Journal of King Saud University – Computer and Information Sciences (2015) 27, 416–429

King Saud University
Journal of King Saud University – Computer and Information Sciences
www.ksu.edu.sa
www.sciencedirect.com

Performance study of Active Queue Management methods: Adaptive GRED, REDD, and GRED-Linear analytical model

Hussein Abdel-jaber

Faculty of Computer Studies, Department of Information Technology and Computing, Arab Open University, Saudi Arabia

Received 23 July 2014; revised 4 December 2014; accepted 26 January 2015
Available online 11 September 2015

KEYWORDS

Adaptive GRED;
Discrete-time queues;
GRED Linear analytical model;
Performance measures;
REDD

Abstract Congestion control is one of the hot research topics that helps maintain the performance of computer networks. This paper compares three Active Queue Management (AQM) methods, namely, Adaptive Gentle Random Early Detection (Adaptive GRED), Random Early Dynamic Detection (REDD), and GRED Linear analytical model with respect to different performance measures. Adaptive GRED and REDD are implemented based on simulation, whereas GRED Linear is implemented as a discrete-time analytical model. Several performance measures are used to evaluate the effectiveness of the compared methods mainly mean queue length, throughput, average queueing delay, overflow packet loss probability, and packet dropping probability. The ultimate aim is to identify the method that offers the highest satisfactory performance in non-congestion or congestion scenarios. The first comparison results that are based on different packet arrival probability values show that GRED Linear provides better mean queue length; average queueing delay and packet overflow probability than Adaptive GRED and REDD methods in the presence of congestion. Further and using the same evaluation measures, Adaptive GRED offers a more satisfactory performance than REDD when heavy congestion is present. When the finite capacity of queue values varies the GRED Linear model provides the highest satisfactory performance with reference to mean queue length and average queueing delay and all the compared methods provide similar throughput performance. However, when the finite capacity value is large, the compared methods have similar results in regard to probabilities of both packet overflowing and packet dropping.

1. Introduction

With the emergence of modern communications and computer networks, network sources require sufficient resources to deliver their data to their destinations (Tanenbaum, 2002). Insufficient resources lead to congestion, which happens when the amount of incoming packets exceeds the available network

E-mail address: habdeljaber@arabou.edu.sa
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2015.01.003
1319-1578

resources (Welzl, 2005). Congestion causes several forms of degradation in network performance such as: (1) instability in average queue length, so many arriving packets will be dropped because of the congested contents of router buffers, (2) high loss and average queueing delay of packets, (3) low throughput, and (4) unbalanced share of network resources among network sources.

Congestion control is important to improve network performance and can enable efficient use of networks by users (Aweya et al., 2001; Welzl, 2005). Congestion control is a challenge because of the sensitivity of network traffic (Welzl, 2005). Congestion control has become an active research field because of the advancement of real-time or demand traffic. At present, several media traffic applications, such as video conferencing, voice over IP (VoIP), video on demand (VoD), video streaming, and so on, can be found online. The amount of users who use these media traffic applications are increasing rapidly, which can cause congestion. When congestion is present in network applications based on TCP, TCP flows will decrease their transmitting rate to manage congestion. Therefore, a fair share of network flows is generated. By contrast, when congestion occurs within non-TCP network applications, non-TCP flows will keep transmitting in their original rate, which leads to an unfair share of network resources.

Several researchers have proposed congestion control mechanisms such as Drop-tail (Brandauer et al., 2001) and Active Queue Management (AQM) Abdeljaber et al., 2011; Aweya et al., 2001; Feng et al., 1999; Floyd and Jacobson, 1993; Floyd et al., 2001; Floyd, 2000. Drop-tail determines congestion when router buffers are overflowed (Braden et al., 1998; Brandauer et al., 2001). Therefore, congestion is identified after the router buffers become full, which means that congestion in Drop-tail is found at a late stage. AQM mechanisms identify congestion before the router buffers become full. In other words, AQM mechanisms discover congestion at an early stage. Random Early Detection (RED) (Floyd and Jacobson, 1993) and its variants (Floyd et al., 2001; Floyd, 2000) were proposed to control congestion as AQM mechanisms and were built based on simulation. The variants of RED have been proposed to deal with the deficiencies of RED, such as the following: (1) The congestion measure of RED (average queue length (*aql*)) varies with the congestion level such that when light congestion exists, the *aql* value is near the minimum threshold position (min *threshold*) on the RED router buffer. On the contrary, when heavy congestion occurs, the *aql* value is near the maximum threshold position (max *threshold*) on the RED router buffer. (2) RED is sensitive to its parameters (min *threshold*, max *threshold*, maximum value of packet dropping probability (*pd* max), and queue weight (*qw*)). (3) The *aql* value relies on the number of TCP flows. Thus, the *aql* value may exceed the max *threshold* position (heavy congestion) when the number of TCP flows is high, which means that every arriving packet will be dropped. (4) RED cannot maintain the *aql* value between min *threshold* and max *threshold* positions when busy traffic is present. Therefore, the *aql* value may exceed the max *threshold* position, which leads to heavy congestion.

The current paper aims to evaluate the performance of three AQM methods under a single queue node. Two of the AQM methods, which are Adaptive GRED and REDD, are implemented based on simulation, whereas the last AQM method was built as a discrete-time queue analytical model, and is called the GRED Linear analytical model.

This paper critically compares congestion control methods primarily Adaptive GRED, REDD, and GRED Linear using different evaluation measures (mql, T, D, P_L, and D_P). This paper aims to identify the method that gives the best performance after deeply analyzing the packet dropping probability which affects mean queue length; average queueing delay and packet overflow. The results of performance measure results can help in choosing the method that can be applied as a congestion control method in computer networks such as the Internet.

The paper is structured as follows: Related works are reviewed in Section 2. Sections 3–5 present the Adaptive GRED method, GRED Linear analytical model, and REDD method, respectively. A previous comparison between the GRED and Adaptive GRED methods is presented in Section 6. The simulation details of Adaptive GRED and REDD are demonstrated in Section 7. Section 8 presents different analytical modeling approaches besides their verification and validation using Simulation. Finally, Sections 9 and 10 show the results of the performance evaluation of Adaptive GRED, REDD, and GRED Linear based on varying values of packet arrival probability and varying values of finite capacity of queues. The conclusions and suggestions for future work are presented in Section 11.

2. Related work

Other examples of AQM mechanisms are analytical models, which were proposed based on discrete-time queue approach (Abdeljaber et al., 2008, 2008; Al-Diabat et al., 2012) and different AQM mechanisms based on simulation. Woodward wrote a book about building discrete-time queue analytical models, in which these analytical models were built by modeling and analyzing the performance of queuing systems, such as computer communications and networks (Woodward, 1993). Abdeljaber et al. (2008, 2008) and Al-Diabat et al. (2012) were built as discrete-time queue analytical models based on DRED, GRED, and BLUE mechanisms. Another analytical model based on RED was presented in Bonald et al. (2000). These analytical models controlled congestion by decreasing packet arrival probability either from fixed value to another or linearly. (Lim et al., 2011) was proposed as an analytical model and uses aggregated internet traffic as the sources of different classes of traffic. This analytical model works as a queue management mechanism for sustaining queueing delay at a specific level on the router buffer (Lim et al., 2011). This developed model utilizes the control method of closed loop feedback (Lim et al., 2011) to limit average queuing delay by implicitly informing the arrival rate and by moving the queuing threshold (Lim et al., 2011). The proposed model derived the relationship between the queueing thresholds and average queueing delay based on the traffic model, which represents aggregated internet traffic using the superposition of the arrival processes of N Markov Modulated Bernoulli Process-2 (MMBP-2). The value of the queueing threshold is revised based on the derived relationship between the queueing thresholds and average queueing delay, as well as the evaluation of average queueing delay feedback. Packets can be dropped