INTERNATIONAL JOURNAL OF SCIENTIFIC PROGRESS AND RESEARCH (IJSPR) ISSN: 2349-4689
Volume-24, Number - 01, 2016

Improving Network Performance using Enhanced Adaptive Virtual Queue

Shikha Tiwari, Prof. Sanjay Gupta

Computer Science & Engineering Department

Vindhya Institute of Technology and Science, Jabalpur

Abstract ***- Mobile ad hoc network facing multiple challenges such as dynamic configuration, energy consumption and network congestion. Congestion within network occurs when the demand of network resources is greater than the available resources. Adaptive Virtual Queue (AVQ) technique which incurs low-delay, few loss and high link utilization at the link. This paper proposed a new named enhanced Adaptive Virtual Queue (EAVQ) for detection and protection from congestion. Proposed mechanism EAVQ improve the network performance such as higher Packet Delivery Ratio, Throughput and low packet loss rate. Simulation results in performed in NS-2.34 which shows that Enhanced Adaptive Virtual Queue over perform than existing AVQ and RED.***

Keywords: Adaptive Virtual Queue, Average Queue Size, Packet Delivery Ratio, Throughput.

I. INTRODUCTION

A mobile ad hoc network (MANET) [1] can be a wireless network without requirement of any central controlling stations. MANETs can be applied in any medical emergencies or military applications, natural disaster or conduct geographic exploration. Mobile ad hoc wireless devices happiness to MANET devices referred to as mobile nodes, such mobile nodes are being characterized through high mobility; restricted storage space, low power, and restricted transmission range for communication. Mobile nodes communicate through bi-directional wireless links thus securing transmission could be a key challenge. MANET communication events are being referred to as sessions in the two communication nodes; particularly the pair of source node and the destination node comprises a session (or Source, Destination pair). A mobile node can directly communicate with different nodes if any such a link exists between intervals of their radio transmission range. If the gap between sessions is simply too apart to ascertain direct contact, then the information should be sent via intermediate nodes that connect the two parties. A minimum of one valid routing path must be established before the source node will send packets to its destination node.

Active Queue Management (AQM)[2][3] is an algorithm that detects and reacts to incipient congestion to avoid queue from overflows.

There are in general two ways to detect congestion: first it can give congestion signal to traffic sources explicitly by setting Explicit Congestion Notification (ECN) bits, second it can be congestion signal to traffic sources implicitly by dropping packets.

II. ACTIVE QUEUE MANAGEMENT

Active Queue Management [3] is router based techniques which bring improvement over Drop-Tail queue with respect to fairness and delay. It helps to adopt the dynamic behavior of network. Objective of AQM is to notify the sender about congestion before the queue becomes full. In order to control congestion, the router has to allocate resource efficiently. The source is being intimated as a feedback either by dropping the packets or marking the packet by setting ECN bit in the IP header. There are many active queue management schemes has been proposed in the past. In this we considered mainly three AQM techniques namely RED, NLRED and REM which is discussed below.

A. Random Early Detection

Random Early Detection (RED) is most widely used AQM techniques [4] now. RED prevents global synchronization and bias against bursty flows to overcome limitation of Drop Tail. RED helps to eradicate bias by using randomized algorithm at the RED gateway for marking a packet. RED technique governed by two important parameters: minimum threshold (min_th) and maximum threshold (max_th). The average queue size (avg_q) is calculated for every packet arrived at the router. Then it compares the avg_q with two defined threshold value min_th and max_th.

There are three cases may arise.

Case 1: avg_q is less than min_th.

Case 2: avg_q is greater than max_th.

Case 3: avg_q is in between min_th and max_th.

When the case 1 is aroused then no packets are dropped/marked. If case 2 arises, then RED drops all incoming packets to lower the queuing delay. For case 3, arriving packet is marked with probability Pa, where Pa is function of avg_q. The probability of marking ranges from 0 to maximum packet dropping probability (maxp). The performance of RED protocol is greatly depending on its parameter setting [18]. The behavior of RED varies with different traffic load and setting of parameters.

B. Random Exponential Marking

REM is another stable AQM technique which has different congestion measure and different marking probability function than RED [5][6]. Through the marking mechanism the end user, called source is notified about the congestion. REM measure congestion by quantity, called price rather using queue length as in RED. The buffering process of RED implicitly updated the queue length, whereas update of price is explicitly controlled by REM. The first objective of REM is to stabilize the input rate and queue irrespective of number of source sharing the link. The second objective of REM is to find aggregate link prices as a measure of congestion and notify the source through the end-to-end marking probability so as to adopt the rate.

C. Adaptive Virtual Queue

AVQ [9][10] is a rate based techniques which incurs low-delay, few loss and high link utilization at the link. In this technique one virtual queue is maintained at router whose capacity is less than actual link capacity. Every time virtual queue is updated after each packet arrived in the real queue. When the virtual queue overflows then packets in virtual queue is discarded and real packets is marked/dropped in real queue.

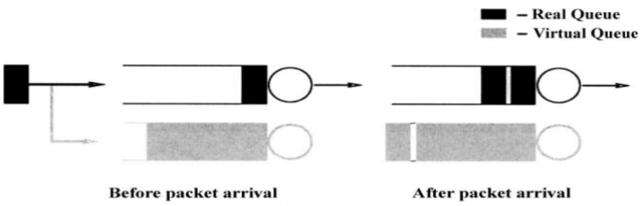

Fig. 1. AVQ: when virtual queue is not full.

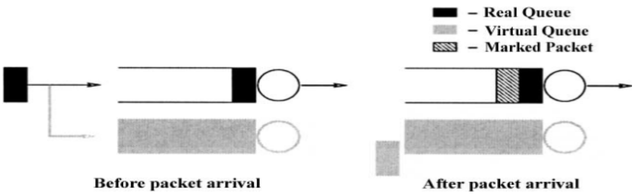

Fig. 2. AVQ: when the incoming packet is dropped from the virtual queue

The virtual capacity at each link is modified in such a way that total flow entering each link achieves a desired level of utilization of that link. Thus, this method controls the virtual queue instead of directly controlling the real queue length using a dropping probability, AVQ controls the virtual queue capacity, which implicitly applies a dropping probability on packets in the real queue. No dropping probability is being calculated directly.

shows an AVQ specification in pseudo-code, where VQ is the number of bytes currently in the virtual queue, b is referred to as the number of bytes of the arriving packet, B is referred to as the total buffer size of virtual queue, and the last arrival variable is used for storing the time of the most recent packet arrival at queue. The 'update VQ' event consists of updating the variable holding the current virtual queue length, since it has changed since the previous packet arrival event, e.g., because of packets being served that have left the queue. Note that this is different from updating the virtual capacity.

```
at each packet arrival do:
   update V_Q;
   if (V_Q + b > B)
      mark packet in real queue;
   else
      V_Q ← V_Q + b;
   endif
   update virtual_capacity;
   update last_arrival;
```

Fig 3: Pseudo code of Adaptive Virtual Queue

III. PROBLEMS WITH AQM

Many AQM strategies use average queue length (avg) to determine network congestion. This may cause some problems [7] like:

a) When large number of bursts arrive at a gateway, the actual queue size is rapidly increased, resulting in queue overflow, sources will reduce their transmission rate after a congestion signal is being triggered due to packet drop at congested node. After congestion has been detected the actual queue size is being decreased, the average queue size will be high due to previous higher value in the actual queue size. Therefore, packet dropping may be continued even though congestion problems have been rectified, which unfairly penalizes packets received after the congestion event.

b) The actual queue size is used for the early indication of congestion. Due to the use of average queue size, the recent variations in the traffic are not recognized by AQM strategies. This can lead to unfair packet drops between connections. However, AQM Strategies instead of using the average queue size tend to use the actual queue size to indicate congestion, suffer from worst cases of unfair packet drops.

c) Parameter configuration in AQM strategies is a tough task. Many AQM modifications have been proposed for enhancing network performance which is being evaluated using Analytic modeling and simulation. Unfortunately, these modifications work only for specific traffic conditions but not for realistic IP traffic.

IV. PROPOSED SOLUTION APPROACH

Adaptive Virtual Queue (AVQ)[9] is designed that gives the results in term of low-loss, low-delay and high utilization of the link. AVQ algorithm maintains an actual queue and a virtual queue. Virtual queue capacity is less than the capacity of the actual queue. All nodes in the system forward packets to the neighboring nodes and all nodes will have a real queue and virtual queue. The size of the virtual queue (Cv) is less than the size of the real queue (Cr).

$Cv < Cr$ …………….. Eqn.1

When a packet enters in the real queue (Cr), a fictitious packet is updated in the virtual queue. Due to the capacity of virtual queue (Cv) is less than the capacity of real queue (Cr), it is clear that virtual queue overflow prior to the filling of real queue. When the virtual queue overflows notification is sent to the source to indicate incipient congestion. If the Cv is much less than the Cr then it will notify the incipient congestion much earlier, thus underutilization of real queue and if the Cv is equal as Cr then it will overflow the packet from both the queues and need to re-forward the packet.

In proposed Solution Enhanced Adaptive Virtual Queue (EAVQ) which will detect the underutilization of real queue (Cr) and it increases the utilization of queue by gradually enhancing the capacity of virtual queue, so that it increases the Queue utilization gradually and reached up to fully utilization of real queue. In the proposed solution we have make the queue sensitive to overflow so that no overflow will occurs in the network. Due to full utilization of real queue the performance of network will increase. experimental/simulation results

Proposed Algorithm

```
At each packet arrival
{
/* update the virtual queue size */
VQ ← max (VQ – Vc (t-s) , 0)
If VQ + b > B
/* virtual Queue overflow */
If (queue_length < Vc)
{ \\ enqueue packet into queue
\\ increase queue length by packet size
VQ ← VQ + b
}
Else if (queue length is  >= Vc)
{
 If (enhanced_AVQ)
{
\\ increase virtual_queue_capacity
\\ mark packet and enqueue into queue
\\ increase queue length by packet size
VQ ← VQ + b
}
/* update last packet arrival time */
s ← t
}
```

V. SIMULATION RESULTS

1. Throughput – Throughput is the average rate of successful packet delivery over a communication network.

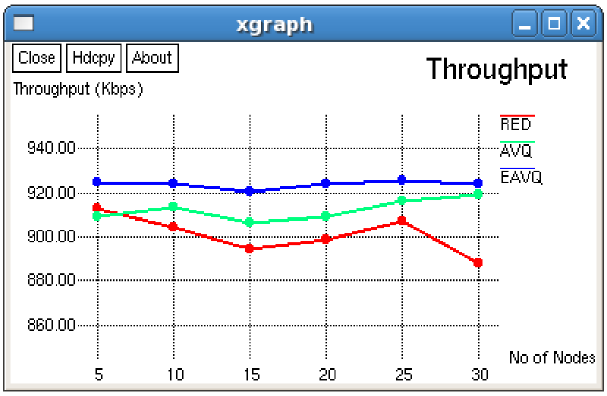

Fig 1: Throughput

Fig 1 shows the throughput of the network, in which enhanced adaptive virtual queue throughput is more than the both AVQ and RED. Quantitative throughput is given in table 1.

No of Nodes	Throughput (kbps)		
	RED	**AVQ**	**EAVQ**
5	912.44	909.20	924.70
10	904.20	913.23	924.17
15	894.29	906.67	920.79
20	898.37	909.37	924.19
25	907.22	916.12	925.27
30	888.28	919.05	924.06

Table 1: Throughput

2. Packet Delivery Ratio - This is the ratio of number of data packets received to the number of data packets transmitted.

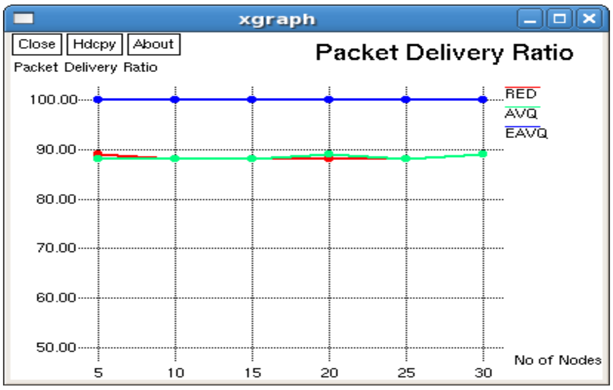

Fig 2: Packet Delivery Ratio

Fig 2 shows Packet Delivery Ratio of the network, in which enhanced adaptive virtual queue PDR is more than the both AVQ and RED. Quantitative measure is given in table 2.

No of Nodes	Packet Delivery Ratio		
	RED	AVQ	EAVQ
5	89	88	100
10	88	88	100
15	88	88	100
20	88	89	100
25	88	88	100
30	89	89	100

Table 2: Packet Delivery Ratio

3. Data Dropped – This is the difference between number of packet transmitted by source and number of packet received by receiver at receiver end.

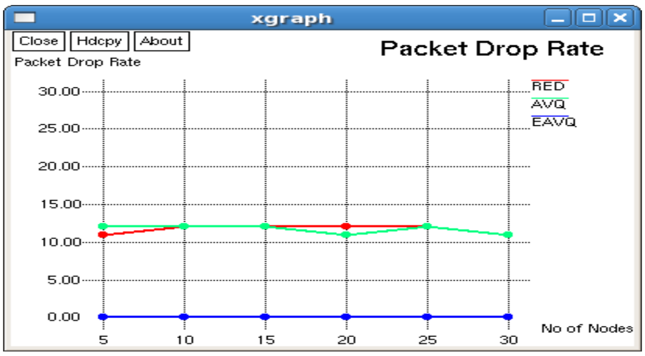

Fig 3: Packet Drop Rate

Fig 3 shows the Data Drop Rate of the network, in which enhanced adaptive virtual queue DDR is less than the both AVQ and RED. Quantitative measure is given in table 3.

No of Nodes	Data Dropped Rate		
	RED	AVQ	EAVQ
5	11	12	00
10	12	12	00
15	12	12	00
20	12	11	00
25	12	12	00
30	11	11	00

Table 3: Data Dropped Rate

4. End-End Delay - the end-to-end delay of packet is duration of time from generation of a packet from the source up to the destination.

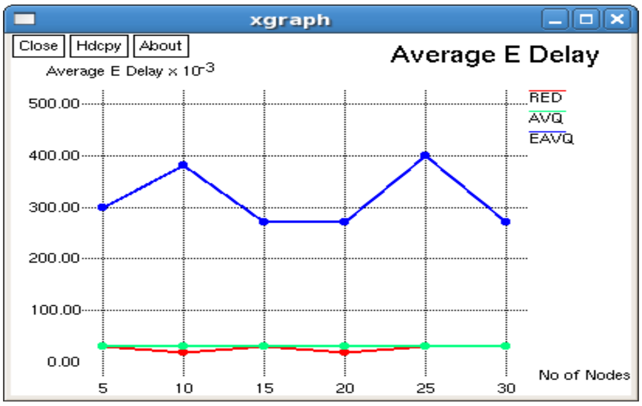

Fig 4: Average End-End Delay

Fig 4 shows the Average end-end Delay of the network, in which enhanced adaptive virtual queue Average E Delay is more than the both AVQ and RED. Quantitative measure is given in table 3.

No of Nodes	Average End-End Delay (ms)		
	RED	AVQ	EAVQ
5	0.03	0.03	0.30
10	0.02	0.03	0.38
15	0.03	0.03	0.27
20	0.02	0.03	0.27
25	0.03	0.03	0.40
30	0.03	0.03	0.27

Table 4: Average End-End Delay

VI. CONCLUSION

In this paper proposed mechanism named enhanced AVQ has improved the performance in terms of throughput, packet Delivery Ratio and packet drop rate in mobile ad hoc network. It has verified from simulation results that EAVQ performance is better than existing adaptive virtual queue and RED, but it has included some overhead which increases average end-end delay in the network. simulation results are performed in NS 2.34.author need to describe experimental/simulation results with graphs and appropriate tables.

REFERENCES

[1] Patel j. k., Dubey j., “Mobile Ad Hoc Network Performance Improvement Using Strategical RED”, Ninth International Conference on Wireless and Optical Communications Networks WOCN2012, 2012, IEEE.

[2] V. Firoiu and M. Borden, “A study of active queue management for congestion control”, In Proceedings of the IEEE Infocom, pages 1435–1444, Tel Aviv, Mar 2000.

[3] S. Athuraliya, V. H. Li, S. H. Low, and Q. Yin, “Rem: Active queue management, IEEE Network, Vol. 15, pp. 48-53, 2001.

[4] S. Floyd and V. Jacobson. “Random early detection gateways for congestion avoidance”, IEEE/ACM Transactions on Networking, 1(4):397–413, 1993.

[5] Chin-Hui Chien; Wanjiun Liao,“A self-configuring RED gateway for quality of service (QoS) networks, Multimedia and Expo”, 2003. ICME '03, Page(s): I - 793-6.

[6] C. Hollot, V. Misra, D. Towsley, and Wei-Bo Gong, “On designing improved controllers for AQM routers supporting TCP flows,” in Proceedings of IEEE/INFOCOM, April 2001.

[7] Nabhan Hamadneh “Analysis and Design of Active Queue Management for TCP-RED Control Strategies”, NOV 2012.

[8] Srisankar S. Kunniyur, R. Srikant, “An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management” IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 12, NO. 2, APRIL 2004.

[9] Mr. A. Chandra, Ms. T. Kavitha, “Adaptive Virtual Queue with Choke Packets for Congestion Control in MANETs”, IRACST – International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250-350 Vol.4, No2, April 2014.

[10] The network simulator. “ns-2”. http://www.isi.edu/nsnam/ns,”.