Volume 3, Issue 10, October 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Strategy for Congestion Control Scheme in Busted Network Using Reactive Protocols (AODV, AOMDV & DSR)

Ranjana Kumari, Arun Kumar Singh

Dept. of Computer Science &

Rajasthan Technical University, India

Abstract— ***Ad-hoc network are self-configurable, easy, flexible and fast to deploy. Congestion is a common problem in computer network so there is a growing need to control the congestion in the computer network. We can say congestion as the loss of utility to a network user due to high traffic loads. Congestion control mechanisms can be used to maximize a user's utility at high traffic. Proposed work in this paper considers the problem of protecting well behaved users from congestion caused by ill behaved users by allocating all users a fair share of the entire network bandwidth. Fairness can be achieved when equal numbers of packets are received from each node and this will be achieved by restricting the queue size and limited bandwidth. This aggregate queue orders packets based on their timestamps rather than order of arrivals. Through ns2 simulator, we show the performance of DSR, AODV and AOMDV protocols.***

Keywords— ***Congestion, congestion control mechanism, Fairness, routing protocols, FIFO***

I. INTRODUCTION

A. Congestion

When the number of packets increases beyond the limit that can be handled by the network resources, the network performance degrades, and this situation is known as congestion. Congestion means blockage due to overloading .It is similar to traffic jam caused by many cars on a narrow road.

B. Congestion control

Two styles of control, proactive and reactive control, are presented. It is shown that congestion control must happen at several different time scales.

A. Proactive and reactive control

Congestion is the loss of utility to a user due to an increase in the network load. Hence congestion control is defined to be the set of mechanisms that prevent or reduce such deterioration. Practically speaking, a network can be said to control congestion if it provides each user with mechanisms to specify and obtain utility from the network. For example, if some user desires low queuing delays, then the system should provide a mechanism that allows the user to achieve this objective. If the network is unable to prevent a loss of utility to a user, then it should try to limit the loss to the extent possible, and, further, it should try to be fair to all the affected parties. Thus, in reservation less networks, where a loss of utility at high loads is unavoidable, we are concerned not only with the extent to which utility is lost, but also the degree to which the loss of utility is fairly distributed to the affected users. A network can provide utility in one of two ways. First, it can request that each user specify a performance requirement, and can reserve resources so that this level of performance is always available to the user. This is proactive or reservation-oriented congestion control. Alternatively, users can be allowed to send data without reserving resources, but with the possibility that, if the network is heavily loaded, they may receive low utility from the network. The second method is applicable in reservation less networks. In this case, users must adapt to changes in the network state, and congestion control refers to ways in which a network can allow users to detect changes in network state, and corresponding mechanisms that adapt the user's flow to changes in this state. In a proactive scheme, the congestion control mechanism is to make reservations of network resources so that resource availability is deterministically guaranteed to admitted conversations. In a reactive scheme the owners of conversations need to monitor and react to changes in network state to avert congestion.

II. RELATED WORK

Although many congestion control algorithm have been proposed in the literature for wireless network, the design of the algorithms are challenged by having to support different services levels, fairness and implementation complexity. Many researchers have compared their proposal schemes on different congestion control schemes, but there is no common, simple and standardized congestion control scheme to make their comparisons with Addisu Eshete and Yuming Jiang[7] This paper presents S-SFQ which is a single queue design and implementation of the well-known Start-time Fair Queueing (SFQ). This aggregate queue orders packets based on their timestamps rather than order of arrivals. SFQ can fairly approximate the fairness of per flow queues. In addition, the author discusses in detail the adverse effect of packet loss synchronization problem common in such aggregate queues. First, the author provides a design and implementation of SFQ based on a single shared queue, rather than a stack of queues. This in turn calls for a non-FIFO queue that can sort packets based on the timestamps. This is because all flows share the queue and the policy for flow scheduling must be different from the order of the packet arrivals. When packets are arrived, packets are assigned start tags computed by the underlying SFQ algorithm. The tag is used to determine the packet position in the queue and its transmission priority. The second important contribution is that author identify and discuss a recurring problem in the context of this work called loss synchronization that has a potential to obliterate the desirable qualities of the underlying scheme. We find that this problem commonly arises in shared queues with no specialized (randomized) buffer management. Thompson and Nagle [10] represent the two points of Implementation in congestion control. The first is the source, where flow control algorithms vary the rate at which the source sends packets .Of course flow control algorithms are designed primarily to ensure the presence of free buffers at the destination hosts but we are more concerned with their role in limiting the overall network traffic the second point of implementation is at the Gateway .congestion can be controlled at gateways through routing and queuing algorithms. Queuing algorithms, which control the order in which packets are sent and usage of gateway's buffer space, do not effect congestion in direct manner, in that they do not change the total traffic on the gateways outgoing line. Queuing algorithm can be thought of as allocating three nearly independent quandaries' bandwidth(which packet get transmitted),promptness (when do these packets get transmitted),and buffer space(which and when packet get discarded by the gateway).Nagle proposed a Fair Queuing algorithm in which Gateway maintain separate queue for packets from each individual.

III. RESULTS AND ANALYSIS

The objective of our dissertation is to design a set of congestion control mechanisms in wireless network. This evaluation will be done through simulation on various network parameter such as varying queue length and number of sender increased. Check the performance of congestion control mechanisms and how mechanism behaves when we increase number of sender and usages.

Performance Metrics

In evaluating a MANET routing protocol different statistics or performance metrics are used. In this subsection, we discuss the essential metrics required to evaluate performance.

a) Network throughput

b) End to end delay

Network throughput

A network throughput is the average rate at which message is successfully delivered between a receiver (destination node) and its sender (source node). It is also referred to as the ratio of the amount of data received from its sender to the time the last packet reaches its destination. Throughput can be measured as bits per second (bps), packets per second or packet per time slot. For a network, it is required that the throughput is at high-level. Some factors that affect MANET's throughput are unreliable communication, changes in topology, limited energy and bandwidth.

End-to-end delay

Packet end-to-end delay is the time delay it takes a network source to deliver a packet to its destination. Thus, the end-to-end delay of packets is the total amount of delays encountered in the whole network at every hop going to its destination. In MANETs, this kind of delay is usually caused by certain connection tearing or/and the signal strength among nodes been low, or some congestion. The reliability of a routing protocol can be determined by its end-to-end delay on a network.

Packet delivery ratio

This refers to the ratio of the total number of data packets that reach the receiver (destination node) to the total number of data packets sent by the source node. This is another performance metric that is used to determine the efficiency and accuracy of MANET's routing protocol because it is used to calculate the rate of loosing packets. Similar to the network throughput, packet delivery ratio (PDR) is expected to be high.

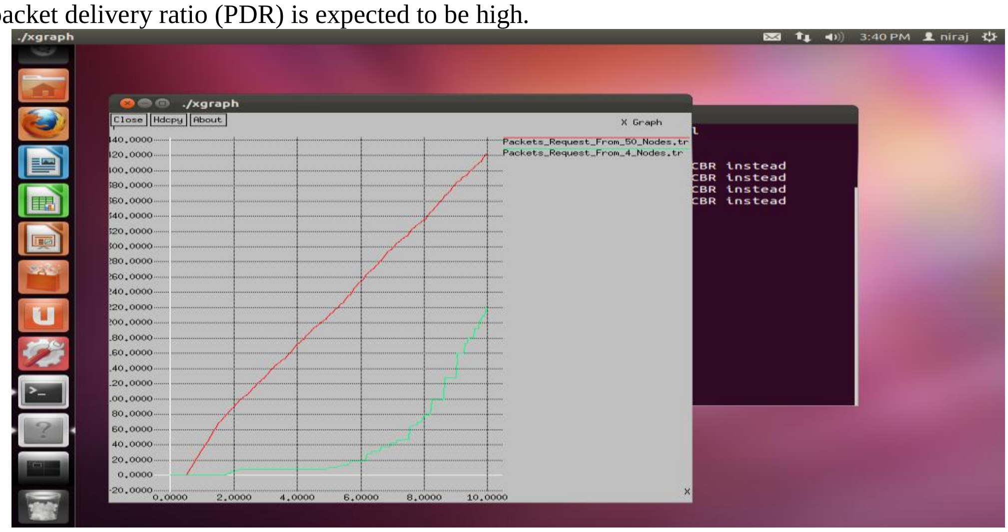

Figure 3.a performance of congestion control mechanism

Here no. of packets per sender is 70 packets, and maximum allowed senders are 4. There- for maximum no. of packets that can be buffered in queue is 280. Figure 3.a shows the simulation results. Here red line shows buffering request from 50 senders is approaching to maximum size of approximately 420 packets. This is because of no congestion control mechanism is applied. Green line shows buffering request from 4 senders is approaching to maximum size of approximately 220 packets, which is less compared to buffering capacity of queue (280 packets in our case). The result shows a mechanism to count active flows, target queue length and drop rate to enforce the targets on a FIFO queue.

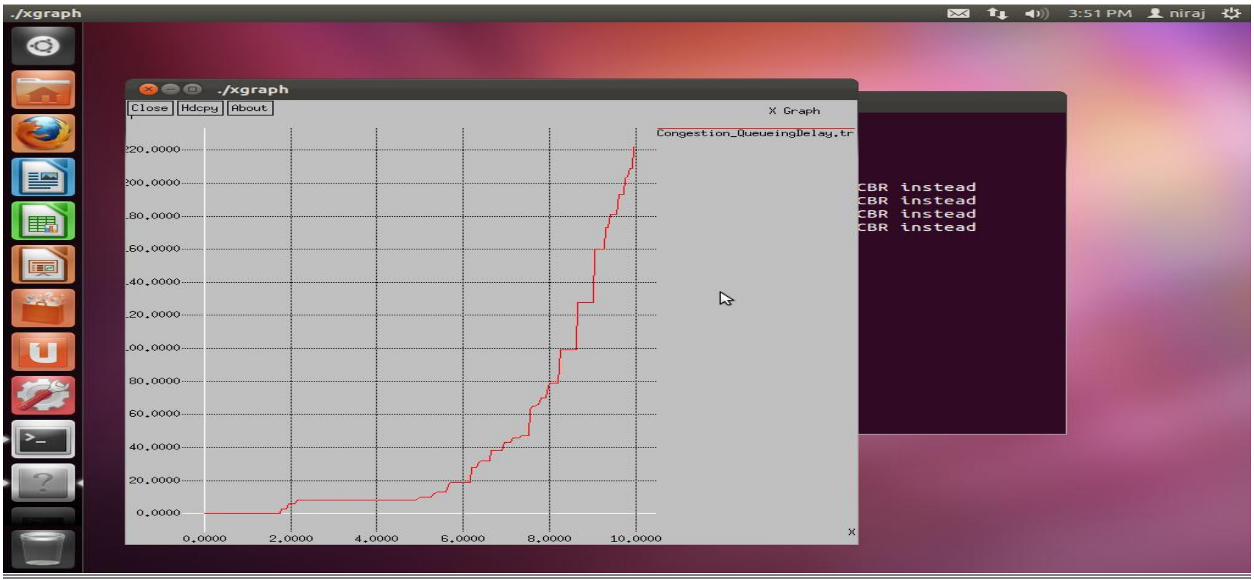

Figure 3.b Queuing Delay

Figure 3.b shows result of Queuing delay Maximum goes to 220 packets, if no congestion control mechanism is applied. When all the 50 nodes are allowed to send packets, due to fixed buffering capacity of the queue (280 packets) remaining 220 packets can't be enforced on queue. This buffering delay shows congested network.

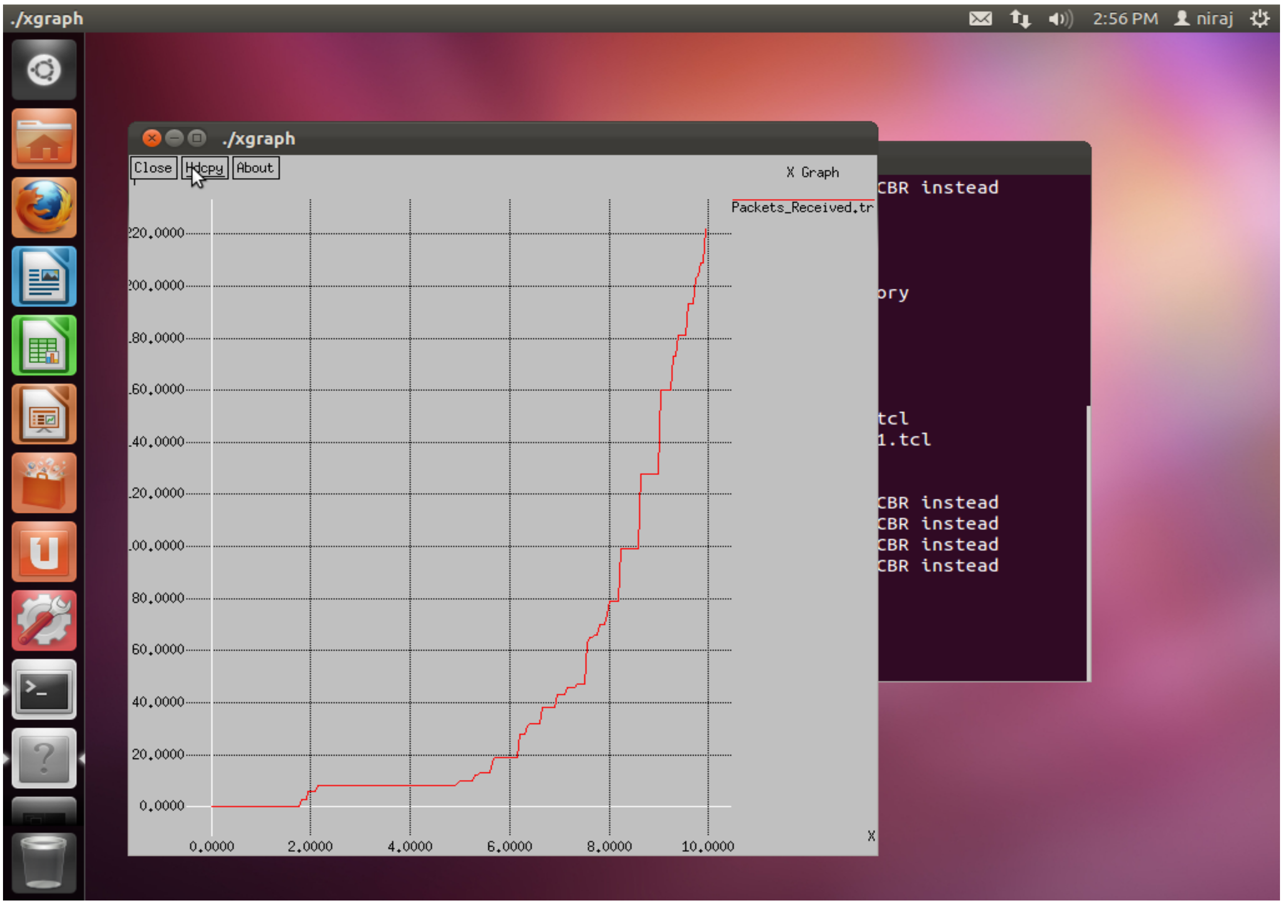

Figure 3.c Packets Received With congestion control mechanism

IV. CONCLUSION

In this paper three reactive protocols DSR, AOMDV and AODV were studied. The performance evaluation parameters were Packets Loss, Throughput and Packet Delivery Ratio by varying queue length and No. of active senders. Congestion depends on the number of active flows and the total storage in the network. Total storage includes router buffer memory and packets in flight on long links. In this paper, a simple flow counting algorithm is presented. The algorithm provides control over congestion by varying the number of packets per sender in proportion to the length of queue. When the packets arrived, packets are assigned timestamp, this timestamp determines the packet position in queue and transmission policy and reduces the packet drop rate. The comparison between DSR, AOMDV and AODV have been shown with the help of some graphs taken under various performance metrics. In all the three protocols fairness is achieved when equal numbers of packets are received from each node. The current works has been limited with fixed no of active senders and fix simulation area with CBR traffic.

REFERENCES

[1] CHONG LIU, YANJUAN ZHENG,”*Analysis and Research on the Traditional Congestion Control Policy and Active Networks Congestion Control Policy”*

[2] Carlo Tarantola, *“Dynamic Active Networks Services”*, Proceedings of the 2004 IEEE International Conference On Mobile Data Management, pp. 46-47, June 2004.

[3] A. Kortebi, S. Oueslati, and J. Roberts,*“Cross-protect: implicit service differentiation and admission control”,* in High Performance Switching and Routing, 2004, pp. 56–60.

[4] Elizabeth M. Belding-Royer Charles Perkins,*”Ad hoc on demand distance vector (aodv)routing”.* IETF Internet Draft, 2003.

[5] David Maltz David Johnson.*” Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks”*. Ad Hoc Networking, pages 139–172, 2001.

[6] Mahesh K. Mariha, Samir R. Das,et al. *“On-demand Multipath Distance Vector Routing in Ad-Hoc Networks”*. IEEE Proceedings of the Ninth International Conference on Network Protocols Page: 14, ISSN:1092 1658,2001

[7] On the Flow Fairness of Aggregate Queues Addisu Eshete and Yuming Jiang Center for Quantifiable Quality of Service in Communication Systems (Q2S)Norwegian University of Science and Technology,Trondheim, Norwayaddisu.eshete@q2s.ntnu.no ymjiang@ieee.org

[8] Van Jacobson, *“Modified TCP Congestion Control Avoidance Algorithm*”. end-2-end-interest mailing list, April 30, 1990.

[9] W. Stevens.” *TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms,*” January 1997, RFC 2001.

[10] Thompson and Nagle’s,”*Analysis andsimulation of Fair Queuing Algorithm”* published in internetworking :Research and Experience,vol. 1,3-26,1990

