International Journal of Computer Applications Technology and Research
Volume 3– Issue 7, 415 - 419, 2014, ISSN: 2319–8656

RED: A HIGH LINK UTILIZATION AND FAIR ALGORITHM

Neha
Department of CE
UCOE, Punjabi University Patiala
Punjab, India

Abhinav Bhandari
Department of CE
UCOE Punjabi University Patiala
Punjab, India

Abstract— Internet and its applications are an integral part of our daily life .These days they are widely used for various purposes such as communication, public services, entertainments, distant educations, etc., each possessing different quality of service (QoS) requirements. How to provide finer congestion control for network emerges as a major problem. To prevent the problem of congestion control and synchronization various active queue management (AQM) techniques are used. AQM algorithms execute on network routers and detect initial congestion by monitoring some functions. When congestion occurs on the link the AQM algorithms detects and provides signals to the end systems. Various algorithms have been proposed in recent years but RED is one of the most influential techniques among all the existing ones. This review paper provides the functioning mechanism of the RED technique with the help of its algorithm & its variants.

Keywords- AQM ,RED ,RED parameters, RED algorithm, RED variants

1 . INTRODUCTION

The Internet intended for openness and scalability in its beginning. In Internet congestion occurs when the total demand for a resource (e.g. link bandwidth) exceeds the existing capacity of the resource. It is necessary to stay away from high packet loss rates in the Internet. In communication networks, available bandwidth and network routers plays an most important role during transmission of data packets. As a result, the routers have to be designed in such a way that they can survive large queues in their buffers in order to accommodate for transient congestion. Earlier the congestion in the network is detected after the packet has been dropped. When a packet is dropped before it reaches its destination, all of the resources it has consumed in transit are wasted. In extreme cases, this situation can lead to congestion collapse [4].But with the time to meet the demands of the network and for providing better throughput various active queue management techniques come into existence. The basic idea of active queue management is used to detect congestion as well as to control the queue size before the buffer overflows. When transmission control protocol (TCP) detects packet losses in the network TCP decreases packet sending rate .since the congestion at a router is relieved due to this decrease of the packet sending rate of TCP, AQM mechanism helps to prevents Buffer flow. AQM mechanisms control the queue length (i.e., the number of packets in the router’s buffer) by actively discarding arriving packets before the router’s buffer becomes full. [11] Figure 1 shows the AQM mechanism. To solve the problem of congestion in internet the Internet Engineering Task Force (IETF) has suggested the use of RED [2], [3], an active queue management scheme which is able to achieve high throughput and low average delay (for TCP traffic) by spreading randomly packets drops between flows.

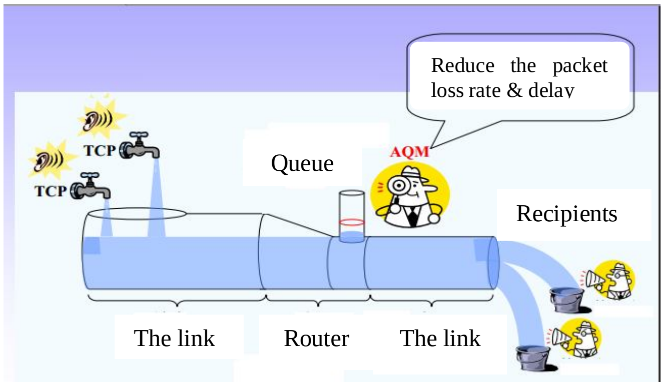

Fig.1. Active Queue Management Mechanism

Among various AQM scheme, Random Early Detection (RED) [8] is probably the most extensively studied. RED is shown to effectively tackle both the global synchronization problem and the problem of bias against bursty sources. Due to its popularity, RED or its variants has been implemented by many router vendors in their products.

This paper is organized as follow; section 1 gives the introduction about congestion control with AQM technique (RED) Section 2 gives a view about the RED technique. Section 3 describes the RED algorithm with its parameters and diagrammatically explains the working of RED algorithm .Section 4 gives a view about the different variants of RED. Section 5 Conclusion & Section 6 References.

2. RED (RANDOM EARLY DETECTION)

Random Early Detection (RED) was first proposed AQM mechanism and is also promoted by Internet Engineering Task

Force (IETF) as in [2]. Random Early Detection (RED) was introduced in 1993 [3] by Floyd and Jacobson. RED provides congestion avoidance by controlling the queue size at the gateway [3]. RED is customized for TCP connection across IP routers it's considered to avoid congestion It notifies the cause before the congestion truly happens rather than wait till it actually occurs. It provides a method for the gateway to provide some feedback to the resource on congestion status. In order to solve the problem of passive queue management technique this section proposes and evaluates a primarily different queue management algorithm called RED. RED is designed to be used in conjunction with TCP, which currently detects congestion by means of timeouts (or some other means of detecting packet loss such as duplicate ACKs) [10]. RED has been designed with the objective to:

- reduce packet loss and queuing wait,
- Avoid global synchronization of sources
- Maintain high link utilization, and
- Remove biases against bursty sources

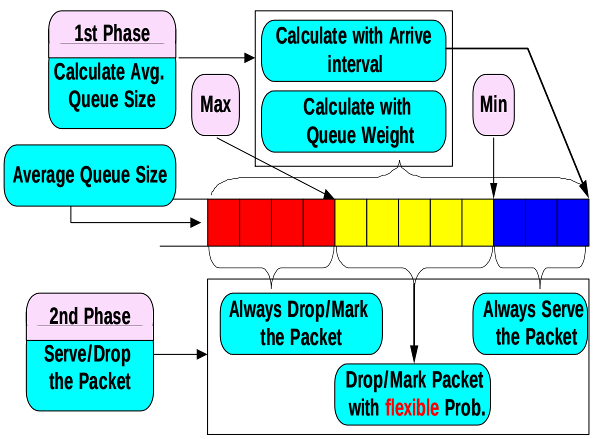

Fig.2. RED mechanism

Fig 2 represents that RED monitors the average queue size and marks packets. If the buffer is almost empty, all incoming packets are accepted. As the queue grows, the probability for dropping an incoming packet grows too. When the buffer is full, all incoming packets are dropped. Thus RED buffer mechanism works with constant bit rate (CBR) traffic can be used at an early stage to know the effect of change of network parameters over system performance. The main aim of RED is to control the queue size and indicating the end hosts when to slow down their packet transmission rate. It takes benefit of the congestion control mechanism of TCP by randomly dropping packets earlier to periods of high congestion, RED tells the packet source to reduce its transmission rate. Assuming the packet source is using TCP, it will reduce its transmission rate until all the packets reach their destination, representing that the congestion is cleared.

3. RED ALGORITHM

The RED algorithm is congestion avoidance scheme used in communication network to keep away from congestion. Compared to other algorithms this avoids congestion at common network bottlenecks, where the system triggers before any congestion actually occurs. RED performance is highly sensitive to its parameter settings ([10]). Table1 represents the various parameters included in RED algorithm.

Table1: Parameters of RED algorithm

Parameter	Meaning
Maxth	Maximum threshold
Minth	Minimum threshold
Maxp	Maximum packet Dropping/Marking Probability
Wq	Weighting factor
P_a	Probability
Avg	Average queue length

RED algorithm differentiate between the temporary and persistent congestion in the network, so as to propose the network to hold busty traffic, rather than shaping busty traffic, in which the average queue length is calculated for each packet arrival. RED has two types of queue length threshold activities: Minth and Maxth. As the packet arrives at a router, RED calculates the new average queue size Avg and compares it with the these two thresholds, and then takes actions according to the given rules: if the average queue length is smaller than the minimum threshold (Avg < Minth), no action is taken; if the average queue length is larger than the maximum threshold (Avg > Maxth), the packet is always dropped; if the average queue length lies between the two thresholds , the newly arriving packet is dropped with some probability(P_a) as shown in figure 3:

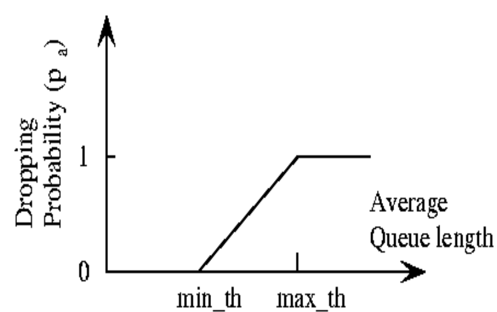

Fig.3 Droping probability in RED

The droping probability of RED algorithm Can be represented as shown in the Figure3. The general RED algorithm can be presented as follows in the figure 4 and the parameters used in the algorithm are defined in Table 1

For each new arrival of packet:
Compute the average queue length;

If (Minth ≤ Avg < Maxth)
{
Calculate the probability Pa,
with probability Pa: mark/drop the arriving packet
}
Else if (Maxth ≤ Avg)
{
mark/drop the arriving packet
}
Else
{
Do not mark/ drop the packet
}

Fig.4. General Algorithm of RED

The average queue size at arrival of a new packet can be calculated by using the formula i.e.
Avg = (1 – weight) X Avg + weight x currQ

Where 0 < Weight < 1
currQ is the current queue length

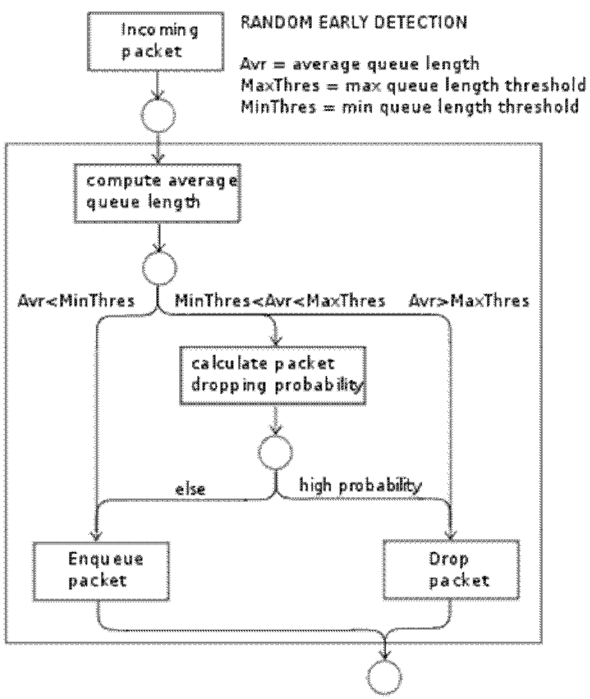

Fig.5. Working of RED algorithm

Random early detection is a queue management scheme that is proposed to respond the shortcomings of drop tail. RED perfectly notifies one of sources of congestion by randomly dropping an arriving packet. The selected source is informed by the packet loss and its sending rate is reduced accordingly. Therefore, congestion is alleviated. It is an early congestion declaration. The dropping probability is a function of average queue length. When the queue tenure grows, congestion builds up. Then, the dropping probability increases in order to supply enough early congestion notifications another goal of RED is to eradicate biases against busty sources in the network. This is done by limiting the queue use so that there is always room left in the queue to buffer transient bursts. In addition, the marking purpose of RED takes into account the last packet marking time in its calculations in order to reduce the probability that successive packets belonging to the same burst are marked. Fig.5 represents the working flowchart for RED algorithm.

4. VARIANTS OF RED

Many variants of RED have been proposed in the past out of all few are briefly explained in the following section. Random Early Detection (RED) was first proposed AQM mechanism and is also promoted by Internet Engineering Task Force (IETF) as in [2]. Random Early Detection (RED) was introduced in 1993 [3] by Floyd and Jacobson and then many variants were also proposed. It is a representative AQM mechanism, have been tenderly studied by numerous researchers. When a packet arrives at a router, the RED router calculates the average queue length. The RED router drops the arriving packet with the probability which is calculated from the average queue length and a configuration of control parameters. If the average queue length becomes large, the change of the packet drop probability of RED will become unstable.
To overcome the drawbacks of RED these different variants comes into existence. All variants depends on RED parameters i.e. average queue length, minimum threshold, maximum threshold and dropping probability in dealing with congestion and achieving the maximum Quality of service .It provide a mechanism to control the congestion collapse in the internet.Following is the brief description of variants of RED that are widely studied to overcome the weaknesses of the general RED algorithm.

4.1 RED Flow Trust (REDFT)

It is a scalable AQM scheme, based on flow trust .It could be used to counter low rate DoS attacks as well as Flooding rate DoS attacks. In this each flow behaviors' are monitored and according to the data collected trust values are calculated.

4.2 Non Linear RED (NLRED)

NLRED is the same as the original RED except that the linear packet dropping probability function is replaced by a nonlinear quadratic function. While inheriting the simplicity of RED, NLRED was shown to outperform RED. In particular, NLRED is less sensitive to parameter settings, has a more predictable average queue size, and can achieve a higher throughput.

4.3 Hybrid RED (HRED)

The Hybrid RED algorithm will effectively change the transfer function of the overall control loop and thus the stability of network. Hybrid RED algorithm gives a better loss rate and link utilization compared to the existing RED and LPF/ODA algorithms. Two modifications to RED are proposed: (i) use of both instantaneous queue size and its Exponential Weighted Moving Average (EWMA) for packet marking/dropping and (ii) reducing the effect of the EWMA queue size value when the queue size is less than minth for a certain number of consecutive packet arrivals.

4.4 Flow Random Early Drop (FRED)

It introduces "per-active-flow accounting" to enforce a drop rate on each flow that is dependent on the flow's buffer occupancy. FRED has two parameters MINq(i) and MAXq(i) which are the minimum and maximum numbers of packets that each flow (i) is allowed to buffer. FRED uses a global variable (avgcq) to calculate approximately the average per-active flow buffer usage. It maintains the number of active flows for each of them, FRED maintains count of buffer packets and a count of times when the flow is responsive (Qlen>MAXq).FRED requires a certain amount of overhead for per active flow counting.

Following is the tabular representation of variants of RED as shown below in table 2

Table 2 Variants of RED

Sr. no	**Name of the Variant**	**Year**	**Aut-hor**	**Advantages**	**Remarks**
1	RED-FT(flow trust)[7]	2013	Jiang et.al	It employs the flow trust to safeguard legitimate flows. It improves the throughput and delay in DoS attacking scenarios.	Potential work is required to enhance the detection accuracy & propagation of flow trust in networks
2	NLRED(Non linear RED) [13]	2009	Wang et.al	In this scheme packet dropping becomes gentler than RED at light traffic load. NLRED achieves a higher and more stable throughput than RED.	Packet dropping becomes more aggressive at heavy load
3	HRED (Hybrid RED)[9]	2008	Haider et.al	HRED shows better utilization of network bandwidth and a lower packet loss rate	Hybrid RED under different traffic mixes such as exponential ON/OFF traffic over UDP with TCP, is still uncovered.
4	FRED(Flow Random Early Drop)[5]	1997	D.Lin and R. Morris	It protects from fragile flows and maintain high degree of fairness by using per active flow accounting.	It requires a certain amount of overhead for per active flow counting.

5.CONCLUSION

RED is the most known active queue management mechanism that is widely studied by many researchers. The best benefit of using RED is instead of considering instant queue length RED uses average queue length to drop a packet. The performance of RED may get worse in many situations but the different variants of RED help to avoid the problems that are faced by the general RED algorithm. Many refinements to fundamental algorithm make it more adaptive and require less variation.

6. REFERENCES

[1]V. Misra, W. B. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," ACM SIGCOMM Computer Communication Review, Vol. 30, pp. 151-160, 2000.

[2] B. Braden, D. Clark, J. Crow croft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclaw ski and L. Zhang RFC 2309: Recommendations on Queue Management in April 1998

[3] S. Floyd and V. Jacobson, "Random early detection gateway for Congestion avoidance,"IEEE/ACM Transaction on Networking, vol. 1, no.4, pp.397-413, Aug. 1993.

[4] V. Jacobson. Congestion Avoidance and Control. In Proceedings of ACMSIGCOMM, pages 314–329, August 1988.

[5] Dong Lin, Robert Morris, ``Dynamics of random early detection''. ACM, Computer Communication Review, vol.27, no.4, Oct. 1997, pp.127-37, USA

[6] S. Floyd, "Recommendations on using the gentle variant of RED," May 2000. Available at http://wwww.aciri.org/ floyd/red/gentle.html..

[7] Xianliang Jiang,Yang ,Jin and Wei Wei, "RED-FT:A Scalable Random Early Detection Scheme with Flow Trust against DoS Attacks,"IEEE Communication, vol. 17. No. 5 MAY2013

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

[9] Aun Haider, and Richard J. Harris "A Hybrid Random Early Detection Algorithm for Improving End-to-End Congestion Control in TCP/IP Networks" African Journal of Information and Communication Technology, Vol. 4, No. 1, March 2008.

[10]Larry L. Peterson, Bruce S. Davie, Computer networks: a systems approach, 2nd edition, San Francisco: Morgan Kaufmann Publishers, 2000.

[11] B. Braden et al., "Recommendations on queue management and congestion avoidance in the Internet," *Request for Comments (RFC) 2309*, Apr. 1998.

[12] Floyd, S., R. Gummadi, and S. Shenker, Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Preprint, available at http://www. icir.org/floyd/papers. html, August, 2001.

[13] Teresa Álvarez, Virginia Álvarez, Lourdes Nicolás, UNDERSTANDING CONGESTION CONTROL ALGORITHMS IN TCP USING OPNET, Spain, 2010.

[14] G.F.Ali Ahmed, Reshma Banu, Analyzing the performance of Active Queue Management Algorithms, International journal of Computer Networks & Communications (IJCNC), Vol.2, No.2, March 2010.

[15] Chengyu Zhu, O.W.W. Yang, J. Aweya, M. Ouellette, Montuno, A comparison of active queue management algorithms using the OPNET Modeler, Communications Magazine, IEEE, volume 40, Pages: 158 – 16, June 2002.

[16] G.Thiruchelvi et al., A Survey on Active Queue Management Mechanisms, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.12, December 2008.

[17] Sunitha Burri, BLUE: Active Queue Management CS756 Project Report, May 5, 2004.

[18] Michael Welzl, Leopold Franzens Network Congestion Control Managing Internet Traffic, University of Innsbruck.