Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 15 (2011) 2369 – 2373

Procedia Engineering

www.elsevier.com/locate/procedia

An Improved Adaptive Active Queue Management Algorithm Based on Nonlinear Smoothing

Jingjun Zhang[a], Wenlong Xu[b*], Liguo Wang[b]

[a] *Department of Science Research, Hebei University of Engineering, Handan, Hebei, 056038, China*
[b] *Hebei University of Engineering, Handan, Hebei, 056038, China*

Abstract

According to analyzing the principle and limitations of RED and its improved algorithms, an improved adaptive RED congestion control mechanism is proposed in this paper, which imposes nonlinear smooth for packet loss rate function of RED algorithm by using the membership function of the ascend demi-cauchy of fuzzy distribution. The speed of growth of packet loss rate is relatively slow near the minimum threshold, while near the maximum threshold the speed of growth of packet loss rate is relatively faster. The $P\max$ will be dynamically adjusted by the length of average queue and the target for adapting the changing of network environment. NS simulation shows that this algorithm has been significantly improved for packet loss rate, throughput and other performance.
Keywords: RED algorithm; nonlinear; adaptive; network simulation

1. Introduction

RED algorithm[1] is the most typical active queue management (AQM) algorithm. Compared with the DropTail algorithm, RED is a more effective congestion control mechanisms and provides a better network performance. The basic idea of RED is that determining whether the network is congestion through testing the average router queue length. And when the average length of the queue reaches a certain threshold, the random selection of some of the new arrival packets are discarded or marked, and notify the source to reduce the congestion windows. The source will reduce the sending rate of packet and relieve congestion. RED algorithm can effectively improve the link bandwidth utilization and reduce the average queue length. With the increasing application of RED, researchers have proposed many improved strategies of RED algorithm, such as ARED[2], SRED[3], BLUE[4], etc. Combining with the main idea of RED algorithm, this paper proposes an improved algorithm—NARED, which replaces the original function of a linear increase in packet loss rate by the membership function of the ascend demi-cauchy of fuzzy distribution, and dynamically adjust the value of $P\max$ to improve the link utilization and throughput, reduces packet loss rate, and further improves the network transmission performance.

2. Random early detection algorithm

* *E-mail address*:jsjxuwenlong@163.com

doi:10.1016/j.proeng.2011.08.444

2.1 The principle of RED

RED algorithm randomly discards the packets which are entering the router by the average queue length based on time, and RED algorithm is mainly composed of two independent algorithms. The first part is the calculation of the average queue length; another part is the calculation of packet dropping/marking probability which determine the router in the current congestion level to drop/mark packets by how much the probability is. The principle of RED was showed by the figure 1. RED has several important parameters which are $Q\min$, $Q\max$, $P\max$ and Wq . $Q\max$ and $Q\min$ are the values of thresholds of RED. $P\max$ is the maximum packet loss rate. And the Wq is a weight.

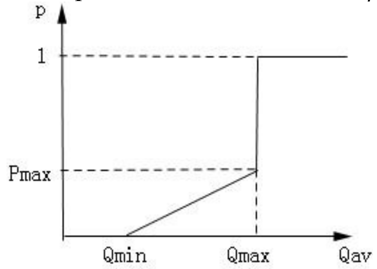


Fig.1 The packet discarding strategy of RED

2.2 The analysis of RED and improved algorithm

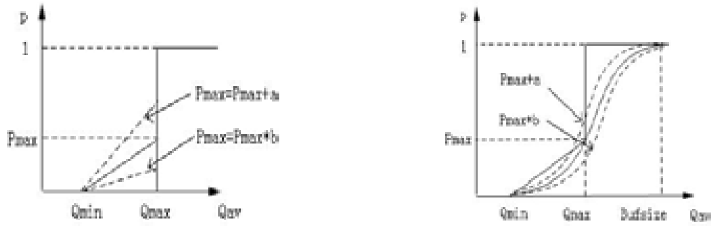
RED is recommended as an effective router-based congestion avoidance mechanism by the IETF. Although this algorithm is an efficient congestion control mechanism, it still has some problems, such as problems of parameter setting, can not effectively estimate the severity of congestion, fairness, etc.

According to the defects of RED, researchers have proposed many improved strategies of RED algorithm by using of the changes of network traffic, adaptively adjusting parameters and controlling the packet marking/discarding probability, such as ARED, SRED, Gentle—RED, BLUE. The SRED obtains the number of connections by probability and statistics, and then adjust the probability of packet dropping based on the number of connections. The ARED is an adaptive RED algorithm whose packet dropping strategy is showed by the figure 2(a). The ARED adaptively adjust $P\max$ according to the situation of network congestion. When the Qav is near to $Q\min$, the value of $P\max$ will be reduced, if Qav is close to $Q\max$, the value of $P\max$ will be increased.

3. An improved RED algorithm—NARED

3.1 The basic principle of NARED

This paper proposes an improved RED algorithm which is NARED, the principle of which is showed by figure 2(b). NARED introduces a new parameter $Bufsize$, because when Qav exceeds $Q\max$, a part of queue is idle, so the probability should not be changed to 1 immediately. When the average queue length is near to $Q\min$, the growth of packet loss rate is not fast. But when the average queue length exceeds $Q\max$, the growth of packet loss rate is faster. When the average queue length reaches the vicinity of $Bufsize$, the packet loss rate will reach to 1. The value of $P\max$ will be dynamically adjusted by using of the average queue length and the target of queue length to adapt to the different situation of network.



(a) (b)

Fig.2 The packet discarding strategy of ARED and NARED

3.2 The specific strategy of NARED

According to the basic idea of NARED, NARED replaces the original function of a linear increase in packet loss rate by the membership function of the ascend demi-cauchy of fuzzy distribution by its basic idea. Packet loss rate is calculated by the function of the ascend demi-cauchy of fuzzy distribution and average queue length, which has implemented the changes of packet loss rate smoothing. The following formula is the function of the ascend demi-cauchy of fuzzy distribution.

$$\mu_A(u)=\begin{cases}0 & u\le\alpha\\ a(u-\alpha)^{\beta}/(1+(u-\alpha)^{\beta}) & u\ge\alpha,\ a>0,\ \beta>0\end{cases}\tag{1}$$

After simulation, the best value of β is 3. u is taken as Qav . α is $Q\min$ .The following formula will be obtained by putting these three values into the formula 1 and the function of the packet dropping probability of RED.

$$p_b=\begin{cases}0 & Qav\le Q\min\\ a(Qav-Q\min)^3/(1+a(Qav-Q\min)^3) & Q\min<Qav<Bufsize\\ 1 & Qav\ge Bufsize\end{cases}\tag{2}$$

The value of a in the formula 2 will be calculated, when Qav is $Q\max$ and the value of packet loss rate is $P\max$.

$$a=P\max/((1-P\max)(Q\max-Q\min)^3)\tag{3}$$

The distribution function of probability of packet drop which bases on the function the ascend demi-cauchy of fuzzy distribution will be obtained by putting the value of the parameter a into the formula 4.

$$p_b=\begin{cases}0 & Qav\le Q\min\\ \dfrac{P\max(Qav-Q\min)^3}{(1-P\max)(Bufsize-Q\min)^3+P\max(Qav-Q\max)^3} & Q\min<Qav<Bufsize\\ 1 & Qav\ge Bufsize\end{cases}\tag{4}$$

The realization of packet dropping strategy of NARED algorithm is as follows:

(1) The group arrival

/*Calculating the average queue length.*/

(2) if the queue not free then

$Qav=(1-Wq)\times Qav+q\times Wq$

/* Calculating the probability of the packet marking/discarding */

(3) if $Qav<Q\min$ then

$p = 0$

(4) if $Q\min \le Qav < Bufsize$ then

/*Adjusting the value of $P\max$ according to the relation of the length of average queue and the target of queue*/

if ($Qav < k1$) $P\max = P\max \times b$

if ($Qav > k2$) $P\max = P\max + a$

$$Pb = \frac{P\max(Qav - Q\min)^3}{(1 - P\max)(Bufsize - Q\min)^3 + P\max(Qav - Q\max)^3}$$

$p = p_b / (1 - count \times p_b)$

(5) if $Qav > Bufsize$ then

$p = 1$

(6) End

$(k1, k2)$ is the range of the value of the target of the queue， $k1 = Q\min + 0.4(Bufsize - Q\min)$； $k2 = Q\min + 0.6(Bufsize - Q\min)$。 a and b are the parameters for adjusting the value of $P\max$. The value of a is the smaller between 0.01 and $P\max/4$. The value of b is 0.9.

4. NS simulation and performance analysis

In this paper, the simulations were implemented by NS2 in the Linux operating system. The network topology of this experiment is shown in Figure 3. The size of queue buffer is set to 100 packets and simulation time is 100s. $Q\min$ =10, $Q\max$ =40, $P\max$ =0.1, Wq =0.003, $Bufsize = 2Q\max$.

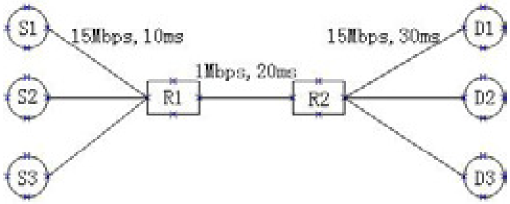


Fig.3 Topology model of network simulation

Table 1 shows the comparison of the changes of packet loss rate under RED, ARED and NARED. Compared with the RED, the packet loss rate of NARED algorithm reduces about 34.9 percent. Compared with ARED, it reduces about 7.8 percent. Figure 4 shows the comparison of throughput under the two algorithms. Compared with RED and ARED, the throughput under NARED has a obvious improvement. But the length of average queue showed in figure 5, the length under NARED is larger than anther two algorithms.

Table 1 The comparison of packet loss rate

Algorithm	The number of send	The number of dropped	The packet loss rate
RED	11805	172	1.46%
ARED	11982	124	1.03%
NARED	11987	114	0.95%

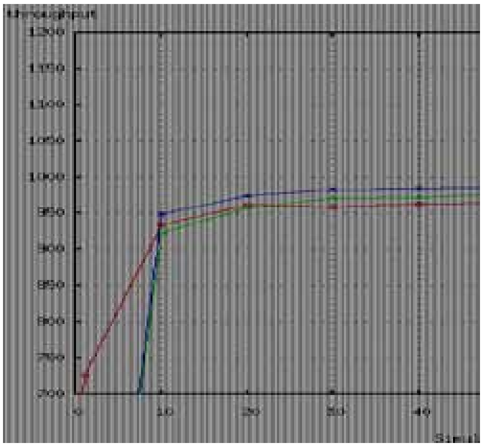


Fig.4 The comparison of throughput

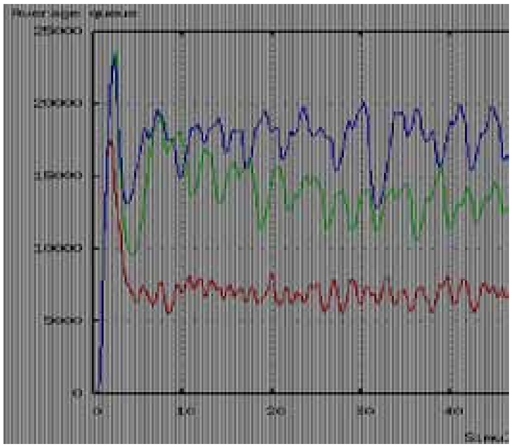


Fig.5 The comparison of average queue

5. Conclusion

This paper makes a brief introduction about RED algorithm and the improved algorithm base on RED, and pointed out their advantages and shortcomings. An improved RED algorithm proposed in this paper which is NARED that has improved the performance of packet loss rate and throughput by simulation, and especially there is a great advantage on the control of packet loss. But there are some deficiencies in NARED, which is the control of the length of the average queue. It is the next focus of the study how to minimize packet loss rate and control the delay in a certain range.

Acknowledgements

This work was financially supported by Natural Science Foundation of Hebei Province (F2010001042).

References

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

[2] FLOYD S, GUMMADI R, SHENKER S. Adaptive RED: all algorithm for increasing the robustness of RED’s active queue management [EB/OL], http://www. Icir.org/floyd/papers/adaptiveRed.pdf, Aug. 2001

[3] Ott T J, Lakshman T V, Wong L H. SRED: Stabilized RED. In: Doshi, B. ed. Proceedings of IEEE INFOCOM. New York, USA: IEEE Communications Society, 1999: 1346-1355

[4] Feng W, Kandlur D, Saha D, Shin K. Blue: A New Class of Active Queue Management Algorithms. Technical Report, U.Michigan CSE-TR-387-99, 1999

[5] NS Project [EB OL].http://www.isi.edu /ns nam /.

[6] Zhi Wang, Weidong Wu. Improved RED algorithm using the Logistic model [J]. Journal of Computer Applications, 2010,**30(6)**: 1472-1474

[7] Jindong Li, Dongtang Ma, Wei Li, Shan Wang. Non-linear Congestion Control Based on RED Arithmetic [J]. Computer Engineering, 2008,**34(20)**:91-92