

# Improving the performance of 802.11 Wireless LANs with MAC Adaptation

Dorothy A. Rambim<sup>1</sup>, Mjumo Mzyece<sup>2</sup> and Karim Djouani<sup>2</sup>

Department of Electrical Engineering

French South Africa Technical Institute of Technology (F'SATI)

Tshwane University of Technology (TUT), P/Bag X680, Pretoria, 0001, South Africa

Tel: +27 12 382-4809, Fax: +27 12 382-5294

email: [darambim8@gmail.com](mailto:darambim8@gmail.com)<sup>1</sup> { [MzyeceM, DjouaniK](mailto:MzyeceM.DjouaniK@tut.ac.za) }@tut.ac.za<sup>2</sup>

**Abstract**—With the increasing deployment of 802.11-based wireless local area networks (WLANs), the need to optimize the performance of 802.11 for emerging network applications has turned out to be a matter of concern. This paper proposes a Transport Control Protocol (TCP) and Medium Access Control (MAC) interaction strategy which manages the TCP packets. A backoff algorithm is introduced to improve the access conflict in the channel. In the backoff algorithm, the contention window is examined in the event of low and high load traffic and adapted appropriately. The solution tries to optimize the performance of WLAN by analyzing different TCP segments and utilizing MAC-layer parameters to prioritize the packets. We demonstrate that significant improvement in TCP throughput can be achieved when using typical Internet applications.

**Keywords-words;** IEEE 802.11, CROSS-LAYER, WLAN, MAC and TCP

## I. INTRODUCTION

The rapid growth in wireless networks strongly indicates that wireless communication is fast becoming the preferred communication service. Wireless Local Area Networks (WLAN) have made significant advances in home, office and public areas, becoming part of the existing Local Area network (LAN) infrastructure [1]. But the new advances opened up by the wireless technologies are also accompanied by new technical challenges such as limited and varying bandwidth resources among others.

The IEEE 802.11 standard defines Medium Access Control (MAC) and physical (PHY) layers of WLAN. The standard also defines a distributed coordination function (DCF) for sharing access to the medium based on the carrier sense multiple access with collision avoidance (CSMA/CA) protocol.

The use of transport control protocol (TCP) as the transport layer protocol is evidently valuable since the Internet access from wireless stations is dominated by the TCP traffic [2]. However, the characteristics of TCP have not been taken into account for transport over WLANs. When TCP runs over WLANs, the TCP data and the TCP ACK compete for the channel, and moreover, when the number of stations increases, the performance of the networks degrades faster due to the probability of access conflict.

Significant research and standardization effort has been taken on to address the challenges facing wireless networks, with MACs protocol playing an essential role in the success of wireless communication. The MAC protocol defines how each station can share the limited resources in an efficient manner. IEEE 802.11e [3] supports quality of service (QoS) operation that allow some nodes to have higher priorities than others. However, in IEEE 802.11, the network performance can be degraded significantly as the number of contending stations grows, in this case, the MAC protocol fails to achieve a reasonable throughput due to load increase.

The goal of this paper is to study the effects of interaction between the TCP and the MAC layer protocols operating in a wireless network: this includes examining the effects of IEEE 802.11 and IEEE 802.11e MAC protocols on TCP applications.

The remainder of this paper is organized as follows: Section II discusses related work, Section III provides an overview of TCP and IEEE 802.11 MAC. Section IV presents the proposed scheme and numerical results demonstrating the significant performance improvement that can be attained. Finally section V concludes the paper.

## II. RELATED WORK

Much research has been done proposing various mechanisms for improving TCP performance in wireless networks [4] and they can be classified into different categories, according to the concerned layer within the protocol stack as discussed below.

The transport layer approach is based on end-to-end solutions [5, 6]. The end-to-end schemes maintain TCP's original idea of end-to-end semantics between a sender and receiver. Split-Connection solutions [7] and link layer solutions try to make the link layer look similar to the wired case for TCP with the most interesting proposal being the snoop protocol [4]. Most of the above mechanisms and protocols deal only with one layer to improve the performance of the channel and they are not adapted to 802.11 networks or require technical changes in the standard [8].

Recent work began to pay attention to the influence of the data link layer, more so the MAC layer of IEEE 802.11 standards [9]. In [10], it is demonstrated how the use of combining the mechanisms of both TCP and MAC protocols improves the wireless network performance. Generally, IEEE802.11e protocol performs better than IEEE 802.11 [11], but the TCP have not been particularly considered. Backoff based priority and hybrid priority have been reported in the literature for DCF and enhanced distributed coordination function (EDCF) [12]. Most of the above mechanisms utilises User Datagram Protocol (UDP) as the transport protocol and not TCP. Hence, evaluating the performance of TCP in a wireless environment and quantifying the effects of the unique characteristics is still an open problem. The proposed algorithm is to improve the performance under different load rates and to increase the service differentiation in EDCF-TCP based network. The scheme extends the basic EDCF by making it more adaptive while taking into account network load conditions.

### III. OVERVIEW OF TCP AND IEEE 802.11 WLAN

The 802.11 WLAN is commonly used for wireless access to the Internet. On the other hand the Internet is dominated by TCP traffic, however, the TCP characteristics have not been adequately considered in the 802.11 WLAN design.

#### A. Transport Control Protocol

TCP is the most popular transport layer protocol on the Internet. TCP offers a reliable byte stream service, transparent segmentation, reassembling of data and handles flow and congestion control. Some of the TCP traffic transmitted over Wireless networks includes World Wide Web (WWW), e-mail, file-transfer protocol (FTP) and even some multimedia applications [2].

A typical TCP packet includes control information (for connection establishment) and application data. The control bits are set in the protocol header. In TCP, data exchange includes certain control packets related to connection establishment (SYN and ACK), disconnection (RST and FIN), and empty acknowledgements, which prevent unnecessary retransmissions and slow-down the transmission rate. Connection establishment is very important for any connection-oriented communication. TCP uses a three-way handshake to establish a connection with both sides transmitting the SYN and ACK control segment as demonstrated in Figure 1. Due to the length of control segments being smaller than data segment, the control segments attract less errors compared to the data segment in the wireless channel since the frame error rate is related to the packet size [13].

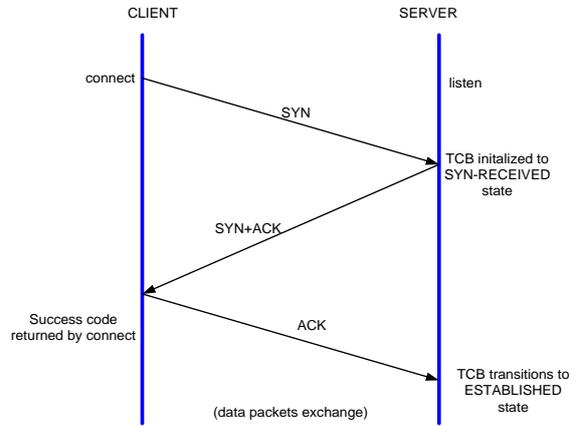


Figure 1: TCP connection establishment

The control packets, while they do not carry application data, plays a very important role in keeping the connection alive and data flows optimal [11].

#### B. IEEE 802.11 DCF

As mentioned in section I, IEEE 802.11 DCF is based on CSMA/CA. In DCF mode, a node with a packet to transmit initializes a backoff timer with a random integer drawn from a uniformly distributed interval  $[0, CW - 1]$ , where  $CW$  is the contention window in terms of time slots.

After a node senses that the channel is idle for an interval called DIFS (Distributed Coordination Function Inter frame Space), the MAC then starts the backoff process by selecting a random back off counter and decreases by one for each idle time slot. When the channel becomes busy due to other nodes' transmissions, the node freezes its backoff timer until the channel is sensed idle for another DIFS. When the backoff timer reaches zero, the node begins to transmit. If the transmission is successful, the receiver sends back an acknowledgment (ACK) after an interval called SIFS (short inter-frame space). Then, the transmitter resets its  $CW$  to  $CW_{min}$ . In case of collisions, if the transmitter fails to receive the ACK from its intended receiver within a specified period, it doubles its  $CW$  up to a maximum value  $CW_{max}$  and the process starts again. When the transmission of a packet fails for a maximum number of retry limit, the packet is dropped.

The  $CW$  size is initially assigned  $CW_{min}$ , and is increased when a transmission fails. After any unsuccessful transmission attempt, another back off is performed using a new  $CW$  value updated by  $2 \cdot (CW + 1) - 1$ , with an upper bound of  $CW_{max}$ .

This reduces the collision probability in case there are multiple stations attempting to access the channel. After each successful transmission, the  $CW$  value is reset to  $CW_{min}$ , and the station that completed the transmission performs DIFS deference and a random back off even if there is no other pending frame in the queue.

Basically, DCF provides channel access with equal probabilities to all stations contending for the channel access in a distributed manner and therefore it does not support the concept of differentiating frames with different priorities.

### C. IEEE 802.11 EDCF and HCF

The DCF provides channel access with equal chances to all stations contending, however, equal access is not desirable among stations with different priority frames [14]. The emerging EDCF provides differentiated, distributed channel accesses for frames with 8 priorities (from 0 to 7) through enhancing the DCF. The EDCF is part of a single coordination function, called the Hybrid Coordination Function (HCF) of the 802.11 MAC. Each frame from the higher layer arrives at the MAC along with a specific priority value. The EDCF adopt eight different access priorities that are further mapped into four access categories (ACs) as shown in Figure 3.

Table I: User priority mapping [15]

| PRIORITY | ACCESS | CATEGORY | DESIGNATION |
|----------|--------|----------|-------------|
| 1        | 0      | (AC_BK)  | Best Effort |
| 2        | 0      | (AC_BK)  | Best Effort |
| 0        | 0      | (AC_BK)  | Best Effort |
| 3        | 1      | (AC_BE)  | Video Probe |
| 4        | 2      | (AC_VI)  | Video       |
| 5        | 2      | (AC_VI)  | Video       |
| 6        | 3      | (AC_VO)  | Voice       |
| 7        | 3      | (AC_VO)  | Voice       |

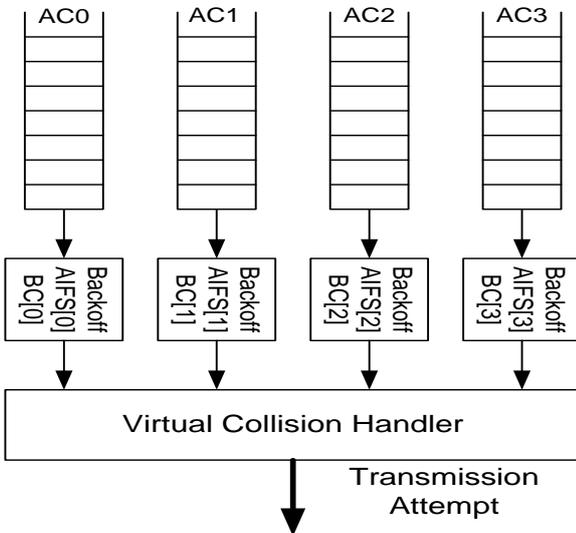


Figure 3: Four access categories (ACs) for EDCF. [15]

Each AC queue works as an independent DCF station and uses its own contention parameters such as Arbitration Inter-Frame Space (AIFS),  $CW_{min}$  and  $CW_{max}$ .

Like a DCF station, each AC starts a backoff counter (BC) after detecting an idle channel for a time interval equal to an AIFS length. The backoff value is chosen to be a number between  $[1, CW + 1]$ ,  $CW$  is set to  $CW_{min}$  initially and increased whenever collision occurs.  $CW$  increases in accordance with the following equation [6]:

$$CW_{new}[AC] = 2 * CW[AC] + 1 \quad (1)$$

Upon successful transmission, the  $CW$  value is reset to  $CW_{min}$ . In EDCF high priority traffic has a smaller AIFS,  $CW_{min}$  and  $CW_{max}$  than low priority traffic. Hence, higher priority traffic entering the contention period and accessing the wireless medium earlier than the lower priority traffic. In [2] it is illustrated that, if  $i$  has a higher priority than  $j$ , then it follows that:

$$\begin{aligned} CW_{min}[i] &< CW_{min}[j], \\ CW_{max}[i] &< CW_{max}[j] \\ AIFS[i] &< AIFS[j] \end{aligned} \quad (2)$$

The problem with EDCF is that the values of contention window and backoff are static and do not take into account the wireless channel conditions.

The static method employed in IEEE 802.11 and 802.11e has been proved to be ineffective in maximizing channel utilization whenever the traffic load increases [16].

The main contribution of this paper is a self controlled adaptive contention window adjustment algorithm (AD\_EDCF). The algorithm dynamically adjusts the contention window to the required point according to the traffic of the network, thus reducing bandwidth inefficiency due to static contention window settings. The algorithm is effective both when there are many and few active stations.

### IV. PROPOSED SCHEME AND OPNET SIMULATION

TCP control packets generally have smaller segments and are more frequent than application data. Therefore, the chances of control packets being lost are greater than that of packets containing only application data [13]. The significance of TCP control packets is not given the consideration it deserves yet before entering the data transfer phase, a connection must be properly established in a multi-step hand shake process. Therefore, there is a need for a method of giving priority to the TCP control packet traffic such that any delays in the delivery of TCP control packets pending delivery of TCP data packets are reduced.

In the MAC layer, an adjustment algorithm is introduced. By utilizing the available MAC layer parameters, two access categories (ACs) are defined to process the control and the data segments of TCP separately. The TCP control segments are identified above the MAC layer using the control field of the TCP segment header. When it is a control segment, the agent put the packet into the queue of AC class 1, which is a high priority class. Otherwise, the segment is put in the queue of class 0.

The AC class 1 clearly has a smaller contention window and AIFS that enable it automatically to obtain high priority service. The data segment of TCP is pushed into the queue of AC class 0 and parameters are set to be the same as the legacy DCF hence, does not influence other package transmission. Arranging the control segments in the queue

of AC class1, which has smaller contention window and AIFS, enables it obtain high priority service. However, the high priority service gains more performance than the lower priority service when the number of access nodes are few [13]. With the increase of load in the network, the ability to obtain network resources with high priority drops due to increase of access service. The result is the access conflict which affects the entire access point (AP) traffic capacity. Therefore, the need to make the proposed scheme adaptable so that the higher priority class updates its  $CW$  in an adaptive way while taking into account the access conflict rate of the channel.

In [16], it is illustrated that by inflexibly increasing the contention window, the following situations may occur: (1) If the contention windows increases slightly, it may result into severe access conflict when channel loading is high and (2) If the contention window grows too much, it may result into many idle periods when channel loading is low. Therefore, the assumption is made that, by adjusting the contention window dynamically according to traffic load, the access conflict is controlled.

The proposed technique increases the transmission probability when the channel is less congested by reducing the contention window and increases the contention window when the channel is more congested. While considering the basic EDCF scheme for wireless network, the  $CW_{min}$  and  $CW_{max}$  values are categorically stated for each priority level with different inter frame space (AIFS) and backoff counter (BC). The number of times the backoff counter BC starting depends on the congestion of the wireless channel; the more access conflict, the more the number of times the backoff counter restarting.

The network traffic condition is determined by defining the counter to represent the number of times the backoff counter restarts of the class 1; transmitting the TCP control segment. The service differentiation between admitted traffic flows is achieved by dynamically controlling channel access parameters according to the following rules:

The average access conflict rate for the backoff counter of higher priority class is calculated as follows:

```

backoff_slot_num >= state (i= 1..... L),
where L is the retry threshold
    if (backoff_slot_num >= retry threshold)
        CWcurr = min(CWmax, W*(avacc_rare+1))
    else if (backoff_slot_num < retry threshold &&
            backoff_slot_num ≠ 0)
        CWcurr = max(CWmin, (W* avacc_rare))
    else CWcurr = CWmin

```

The window increases faster when there is access conflict in the channel and when a station successfully transmits consecutive frames, the  $CW$  is decreased until the original low limit is attained.

#### A. System Model and Simulation

In the network model used for this study the LAN is extended using a WLAN Ethernet router that forms a WLAN together with some mobile hosts (MHs). The model consists of two Servers, and some mobile hosts (FHs). Server and MH clients are connected to the WLAN router through switches using 100baseT link model. We use the 802.11g PHY for our simulations.

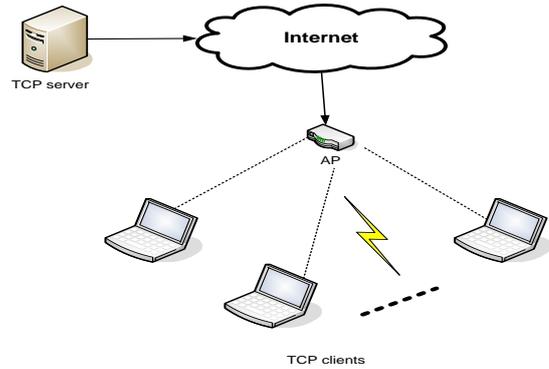


Figure 4: TCP traffic flows over infrastructure WLAN.

The Application and Profile configuration nodes are configured to generate different applications. The CBR application accesses the Constant Bit Rate (CBR) server and it is mainly used to increase chances of access conflict in the wireless link and the simulation is conducted using OPNET version 14.5.

In the simulation, the control field of the TCP segment header is identified by the agent above the MAC layer; which pushes the control segments into the AC class 1 otherwise, into the AC class 0.

Table II shows the EDCF parameters used for each traffic type along with the corresponding priorities and AC values.

Table II: Simulation parameters

| Simulation parameters | 802.11g     |
|-----------------------|-------------|
| $CW_{min}$            | 16          |
| $CW_{max}$            | 1024        |
| Slot time             | 9 $\mu$ s   |
| SIFS Time             | 16 $\mu$ s  |
| PHY overhead          | 20 $\mu$ s  |
| Max Propagation Delay | 0.5 $\mu$ s |
| Data Rate             | 54 Mbps     |
| Basic data rate       | 6           |
| ShortRetryLimit       | 7           |

Each active station attempts to send TCP packets in a greedy manner; one after another. The parameters are kept constant while the number of stations is changed from 2, 4, 8, 16 and 32 in the simulation time of 600 seconds.

The throughput of the TCP traffic is measured under three scenarios: (1) Using the legacy DCF, (2) when EDCF are used and (3) when adaptable prioritization scheme (AD\_EDCF) is used. The measured throughputs are taken in each scenario and compared.

The first model initially consists of an Access Point (AP) connected to two clients. The network load is increased continuously up to a maximum of 32 clients. The throughput is measured after the addition of a number of clients.

We examine the effect of the network load on the TCP throughput and the general packet delay for the legacy DCF, EDCF and AD\_EDCF. The layered diagram for simulation results are as shown in the figure 5, 6 and 7.

To evaluate the performance of the different schemes, the following metrics are used:

- **TCP Delay:** Delay (in seconds) of packets received by the TCP layer in a particular node, for all connections.
- **Throughput:** Total data traffic in bits/sec successfully received and forwarded to the higher layer by the WLAN MAC.
- **End-to-end delay:** This refers to the time it takes a packet to be successfully delivered from traffic source to application layer of the destination.

In figure 5, we simulate using different clients from a minimum of 2 to a maximum of 32 clients for the EDC, EDC and AD\_EDCF.

The results of the simulation show that the throughput decreases as the number of nodes increases. This is appropriate because a higher number of TCP connections are opened up as different nodes try to compete for the medium to send data, thereby, increasing the overall overhead and sharing of the channel.

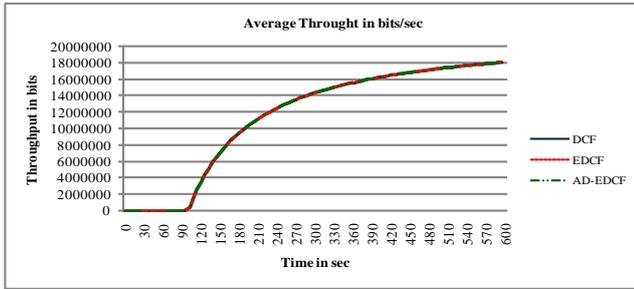


Figure 5(a): Average throughput with 2 stations

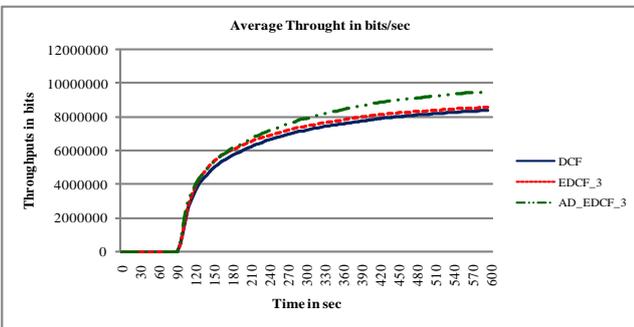


Figure 5(b): Average throughputs with 32 stations

Figure 5(a) and (b) shows average throughput of different schemes with different traffic load.

The throughput for the three schemes is more or less the same when the load is small. This is due to availability of enough network resources for all the traffics. When load is heavy, however, the individual throughput of each scheme becomes clearly different. The reason is that there are more

access conflicts in the network, especially when the wireless network has more access nodes. The throughput comparison when there is light load and heavy load is shown in figure 5(a) and (5c) respectively. It can be seen in figure 5(c) when there is heavy load that substantial throughput improvement can be achieved while using adaptive EDCF.

The AD\_EDCF has a higher throughput since it can decrease the chances of access conflict by adaptively adjusting the contention window. Comparing with the DCF the percentage improvement for two stations is negligible, but, with the increase of 32 stations the percentage is 13.5%. When we compare with EDCF for 32 stations the improvement is 3.6%.

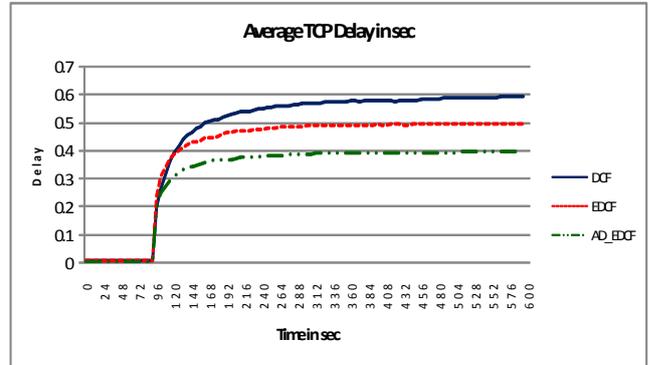


Figure 6(a): average TCP delay with 2 nodes

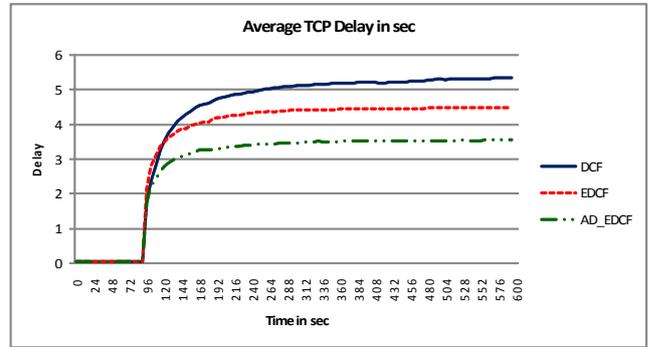


Figure 6(b): average TCP delay with 32 nodes

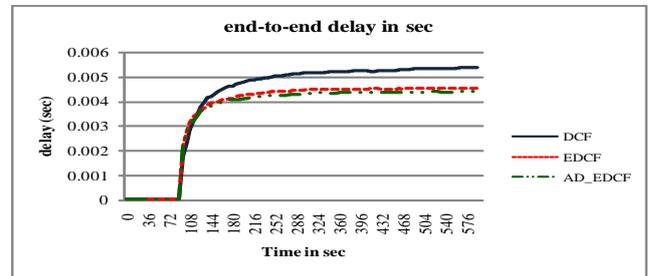


Figure 7(a): average end-to-end delay with 2 nodes

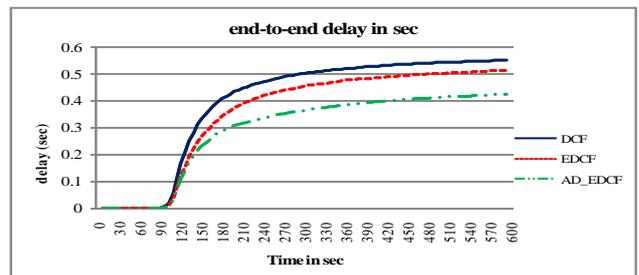


Figure 7(b): average end-to-end delay with 32 nodes

Figures 6 and 7 depict the average TCP delay and average end-to-end delay respectively as the number of stations increases. The average delays for DCF, EDCF and adaptive\_EDCF is shown in figure 6(a), (b) and Figure 7(a) and (b) for various traffic load.

As illustrated in 7(a), we can see that there is a difference in performance measurements. The proposed scheme provide a better performance than the legacy DCF and EDCF as the number of stations are increased,. The DCF and EDCF have a longer delay than the AD\_EDCF even in overloaded scenarios since it tends to select a larger contention window when there is access conflict in the channel. Therefore, it is clearly point out from the simulation that the adaptive algorithm can decrease the average delay and hence improve the performance of the wireless LAN.

## CONCLUSION

This paper discussed the need and scope for performance improvements in TCP throughput while using the IEEE 802.11e MAC layer protocol. The technique of assigning priority to TCP control packet is put forward. The proposed technique influences layer coordination between TCP and MAC to utilize the TCP control packets and improve the efficient usage of wireless network. The work demonstrates that the TCP control segments can be used to improve the throughput on 802.11 WLAN particularly in decreasing the access conflict in the network. Simulation results show that the proposed algorithm outperforms the DCF and the EDCF. It is found that the throughput performance is strongly dependent on the total load on the system

Future works could include adapting other parameters such as CWmax, the maximum number of retransmissions and not only to consider the load but also different packet sizes.

## REFERENCES

- [1] U. Varshney, "The Status and Future of 802.11-Based WLANs," *Computer*, vol. 36, pp. 102-105, June 2003.
- [2] J. Premalatha and P. Balasubramanie, "Simulation Analysis of QoS paramters by combining MAC and TCP in MANETS," *ICGST-CNIR*, vol. 8, January 2009.
- [3] D. He and C. Q. Shen, "Simulation Study of IEEE 802.11e EDCF," 2003.
- [4] H. Balakrishnan, *et al.*, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links (3)," *IEEE/ACM TRANSACTIONS ON NETWORKING*, vol. 5, December 1997.
- [5] M.Mathis, *et al.* ( 1996, TCP Selective Acknowledgment Options (SACK). . *RFC 2018*.
- [6] C. Casetti, *et al.*, "TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks," *In Wireless Networks Journal* vol. 8, pp. 467-479, 2002.
- [7] A. V. Bakre and B. R. Badrinath, "Implementation and Performance Evaluation of Indirect TCP," *IEEE Transactions on Computers*, vol. 46, pp. 260 - 278 March 1997.
- [8] S. Lohier, *et al.*, "Cross-layer loss differentiation algorithms to improve TCP performance in

- WLANs " *Telecommunication Systems*, vol. 36, pp. 61-72, November, 2007.
- [9] IEEE, "802.11-2007 IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," ed, 2007.
- [10] Ghamri\_Doudane and G. Pujolle, "MAC-Layer adaptaion to improve TCP flow performance in 802.11 wireless networks," presented at the Wireless and Mobile Computing, Networking and Communications, 2006.
- [11] W. Lin and J. Wu, "Modified EDCF to improve the performance of IEEE 802.11e WLAN," *Computer Communication*, vol. 30, pp. 841 - 848, 2006.
- [12] Y. Xiao. (2004) IEEE 802.11e: A QoS provisioning at the MAC layer. *IEEE Wireless Communication*. 77-79.
- [13] H. Xie and B. Yi, "A MAC-Layer Adaptaion Algorithm Based on TCP control Segment in Wireless LANs," 2007.
- [14] C. Kuan and N. Yong, "Providing QoS in congested IEEE 802.11 hot spots," in *First International Conference on Distributed Frameworks for Multimedia Applications*, 2005.
- [15] S. Choi, "IEEE 802.11e contention based channel access (EDCF) performance evaluation," 2003.
- [16] L. Romdhani, *et al.*, "A daptive EDCF: enhanced service differentiation for IEEE 802.11 wireless ad-hoc Networks," *IEEE Wireless Communications and Networking*, vol. 2, pp. 16-20, 2003.

**Dorothy Rambim** received her Bachelor's degree in Computer Science from Moi University, Kenya. She is currently pursuing M'Tech degree at TUT, South Africa. Her research interests include Cross layer optimization and QoS in wireless networks.