# An Adaptive Mac Scheme to Enhance the Performance of 802.11

P. Ravi Shankar[1], J. Elison Antony[2]

[1]Assistant Professor, Department of Mechatronics, Nehru Institute of Engineering and Technology
[2]Department of Mechatronics, Nehru Institute of Engineering and Technology, Coimbatore

## ABSTRACT

**The main function of Medium Access Control (MAC) is to share the channel efficiently between all nodes. In the real-time scenario, there will be certain amount of wastage in bandwidth due to back-off periods. More bandwidth will be wasted in idle state if the back-off period is very high and collision may occur if the back-off period is small. So, an optimization is needed for this problem. The main objective of the work is to reduce delay due to back-off period thereby reducing collision and increasing throughput. Here a method, called the virtual back-off algorithm (VBA) is used to optimize the back-off period and thereby it increases throughput and reduces collisions. The main idea is to optimize the number of transmission for every node. A counter is introduced at each node to implement this idea. Here counter value represents the sequence number.VBA is classified into two types VBA with counter sharing (VBA-CS) and VBA with no counter sharing (VBA-NCS). These two classifications of VBA are compared for various parameters. Simulation is done in NS-2 environment. The results obtained are found to be promising.**

**Keywords: VBA, sequence number, counter, back-off period.**

## I. INTRODUCTION

MAC layer handles various functions and one of the main functions is to provide efficient channel access. But achieving this task is very difficult due to the increasing number of nodes and changing network condition. In recent years there is tremendous increase in the usage of wireless networks, so the no of nodes is also being increased. Since the number of nodes is increased there is large possibility of collision to occur. Many authors proposed various methods to reduce the collision. Some of the methods are discussed in this section. A node which has experienced a collision on a network waits for a amount of time before attempting to retransmit. A random back-off minimises the probability that the same nodes will collide again, even if they are using the same back-off algorithm. Increasing the back-off period after each collision also helps to prevent repeated collisions, especially when the network is heavily loaded.

The IEEE 802.11 protocols are today the de-facto standards for wireless local area networks (WLANs) explained in [6],[9],[10] and [12]. The medium access control (MAC) is based on the carrier sense multiple access (CSMA) mechanism, which requires nodes to sense the channel idle for a specific time interval before attempting any packet transmission. In case the channel is sensed busy, nodes defer their transmission attempts to a later time, on the basis of a back-off mechanism. This transmission modality is referred as basic access. Collisions can still occur, in case of hidden nodes. To alleviate this problem RTS/CTS mechanism is proposed by the standard.

Here DCF protocol is mainly focused and there are two methods are used in DCF they are Slotted Binary Exponential (SBE) [11], [12] and Binary Exponential Back-off (BEB) [7]. In the SBE back-off method, a number of slots are divided into idle DIFS periods, and in the beginning, these periods should be used to transmit the data by any station. In BEB 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 back-off counter value, which is randomly selected from a uniform distribution interval [0,CW], where CW is the integer within the range CWmin ≤ CW ≤ CWmax. 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 back-off value. BEB aims to distribute the idle slots by giving random back-off values. But both the methods suffer from collision.

J. Choi [1] used a method called Early Back-off Announcement (EBA) to reduce the collision and simultaneously increase the throughput of the network. The main idea of this method is to announce the back-off intervals before the transmission of data. Here a station announces its future back-off information via MAC header. But, this method requires many updates for back-off periods due to the change in network condition and also it fails to number of collision when network size gets increased.

Y. Xiao [2] introduced a method called Back-off Counter Reservation and Classifying Stations (BCR-CS). This method announces the Back-off counter in advance and avoids possible collisions among reserved stations. But there is communication delay while transmitting back-off counter value and hence it will cause collision and it should maintain the state information of all the nodes.

R.O Baldwin [3] introduced a real time MAC protocol for Ad-Hoc network. Here two methods are used they are Transmission Control procedure (TC) and Enhanced Collision Avoidance procedure (ECA). TC is used check the deadlines of packets. If the dead line of the packet is expired then transmission will be varied. Here ECA is used to compare the current back-off value with the back-off value of other stations. A station whose back-off value is low can access the channel.

The main idea of VBA is to limit the total number of transmission of a node to a sequence number named ‘k’. So that every node can transmit equal number of times in a given period ‘T’. The idea behind this algorithm is sequencing technique mentioned in [5] and [8].

## II. ESTIMATION OF SERVICE TIME

Service time is the time taken by a node to complete its transmission. It used evaluate performance of the network, generally service time should be minimum and constant. This is one of the most important parameter to evaluate the quality of service of a network.

Total service time $t_{tot}$ can be calculated using the equation:

$$T_{tot}=[[\sum_{n=1}^{k} t_{rts} * n]*\mu+[t_{dat} * \Omega + t_{dat} * \chi + t_{ack} * \sigma]]+\tau$$

Where

$\mu, \Omega, \chi, \sigma, \tau > 0$;
K is a positive integer.

$\mu, \Omega, \chi, \sigma, \tau$ are referred as delay parameters. Here, $t_{rts}$ is the time taken to send the RTS packet, $t_{dat}$ be the time taken to send data on receipt of the CTS packet and is equal to the SIFS and $t_{ack}$ be the time taken to receive the ACK packet. The main aim of VBA is to provide constant service time.

## III. OPTIMIZATION OF BACK-OFF PERIOD

Back-off period is defined as the number of idle slots that a node has to wait before it transmits the data. Back-off period is used to avoid the collision. If this back-off period is too large there may be wastage of bandwidth and if back-off period is too small then it may lead to collision. Many methods are introduced to make this back-off period optimized. But every method has some disadvantage. So, a new method called Virtual Back-off Algorithm is used to optimize the back-off period. Still the performance of this method depends on the discipline of the nodes during the idle time slot. This method has been further classified into two types one is Virtual Back-off Algorithm with Counter Sharing (VBA-CS) and Virtual Back-off Algorithm with No Counter Sharing (VBA-NCS). In Counter Sharing method counter information of each node is shared with all other nodes in the network before accessing the channel. These two algorithms are explained below in detail.

**Virtual Back-off Algorithm with Counter Sharing:**

Step 1: Declare the number of nodes.
Step 2: Set the sequence number K.
Step 3: Initialize the counter value counter [i]=0.
Step 4: Consider a node and check the counter [i] <= K.
Step 5: check the channel is idle.
Step 6: check the node maintains correct discipline by sharing the counter information, if so, access the channel else defer.
Step 7: Increment counter [i].
Step 8: Repeat step 4 to 7 until counter [i] >K
Step 9: Consider next node and proceed the steps from 3 to 8

**Virtual Back-off Algorithm with No Counter Sharing:**

Step 1: Declare the number of nodes.
Step 2: Set the sequence number K.
Step 3: Initialize the counter value counter [i]=0.
Step 4: Consider a node and check the counter [i] < = K.
Step 5: Check the channel is idle. If so, access the channel.
Step 6: Increment counter [i].
Step 7: Repeat step 4 to 6 until counter [i] >K
Step 8: Consider next node and proceed the steps from 3 to 7

The steps involved in Virtual Back-off Algorithm (VBA) are explained above. Here in both the algorithm it is very clear that each node can access a channel during its time slot and the channel access is permitted if the channel is idle and also if the counter value of that particular node is less than or equal to the sequence number k. Else the channel access is denied for that node. Every time when a node accesses the channel its counter value is incremented by one. The sequence number is fixed and can take any positive integer. VBA-CS differs from VBA-NCS only in the step 6 where, in VBA-CS it checks for node discipline and shares the counter information among the other nodes.

The complexity of VBA lies in updating the counter value. The counter value of a node is increased by one when accesses a channel each time. This operation requires O (1) complexity for each station. Here two methods called VBA-CS and VBA-NCS are introduced. VBA-NCS method requires O(n) operation for updating counter information of n nodes. VBA-CS method requires O (n*m) operations for updating the counter of each node and sharing counter information among all the nodes. Here the best-case complexity is O (n) and worst-case is O (n*m). The main disadvantage is additional communication overheads are required for sharing the counter information among the all nodes. But, these overheads can be reduced by sending the counter information while accessing the channel. If the sequence number K is increased the success probability is also increased simultaneously the collisions are reduced.

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, …… }. Preceeding the first success of independent attempts to access a channel. Then, the probabilities of each attempt will be p, (1−p)p,…and

$$E(X) = \sum xp(x) = 0.p + 1.(1-p)p + 2.(1-p)^2 p + \cdots$$
$$= pq(1+2q+3q^2+\ldots)$$
$$E(X) = \frac{1-p}{p} \qquad (3.13)$$

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

$$P(x=r) = p(r\ \text{success},1\ \text{failure}) + p(r\ \text{failures},1\ \text{success})$$
$$P(x=r) = p^r q+q^r p \qquad (3.14)$$

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.

$$E(x) = \sum_{i=1}^{\infty} rp(x=r)$$
$$= \sum_{i=1}^{\infty} r(p^r q + q^r p) \qquad (3.15)$$

$$E(X) = \frac{p}{q} + \frac{q}{p} \qquad (3.16)$$

The two factors in (3.15), i.e., (p^r)*q and (q^r)*p, 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) (p^r)*q should be high when compared with the pdf (q^r)*p. Hence, the following equation will provide ideal conditions for getting more successful attempts and a few failure attempts:

$$p^r q=SW, \text{ where } SW = S+F$$
$$q^r p= 0 \qquad (3.17)$$

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.,

$$P^{r}q=K$$

$$p^{r}q = K= pq(1-p)^{-2} = \frac{p}{q}$$

$$p = \frac{K}{K+1} \qquad (3.18)$$

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

## V. SIMULATION RESULTS

In this section two methods of VBA are simulated and their performances are compared with some existing methods. The parameters considered for evaluating the performance of VBA are throughput, collision, service time and probability of success. All simulations are performed on a Linux platform using NS-2 with a simulation time of 160 s. The simulation setup includes 260 wireless LAN nodes trying to access the channel with a bit rate of 1 Mbps. The table below shows the value of each parameter used and the protocol used.

TABLE I: SIMULATION PARAMETERS

| Parameter | Value |
|---|---|
| DIFS | 50µs |
| SIFS | 10µs |
| MAC | 802.11 DCF |
| Routing | AODV |
| Network type | Single hop |
| Network size | 260 |
| Sequence number, k | 50,100,150,200,250,300 |
| Traffic type | CBR |
| Physical data rate | 1 Mbps |
| PLCP preamble | 144µs |
| PLCP header | 48µs |
| Frame size | 2304 octets |
| Acknowledge time out | 50µs |

Throughput of a network shows the total number of packets successfully transmitted from source to destination. From the fig 1 it has been proved that VBA-CS and VBA-NCS have better throughput when compared to EBA and ECA.

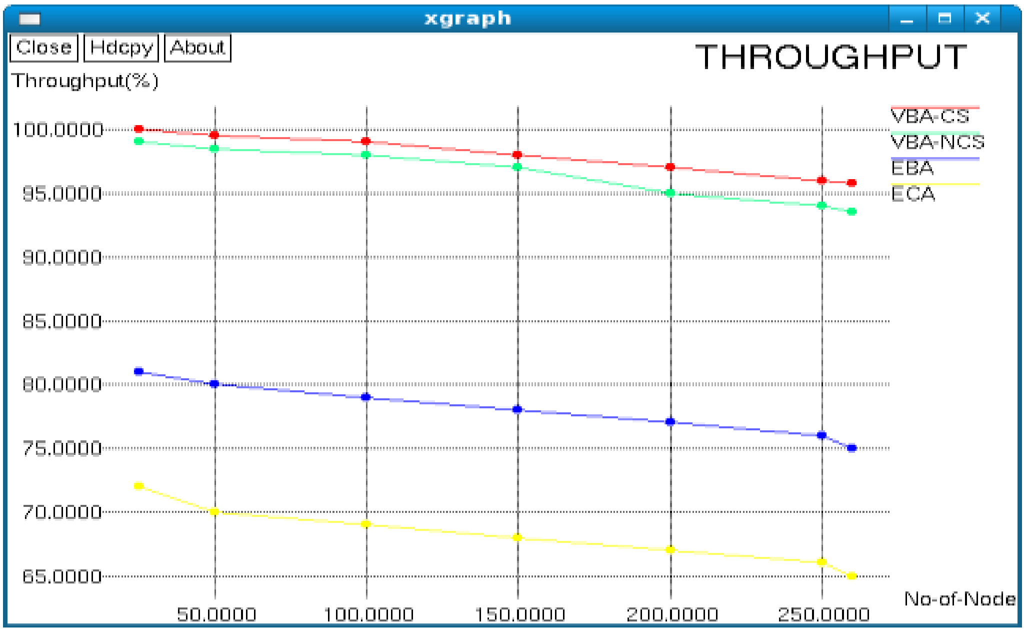

Fig 1: VBA throughput comparison with EBA and ECA

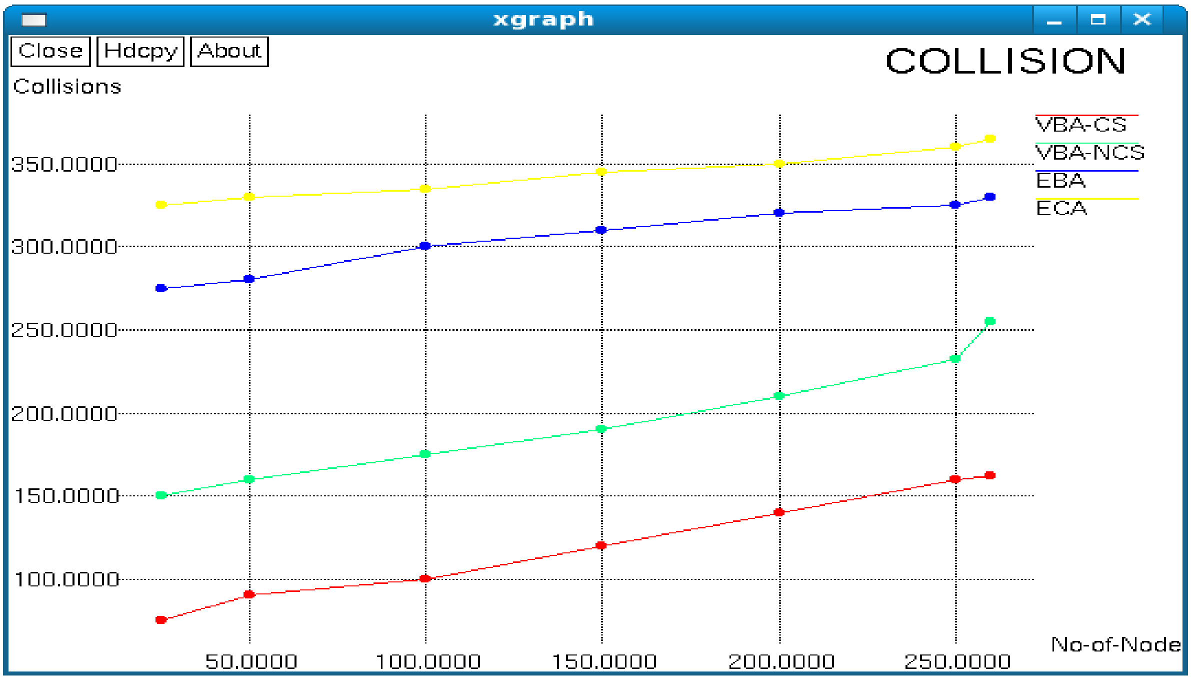

**Fig 2: Comparison of collision**

When more than one node transmits the data at same time there is a possibility for the data packets to collide and packet drop occurs. The amount of collision should be low for efficient wireless network. From the above graph, it is observed that the number of collisions for the VBA-CS and VBA-NCS are less when compared with EBA and ECA. The strength of the VBA method is significantly reducing the number of collision and therefore, increasing the quality of service for wireless networks. The following graph shows the probability of success plot for the VBA method for different values of K. From the graph, it is clear that success probability increases with increase in the value of sequence number (K). Depending on the network traffic this K value can be increased up to 300.

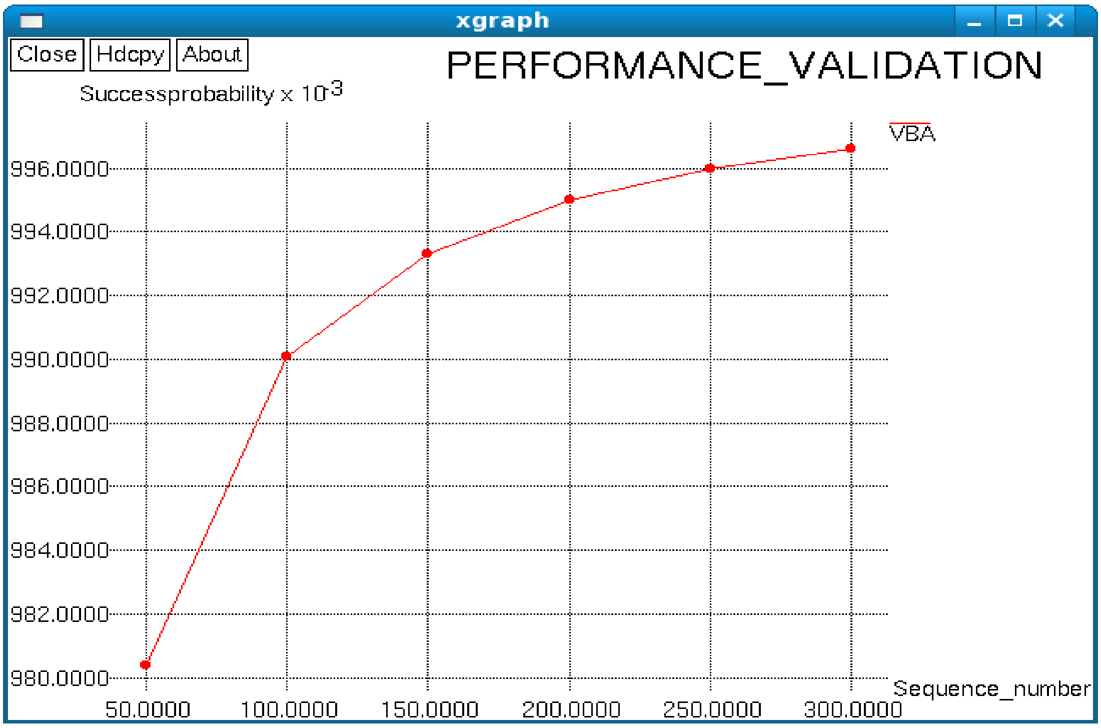

**Fig 3: performance validation of VBA when K increases**

Service time is the time taken by a node to complete its transmission. It used evaluate performance of the network, generally service time should be minimum and constant.

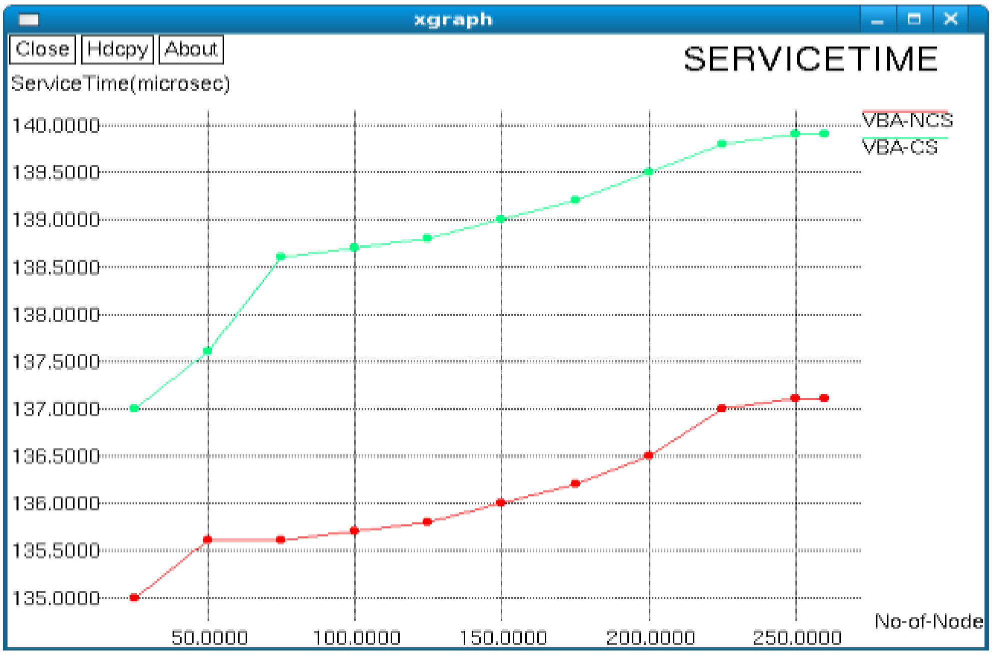

**Fig 4: Service Time comparison**

## CONCLUSION

Optimization of back-off period is done using a method called Virtual Back-off Algorithm (VBA). Two variants of VBA namely VBA with Counter Sharing (VBA-CS) and VBA with No Counter Sharing (VBA-NCS) are compared. Here four parameters namely Throughput, MAC efficiency, Collision, Service time are considered to evaluate the performance of VBA-CS and VBA-NCS. The simulation results show that both the variants of VBA perform well for all the four parameters. But comparatively VBA-CS performs better than VBA-NCS. The basic principle used in this method is limiting the number of transmission of a node based on sequence number. The VBA scheme uses fair distributed mechanisms to access a channel. A counter at each node is introduced to maintain the discipline of the nodes**.** Hence, by optimizing the back-off period it is noticed that a great reduction in the amount of collision and significant increase in throughput. The sequence number used for simulation is constant for all the nodes. In the future work VBA can be made as adaptive, that is, to vary the value of the sequence number for each node according to the data availability of the node.

## REFERENCES

[1]. Choi J, Yoo J, Choi S, and Kim C (2005), “EBA: An enhancement of the IEEE 802.11 DCF via distributed reservation,” IEEE Trans. Mobile Comput., Vol. 4, No. 4, pp. 378–390.

[2]. XiaoY, Li H, Wu K, Leung K K and Ni Q , “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.

[3]. Baldwin R O, Davis N J, and Midkiff S E (1999), “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

[4]. Calì F, Conti M, and Gregori E (2000), “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.

[5]. Krishna P V and Iyengar N (2008), “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

[6]. Fitzek F, Kopsel A, Wolisz A, Krishnam M, and Reisslein M (2002), “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.

[7]. Kleinrock L and Tobagi F A (1975), “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.

[8]. Krishna P V and Iyengar N (2008), “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.

[9]. Mangold S, Choi S, Hiertz G R, Kle O, and Walk B (2003), “Analysis of IEEE 802.11e for QoS support in wireless LANs,” IEEE Wireless Commun.,Vol.10,No.6,pp.40–50.

[10]. Zhu H, Li M, Chlamtac I, and Prabhakaran B (2004), “A survey of quality of service in IEEE 802.11 networks,” IEEE Wireless Commun., Vol. 11, No. 4, pp. 6–14.

[11]. Vaidya N, Gupta S, and Bahl P (2005), “Distributed fair scheduling in wireless LAN,” IEEE Trans. Mobile Comput., Vol. 4, No. 6, pp. 616–628.

[12]. Pattara-Atikom W, KrishnaMurthy P, and Banarjee S (2003), “Distributed mechanisms for quality of service (QoS) in wireless LANs,” IEEE Wireless Commun., Vol. 10, No. 3, pp. 26–34.