International Journal of Research in Computer Science
eISSN 2249-8265 Volume 2 Issue 1 (2011) pp. 11-19

www.ijorcs.org

EFFICIENT REAL-TIME VIDEO TRANSMISSION IN WIRELESS MESH NETWORK

Rajanikanth Aluvalu[1], NitinBirari[2], Supriya Byreddy[3]

**Dept of Computer Engineering, School of Engineering, R.K.University*

Abstract: We are experiencing rapid improvement in video transmission and picture quality. The technologies are supporting us with real-time video transmissions such as video conferences and webinars. Soon reliable, simple, low cost real-time video will become essential, providing its extensive support to mobiles; PDA's etc. These enhancements are going to affect the consumer behaviour, business culture. Video transmission includes transmitting packets through Wireless Mesh Network (WMN), which turned out to be a challenging scenario for real-time video services. Due to the existing nature of wireless networks include the presence of unpredictable delays and high packet error rates due to the error-prone nature of the wireless links. In this paper, we will propose a model which provides high capacity data transfer with reduced delay. We have simulated and achieved results with higher through put.

Keywords: *Wireless mesh network, Real time video, Contention window, CSMA, IEEE 802.11*

I. INTRODUCTION

A. Wireless Mesh Network

If we have n nodes in a network, where the term "node" refers to a communications device that can transport data from one of its interfaces to another, then the ability of each node to communicate with every other node in the network represents a mesh network topology[1]. We can view the structure of a mesh network by simplifying the number of nodes in the network from a value of n, which is what mathematicians like to work with, to an easy-to-visualize number, such as three, four, or five.

Wireless Mesh Network (WMN) is a promising wireless technology for several emerging and commercially interesting applications, e.g., broadband home networking, community and neighborhood networks, coordinated network management, intelligent transportation systems. It is gaining significant attention as a possible way for Internet service providers (ISPs) and other end-users to establish robust and reliable wireless broadband service access at a reasonable cost. WMNs consist of mesh routers and mesh clients[3].

In this architecture, while static mesh routers form the wireless backbone, mesh clients access the network through mesh routers as well as directly meshing with each other.

Different from traditional wireless networks, WMN is dynamically self-organized and self configured. In other words, the nodes in the mesh network automatically establish and maintain network connectivity. This feature brings many advantages for the end-users, such as low up-front cost, easy network maintenance, robustness, and reliable service coverage. In addition, with the use of advanced radio technologies, e.g., multiple radio interfaces and smart antennas, network capacity in WMNs is increased significantly. Moreover, the gateway and bridge functionalities in mesh routers enable the integration of wireless mesh networks with various existing wireless networks, such as wireless sensor networks, wireless-fidelity (Wi-Fi), and WiMAX[11]. Consequently, through an integrated wireless mesh network, the end-users can take the advantage of multiple wireless networks.

1. Benefits and Characteristics of Wireless Mesh Networks

In WMNs, the wireless mesh routers provide redundant paths between the sender and the receiver of the wireless connection. This eliminates single point failures and potential bottleneck links, resulting in significantly increased communications reliability. Network robustness against potential problems, e.g., node failures, and path failures due to RF interferences or obstacles, can also be ensured by the existence of multiple possible alternative routes[3]. Therefore, by utilizing WMN technology, the network can operate reliably over an extended period of time, even in the presence of a network element failure or network congestion. Recently, the main effort to provide wireless connection to the end-users is through the deployment of 802.11 based Wi-Fi Access Points (APs). To assure almost full coverage in a metro scale area, it is required to deploy a large number of access points because of the limited transmission range of the

APs[3]. The drawback of this solution is highly expensive infrastructure costs, since an expensive cabled connection to the wired Internet backbone is necessary for each AP. On the other hand, constructing a wireless mesh network decreases the infrastructure costs, since the mesh network requires only a few points of connection to the wired network. Hence, WMNs can enable rapid implementation and possible modifications of the network at a reasonable cost, which is extremely important in today's competitive market place. Currently, the data rates of wireless local-area networks (WLANs) have been increased, e.g., 54 Mbps for 802.11a and 802.11g, by utilizing spectrally efficient modulation schemes [5]. Although the data rates of WLANs are increasing, for a specific transmission power, the coverage and connectivity of WLANs decrease as the end-user becomes further from the access point. Multi-hop and multi-channel communications among mesh routers and long transmission range of WiMAX towers deployed in WMNs can enable long-distance communication without any significant performance degradation. Wireless mesh networks are dynamically self-organized and self-configured. In other words, the mesh clients and routers automatically establish and maintain network connectivity, which enables seamless multi-hop interconnection service [3]. For example, when new nodes are added into the network, these nodes utilize their meshing functionalities to discover all possible routers and determine the optimal paths to the wired Internet. Furthermore, the existing mesh routers reorganize the network considering the newly available routes and hence; the network can be easily expanded. Mesh routers are resource-rich nodes equipped with high processing and memory capabilities, while mesh clients have limited memory and computational power.

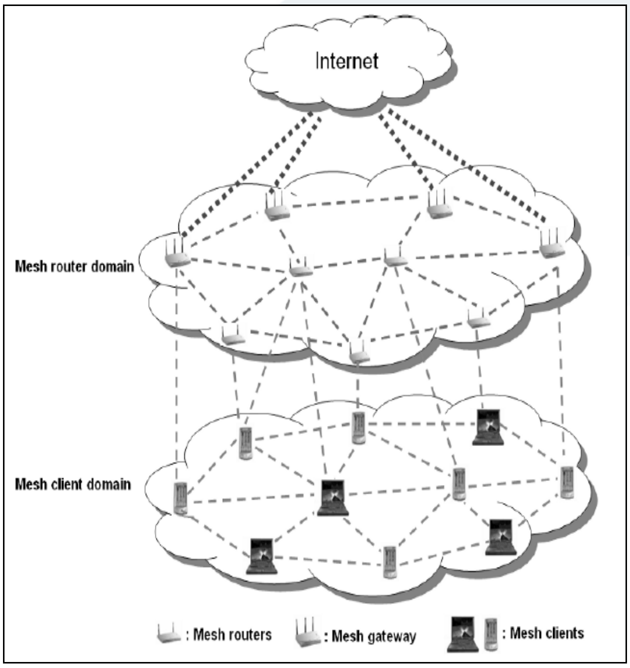

Fig.1: Wireless mesh network architecture

2. Classification of multihop wireless networks:

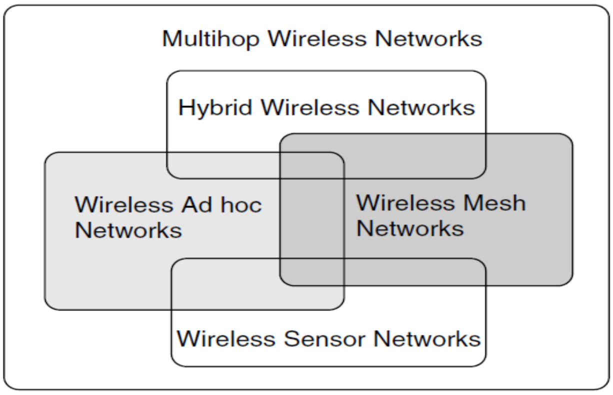

Fig. 2: Classification of multihop wireless networks

Fig. 2 shows the classification of multihop wireless networks; these constitute the category of wireless networks that primarily use multihop wireless relaying. The major categories in the multihop wireless networks are the ad hoc wireless networks, WMNs, wireless sensor networks, and hybrid wireless networks. This book mainly focuses on WMNs. Ad hoc wireless networks [2] mainly lack in infrastructure having highly dynamic topology. Wireless sensor networks, formed by tiny sensor nodes that can gather physical parameters and transmit to a central monitoring node, can use either single-hop wireless communication or a multihop wireless relaying. Hybrid wireless networks utilize [3] both single- and multihop communications simultaneously within the traditionally single-hop wireless networks such as cellular networks and wireless in local loops (WiLL). WMNs use multihop wireless relaying over a partial mesh topology for its communication. The primary differences between these two types of networks are mobility of nodes and network topology. Wireless ad hoc networks are high mobility networks where the network topology changes dynamically. On the other hand, WMNs do have a relatively static network with most relay nodes fixed. Therefore, the network mobility of WMNs is very low in comparison with wireless ad hoc networks.

B. Real time Video

Real time video at sender's side is video that is being transmitted live and at receiver's side, is being watched live. Video technologies are improving dramatically and rapidly, supporting mobile and ubiquitous real-time video experiences [8]. Low cost, simple platforms for real-time video will become an essential part of the way we communicate with each other, and will spawn the next generation of consumer behavior, business practice, media culture and economics, and innovation policy.

Today, many devices are able to conduct live video communications, and many more are in the pipeline transforming the design, implementation, and use of those devices. Simultaneously, as real-time video communications become part of our daily lives and our suite of business tools, we are seeing the beginnings of persistent conversations across contexts tied more to the user than the devices being used. As we move through office, car, home, and elsewhere, our devices will be coordinated and linked to maintaining the continuity of our communication events. In other words, I could start a conference call on my office computer, shift the call to my mobile device for my commute home, and finish the call on my home computer or web-enabled television.

I. PROBLEM & SOLUTION

A. Routing Problem

Streaming videos have high-bandwidth requirements. The routing problem is to determine paths between each video source and the receiver such that all flows get a good throughput while utilizing the available bandwidth effectively. Since all flows end at the receiver, this problem is same as constructing an aggregation tree with receiver as the root and sources as leaves or intermediate nodes of the tree.

Total number of bytes that can be received by the receiver in a unit time is limited by the capacity of the channel. This is the upper bound for the sum of the throughput of all flows. However, the actual aggregate throughput is usually much less than this. The reason for this is as follows. Since all nodes are operating in the same frequency band, the nodes that are within each other's sensing range contend for the channel access. Intra-flow contention occurs when nodes along a multi-hop path carrying the same set of flows contend with each other. This limits the total throughput along a multi-hop path. In contrast, when one or more flows merge together or when they are spatially close enough to contend with each other, the capacity is shared among the flows and the throughput of each flow reduces. While it is hard to eliminate intra-flow contention for a single-channel mesh network, spatial separation of routes for different flows can reduce inter-flow contention and improve the throughput for each flow.

B. Fairness and Rate Allocation Problem

Since our system model assumes that the video sources are identical and generate similar bit-rate streams, the network resources need to be fairly shared among all the flows. This can be achieved by a rate based flow control at each of the sources [7]. All the flows are assigned the same bit-rate that can be transported successfully to the tree root. The rate allocation and fairness problem is to determine the actual rate that can be supported by the aggregation tree. This problem is different from the routing, which deals with the tree construction, while here we are interested in finding the maximum per flow bit-rate for a given instance of the tree.

C. Packet Loss and Delay-Jitter Problem

There are two kinds of packet losses that occur for real-time video transmission over multi-hop wireless networks. Firstly, a packet might be received corrupted due to channel errors [6]. 802.11 MAC uses retransmissions to improve the reliability. Secondly, for playback of real-time video every packet has a deadline before which it has to be received at the receiver. Packets that arrive late are considered lost and discarded. Packet losses induce distortion in the reconstructed video and degrade the quality of the video. Thus it is desirable to reduce losses. Packet losses and network congestion cause large variations in the one-way delay experienced by packets of a flow. Standard technique to smooth out this jitter is to employ a playback buffer that adds some delay between actual streaming and playback time. Packets received ahead in time are buffered before being played back. Theoretically, jitter can be completely eliminated by having an infinite playback buffer. But due to the nature of live streaming, it is desirable to have a low delay before viewing. Therefore, the goal is to employ as little playback delay as possible. Moreover, the lower delay also implies lower buffer size requirements. In order for video to be played back without disruption, the playback buffer should never be empty.

1. Simulation setup and scenarios

In order to check the effect of the size of the contention window at MAC Layer for real-time video transmission in WMN, I use OPNET Modeler [10] 11.5 as a simulation software package with Microsoft visual C++ 6.0 as a supporting tool on Microsoft Windows XP operating system. The Opnet simulation software package is structured so that each network is modelled as a configuration of nodes which are interconnection of specific modules representing the various processes those take place in the actual communications equipment. In the case of wireless network, the interconnections among the nodes are automatically determined during the process of the simulation as a function of user-supplied propagation parameters, such as the effective transmission range of nodes.

Scenario for simulation is shown in fig below. The size of scenario is office level and contains total 16 numbers of nodes.

In Fig.3 we assume that all stations are steady except *video receiver*. *Video receiver* follows the path which is shown using green line and experiences two handovers at access point. *ftp_client* and *ftp_server* are for generating traffic.

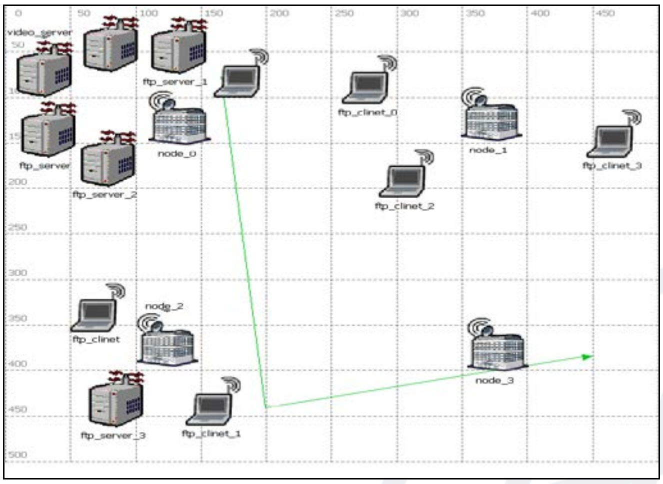

Fig.3 Simulation scenario

2. Node Model Editor

The Node Editor is used to create models of nodes. The node models are then used to create node instances within networks in the Project Editor. Internally, OPNET node models have a modular structure. You define a node by connecting various modules with packet streams and statistic wires. The connections between modules allow packets and status information to be exchanged between modules. Each module placed in a node serves a specific purpose, such as generating packets, queuing packets, processing packets, or transmitting and receiving packets.

This is an OPNET node model for wireless network nodes which incorporates the proposed scheme for network simulation. Wireless_lan_receiver and *wireless_lan_ transmitter* process model of node model implements the functionality of the Physical Layer. *Wireless_lan_mac* and *wireless_mac_intf* process model implements functionality of the Link Layer for 802.11 MAC Protocol. *Dsr_routing* and *dsr_intf* process model implements functionality for Network Layer using DSR Routing Protocol. Process models above this process models implements functionality of Upper Layers.

Here the encoder process uses synthetic video source based upon the group-of-pictures gamma-beta auto-regression (GOP **GBAR**) model. It is used for variable-rate MPEG video sources.

Network Abstraction Layer (NAL) is a part of the H.264 Video Coding Standard. The coded video data is organized into NAL units, each of which is effectively a packet that contains an integer number of bytes. The first byte of each NAL unit is a header byte that contains an indication of the type of data in the NAL unit, and the remaining bytes contain payload data of the type indicated by the header. The NAL unit structure definition specifies a generic format for use in both packet-oriented and bit stream-oriented transport systems, and a series of NAL units generated by an encoder is referred to as a NAL unit stream [11].

Video Server node model:

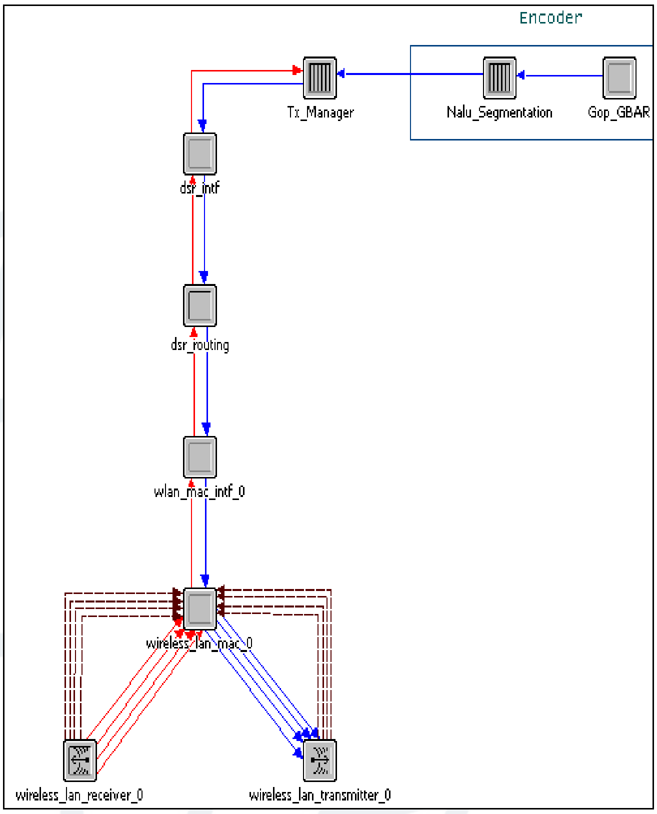

Fig.4 Video server node model

Tx Manager keeps track to the information for packets forwarding.

Video Receiver node Model:

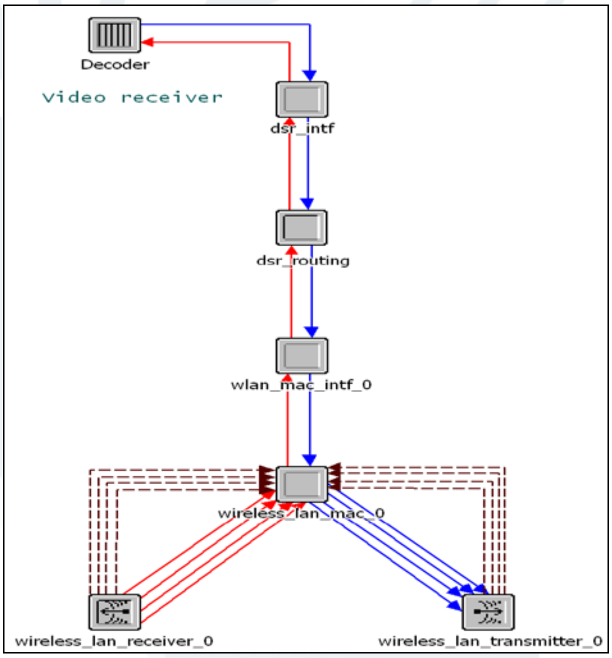

Fig.5 Video receiver node model

Decoder process decodes and reforms video packets received.

Automatic Repeat Request (ARQ) manager process responsible for receive packet and send back acknowledge to the server back.

Access point node Model:

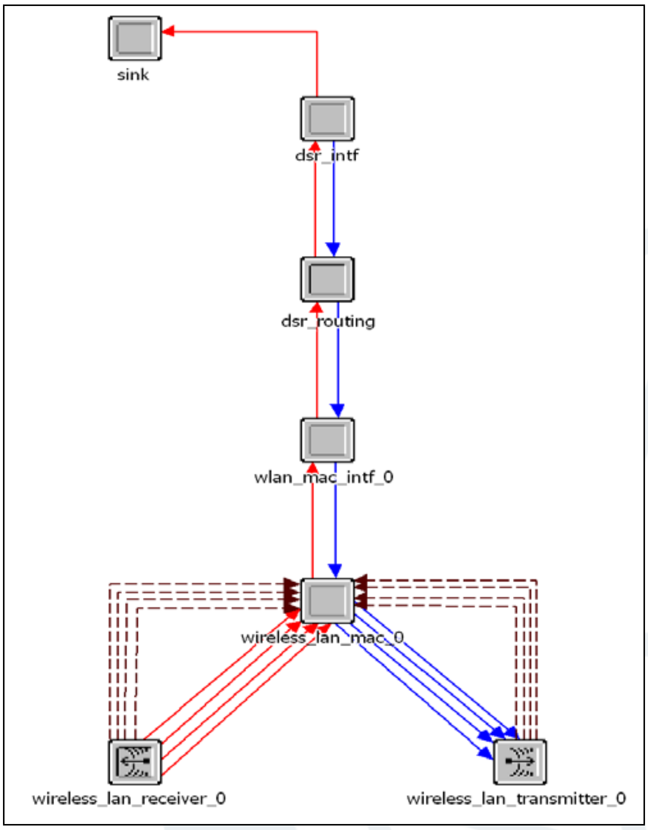

Fig.6 Access point node model

Sink process receives the packet store in queue and forwards towards the destination based on FIFO.

FTP Server node model:

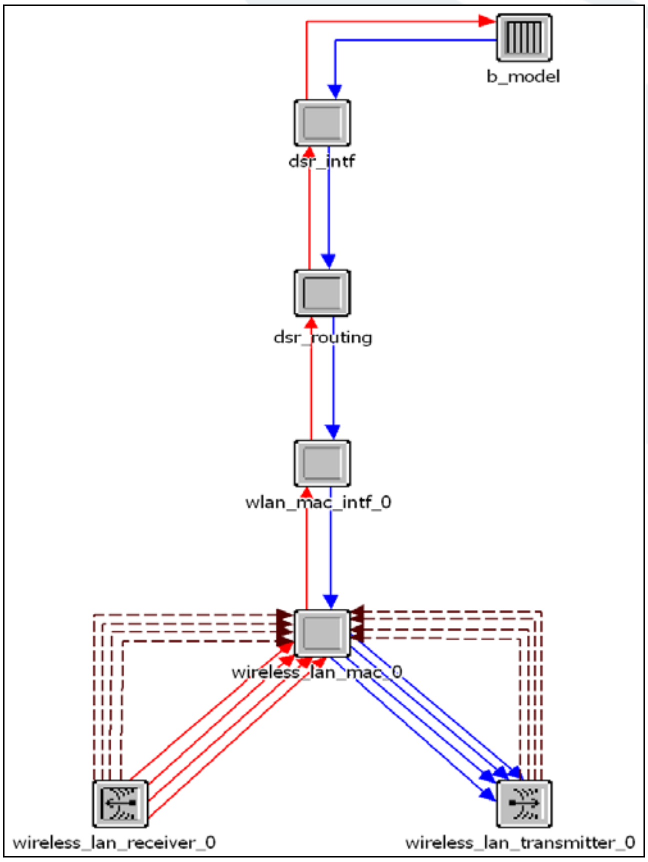

Fig.7 FTP server node model

FTP Client node model:

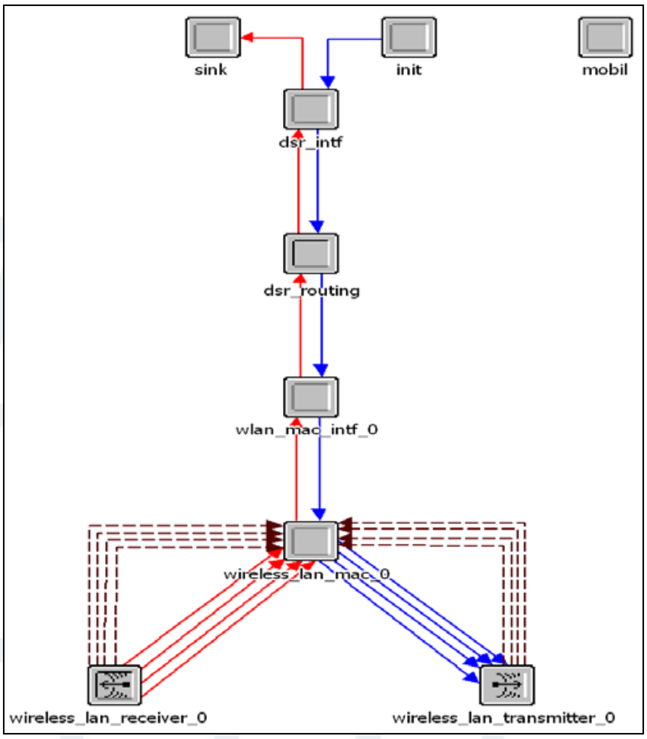

Fig.8 FTP client node model

3. MAC Layer Process Model Editor

To create process models which control the underlying functionality of the node models created in the Node Editor one can use the Process Editor. Process models are represented by finite state machines (FSMs) and are created with icons that represent states and lines that represent transitions between states. Operations performed in each state or for a transition are described in embedded C or C++ code blocks[9].

INIT

Initialization of the process mode. All the attributes are loaded in this routine.

BSS_INIT

Schedule a self interrupt to wait for MAC interface to move to next state after registering

IDLE

The purpose of this state is to wait until the packet has arrived from the higher or lower layer.

In this state following interrupt can occur:

1. Data arrival from application layer
2. Frame (DATA, ACK, RTS, CTS) rcvd from PHY layer
3. Busy interrupt stating that frame is being rcvd
4. Call interrupt indicating that more than one frame is received.

When Data arrives from the application layer, insert it in the queue. If receiver is not busy then set Deference to DIFS and Change state to "DEFER" state.Rcvd RTS, CTS, DATA, or ACK (frame rcvd interrupt) set Backoff flag if the station needs to backoff[11].

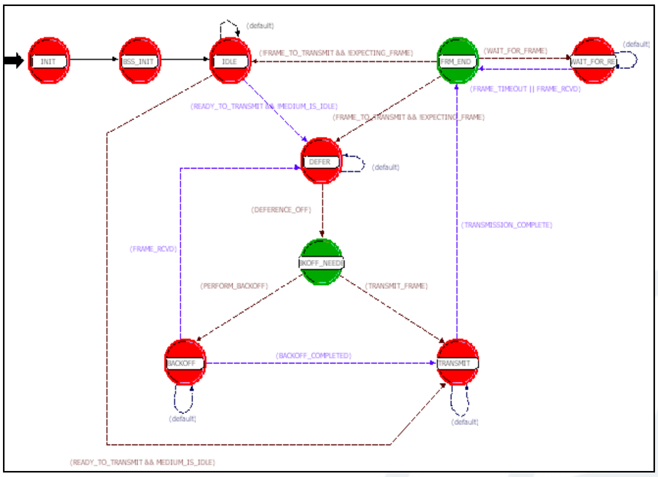

Fig.9 MAC Layer process model

If the frame is destined for this station then send appropriate response and set deference to SIFS clear the receiver busy flag and clamp any data transmission.

If it's a broadcast frame then set deference to NAV and schedule self interrupt and change state to "DEFER". Copy the frame (RTS/DATA) in retransmission variable if receiver start receiving frame (busy stat interrupt) then set a flag indicating receiver is busy, if receiver start receiving more than one frame then set the rcvd frame as invalid frame set deference time to EIFS.

DEFER

Call the interrupt processing routine for each interrupt.

WAIT_FOR_RESPONSE

The purpose of this state is to wait for the response after transmission. The only frames which require acknowledgements are RTS and DATA frame. In this state following interrupt can occur:

1. Data arrival from application layer
2. Frame (DATA, ACK, RTS, CTS) rcvd from PHY layer
3. Frame timeout if expected frame is not received
4. Busy interrupt stating that frame is being received
5. Collision interrupt stating that more than one frame is received

Queue the packet as Data Arrives from application layer

If received unexpected frame then collision is inferred and retry count is incremented if a collision stat interrupt from the receiver then flag the received frame as bad.

FRM_END

The purpose of this state is to determine the next unforced state after completing transmission.

There are three cases:

1. If just transmitted RTS or DATA frame then wait for response with *expected_frame_type* variable set and change the states to wait for Response otherwise just DEFER for next transmission.
2. If expected frame is rcvd then check what is the next frame to transmit and set appropriate deference timer.
   a. If all the data fragments are transmitted then check whether the queue is empty or If not then based on threshold fragment the packet and based on threshold decide whether to send RTS or not.
   b. If there is a data to be transmitted then wait for DIFS duration before contending for the channel.
   c. If nothing to transmit then go to IDLE and wait for the packet arrival from higher or lower layer.
3. If expected frame is not rcvd then infer collision, set back off flag, if retry limit is not reached retransmit the frame by contending for the channel.

If there is no frames expected then check to see if there is any other frame to transmit. Also mark the channel as idle.

BKOFF_NEEDED

Determining whether to backoff or not. It is needed when station preparing to transmit frame discovers that the medium is busy or when the station infers collision.

Backoff is not needed when the station is responding to the frame.

If backoff needed, then check whether the station completed its backoff in the last attempt. If not then resume the backoff from the same point, otherwise generate a new random number for the number of backoff slots.

At the Enter execs level change in code is shown in the fig.10 for changing the size of contention window.

BACKOFF

Call the interrupt processing routine for each interrupt.

TRANSMIT

If the packet is received while the station is transmitting then mark the received packet as bad.

The behaviour of the simulation model governed by a number of user defined parameters, lumped under the Wireless LAN Parameters attribute and selected via an OPNET graphical user interface. The parameters are listed in figures below. Some parameters are of numeric type where as others are popup list type.

3.4 Parameter values:

The behaviour of the simulation model governed by a number of user defined parameters, lumped under the Wireless LAN Parameters attribute and selected via an OPNET graphical user interface. The parameters are listed in figures below. Some parameters are of numeric type where as others are popup list type.

```
/* Checking wether backoff is needed or not */
if (wlan_flags->backoff_flag == OPC_BOOLINT_ENABLED)
    {
    if (backoff_slots == 0)
        {
        /* Compute backoff interval using binary exponential process    */
        if (retry_count != 0)
            {
            /* Set the maximum backoff for the uniform distribution */
            max_backoff = max_backoff * 2 + 1;
            }
        else
            {
            /* if retry count is set to 0 then set the  */
            /* maximum backoff slots to min window size */
            max_backoff = cw_min;
            }

        /* The number of possible slots grows exponentially */
        /* until it exceeds a fixed limit.                  */
        if (max_backoff > cw_max)
            {
            max_backoff = cw_max;
            }

        /* Obtain a uniformly distributed random integer between 0 and the minimum contention window size   */
        /* Scale the number of slots according to the number of retransmissions.                           */
        backoff_slots = floor (op_dist_uniform (max_backoff + 1));
        }

    /* Set a timer for the end of the backoff interval.                 */
    intrpt_time = (current_time + backoff_slots * slot_time);

    /* Scheduling self interrupt for backoff */
    backoff_elapsed_evh = op_intrpt_schedule_self (intrpt_time, WlanC_Backoff_Elapsed);

    /* Reporting number of backoff slots as a statistic */
    op_stat_write (backoff_slots_handle, backoff_slots);
    op_stat_write (backoff_slots_handle, 0.0);
    }
```

Fig.10 Code change

Attribute	Value
name	video_receiver
model	SR_dsr_receiver_simple
trajectory	VR_Path
wireless_lan_mac_0.Wireless LAN Parameters	(...)
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
Data Rate (bps)	1 Mbps
Physical Characteristics	Frequency Hopping
Short Retry Limit (slots)	7
Long Retry Limit (slots)	4
Access Point Functionality	Disabled
Channel Settings	(...)
Bandwidth (Khz)	10
Min Frequency (Mhz)	30
Buffer Size (bits)	256000
Max Receive Lifetime (secs)	0.5
dsr_intf.Transmit	Yes
dsr_intf.destination	1
dsr_routing.Dsr Address	use the mac address
dsr_routing.Dsr Non Propagating Request	activated
dsr_routing.Dsr Request Life Time	5.0
dsr_routing.Dsr Wait Ack Time	10
dsr_routing.Dsr Wait Reply Max Time	20
dsr_routing.Dsr Wait Reply Min Time	5.0
dsr_routing.Mac Address Field	wireless_lan_mac_0.station_address
wireless_lan_mac_0.station_address	0
wlan_mac_intf_0.Destination Address	Random

Fig.11 Attributes of Video Receiver

Attribute	Value
name	video_server
model	SR_dsr_video_server
trajectory	NONE
wireless_lan_mac_0.Wireless LAN Parameters	(...)
Rts Threshold (bytes)	256
Fragmentation Threshold (bytes)	1024
Data Rate (bps)	1 Mbps
Physical Characteristics	Frequency Hopping
Short Retry Limit (slots)	7
Long Retry Limit (slots)	4
Access Point Functionality	Disabled
Channel Settings	(...)
Bandwidth (Khz)	10
Min Frequency (Mhz)	30
Buffer Size (bits)	256000
Max Receive Lifetime (secs)	0.5
dsr_intf.Transmit	Yes
dsr_intf.destination	0
dsr_routing.Dsr Address	use the mac address
dsr_routing.Dsr Non Propagating Request	activated
dsr_routing.Dsr Request Life Time	5.0
dsr_routing.Dsr Wait Ack Time	10
dsr_routing.Dsr Wait Reply Max Time	20
dsr_routing.Dsr Wait Reply Min Time	5.0
dsr_routing.Mac Address Field	wireless_lan_mac_0.station_address
wireless_lan_mac_0.station_address	1
wlan_mac_intf_0.Destination Address	Random

Fig.12 Attributes of Video Server

II. RESULTS AND OUTCOMES

Results are obtained as object level statistics, Delay time and Throughput, collected by executing the simulation model for 40 seconds and compared with results obtained for standard MAC.

A. Delay Statistic

The delay in real-time MAC layer is exploited by the broadcast nature of wireless medium and limited loss tolerance of the applications. Each additional transmission consumes additional power and increases network load is time delayed in delivery of data. Here delay is introduced due to handovers. We can find the difference in delay statistics for standard and modified MAC. We can find the reduction of time delay in modified MAC. The difference is shown in Fig 13 and Fig 14.

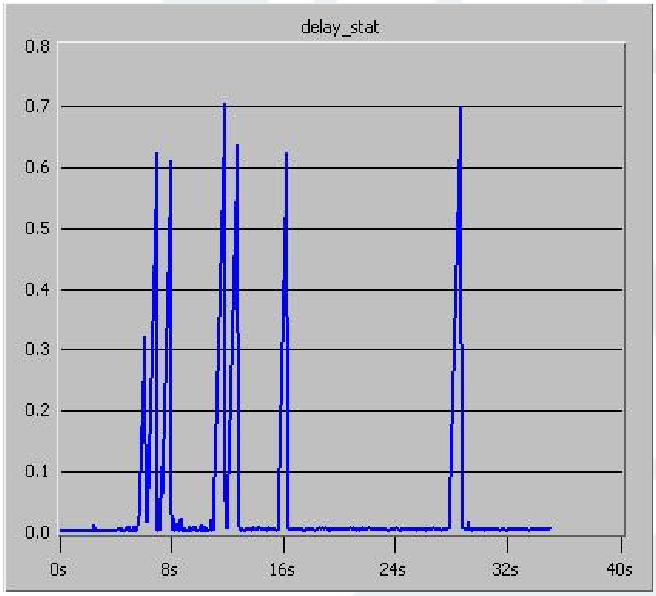

Fig.13 Delay Statistics for Standard MAC

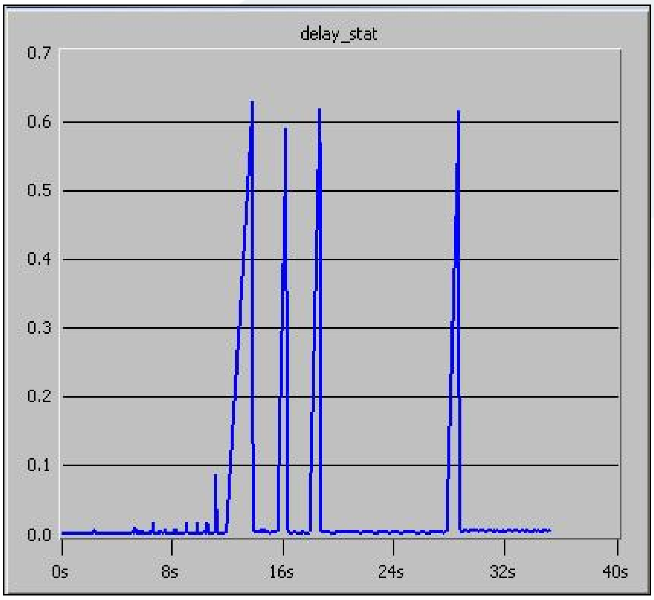

Fig.14 Delay Statistics for Modified MAC

B. Throughput Statistic

Throughput is calculated based on time taken for successful data delivery. Fig 15 we can find the through put achieved for standard MAC and in fig 16 we can find the through put achieved for modified MAC. We can find improved through put in Modified MAC.

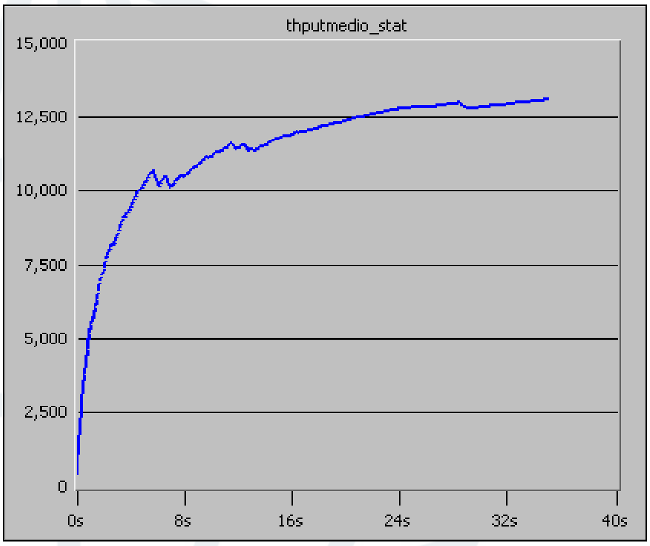

Fig.4.3 Throughput for Standard MAC

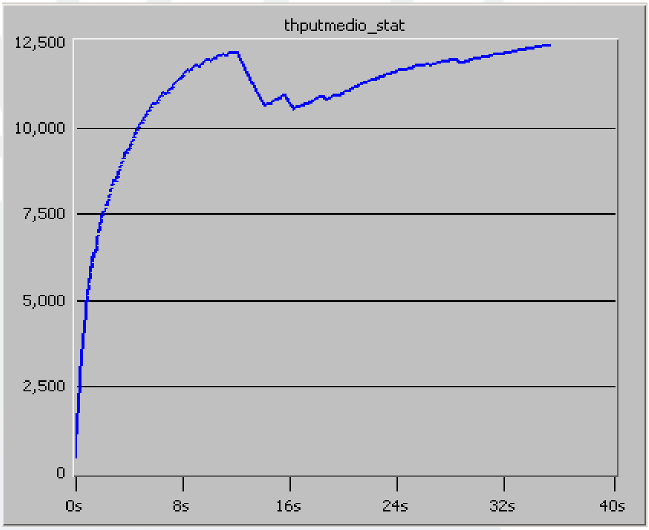

Fig.4.4 Throughput for Modified MAC

4.3 Result Analysis

By comparing the results obtained for delay time for standard MAC and modified MAC, we can see that delay time is decreased up to 0.1 seconds. By comparing statistics for throughput using Standard MAC and modified MAC, we can see that throughput is increased. Finally we are able to simulate transmission with less delay time and improved throughput.

III. CONCLUSION

In this paper we have proposed a Simulation model to demonstrate higher through put by reducing delay time during real time video transmission in wireless mesh networks. When we use WMNS, the complexity lies in Routing, Hence we have shown emphasis on routing, which causes more delay time during video transmission, if we don't have an efficient mechanism. We have verified the effect of the size of contention window at MAC Layer for real time video transmission in WMN. We have used OPNET Modeler 11.5 as a simulation software package with Microsoft visual C++ 6.0 as supporting tool on Microsoft Windows XP operating system. Finally the results are satisfactory

IV. REFERENCES

[1] A technical tutorial on the IEEE 802.11 protocol by PabloBrenner, www.sss-mag.com/pdf/802_11tut.pdf

[2] A. Iwata, C.C. Chiang, G. Pei, M. Gerla, and T.W. Chen, ''Scalable Routing Strategies for Ad hoc Wireless Networks,'' IEEE Journal of Selected Areas in Communications, vol. 17, no. 8, pp. 1369–1379, August **1999**. doi:10.1109/49.779920

[3] Akyildiz, I.F., Wang, X., Wang, W., "Wireless Mesh Networks: A Survey", Computer Networks Journal (Elsevier), March **2005**. doi:10.1016/j.comnet.2004.12.001

[4] Amre El-Hoiydi, "Implementation options for the distribution system in the 802.11 Wireless LAN Infrastructure Network", Proc. IEEE International Conference on Communications 2000, vol. 1, pages 164-169, New Orleans, USA, June **2000**. doi:10.1109/ICC.2000.853085

[5] An Adaptive *RTS/CTS* Control Mechanism for *IEEE* 802.11MACProtocol ,www.ee.ucla.edu/~ains/rubin/papers/Laura_VTC03_full_0227.pdf

[6] Anne H. Ren, Gerald Q. Maguire Jr., "An adaptive realtime IAPP protocol for supporting multimedia communications in wireless LAN systems", Proc. Of InternationalConferenceonComputerCommunications, pp. 437 - 442, Japan, Sept**. 1999**.

[7] Distributed Coordinated Function (DCF) http://www.vocal.com/network/802_11_index.html

[8] Future of Real time video communication http://www.iftf.org/FutureofRealTimeVideoCommunication

[9] IEEE standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, ISO/TEC 8802-11:1999(E), Aug**. 1999**.

[10] S. Kompella, S. Mao, Y. Hou, and H. Sherali, "Cross-layer optimized multipath routing for video communications in wireless networks," IEEE J. Select. Areas Commun., vol. 25, no. 4, pp. 831–840, May **2007**. doi:10.1109/JSAC.2007.070518

[11] S. Mao, Y. T. Hou, X. Cheng, and H. D. Sherali, "Multipath Routing for Multiple Description Video in Wireless Ad Hoc Networks," in IEEE INFOCOM, Miami, FL, pp. 740–750, Mar**. 2005**.

How to cite

Rajanikanth Aluvalu, "Efficient Real-Time Video Transmission in Wireless Mesh Network". *International Journal of Research in Computer Science*, 2 (1): pp. 11-19, December 2011. doi:10.7815/ijorcs.21.2011.010