ISSN:0975-9646

Urvashi Handa et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 7 (1) , 2016, 56-59

A Study on Improved Hop by Hop Congestion Control Algorithm

Urvashi Handa[1], Padma Bonde[2]

[1,2]*Computer Technology and Application,*
Chhattisgarh Swami Vivekanand Technical University, India

***Abstract*—As the development of internet continues, congestion control has become an important issue for computer network society. It has become very important to detect and control congestion in order to improve the performance of network. This paper focuses on hop by hop congestion control technique. The proposed method is efficient in decreasing the rate of congestion. Each router in the network keeps track at its queue. Based on the queue length the router effectively performs necessary action to control congestion. The experimental result show that the proposed algorithm can achieve short end to end delay, greater packet delivery ratio and greater throughput.**

***Keywords*— Congestion detection, congestion control techniques, hop by hop congestion control, AODV.**

I. INTRODUCTION

Nowadays, our capabilities to get multimedia information from the internet have been increasing rapidly. Real-time multimedia applications have been introduced which leads to problem of quality degradation and packet loss. As the growth of internet and its traffic continue, it is very important to detect and control congestion. Network Congestion occurs when demand for network resources is greater than the available resources. There are many network resources such as buffers, link bandwidth, processor times, servers, destinations etc.

Today to avoid congestion, computer networks use congestion avoidance and congestion control techniques. Controlling and detecting of network congestion is important to provide high transmission rate with high efficiency and reliability.

The main aim of any network is to detect the congestion in the network and successfully overcome it. Congestion detection and its control are very important for a reliable and efficient network. There exist many congestion control techniques but search for new techniques continue. One such congestion control techniques Hop-by-hop congestion control mechanism is discussed here.

Hop-by-hop congestion control mechanism (sometimes also called *backpressure*), provide feedback about congestion state at a node to the preceding hop. Feedback provided is based on the queue length at the congested node. The preceding node then adapts its transmission rate based on this feedback. If the queue length at the node exceeds a threshold, congestion is indicated and the preceding node is notified to decrease its transmission rate. It is found that such technique react to congestion faster than end to end technique which thus decreases the delay and gives better performance. There are many routing protocols used. The one used for this project is AODV i.e., Ad hoc On-Demand Distance Vector Routing. It is a reactive type of congestion control scheme. In AODV route is calculated on demand. AODV maintains routing table where it maintains one entry for each destination. By using destination sequence number it avoids the problem of counting to infinity. This makes AODV to be loop free. It provides both kinds of communication i.e., unicast and multicast. It discover routes when as and when required, this eliminates the need for maintaining routes which are not used and also reduces the number of active routes between active source and destination. AODV uses routing table to store routing information. When a node wants to send a packet to some destination, its checks the routing table to determine whether the path to the destination exist. If route to the destination exist it will send the packet to destination through that route. Otherwise it uses route discovery process to find out the route for packet transfer. There are 3 types of message for route discovery process: RREQs (Route Request), RREPs (Route Reply), and RERRs (Route Error). Route discovery process starts with RREQ packet generated by the source. The packet contains broadcast ID, destination IP address, destination sequence number, source sequence number, hop count. Broadcast Id increment each time the source uses RREQ packet. Broadcast ID and source IP address are unique for each RREQ packet. RREP messages are used to finalize the route. Discovered routes are used to send data packets.

Broadcasting is done using flooding. In flooding sender broadcasts the control packet to all its neighbours. Each node that receives the control packet forwards the control packets to its neighbours. Sequence number in control packet helps to avoid forwarding the same packet twice. As the control packet reaches the destination it doesn't forward the packet further. As the route from source to destination gets finalised, the packets to be send to destination is transmitted through that route.

The proposed methodology takes necessary action before occurrence of congestion based on the queue length at the node.

II. Related Works

In [1], a hop-by-hop cross-layer congestion control (HCCC) scheme has been presented. HCCC detects local congestion at proper moments and delivers the congestion information to upstream nodes by exploiting the transmission of RTS and CTS frames. Meanwhile, it adapts the channel access priorities and data transmission rates of sensor nodes. Thus it can adaptively adjust the allocation of channel resource among sensor nodes.

In [2], congestion is detected by measuring the queue length. The c1ongestion is controlled by using three techniques i) hop-by-hop flow control , ii) source rate limiting, and iii) prioritized MAC. Even in high offered load it claims to achieve good throughput and fairness. In [3], a congestion control technique in which packet service time is used to infer the available service rate and therefore detects congestion in each intermediate sensor node. The congestion is controlled by hop-by-hop technique and it uses rate adjustment based on the available service rate and number of child nodes. However, it cannot utilize the available link capacity efficiently when some nodes are in sleep state. An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

In [4] the author proposed a metrics called Depth of Congestion (DC) to detect congestion. The Depth of Congestion (DC) is given as input to the congestion detection unit. When the measured DC is less than pre-defined threshold value, then, there is no congestion whereas when DC is equal to or more than pre-defined threshold value, then it intimates the occurrence of congestion. Once congestion is detected, notification signal is send to all the nodes to control the congestion. On receiving the notification signal each node adjust the transmission rate by implementing Hop-by-Hop Rate Control Technique.

The methodology in [5] has given a efficient method for both congestion detection and avoidance, the authors gets the notification of congestion by taking a term Intelligent Congestion Detection (ICD), the packet service time and inter arrival time is taken into account for the detection. When the above method ends in the congestion phase a notification to all the nearby nodes is been provided by the method called implicit congestion notification (ICN). The avoidance of congestion is been made out by assigning (PRA) priority by taking the packet delivery rate as key.

In [6] Present a congestion avoidance protocol, which includes *source count* based hierarchical medium access control (HMAC) and weighted round robin forwarding (WRRF) as the two main method to overcome congestion . The Simulation result of the this technique avoid packet drop due to buffer overflow and achieves much higher delivery ratio even under high traffic condition, which claims a good enough method for reliable event detection.

III. SYSTEM OVERVIEW

The main motivation behind this methodology is to present an effective and efficient mechanism to detect and control congestion. In the proposed method, each internet router maintains a queue. Every time a new packet arrives at the router, it keeps track of the number of packets in the queue. The router will try to send the packets to their desired destination till the number of packets in the queue is between the parameters “min” and “max”. If the number of packets is beyond a parameter “min”, then router or the intermediate node asks the source to slow down the transmission. If the number of packets is beyond a parameter “max”, then the node won’t queue any packet onwards and it simply asks the source to stop sending packet and try for new route. The sender then broadcasts RREQ packet to its neighbours in search for the new route. As the sender gets RREP packet the sender then starts transmitting the packets through the new route found.

A. *Algorithm Description of MOD-AODV*

```
Set Queue length = Q
Set min = (2/3 of Q)
    max =(4/5 of Q)
While
        Packets are being transmitted in the network &
received at the nodes.
do
        [Compare No. of packets in the queue with ‘ min’
& ‘ max’]

        if (N ≥ min)
        then
                [Packet transmission rate is greater than
                packet receiving rate]

                checkToSend() i.e., Node will ask the
                sender to slow down the transmission.

        elseif (N ≥ max)
        then
                [Packet transmission rate is much greater
                than packet receiving rate]

                Node asks the sender to stop sending
                packets and try for new route.
        else
                No congestion
        End if
End while
```

B. *Example*

Consider following network that uses Modified AODV as an example. S is source and D is destination and the remaining nodes are intermediate nodes. Source S is sending packets to destination D through intermediate node B. The intermediate node keeps monitoring its queue. As the queue length at node B goes beyond max parameter is $4/5^{th}$ of the queue length, it asks the sender to stop sending packets and try for new route. The sender then broadcasts RREQ message to all its neighbours i.e., nodes A, B, C. The node that is ready to send the packet to destination replies with RREP message. Suppose in our example let us say its node A. After getting RREP message the sender S starts sending the packets to the destination through that new route.

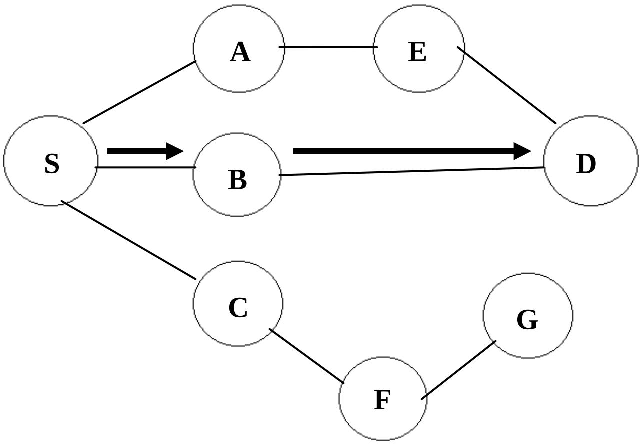

Fig. 1 Data flow form Sender S to Destination D

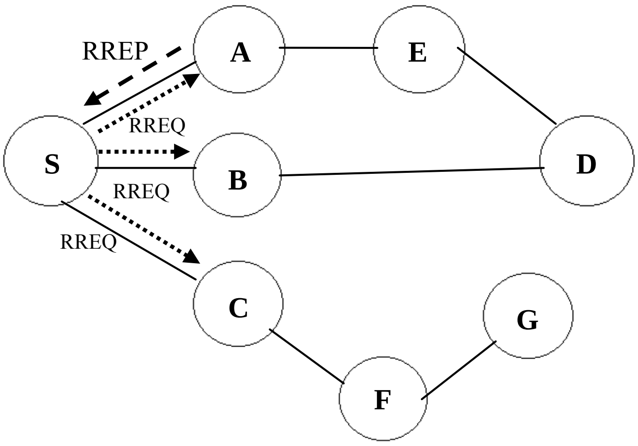

Fig. 2 Searching for new route as queue length at B exceeded 4/5th of queue.

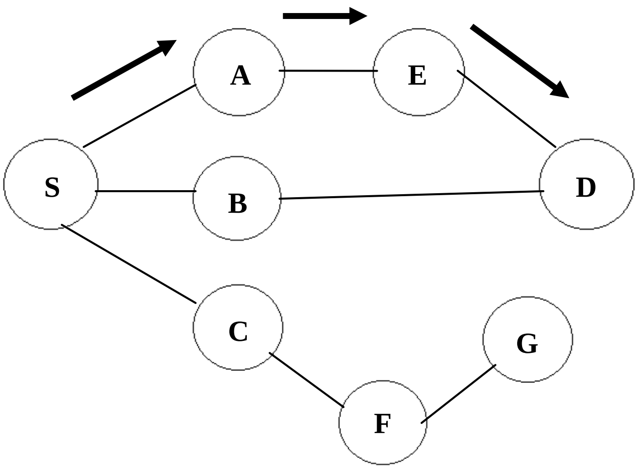

Fig. 3 New path for data transfer.

C. Simulation Results

The main aim of this section is to demonstrate how the proposed methodology i.e., Improved Hop by Hop congestion control is better than the existing protocols. Routing protocol used for the project is AODV. We present the comparison between AODV and Modified AODV protocols. The performance of MOD-AODV is analysed in NS2. Comparison between AODV and MOD AODV is done on the basis of average end to end delay, throughput, packet delivery ratio and normalized routing load. The packet is generated at the source and transmitted to other nodes till it reaches the destination. As the congestion is detected, transmission rate is adjusted to control congestion.

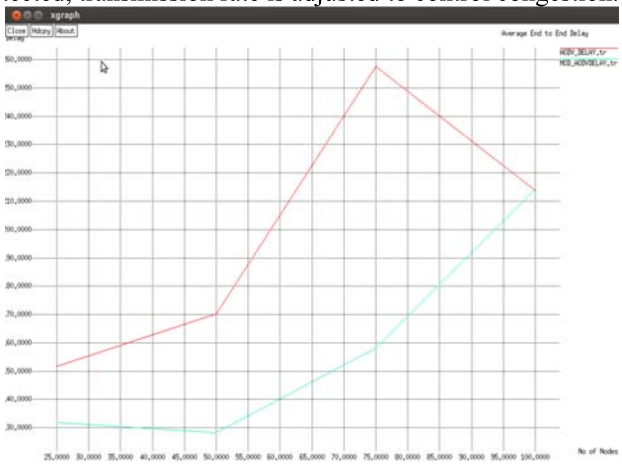

Fig. 4 Average End-to-End Delay

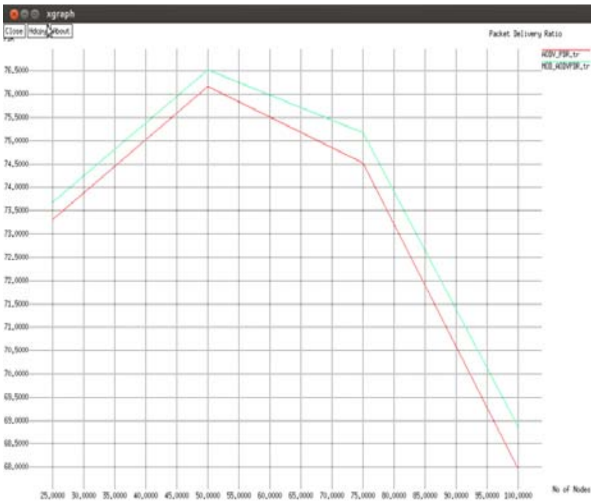

Fig. 5 Packet Delivery Ratio

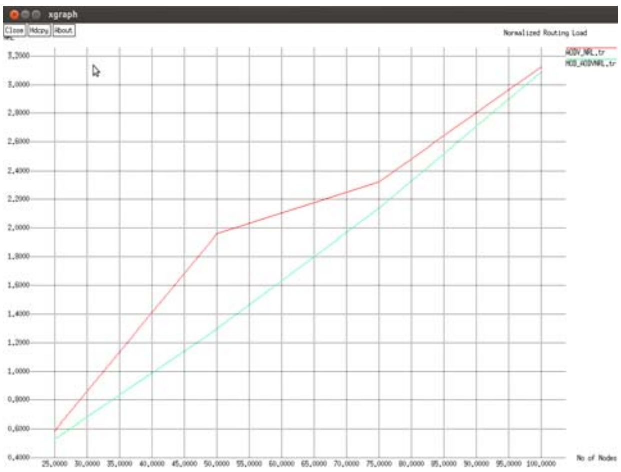

Fig. 6 Routing Load

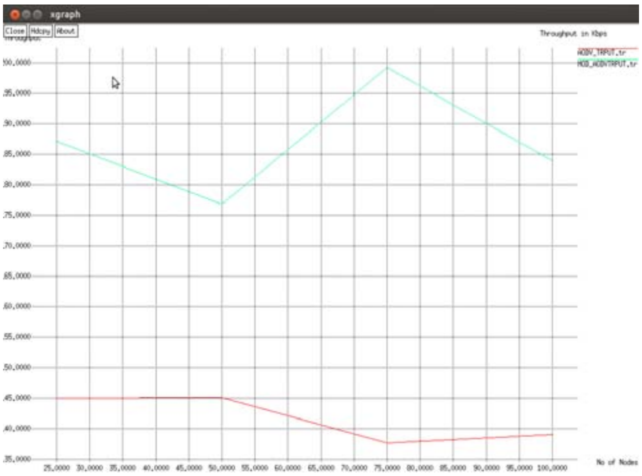

Fig. 7 Throughput

Fig. 4 shows the average end to end delay. It is the average time taken by the data packet to reach the destination, delay caused due to route discovery process and queue in data packet transmission. Figure demonstrates that end to end delay in AODV is more than the proposed MOD-AODV.

Average end to end delay = ∑ (Arrival time-Send time) / ∑ Number of connections

Fig. 5 shows that the packet delivery ratio of MOD-AODV is higher than AODV.

Packet Delivery Ratio= Number of packets received / Number of packets send

Fig. 6 shows routing load. Routing load is number of packets transmitted per data packet send to destination.

Fig. 7 shows the throughput. Throughput is the amount of data received at the destination per unit time. Graph demonstrates that the throughput of MOD-AODV is better than AODV.

IV. Conclusions

Congestion detection and control is an important issue in today's internet world. The proposed methodology helps in obtaining a network with reduced transmission delay, higher throughput, higher packet delivery ratio and less packet loss.

References

[1] G. Wu, F. Xia, Y Zhang and Y. Zhu, "A Hop-by-hop Cross-layer Congestion Control Scheme for Wireless Sensor Networks", Journal of software, Vol 6, No. 12 (2011), 2434-2440, Dec 2011.

[2] B. Hull, K. Jamieson, and Balakrishnan, "H. Mitigating Congestion in Wireless Sensor Networks", In Proceedings ofACM SenSys, pp. 134-147, 3-5 November 2004.

[3] C.T. Ee and R. Bajcsy, "Congestion Control and Fairness for Many-to-One Routing in Sensor Networks", In Proceedings of the ACM SenSys, pp. 148-161, 3-5 November 2004.

[4] R. Chakravarthi and C. Gomathy "*Hop-by-Hop Rate Control Technique for Congestion Due to Concurrent Transmission in Wireless Sensor Network",* World of Computer Science and Information Technology Journal(WCSIT), ISSN: 2221-0741, Vol. 1, No. 8, pp. 351-356, 2011.

[5] C.Y. Wan, S. B. Eisenman, A. T. Campbell, and J. Crowcroft, "Overload Traffic Management for Sensor Networks", ACM Transactions on Sensor Networks, Vol. 3, No. 4, Article 18, October 2007.

[6] Y. Sankarasubramaniam, A. Ozgur, I. Akyildiz, "ESRT Event-to-Sink Reliable Transport in wireless sensor networks", In the proceedings of ACM Mobihoc, pp. 177–189. ACM Press, New York (2003)

[7] M. Zawodniok and S. Jagannathan "Predictive Congestion Control Protocol for Wireless Sensor Networks", IEEE Transactions on Wireless Communications, Vol. 6, No. 11, pp. 3955-3963, 2007.

[8] M. Welzl, *Network congestion control: managing Internet traffic,* John Wiley & Sons Ltd, 1973.

[9] K. Sharma, Dr. H. Singh and Dr. R.B. Patel, "A Hop by Hop Congestion Control Protocol to Mitigate Traffic Contention in Wireless Sensor Networks", International Journal of Computer Theory and Engineering, Vol. 2, No. 6, pp. 986-991, December 2010.

[10] Y. Yi and S. Shakkottai, "Hop-by-hop Congestion Control over a Wireless Multi-hop Network", Proc. IEEE Infocom, Vol. 4, pp. 2548-2558, March 2004.

[11] K. Tan, F. Jiang, Q. Zhang and X. Shen, "Congestion Control in Multihop Wireless Networks", IEEE Transaction on Wireless Communications, Vol. 56, Issue. 2, pp. 863 – 873, 2007.

[12] Y. Hsu and K. Feng, "Cross-Layer Routing for Congestion Control in Wireless Sensor Networks", Radio and Wireless Symposium, pp. 783-786.

[13] C. Wang, Y. Eisenman and A. Campbell, "CODA: Congestion Detection and Avoidance in Sensor Network, Proc. of ACM SenSys, pp. 1003-1016.

[14] J. Wan, X. Xu, R. Feng and Y. Wu, "Cross Layer Active Predictive Congestion Control Protocol for Wireless Sensor Networks", Sensors 2009, Vol. 9, No. 10, pp. 8278-8310, 2009.

[15] Y. Yi and S. Shakkottai, "Hop-by-hop Congestion Control over a Wireless Multi-hop Network", Proc. IEEE Infocom, Vol. 4, pp. 2548-2558, March 2004.

[16] R. Jain, "Congestion Control in Computer Networks: Issues and Trends", IEEE Network Magzine, pp. 24-30, 1990.

[17] R. Banner and A. Orda, "Multipath Routing Algorithm for Congestion Minimization". IEEE/ACM Transaction on Networking, Vol. 15, No. 2, pp. 413-424, 2007.

[18] ELEC3030 (EL336) Computer Networks S Chen, Available:users.ecs.soton.ac.uk/sqc/EL336/CNL-11.pdf.

[19] Congestion Control in Computer Networks – EECS Technical Reports, Available:techreports.lib.berkeley.edu/accessPages/CSD-91-649.html.