IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 5, May 2010
ISSN (Online): 1694-0784
ISSN (Print): 1694-0814

Performance Comparison between Active and Passive Queue Management

Arash Dana[1] and Ahmad Malekloo[2]

[1] Department of Electrical Engineering, Islamic Azad University, Central Tehran Branch
Tehran, Iran

[2] Department of Computer Engineering, Islamic Azad University, Arak Branch
Arak, Iran

Abstract

The queue management algorithm, which is applied to a router, plays an important role in providing Quality of Service (QoS). In this paper we present a simulation-based performance evaluation and compare two popular queue management methods; Random Early Detection (RED) and Drop Tail, in terms of queue size, queuing delay, queuing delay variation, packet drop rate and bandwidth utilization. We also study the effect of buffer behavior on each one of these QoS measurements, as well as considering the effects of using ECN on RED Performance. Simulation results indicate that RED performed better related to Drop Tail in terms of queuing delay, queuing delay variation, and (lower) packet drop rate. Furthermore, using ECN improves RED's bandwidth usage and packet drop rate. Also by measuring the router CPU utilization, we indicate that the difference between active and passive queue management algorithms in terms of CPU utilization, is mostly depend on the buffer size and the network traffic load.

Keywords: *Queue Management, Random Early Detection, QoS, Congestion Control.*

1. Introduction

Recently, the exponential growth of the Internet and the use of new services such as e-business, voice over IP (VoIP) and multimedia applications has been risen the need of supporting Quality of Service (QoS) requirements. The queue management algorithm, which is applied to a router, plays an important role in all of QoS measurements (include delay, jitter, bandwidth and packet loss). Queuing management algorithm is responsible for accepting the arriving packets or not accepting them and consequently it directly affects the packet loss quantity parameter. The queue management algorithm also should ideally keep the queue occupation level as low as possible, to ensure low delay. However, to ensure maximum utilization of the outgoing link, the queue should never be empty. Furthermore, maintaining queue stability is important as some applications are sensitive to jitter.

As a result, until now many queue management schemes have been introduced in the literature in order to avoid congestion and improve the QoS in the network. But there are only a few algorithms which standardized and implemented. Drop Tail is the most widely used queue manage algorithm in today's IP networks due to its simple implementation [1]. While Random Early Detection (RED) [2], an active queue management scheme, is available in most routers nowadays, it is often disabled by default [3]. RED has been recommended by the Internet Engineering Task Force (IETF) as the default active queue management scheme for the next generation networks [4, 5]. Almost all other active queue management schemes are presented as improvements on RED.

The main aims of this paper are i) to compare two popular queue management methods, RED and Drop Tail, in terms of queue size, queuing delay, queuing delay variations, packet drop rate and bandwidth utilization; ii) considering the effects of using ECN mechanism on RED algorithm Performance; iii) considering the effect of buffer behavior on QoS measurements; iv) to compare and find the difference of CPU utilization between active and passive queue management algorithms.

The Rest of the paper is organized as follows. In section 2 we will review Drop Tail and RED algorithms briefly. In section 3 we will explain the ECN mechanism briefly. Section 4 will present our simulation design, parameter settings, and simulation results. Finally, we conclude our paper in section 5.

2. Active and Passive Queue management

From the point of dropping packets, queue management can be classified into two categories; Passive Queue Management (PQM) and Active Queue Management (AQM). Drop Tail is a representative PQM algorithm which only sets a maximum length for each queue at the router. When the queue length is smaller than the maximum length, all packets are accepted, and if the queue reaches its maximum length all subsequent incoming packets are dropped until queue length decreases to be less than the maximum length. It was shown in [6] that under heavy load conditions, Drop Tail routers cause global synchronization, a phenomenon in which all senders sharing the same bottleneck router/link shut down their transmission windows at almost the same time.

To solve this problem, AQM was proposed. AQM techniques try to detect and react to congestion before its consequences such as queuing delay and packet loss. One well known AQM algorithm is random early detection (RED) [2] which has been recommended by the IETF as the default AQM scheme for routers of next generation networks (NGN) [4, 5]. The basic idea of RED is that a router detects congestion early by computing the average queue length avg, and sets two buffer thresholds; maximum threshold (max_{th}) and minimum threshold (min_{th}) for packet drop. The average queue length at time t, is computed using an exponentially weighted moving average (EWMA) of the previous queue length, as shown in equation 1:

$$avg(t) = (1 - W_q) \times avg(t-1) + W_q \times Q(t) \qquad (1)$$

where Q is the current queue length and W_q is a weight parameter, $0 \leq W_q \leq 1$. W_q determines how rapidly avg changes in response to changes in Q.

The two thresholds are used to establish three zones. When the average queue length is smaller than the min_{th}, RED is in the normal operation zone and all packets are accepted. On the other hand, if the average queue length is larger than the max_{th}, RED is in the congestion control region and all the arriving packets will be dropped. If the average queue length is between both thresholds, RED is in the congestion avoidance region and the packets are discarded with a given drop probability. The probability that a packet arriving at the RED queue is dropped depends on the average queue length, the number of packets since last discard, and the maximum drop probability parameter max_p. The drop probability Pa is computed as:

$$Pa = \frac{P_b}{(1 - count\ .\ P_b)} \qquad (2)$$

where count is the number of accepted packets since the last packet discard and P_b is a temporary probability, given by:

$$P_b = \max{}_p * (avg - min_{th}) / (max_{th} - min_{th}) \qquad (3)$$

whose maximal value given by max_p is reached when the average queue length is equal to max_{th}. For a constant average queue length, all incoming packets have the same probability to get dropped. As a result, RED drops packets in proportion to the connections' share of the bandwidth. The RED algorithm is shown in Fig 1.

```
for each packet arrival
 compute the average queue size avg
 if min_th ≤ avg < max_th
    calculate probability P_a
    drop packet with probability P_a
 else if max_th ≤ avg
    drop the arriving packet
```

Fig. 1 RED algorithm.

3. Explicit Congestion Notification (ECN)

ECN [7] is a bit-based end-to-end congestion avoidance mechanism proposed for TCP/IP networks and incorporated into RFC 2481. RED can be easily extended to support ECN. ECN extends RED enabled routers to mark packets instead of dropping them when the avg is between min_{th} and max_{th}. This mechanism turns out to be useful to protocols like TCP that are sensitive to even a single packet drop. Upon receipt of marked packets, the TCP receiver informs the sender (in the subsequent ACK) about incipient congestion. This will in turn trigger the congestion avoidance algorithm once per window. ECN requires support from both the routers and the end hosts.

With ECN traffic, routers have to identify whether a packet is ECN capable or not. Routers mark only ECN capable packets when congestion occurs. This requirement has resulted in allotting two bits in the IP header. The ECN Capable Transport (ECT) bit is set by the sender end system if both the end systems are ECN capable. This is confirmed during the TCP's pre-negotiation connection setup phase. On congestion, the router marks the ECT packets using the Congestion Experienced (CE) bit on their way to the receiver, with a probability proportional to the average queue size as done in RED [8].

4. Simulation Results

The simulations presented in this paper were performed with the OPNET [9] network simulation tool. To enable a fair comparison of our work with previous works, we have used the simulation configuration shown in Fig. 2, which has been used by previous researchers [1, 8, 10]. There are two routers connected by a DS1 link and all other links are 10BaseT. Five clients run FTP sessions to the five servers (chosen randomly) across the routers. The link between router A and router B is made a low bandwidth link to create enough congestion to analyze the effect of Drop Tail, RED and RED-with-ECN algorithms in router A. The RED parameters were set as follows, as recommended in the original paper [2], $min_{th} = 25$, $max_{th} = 75$, $W_q = 1/2^9$ and $max_p = 0.1$. The maximum queue size at the congested router was set to 100 packets. Simulations were ran for 10 minutes.

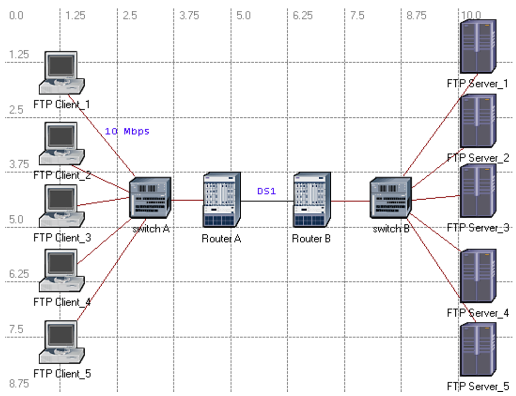

Fig. 2 Simulation topology.

In this experiment we will compare two popular queue management algorithms, RED and Drop Tail, in terms of queue size, queuing delay, queuing delay variations, packet drop rate and bandwidth utilization. We will also study the effect of buffer behavior on QoS measurements as well as considering the effects of using ECN on RED algorithm Performance. Furthermore, we try to find the difference of utilization of router processor (CPU utilization) between active and passive queue management algorithms.
Figure 3 indicates the buffer utilization of the queue management mechanisms versus simulation time (buffer behavior).

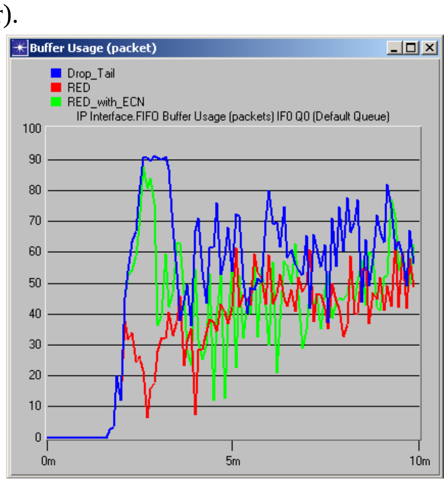

Fig. 3 Buffer usage vs. simulation time.

To study the effects of buffer behavior on QoS measurements, such as delay and dropping packets, we predict each one of QoS measurement according to Fig. 3, and then compare our predictions and actual simulation results.

4.1 Comparing the queuing delay

As it is observed from Fig. 3, Drop Tail has the maximum buffer usage during the simulation period related to the other queue management algorithms. This is because of the reason that in this method, new packets are accepted until there was still a packet to entrance to the queue and queue didn't reach its maximum length. Hence we expected that this algorithm will have the maximum queuing delay.
The averages queuing delay of these three algorithms were presented in Fig. 4. According to this figure, as we have expected, RED algorithm due to its random packet dropping mechanism, has the minimum queuing delay and Drop Tail algorithm due to its high buffer utilization, has the maximum queuing delay.
As it is observed from Fig. 3 and Fig. 4 using the ECN mechanism will increase the buffer utilization and consequently increasing the queuing delay. The reason is that, as it was said before, when the average queue length exceeds the minimum threshold, RED-with-ECN instead of dropping packet, set a specific bit in the header of that packet and forward it to the receiver. Upon receiving the packet, the receiver will set another bit in its next ACK and notify the sender to reduce its transmission rate. According to Fig. 4 by increasing the simulation time, queuing delay for both RED and RED-with-ECN are going to be in close. It is because of the fact that by increasing the simulation time, TCP senders receive ACK massages and reduce their transmission rates.

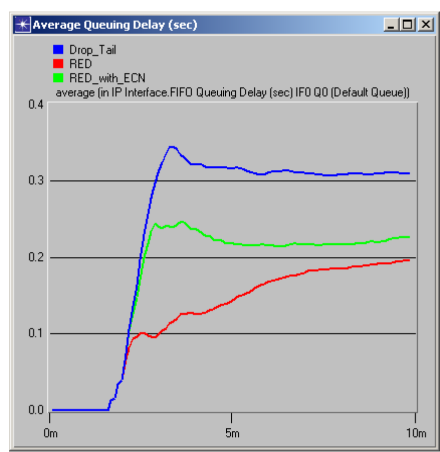

Fig. 4 Average queuing delay.

4.2 Comparing the queuing delay variation

As it is shown in Fig. 3 Drop Tail scheme, because of its global synchronization problem, has more variation in buffer utilization in comparison with RED algorithm. In this algorithm queue reaches its maximum length at 156 sec, and buffer was full until 192 second. Then by finishing the waiting time for receiving the ACK messages, without receiving them (because of packet dropping in congested router), all TCP senders reduced their transmission rates in the same time (global synchronization problem). So we estimate that Drop Tail algorithm has more delay variation (jitter) in compression with RED. Figure 5 indicates the accuracy of our estimate by representing average delay variation.

It is observed from this figure that queuing delay variation of RED-with-ECN, which was more than RED at commence of the simulation, by increasing the time and receiving ACK messages to TCP senders, became almost equal to RED.

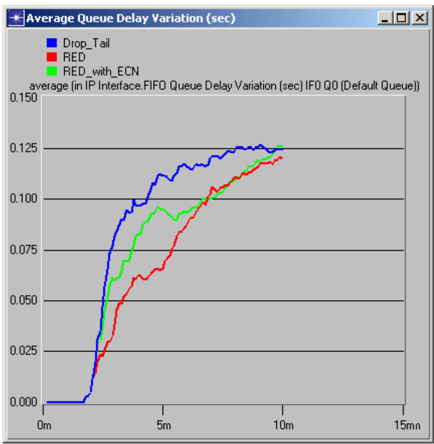

Fig. 5 Average queuing delay variation (jitter).

4.3 Comparing the packet dropping rate

Figure 6 indicates packet drop rate for each queue management algorithm versus simulation time. Comparing Fig. 3 and Fig. 6 indicates that at the same time which the RED's buffer utilization exceeded the minimum threshold (25 packets) in Fig. 3 (around second minute of the simulation time), this algorithm started to drop some arriving packets randomly in Fig. 6. It cause that some TCP senders reduced their transmission rates and therefore RED has less packet dropping during the simulation period related to the Drop Tail algorithm.

Comparing the packet drop rate in RED and RED-with-ECN algorithms in Fig. 6 indicates that RED's packet drop rate reduced due to ECN mechanism. Using ECN mechanism in RED is useful for traffics which are sensitive to packet dropping.

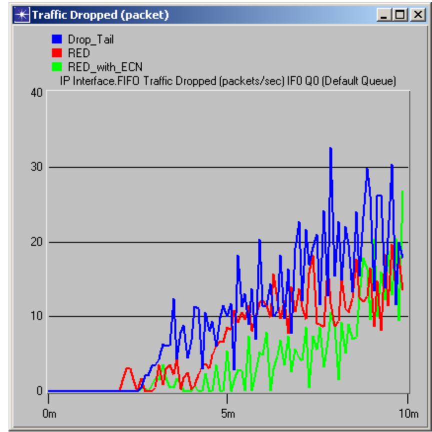

Fig. 6 Packet drop rate.

4.4 Comparing the bandwidth utilization

Figure 7 shows the utilization of outgoing link, by router A versus simulation time. Comparing Fig. 3 and Fig. 7 indicate that when the RED's buffer usage is reduced (due to its random packet dropping mechanism), its bandwidth utilization is reduced too. In addition, using ECN mechanism, which caused the increasing in RED's buffer utilization, also caused the increasing in RED's bandwidth utilization.

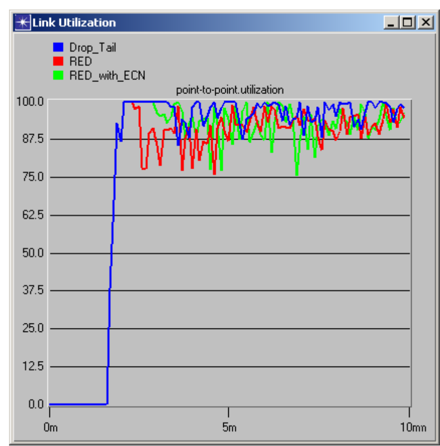

Fig. 7 Link utilization.

4.5 Comparing the router CPU utilization

Figure 8 shows the percentage of CPU utilization for each queue management algorithm versus simulation time in router A. Comparing the Fig. 3, Fig. 6 and Fig. 8 indicate that at the second minute of the simulation time, which the average queue length exceeded the min_{th} (according to Fig. 3) and the RED algorithm started to drop some arriving packets randomly (according to Fig. 6), CPU utilization of this algorithm was increased related to Drop Tail algorithm in Fig. 7. Also at 156 sec. which Drop Tail algorithm reaches to its maximum length and started to packet dropping, its CPU utilization was increased. So we can say queue management algorithms use the processor more at the packet dropping moment. It is observed that after the time that all algorithms started to packet dropping, their CPU utilization were almost as same as each other.

So it can be concluded that the difference between active and passive queue management algorithms, in term of CPU utilization, is mostly depend on their difference moment starting the packet dropping. Packet dropping start moment in different queue management algorithms is depend on buffer size and network traffic load. If the buffer size was small or the network traffic load was heavy, buffer usage graph would have the more gradient. So a few seconds after queue length exceeds the min_{th} (for RED algorithm), it will reach to the maximum length for Drop Tail algorithm. Thus the difference of packet dropping start moment in RED and Drop Tail algorithms and consequently difference between them in CPU utilization, will be decreased.

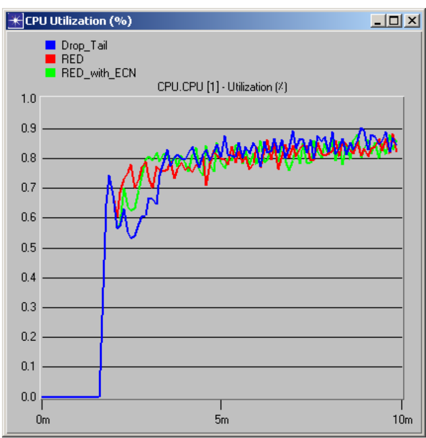

Fig. 8 CPU utilization (%).

5. Conclusion and results

In this paper we presented a simulation-based performance evaluation and compared Drop Tail queue management algorithm, which used in today's IP networks, and RED queue management algorithm, which is recommended as the default active queue management scheme for NGN, in different aspects. The simulation results show RED outperforms Drop Tail in terms of queuing delay, queuing delay variations, and packet drop rate. We also noticed that using ECN mechanism has greatly improved RED's packet drop rate and bandwidth utilization performance measures. Queuing delay and queuing delay variation increasing due to using ECN, will be decreased gradually as the time of simulation is progressed and ACK packet received to the senders. This mechanism turns out to be useful to protocols like TCP that are sensitive to even a single packet drop. Also it was shown that by experimenting buffer's behavior, we can have respective comparisons about delay, delay variation, packet drop rate, and bandwidth utilization QoS measurements. Finally we observed that the difference between router CPU utilization by active and passive queue management algorithms, is mostly depend on their different moment starting the packet dropping and consequently is depend on the buffer size and the network traffic load.

References

[1] T. Reddy, A. Ahammed, and R. banu, "Performance Comparison of Active Queue Management Techniques", IJCSNS International Journal of Computer Science and Network Security, Vol. 9 No. 2, February 2009, pp. 405-408.

[2] S. Floyd, V. Jacobson, "Random early detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking1, 1993, pp. 397-413.

[3] A. Bitorika, M. Robin and M. Huggard, "A comparative study of Active Queue Management schemes", In Proceedings International Conference on Communications (ICC 2004), June 2004.

[4] D. Que, Z. Chen, B. Chen, "An improvement algorithm based on RED and its performance analysis", 9th Int. Conf. on Signal Processing, Oct. 2008, pp. 2005- 2008.

[5] B. Zheng, M. Atiquzzaman, "DSRED: A New Queue Management Scheme for the Next Generation Networks" IEICE Trans. on Communications, Vol. E89-B, No. 3, 2006, pp. 764-774.

[6] K. Zhou, K. Yeung, V. Li, "Nonlinear RED: a simple yet efficient active queue management scheme" Computer Networks: The Int. Journal of Computer and Telecommunications Networking, Vol. 50, No. 18, 2006, pp. 3784-3794.

[7] K. Ramakrishnan and S. Floyd, "A proposal to add Explicit Congestion Notification (ECN) to IP", Available as RFC 2481.

[8] D. Packer, H. Peyravi "Simulation Modeling and Performance Comparison of RED Variant Algorithms," OPNET WORK 2001, Washington DC, Aug 27-31, 2001.

[9] OPNET Technologies, Inc. http://www.opnet.com.

[10] W. Eddy, M. Allman, "A comparison of RED's byte and packet modes," Computer Networks, Vol. 42, No. 2, June 2003, pp. 261-280.