Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 118 (2018) 50655-50660

ISSN: 2229-712X

Classification of RED AQM and Performance Comparisons

Ramadevi Chappala and P. Sri Ram Chandra Murthy

Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, A.P., India.

ARTICLE INFO

Article history:

Received: 2 April 2018;
Received in revised form: 28 April 2018;
Accepted: 8 May 2018;

Keywords

AQM,
Congestion Control,
Congestion Avoidance,
Performance Parameters,
Communication Network,
Congestion Detection.

ABSTRACT

As the demand increases for day-day applications, rapid transfer of high amount of data over high speed networks must be required also Bandwidth must be high enough for these applications. Congestion Control is an important subject relevant to these applications to maintain stability for any kind of network. In this paper, review on various congestion control mechanisms and their performance measurement parameters are to be compare with each other. Active Queue Management is one of the method to get control over congestion by dropping packets from buffer queue as an indication to other end node to slow down transfer of packets.

I. Introduction

Too much traffic over subnet in a network will cause congestion. When multiple users competing for finite resources like queue length, buffer size and bandwidth over shared networks the problem of congestion is occurred. As traffic over network increases packets will queue in buffer, if it full, incoming packets are dropped. Either increasing bandwidth or buffer size is not solution for controlling congestion because it causes to more delay in network. Congestion is the blocking of traffic in networks. Congestion takes place where n number of links goes into a single link, such as numbers of internal LAN's are connected to a single WAN link. When packets increased in number as they travel over network than limited capacity of network then packets blocked at router. When routers don't have sufficient capacity to accommodate packets then it leads to congestion. So, congestion occurs at router where nodes are subjected to more traffic than they are designed to handle.

Congestion [1][2] occur when load on network is greater than the capacity of the network due to this, performance of network is degraded. Finally, burst traffic, slow processors, insufficient memory to store arriving packets and when packet arrival rate exceeds the outgoing link capacity are the factors that causes to congestion. Congestion control is different from flow control. Data link layer deals with flow control and it concerns a single sender outrunning a single receiver (point-point link) and it is local. Whereas network layer concern congestion control and it is global. Setting up different design parameters at Data Link Layer, Network layer and Transport layer are used to prevent and eliminate congestion problem. For example, Flooding at DLL is fast but it generates duplicate packets also timer set at DLL for too short which leads to unnecessary retransmission of packets.

II. Congestion Control and Network Performance Parameters

When Load Increases Than The capacity of network then congestion takes place on the network. Due to this- packet loss and delay is increased, through put and bandwidth utilization is reduced finally the overall performance of network will be degraded which leads to deadlock. Congestion control can be categorized into two ways, one is congestion prevention before congestion happening and another one which removes congestion after is has taken place.

Open loop congestion which can be handled by source/destination which can be used to prevent or avoid congestion before it happens and the methods used are retransmission policy, window policy, acknowledgement policy, discarding policy and admission policy. Where as in closed loop congestion control mechanism it tries to eliminate congestion after congestion has happened. It handles congestion in networks based on feedback from different sources. And the policies used in this are back pressure, ChoKe-point, implicit signaling and explicit signaling. Congestion control algorithm performance has been measured using the following parameters [3]:

i. **Fairness:** A fairly distribution of network or system resources among users and applications is specified by the fairness. The Jain's equation states

Fairness = $(\sum_{i=1}^{n} xi)2$ / n. $\sum_{i=1}^{n} xi)2$

This result ranges from 1/n (worst case) to 1(best case). Channels that are not utilized and non-sensitive to network flow patterns are identified by this metric.

ii. **Throughput:** The rate at which the information is transferred that is the number of messages successfully transmitted per time unit and it is managed by available bandwidth, signal-to-noise ratio and hardware limitations.

iii. **Link capacity:** For transferring bits reliably how much maximum throughput a link can offered.

Available capacity = Link Capacity – Utilized Capacity

iv. **Link Utilization:** Ratio between throughput and access rate expressed as a percentage.

Link Utilization (%) = Throughput / Access rate

Access rate is the maximum data rate.

Tele:
E-mail address: ch.ramadevi@gmail.com

v. **Mean Queue Length:** Number of packets accommodated in a buffer from servicing in a network.

vi. **Packet loss Probability:** Ratio between total number of dropped packets and total number of transmitted packets.

Packet loss (%) = total number of dropped packets/total number of transmitted packets. If the fraction is increased transmission rate is reduced then packets sending rate must be reduced to control the congestion. A network with packet loss 5-15% is considered that network is congested heavily.

vii. **Latency:** RTT is used to calculate latency. It is the duration between departure of packet at source and receiving of an acknowledgement that reached its destination.

Latency = RTT $+W_t+P_t$

W_t is the waiting time for Queue at router

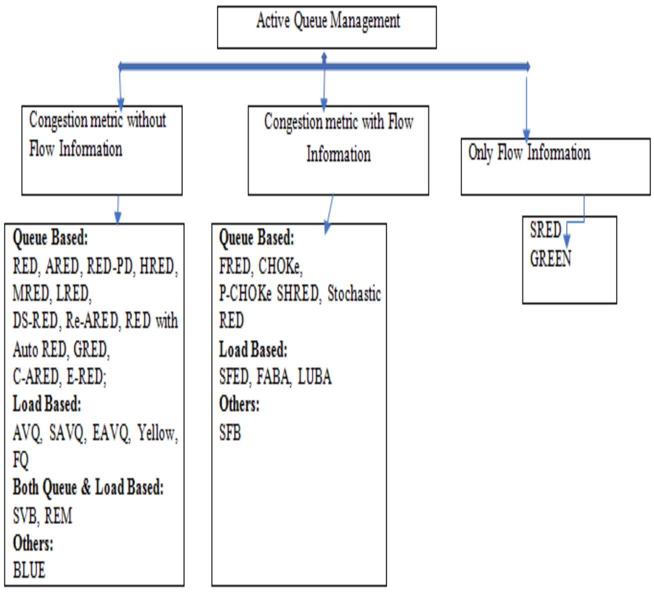
P_t is the time taken for destination to process packet and acknowledge generation.

viii. **Jitter:** When packets travel over network the deviation in the data rate is termed as Jitter. It can be classified as Latency Jitter and Delay Jitter. The deviation corresponding to latency is termed as Latency jitter. And the deviation with respect to the time taken for packets to reach its destination is termed as Delay Jitter.

ix. **Availability:** The amount of time period that network is not available during communication.

x. **Reliability:** Both availability and packet loss are related to reliability. The frequency at which the packets modified or corrupted due to network problem and are different from packet loss. But packet loss includes corrupted along with lost packets.

III. Active Queue Management and Its Classifications



A. Metric without Flow Information AQM method

i. Queue Based:

1. RED (Random Early Detection): It is discussed at [4] in which Avg_{ql} is determined by using EWMA low pass filter and if it less than min_{th} packets never dropped and if it exceeds max_{th} every arrival packets at router will be discarded. In between min_{th} and max_{th}, packets are marked according to drop/mark probability D_p which is a linear function of queue length. RED overcome drawback of global synchronization but is not accurate when load changes dynamically also at the router Avg_{ql} cannot controlled effectively.

2. ARED (Adaptive RED): Is discussed at [5] using AIMD, P_{max} is adapted to maintain Avg_{ql} within a target range in between min_{th} and max_{th} and it is helpful to overcome the stabilizing Avg_{ql} that takes place in RED. ARED reduce the packet loss rate and the variance of the queue size by adapting parameters. The main goal is to improve the average queuing delay by adapting P_{max} periodically. According to studies changes in amplitude of PD-RED queue length is less and changes of D_p is still less than ARED. Re-ARED is variant of ARED in which delay is decreased and throughput is increased.

3. RED-PD (Proportional Derivative-RED): Studied in [6], is based on PD control principal. Flows that utilize more bandwidth will be recorded in drop history and they are prefer to drop and the remaining flows controlled by ordinary RED. If the arrival packet flow suspected as beyond its fair share then packet drop with a flow-specific probability; else consider as too much consuming bandwidth and marked with drop probability according to normal RED. Benefit of RED-PD over RED is fairness and it avoids starvation for monitored flow.

4. H-RED (Hyperbola – RED): is introduced by [7] in which hyperbola curve is used instead of linear as in RED for Drop probability D_P. The reference queue length initialized by user and monitored by control theory of HRED. It is not sensitive to the level of network load and is target for more network utilization because of larger queue size and queuing delays.

5. M-RED (Multilevel RED): is discussed at [8] which uses for differentiated service networks. Packet drop probability D_p is measured independently by maintaining 'N' set of RED parameters for different drop precedence. Different methods used to estimate Avg_{ql} which leads to different variations like RIO-C and RIO-D.

6. L-RED (Loss Ratio Based –RED): introduced by [9], in this packet loss ratio is the clear indication for severe congestion which is used to dynamically adjust packet drop probability. Queue length is also used in small time-scale to make the algorithm more responsive in regulating the length to a target value and LRED tries to combine the response time and packet drop probability there by making its response time almost independent of network status. This combination enables LRED to achieve fast response time and to achieve maximum robustness.

7. DS-RED (Double-Slope-RED): Introduced by [10], based on Avg_{ql} to support smoothly raising drop action both RED and DSRED uses linear drop probability function. And Avg_{ql} is used by two segments drop function which relates to long term congestion level packet drop increase with higher rate and is proportional to congestion rate which gives early warning for long term congestion.

8. Re-ARED (Refined- Adaptive RED): proposed in [11] which enhance the performance of ARED by adjusting max_p it is limited to be in the range [0.01; 0.5]. TCP good put and TCP/UDP packet drop rate are used as performance analysis in Re-ARED. Queue-variation ARED [12] is another variation based on changes to queue size per hour. In this end-end delays reduced, packet dropping is reduced and handle burst traffic dynamically. Re-ARED maintain low delay, high throughput by keeping Average queue size with $q_{min}+0.48(q_{max}-q_{min})$, $q_{min}+0.52(q_{max}-q_{min})$. It maintains constant average queue size, packet drop rate is slightly decreased than RED and ARED.

9. GRED (Gentle RED): It is proposed by [13] in which three congestion indicator parameters used Min_{th}, Max_{th}, $Double_{maxth}$ to stabilize Avg_{ql} than RED when Avg_{ql} crosses

Max_{th} with higher drop probability D_p to avoid buffer overflow. When $Avg_{ql}<Min_{th}$ and $Min_{th}<=Avg_{ql}<=Max_{th}$ the actions for these cases same as RED.

When $Max_{th} < Avg_{ql} <Double_{maxth}$ then packets dropped with D_p and when $Avg_{ql} > Double_{maxth}$ then every packet drop with D_p which is set to 1.

10. C-ARED (Cautious ARED): studied in [14] both ARED and Re-ARED properties are combined together and try to eliminate drawbacks exits in both. When level of congestion changes from light to moderate throughput is reduced because of fixed and conservative approach of adapting max_p in ARED. This drawback rectified in Re-ARED but when traffic load is high it also reduces through put, since no idea about when traffic is heavy and low. C-ARED adapts max_p conservatively or aggressively depends on level of traffic load. When new Avg_q size higher than old Avg_q size traffic load is higher else smaller. According to this C-ARED set variables like min_{th}, max_{th}, w_q and target queue delay to maintain stability and reduce delay and increase throughput.

11. E-RED (Exponential-RED): is proposed by [15] and uses primal-dual algorithm to evaluate dropping variables from optimization theory and is an extension for RED. Dropping/marking probability for arrival packet is as follows:

$$P_m = \begin{cases} 0 & if \quad 0 \le Q_c \le Q_{min} \\ P_m^{min} \cdot e^{\frac{Q_c(Q_c - Q_{min})}{C}} & if \quad Q_{min} < Q_c < Q_{max} \\ 1 & if \quad Q_{max} \le Q_c \end{cases}$$

ii. Load Based:

1.AVQ (Adaptive Virtual Queue): is discussed in [16] and the VQ is modified as same as packet reaches at real queue to signify new packet is arrived. When virtual buffer queue overflows the packets are indicated as dropped/marked. And virtual capacity of link is updated in a way that the entire flow coming each link achieves a desired utilization of the link. This takes place by aggressive marking when the link utilization increased more than desired utilization increased more than desired utilization and less aggressive when link utilization is less than the desired utilization. Two parameters used to implement this algorithm are desired utilization and damping factor. And they evaluate stability of algorithm and achieve low loss with high utilization.

2. SAVQ (Stochastic AVQ): is discussed at [17] in which it stabilizes the dynamics of queue maintain a high link utilization by adaptive setting ϒ according to queue size and given reference instantaneous queue value.

3. EAVQ (Enhanced AVQ): is studied at [18] where the arrival rate at network link used as principal measure of congestion, desired link utilization ratio used as subordinate measure to eliminate problems as difficulty with anti-disturbance and low link capacity losses and to achieve faster dynamic response.

4. Yellow: is proposed in [19] which use the primary parameter which indicates difference between input rate and link capacity. And queue size used as secondary parameter. By queue control function the queue length which is calculated by non-linear hyperbola function affects the load factor. With the introduction of UDP flows Avg_{ql} and standard deviation of queue length of Yellow are affected by load.

5. FQ (Fair Queuing Algorithm): is discussed at [20] and used in Multimedia networks to achieve good fairness and low delay. Weighted Round Robin (WRR) is a frame based and class based on round robin algorithm and serviced for each flow as it is assigned weight. Stochastic Fair Queuing algorithm [26] is used in high-speed computer networks but have some drawbacks with this algorithm are unfair behavior when flows colliding with another flow and drop down its performance under heavy load and unexpected failures.

iii. Both Queue & Load Based:

1. SVB (Stabilized Virtual Buffer): is discussed at [21]. By considering both packet arrival rate and queue size to stabilize with respect to target value. It maintains virtual queue and responds to traffic dynamic faster for better stability against short flows. And results show that it provides minimum loss rate, more stability and high throughput in dynamic workloads when compare to other AQM mechanism like REM, AQM and RED.

2. REM (Random Early Marking): is discussed at [22] which target for high utilization, negligible packet loss and delay. Basic idea is de-coupling of congestion measure from performance measure like packet loss and queuing delay and leads to improve performance of TCP over wireless network.

iv. Others:

1. BLUE: Proposed in [23] which uses packet loss and link idle instead of queue size to manage congestion by drop or marking probability P_m. If queue packet drops rate increase, then BLUE increased P_m indicates the rate at which it sends, back congestion notification and P_m decreases. When queue becomes empty which allows BLUE to "learn" the rate at which it sends back congestion notification? Apart from BLUE use 3 parameters δ_1, δ_2 and freeze time respectively indicates the amount by which P_m is incremented when queue overflow, the detrimental amount when queue is empty and the time gap between 2 consecutive changes in P_m also and δ_1 is set larger than δ_2. Compare to RED packet loss is reduced and keep the buffer in stable.

Congestion metric with Flow Information AQM method

i. Queue Based:

1. FRED (Fair RED): It is proposed by [24] to eliminate non-responsive and misbehaving flows which takes too much bandwidth occurred in RED. Target of FRED is imposing multiple dropping techniques for different kinds of flows. Bursty and low-speed flows should be protected and spared from dropping.

a. Incoming packets that follow condition can be accepted as a fair share on flow i:

($Q_i <= Q_{minAvg}$) and ($Q_{avg} < Q_{maxAvg}$)

b. Packets are dropped according to drop probability D_p when the following condition met:

($Q_{minAvg} < Q_{avg} <= Q_{maxAvg}$), ($Q_i > Q_{i, min}$) and ($Q_i > Q_{avg}$)

Avg_{ql} is updated for every packet arrival and leaving of system.

2. ChoKe Algorithm (CHOose and Keep for responsive flows And CHOose and kill for unresponsive flows): discussed at [25], when packet arrive at congested router a packet is dropped at random and compare with newly arrived packet. If packet belongs to same flow drop the packets. If Avg_q size greater than min_{th} then every arrival packet flow-id and randomly selected packet is compared and packet is dropped if both belongs to same flow. If Avg_q size greater than max_{th} then drop packets else assign with d_p. When no specific data structure is required it is easy and is stateless. For heavy flow this is not suitable, the drawbacks in ChoKe eliminated by this A-ChoKe. Packet drop/mark probability is same as ChoKe, but it doesn't store much information else it may suffer from overhead and become Non-scalable. It uses both queue based and flow information. It reduces packet loss and queue delays and network utilization with well adaptively tuned parameters.

3. P-ChoKe (Piggy Backing-ChoKe): is discussed at [26], multiple senders when sends packets to gateway/router it provides better packet delivery ratio with low queuing delay. But fair bandwidth allocation is not place.

ii. Load Based:

1. SFED: is proposed in [27] can be combined with any scheduling algorithm which is rate control based AQM algorithm and uses token bucket for every flow. When packet arrives, tokens are removed from corresponding buckets whether packets are enqueued or dropped depends on bucket occupancy at that time. And it uses O(N) operations for enqueue and dequeue.

2. FABA (Fair Adaptive Bandwidth Allocation): is proposed in [28], is extension of SFED, rate control based AQM and uses O (1) operations which is also maintains per active-flow state with scalable implementation. In applications like FTP, Telnet and HTTP it maintains high values of fairness. Even in the presence of the non-adaptive flows it provides fairness amongst competing flows. It performs better than RED and CHOKe. In case of buffer sizes constrained, it performs significantly better than FRED. Performance is superior even for a large number of connections passing though the routers.

3. LUBA (Link Utilization Based Approach): is proposed in [29] in which malicious flows at router were identified which causes for congestion. These malicious flows are assigned by drop probability by that it never gets more than its fair share of network. When over load factor U=λ/μ is below target link utilization router is not congested and packets at router are not marked or dropped, otherwise every packet is monitored. For this very reason a history table is maintained to monitor flows which take more than their fair share of bandwidth. Under different network conditions also LUBA works well. And even under large number of non-responsive flows the complexity of algorithm does not increased.

iii. Others:

1. SFB (Stochastic Fair BLUE): is proposed in [30] is a FIFO queueing discipline and is extension to BLUE which is based on Bloom filter that identify and limit the rate of Non-responsive flows; SFB maintains N*L accounting bins where L is the number of levels and N is the number of bins in each level also SFB maintains L Independent Hash functions. Each in maintain marking probability P_m which is updated based on bin occupancy. When packet arrives to queue, it is hashed into one of the N bins in each of the L levels. If the number of packets mapped to a bin goes beyond certain threshold then P_m is increased. And if the number of packets drops to zero at bin then P_m is decreased.

C. Only Flow Information based AQM

1. SRED (Stabilized RED): It is proposed by [31], which overcome the dependency problem found in RED while calculating Avg_{ql} and number of TCP connections. Zombie is a list of size N used to maintain number of active flows, counting variable set 0 initially and timestamp set for every packet arrival without gathering or analyzing state information on individual flows. The packet drop probability for newly arrival packet is takes according to the two formulas:

Where B is total Queue size, P_m^{max} is maximum dropping/marking probability, Q_c is current queue length, $P_{est}(t)$ is a factor for calculating number of active flows, hit(t) is 0 if packet hit in list and is 1 if no hit in zombie list. Due to short queue length, packet loss increases and is proportional to number of active flows.

$$P_{SRED}(Q_C) = \begin{cases} 0 & if \quad 0 \le Q_C < \frac{B}{6} \\ \frac{1}{4} \cdot P_m^{max} & if \quad \frac{B}{6} \le Q_C < \frac{B}{3} \\ P_m^{max} & if \quad \frac{B}{3} \le Q_C < B \end{cases}$$

The probability of dropping/marking is then:

$$P_m^{max}(Q_C) = P_{SRED}(Q_C) \cdot min(1; \frac{1}{(256 \cdot P_{est}(t))^2}) \cdot (1 + \frac{Hit(t)}{P_{est}(t)}$$

2. Green: is proposed by [32] in which it reduces the packet loss and increases link utilization parameters need not be tuned to active optimal benefit in a given situation. The congestion notification probabilities are measured based on number of flows and RTT of individual flow. And drop/mark probabilities are differing from each flow. By using TCP end-host behavior and flow variable it avoids congestion when overflow and avoid congestion from happening.

Some other congestion control method

1. Dec-bit: is proposed by [33], Congestion indication bit in packet header used to provide feedback to sources for control the congestion at router. Congestion indication bit set to one for every arrival of packet in its header when MQL exceeds one. Sender notifies how many congestion bits set to one and take measurements to balance bandwidth with respect to delay and dynamically manage window to eliminate congestion. Drawback is Avg_{ql} is too short and congestion detection indication is not differentiated.

2. AIMD (Additive Increase and Multiplicative Decrease): is discussed at [34] and referred as "dynamic window adjustment" which collectively uses linear growth and exponential reduction of congestion window when congestion takes place. When timeout or acknowledge message received by sender multiplicative decrease starts according to w (t) *b, where b is ½ and w (t) be the sending rate, else increases transmission rate with w(t)+a where a is additive increase factor which is greater than 0.

3. ATM-RED (Asynchronous Transmission Mode RED): It is proposed by [35], for performance reasons when TCP supported by ATM only cells (48 bytes) should be retransmitted when packet corrupted instead of retransmitting the whole packet (1500 bytes). PPD, EPD, SPD and FBA are former to ATM-RED and all these methods target to improve throughput, Bandwidth utilization and fair distribution of TCP flows. In ATM-RED, the packet drops probability D_p =1- (1- P_c)^n when P<<1, where P_c is drop probability for cells and n is the total number of cells in packet.

4. NL-RED (Non-linear RED): studied in [36], instead of using linear packet drop function as RED, NLRED uses Non-linear quadratic function to drop packets when Avg_q length exceeds min_{th}. When traffic load is low it is smoother than RED but more aggressive when load increases. It achieves higher throughput and more sensitive to setting like threshold values and Avg_q sizes.

5. NN-RED (Neural Network -RED): is studied in [37] which calculate future predicted values of queue size and if value exceed target value t_v then mark/drop probability used to indicate packets as congested packets. It is based on neural network since it used as a future prediction tool. By using this information router send prior information to source regarding probable congestion and eliminate congestion in network before it takes place.

6. Drop Tail: is a non-adaptive simple AQM method proposed by [38] used in many routers but not efficient AQM method.

When arrival rate at input link exceeds sending rate of output link, it simply discards packets from tail of queue. When source detect packet loss they slow down the arrival rate of packets to queue will be less than the capacity of the link and the packet backlog in the queue decreases. But the drawbacks are lack of fairness, Lockout, burst traffic which reduces throughput, link utilization and finally QoS is affected. QoS must be maintained for constant transmission of high- bandwidth video and multimedia information. This type of transmitting the content is difficult in the current Internet and network through DT.

IV. Conclusions

In this paper, we presented survey on various AQM algorithms and we conclude the performance of rate based AQM scheme is better than Queue based scheme. Performance comparison parameters like packet loss, throughput utilization and link utilization were focused mainly. The basic task of AQM is classified into Congestion Monitor, Bandwidth Controller, Congestion Controller and Queue Controller; And the taxonomy of AQM is based on these four tasks.

Acknowledgments

I am profoundly grateful to my project guide, **Mr. P. Sri Ram Murty**, who devoted his valuable time and helped me in all possible ways towards successful completion of this work and I am so thankful for giving valuable advice and enthusiastic guidance. I express my gratitude to all the faculties for the interest that has shown in this survey. Last but not the least, I wish to avail myself of this opportunity, to express a sense of gratitude and love to my family members for their manual support, strength, help and for everything.

References

[1] A. D. Gaya, H. B. Danbatta, A.G. Musa, “Survey on Congestion Control in High Speed Network”, IJARCSSE, vol. 5, no. 3, pp. 1237-1244, 2015.

[2] D. Papadimitriou, "Open research issues in Internet Congestion Control", Internet Engineering Task Force, RFC6077, 2011.

[3] Manjaragi S. V., “Comparative study of link utilization, queue, delay and loss Characteristics of Active queue management Schemes in Wireless networks”, IJERT, vol. 2, no. 4, pp. 2465-2471, 2013.

[4] S. Floyd, V. Jacobson, "Random Early Detection Gateway for congestion avoidance", IEEE/ACM Transaction on Networking, vol. 1, no. 4, pp. 397-413, 1993.

[5] S. Shankar, S. Floyd, R. Gummadi, "Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management", 2001.

[6] R. Mahajan, S. Floyd, and D. Wetherall. Controlling high-bandwidth flows at the congested router, 2001.

[7] Liujia Hu., Ajay D. Kshemkalyani., “HRED: A simple and Efficient Active Queue Management Algorithm” 13th International Conference on Computer Communications and Networking ICCCN 2004, October 2004

[8] Jahoon Koo., Byunghun Song., Kwangsue Chung., Hyukjoon Lee., Hyunkook Kahng.,” MRED: A New Approach to Random Early Detection” 15th International Conference on Information Networking, February 2001

[9] Chonggang Wang., Bin Liu., Y. Thomas Hou., Kazem Sohraby., Yu Lin., “LRED: A Robust Active Queue Management Scheme Based on Packet Loss Ration “23rd Annual Joint conference of IEEE Computer and Communication Societies INFOCOM, March 2004

[10] Zheng B., Atiquzzaman M. Dsred: An Active Queue Management Scheme for Next Generation Networks, In: Proceedings of the 25th Annual IEEE Conference on Local Computer Networks, LCN-00. Washington DC, USA, IEEE Computer Society, 2000.

[11] Tae-Hoon Kim., Kee-Hyun Lee” Refined Adaptive RED in TCP/IP Networks”, IEEE ICASE, October 2006

[12] Jeong-Hwan Seol, Ki Young Lee., Yoon Sik Hong “Performance Improvement of Adaptive AQM Using the Variation of Queue Length” IEEE Region 10 Conference TENCON, November 2006

[13] S. Floyd, "Recommendations on Using the Gentle Variant of RED," in http://www.aciri.org/floyd/red/gentle.html, 2000.

[14] Tahiliani M. P., K. C. Shet A, T. G. Basavaraju et al., CARED: Cautious Adaptive RED gateways for TCP/IP networks ELSEVER Journal of Network and Computer Applications. (2011): 10. 1016 /j. jnca. 2011.12.003.

[15] Shao Liu, Tamer Basar, and R. Srikant. Exponential-red: a stabilizing aqm scheme for low- and high-speed tcp protocols. IEEE/ACM Trans. Netw., 13(5):1068–1081, 2005.

[16] Srisankar S. Kunniyur., R. Srikant “An Adaptive Virtual Queue (AVQ) for Active Queue Management”., IEEE/ACM Transactions on Networking, April 2004

[17] Cheng-Nian long., Bin Zhao., Xin-Ping Guan., “SAVQ: stabilized Adaptive Virtual Queue Management Algorithm”., IEEE Communications Letters., January 2005

[18] Qian Yanping, Li Qi, Lin Xiangze, Ji Wei,” A Stable Enhanced Adaptive Virtual Queue Management Algorithm for TCP networks” May30 to June 1,2007, IEEE International Conference on Control and Automation

[19] Chengnian Long., Bin Zhao., Xinping Guan., Jun Yang.,” The Yellow active queue management algorithm”, Computer Networks, November 2004

[20] N. Pekergin. Stochastic Bounds on Delays of Fair Queueing Algorithms. Technical Report PRISM, UVSQ 10, University“ de Versailles-St-Quentin, 1998.

[21] Xidong Deng., Sungwon Yi., George Kesidis., Chita R. Das., “Stabilized Virtual Buffer (SVB)-An Active Queue Management Scheme for Internet Quality of Service”, IEEE Globecom November 2002.

[22] Athuraliya., D.E Lapsley., S.H Low” Random Exponential Marking for internet congestion control” IEEE Transactions on Network, June 2001

[23] Wu –Chung Feng, Kang G. Shin, Dilip D. Kandlur, Debanjan Saha, "The Blue Active Queue Management Algorithm", IEEE ACM Transaction on Networking, 2002.

[24] D.Lin., R. Morris.,” Dynamics of Random Early Detection”, Proceedings of ACM SIGCOMM, October 1997

[25] A Stateless Active Queue Management Scheme for Approximating Fair Bandwidth Allocation. Rong Pan, Balaji Prabhakar, and Konstantinos Psounis. ChoKe, IEEE INFOCOM, Mar 2000

[26] G. Sasikala, E. George Dharma Prakash Raj, “P-ChoKe: A Piggybacking-ChoKe AQM Congestion Control Method”, IJCSMC, Vol. 2, Issue. 8, August 2013.

[27] A. Kamra, S. Kapila, V. Khurana, V. Yadhav, H. Saran,” SFED: a rate control based active queue management discipline, IBM India research laboratory Research Report, available online from http://www.cse.iitd.ernet.in/srajeev/publications.htm

[28] Abinav Kamra., Huzur Saran., Sandeep Sen., Rajeev Shorey “Fair Adaptive Bandwidth allocation: a rate control

based active queue management discipline", Computer Networks, July 2003

[29] Manoj K. Agarwal., Rajeev Gupta., Vivekanad Kargaonkar., "Link Utilsation Based AQM and its Performance", IEEE Communications Society, Globecom 2004, December 2004

[30] Wu –Chung Feng., Kang G. Shin., Dilip D. Kandlur., Debanjan Saha., "The BLUE Active Queue Management Algorithms "IEEE ACM Transactions on Networking, August 2002

[31] Shao Liu, Tamer Basar, and R. Srikant. Exponential-red: a stabilizing AQM scheme for low- and high-speed tcp protocols. IEEE/ACM Trans. Netw., 1068–1081, 2005.

[32] W. Feng, Apu Kapadia, Sunil Thulasidasan, "GREEN: Proactive Queue Management over a Best-Effort Network", IEEE GlobeCom (GLOBECOM 2002), Taipei, Taiwan, November 2002.

[33] K. K. Ramakrishnan, R. Jain, "A Binary Feedback Scheme for congestion avoidance in computer network", ACM Transaction on computer system, vol. 8, no. 2, pp. 158-181, 1990.

[34] Chiu, Dah-Ming; Raj Jain, "Analysis of increase and decrease algorithms for congestion avoidance in computer networks". Computer Networks and ISDN systems. 17: 1–14, 1989.

[35] Vincent Rosolen, Olivier Bonaventure, and Guy Leduc. A red discard strategy for ATM networks and its performance evaluation with tcp/ip traffic. SIGCOMM Comput. Commun. Rev., 29(3):23–43, 1999.

[36] Kaiyu Zhou, Kwan L. Yeung, Victor O. K. Li, Nonlinear RED: A Simple Yet Efficient Active Queue Management Scheme, K. Zhou et al. / ELSEVER Journal of Computer Networks 50, (2006) 3784–3794.

[37] B. Hariri and N. Sadati. NN-RED: an AQM Mechanism Based on Neural Networks, Electronics Letters 2007, 43:1053–5.

[38] F. Postiglione, G. De Marco, M. Longo, "Run Time Adjusted congestion control for multimedia experimental results", Journal of Interconnection Network, vol. 5, no. 3, pp. 249-266, 2004.

AUTHORS PROFILE

Ms. Ramadevi received B. Tech degree in computer Science from Kakatiya University, and M. Tech degree in computer science from JNTUH, in 2004 and 2009 respectively. Currently doing her research from ANU. Her area of research is Congestion control methods in heterogeneous networks and also interested in Networks & security and Network applications.

Dr. Patnala S.R. Chandra Murty, Assistant Professor with Department of Computer Science and Engineering, Acharya Nagarjuna University. He has received his Ph.D. from JNTUK in 2013. He is having many national and international publications also participated in international conferences. His current research interest includes Digital Image Processing, Data Mining and Network Security.