAN," *IEEE Trans. Mobile Comput.*, vol. 4, no. 6, pp. 616–628, Nov./Dec. 2005.

[9] P. Karn, "MACA—A new channel access method for packet radio," in *Proc. 9th ARRL Comput. Netw. Conf.*, 1990, pp. 134–140.

[10] P. V. Krishna and N. C. S. N. Iyengar, "Sequencing technique: An enhancement to 802.11 medium access control to improve the performance of wireless networks," *Int. J. Commun. Netw. Distrib. Syst.*, vol. 1, no. 1, pp. 52–70, Feb. 2008.

[11] P. V. Krishna, "A study of quality of service metrics for wireless systems," Ph.D. dissertation, VIT Univ., Vellore, India, Jan. 2008.

[12] P. V. Krishna and N. C. S. N. Iyengar, "Design of sequencing medium access control to improve the performance of wireless networks," *J. Comput. Inf. Technol.*, vol. CIT 16, no. 2, pp. 81–89, Jun. 2008, DOI:10.2498/cit.1001007.

[13] R. O. Baldwin, N. J. Davis, and S. E.Midkiff, "A real-time medium access control protocol for ad hoc wireless local area networks," *Mobile Comput. Commun. Rev. (MC2R)*, vol. 3, no. 2, pp. 20–27, Apr. 1999.

[14] R. O. Baldwin, N. J. Davis, S. E. Midkiff, and R. A. Raines, "Packetized voice transmission using RT-MAC, a wireless real-time medium access control protocol," *Mobile Comput. Commun. Rev.*, vol. 5, no. 3, pp. 11–25, Jul. 2001.

[15] Y. Xiao, H. Li, K. Wu, K. K. Leung, and Q. Ni, "On optimizing backoff counter reservation and classifying stations for the IEEE 802.11 distributed wireless LANs," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 7, pp. 713–722, Jul. 2006.

[16] J. Choi, J. Yoo, S. Choi, and C. Kim, "EBA: An enhancement of the IEEE 802.11 DCF via distributed reservation,"*IEEE Trans. Mobile Comput.*, vol. 4, no. 4, pp. 378–390, Jul./Aug. 2005.

[17] G. Bianchi, L. Fratta, and M. Oliveri, "Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs," in *Proc. PIMRC*, Taipei, Taiwan, Oct. 1996, pp. 392–396.

[18] F. Calì, M. Conti, and E. Gregori, "IEEE 802.11 protocol: Design and performance evaluation of an adaptive backoff mechanism," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 9, pp. 1774–1786, Sep. 2000.

[19] NS-2 Simulator. [Online]. Available: http://www.isi.edu/nsnam/ns/tutorial/ns.html