Multimedia Traffic Support for Asynchronous Ad hoc Wireless Networks*

T. Bheemarjuna Reddy, B. S. Manoj, and C. Siva Ram Murthy† †
Department of Computer Science and Engineering
Indian Institute of Technology, Madras, India 600036
{arjun, bsmanoj}@cs.iitm.ernet.in and murthy@iitm.ernet.in

Abstract

In this paper we address the issue of providing multimedia traffic support in asynchronous Ad hoc wireless networks. Since multimedia traffic has stringent bounds on end-to-end delay we do resource reservation for transmitting such traffic. The existing asynchronous MAC protocols such as RTMAC [1] and MACA/PR [2] when used for multimedia traffic provide inefficient utilization of network resources and affect call acceptance ratio and call drop ratio of multimedia traffic severely. Hence in this work we modify the RTMAC protocol for supporting multimedia traffic, so that it overcomes these limitations and improves packet delivery ratio and end-to-end delay of such traffic.

The core concept of this protocol is a novel slot allocation strategy for efficient utilization of the available bandwidth for carrying multimedia traffic and best-effort traffic. Extensive simulations were performed to assess the performance of the protocol under varying network conditions. The simulations clearly indicate the gains in using such a slot allocation strategy for carrying multimedia traffic.

1. Introduction

The characteristics of Ad hoc wireless networks, such as unrestricted mobility of nodes, imprecise routing information, lack of centralized coordination, and limited resource availability (battery power and bandwidth), make QoS provisioning a very challenging problem for multimedia applications. Since it is almost impossible to transmit the multimedia traffic in uncompressed (raw) form over wired or wireless networks due to its huge bandwidth requirement, audio or video traffic is first compressed and then transmitted over the network.

*This work was supported by the iNautix Technologies India Private Limited, Chennai, India and the Department of Science and Technology, New Delhi, India.

††Author for correspondence

There are two major techniques to encode (compress) traffic generated by a video source, namely, Constant Bit Rate (CBR) encoding and Variable Bit Rate (VBR) encoding. In CBR encoding technique, the target bit rate is kept constant regardless of the complexity of frames in the video. But in the case of VBR encoding technique, the target bit rate varies depending on the complexity of frames to be encoded. Even after applying compression on the traffic generated by the multimedia source, the applications involve transmitting huge quantities of multimedia traffic, thereby consuming huge bandwidth on the path chosen for multimedia streaming. Here, multimedia streaming refers to the technique which allows the user to immediately playback the multimedia content received so far without waiting for downloading the whole media of the session. Multimedia streaming imposes strict bounds on delay and jitter, while it can tolerate small amount of packet loss. With the advent of the IEEE 802.11 a/b/g technology, which offers a data rate upto 54 Mbps for the wireless channels, multimedia streaming has become feasible in Ad hoc wireless networks. Applications such as radio broadcasting, voice communication, video-on-demand, video conferencing, and distributed gaming require multimedia streaming.

In streaming applications, the video is often pre-encoded and stored at streaming servers. The VBR video stream can be characterized with four parameters, namely, *minVbrPktSize*, *meanVbrPktSize*, *standardDeviationVbrPktSize*, and *maxVbrPktSize*. In terms of multimedia streaming over wired or wireless networks, CBR encoded traffic requires fixed amount of bandwidth while VBR encoded traffic requires variable bandwidth on the path chosen for carrying the multimedia traffic.

In this paper, we address transmission of spatial scalable encoded multimedia traffic [3] in Ad hoc wireless networks. Here we assume that a two-layer spatial scalable encoding technique is used to code the video into a Base Layer (BL) and one Enhancement Layer (EL). A BL coder is used to generate BL packets at a low bit rate with moderate image quality. In the spatial scalable encoding technique, the EL coder takes the input video frame, the decoded output of

Proceedings of the First International Conference on Broadband Networks (BROADNETS'04)
0-7695-2221-1/04 $ 20.00 IEEE

the BL packet, and encodes the difference between them into an EL packet. In this scalable video coding approach, the BL packets must be received completely in order to decode and display video with a basic quality. The reception of EL packets improves upon the basic video quality. The two-layer video coding technique has two main variants. In the first variant, the generated bit rates of both the layers are variable (VBR-VBR), while in the second variant the base layer produces CBR traffic and the enhancement layer produces VBR traffic (CBR-VBR) [4]. In this paper, we address the transmission of multimedia traffic that is encoded using CBR-VBR video coding technique. Combined with unequal error protection, spatial encoding technique provides a graceful degradation of the image quality in the case of transmission errors, which otherwise can result in a complete packet loss. This feature is especially important in wireless communication, where channel conditions degrade significantly over certain periods in time [5].

2. Related work

The contention-based MAC protocols such as the IEEE 802.11 [6], Multiple Access Collision Avoidance (MACA) [7], and Multiple Access Collision Avoidance protocol for Wireless LANs (MACAW) [8] cannot support real-time traffic as these kinds of traffic require QoS guarantees to be provided. These protocols lack any resource reservation scheme to do reservation for real-time traffic a priori. In order to support real-time traffic, certain protocols are proposed in the literature, which have built-in mechanisms for reserving resources a priori. These protocols can be further classified into *synchronous* and *asynchronous* protocols. Synchronous protocols require time synchronization among all participating nodes in the network, so that reservations made by a node are known to its neighbor nodes. Examples of synchronous protocols are Cluster TDMA [9], Distributed Packet Reservation Multiple Access (D-PRMA) [10], and Soft Reservation Multiple Access protocol with Priority Assignment (SRMA/PA) [11]. But it is difficult to achieve global time synchronization in Ad hoc wireless networks due to their inherent characteristics as mentioned earlier. On the other hand, asynchronous protocols do not require any global synchronization among nodes in the network. These protocols usually use relative time information for making reservations. Examples of asynchronous protocols are Multiple Access Collision Avoidance/Piggyback Reservation (MACA/PR) [2] and Real-time MAC (RTMAC) [1]. MACA/PR and RTMAC protocols assume that real-time traffic is CBR in nature. In the following we explain RTMAC protocol in brief and discuss how the existing asynchronous protocols are inefficient in supporting multimedia traffic.

2.1. RTMAC

RTMAC protocol supports real-time CBR traffic along with best-effort (BE) traffic. It consists of two parts: (a) a MAC layer protocol, which is a real-time extension of IEEE 802.11 DCF, in which the bandwidth reservation is done for real-time connections and (b) a QoS extension of the DSDV [12] protocol, which is responsible for end-to-end reservation and release of resources. The MAC layer protocol is in turn divided into two parts, a contention-based medium access protocol for BE traffic and a reservation-based medium access protocol for real-time CBR traffic. BE packets are sent using the normal RTS-CTS-DATA-ACK handshaking mechanism as described in [6]. For real-time CBR sessions, bandwidth reservations are made by reserving variable length time slots on super-frames, which are sufficient enough to carry the CBR traffic. Each super-frame consists of a number of reservation-slots (resv-slots). A node that has real-time packets for transmission, reserves a block of consecutive resv-slots, which are henceforth called connection-slots (conn-slots) on a super-frame and uses the same conn-slots to transmit in successive super-frames. The slot allocation differs from the TDMA scheme because in this case no time synchronization is assumed and the protocol uses a relative time for all reservation purposes. A three-way handshake mechanism (*ResvRTS-ResvCTS-ResvACK*) is used to effect the reservation.

2.2. Limitations of existing asynchronous protocols

The existing asynchronous solutions such as MACA/PR and RTMAC provide QoS support for only the real-time CBR traffic. Since a CBR application generates fixed length packets, RTMAC protocol reserves a set of resv-slots in super-frame, which are sufficient enough to accommodate Real-time DATA (CBR-DATA) and Real-time ACK (CBR-ACK) packets. But for carrying multimedia traffic that is encoded using CBR-VBR video coding technique, we have to modify the slot allocation strategy of RTMAC protocol. A naive slot allocation strategy (similar to the one used in RTMAC protocol) can be stated as follows. Since CBR-VBR encoding generates constant length packets (CBR-DATA packets) and variable length packets (*i.e.*, VBR-DATA packet length varies between *minVbrPktSize* and *maxVbrPktSize*), reserve a set of resv-slots in super-frame, which are sufficient enough to accommodate CBR-DATA packet, CBR-ACK packet, VBR-DATA packet of size *maxVbrPktSize*, and VBR-ACK packet. Though this approach ensures that multimedia traffic received by the receiver is of high image quality, it has the following limitations.

- Since the ratio of *maxVbrPktSize* to *meanVbrPktSize* is high (to the tune of 2 to 20 depending on the nature of

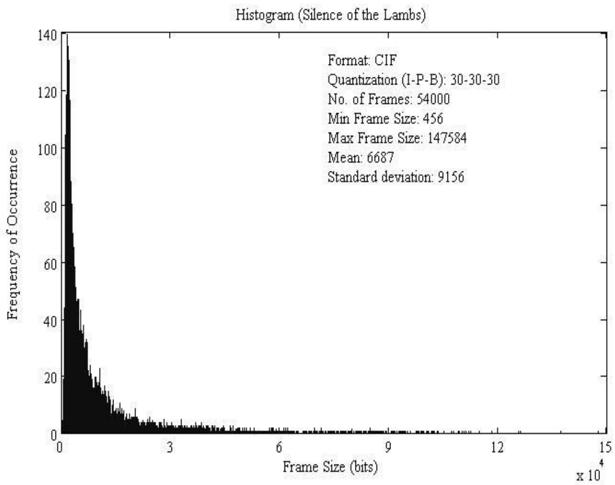

Figure 1. Histogram for the enhancement layer (VBR) traffic of Silence of the Lambs video trace, available at [3].

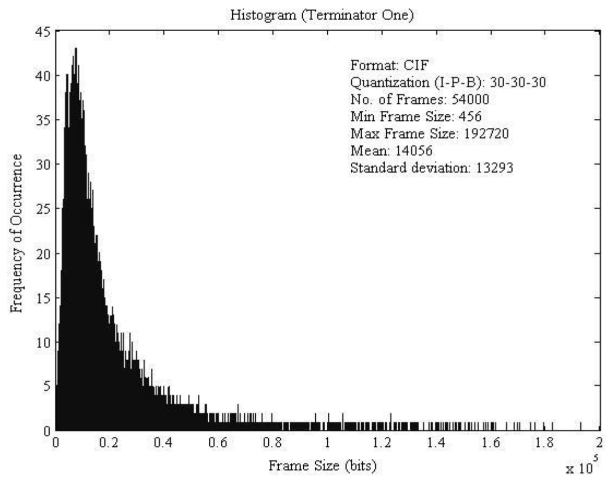

Figure 2. Histogram for the enhancement layer (VBR) traffic of Terminator One video trace, available at [3].

the given video stream, see Figures 1 and 2), the above mentioned allocation strategy results in inefficient allocation of limited resv-slots present in super-frame. Hence, the average call acceptance ratio and call drop ratio of multimedia traffic get affected.

- In Figures 1 and 2, we have plotted histograms of the enhancement layer (VBR-DATA) packets as a function of frame size for two of the video traces available at [3]. As shown in the figures, the sizes of the most of the VBR-DATA packets (approximately 80% to 90% of VBR-DATA packets depending on the nature of the given video stream) lie below the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize*. Hence during the multimedia session, most of the slots reserved for carrying the VBR-DATA packets remain unused. As it is not possible to make use of these unused reserved slots, throughput of the network remains low.

  Hence for supporting multimedia traffic and BE traffic, we need a MAC protocol with the following features.

  - Since scalable encoded multimedia traffic can tolerate packet loss to some extent, we need an efficient slot allocation strategy that exploits this feature while reserving resv-slots in super-frame, thereby improving average call acceptance ratio and call drop ratio of multimedia traffic.
  - The protocol should allow BE packets to dynamically make use of unused reserved slots (if any) of multimedia sessions, thereby increasing throughput of the network.

In the following section, we provide a MAC protocol with these features. The proposed protocol is a modification of RTMAC protocol for supporting multimedia traffic and BE traffic.

3. Our MAC protocol

In this paper we consider the transmission of spatial scalable encoded multimedia traffic and BE traffic in Ad hoc wireless networks. As mentioned earlier, with scalable coding, each frame generated by the video source is encoded into a fixed length base layer (CBR-DATA) packet and a variable length enhancement layer (VBR-DATA) packet. We use terms CBR-DATA packet and CBR packet, and VBR-DATA packet and VBR packet interchangeably in this paper.

3.1. Slot allocation strategy

Slot allocation strategy for BL packets is similar to the one used in RTMAC protocol. That means a certain number of resv-slots in super-frame needs to be reserved for these fixed length (CBR-DATA) packets. However, slot allocation process for EL (VBR-DATA) packets is not trivial. Here the problem is in determining how many number of resv-slots in super-frame needs to be kept aside for these packets. Our slot allocation strategy exploits the feature that the sizes of most of the VBR packets lie below the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize* and the ability of multimedia traffic to sustain some fraction of packet loss, while reserving slots for VBR packets. It allocates a set of resv-slots in super-frame that are sufficient enough to carry all the VBR packets whose packet sizes lie below the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize*. The remaining VBR packets are treated similar to BE packets. Each node maintains a BE priority queue. It holds QoS-DSDV control packets, the remaining VBR packets, and BE data packets in this order. These remaining VBR packets need to be transmitted within a super-frame time after insertion into the queue. If it is not possible to do so, such packets are discarded from transmission and removed from the corresponding priority queue.

We define $freeSlotSize$ as the maximum number of freely available resv-slots. $cbrSlotSize$ is the number of slots required to carry CBR-DATA and CBR-ACK packets. $type1VbrSlotSize$ is the number of slots required to carry VBR-ACK packet and VBR-DATA packet whose size lies below the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize*. $type2VbrSlotSize$ is the number of slots required to carry VBR-ACK packet and VBR-DATA packet whose size lies below *meanVbrPktSize*. If node A wants to establish a multimedia session with its neighbor node B, node A checks its reservation table along with its neighbor reservation information in order to determine a set of free slots for carrying CBR and VBR packets. This slot allocation strategy (SAS) is described in the following algorithm 1.

Algorithm 1 SlotAllocationStrategy ( )

if $(freeSlotSize \geq (cbrSlotSize + type1VbrSlotSize))$ **then**
  node A sends a *ResvRTS* containing relative time information of $(cbrSlotSize + type1VbrSlotSize)$ slots.
else if $(freeSlotSize \geq (cbrSlotSize + type2VbrSlotSize))$ **then**
  node A sends a *ResvRTS* containing relative time information of $(cbrSlotSize + type2VbrSlotSize)$ slots.
else if $(freeSlotSize \geq cbrSlotSize)$ **then**
  node A sends a *ResvRTS* containing relative time information of $cbrSlotSize$ slots.
else
  node A rejects the call request.
end if

The slot reservation mechanism is illustrated in Figure 3. In this case, the size of VBR packet considered is the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize*. Here node A sends *ResvRTS* packet which contains the relative time information of starting and ending of the set of resv-slots to be reserved for CBR packets and the relative time information of ending of the set of resv-slots to be reserved for VBR packets. Here we have to note that starting resv-slot for VBR packets is the one that succeeds the ending resv-slot for CBR-ACK packets. Upon reception, the receiver node B checks its reservation table to see whether it can receive in those resv-slots. If so, it replies with a *ResvCTS*. Upon hearing this *ResvCTS*, all neighbors of node B update their reservation tables. Node A acknowledges reception of *ResvCTS* with *ResvACK* and makes a valid reservation for the corresponding conn-slots in its reservation table. *ResvACK* notifies neighbors of node A about the reservation made currently. Once the reservation is made, CBR-DATA, CBR-ACK, VBR-DATA, and VBR-ACK packets are transmitted in these reserved slots in this order.

If node B receives a *ResvRTS* from node A on a slot which is already reserved by a neighbor of node B, then it will not respond to the *ResvRTS* by sending *ResvCTS*, instead it just discards it. This is because originating a positive or negative acknowledgment would cause collision with the actual reservation done by node B's neighbor. Hence node A may need to retry *ResvRTS* after some time. Node A may retry for *MaxResvRTSRetryLimit* times after which the reservation request is dropped. If *ResvRTS* is received successfully in a free slot, but the requested resv-slots are not free at the receiver node B, then it sends a Reservation Negative CTS (*ResvNCTS*) back to the node A. Upon receiving this, the node A makes another attempt following same procedure with another set of free resv-slots. Based on size of the conn-slots, CBR packets and most of the VBR packets are transmitted to the receiver node. The remaining VBR packets are inserted into BE priority queue and will be transmitted depending on the availability of free slots in the respective super-frames. Such packets need to be transmitted before sending subsequent CBR packets in the next super-frame. To this effect, a *worstCaseVbrDataTxTime* is defined for each VBR packet inserted into BE queue which is the maximum time limit before which the VBR packet needs to be transmitted to node B. The value of parameter *worstCaseVbrDataTxTime* depends on the super-frame time and the laxity (deadline - current time) of the VBR packet to be transmitted. Node A transmits a VBR packet from its queue, only if *worstCaseVbrDataTxTime* of that packet is greater than its local clock-time. Otherwise node A discards that VBR packet from transmission. Upon replying to the currently received CBR packet with a CBR-ACK packet, based on the laxity of the CBR packet received, node B will set a timer and waits for receiving VBR packet. Note that the value of timer lies in the range [0, super-frame time). If node

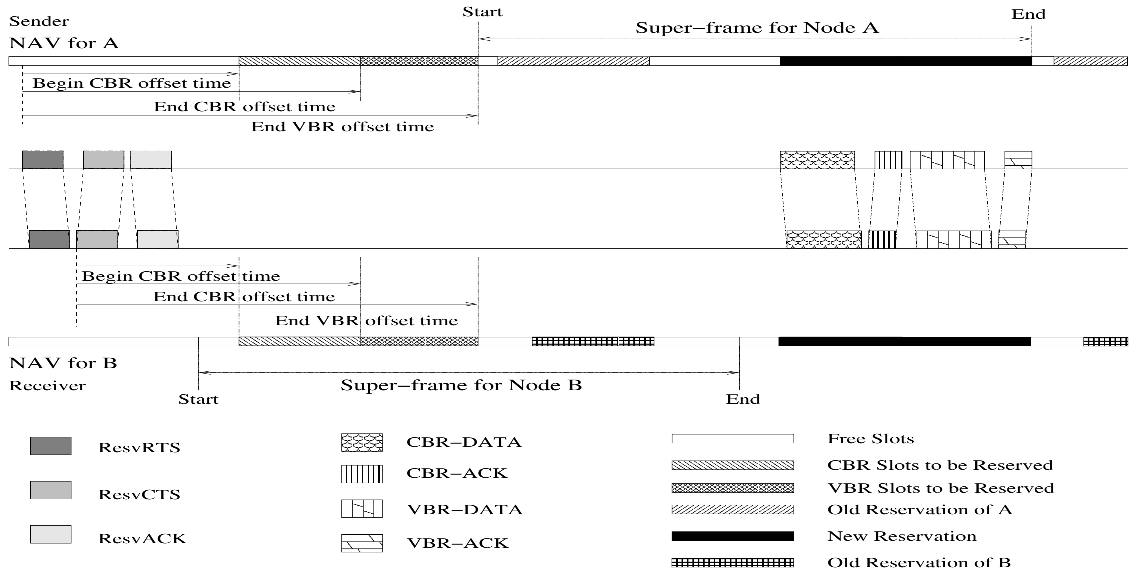

Figure 3. Illustration of the slot reservation mechanism for a multimedia session.

B receives the corresponding VBR packet before the timer expires, it cancels the timer, sends both CBR and VBR packets to its QoS-DSDV routing module, and acknowledges to the sender A with a VBR-ACK packet. Since multimedia traffic is delay-sensitive, the sender node does not retransmit unacknowledged CBR packets. But in case of failing to receive VBR-ACK packet, it stores VBR packet in its BE priority queue and will try to transmit within a super-frame time.

3.2. Transmission of BE traffic

For transmitting packets in the BE queue, at the time of backoff timer expiry, the remaining free slots in the current super-frame are checked. If the available free slots before the next conn-slots is greater than or equal to the slots required for transmitting the current packet from the queue, it will be sent. Otherwise, the node waits for the conn-slots to finish and then repeats the above steps again. If the current packet is a VBR packet, it will be sent only if its corresponding *worstCaseVbrDataTxTime* is greater than the local clock-time. Otherwise it is discarded and next packet in the queue will be considered for transmission.

Consider the scenario in which one of the intermediate nodes $IN1$ on the path from the source to the destination of a multimedia session reserved the conn-slots with its downstream node $IN2$ that are sufficient enough to carry CBR packet and either a VBR-DATA packet whose size lies below the sum of *meanVbrPktSize* and *standardDeviationVbrPktSize* or below the *meanVbrPktSize*. Assume that node $IN1$ has received only CBR packet from its upstream node on the path. Upon reception of CBR-ACK after transmitting that CBR packet to its downstream node $IN2$, node $IN1$ does not have the corresponding VBR for sending in the slots that are reserved for VBR packets. Instead of leaving these slots remain unutilized (wasted) in the current super-frame, node $IN1$ tries to send the current packet from the BE queue. For transmitting the current packet, the slots required to transmit it and get back its acknowledgment (for uni-cast packets) should be less than or equal to the remaining unused slots available in the conn-slots. Here we have to note that the receiver of the packet that is going to be transmitted from the queue need not to be the same downstream node $IN2$. Further, if the packet that is to be transmitted is a BE data packet, it is not necessary to exchange RTS and CTS control packets before its transmission. This is because of the current packet in the queue making use of the slots that are reserved for one of the multimedia sessions passing through that node $IN1$.

3.3. Reservation release mechanism

If a real-time session finishes or a path break is detected by a sender node, it releases the slots reserved in super-frame for that session, by sending Reservation Release RTS (*ResvRelRTS*) packet. *ResvRelRTS* packet is a broadcast packet which indicates that the sender requests its neighbors to release the resources,

Proceedings of the First International Conference on Broadband Networks (BROADNETS'04)
0-7695-2221-1/04 $ 20.00 IEEE

rather than the receiver itself. However, if the receiver hears the *ResvRelRTS* from the sender, it sends a Reservation Release CTS (*ResvRelCTS*) thereby informing its neighbors to release the reserved resources. A more complex situation occurs, when the receiver node or a neighbor node of the sender or the receiver moves away with an existing reservation. Such moved away nodes use a timeout mechanism to release the locked-up reserved slots in their reservation tables.

3.4. QoS-DSDV

The QoS routing protocol holds the responsibility of finding an end-to-end path which matches the QoS requirements such as bandwidth, delay, and buffer space. Here bandwidth is the QoS constraint which contributes to the advantages in end-to-end delay and packet delivery ratio. The QoS routing protocol used is an extension of the Destination Sequenced Distance Vector (DSDV) routing protocol [12]. DSDV is a table-driven routing protocol which can be used to do a fast reservation, wherein explicit end-to-end connection setup control packets are not required. Since a node reserves the bandwidth by explicitly giving the relative time (offset from the current time) to the free slot, the node should keep track of the neighboring nodes' reservation information. In order to maintain a consistent view of reservation tables of the neighboring nodes at each node, each node transmits its reservation information along with the route update packet which is defined as part of DSDV. Once a node receives this information, it updates the reservation information corresponding to that node.

Consider the scenario in which the QoS-DSDV routing module of node A wants to reserve a set of resv-slots with its downstream node B for a multimedia session. Then it finds a set of resv-slots by checking the reservation information of its neighbors and informs about it to its MAC layer module which in turn makes reservation with node B. This increases the chances of successful reservation at the first attempt of *ResvRTS* itself. After making reservation, node A sends a packet of that multimedia session in the corresponding conn-slots to node B. Upon receiving first packet from node A, the QoS-DSDV routing module of node B in turn reserves a set of resv-slots with its downstream node towards the destination of the multimedia session and sends the corresponding packets in that conn-slots.

In the above scenario, assume that node A only manages to reserve $cbrSlotSize$ slots with node B, while node B reserves more than that (*i.e.,* $cbrSlotSize$ + $type1VbrSlotSize$ or $cbrSlotSize$ + $type2VbrSlotSize$) with its downstream node. Such excess reservation by node B can be justified as follows. After sending a CBR packet to node B, node A inserts the corresponding VBR packet into the BE priority queue. This VBR packet is given more priority over BE packets and will be transmitted if there exists enough free slots in the upcoming super-frame. Since sizes of most of VBR packets are small and some of the on-going sessions may terminate in the mean time, it is most likely that most of VBR packets can reach downstream node B. Even if node B does not have the corresponding VBR packet after transmitting a CBR packet to its downstream node, instead of leaving these VBR slots remain unutilized (wasted) in the current super-frame, node B tries to send the current packet from the BE priority queue.

Consider the scenario in which one of the intermediate nodes $IN1$ on the path from the source to the destination reserved a set of resv-slots with its neighbor node $IN2$. Assume that QoS-DSDV routing module of node $IN2$ has received a CBR packet from its upstream node $IN1$. Before forwarding this packet, node $IN2$ checks whether it is worth to forward this packet to the downstream node. For this, node $IN2$ subtracts the transmission delay of the current hop from the value present in the remaining end-to-end delay (laxity) field of the QoS-DSDV header of CBR packet. If the packet is about to miss its deadline time, it discards this packet and the corresponding VBR packet (if available). Otherwise node $IN2$ sends them to its downstream node.

4. Simulation results

We simulated our protocol using GloMoSim. For BE traffic, we have used CBR sessions which generate datagram packets each of size 216 bytes, every 100 ms. For real-time (RT) traffic, we have used multimedia sessions which generate CBR packets each of size 216 bytes and VBR packets, every 100 ms. For VBR stream, *minVbrPktSize* is 10 bytes, *meanVbrPktSize* is 70 bytes, *standardDeviationVbrPktSize* is 30 bytes, and *maxVbrPktSize* is 200 bytes. 500 ms is taken as the deadline time. The length of super-frame is 90 ms. Simulation time was taken as 600 sec and sessions of 200 sec were generated randomly between 50 and 350 sec of the simulation. Terrain area was 1000 m$\times$ 1000 m. Transmission range of a node was 250 m. Mobility model used is the random way-point model with pause time of 25 sec. The channel capacity is 2 Mbps.

Extensive simulation experiments are carried out to assess the performance of our protocol by varying mobil-

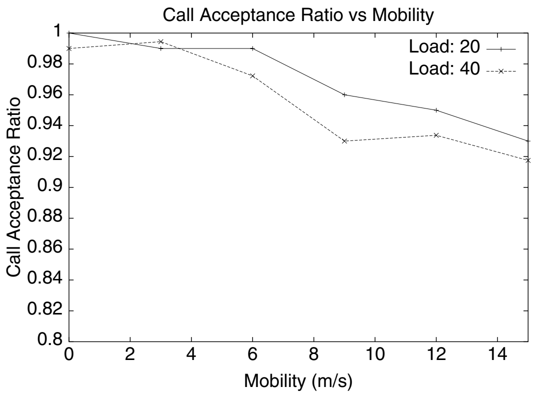

Figure 4. Variation of Call Acceptance Ratio vs Mobility.

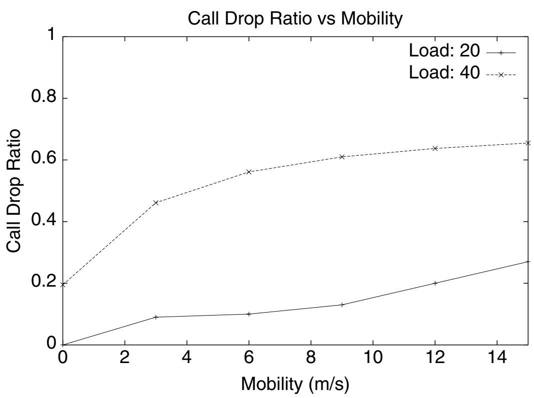

Figure 5. Variation of Call Drop Ratio vs Mobility.

ity of nodes, network density, and offered load in the network. Finally, we compared the simulation results obtained by using our novel VBR slot allocation strategy with those obtained using the CBR slot allocation strategy.

4.1. Effect of mobility

In this study, we have 50 nodes uniformly distributed in the terrain area. Mobility was varied from 0 m/s to 15 m/s with an increment of 3 m/s. The average call acceptance ratios of RT traffic are shown in Figure 4 for offered loads 20 (10 BE and 10 RT sessions) and 40 (20 BE and 20 RT sessions). As expected the call acceptance decreases with increasing speed. As the number of path breaks increases with mobility, the average call drop ratio of RT traffic (Figure 5) increases with mobility. The average end-to-end delay for both RT and BE traffic is shown in Figure 6. The figure shows a trend of stable average end-to-end delay for RT sessions because of having bandwidth explicitly re-

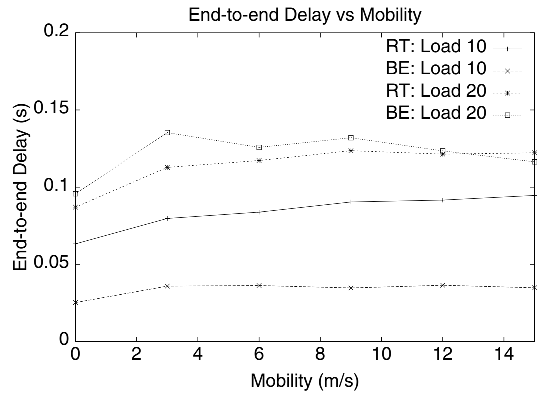

Figure 6. Variation of End-to-end Delay vs Mobility.

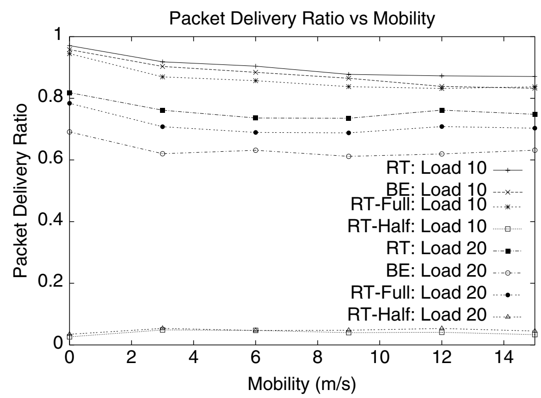

Figure 7. Variation of Packet Delivery Ratio vs Mobility.

served along the paths. However it is slightly higher at offered load 40 than the one at offered load 20 due to increase in call setup times at higher offer loads. In case of BE traffic, the average end-to-end delay increases when offered load increases. The variation of packet delivery ratio versus mobility is given in Figure 7. Here RT-Full refers to RT packets which are received at the destination with both CBR and VBR parts intact, while RT-Half means RT packets that lost their VBR parts during transmission. RT is the sum of RT-Full and RT-Half packets. The packet delivery ratios of RT traffic decrease slightly with increasing mobility. Though RT and BE traffic have almost identical packet delivery ratios at offered load 20, RT traffic has better packet delivery ratio at offered load 40. This is because of dropping of BE packets when BE priority queues overflow due to congestion in the network at higher offer loads.

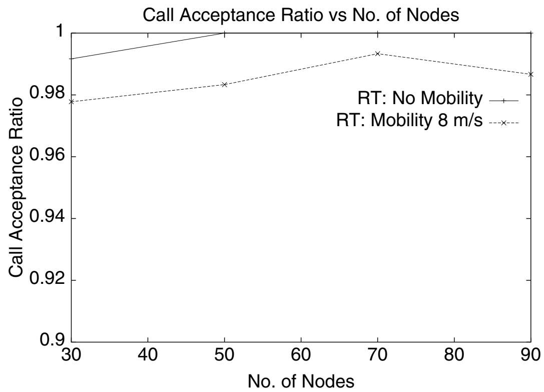

Figure 8. Variation of Call Acceptance Ratio vs Number of Nodes.

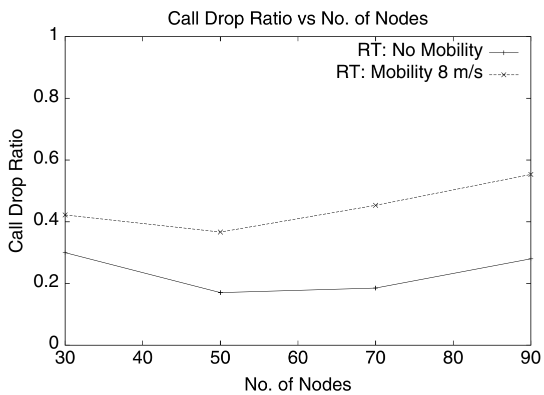

Figure 9. Variation of Call Drop Ratio vs Number of Nodes.

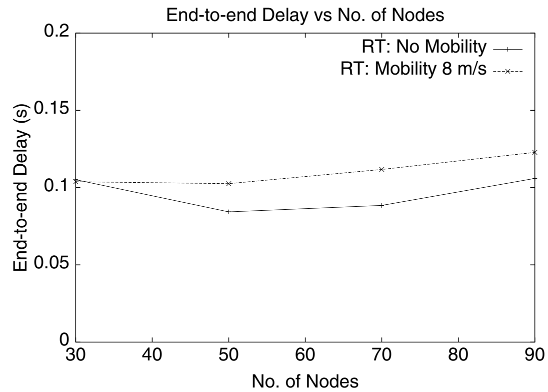

Figure 10. Variation of End-to-end Delay vs Number of Nodes.

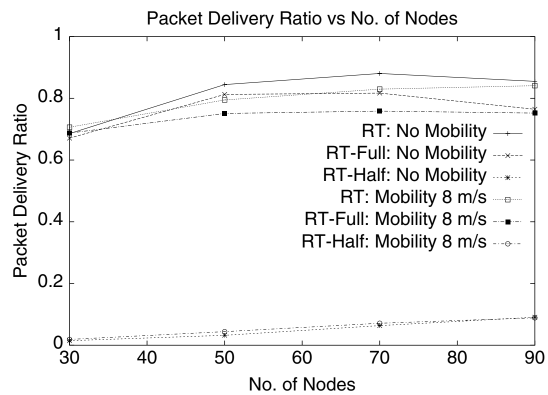

Figure 11. Variation of Packet Delivery Ratio vs Number of Nodes.

4.2. Effect of density of nodes

In this study, we assumed nodes are uniformly distributed in the terrain area. We considered 15 RT and 15 BE sessions that are distributed randomly among nodes in the network. The average call acceptance and call drop ratios of RT traffic are shown in Figures 8 and 9, respectively. Since the number of RT sessions is kept constant at 30 and are distributed randomly among nodes, the load on network is decreasing with increase in network density. Because of this, the call acceptance slightly increases with increasing the number of nodes. Since the average neighbor degree reaches its optimal value, the call drop ratio is low when network density is 50 [13]. Further at high network density, contention increases which in turn reduces efficiency of the channel usage.

The variation of average end-to-end delay versus number of nodes for RT traffic is shown in Figure 10. The figure shows a trend of almost constant average end-to-end delay for RT traffic, however it is slightly higher at mobility 8 m/s than the one at no mobility because of path breaks due to mobility of nodes. Variation of packet delivery ratios versus number of nodes are shown in Figure 11. We find that packet delivery ratio of RT traffic almost remains stable with increase in network density except for the case in which we have 30 nodes in the network. The reason is when there are 30 nodes in the network, the load on the network is high as we have taken 15 RT and 15 BE sessions in this simulation setting.

4.3. Comparison results

We compared the slot allocation strategy (SAS) used in our protocol with a naive slot allocation strategy. In the naive strategy, which we refer to as maximum allocation strategy (MAS), sender node admits an RT session only if there exists a set of free slots that are sufficient enough to carry CBR packet and VBR packet (whose packet size lies below the *maxVbrPktSize*). Since reserved slots can accommodate all VBR

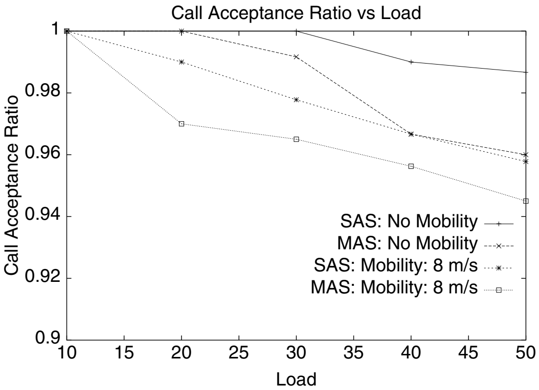

Figure 12. Variation of Call Acceptance Ratio vs Load.

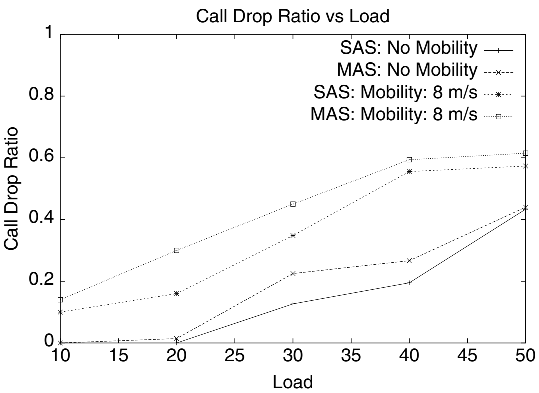

Figure 13. Variation of Call Drop Ratio vs Load.

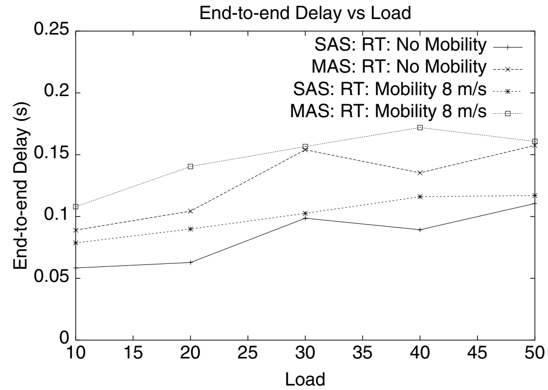

Figure 14. Variation of End-to-end Delay vs Load for RT Traffic.

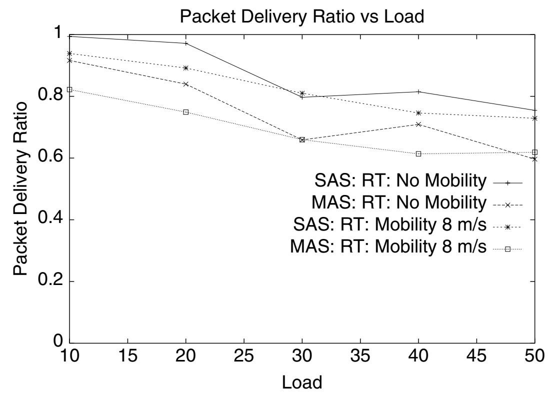

Figure 15. Variation of Packet Delivery Ratio vs Load for RT Traffic.

packets in MAS, there is no need to insert any such packets into BE queue and multimedia traffic that received at the receiver is of high image quality. We have studied the effect of load on RT and BE traffic under these two strategies. In these experiments we have 50 nodes uniformly distributed in the terrain area. We have uniformly increased the load by increasing BE and RT sessions in equal numbers. In Figure 12, the variation of the call acceptance versus load under these two strategies is shown. Since SAS (our slot allocation strategy) make uses of available free slots in super-frame more efficiently than MAS, the call acceptance for RT traffic is higher in it. Similarly, the call drop ratio of RT traffic is also better in SAS (Figure 13).

The variation of average end-to-end delay versus offered load for RT traffic is shown in Figure 14. The figure shows that SAS has lower end-to-end delay than the alternative scheme. The reason for this can be explained as follows. Consider the scenario in which one of the intermediate nodes $IN1$ on the path from the source to the destination has received only the CBR packet from its upstream node in our scheme. Due to lack of corresponding VBR packet, node $IN1$ forwards only the CBR packet to node $IN2$. The CBR packet carries the information that the corresponding VBR packet is no longer available. Upon receiving CBR packet, node $IN2$ forwards it immediately in the conn-slots of forthcoming super-frame to its downstream node without waiting for arrival of lost VBR packet. Since this type of fast forwarding is possible only rarely in MAS (*i.e.*, when the corresponding VBR packet is lost in transmission), it has slightly higher end-to-end delay than SAS.

In Figure 15, the variation of packet delivery ratio versus offered load for RT traffic is shown. The figure shows that our scheme has higher packet delivery ratio than the alternative scheme. The reason for this can be explained as follows. Since the average end-to-end delay is higher in the alternative scheme, it may result in dropping more number of CBR packets and the corresponding VBR packets at the intermediate nodes when they miss their deadlines.

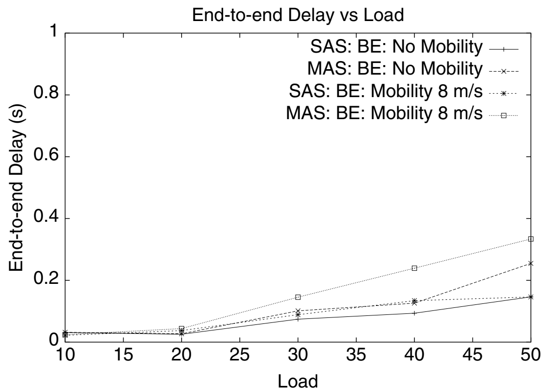

Figure 16. Variation of End-to-end Delay vs Load for BE Traffic.

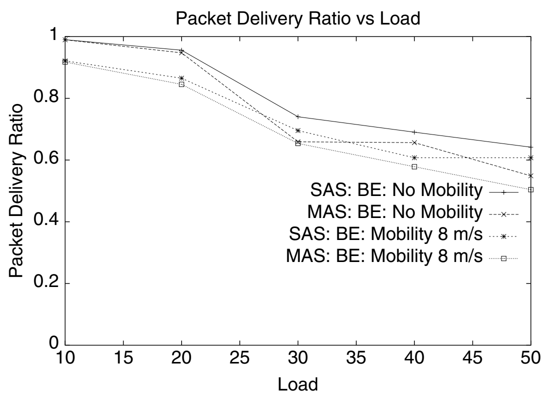

Figure 17. Variation of Packet Delivery Ratio vs Load for BE Traffic.

The variation of average end-to-end delay versus offered load for BE traffic is shown in Figure 16. In case of BE traffic, the average end-to-end delay increases when offered load increases. Since the number of free slots available decreases with increase in RT offered load, the end-to-end delay of BE traffic increases by increasing RT offered load. In Figure 17, variation of the packet delivery ratio versus load, under these two slot allocation strategies is shown. Since our scheme allows BE traffic to dynamically make use of unused reserved slots (if any) of multimedia sessions, for BE traffic it has slightly lower end-to-end delay and higher packet delivery ratio than that of MAS.

5. Conclusions and future work

In this paper we modified RTMAC protocol for supporting multimedia traffic in asynchronous Ad hoc wireless networks. Specifically, we proposed a novel slot allocation strategy for efficient utilization of the available bandwidth for carrying multimedia traffic and BE traffic. Currently we are extending this protocol, for carrying Fine Granularity Scalability (FGS) encoded multimedia traffic.

References

[1] B. S. Manoj and C. Siva Ram Murthy, "Real-Time Traffic Support for Ad Hoc Wireless Networks," *Proceedings of IEEE ICON 2002*, pp. 335-340, August 2002.

[2] C. R. Lin and M. Gerla, "Real-Time Support in Multi-Hop Wireless Networks," *ACM/Baltzer Journal of Wireless Networks*, vol. 5, no. 2, pp. 125-135, March 1999.

[3] M. Reisslein, J. Lassetter, S. Ratnam, O. Lotfallah, F. H. P. Fitzek, and S. Panchanathan, "Video Traces: MPEG-4: Single Layer, Temporal, and Spatial Scalability with PSNR Values," available at *http://www.eas.asu.edu/trace*.

[4] M. Ghanbari and J. Azari, "Effect of Bit Rate Variation of the Base Layer on the Performance of Two-Layer Video Codecs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 1, pp. 8-17, February 1994.

[5] U. Benzler, "Spatial Scalable Video Coding Using a Combined Subband-DCT Approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 7, pp. 1080-1087, October 2000.

[6] IEEE Standards Board, *Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, The Institute of Electrical and Electronics Engineers Inc., 1997.

[7] P. Karn, "MACA – A New Channel Access Method for Packet Radio," *Proceedings of ARRL/CRRL Amateur Radio Computer Networking Conference*, pp. 134-140, September 1990.

[8] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs," *Proceedings of ACM SIGCOMM 1994*, pp. 212-225, August 1994.

[9] M. Gerla and J. T. Tsai, "Multicluster, Mobile, Multimedia Radio Network," *ACM/Baltzer Journal of Wireless Networks*, vol. 1, no. 3, pp. 255-265, 1995.

[10] S. Jiang, J. Rao, D. He, and C. C. Ko, "A Simple Distributed PRMA for MANETs," *IEEE Transactions on Vehicular Technology*, vol. 51, no. 2, pp. 293-305, March 2002.

[11] C. W. Ahn, C. G. Kang, and Y. Z. Cho, "Soft Reservation Multiple Access with Priority Assignment (SRMA/PA): A Novel MAC Protocol for QoS-Guaranteed Integrated Services in Mobile Ad Hoc Networks," *Proceedings of IEEE Fall VTC 2000*, vol. 2, pp. 942-947, September 2000.

[12] C. E. Perkins and P. Bhagawat, "Highly Dynamic Destination-Sequenced Distance-Vector (DSDV) for Mobile Computers," *Proceedings of ACM SIGCOMM '94*, pp. 234-244, August 1994.

[13] E. M. Royer, P. M. Melliar-Smith, and L. E. Moser, "An Analysis of the Optimum Node Density for Ad hoc Mobile Networks," *Proceedings of IEEE ICC 2001*, vol. 3, pp. 857-861, June 2001.