

Simulation Study of RED/WRED Mechanism Based on OPNET

Rui-juan Jiang, Yun Pan, Yi-Lan Liu, Xiao-Jin Xue

Department of Computer, Communication University of China
Beijing, China

jiangruijuan@cuc.edu.cn, pany@cuc.edu.cn, 1060869565@qq.com, 986221779@qq.com

Abstract—Nowadays with the addition of application demands, the quality of service (QoS) of network becomes more and more important. Network QoS has a variety of support mechanism, this paper mainly studies two congestion avoidance mechanisms: Random Early Detection (RED) and weighted RED (WRED). Using OPNET software to create network simulation models with three different scenarios. Each scenario support different QoS guarantee mechanism, here, the Scenario 1 has no QoS guarantee, the Scenario 2 supports RED mechanism and the Scenario 3 supports WRED mechanism. Do simulation experiments ten minutes, and verify the validity and importance of RED and WRED these two kinds of mechanisms in terms of providing QoS guarantee by analyzing the simulation results of three scenarios. And through comparing the statistical results between Scenario 2 and Scenario 3, we realized that RED provides the same level QoS for different types of traffic, however WRED provides different levels QoS for them. This paper concludes that RED and WRED two congestion avoidance mechanisms can provide certain QoS guarantee for network applications.

Keywords-QoS; Congestion Avoidance; RED; WRED; OPNET

I. INTRODUCTION

With the development of computer network, people increase the demands of Internet applications on the number and type. Different types of applications are different from the requirements of QoS, for example, audio application and video application require higher QoS to ensure the real-time capability and reliability of applications. The "Best-effort" service model, which is achieved based on FIFO (First In/First Out) queue scheduling [1], makes network try its best to send messages, but does not guarantee delay and reliability, so it can not provide desired QoS for multimedia real-time applications. How to provide QoS guarantee for different applications is a popular research topic.

The queue scheduling mechanisms, such as Priority queuing (PQ) and Weighted Fair Queuing (WFQ) [2], determine how to schedule data packets in the logic queue to leave queue, thus achieve the support for QoS guarantee.

The basic idea of congestion avoidance mechanism is that, when a router's buffer utilization reaches a certain threshold, it will drop packets, according to certain probability, to notify TCP sources to reduce the congestion window, so as to achieve the aim of congestion avoidance [3]. RED and WRED are the frequently-used congestion

avoidance mechanism, these two mechanisms have been analyzed and summarized in [4]. This paper is based on OPNET [5] simulation platform to create different experiment scenarios, and do performance simulations for RED and WRED these two kinds of congestion avoidance mechanisms. By comparing the experimental results with the traditional FIFO queuing mechanism which has no QoS support, we verify the validity of RED and WRED in providing QoS guarantee.

II. QUALITY OF SERVICE (QoS)

Quality of Service (QoS) [6] can be understood as the ability that network guarantees service level to various types of applications. QoS is the quality agreement about information transmission and information sharing between network and users, users and users that can communicate with each other. QoS is one kind of network security mechanism, which can be used to solve the problems such as network latency and congestion [4]. Under normal circumstances, the network only serves the applications which have no time limit, so do not need QoS to do special business security. If there are a lot of multimedia applications, which need higher real-time requirements in network, the QoS can guarantee the important applications' transmission, at the same time ensure the efficient use of network resources.

Different types of applications have different demands for network conditions, such as network reliability, end-to-end delay, delay jitter and network bandwidth, so the service levels they needed is also different. The Internet needs to judge the application types by various means, and generates a suitable service strategy according to application types [7], thus provides a higher level service for them. The QoS requirements of several typical applications as shown in TABLE I.

TABLE I. THE QoS REQUIREMENTS OF SEVERAL TYPICAL APPLICATIONS

	<i>Reliability</i>	<i>Delay</i>	<i>Jitter</i>	<i>Bandwidth</i>
FTP	High	High	Low	Middle
VoIP	Low	High	High	Low
Video	Low	High	High	High

III. RED/WRED

A. Random Early Detection (RED)

RED algorithm is a representative algorithm of Active Queue Management (AQM) [8] algorithm, it was proposed by Sally Floyd and Van Jacobson [9], and it is used for congestion avoidance based on FIFO queue scheduling policy.

FIFO mechanism will not send information to TCP sources before appearing congestion conditions on the network link. Tail drop caused by queue overflow will make many TCP connections into slow start status at the same time, the reduction of traffic relieves the congestion, but it will cause a new round of congestion again after network returning to normal status, so leading to the oscillating phenomenon of global synchronization [10]. The global synchronization reduces the utilization and throughput of network link, therefore the traditional FIFO mechanism cannot provide high quality service for multimedia applications.

Compared to traditional tail drop mechanism, RED avoids congestion by monitoring the average queue length on the router output port. When the average queue length reaches a certain level, RED can notify TCP sources about impending congestion through the way of probabilistic dropping packets, and TCP sources will reduce the congestion window size and slow down the sending rate. Because RED uses average queue length rather than instant queue length to calculate the drop probability of packets, so this mechanism can absorb the temporary burst traffic as much as possible.

RED algorithm includes two parts: calculating average queue length and calculating drop probability [11, 12].

When packets arrive at router, RED will calculate average queue length, the formula as follows:

$$\text{avg} = (1 - W_q) \times \text{avg} + W_q \times q \quad (1)$$

Here, W_q is the weight whose value range is from 0 to 1, q is the actual queue length of sampling measurement.

According to average queue length to calculate the drop probability, the formula as follows:

$$p_b = p_{\max} \times (\text{avg} - \text{MIN}_{\text{th}}) / (\text{MAX}_{\text{th}} - \text{MIN}_{\text{th}}) \quad (2)$$

$$P = p_b / (1 - \text{count} \times p_b) \quad (3)$$

Here, MIN_{th} and MAX_{th} respectively are the minimum threshold and the maximum threshold of average queue length. p_{\max} is the biggest drop probability when the average queue length reaches MAX_{th} . count is the number of queued packets when avg stays between MIN_{th} and MAX_{th} .

When $\text{avg} \leq \text{MIN}_{\text{th}}$, RED does not drop the arriving packets; When $\text{MIN}_{\text{th}} \leq \text{avg} < \text{MAX}_{\text{th}}$, RED drops the arriving packets with the probability p ; When $\text{avg} \geq \text{MAX}_{\text{th}}$, RED drops every arriving packet. The drop probability of RED algorithm is shown in Fig. 1.

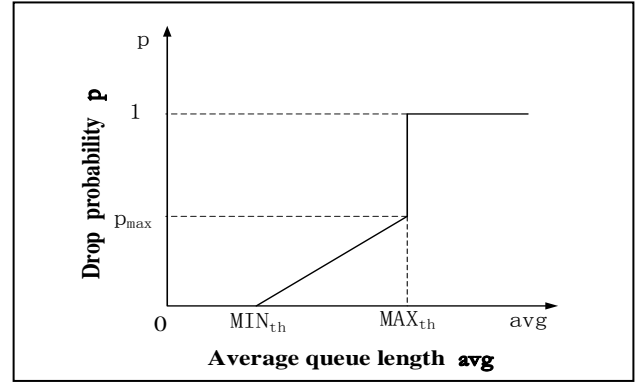


Figure 1. The drop probability of RED algorithm

B. Weighted Random Early Detection (WRED)

WRED is a variant of RED, it can control the congestion effectively according to different service priority at the beginning of the congestion [5], and provide preferential communication processing for high-priority packets.

In fact, WRED is the combination of RED and Priority Queuing, it sets different minimum threshold MIN_{th} , maximum threshold MAX_{th} and the biggest drop probability p_{\max} for packets with different types of services (ToS).

WRED can parse priority tags of packets, according to the priority is high or low, this mechanism will distinguish handle the different levels of packets, thus achieves the goal of providing different services for different types of traffic. When the congestion is impending, WRED will drop packets selectively according to the priority. The drop probability of low priority packets is greater than high priority packets, so the high priority packets are not easy to be dropped and are more likely to be sent to destination. Before congestion happening, WRED can reduce the chance of tail drop by selectively dropping packets. It avoids a large loss of packets as a result of buffer overflow, so that the risk of global synchronization is reduced. The drop probability of WRED algorithm is shown in Fig. 2.

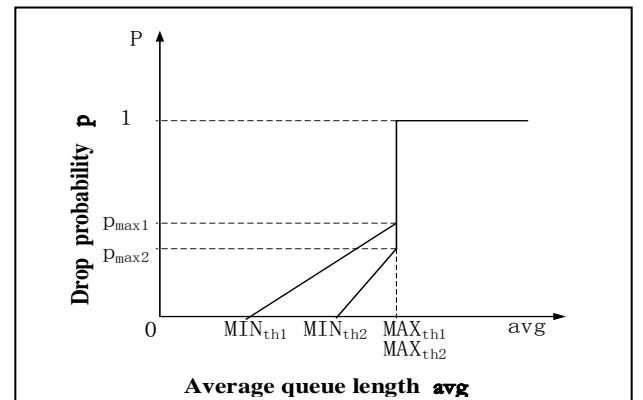


Figure 2. The drop probability of WRED algorithm

IV. SIMULATION EXPERIMENTS

A. Simulation Models

This paper uses OPNET as the simulation platform to study the validity of RED/WRED providing support for QoS. We create a network topology is shown in Fig. 3.

Here, the Client 1, Client 2, Client 3 and Client 4 are source nodes; Router 1 and Router 2 are routers; Server 1 and Server 2 are destination nodes. Demand is defined between two nodes, it mainly relies on the modeled traffic types. There are four demand objects in this topology, respectively modeling for different types of traffic between source nodes and destination nodes [5].

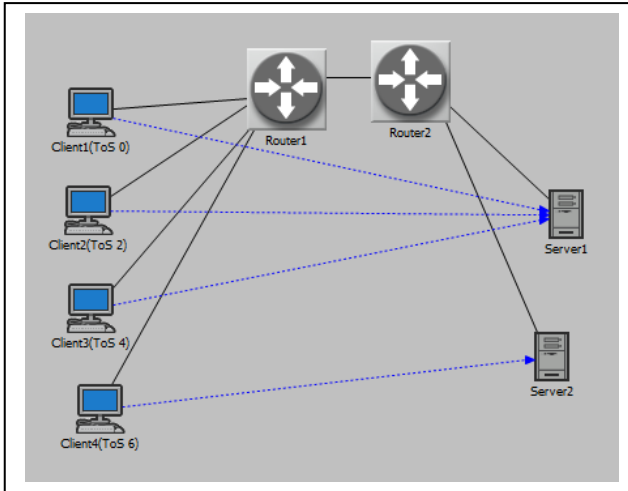


Figure 3. The network topology

In experiments, three scenarios are designed, and each scenario supports different QoS guarantee mechanism. The QoS support of each scenario is shown in TABLE II.

TABLE II. THE QoS SUPPORT OF EACH SCENARIO

	Queuing profile	QoS mechanism
Scenario 1	FIFO	—
Scenario 2	FIFO	RED
Scenario 3	FIFO	WRED

B. Parameter Configuration

Configuring all demand objects' data transmission rate is 1.53 (Mbit/s), the data rate on the link between Router 1 and Router 2 is 5 (Mbit/s). Each demand object requires to carry different types of traffic, as shown in TABLE III.

TABLE III. THE TYPES OF TRAFFIC DEMAND OBJECTS CARRIED

Demand objects	The types of traffic
Client1 (ToS 0) —>Server1	Best Effort(0)
Client2 (ToS 2) —>Server1	Standard(2)
Client3 (ToS 4) —>Server1	Streaming Multimedia(4)
Client4 (ToS 6) —>Server2	Interactive Voice(6)

RED parameters are configured as $MIN_{th} = 100$, $MAX_{th} = 200$, $p_{max} = 0.1$.

In OPNET, for WRED parameters configuration, only allowing to change p_{max} value, while MIN_{th} and MAX_{th} for all types of traffic are identical, so the configurations of WRED's MIN_{th} and MAX_{th} are as same as RED. The

corresponding p_{max} values of four types of traffic respectively are 1, 0.2, 0.1 and 0.05.

C. Results Analysis

(1) Collecting statistics: Buffer Usage (bytes), Queuing Delay (sec) in Node Statistics and Packet End - to - End Delay (sec) in Demand Statistics.

The comparison results of Buffer Usage (bytes) and Queuing Delay (sec) on Router1' outgoing port of three scenarios are shown in Fig .4 and Fig .5, respectively. In Fig .4, the horizontal axis is simulation time, the vertical axis is buffer usage. And in Fig .5, the horizontal axis is simulation time, the vertical axis is queuing delay.

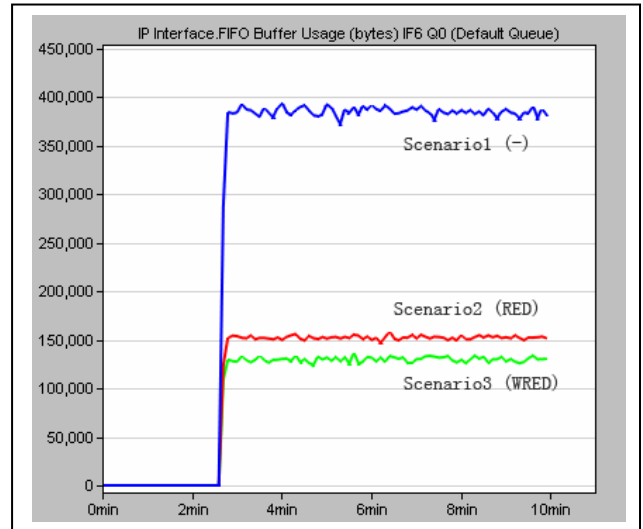


Figure 4. Buffer Usage (bytes)

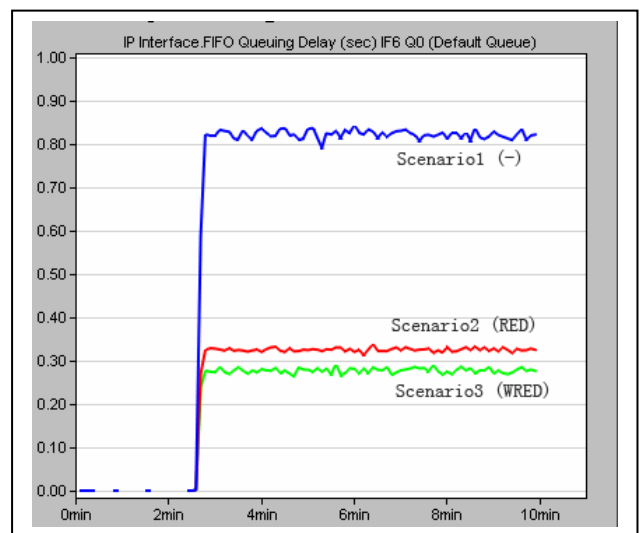


Figure 5. Queuing Delay (sec)

The traditional FIFO queuing mechanism drops the arriving packets until the buffer queue is full, so the buffer queue usage rate of router will be very big, in other words the average queue length will be longer. However the RED and WRED drop packets don't have to wait the buffer queue going be full. When packets arrives at buffer queue, these two mechanisms will calculate the average queue length and compare it with two thresholds, therefore determining the processing actions of packets: dropping,

no dropping or dropping at a certain probability. This kind of strategy reduces the router buffer queue occupancy, and effectively controlling the average queue length. As can be seen from Fig .4, the buffer utilization in Scenario 2 (RED mechanism) and Scenario 3 (WRED mechanism) are significantly lower than the buffer utilization in Scenario 1 (no QoS support).

We can see from Fig .5, the packets queuing delay on Router1' outgoing port in Scenario 2 (RED mechanism) and Scenario 3 (WRED mechanism) are less than that in Scenario 1 (no QoS support). Because RED and WRED maintain smaller average queue length, so leading to the reduction of queuing delay.

The statistical result of packets average end-to-end delay is shown in Fig .6.

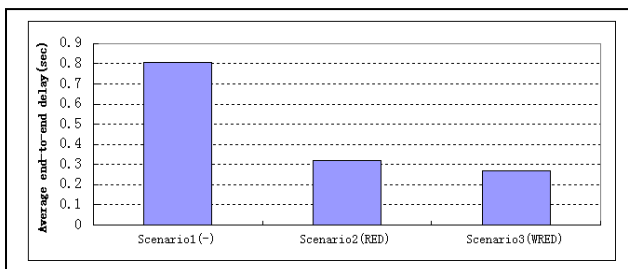


Figure 6. Packets average end-to-end delay

Because Scenario 2 and Scenario 3 respectively using RED and WRED mechanism, the average queue length can be effectively controlled, so packets needn' t spend much more time on queuing, and it saves the forwarding time of packets on the Router1' outgoing port, thus reducing the end-to-end delay of the whole transmitting procedure.

By observing and analyzing the above experimental results, we verified the validity of RED and WRED mechanism providing support for QoS.

(2) For Scenario 2 and Scenario 3, choosing Traffic Received (packets/sec) in Demand Statistics as a statistic.

The statistical result of delivery traffic is shown in Fig .7.

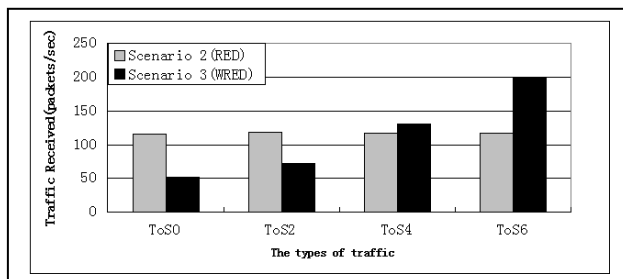


Figure 7. The delivery traffic

As can be seen from the Fig .7, under RED mechanism, the delivery of different priority traffic is mostly same. However under WRED mechanism, the delivery of traffic will be growing with the higher and higher ToS value. As the same as RED, WRED also randomly drops packets at a certain probability according to the average queue length, so that avoiding congestion. While WRED can

discriminate different traffic types, when congestion will occur, WRED chooses the lower priority traffic to drop according to the traffic's priority is high or low, rather than dropping packets simply. The drop probability of the higher priority traffic is smaller, so the traffic delivered to destination node will be larger. This feature makes WRED can provide different types of traffic with different levels of QoS guarantee.

By comparing the delivery of different priority traffic, we realized that WRED mechanism can provide different levels of service for different types of applications.

V. SUMMARY

This paper implemented the simulation study of RED and WRED two kinds of congestion avoidance mechanisms based on the OPNET simulation platform. Through analyzing the experimental results of three scenarios, we concluded that RED and WRED can provide certain QoS guarantee for applications comparing to traditional FIFO mechanism. And by analyzing the statistical results of delivery traffic in Scenario 2 and Scenario 3, we realized that WRED utilizes the combination of RED and Priority Queuing to provide different levels QoS guarantee for different types of applications.

ACKNOWLEDGMENTS

This work is supported by Communicate university of China Engineering Planning Project (3132014XNG1445).

REFERENCES

- [1] ZHU Xi-jun. Dynamic Priority Queue Research and Implementation of DiffServ[D]. Kunming University of Science and Technology, 2012.
- [2] HUANG He. Simulation of Queue Strategy Mechanism in QoS Based on OPNET[D]. Jiangnan University, 2008.
- [3] Dr Helonde J B,Dr. Wadhai V,Vivek Deshpande,Shiv Sutar. EDCAM: Early Detection Congestion Avoidance Mechanism[J]. International Journal of Computer Applications, 2010, 72:.
- [4] ZHU Yi, WU Xin,The Study of WRED Algorithm Based on QoS Queue Management[J]. Journal of Huainan Normal University, 2010, 03: 31-32.
- [5] Adarshpal S.Sethi, Vasil Y.Hnatyshin. The Practical OPNET User Guide for Computer Network Simulation. CHINA MACHINE PRESS.
- [6] Hrvoje Kozačinski,Petar Knežević. Configuration of Quality of Service Parameters in Communication Networks[J]. Procedia Engineering,2014,69:.
- [7] CUI Heng. The Theory Research of Computer Network Quality of Service (QoS)[D]. Wuhan University of Technology, 2008.
- [8] ZHANG Jie. The Design and Implementation of RED Congestion Control Improved Algorithm Based on the Parameter Study[D]. Northwest University, 2012.
- [9] CAO Zhi-bo. Simulation and Optimization of RED Algorithm[J]. Computer Technology and Development, 2010, 08: 188-191.
- [10] JIANG Jian. Optimization Research of RED Algorithms in Router Congestion Control[D]. Jilin University, 2014.
- [11] ZHU Yi, WU Xin. Analysis and Improved Research of Random Early Detection Algorithm[J]. Computer Knowledge and Technology, 2011, 01: 94-96.
- [12] Jianyong Chen,Cunying Hu,Zhen Ji,Ming Li. Self-Tuning Random Early Detection Algorithm to Improve Performance of Network Transmission[J]. Mathematical Problems in Engineering,2010,2011:.