

# Traffic Priority Based Adaptive and Responsive Congestion and Rate Control for Wireless Mesh Networks

Maheen Islam<sup>1</sup>, M. Lutfar Rahman<sup>1,2</sup> & Mamun-Or-Rashid<sup>1,3</sup>

<sup>1</sup> Department of Computer Science and Engineering, East West University, Dhaka, Bangladesh

<sup>2</sup> Department of Computer Science and Engineering, University of Dhaka, Dhaka, Bangladesh

<sup>3</sup> Daffodil International University, Dhaka, Bangladesh

Correspondence: Maheen Islam, Department of Computer Science and Engineering, East West University, Dhaka, Bangladesh. E-mail: maheen@ewubd.edu

Received: March 30, 2014      Accepted: April 25, 2014      Online Published: April 28, 2014

doi:10.5539/cis.v7n2p99

URL: <http://dx.doi.org/10.5539/cis.v7n2p99>

## Abstract

Congestion Control in Wireless Mesh Networks (WMNs) has been a widely studied context oriented utilization maximization problem. The traffic in WMNs has a wide range of variations (i.e. bandwidth, jitter, delay jitter sensitive applications) due to its use as the backbone network for accessing Internet. Eventually, traffic variation contributes to channel saturation and may bring up congestion due to contention caused by concurrent transmission, buffer overflows and time varying wireless channel condition. In this paper, we propose a distributed congestion control scheme for wireless mesh networks to ensure higher network throughput while avoiding congestion and maintaining inter-flow fairness where real time and non-real time traffic coexists. Our proposed technique handles congestion by restricting the available transmission rate of downstream nodes among the upstream nodes according to their flow demands based on three basic parameters: packet arrival rate, service rate and buffer occupancy. We also introduce prioritized queues in each node to treat real time and non-real time traffic differently. Therefore, congestion degree calculation, rate allocation and prioritizing traffic ensures higher network throughput and guaranteed delivery of real time traffic. Experiments conducted on ns-2 simulations demonstrate that our proposed algorithm can achieve significant improvements in both overall network throughput and inter-flow fairness for both non real time and delay bound traffic.

**Keywords:** wireless mesh network, congestion control, traffic prioritization

## 1. Introduction

Wireless mesh networking has become one of the most transpiring technologies for large scale network solutions. It covers most of the uncovered networking issues such as flexible, adaptive and reconfigurable architecture and at the same time, offers a cost-effective solutions to service providers. In WMNs, nodes are comprised of mesh routers and mesh clients, where mesh routers have minimal mobility and form the backbone of WMNs. They provide network access for both mesh and conventional clients. The integration of WMNs with other networks such as the Internet, cellular, IEEE 802.11, IEEE 802.15, IEEE 802.16, sensor networks, etc., can be accomplished through the gateway and bridging functions in the mesh routers. Mesh clients can be either stationary or mobile, and can form a client mesh network among themselves and with mesh routers. WMN is a promising wireless technology for numerous applications, e.g., broadband internet access, community and neighborhood networks, enterprise networking, building automation, etc (Akyildiz & Wang, 2005). The traffic in WMNs has a wide range of variations (i.e. bandwidth, jitter, delay jitter sensitive applications) due to its use as the backbone network for accessing Internet. Eventually, traffic variation contributes to channel saturation and may bring up congestion due to contention caused by concurrent transmission, buffer overflows and time varying wireless channel condition. As WMN is a multihop network, congestion taking place at a single node may diffuse to the whole network and degrade its performance drastically. Therefore, congestion control is a crucial issue for WMNs. Congestion control mechanism is responsible for preventing the occurrence of congestion, as well as for alleviating the impact of congestion on network if it occurs. Also, congestion control scheme should ensure a fair distribution of resources among nodes.

In the Internet, congestion has been resolved by applying end-to-end congestion control algorithm, i.e. Transmission Control Protocol (TCP). However, it has been reported that TCP does not perform well in a multi hop wireless environment (Holland & Vaidya, 2002), which usually results in inefficient and unfair bandwidth allocation among different flows. One of the well known reasons for TCP performance degradation is that traditional TCP assumes that the packet loss happens only due to the congestion in the network. However this assumption may not be true in wireless networks, since the packet loss also happens due to erroneous wireless characteristics, MAC contention, unstable network conditions and mobility of nodes. As TCP assumes all losses indicate congestion, when non congestion packet losses occur in wireless networks, besides retransmitting the lost packet, TCP also reduces its transmission rate as a result the network throughput drops quickly. Moreover once wireless channels are back to the normal operation, the classical TCP cannot be recovered quickly. Also an end-to-end congestion and rate control is inappropriate for wireless mesh networks, because it suffers from the adverse effects of multi-hop wireless environments, such as variable round-trip-times (RTT), high BER and radio interferences. Hop-by-hop schemes result in better performance than a corresponding end-to-end scheme by reacting to network congestion faster than end-to-end mechanisms. Besides TCP, many applications in internet such as audio and video streaming use User Datagram Protocol (UDP) as a transport protocol, since they require timely delivery of data rather than reliable transmission. As UDP protocol injects traffic into the network without seeking any feedback regarding the capacity of the network, more packets get collided and congestion situation gets worse. Coexistence of real time and non-real time traffic may direct the situation even worse. Rate control algorithms are incompatible for real time traffic as it has certain significant priority on time bound rather than reliability. Therefore, to avoid congestion WMN requires an effective congestion control protocol which is capable of avoiding/controlling congestion at the same time ensures a guaranteed service for the prioritized traffic, while assigning rates to flows, real time traffic should get higher opportunity to be transmitted as compared to non real time traffic.

In this paper, we propose a distributed congestion control scheme for wireless mesh networks to ensure higher network throughput while avoiding congestion and maintaining inter-flow fairness where real time and non-real time traffic coexists. Our proposed technique deals with three following basic parameters: packet arrival rate, service rate and buffer occupancy. Based on these basic parameters, we have calculated the congestion degree of a node which determines three different states: no congestion, anticipated congestion and congestion. In anticipated congestion state the downstream node restricts the available transmission rate among the upstream nodes according to their flow demands. In congestion state, the downstream node scales down the available rate among the upstream nodes proportionately to increase free buffer and thus alleviate congestion from the network. Our proposed technique also introduces two different queues in each node for handling real time and non-real time traffic. In each state, as we defined earlier, real time traffic gets more share of the allocated rate than the non-real time traffic. Therefore, congestion degree calculation, rate allocation and prioritizing traffic ensures higher network throughput and guaranteed delivery of real time traffic.

The rest of the paper is organized as follows. Section 2 discusses the related work in the literature. Section 3 describes network and node architecture, Section 4 describes the problem of congestion control in WMN and Section 5 presents the proposed protocol. Performance evaluation is carried out in Section 6 and the paper concludes in Section 7.

## 2. Related Works

In recent publications several approaches for congestion control in wireless multihop networks have been proposed. So far, there are two different types of approaches for congestion control in the existing literature: end-to-end congestion and hop-by-hop congestion control. However, a considerable research has been performed on improving the throughput of TCP for wireless environment. These works attempt to improve wireless TCP throughput by distinguishing the packet loss due to congestion and link failure. We have articulated the pros and cons of the above mentioned technique in the following: A lot of work has been done focusing the area of TCP enhancement where TCP responds to special wireless characteristics. To discern between congestion and non congestion losses in multihop wireless networks several variants of TCP such as TCP ELFN (Holland & Vaidya, 2002), TCP Feedback (Chandran & Prakash, 2001) and TCP-Bus (Kim & Choi, 2001) were proposed. Upon detection of a route failure event by the network layer, these protocols ceases further packet transmission until the route recovery. In (Liu & Sun, 2007), a TCP enhancement called Congestion Coherence for WMN has been proposed which distinguishes congestion losses from transmission errors and multipath reordering based on ECN (Explicit congestion notification) marking by intermediate routers and thus reduces false retransmission, timeouts, unnecessary congestion window reductions, and thus provides improvements than existing wireless TCP enhancements. In ECRFN (Zimmermann & Gunes, 2007), the TCP sources are alerted by the network

nodes regarding wireless errors and route failure, to prevent useless window size reduction and halt packet transmission till route is re-established. ATCP (Liu & Singh, 2001), introduces a layer known as Adhoc TCP between the TCP and IP layers which receives distinct feedback from network layer and internet layer regarding route failure, network congestion and packet corruption.

A good number of end-to-end rate based congestion control protocols has been discussed in literature, where the rate of the source node is regulated by its destination. Here each intermediate node in the network estimates the available bandwidth at the node and attaches the information with all forwarding packets. Upon receiving these packets, the receiver calculates the allowable bandwidth on the path and notifies the source through acknowledgement. In ATP (Sundaresan & Sivakumar, 2003), intermediate nodes in a connection stamps the sum of the average queuing and transmission delay experienced by packets traversing through them and the receiver sends rate feedback to the sender using the weighted average the delay. ATP sender uses the inverse of this delay as the sending rate. In EXACT (Chen & Vaidya, 2004), routers calculate sustainable rates for each flow as the inverse of MAC contention and transmission time of a packet and then the rates are piggybacked. As both ATP and EXACT controls the rate only at the source node based on feedback from the network, the time until the feedback eventually reaches the source node, the rapidly varying medium conditions may change which leads to misleading rate adaptation by the source. In LRTP (Raniwala & Chiueh, 2007), each intermediate node measures the effective bandwidth of each of its wireless link, fairly allocates it among the flows going through it and the sender adjusts their sending rates based on the minimum allocation across all hops. Since WMN nodes in the proposed architecture run the LRTP protocol and the end user mobile nodes and the nodes in the wired network runs TCP, in LRTP each ingress/egress WMN node has to employ a TCP-LRTP proxy. LRTP protocol also suffers from unfair channel sharing problem as when an intermediate node allocates channel bandwidth among the input flows, it does not consider the interference relationship among the neighboring nodes and the number of flows going over them. In AR-TCP (Venkata & Siva, 2006), link quality is measured considering the signal strength received from each of its neighboring node and available bandwidth is estimated. RBCC (Zhai & Fang, 2005), measures the sending rate of each flow by the channel utilization status at the bottleneck node which is the ratio of the total length of the busy period to the total time during a time interval. WCP (Rangwala & Govindan, 2008) is an AIMD-based rate-control protocol which explicitly reacts to congestion within a wireless neighborhood. On congestion at a link, WCP signals all flows traversing the neighborhood of that link, to multiplicatively reduce its rate by half. The main difference with TCP is that congestion is signaled to all flows traversing the neighborhood of a congested link. In EWCCP (Rangwala & Govindan, 2008), the intermediate routers calculate the congestion feedback based on the size of the neighborhood queue on each wireless link and adds feedback in the congestion header of the packet. When the packet arrives at the receiver, the feedback field in the congestion header holds the sum of all feedbacks given by all wireless links along the path. The aggregated congestion feedback is echoed back to the sender with an acknowledgement from the receiver. The main objective of TCP-AP (ElRakabawy & Lindemann, 2005), is to adaptively set a TCP senders transmission rate using an estimate of the current 4-hop propagation delay and the coefficient of variation of recently measured round-trip times. It is well known that the packets in multihop wireless networks do not collide with the transmission of the packets four hops away. Hence, the problem of packet bursts is solved by spreading the transmission of successive data packets according to the calculated 4-hop propagation delay from the round trip time (RTT) estimated at the source. As TCP-AP schedules packets in TCP layer, for multiple flows between source and destination it suffers from improper scheduling of packets at MAC layer that leads to self contention of packets in the network. This protocol also shows severe unfairness towards the sources that are located longer distance away from their destinations, since it depends on RTT to schedule packets. ElRakabawy and Lindemann (2009) introduces MAP, which operates at the wireless TCP source as well as at the mesh gateway, and transmits TCP packets by adapting the transmission rate according to the current network state. MAP identifies the current load in the neighborhood by measuring the current level of contention by means of the coefficient of variation of recently measured round trip times and also accounts for the spatial reuse constraint of IEEE 802.11 mesh networks by measuring Out-of-Interference Delay (OID) as the time elapsed between transmitting a TCP packet by the TCP source node and receiving the packet at a hidden terminal. In (Liu & Sun, 2007), when a routers queue length exceeds a threshold, the router stamps every incoming packet with a ECN (Early Congestion Notification) bit set. With ECN, when a packet is dropped by a congested router, the ECN congestion signal carried by that packet is lost, but packets before and after the lost packet maintain coherent congestion information. A packet loss is considered as a congestion loss if any packet in its coherence context is marked. In this case, the receiver responds with duplicate ACKs to trigger an end-to-end retransmission and window reduction at the source. Al Islam and Raghunathan (2011) devise a neural network based congestion control technique named iTCP, where competing flows are treated independently and fairness is not addressed.

Hop-by-hop congestion control studied in literature has been discussed in following, where local feedback on the sustainable rate for each node is transmitted to the respective upstream node, in order to establish some kind of backpressure towards the source. SECC (Sadeghi & Yang, 2006), uses local information available at nodes to detect congestion, computes target rate for the congested flows, notifies upstream nodes the target rates and finally upstream nodes modifies 802.11e channel access parameters to adjust the MAC transmission rate to remove. SECC outperforms TCP in wireless networks due to being able to reacting to dynamic wireless condition and neighborhood congestion promptly. But SECC congestion control in one part of the network may cause congestion in other part of the network. Additionally, this approach uses explicit feedback to the upstream nodes that imposes feedback traffic on the network. In AR-TP (Pace & Gungor, 2007) protocol the router calculates the rates for the flows passing through it based on two different threshold levels of router buffer; employs a back pressure mechanism to request the previous neighbor in the data flow for a rate adaptation and also asks the next hop neighbor to allow the congested node to increase the sending rate for that kind of traffic that caused congestion through a forward threshold adaptation technique. In order to mitigate congestion in one part of the network, AR-TP (Pace & Gungor, 2007) may introduce congestion on the other parts of the network. By using the forward threshold adaptation mechanism, a node may release its own congestion, but introduce congestion in its downstream nodes. Link Layer Adaptive Pacing (LLAP) (Franklin & Murthy, 2008) scheme tries to reduce the MAC contention in the network by properly scheduling the packets at the source node. In LLAP, the four hop transmission delay in a path is estimated by measuring the queuing and transmission delay incurred at the bottleneck node in a distributed manner and accordingly packets are paced for transmissions to reduce self contention. In CXCC (Scheuermann & Mauve, 2008), a fast and efficient implicit backpressure mechanism is established towards the packet source by preventing the transfer of a second packet belonging to a particular flow to a node until this node has forwarded the previous one. Thus a CXCC requires keeping per flow state in the intermediate nodes which requires certain computational overhead. As CXCC is a cross-layer protocol, its implementation will also require significant changes over the network participants protocol stack. In CFRC (Alam & Lee, 2011), the rates of the links are maintained for a certain weight, and the injection rates of the flows are set accordingly based on the weight of the flows and the weight for which the links rate are maintained. If the average queue size of any link exceeds a predefined threshold, CFRC assumes that the link is congested. Links that route their traffic via the same bottleneck link synchronize their rate decrement, enforced through a rate-bounded backpressure sent by the bottleneck node to all links that use the bottleneck link to forward their packets. However, if the links obtain the rates based on their respective downstream bottleneck links, flows using different bottleneck regions might achieve different throughput. To avoid congestion, Masri and Gaiti (2012) anticipate the future buffer usage by combining the queue lengths during previous control intervals. NICC provides implicit multi-bit congestion feedback using underexploited fields in the 802.11 frames header to ensure accurate rate control without inducing additional overhead. Jamshaid and Levis (2014) discusses the role of buffer sizes in reducing queueing delays in wireless mesh networks. The proposed work aims to distribute the neighborhood buffer among the bottleneck contention neighborhood to maximize bottleneck utilization. In (Jamshaid & Shihada, 2013), authors propose a gateway controlled centralized scheduling mechanism for managing flow rates to avoid congestion and improve fairness between network flows.

### 3. System Models

#### 3.1 Network Model

We consider a Wireless Mesh Network modeled by a network graph  $G_N = (V_N, E_N)$ , where  $V_N = \{1, \dots, n\}$  is the set of nodes (mesh clients, mesh routers and gateways) and  $E_N = \{(i, j) : i, j \in V_N\}$  is the set of bi-directional links between the nodes. Some of the router nodes in the backbone network, called gateway nodes, connect to the Internet through wired link. The client nodes utilize the mesh backbone network by getting connected to the edge routers of the WMNs. In turns, the core router nodes form a multihop adhoc network among themselves to relay the traffic to and from mesh client nodes.

Table 1. NS-2 simulation parameters

Parameter	Value
Total Area	1000 m × 1000 m
Number of nodes	100
CPTreshold value	1.55902e-11
CSThreshold value	3.25209e-10
Routing Protocol	AODV
Packet size	64 Bytes
Buffer length	50
Data Sources	5~25
Offered load	5~100 pkts per sec (pps)
Simulation time	50 s

### 3.2 Node Model and Traffic Classifier

Coexistence of different priority based network traffic led us towards traffic classifier with a view to offer priority based treatment. In our proposed technique, we consider two different traffic classes namely: real time traffic class (RT class) and non real-time traffic class (NRT class). The RT class is assigned to high priority real time traffic such as real time audio/video. For NRT traffic, having a low delay is not too important. Higher priority traffic classes need to have higher throughput and low delay bound. We implement separate queues for each traffic class. To discriminate traffic classes from each other, the mesh node adds a traffic class identifier to its local packets and places them in the proper queue. This identifier represents the traffic class of each packet. An intermediate node receives packet in input queue, which are then sent to different traffic class queues according to their traffic class. To provide quality of service for high priority RT traffic flows in a wireless mesh network, a scheduler moves the packets from the Traffic class Queue to Transmission Queue using weighted round robin (WRR) scheduling to be discussed in Section 5.2.1.

### 4. Congestion in WMN

To understand the effect of congestion in WMN we perform the following experiments using the popular tool ns-2 (ns-2, 2008) with 100 nodes under various traffic loads. The parameters used in our study are listed in Table 1. We estimate the achieved throughput in Figure 1 for different number of source nodes by varying the offered load in the network. The characteristics observed in throughput vs offered load graph signifies the fact that under lower offered load, the achieved end-to-end throughput increases linearly but reaches a maximum and starts decreasing with increasing offered load. The same observation is depicted in Figure 2, where end-to-end delay has similar linear and exponential relation with offered load. This confirms the urgent need of a congestion control protocol in WMN.

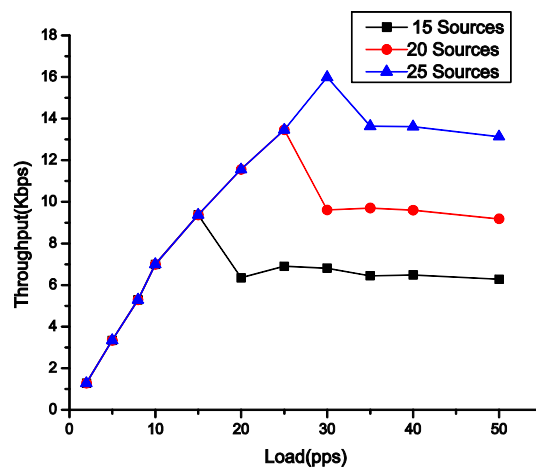


Figure 1. Achieved throughput for varying offered load for different number of sources

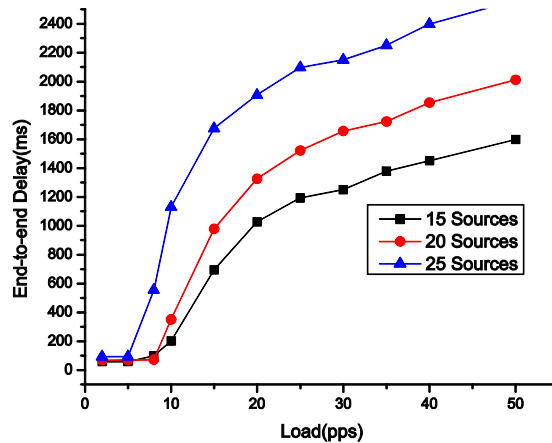


Figure 2. End-to-end delay for varying offered load for different number of sources

We now take a closer look at the load vs throughput and load vs delay characteristic and identify four different operating regions from the experiment and observed behavior. In Figures 3 and 4 we have replicated representative graphs for 15 sources from Figure 1 and Figure 2 respectively. Hence, our discussions and results in this paper apply to a general load vs. throughput and delay in WMN with any set of parameter values, with the specific case (Number of sources = 15) used only for illustration purposes. From Figures 3 and 4, it is evident that with increased traffic load the system progress through the following four operating regions:

- No Congestion, Low Throughput, Low Delay (NC, LT, LD): Initially when the traffic volume is low, network is congestion free, delivers packets quickly but throughput is low
- Anticipated Congestion, High Throughput, Low Delay (AC, HT, LD): More increase in traffic increases network throughput to a maximum level but heading towards congestion.
- Congestion, High Throughput, High Delay (C, HT, HD): Finally when network can not handle excessive flow of packets, congestion occurs resulting in packet loss and higher end-to-end delay.
- Congestion, Low Throughput, High Delay (C, LT, HD): As congestion state gets worse with further traffic load, more packets are dropped at congested nodes. Since the network capacity is wasted in partially moving packets from the source to the destination, network throughput drops off.

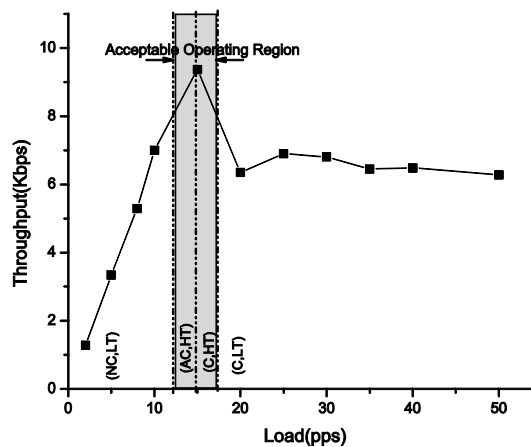


Figure 3. Four different operating regions and the acceptable operating region in throughput vs offered load graph

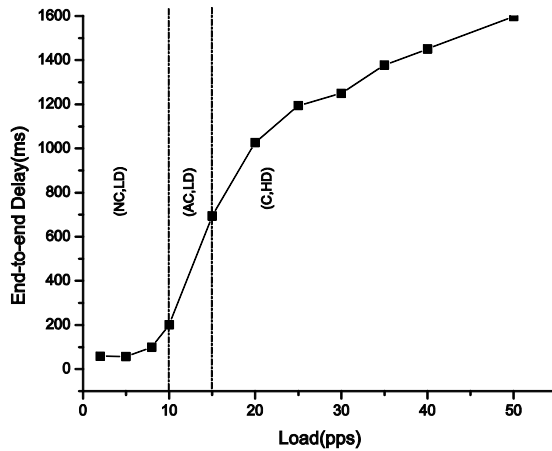


Figure 4. Three different operating regions in end-to-end delay vs offered load graph

Our aim is to maintain higher network throughput and lower delay while avoiding congestion, which can be achieved if traffic flow resides in acceptable operating region as mentioned in Figure 3.

Another significant observation regarding MAC level congestion is identified in the following.

In WMN, since neighboring nodes using CSMA like protocol share the same wireless channel, collisions could occur when multiple nodes try to grab the channel at the same time. This neighborhood congestion increases packet service time at nodes. Here the packet service time, is defined as the average processing time of the data packets which covers packet waiting, collision resolution and packet transmission times at MAC layer. When packet service time at a node is greater than the packet inter-arrival time, then there should be backlogged packets inside the node and the nodes queue builds up, which lead to significant delay in delivering packets, may cause packet drop and thus decrease overall system throughput. We define the ratio of packet service time and packet inter-arrival time at a node as congestion degree. We have measured the packet delivery ratio and congestion degree for various offered load in the network which is shown in Figure 5. We make the following observations from Figure 5.

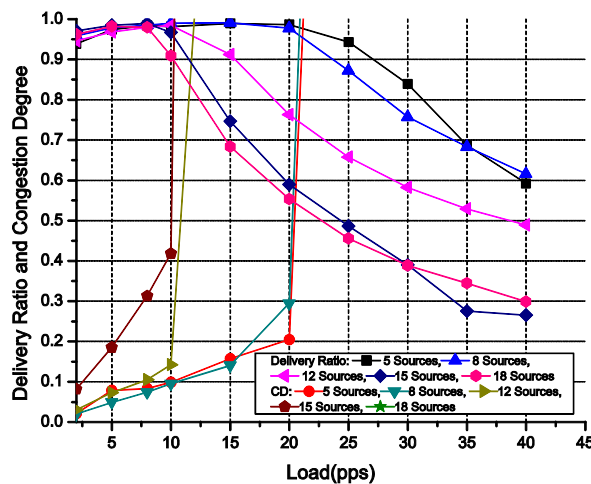


Figure 5. Delivery ratio and congestion degree for different offered load

- The delivery ratio of packets is high until a certain level of offered load, beyond which the delivery ratio drops. Such decrease in delivery ratio is observed regardless of the number of source nodes. Moreover it is also apparent that with greater number of source nodes the delivery ratio falls more quickly than smaller number of source nodes. This situation occurs since the network is unable to handle the increased injection of data packets and packets are dropped due to congestion which leads to poorer network throughput.

- For any number of source nodes, the value of congestion degree increases for increased offered load. It has also been observed that when the value of congestion degree is greater than 1, the delivery ratio of packets decreases. This is because when the nodes in a network serve the packets at a rate smaller than the rate at which the packets arrive, the value of congestion degree goes beyond 1. As a result arriving packets at nodes build up queues and at some point packets get dropped and thus lower the packet delivery ratio. Based on these observations, Congestion Degree seems to be a significant parameter that can be used for anticipating upcoming congestion.

## 5. Proposed Protocol

In this section we propose a congestion control mechanism which relies on information locally available at nodes. The developed protocol uses hop by hop signaling to notify the upstream nodes of the congestion and the rates the downstream nodes can support. The upstream nodes then adjust their transmission rates. In the proposed architecture, Congestion Detection module checks for congestion at a node considering local nodes input traffic rate, output traffic rate and buffer occupancy. Depending on the nodes congestion state, the Rate Adjustment module calculates the modified rates for the upstream nodes considering individual upstream nodes average input traffic rate. Congestion Signaling Module disseminates the modified rates to upstream nodes to relieve downstream nodes congestion state.

### 5.1 Congestion Detection

In order to precisely measure local congestion level at node  $V_i$ , we define two parameters for node  $V_i$ : average queue size,  $Q_{avg\_V_i}$  and congestion degree  $CD_{V_i}$ . Suppose the current queue size at node  $V_i$  is  $Q_{curr\_V_i}$ , then the average queue size is computed using EWMA (exponential weighted moving average) algorithm as follows in Equation (1).

$$Q_{avg\_V_i} = (1 - \beta) \times Q_{Avg\_V_i} + \beta \times Q_{cur\_V_i} \quad (1)$$

where  $\beta$  is a weight factor. A value of  $\beta$  closer to 0, calculates the average queue size giving more emphasis on the historical behavior of the queue. To find the effective value of the parameter  $\beta$ , in our simulation we varied the value of  $\beta$  to measure the number of packet drops at downstream nodes for variable number (2 to 5) of upstream nodes. The result listed in Table 2 reveals that the best possible result can be achieved by selecting a smaller value for  $\beta$ . In our simulation we used a value of  $\beta = 0.15$ .

Table 2. Effect of the value of parameter  $\beta$  on buffer drop rate

$\beta$	No. of Buffer Drops	Buffer Drop Rate (%)
0.1	98	0.28
0.15	97	0.26
0.2	201	2.21
0.25	326	2.93
0.3	482	3.25

Congestion degree of node  $V_i$ , denoted as  $CD_{V_i}$ , is defined as the ratio of the mean packet service time ( $T_{serv\_V_i}$ ) over the mean packet inter-arrival time ( $T_{arr\_V_i}$ ) over a predefined time interval in node  $V_i$  as in Equation (2):

$$CD_{V_i} = \frac{T_{serv\_V_i}}{T_{arr\_V_i}} \quad (2)$$

Here the mean packer inter-arrival time  $T_{arr\_V_i}$ , is defined as the interval between the arrival of two adjacent data packets from upstream traffic, and the mean packet service time  $T_{serv\_V_i}$ , is defined as the average processing time of the data packets at node  $V_i$ .  $T_{serv\_V_i}$  of a packet is the time elapse between the instant that a packet enters the queue to the instant that a link layer acknowledgement is recieved from the corresponding downstream or the packet is dropped after a predefined maximum number of retransmissions. Thus  $T_{serv\_V_i}$ , covers backoff delay, channel busy time and physical transmission delay. To measure  $T_{arr\_V_i}$  and  $T_{serv\_V_i}$  at each node  $V_i$ , EWMA (exponential weighted moving average) algorithm is used.

The congestion degree  $CD_{V_i}$  is proposed to reflect the current congestion intensity at each mesh node. When the inter-arrival time is smaller than the packet service time, the congestion degree is larger than 1 and the node



experiences congestion. Otherwise when the congestion degree is smaller than 1, it indicates that the volume of load offered to the node doesn't exceed the node's forwarding capability. Our proposed congestion detection algorithm categorizes a node's congestion state into one of the following by (i) monitoring node's congestion degree  $CD_{V_i}$  and (ii) comparing the node's average queue size  $Q_{avg_{V_i}}$  with a maximum queue threshold  $Q_{max_{V_i}}$ .

- (No Congestion state):  $Q_{avg_{V_i}} < Q_{max_{V_i}}$  and  $CD_{V_i} < 1$ . In this state no congestion is experienced as the average queue occupancy level is lower than the queue threshold.
- (Anticipated Congestion state):  $Q_{avg_{V_i}} < Q_{max_{V_i}}$  and  $CD_{V_i} > 1$ . When the packet arrival rate is greater than the departure rate, it indicates that congestion may occur at near future.
- (Congestion state):  $Q_{avg_{V_i}} > Q_{max_{V_i}}$  and  $CD_{V_i} > 1$ . When the average queue size exceeds the predefined maximum threshold, congestion is experienced by that node.

### 5.2 Rate Adjustment

In the proposed network model, each node  $V_i$  receives data packets from each of its upstream node  $V_j(Up_{V_i})$  at a rate  $Rate_{in_{V_j}}(Up_{V_i})$  and calculates the average input rate,  $Rate_{avg_{V_j}}(Up_{V_i})$  of upstream node  $V_j(Up_{V_i})$ , as Equation (3):

$$Rate_{avg_{V_j}}(Up_{V_i}) = (1 - w_q) \times Rate_{avg_{V_j}}(Up_{V_i}) + w_q \times Rate_{in_{V_j}}(Up_{V_i}) \quad (3)$$

where  $0 < w_q < 1$  is a constant. The total traffic input rate, denoted as  $Total_{in_{V_i}}$ , at node  $V_i$  is the sum of average input rates of all the upstream nodes  $V_j(Up_{V_i})$ .

Average queue size  $Q_{avg_{V_i}}$  and congestion degree  $CD_{V_i}$ , enables node  $V_i$  to detect congestion and average input rate of upstream nodes  $Rate_{avg_{V_j}}(Up_{V_i})$  and output rate of node  $V_i$   $Rate_{out_{V_i}}$ , provides information to calculate a target rate  $Rate_{tr_{V_j}}(Up_{V_i})$  for each of its upstream node  $V_j(Up_{V_i})$  to avoid or mitigate congestion. The algorithm listed in Table 3, summarizes the procedure of congestion detection and rate readjustment.

Table 3. Congestion detection and Rate adjustment Algorithm

---

Algorithm: Congestion detection and Rate adjustment

---

Input:  $Q_{max_{V_i}}$ ,  $T_{arr_{V_i}}$ ,  $T_{serv_{V_i}}$ ,  $Rate_{in_{V_j}}(Up_{V_i})$  for each upstream node  $V_j$

Result: Congestion state and  $Rate_{tr_{V_j}}(Up_{V_i})$

1: Initialize node information

2: Compute  $Q_{avg_{V_i}}$ ,  $CD_{V_i}$ ,  $Rate_{avg_{V_j}}(Up_{V_i})$ ,  $total_{r_{in}}$

3: if  $Q_{avg_{V_i}} < Q_{max_{V_i}}$  and  $CD_{V_i} < 1$  then

4: no congestion

5: end if

6: if  $Q_{avg_{V_i}} < Q_{max_{V_i}}$  and  $CD_{V_i} > 1$  then

7:  $Rate_{out_{V_i}} = \frac{1}{T_{serv_{V_i}}}$

8:  $Rate_{tr_{V_j}}(Up_{V_i}) = Rate_{out_{V_i}} \times \frac{Rate_{avg_{V_j}}(Up_{V_i})}{Rate_{total_{V_i}}}$

9: end if

10: if  $Q_{avg_{V_i}} > Q_{max_{V_i}}$  then

11:  $Rate_{out_{V_i}} = \frac{1}{T_{serv_{V_i}}}$ ;

12:  $Rate_{tr_{V_j}}(Up_{V_i}) = Rate_{out_{V_i}} \times \frac{Rate_{avg_{V_j}}(Up_{V_i})}{Rate_{total_{V_i}}}$

13:  $Rate_{tr_{V_j}}(Up_{V_i}) = \frac{Rate_{tr_{V_j}}(Up_{V_i})}{CD_{V_i}}$

14: end if

---

The mechanism of congestion detection and rate adjustment is performed periodically at a fixed time interval. Node  $V_i$  obtains its average queue size  $Q_{avg\_V_i}$ , mean packet inter-arrival time  $T_{arr\_V_i}$ , mean packet service time  $T_{serv\_V_i}$  to check for the congestion state. Our proposed algorithm reacts in different ways for different congestion state as following:

- As long as a node's buffer occupancy is underutilized, which has been recognized as (NC, LT, LD) region in Section 4, it does not send any rate adjustment notification to the upstream flows.
- When new nodes become active in neighborhood of node  $V_i$  or some of the upstream nodes produce more traffic, packet inter-arrival time decreases and as an impact congestion degree of  $V_i$  will increase and the  $V_i$  enters (AC, HT, LD) region. In such case node  $V_i$  experiences an early congestion. In order to avoid congestion, node  $V_i$  distributes its maximum allowable output rate among the upstream nodes based on their average input rate which ensures the network to operate in the expected operation region as mentioned in Figure 3.
- Finally when the average queue size exceeds the maximum queue threshold, node  $V_i$  experiences permanent congestion, highlighted as (C, LT, HD) region in Figure 3. In such a situation, upstream node's transmission rate is decreased more aggressively in order to reduce packet loss and alleviate congestion quickly. The target rates of the upstream nodes are scaled down based on node  $V_i$ 's output rate and degree of congestion.

### 5.2.1 Traffic Prioritizing

In our proposed method, a downstream node prioritizes to schedule packets for transmission considering their traffic type. To accomplish this, the downstream node maintains separate virtual queues for each traffic type ( $VQ_{NRT}$  for NRT class and  $VQ_{RT}$  for RT class) and a queue (VQ) to manage empty buffer spaces. Upon reception of a packet, the downstream node reserves the head bucket from the empty virtual queue for the packet and adds the bucket to the tail of the appropriate queue depending on the traffic class of the packet. When a packet is transmitted successfully, buffer space is made free and placed at the tail of empty virtual queue.

On reception of the first packet from an upstream node, downstream node assigns it to the appropriate virtual queue and the initial queue weight is 1 for the RT class traffic and 2 for NRT class traffic. When a packet is scheduled to be transmitted, the node selects packet from the virtual queue which has lowest weight. On transmission of each packet from the queue  $VQ_i$ , where  $i \in (RT, NRT)$ , its weight is updated as Equation (4).

$$w(VQ_i) = w(VQ_i) + \frac{Pckt\_size}{Pckt\_priority} \quad (4)$$

Where, Pckt\_size is the size of a packet in bytes and the value of Pckt\_priority is 1 for NRT traffic and 2 for RT traffic. Thus the scheduler serves RT traffic with higher priority while avoids the chance of starvation by using Round Robin (RR) scheduling.

## 6. Performance Evaluation

To assess the performance of our proposed scheme, we have performed extensive simulation using ns-2 simulator (ns-2, 2008).

### 6.1 Simulation Metric and Parameters

The simulation parameters are set as follows. 100 nodes are randomly distributed in a square region of 1000 m  $\times$  1000 m. For the simulation setup, the transmission range and carrier sense range are set to 250 m and 550 m, respectively. The routing protocol used is AODV (Adhoc on Demand Distance Vector Routing). The channel capacity is set to 11 Mbps and we assume that the queue size of all nodes to be 50. The range of channel contention window size is [1, 63]. The size of every packet is 64 Bytes. Number of sources transmitting packets, varies from 5 to 30 and the offered load varies from 5pps (packets/second) to 100 pps.

The following metrics has been realized to validate the efficiency of our proposed scheme.

- Throughput: The system throughput or aggregate throughput is the sum of the data rates that are delivered to all destinations in a network. We measured the throughput using the formula (total number of bytes received by all destinations / duration of the flow).
- Number of collisions: The number of dropped packets due to collision.
- Number of buffer drops: The number of dropped packets due to buffer overflows.
- Delivery ratio: Packet delivery ratio is defined as the ratio of data packet received by the destinations to those generated by the sources.
- End-to-end delay: End-to-end delay refers to the time taken for a packet to be transmitted across a network

from source to destination consisting of transmission and propagation delays plus variable queuing and processing delay.

- Congestion degree: Congestion degree of a node is the ratio of the mean packet service time over the mean packet inter-arrival time over a predefined time interval in the node.
- Efficiency: It is measured as the ratio of number of hops traveled by each successful reception of a packet at destination to the total number of transmissions required for the packet in the entire path.

The performance comparisons of the following four mechanisms are carried out.

- No Congestion Control (NoCC): Under this scheme packets are transmitted without controlling transmission rates at the sources and forwarders.
- Link Layer Adaptive Pacing (LLAP): LLAP (Franklin & Murthy, 2008) tries to reduce the contention in the network by scheduling packets from a source with an interval of FHD (Four Hop Delay) so that they do not collide to each other in the path. In this scheme, the downstream congested nodes packet forwarding delay is propagated to the source node by introducing additional delay to the transmission of packets at all the intermediate core nodes.
- Adaptive and Responsive Transport Protocol (AR-TP): It is a hop by hop congestion control mechanism explained in Pace and Gungor (2007). In AR-TP, every intermediate router is equipped with one buffer for each neighbor and two different thresholds are used in each buffer to monitor the magnitude of traffic load at the router. The rate control scheme of AR-TP protocol contains a back pressure mechanism to request the upstream node for a rate adaptation and a forward threshold adaptation mechanism which asks the downstream to allow the congested node to increase the sending rate to release congestion.
- Proposed Protocol: Our proposed technique employs congestion degree along with the buffer occupancy level at a node to determine different congestion states: no congestion, anticipated congestion and congestion. In anticipated congestion state the downstream node restricts the available transmission rate among the upstream nodes according to their flow demands. In congestion state, the downstream node scales down the available rate among the upstream nodes according to congestion degree to mitigate congestion immediately while maintaining high network throughput. In each node two different queues has been introduced for handling real time and non-real time traffic. For any congestion state of a node, real time traffic gets more share of the allocated rate than the non-real time traffic.

## 6.2 Simulation Metric and Parameters

Figure 6 shows the throughput of the different protocols for varying network loads. For NoCC, as the source nodes increase the amount of the traffic they transmit, network throughput increases. After the network reaches saturation, however, without use of congestion control the throughput is reduced due to collisions and contention overhead. But, with LLAP the reduction in throughput is less, as the sources adaptively controls the offered load into the network based on FHD of packets in the network, it guarantees that packets originating from the same source are not dropped due to self contention. AR-TP also achieves higher throughput than NoCC, as the upstream nodes pump packets into the network at a rate regulated by their downstream nodes taking into account the buffer state, which reflects the congestion status of the network.

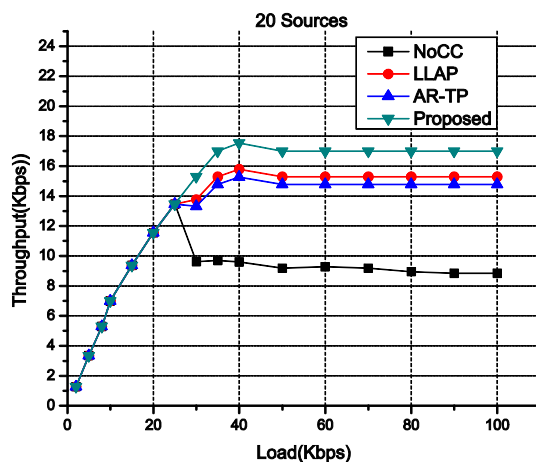


Figure 6. Throughput of different protocols for varying offered load

As seen in Figure 6, the proposed protocol outperforms other protocols under comparison, since it dynamically adjusts data rate of the flows in a responsive manner due its hop-by-hop nature and avoids network congestion, while maximizing the network utilization efficiency.

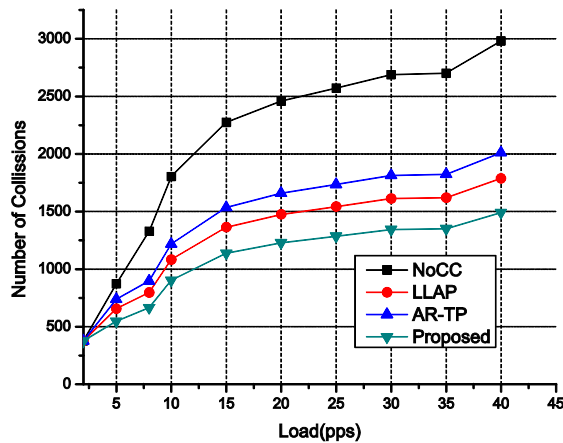


Figure 7. No. of collisions for varying offered load

Figure 7 and Figure 8 shows collision drop rate for various protocols. In case of NoCC, drop rate increases sharply with the increased number of injected packets and data sources. Since all nodes try to acquire the media for transmission, huge number of collision occurs. In LLAP, as all intermediate nodes synchronize their packet departure with the delay accounted by their immediate downstream nodes; they are able to reduce useless packet transmission retries that eventually reduce MAC collision. All intermediate nodes in AR-TP protocol use a Backpressure mechanism to notify its immediate downstream nodes the supported rate of the network which prohibit the downstream nodes to capture the congested shared media that eventually reduces number of collisions. But when a node tries to relieve its congested state, it uses a forward threshold policy where it

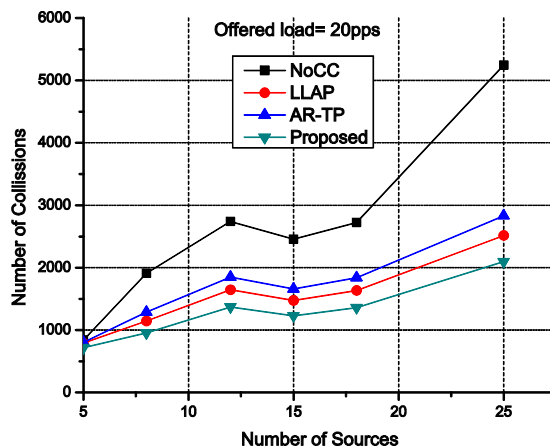


Figure 8. No. of collisions for different number of sources

requests the immediate upstream node to adjust the upstream nodes buffer threshold to give a chance to increase the downstream nodes rate without considering the neighborhood congestion phenomena. As a result when the neighborhood is already congested the rate increase also increases the collision rate. In the proposed method a node monitors its local congestion through its own buffer occupancy level and the neighbors congestion status by the congestion degree parameter. So when a node notifies a reduced rate to its immediate upstream, it reflects the overall neighborhood congestion condition of the networks. Thus in a busy environment, all nodes at the vicinity of a congested neighborhood, refrain itself from attempts to capture the media to avoid collision.

Buffer drop rates of different protocols are plotted in Figure 9 and 10. Uncontrolled rate of transmission mechanism in NoCC is the main reason of higher packet drops. LLAP and AR-TP scheme experience lower packet drops than NoCC, since they use rate control mechanism to control congestion. In proposed method, to minimize buffer drop, buffer occupancy level is monitored and compared with a threshold level. Whenever a rate feedback is sent to an upstream, buffer use and the time taken to serve a packet is taken into consideration; which works as an indicator of future high buffer occupancy level.

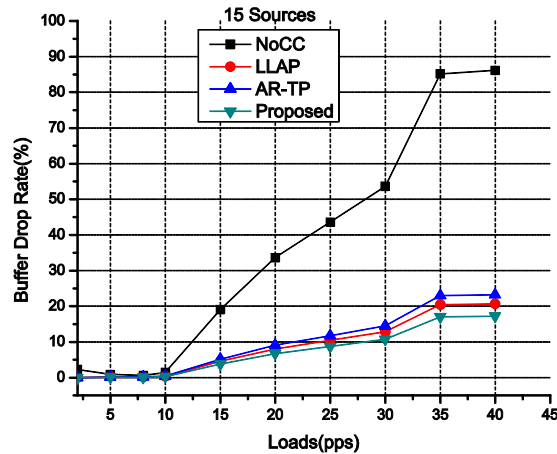


Figure 9. No. of packet drops due to buffer overflow for varying offered load

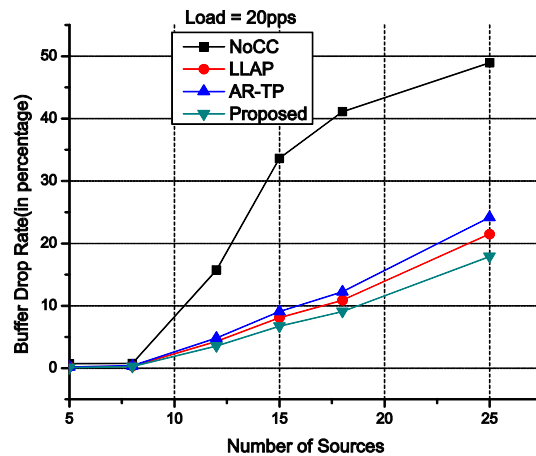


Figure 10. No. of packet drops due to buffer overflow for different number of sources

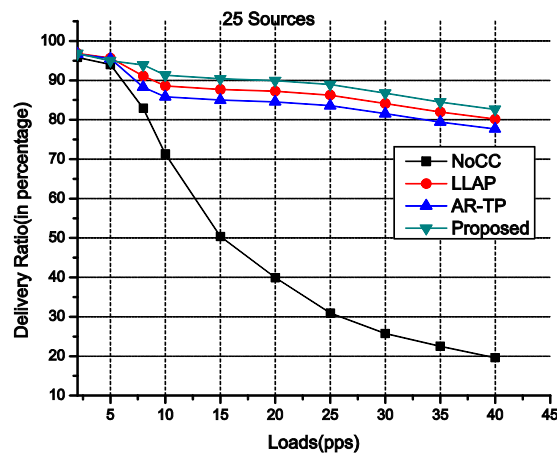


Figure 11. Packet delivery ratio for varying offered load

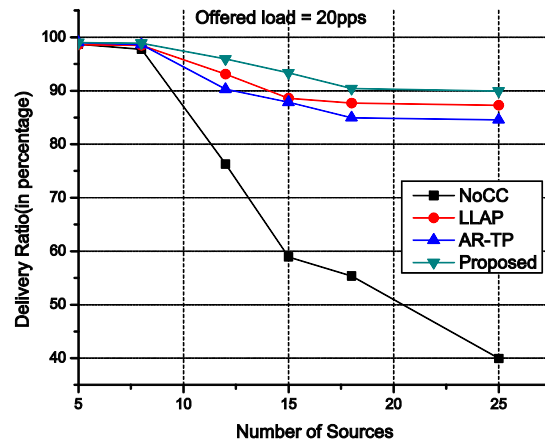


Figure 12. Packet delivery ratio for different number of sources

According to Figures 11 and 12, since NoCC, experiences huge amount of collision and buffer drops, the ultimate delivery ratio is very poor. AR-TP also exhibits poor delivery ratio as it sends explicit rate feedback notification as well as through forward threshold algorithm it induces large number of collisions. Our proposed protocol can achieve around 90% delivery ratio.

Figure 13 compares end-to-end delay for different protocols under diverse offered loads. In NoCC, the increase in end-to-end delay is due to the fact that, packet queuing delay and the contention delay in the path increases with increased data rates of the sources. In the case of LLAP, the ingress node pushes packets into the path with interval of FHD between consecutive packets. So, the increase in end-to-end delay is due to the packet queuing delay at the ingress node. The end-to-end delay saturates when the queue is full. AR-TP can reduce end-to-end delay significantly over NoCC and LLAP as it controls nodes transmission rate according to network load. In the proposed method sources maintains the traffic injection rate at a level so that the network works around the maximum throughput zone. Here the intermediate packet queuing and processing delay is minimized which ensures lower end-to-end delay.

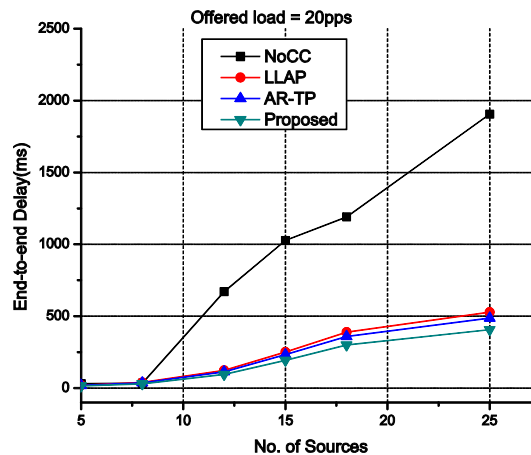


Figure 13. End-to-end delay for varying offered load

Figure 14 shows the delivery ratio and efficiency for different retransmission limits. From the graphs in the figure, it can easily be perceived that the higher number of retransmissions increases the delivery ratio but lowers the efficiency. Our proposed protocol can achieve more than 90% delivery ratio and 98% efficiency with retransmission limit 1.

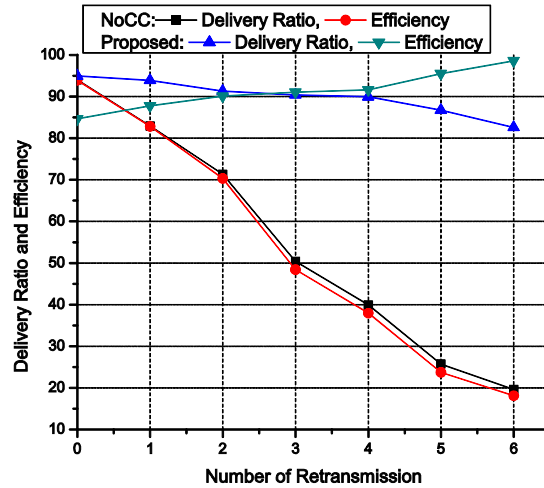


Figure 14. Delivery ratio and Efficiency for different retransmission limit

Figure 15 shows the delivery ratio and congestion degree for different packet loads. From the graph in the figure, it is apparent that with NoCC, under the higher traffic load, increased packet service time increases congestion degree and also lowers delivery ratio. When a node gets congested, the value of congestion degree promptly exceeds 1, at that point nodes start dropping off packets and more load increase in node decreases the delivery rate drastically. Our proposed method tries to limit the packet arrival rate below the packet service rate at a node, which eventually allows a node to push the entire received packet into the network without dropping them and hence maintain the delivery ratio. Figure 15 shows that for the proposed method, the delivery ratio is high as long the value of congestion degree is less than 1.

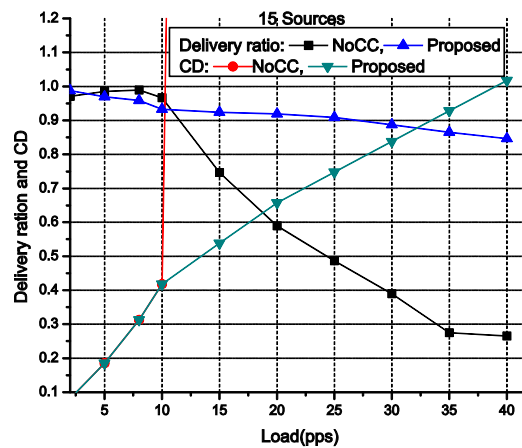


Figure 15. Delivery ratio and CD for various offered load

Our proposed protocol uses a traffic classifier which prioritizes real time traffic over non real time traffic. To evaluate the performance of the proposed traffic classifier, we assumed that each node has different traffic classes: RT and NRT. We evaluate the performance under WRR scheduler which has been discussed in Section 5.2.1. Since WRR scheduler ensures that packets with lower weights receive more network bandwidth than those with more weight in Figure 16, we can observe that the proposed model, can assign network bandwidth to each traffic class based on its weight. The RT class has higher throughput than NRT class. Since NoCC protocol cannot distinguish between traffic classes, its total throughput is plotted in Figure 16.

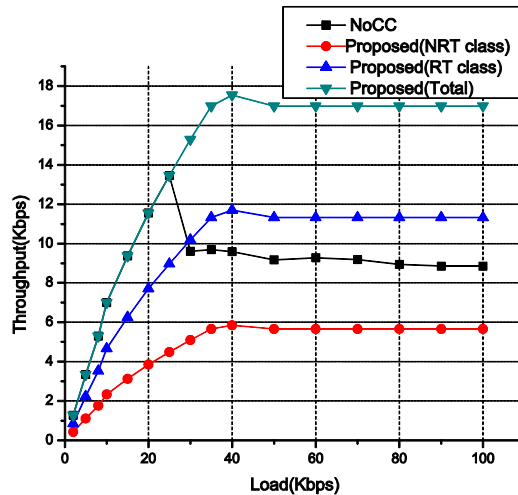


Figure 16. Throughput of NRT and RT traffic class

Figure 17, shows the end-to-end packet delay for both NoCC and proposed protocol. The RT class has the highest priority, so the end-to-end delay for this class is always lower than that of NRT class. The NoCC protocol uses a common buffer for all traffic classes, and hence cannot ensure low delay necessary for high priority RT traffic class. Thus, NoCC has the highest end-to-end delay.

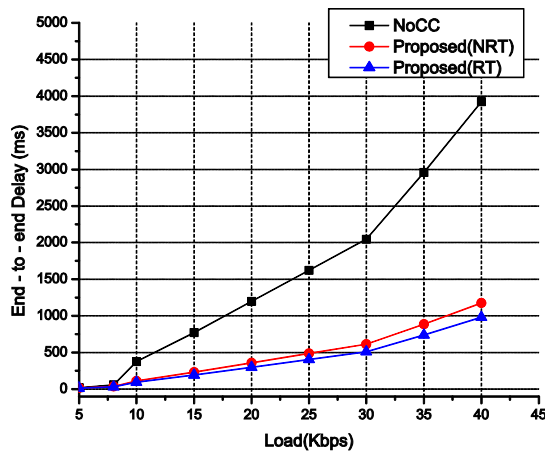


Figure 17. End-to-end delay of NRT and RT traffic class

## 7. Conclusions and Future Work

In this paper, we proposed a novel load adaptive and traffic priority sensitive congestion control technique for WMN. Our target was to maintain higher network throughput while avoiding congestion and granting prioritized treatment for real time traffic. The significant contributions to this paper are: i) the use of three elementary in-node parameters: packet input rate, packet service rate and buffer occupancy, to reflect the current congestion state of a node and its neighborhood, ii) restrict the rate of upstreams proportionately according their requirement and iii) putting emphasis on scheduling delay bound traffic. Through ns-2 simulations, we found that the employment of our proposed protocol greatly reduces the collision drops and avoids buffer drops, which in turn increases delivery ratio and network throughput for both real time and non real time class traffic.

Recent studies reveal that the use of multiple radios at each node can improve the bandwidth utilization and reduce neighborhood contention on WMN. Assignment of available channels to radio interfaces at the vicinity of contending neighborhood to mitigate congestion is a challenging issue. We plan to extend our work in multi radio multi channel (MRMC) environment where complex inter-flow and intra-flow interference among contending links limit the utilization of available network bandwidth. Based on the congestion and rate control



mechanism proposed in this paper, our future work focuses on developing an intelligent and dynamic channel assignment strategy to minimize congestion and maximize traffic flow rate.

## References

- Akyildiz, I. F., Wang, X., & Wang, W. (2005). Wireless mesh networks: a survey. *Computer Networks*, 47(4), 445-487. <http://dx.doi.org/10.1016/j.comnet.2004.12.001>
- Al Islam, A. A., & Raghunathan, V. (2011, March). End-to-end congestion control in wireless mesh networks using a neural network. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE* (pp. 677-682). IEEE.
- Alam, M. M., Islam, M. S., Hamid, M. A., Hong, C. S., & Lee, S. (2011). Congestion-aware fair rate control in wireless mesh networks. *Annals of Telecommunications-Annales des Telecommunications*, 66(5-6), 275-291. <http://dx.doi.org/10.1007/s12243-010-0199-6>
- Chandran, K., Raghunathan, S., Venkatesan, S., & Prakash, R. (2001). A feedback-based scheme for improving tcp performance in ad hoc wireless networks. *Personal Communications, IEEE* 8(1), 34-39. <http://dx.doi.org/10.1109/98.904897>
- Chen, K., Nahrstedt, K., & Vaidya, N. (2004). The utility of explicit rate-based flow control in mobile ad hoc networks. In *Wireless Communications and Networking Conference, 2004, WCNC* (vol. 3, pp. 1921-1926). 2004 IEEE. <http://dx.doi.org/10.1109/WCNC.2004.1311847>
- ElRakabawy, S. M., Klemm, A., & Lindemann, C. (2005). Tcp with adaptive pacing for multihop wireless networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing* (pp. 288-299). <http://dx.doi.org/10.1145/1062689.1062726>
- ElRakabawy, S. M., & Lindemann, C. (2009). Practical rate-based congestion control for wireless mesh networks. In *Kommunikation in Verteilten Systemen (KiVS)* (pp. 3-15). [http://dx.doi.org/10.1007/978-3-540-92666-5\\_1](http://dx.doi.org/10.1007/978-3-540-92666-5_1)
- Franklin, A. A., & Murthy, C. (2008). A link layer adaptive pacing scheme for improving throughput of transport protocols in wireless mesh networks. *Computer Networks*, 52(8), 1583-1602. <http://dx.doi.org/10.1016/j.comnet.2008.02.001>
- Holland, G., & Vaidya, N. (2002). Analysis of tcp performance over mobile ad hoc networks. *Wireless Networks* 8(2-3), 275-288. <http://dx.doi.org/10.1023/A:1013798127590>
- Information Sciences Institute. (2008). *ns-2 network simulator*. Software package. Retrieved from <http://www.isi.edu/nsnam/ns/>
- Jamshaid, K., Shihada, B., Showail, A., & Levis, P. (2014). *Deflating Link Buffers in a Wireless Mesh Network*. Ad Hoc Networks. <http://dx.doi.org/10.1016/j.adhoc.2014.01.002>
- Jamshaid, K., Ward, P., Karsten, M., & Shihada, B. (2013). The efficacy of centralized flow rate control in 802.11-based wireless mesh networks. *EURASIP Journal on Wireless Communications and Networking*, 2013(1), 1-17.
- Kim, D., Toh, C., & Choi, Y. (2001). Tcp-bus: Improving tcp performance in wireless ad hoc networks. *Journal of Communications and Networks*, 3(2), 175-186.
- Liu, C., Shen, F., & Sun, M. T. (2007). A unified tcp enhancement for wireless mesh networks. In *Parallel Processing Workshops, 2007* (pp. 71-71). ICPPW 2007
- Liu, J., & Singh, S. (2001). Atcp: Tcp for mobile ad hoc networks. *Selected Areas in Communications, IEEE Journal on*, 19(7), 1300-1315. <http://dx.doi.org/10.1109/49.932698>
- Masri, A. E., Sardouk, A., Khoukhi, L., & Gaiti, D. (2012). An efficient and fair congestion control protocol for IEEE 802.11-based Wireless Mesh Networks. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on* (pp. 738-744). IEEE.
- Pace, P., Natalizio, E., & Gungor, V. C. (2007). An adaptive and responsive transport protocol for wireless mesh networks. In *Proceedings of IEEE ICC*.
- Rangwala, S., Jindal, A., Jang, K.Y., Psounis, K., & Govindan, R. (2008). Understanding congestion control in multi-hop wireless mesh networks. In *Proceedings of the 14th ACM international conference on Mobile computing and networking* (pp. 291-302). <http://dx.doi.org/10.1145/1409944.1409978>
- Raniwala, A., Sharma, S., De, P., Krishnan, R., & Chiueh, T. C. (2007). Evaluation of a stateful transport

- protocol for multi-channel wireless mesh networks. In *Quality of Service, 2007 Fifteenth IEEE International Workshop on* (pp. 74-82).
- Sadeghi, B., Yamada, A., Fujiwara, A., & Yang, L. (2006). A simple and efficient hop-by-hop congestion control protocol for wireless mesh networks. In *Proceedings of the 2<sup>nd</sup> annual international workshop on Wireless internet* (p. 4). <http://dx.doi.org/10.1145/1234161.1234165>
- Scheuermann, B., Lochert, C., & Mauve, M. (2008). Implicit hop-by-hop congestion control in wireless multihop networks. *Ad Hoc Networks*, 6(2), 260-286. <http://dx.doi.org/10.1016/j.adhoc.2007.01.001>
- Sundaresan, K., Anantharaman, V., Hsieh, H. Y., & Sivakumar, R. (2003). Atp: a reliable transport protocol for ad-hoc networks. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing* (pp. 64-75).
- Tan, K., Jiang, F., Zhang, Q., & Shen, X. (2007). Congestion control in multihop wireless networks. *Vehicular Technology, IEEE Transactions on*, 56(2), 863-873. <http://dx.doi.org/10.1109/TVT.2007.891405>
- Venkata Ramana, B., Manoj, B., & Siva Ram Murthy, C. (2006). Ar-tcp: a loss-aware adaptive rate based tcp for ad hoc wireless networks. *Journal of High Speed Networks*, 15(1), 53-72.
- Zhai, H., Chen, X., & Fang, Y. (2005). Rate-based transport control for mobile ad hoc networks. In *Wireless Communications and Networking Conference, 2005 IEEE* (vol. 4, pp. 2264-2269). <http://dx.doi.org/10.1109/WCNC.2005.1424868>
- Zimmermann, A., Schaffrath, D., Faber, M., Wenig, M., & Gunes, M. (2007). Improving tcp performance through explicit corruption and route failure notification (ecrfn). In *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems* (pp. 284-288).

### Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).