International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 06 | June-2016 www.irjet.net p-ISSN: 2395-0072

PACKET TRANSMISSION IN MANET USING DYNAMIC QUEUE MANAGEMENT SCHEME

Jebah Jaykumar[1], Prarthana T V[2]

[1]*Assistant Professor, Dept. of Computer Science and Engineering, BNMIT, Bangalore, India*
[2]*Assistant Professor, Dept. of Computer Science and Engineering, BNMIT, Bangalore, India*

---***---

Abstract - *Mobile Ad hoc Network [MANET] is a wireless network of many mobile nodes with dynamic topology. The proper maintenance of resources to avoid network congestion is the main issue in MANETS. Congestion occurs when the requirement of resource is much higher than the available resources or due to interference or by the frequent path breaks due to mobility of the nodes. Congestion will results in high data loss, increased queuing delay and degradation of the throughput. In this project to effectively utilize the existing network resources and to maintain the network load below the limit a new scheme of buffer management to handle queues in MANET is introduced, namely Dynamic Queue Management (DQM). DQM dynamically allocates and updates the buffer space of the nodes. DQM is implemented on the centrally communicating MANET node called as Queue Management Node (QMN). It dynamically allocates the buffer space in proportion to the number of packets received and an allowable extension is also available to avoid the under utilization of resources. QMN also plays an important role in identifying the misbehaving nodes. We extend the DQM mechanism by incorporating the scheduling mechanism at the de-queuing phase.*

***Key Words*: Active Queue Management (AQM), Mobile Ad hoc Network (MANET), Queue Management Node (QMN).**

1.INTRODUCTION

Mobile Ad hoc Network [MANET] consists of set of mobile nodes, where nodes communicate with each other using multihop links. For communication between the nodes there is no permanent infrastructure or base station. Each node acts as a host or router for forwarding and receiving packets to and from the nodes. The major challenge in MANET is to handle the congestion. Congestion results in congestion collapse, it is the situation in which the throughput drops to the low level and thus little useful communication occurs. The effects of congestion collapse are queuing delay and packet loss. Another reason that causes the packet loss is mobility. Any communication network consists of a network of queues. Packets are queued up in the memory buffers of the network device like routers. In a specific manner packets are arranged in the device buffer and are known as queuing techniques. A queue is a collection of request waiting to be executed one at a time. Mechanism for protecting individual flows from congestion is provided by Queue Management. The main idea in Queue Management is to identify the congestion and to reduce the transmission rates before queue overflows and packets are dropped.

Drop Tail is a traditional way to control the queue length at the routers. Maximum length for each queue at the router is set by the drop tail and all the incoming packets are accepted until the maximum queue length is reached. Once the maximum queue size is reached, the algorithm drops all the incoming packets until the queue size is again below the maximum. It is the simplest queue management technique. Large queue, delay, low global powers accommodate transient congestion periods and global synchronization problem are the draw backs of drop tail [13]. To overcome this problem, a novel scheme called Random Early Detection (RED) triggered a new discipline called Active Queue Management (AQM). A class of algorithms is designed to provide the improved queuing mechanisms for routers by the AQM schemes. These schemes are called active because they dynamically signal congestion to sources; either explicitly by marking packets or implicitly by dropping packets. The underlying principle marks or drops the packets before a queue overflows so that sources respond to congestion thereby avoiding buffer overflow. AQM queues can operate with minimal packet loss with ECN. AQM schemes are effective techniques to avoid congestion and delay. ECN allows end to end notification about the network congestion without dropping packets. To prevent unnecessary packet drops and to detect congestion, ECN mechanism is used for notification at the end nodes.

2. RELATED WORK

In both wired and wireless forms of network several works has been done in the field of packet queue management. Several mechanisms were proposed over the years for the adequate control of congestion that occur in the network. Active Queue Management (AQM) is one such mechanism which yields better control in the recent years. It works at the router for governing the number of packets in the router's buffer by strongly rejecting an arriving packet. Many schemes were scheduled, which give better delay performance and high throughput over different traffic conditions. In MANET environments some good efforts have also been made. The pioneering steps in this direction are:

Kulkarni et al. [1] proposed and tested a queue management scheme called PAQMAN against traditional RED algorithm in IP networks. PAQMAN takes the leverage of the predictability in the basic traffic to measure the average queue length in the future by applying the RLS algorithm. It is simple to implement and does not require any preceding knowledge of the traffic model. It precisely captures the fluctuations in the input traffic and assembles immediately in less than a second. The performance of PAQMAN is checked by comparing it with that of RED (S.Floyd et al., 1993) with respect to other performance metrics for differing number of flows and a variety of traffic mixes. PAQMAN scheme accurately predicts the average queue length at a future time point and manages an almost low queue size, high link utilization and low packet loss (and hence improved Quality of Service (QoS)) in correlation to RED is shown by the simulation results.

Essam and Adznan [15] proposed an active queue management scheme called Fuzzy-AQM. It is a novel AQM algorithm (Fuzzy-AQM) based on fuzzy logic system and is used to overcome the congestion in adhoc networks. The utilization of fuzzy logic to the issue of congestion control allows determining the relationship between queue parameters and packets dropping probability. The fuzzy logic algorithm would be able to translate or interpolate these rules into a nonlinear mapping. This algorithm for initial packets dropping is enforced in wireless ad-hoc networks in order to yield effective congestion control by attaining high queue utilization, low packet losses and delays. The proposed scheme is varied with a number of well-known AQM schemes through a wide range of scenarios. From the simulation results, the efficiency of the proposed fuzzy AQM policy in terms of routing overhead, average end-to-end delay and average packet losses are noticeable than other AQM polices, with efficiencies of acclimating to high variability and ambiguity in the mobile ad-hoc networks.

A. Chandra and T. Kavitha [14] proposed a mechanism called Adaptive Virtual Queue with Choke Packets (AVQCP). Adaptive Virtual Queue manages a virtual queue whose capacity is less than the actual capacity of the link. Whenever a packet enters, a duplicate packet will be restored in the virtual queue, when the virtual queue is full or drops a packet feedback regarding congestion will be asserted to the source using a choke packet. When the virtual buffer overflows, choke packets are sent to the source. When the source receives a choke packet it reduces the traffic sent to a particular destination by some percentage. For a fixed interval of time sources reject repeating choke packet. After a certain time if no further choke packets arrive, the source will again increase the traffic. The proposed mechanism was compared with RED and REM and the performance of AVQCP is comparatively good. It involves additional overhead to the traffic and maintenance of virtual queue depletes additional buffer space.

3. PROPOSED WORK

A new design of buffer management called as Dynamic Queue Management (DQM) for packet queues in MANETs for mobile nodes is proposed. For a MANET node, the packet queue is managed in such a form that a same buffer space is allocated to each adjoining source and an acceptable extension is also applicable to each neighbor to avert any underutilization of resources. DQM is an Active Queue Management approach, the allocation is made in the buffer of a centrally communicating MANET node and it is based on number of packets received in the queue at node's buffer to handle the buffer space conveniently without any monopolization of some neighboring source. A MANET model working on Ad hoc On demand Distance Vector (AODV) routing protocol is considered.

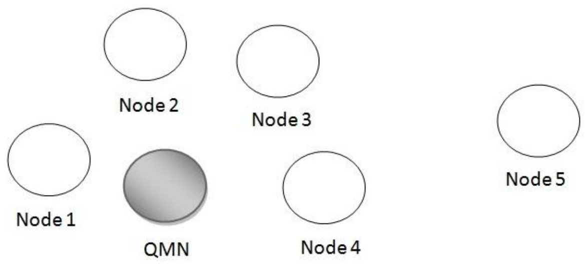

Fig. 1 The considered MANET scenario.

Centrally communicating MANET node is called as Queue Management Node (QMN). It is encompassed by four neighbors i.e. "Node 1" to "Node 4" as shown in Fig.1. There is another node, "Node 5" which is not considered as a neighbor of QMN at initiative stage. QMN is used to allocate buffer space to its neighbors according to the proposed design. Let the buffer space of QMN to be 40 packets and it is treated empty at initiative stage. The node QMN allocates same buffer space to all of its neighbors in the initialization phase of allocation. If "bs" is buffer space and "nn" is number of neighbors of QMN, then for each neighbor, the allocated buffer space "abs" is

$$abs= rbs/nn \qquad (1)$$

In the considered case, each of the four neighboring nodes gets the buffer space of 10 packets in QMN. Though, it is decisive to examine if a neighboring node reaches its maximum allocation in QMN but the space is left in the total buffer due to the speculation that one or more other neighbors are not transferring or dispatching the packets to QMN. So, if a node is confined to an allocation of 10 packets but the space is still left in the buffer, there is a chance that resource would be underutilized. So an adjunct is provided to each source which reaches or is about to reach the assigned limit of 10 packets up to a portion of residual buffer space, i.e., the transitory space left in the buffer. Here Node 1 is sending 10 packets to QMN is assumed and attained the assigned limit in the buffer space of QMN. However, a part of the buffer space of QMN is still unfilled because other neighbors are not transferring packets to QMN or they are dispatching with small number of packets than their assigned limit of 10 packets per neighbor. Thus, for a conducive utilization of the remaining buffer space of QMN, the residual buffer space “rbs” is resolved and it is divided amongst all the neighboring nodes in proportion to their actual occupied buffer spaces. This division is sensible and it adequately reduces the allocated buffer space for nodes which communicate with small number of packets but hikes the same for nodes which stance same or nearest to their assigned buffer space limits. Therefore, at this stage it is considered as fair because it gives a fair share in the buffer of QMN to nodes which is sending more number of packets. The extended buffer space “ebs” allocated to Node 1 is calculated through the accord specified in Eq. (2).

$$ebsNode1=\left\lfloor rbs/\sum_{i=1}^{nn} i \right\rfloor * nn \qquad (2)$$

Here i represent the index of the number series from 1 to number of neighbors. For better understanding, let transitory occupied spaces in the buffer of QMN by the specified four neighbors be 10, 6, 8 and 4 packets from Node 1 to Node 4, respectively. As 10 packets is the maximum assigned limit at the initiative stage, it reveals that Node 1 has reached its limit whereas total occupied buffer space is 28 packets by all neighboring sources, i.e., 12-packet buffer space is presently unoccupied. Therefore, the algorithm used in the project divides the residual buffer space, i.e., 12 with a summation figure attained by adding a number series from 1 to number of neighbors. In this designed case, it is 4. Therefore, it divides 12 by 10 (i.e., 4+3+2+1) and multiplies the result with the largest number of the series which is 4 in this case. The largest number is multiplied so that the biggest share from residual buffer space can be allocated to the node having reached the assigned maximum limit. The floor value is taken from the result and it is added to the transitory buffer space occupied by Node 1. In this considered case,4 is added to 10 which increases the maximum buffer space allocated to Node 1 and it now becomes 14. Correspondingly, the neighbor that occupies the largest buffer space in QMN after Node 1 is Node 3 with 8 packets. It is about to reach its assigned limit of 10 packets in QMN. Thus, the proposed scheme should provide it an extension in the buffer space. For Node 3, the calculated ebs is

$$ebsNode3=\left\lfloor rbs/\sum_{i=1}^{nn} i \right\rfloor * (nn-1) \qquad (3)$$

From Eq. (3), it can be checked that 12 is divided by 10 as before but the result is multiplied with the second largest number of the series which is 3 (i.e., 4-1) in this case. Here it shows that the series nn, nn-1, nn-2…, 1 completely consists of normalization factors along with the summation of neighbors indices for the proposed buffer allocation design. The multiplication factor nn of Eq. (2) is replaced by nn-1 for Node 3. The floor value is taken from the result and it is added to the transitory buffer space occupied by Node 3. In the designed case, 3 is added to 8 which also extends the maximum buffer space allocated to Node 3 and it now becomes 11. It shows that an extension is also given to Node 3 in fair means as it is about to reach its maximum allocated space. In the same way, allocated buffer spaces for Node 2 and Node 4 are also accustomed and their new allocations are 8 and 7 packets respectively. The multiplication factor nn of Eq. (2) is replaced by nn-2 for Node 2 and nn-3 for Node 4. It can be seen that the maximum allowable buffer space is shrinked for both nodes but it is fair against accommodating the same for neighbors communicating with more numbers of packets. Ultimately, the total buffer space in QMN limited to 40 packets (i.e., 14+11+8+7) according to the deliberation. The hypothesis behind dividing rbs with a summation of neighbors indices and multiplying with components of the series of such indices is to assign the higher calculated limits to the neighboring nodes which are nearest to assigned buffer space limits in proportion to the number of packets received. It can be seen that the proposed scheme accredits dynamic buffer space to neighboring nodes proportional to the number of packets received and hence regulates the packet drop probabilities for them as explained above.

An imperative scrutiny is how the algorithm recalculates and retains buffer space allocations for neighbors when packets are discrete, i.e., increased or decreased while processed in the buffer. First of all, maximum and minimum buffer space limit that a single node can occupy is applied. For this purpose, let us speculate that a neighboring node cannot get buffer space of more than 15 packets in QMN so that it cannot misuse the buffer through combative mode of sending packets. Similarly, lower limit of 4 packets is applied so that a node cannot be neglected in the buffer space when it is a neighbor of QMN. Since one of the nodes has attained its assigned limit, the algorithm prompts again, recalculates buffer space allocations for neighbors as explained above, and assigns corresponding limits to neighbors. The calculations are made and buffer allocations are accustomed only when a neighboring node approaches the assigned upper limit of buffer space in QMN so that the overhead of proposed scheme remains decent. In this way, dynamic buffer space is assigned to neighboring nodes proportional to the number of packets received and appropriately packet drop probabilities are restrained.

ALGORITHM 1: DYNAMIC ALLOCATION OF BUFFER SPACE

Calculate total instantaneous buffer space occupied.

2- Determine gap between assigned limit and buffer space occupied by each node.

3- Arrange the gap values in ascending order with corresponding nodes.

4- Obtain sum of number series (1 →number of neighbors “nn”) say “Sum”

5- Obtain difference between total buffer space and total buffer space occupied, i.e., residual buffer space “rbs”.

6- Calculate (rbs / Sum) * nn and add it to buffer space occupied by the node with the least gap available through Step 3.

7- Calculate (rbs / Sum) * (nn - 1) and add it to buffer space occupied by the node with the gap more and closer than the least available through Step 3.

8- Repeat Step 7 for all remaining neighbors with decreasing value of “nn” by “1” each time and selecting nodes with respect to increasing values of gaps available through Step 3.

When number of neighbors is bartered at an instant, i.e., a new neighbor arrives and it is notified to QMN, the proposed scheme rapidly reconstructs assigned buffer spaces in such a way that identical share is allocated to all neighbors including the new one. However, it does not alter the present tenancy of packets and the new neighbor has to interim for processing of packets in the queue until the space is obtainable equal to its originally assigned limit in QMN. The new incoming packets from current neighbors are discarded by QMN during this waiting period, after they reach their analogous maximum limits until the new neighbor holds its share in the buffer of QMN according to its originally assigned limit. After that, the process pursues as explained before and the buffer space is dynamically accustomed and allocated to all existing neighbors according to transitory share of neighbors in the QMN’s buffer and the gap between the occupied and allocated buffer spaces. On the other hand, when a neighboring node is driven away and it is no more a neighbor of the QMN then a new portion of the buffer is feasible for existing neighbors. In this situation, the proposed scheme directly checks the buffer space accessible against eviction of the neighbor and allocates identical share of the space to current neighbors. In this way, maximum limit of presently allocated buffer space is adequately expanded for each current neighbor. For improved perceptive again, we assume that Node 5 mentioned in Fig.2 has now become a neighbor of QMN as shown in Fig.2

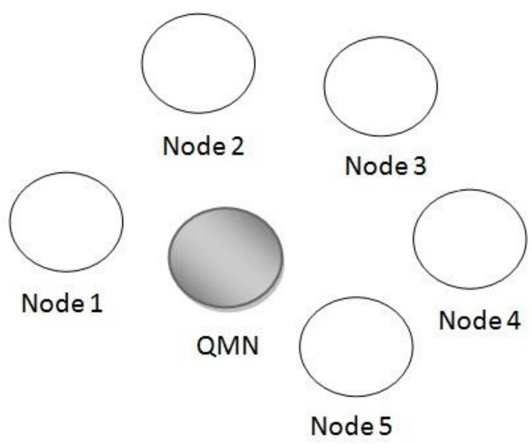

Fig. 2 The QMN node surrounded by five neighbors.

When QMN is notified about this neighbor after accepting HELLO message from Node 5, the assigned buffer spaces is reconstructed by the proposed scheme in such a way that identical share is allocated to all neighbors including Node 5. When Node 5 is also treated as a neighbor of QMN for the first time, the buffer space is divided into amended number of neighbors which is now 5. Maximum allocated limit to each neighboring node is now assigned as an 8 packet buffer space. Node 5 has to interim for processing of packets in the queue until space is obtainable equal to its originally assigned limit of 8 packets in the QMN's buffer. During this waiting period, new incoming packets from nodes which have reached the assigned maximum limits are dropped until they reach their new maximum limits of 8 packets each and Node 5 holds its own share of 8 packets in the buffer. At the dupe time, the buffer can accept the packets from those nodes which are about reach the limits. After that, the process endures as explained before and buffer space is dynamically regulated and allocated to all existing neighbors according to transitory share of neighbors in the QMN's buffer and the gap between the occupied and allocated buffer space.

In the proposed scheme an active queue management approach is applied to notify a neighboring sender when its assigned limit is about to reach in the buffer of QMN. Upon getting this notification, the neighbor can stop sending data or decrease the rate of sending data so that the current occupied space allocated to that peculiar sender can be made sufficiently feasible again after processing and de-queuing of packets from the QMN's buffer. After a sufficient space is again accessible in the queue for that peculiar neighbor, it can hike the packet sending rate. The purpose of AQM in this scheme is very fruitful to avert packet losses because a sender receives an alert to down turn sending packets on the expected incidence of reaching its assigned buffer space limit until a recalculated buffer space is catered to accommodate more packets in the queue from that source. For this purpose, ECN is applied in the packet header sent from QMN to its neighbor during the transfer of routing information to keep renewed routing tables. A threshold value of 0.9 is used, i.e., QMN sends ECN empowered packets to the neighbor when 90% of its allocated space in the buffer is occupied. When the neighbor is notified about congestion by this ECN information, it rapidly responds by accomplishing its congestion window one tenth of the actual width. In this way, initial notification of congestion is made desirable through the AQM applied scheme which reduces packet losses and revamps transmission efficiency in the network.

ALGORITHM 2: Handling A New Neighbor

1- Calculate total instantaneous buffer space occupied by each node.

2- Increase neighbor count 'nn' by '1'.

3- Divide total buffer space by 'nn'.

4- Assign calculated buffer spaces (equal space values obtained through step 3) to all neighbors including the new one as updated buffer space allocations.

It can be viewed that the proposed scheme reflects the accession of max-min fairness algorithm. The max-min fairness algorithm grants priority to data flows acquiring minimum flow rate. However, the proposed method targets on buffer space and awards priority to packets impending from those nodes having higher contribution in the communication as correlated to others. The packet eruption can be noticed by QMN either from a legitimate donor or from some misbehaving node in the MANET. The function of maximum buffer space limit is the starting step towards defending QMN's buffer form eruption of packets impending from a neighbor. Moreover, it is generally expected that a legitimate neighbor will not regularly or frequently occupy the space in QMN's buffer equal to the maximum limit applied. If it is the case with some neighbor, it is possibly be a misbehaving node and the proposed scheme should have some elucidation to this problem. Thus, an observation window is designed with assertive time intervals and an algorithm is designed to inspect nodes (neighbors) which occupy the maximum buffer space limit in QMN. If a neighbor is erect acquiring occupied maximum buffer space limit for three successive observation intervals, it is considered as a misbehaving node for which the buffered packets are dropped and the assigned buffer space limit of that assertive neighbor is contrived to become the minimum, i.e., 4 packets in the QMN's buffer. The newly accessible buffer space, that becomes free as a result of drop of the buffered packets of misbehaving node, is apportioned amongst tarrying nodes (neighbors) in uniform manner.

ALGORITHM 3: Identification of the misbehaving nodes

1- Drop buffered packets of misbehaving node (neighbor).

2- Force the assigned buffer space of misbehaving node to become to the minimum limit to be assigned to a neighbor.

3- Distribute the available buffer space (due to recent packet drops) amongst other neighbors in equal manner.

4- Apply the updated buffer space allocations.

Even if the suspected neighbor is not an willful misbehaving node, the observation that it occupies maximum buffer space limit in QMN for three successive observation intervals provides an belief that this neighbor is becoming a matter of congestion, thus the traffic drop of that peculiar node is sane. Moreover, a legitimate neighbor would surely react to ECN signals and stop sending massive traffic in an unceasing mode. As a result, the processing of packets would consent more space feasible to the node in QMN's buffer and hence scopes of meeting the maximum limit are shortened. Even in this case, if a neighbor is occupying the buffer space with maximum limit for three successive observation intervals then it must be an ambiguous node with misbehavior or congestion. Any detachment of the node from MANET is averted since this neighbor can be a legitimate node annoying to build the communication with QMN. Therefore, it can frame up its allocated buffer space limit by communicating again with QMN and the rest is done according to the designed scheme of buffer space allocation. The depiction of observation window is given in Fig. 3

It is presented in Fig.3 that observation intervals are named as tobs1; tobs2; and tobs3 for which the algorithm interim before verifying the occupancy of nodes (neighbors) in the buffer of QMN. Interposed intervals between tobs1 and tobs2, and tobs2 and tobs3 also occur during which the buffer occupancy of nodes is literally verified after an observation interval is moved. These interposed intervals also grant an extra space of time to neighbors so that they might get a contingent to get purge of maximum buffer space occupancy in QMN (if observed) by making use of this rim of time. Every interposed interval is not more than 0.25 times of an observation interval. In addition, the observation intervals are identical in length, i.e., tobs1 = tobs2 = tobs3. If a neighbor is found having occupied maximum allowable buffer space (determined as 15 packets) during tobs1; tobs2; and tobs3 along with two interposed intervals in successive manner, the buffered packets of such a node is dropped from the QMN's buffer in excess of 4-packet buffer space which is the minimum configured limit of buffer space in the proposed work.

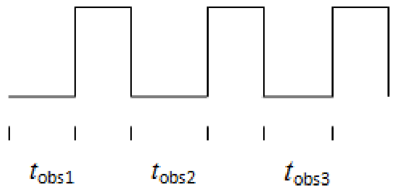

Fig.3 Sketch of the observation window to determine misbehaving nodes.

The processing of packets in the buffer can be enforced through the scheduling mechanism. It applies on the packet de-queuing phase in the buffer. The proposed scheme accords with allocation of buffer space and every node is evaluated equally for packet en-queuing phase in the buffer and allocates buffer space in consonance with number of packets received. Time Division Multiple Access (TDMA) is adapted for scheduling in the de-queuing phase. It is a multi channel scheduling mechanism. It allocates a fixed time slot per packet to each node over multiple channels. It is a channel access method in which nodes transmit in rapid succession, one after the other each using its own time slot.

4. PERFORMANCE EVALUATION

Simulation of the proposed DQM design was done using network simulator 2 (NS2). The performance of the proposed scheme is correlated with Drop Tail queue management and ECN enabled PAQMAN scheme with AODV routing scenarios contrived in the simulator in terms of packet loss ratios and transmission efficiencies.

Fig. 4 Snapshot of Buffer Allocation

For evaluation prospect, a MANET with 30 nodes is designed. The packet loss ratio in 30- node scenario is invaded where a precise node is selected as QMN in the unit of 6 nodes and other 5 nodes are its neighbors is shown in fig.4. It means that there are 5 QMN nodes in the scenario and each of them is surrounded by 5 different neighbors. The comparisons show that the proposed scheme yields better transmission efficiency, packet loss ratio in MANET as compared to Drop Tail and PAQMAN schemes for tested flow arrival rates is shown in fig.5.

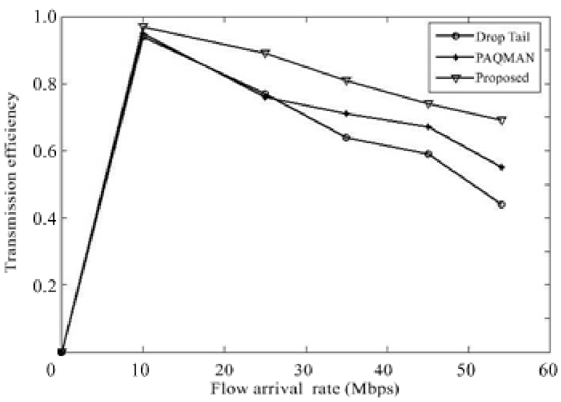

Fig. 5 Flow arrival rate vs. transmission efficiency

5. CONCLUSION AND FUTURE WORK

In this paper a new scheme called Dynamic Queue Management that beats the demerits in well known queue management algorithms has been proposed. This mechanism is used to allocate the buffer space to every node in fair terms. The algorithms are simple, sturdy, low in computational ramification, easily designed and easy to set up. The algorithms are used for regulating packet drops in wireless ad hoc networks in order to render efficient congestion avoidance by attaining better transmission efficiency, low packet loss and delay. The proposed scheme is contrasted with Drop Tail and PAQMAN schemes in terms of throughput, packet end to end delay and jitter statistics and found it better. In future effectiveness of DQM will be analyzed using different protocol in MANET having more variations of flow arrival rates.

REFERENCES

[1] P. G. Kulkarni, M. Nazeeruddin, S. I. McClean, G. P. Parr, M. M. Black, B. W. Scotney, and P. Dini, Deploying lightweight queue management for improving performance of mobile ad-hoc networks (MANETs), in Proc. Int. Conf. On Networking and Services (ICNS), 2006.

[2] V. Friderikos and A. H. Aghvami, Hop based queueing(HBQ): An active queue management technique for multi hop wireless networks, in Proc. IEEE ICICS/PCM 2003, pp. 1081-1085.

[3] J. Lutz, C. J. Colbourn, and V. R. Syrotiuk, Variable weight sequences for adaptive scheduled access in MANETs, in Sequences and Their Applications SETA 2012, T. Helleseth, J. Jedwab, Ed. Springer Berlin Heidelberg,2012, pp. 53-64.

[4] S. Chen and B. Bensaou, Can high-speed networks survive with DropTail queues management? Comput. Netw., vol. 51, no. 7, pp. 1763-1776, 2007.

[5] B. Abbasov and S. Korukoglu, Effective RED: An algorithm to improve RED's performance by reducing packet loss rate, J. Netw. Comput. Appl., vol. 32, no. 3, pp. 703-709, 2009.

[6] S. Dimitriou and V. Tsaoussidis, Promoting effective service differentiation with size-oriented queue management, Comput. Netw., vol. 54, no. 18, pp. 3360-3372, 2010.

[7] M. Aamir, M. Zaidi, and H. Mansoor, Performance analysis of Diffserv based quality of service in a multimedia wired network and VPN effect using OPNET, International Journal of Computer Science Issues, vol. 9, no. 3, pp. 368-376, 2012.

[8] L. Chrost and A. Chydzinski, On the evaluation of the active queue management mechanisms, in Proc. 1st Int. Conf. On Evolving Internet (INTERNET), 2009, pp. 113-118.

[9] K. R. S. Floyd and D. Black, The addition of explicit congestion notification (ECN) to IP, RFC 3168, Sep. 2001.

[10] C. Perkins, E. Belding-Royer, and S. Das, Ad hoc on-demand distance vector (AODV) routing, RFC 3561, July 2003.

[11] K. D. Kumar, I. Ramya, and M. R. Masillamani, Queue management in mobile adhoc networks (Manets), in Proc. IEEE GreenCom/CPSCom 2010, 2010.

[12] M. Aamir and M. Zaidi, QoS analysis of VoIP traffic for different codecs and frame counts per packet in multimedia environment using OPNET, in Proc. 15th IEEE Int. Multitopic Conf., 2012, pp. 275-281.

[13] Shweta Mishra and Nidhi Jain, Effective Impact of ECN with RED to Control Heavy Congestion, (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (3) , 2012,4290 – 4293.

[14] A. Chandra and T. Kavitha, Adaptive Virtual Queue with Choke Packets for Congestion Control in MANETs, IRACST – International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250-3501 Vol.4, No2, April 2014.

[15] Essam et al, Fuzzy Active Queue Management for Congestion Control in Wireless Ad- Hoc, The International Arab Journal of Information Technology, Vol. 4, No. 1, January 2007

[16] Kajal and Gaurav, Performance Improvement by Dynamic Queue Management in Mobile Ad hoc Network, International Journal of Computer Applications (0975 – 8887) ,Volume 72– No.1, May 2013