Enhanced Multimedia and Real-Time Stream Throughput in Wireless Ad-Hoc LAN Using Distributed Virtual Channel Based MAC*

Md. Mustafizur Rahman and Choong Seon Hong[**]

Department of Computer Engineering, Kyung Hee University, South Korea
mustafiz@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. Recent and upcoming CSMA/CA based MAC protocols for wireless LANs offer block-acknowledgement or aggregated frame exchange in order to provide high data rate for multimedia and real-time data, while all of them avoid collisions by exclusive use of wireless medium that block neighboring nodes of both sender and receiver from participating in concurrent transmission; and thus downgrade medium utilization and overall throughput. In this work, we offer to create concurrent virtual channels over same physical channel by synchronizing the transmit/receive switching of senders and receivers and allow parallel data transfer. Each virtual channel is used by a transmitter/receiver pair and all virtual channels within 2-hop network utilize the medium by distributed coordination and avoid inter virtual channel interference. Simulation result shows that proposed scheme removes the obstruction to participate in parallel transmission for some neighboring nodes of sender and receiver and improves the network performance.

1 Introduction

Multimedia data and real time transmission necessitates increasing demand for reduced average latency at nodes and higher network throughput in wireless LAN environment. Wireless LANs use shared medium that requires an efficient channel access function for successful data transmissions. The choice of medium access scheme is difficult in ad-hoc networks due to the time varying network topology and the lack of centralized control [1]. The access problem when same receiver hears transmissions from many nodes has been much studied since the ALOHA, and it bounds on the throughput of successful collision-free transmissions and forces to formulate efficient transmission protocols [2]. Sharing channels in networks does lead to some new problems associated with hidden and exposed terminals. The protocol MACA [3] and its extension MACAW [4], use a series of handshake signals to resolve these problems to a certain extent. This has been standardized and used in DCF/EDCA/ADCF in IEEE 802.11 family of protocols. Such protocols prohibit all hidden and exposed nodes from participating in concurrent transmission (see fig. 1(a)) and hence packets

---

experience latency. Although this scheme supported the wireless LANs for more than a decade, it requires enhancement for increasing demand for higher network throughput (100mbps in 802.11n) for multimedia data. An important approach to improving transfer efficiency is provided with aggregate exchange (AE) sequences for the upcoming IEEE 802.11n that acknowledges multiple MPDUs with a single block acknowledgement (Block ACK) in response to a block acknowledgement request (BAR)[5][6]. This protocol effectively eliminates the contention for every MPDU. If attempting to use the existing MAC protocols without aggregation, a PHY rate of 500 Mbps would be required to achieve the throughput goal of 100 Mbps at the MAC Service Access Point (MAC SAP) [7]. The aggregated exchange (or the block ACK in 802.11e) with conventional RTS/CTS mechanism, however, increases latency for data/multimedia packets at neighboring nodes, because of prolonged delay before channel access.
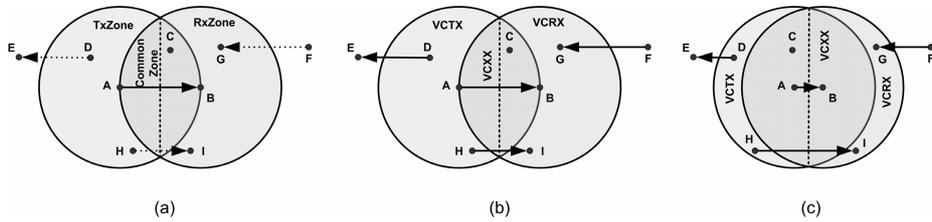
Fig. 1. (a) Conventional Channel reservation where transfer between A and B is possible only; (b) Proposed Virtual Channel Based transfer allowing multiple concurrent transmission; and (c) Virtual Channel scheme when A and B is closer to each other than (b)

In this work, we try to enhance the 802 family of MAC protocols toward distributed spatial reuse of wireless medium for the aggregated exchange or block ACK schemes so that multimedia and real-time data can be transferred in parallel. Our proposed protocol creates separate virtual channel (VC) for every sender-receiver pair and multiple VCs can coexist within same physical channel in 2-hop networks; and, packets can be transferred over these virtual channels concurrently reducing the average latency of the network and increased network throughput.

This rest of this paper is organized as: related works has been described in section 2; section 3 introduces the virtual channel concept and describes how they can coexist without interfering each-other, section 4 describes the mechanism for VC-MAC. Section 5 gives an analytic model for VC delay overheads and discrete system simulation and numerical results obtained from the simulation are presented and described in section 6. Finally, section 7 concludes the paper and contains scope for future work.

2   Related Works

Numerous works supporting concurrent transmissions have been done to improve the capacity in 2-hop network. Sigh et al. [8] proposed a distributed spatial reuse (DSR) MAC protocol for IEEE 802.11 ad hoc wireless LANs to increase bandwidth utilization and reduce power consumption using power controlled transmission. The power

controlled schemes, however, depend on enhancement in the physical layer. In addition to this, finding the maximum independent set (MIS) from the Interference Graph (IG), obtained from IIM signals, is an NP-complete problem and it adds complexity at the MAC layer.

Acharya et al. [9] presents the initial design and performance study of MACA-P, a RTS/CTS based MAC protocol, that enables simultaneous transmissions in multi-hop ad-hoc wireless networks. Like 802.11, MACA-P uses a contention-based reservation phase prior to transmissions. The data transmission is delayed by a control interval, which allows multiple sender-receiver pairs to synchronize their data transfers, thereby avoiding collisions. The enhanced version of MACA-P [10] that they have devised with adaptive learning, avoids the collisions caused by MACA-P's attempts at concurrent transmissions. The control gap in the proposed MACA-P protocol uses for every packet transfer, and hence it increases the overhead.

Rozovsky et al. [11] proposed another protocol for media access control in ad hoc networks that seeks to avoid collisions without making explicit reservations for each and every packet. They used a random schedule driven by a pseudo-random number generator and the nodes publish their schedules to all hidden as well as exposed nodes. This allows each node to opportunistically choose transmission slots. But the protocol would work better for the hardware where synchronized slot is available.

Directional antenna based several solutions have been proposed [12, 13]. However, the hardware cost for using directional antennas to achieve spatial reuse is too high.

Our scheme is proposed as an enhancement of the MAC layer protocols that can be implemented over existing physical layers with better throughput for multimedia and real-time data streams. The complexity in synchronization between nodes is small compared to other proposed protocols.

3   The VC-MAC

The major confronting element in supporting parallel use of the channel in the wireless MAC protocols is to avoid interference at the receiver. We offer transferring data over half-duplex virtual channels between each sender-receiver pair, to support parallel transfer. Before sending data, the sender and the receiver (hereinafter referred as Originator and Recipient) establish the virtual channel and then transfers data. Concurrent virtual channels within 2-hop is allowed when they are synchronized with each other. Each virtual channel is closed after completion of data transfer.

3.1   Virtual Channel Concept

A virtual channel is a series of alternating Data and Control windows (DW and CW) providing a half-duplex connection between sender and receiver in order to transfer blocks of data packets (referred as data frames) using aggregated frame or block ACK

Data	CW	Data	CW	Data	CW

Fig. 2. The construct of a virtual channel

scheme. The originator transmits data frames one after one during the data window; and the recipient acknowledges received packets during the control windows. Synchronized concurrent virtual channels in neighboring nodes allow non-interfering data transfers within 2-hop network.
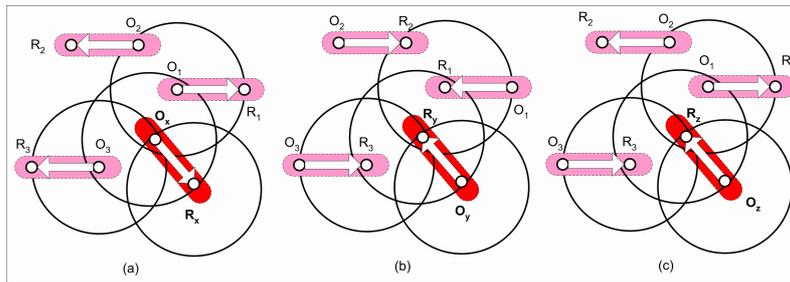
3.2 Concurrent Virtual Channels

In order to introduce spatial reuse of wireless medium, we allow concurrent virtual channels in several cases. To determine whether two nodes can establish a virtual channel, we introduce a new variable *vcState* at each node. The *vcState* indicates the existence and type of virtual channels in neighboring nodes. The *vcState* values and corresponding meaning are given in the following table:

Table 1. Values of *vcState* Variable

Value	Description
VCNO	No VC exist within node's neighborhood
VCTX	One or more in-sync (with common control window) VC Originator(s) exist within its neighborhood
VCRX	One or more in-sync (with common control window) VC Recipient exist within its neighborhood
VCXX	More than one of different type (RX/TX) or out of sync Virtual Channels exist within neighborhood

In-Sync VCs: If more than one same type of VC participants (originator or recipient) exist within the neighborhood of a node, where the data windows in each VC start simultaneously, then all such VCs are said to be in-sync with each other. Trivially, single VC within neighborhood of any node is said to be in-sync. In the following figure 3, neighbors O_1 and O_3 of node O_x are originators of two VCs. If the data and control windows of these two VCs start simultaneously, then these two VCs will be in-sync VCs.

Fig. 3. Concurrent Virtual Channel origination/acceptance cases: (a) when neighbor is an originator, (b) when neighbor is a recipient, and (c) when a neighbor is an originator and another neighbor is a recipient

When a node finds its *vcState* value either VCNO or VCTX and it has data stream to send, it can originate the VC creation process; and the recipient can accept a VC formation request depending on its *vcState* value. Nodes having VCNO value can accept any request whereas VCRX valued nodes can accept VC request if the new VC can be in-sync with the other VC at recipient's neighborhood. In Fig. 3(a) and 3(b) O_1 and O_3 originated VCs because they didn't have any neighboring node participating in any other VC. O_2 finds a neighbor (with VCTX state) with one VC in fig 3(a) so it can originate and aligned (in-sync) with that virtual channel whereas O_x in the same figure finds two VCs within its neighborhood and may be with different data and control window timing (out of sync), so it can have either VCXX or VCTX (barely). In case of VCXX, O_x cannot originate a VC establishment. But if the order of origination is reverse (O_x creates VC earlier than O_3) then both of them would have VCTX in *vcState* and concurrent virtual channels could be established. Similarly, if the *vcState* value at R_y in figure 3(b) is VCRX then it can accept the VC establishment request from O_y. In the figure 3(c), the R_z node has two neighbors with VCs but one is originator and the other is recipient. So, it is impossible to originate or accept request for new VC establishment. The *vcState* value for such nodes becomes VCXX when VC participations found heterogeneous within neighborhood. This is also true for common neighbors of the originator and recipient of same VC.

The formation of virtual channels is completely distributed; local situations in each 2-hop network would allow/deny the concurrent virtual channels. The local VC creation order takes vital role in VC chaining.

4 The Mechanism and Operational Strategies

As mentioned earlier, a VC is a point to point half-duplex channel where the originator and recipient synchronize themselves by predefined time periods. The entire data transfer over VCs is a 3-phase process: Virtual Channel Opening, Transfer, and Closing down. Five new control frames (VCREQ, VCRES, VCCFM, VCDEL, and VCDCK response) are offered to handle the virtual channel establishment and closing phases. Data transfer can be done using existing BAReq and BA frames in IEEE 802.11e MAC protocol or aggregated frame scheme offered for proposed 802.11n.
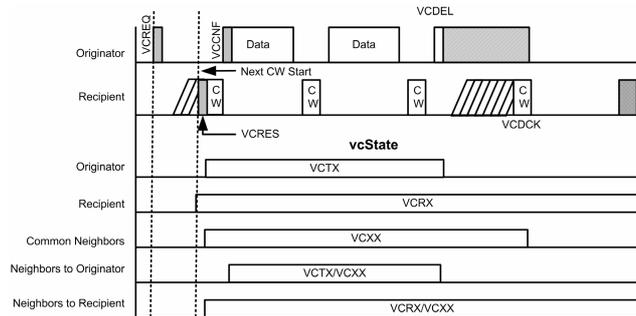

Fig. 4. Virtual Channel Signals and vcState values

4.1   Virtual Channel Opening

The originator, upon obtaining the transmission opportunity (TXOP), initiates the virtual channel opening phase by sending a *VCREQ (*Vritual Channel Request) frame to the intended receiver (recipient). The *VCREQ* frame contains data stream size, proposed Data Window size and *vcState* of the originator. According to the conditions described in section 3.2, the recipient determines whether it is able to accept the request or not. The recipient replies with *VCRES* (VC Request Response) only when an in-sync virtual channel formation is possible. The *VCRES* frame contains data window size, next control window start time and receive-buffer size. Recipients having VCNO state starts the control window immediately, and the *VCRES* frame is sent to the originator in the control window. On the other hand, the recipients with VCRX state wait for next control window for other VC(s) within its neighborhood and sends *VCRES.* The *VCRES* frame from VCRX recipients contains modified data-window size, next control-window time to make the VC in-sync with other VCs. Neighbors of the recipient update their *vcState* value with VCRX upon receipt of the VCRES frame; and common neighbors of both originator and recipient turn into VCXX state.

The originator then waits for the next data window and then sends the VC confirm frame (VCCFM) followed by first block of the data stream. The VCCFM is used to inform neighbors of the originator about new virtual channel. Neighbors of the originator update the *vcState* value to VCTX or VCXX and can participate in data transfer by forming another synchronized virtual channel only.

4.2   Data Transfer

After virtual channel is established, the originator sends data stream in a series of blocks separated by control windows during the data windows. Each burst of data block contains several data frames interleaved by shortest inter frame space period (SIFS in IEEE 802.11). Recent developments in physical layer inspired researchers to adapt block acknowledgement in MAC layer and recently IEEE Computer Society released the IEEE 802.11e MAC protocol supporting block acknowledgement [1]. In VC-MAC we follow the immediate policy of block acknowledgement scheme in the 802.11e standard.

Before sending the first data frame in the first data window of a virtual channel, the originator transmits the *VC confirmation* (VCCFM) frame first then starts the data burst. This VCCFM is sent to inform neighbors of the originator that are out of range from the recipient about VC establishment to update their *vcState*. This is required to synchronize future virtual channels.

4.3   Closing Down

When there is no more data left to send at the originator and the final block acknowledgement has been received by the originator over any of the existing virtual channel; the virtual channel closing process is initiated by the originator by sending a *virtual channel delete* (VCDEL) frame to the recipient. In addition to this, if any of the stations moved beyond the range of the other, then the receiver initiates the closing operation. The VCDEL is acknowledged by VCDCK by the recipient during the next control window.

4.4   Joining EDCA

After closing a virtual channel down, the originator or recipient may contend to get access to the medium and start sending data packets in presence of other virtual channels resulting in error in received data at neighboring recipients with virtual channels. In order to eliminate such situations, contending the medium for data transmission after closing virtual channel is delayed for next data/control windows and the delaying continues until it receives VCDEL or VCDCK frame from other originator or receiver.

In such cases, each participant in virtual channels or the neighbors of such participants takes different actions. The originator senses the medium just after issuing the VCDEL and if it finds any transmission, its virtual channel is closed but it continues the VCTX state. However, if it does not receive any signal, it turns into the VCNO state. This is same for other neighbors in the originator side only. In the recipient side, as soon as the recipient gets the control window, it closes the virtual channel but continues the VCRX state for next control windows until it receives the VCDCK frame from the last recipient. If it gets any corrupt data during next control windows; it is inferred that there exists other virtual channels within its neighborhood and it continues with VCRX state. Neighbors of both originator and recipient turns into VCNO state after it receives the VCDSK from last recipient (if any).

```
Procedure JoinEDCA(var vcState)
Var data;
Begin
   If( vcState = VCTX)
       while not DataWindow do no-op;
       data := ReceiveFrame();
       if data = notCorrupt and
        (data=noReceivedData or data=VCDEL) then
         vcState := VCNO;
       else
         Call JoinEDCA (vcState);
Else if (vcState = VCRX)
       while not ControlWindow do no-op;
       data := ReceiveFrame();
       if data = notCorrupt and
        (data=noReceivedData or data=VCDCK) then
         vcState := VCNO;
       else
         Call JoinEDCA (vcState);
Else if (vcState = VCXX)
       data := ReceiveFrame();
       if data = notCorrupt and
        (data=noReceivedData or data=VCDEL) then
         vcState := VCRX;
         Call JoinEDCA(vcState);
       Else if data = notCorrupt and data=VCDCK) then
         vcState := VCTX;
         Call JoinEDCA(vcState);
End;
```

Fig. 5. Algorithm for the receivers to join the EDCA contention after closing the virtual channel

5 Analysis

5.1 Assumptions

We consider a 2-hop wireless ad-hoc network having stations negligible propagation delay. Influence of packet retransmissions is out of scope for this work as we select errorless packet delivery by the physical layer. In lossy channels, the analysis becomes more complex which should be investigated in the future work. The multimedia stream (data) generation pattern follows the Poisson distribution functions at any station and stream (data) sizes are uniformly distributed.

5.2 Overhead Modeling

Channel Establishment Cost: Time required for the setup phase for a VC is:

$$T_{VCsetup} = T_{VCREQ} + t_{remainDW} + T_{CW} \tag{1}$$

where, T_{VCREQ} is the time required to transmit *VCREQ* Frame, $T_{remainDW}$ is the remaining time of the current Data Window after receiving the *VCREQ*, T_{CW} is the Control Window period.

The $T_{remainDW}$ is random and depends on the arrival time of the *VCREQ* frame in current data window. The value of $T_{remainDW}$ is between 0 and data window period. Hence, the total overhead for the setup phase compared to 802.11e Block Acknowledgement scheme is:

$$\tau_{setup} \approx t_{remainDW} + T_{CW} \tag{2}$$

Data Transfer Costs: For a data stream requiring n data bursts consisting of s_i ($i | i \in 1 \cdots n$) data frames in the *i*-th data burst, the required time in slots over a virtual channel is-

$$T_{tr} = \sum_{i=1}^{n} \left(s_i \times T_{Data} + t_{remainDWi} + T_{CW} \right) \tag{3}$$

where, $t_{remainDWi}$ is the remaining time in the data window for the *i*-th data-burst, and T_{Data} is the average data-frame transmit time.

From the equation for transferring data frames, overhead in time for the latency experienced due to delayed responses from the receiver compared to the immediate block-acknowledgement mode becomes:

$$\tau_{transfer} = \sum_{i=1}^{n} \left(T_{remainDW_i} + T_{CW} \right) \tag{4}$$

For simplicity, we consider the average remaining time in each data window as $T_{remainDW}$, and the total overhead for transferring data of single data stream over each virtual channel becomes,

$$\tau_{transfer} = nT_{remainDW} + nT_{CW} \tag{5}$$

Therefore, the overhead can be reduced if and only if we can limit the control window size as minimum as possible and if each data stream can maximize the utilization of the data window. But the control window size cannot be minimized below the time slots required to send Block Acknowledgements with highest frame size.

For m concurrent virtual channels sharing common Control Window; the average overhead becomes

$$\tau_{transfer} = \frac{n}{m}(T_{remainDW_i} + T_{CW})$$ (6)

Channel Closing Cost: Required time to close a virtual channel is calculated from:

$$T_{clo\sin g} = T_{VCDEL} + T_{remainDW} + T_{CW}$$ (7)

where, T_{VCDEL} is the transmitting time for the VCDEL request.

The total time required to close the virtual channel and join the EDCA contention is:

$$T_{Join} = T_{clo\sin g} + (n-1)(T_{DW} + T_{CW})$$ (8)

where n is the number of concurrent virtual channels before closing the channel. Therefore, the overhead due to closing of the virtual channel is:

$$\tau_{join} = T_{remainDW} + T_{CW} + m \times (n-1)(T_{DW} + T_{CW})$$ (9)

Total time required to send n data streams over n concurrent virtual channel is given by,

$$T_n^{vc} = n \times T_{setup} + T_{transfer} + T_{join}$$ (10)

6 Simulation

The VC-MAC protocol has been simulated on x, y grid topology; from 2×2 to 5×6 grids and compared with recent 802.11e block acknowledgement (BA) scheme. The channel is assumed to be reliable enough to transfer data without retransmission as discussed in 5.1. Our simulation considers parameters as follows. The data rate at physical layer is 54Mbps and in the MAC layer IEE802.11e EDCA and Block Acknowledgement (BA) scheme is used without prioritization. Data-frame size was fixed to 5K bytes. The entire DW-CW cycle in a virtual channel is 3000µs, where 2965µs allocated to data window and the rest 35µs for control window. The control window duration is sufficient to transmit VCRES, BA and VCDCK signals. Multimedia data stream sizes vary from 5K bytes to 250 K bytes and their arrivals at nodes follow Poisson distribution.

The rate of added latency for each virtual channel due to delay in initial response and the control window intervals compared to block acknowledgement BA scheme in 802.11e is depicted in Fig. 6(a). Simulation result shows that the added latency (mean

delay) in each virtual channel decreases as the multimedia stream size increases for fixed data window duration. Latency in VCs reduces with increased stream size because of smaller influence of $t_{remainDW}$ over total turnaround time. For streams of 100K bytes, the added latency is approximately 10% and less than 30% for the first and consecutive VCs in each 2-hop network respectively. Corresponding reduction in data rate at each virtual channel is presented in the Fig 6(b). Virtual channel capacity increases with increased stream size, and approaches to the physical channel capacity. The patterns in both Fig. 6(a) and 6(b) follow the analytical approach as described in section 5.2.

(a)   (b)

(c)   (d)
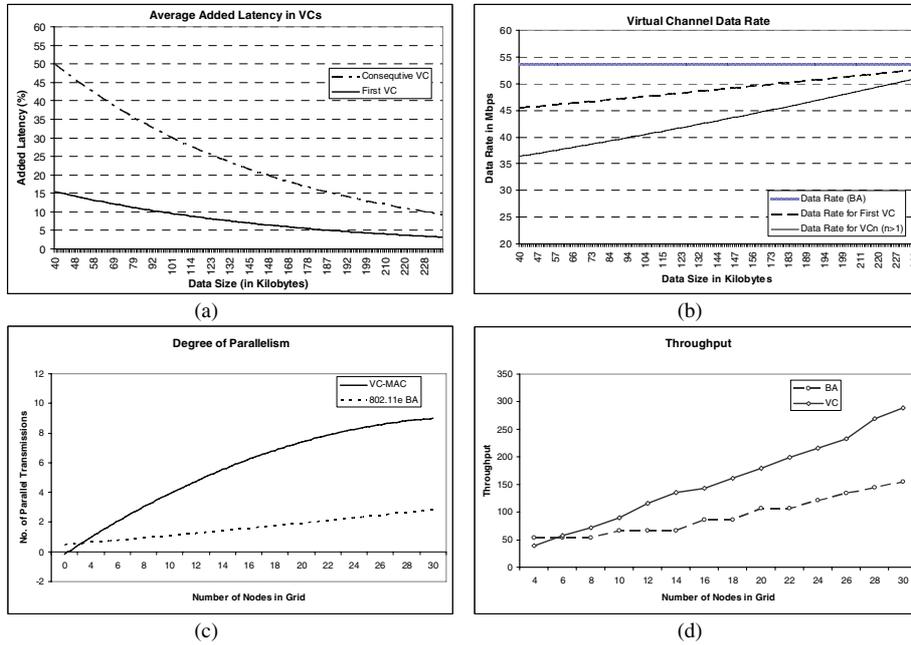
Fig. 6. Simulation Results: (a) Added latency in VCs, (b) Data rate in VCs, (c) Observed degree of parallelism in Grids, and, (d) Network throughput

The curve in Fig. 6(c) shows the average degree of parallelism obtained for the grid networks. The number of parallel VCs increases at higher rate than that of IEEE 802.11e BA scheme with increased number of nodes in the grid. Fig. 6(d) shows the influence in throughput for the degree of parallelism obtained. Control window durations in VCs reduces the network throughput due to reduction in channel capacity, however, overall throughput of the network increases when the network can create concurrent virtual channels within 2-hops. The proposed VC-MAC offers better throughput because it allows hidden and exposed terminals to participate in parallel transmissions, whereas conventional RTS/CTS bases MAC protocols cannot.

7 Conclusion

This work investigates the bottleneck in improving medium utilization for multimedia streams in WLAN environment. It is identified that conventional CSMA/CA MAC, that uses RTS/CTS handshaking, forbids all hidden and exposed nodes to participate in parallel transmissions. Denial of such parallel transmissions in these protocols, imposed due to transmit/receive switching of the nodes during data transmissions, increases latency for multimedia streams. Our investigation identifies several cases where parallel transmissions of multimedia streams are possible; and proposes the virtual-channel based MAC (VC-MAC) to decrease the packet latency and improve overall network performance. The VC is a schedule of synchronized transmit/receive switching of the participating nodes that allows hidden or exposed nodes to initiate parallel transmissions. The synchronization technique avoids collisions at receiving nodes.

Proposed VC-MAC scheme is analyzed and simulated on grid topology, and then results are compared with the Block Acknowledgement scheme of IEEE 802.11e standard for large data or multimedia streams. Analysis and simulation shows that VC-MAC can successfully avoid interference at the receivers and can allow hidden and exposed terminals to create parallel virtual channels. This scheme improves the overall network throughput although it reduces the data rate of each connection pair (virtual channel).

In this work, the evaluation of VC-MAC is done in an error-free environment. The value or otherwise of a MAC tends to show when it is subjected to errors. Impact of retransmissions on network capacity or throughput and finding the optimum data window size and incorporating service differentiation for QoS are open issues for future work.

References

[1] S. Ramanathan, and M. Steenstrup. A survey of routing techniques for mobile communication networks. Mobile Networks and Applications, Vol. 1, no 2, pp. 89-104, 1996.

[2] R. Gallager. A perspective on multi-access channels. IEEE Trans. Infrom. Theory (Special issue on Random Access Communications), Vol. IT-31, Mar. 1985.

[3] P. Karn. MACA: A new channel access method for packet radio. In proceedings of the 9th Computer Networking Conference, pp. 134-140, 1990.

[4] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: A media access protocol for wireless LANs. In Proceedings of the ACM SIGCOMM'94, pages 212–215, London, UK, Aug. 1994.

[5] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications (Part 11), Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, Std. 802.11e, Nov. 2005.

[6] Data Link Frame Aggregation Protocol, J. Michael Meehan, Philip A. Nelson, David Engeset and Athos Pashiardis, Information: An International Interdisciplinary Journal, Volume 7 Number 1, pp. 59-68, January 2004.

[7] J. M. Wilson. The Next Generation of Wireless LAN Emerges with 802.11n, Intel Magazine, URL http://www.deviceforge.com/articles/AT5096801417.html, 2003.

[8] K. P. Sigh, C. M. Chou, M. Y. Lu, and, S. M. Chen. A Distributed Spatial Reuse (DSR) MAC Protocol for IEEE 802.11 Ad-Hoc Wireless LANs. In the proceedings of the 10th IEEE symposium on Computers and Communications, ISCC 2005, pages 658-663, 2005.

[9] A. Acharya, A. Misra, and S. Bansal. MACA-P: A MAC for concurrent transmissions in multi-hop wireless network. In proceedings of the IEEE International Conference on Pervasive Computing and Communications, pages 505-508, 2003.

[10] A. Acharya, A. Misra, and S. Bansal. Design and Analysis of a Cooperative Medium Access Scheme for Wireless Mesh Networks. In proceedings of the First International Conference on Broadband Networks (BROADNETS'04), pages 321-631, 2004.

[11] R. Rozovsky, and P. R. Kumar. SEEDEX: A MAC protocol for ad hoc networks. In proceedings of ACM Mobihoc, 2001.

[12] R. R. Chowdhury, X. Yang, N. H. Vaidya, and, R. Ramanathan. Using directional antennas for medium access control in ad hoc networks. In Proceedings of the ACM international Conference on Mobile Computing and Networking, pages 59-70, 2002.

[13] Y. Wang, and J. J. Garcia-Luna-Aceves. Spatial reuse and collision avoidance in ad hoc networks with directional antennas. In proceedings of the IEEE Global Telecommunications Conference, volume 1, pages 112-116, 2002.