International Journal of u- and e- Service, Science and Technology
Vol.9, No. 2 (2016), pp.375-384
http://dx.doi.org/10.14257/ijunesst.2016.9.2.36

Research on the Improved Way of RED Algorithm S-RED

ZHAO Yu-hong[1,2], ZHENG Xue-feng[1] and TU Xu-yan[1]

(1. School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China, 100083;
2. School of Information Engineering, Inner Mongolia University of Science and Technology,Baotou,China,014010
E-mail: zhaoyuhong35@163.com

Abstract

In order to improve sensitivity of parameter and network load, an improved random early detect ion (RED) mechanism orienting to stability, which is named extended stability of random early detection (S-RED) , was proposed. This algorithm sectionally calculates the probability of packet loss by using the quadratic function to make its variability smoother, meanwhil ,and simplify the calculation formula ofprobability by decreasing the number of parameters in the calculation to properly weaken the performance influence due to parameter settings.

We tested our proposed algorithm using the Network Simulator version 2 (ns-2) and found that it shows better stable performance than other typical RED variants and close throughput.

Keywords*: RED, congestion control, stability, sensibility,nonlinearity*

1. Introduction

Nowadays, along with the rapid growing of network technology and application, network congestion being increasingly serious, the congestion control is becoming more and more urgent. The congestion control mechanism of TCP/IP[1,2] protocol considered packet loss instruction as an important indicator of whether the network congestion or not.

While the buffer of routing node is filled, it will discard the next arriving packet. The sender will judge whether the network congestion occurs according to specific feedback and take corresponding measures. Although, at present the congestion control mechanism has made some achievement, then some problems are still existed, such as unfairness, poor adaptivityand heavier burden for end node, and so on[3,4]. There are many problems still can't be well resolved simply using TCP end-to-end congestion control mechanism, so many scholars started to research congestion control about intermediate router, expecting to achieve better congestion control .As routing scheduling and practitioner, the router can get the status of the network congestion in time. If it can adjust the link resources according to the congestion indication in advance, the purpose of the method which can realize avoiding congestion or rapid recovery.

Routing Queue Management mechanism is important measures of the network node congestion control mechanism, which can be roughly divided into passive queue manager (PQM) and active queue management (AQM) . Typical passive queue management such as droptail algorithm, it don’t forecast the network load and take measures, packet will be discarded or labeled only when the buffer overflow. However, buffer overflow can cause the global synchronization, dead lock and large oscillation of queue length[5,6]. In 1993, Floyd proposed a kind of active queue management algorithm, random early detection (RED) [7]. It forecasts buffer queue length first and to take measures according to predicted values. In 1999, Gentle RED[8] was proposed based on RED by Floyd, to solve the problem that packet loss probability is increasing linearly to 1, rather than mutations

ISSN: 2005-4246 IJUNESST

to 1 as the average queue length is larger than the maximum set in the RED. Later, many variants of RED were released, such as adaptive RED[9].

However, these algorithms have a common problem, which they need to set many parameters and the changing of parameter will largely influence the algorithm performance, at the same time algorithm performance is different under different network load. In order to overcome parameter configuration and sensitive for network load in RED algorithm, we proposed an improved algorithm of RED, named S-RED.

2. Overview of RED

As a kind of active queue management algorithm, RED, executed in router, the router detects incipient congestion by computing the average queue size and could notify connections of congestion either by dropping packets arriving at the router or by setting a bit in packet headers. In order to avoid the happening of congestion, the key is how to detect congestion and how to calculate packet loss rate. In the RED algorithm, introduces two important parameters of min_{th} and max_{th} respectively queue threshold, the maximum and minimum values is used to detect congestion.

Definition 1- Assuming min_{th} and max_{th} for the minimum and maximum threshold for queue, the q for current instantaneous queue size, avg as average queue size, p_a for the final data packets probability, pb as a intermediate variable, max_p for a maximum value of p_b.

The RED algorithm can be described by the followings in figure1.

```
Begin
   Initialization:
     avg =0
     count= −1
     for each packet arrival to calculate avg
     if the queue is nonempty
         avg= (1−wq)avg+ wqq
         else
         m= f (time−q-time)
         avg= (1− wq)^m avg
         ifminth≤ avg<maxth
         count=count+1
         pb = maxp (avg− minth)/( maxth − minth)
         pa = pb/(1−count ×pb)
         with pa marking or dropping the arriving packet
         count= 0
       else if maxth ≥avg
       marking or dropping the arriving packet
       count= 0
           else count= −1
           when queue becomes empty
           q- time= time
Variablesinstructions : time fo current time; q time: start of the queue empty time
                        F(t): a linear function of the time t; wq: queue weight;
                        count: packets since last marked packet
```

Figure 1. Principle of RED Algorithm

The value of parameters in algorithm and the concrete meaning can see [7]. To obtain relatively evenly distribution of packet loss and avoid bias of bursty, RED algorithm respectively introducing variables p_a and count. Such can realize packet loss evenly spaced and avoid global synchronization caused by continuous packet loss. in order to obtain the stable performance, Gentle RED changed the calculation of p_b. The result is that p_b is not mutation to 1 from max_p, but linear growth When avg is larger than

max_{th}.

RED, Gentle RED algorithm's a common defect is that packet loss probability changes too fast with different network loads. Namely, if the congestion is not too serious or the value of $max_p$is too large, the average queue near to min_{th} and If the congestion is serious or $max_p$is too small , the average queue near to max_{th} . Especially, when avg is larger than max_{th} , the probability of packet loss is mutation to 1. Another problem is parameter sensitivity. Different parameter Settings in the algorithm, performance present great changes. Although a lot of research have done in many variants of RED algorithm, we try to do some work from the aspects of simplified configuration of parameter and the stability of the performance and gain a certain researching achievement.

3. ProposedS-RED Algorithm

3.1 Problematic Presentation

AQM mainly solve the following problems: detecting incipient congestion by computing the average queue size and discarding randomly or labeling packet to notify the source side take measures to prevent the possibility of congestion ;Fair treatment including burst, persistent and episodic TCP traffic flow; avoiding global synchronization due to queue overflow for multiple TCP connections; keeping short queue length and making reasonable balance between high throughput and low delay. RED algorithm will initially be as the only candidate AQM strategies algorithm by IETF and get the major manufacturers supporting. But there are two main problems in RED according to the previous description, swings of packet loss probability and parameter sensitivity. Stability problems exist in RED due to its first-order filtering and nonlinear proportional in reference [10]. Self- oscillation is main reason of queue's periodic motion, which is evoked by nonlinear structure of packet loss probability curve in [11]. To improve problem of parameter sensitivity, many algorithms adopt dynamic configuration. Although some achievement gained by many scholars, the algorithm efficiency and performance is still flawed. We think it is a crucial problem in research,that includes stability, robustness and simplicity in the algorithm. Our improved algorithm –S-RED focus on improving linear structure of packet loss probability and reducing quantity of setting parameters based on simulation experiment and mathematical analysis by many scholars.

3.2 Introduction of S-RED

3.2.1The Problem and Solutions

Firstly, setting of parameters is considered. Max_p and max_{th} in RED algorithm is an important parameter, because it determines the intensity of the congestion control and it is also an unintelligible parameter at the same time, because so far its value has no appropriate standard solution. It is shown through the analysis and simulation of the RED algorithm that max_p is a variables, changing in interval[0,1], which have the effect of a moderating for the calculation of pb. IRED achieve the purpose of the same moderating probability by increasing denominator in calculation formula of Pb, changing from maxth-minth to Qmax-minth. Qmax is the router's cache size. Here we haven't introducedmax_p, but to adjust the calculation of Pb by extending calculation interval to achieve the same purpose. At the same time we replace max_{th} with cache capacity, on the one hand to increase the link utilization also to solve problem of set of max_{th}, and we forecast congestion in advance and take measures to avoid t buffer overflow.We can not only improve the sensitivity of the parameter in RED algorithm and also prevent the problem of the packet loss probability mutation to1.

Secondly, the issue is that performance of RED algorithm is unstable.Discussed in previous reference, instability mainly caused by linear structure of calculation of p_b in RED.P_b in RED algorithm is a linear function of avg, so the change ratio ofp_b directly mapping to avg variability, namely jitter. This is the main cause triggering instability of algorithm performance.We use the quadratic function to calculate the packet loss probabilityquadratic function has a nonlinear characteristic than a linear function.In the condition of key parameters known, variability of quadratic function is always larger than one of linear function to make the result calculated by the quadratic function smaller, smoother for probability of packet loss and smaller jitter of the queue.

So quadratic function can meet the calculation demand of the packet loss probability, and at the same time can slow changing speed of the packet loss probability and decrease jitter of the average queue lengthand delay of end-to-end, then eventuallythe stability of the algorithm is improved. These advantages can be verified from following simulation experiment.We found that the probability of packet loss by nonlinear calculation inexperiments is relatively smooth, but the jitter rate hasn't get good improvement.On the condition of low load in network, because packet in cache base is small, even if the setting of packet loss probability is larger, there have not severe impact on throughput. When network load is high, if packets haven't discarded timely, it will cause buffer overflow, global synchronization and declining of throughput.So we should set up the packet loss rate according to the different load condition. Loads can be reflected from the queue length, namely, we can change the set of packet loss rateaccording to the queue length.

3.2.2The Details of S-RED

As mentioned above S-REDalgorithm implement calculation of packet loss probability by expandingdenominator range, thus the amount of parameter settings are reduced and parameter sensitivity is weakened in the algorithm. To get smooth of variability of packet loss and reduce jitter rate by using the quadratic function replacing linearized equations.In order to guarantee the performance of the algorithm under different load levels, packet loss probability is set in terms of segmented queue length to meet the demand. The selection of segmentationpointis very important for algorithm performance. Toimprove the throughput and increase the link utilization, this point is set 0.6 times of the maximum cachethrough a large amount of simulation test, where named avg_{seg}. Thetheoretical basis of improved algorithmhas been discussed above, the effectiveness will be verified in the following experiment. Other parameters in the algorithm is set by reference to RED.The detailed algorithm for S-RED is described in figure 2.

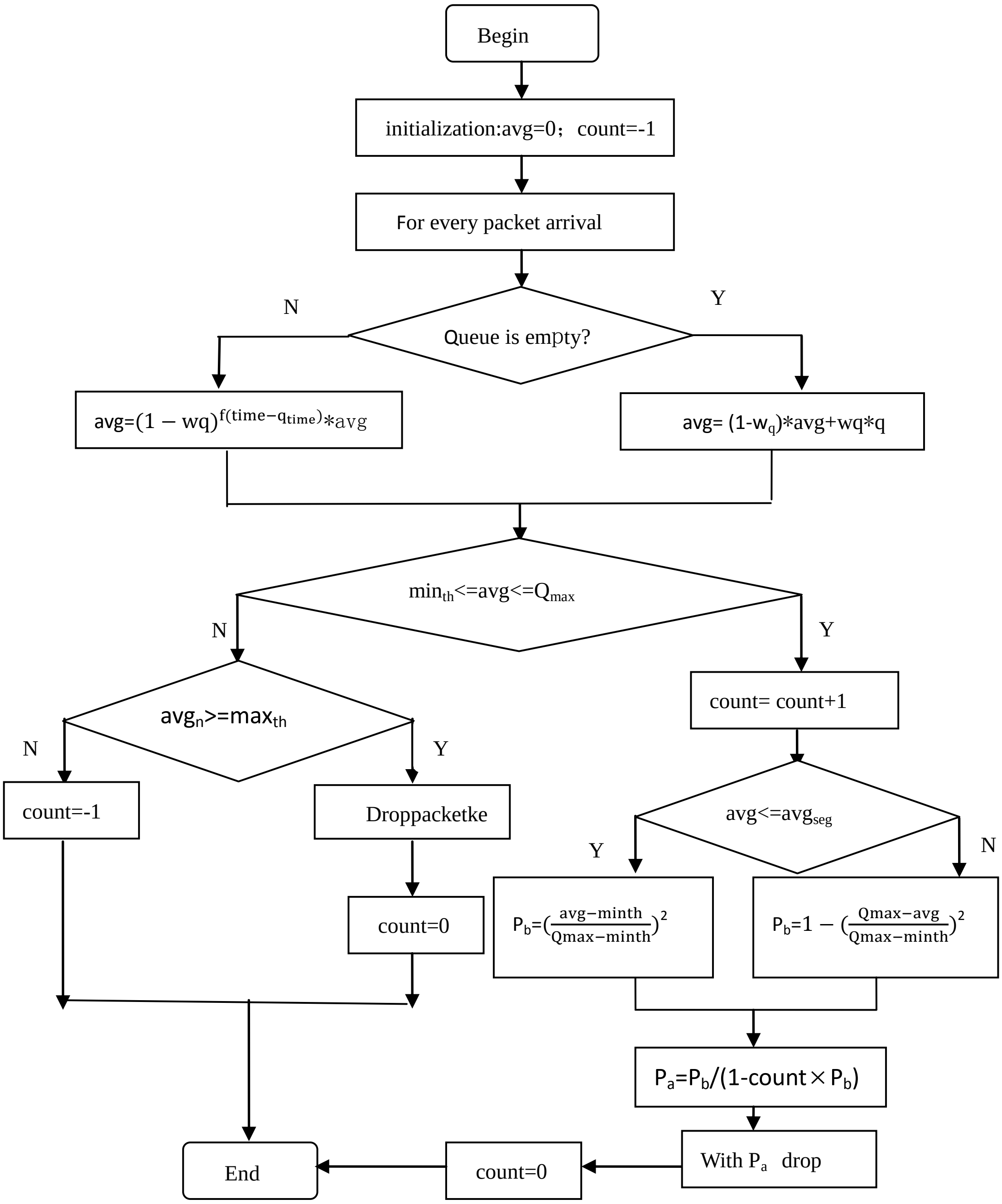

Figure 2. The Flowchart of S-RED

4. Simulation Results and Analysis

4.1 The Simulation Environment

We tested our proposed algorithm using the Network Simulator version 2 (ns-2) and compared the same with other major TCP variants. We have used a mixed network in order to evaluate the performance of our proposed algorithm. Figure 3 shows the network topology used in the simulations.

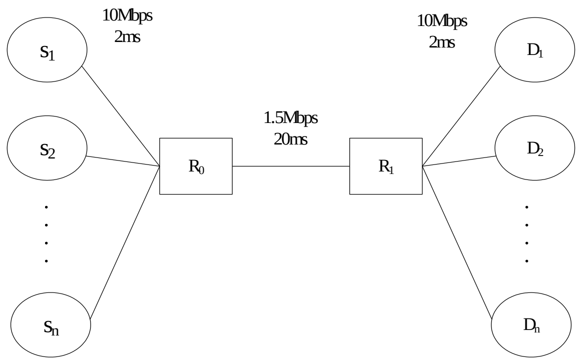

Figure 3. Network Topology of Simulation Experiment

The simulationuse a simple dumbbell topology with a congested link of 1.5Mbps. The source from S1 to Sn are TCP sender, which respectively send data correspondence the terminal from D1 to Dn, namely, Si send data to the Di . Link rate is 10Mbps and delay of 0.2 seconds. Ro and R1 are routers. The buffer accommodates 35 packets, which, for 1000-byte packets, corresponds to a queuing delay of 0.2 seconds. In all of the simulations, $w_q$is set to 0.0027, min_{th} is set to five packets, and max_{th} is set to 20 packets. Reception window set is enough large to make transferring controlled only by the congestion window .S1 to Sn are persistent source FTP, R0 to link rate of 10 megabits per second, delay 2 ms. We have explored these simulations for a range of scenarios, including a range of link bandwidths and mixes of web traffic.

4.2 The Analysis of Experiment Results

Set simulation environment according to the layout of the figure 2. Timing is initiated, with 30 long-lived TCP flows, each with a roundtriparound 2ms, competing over a 15 Mbps link .After 30 seconds , starting the 90 TCP sender, while 60 seconds ,90 TCP sender are stop. All sender stop transferring data in 100 seconds and simulation ends. Simulation scenario is set , which start low congestion from 0 to 30 seconds and reach peak of congestion from 30 to 60 seconds with the maximum of queue length. While 60 second, congestion isgradually eliminated and the simulation to end in 100 seconds.Some parameters is setas described above and the rest is used the default values in NS2 [12].Simulation experiments were done using respectively RED, gentle-RED A-red and IRED algorithm aiming at performance indexof end-to-end delay, the average queue length and throughput. The simulation results are shown in figure from 4 to6, respectively.

Delay1, delay2, delay3 and delay4 respectively indicates the change of roundtrip delay for RED, Gentle-RED, Adaptive-RED and S-RED four algorithms in figure 4. Reflected from the figure 4, the delay in the S-RED algorithm is smallest. Due to the lag of networkreaction,the peak of congestion is emerged near 30 seconds, resulting in the largest delay. Network congestion is lowest close to 60 seconds, with minimum delay.

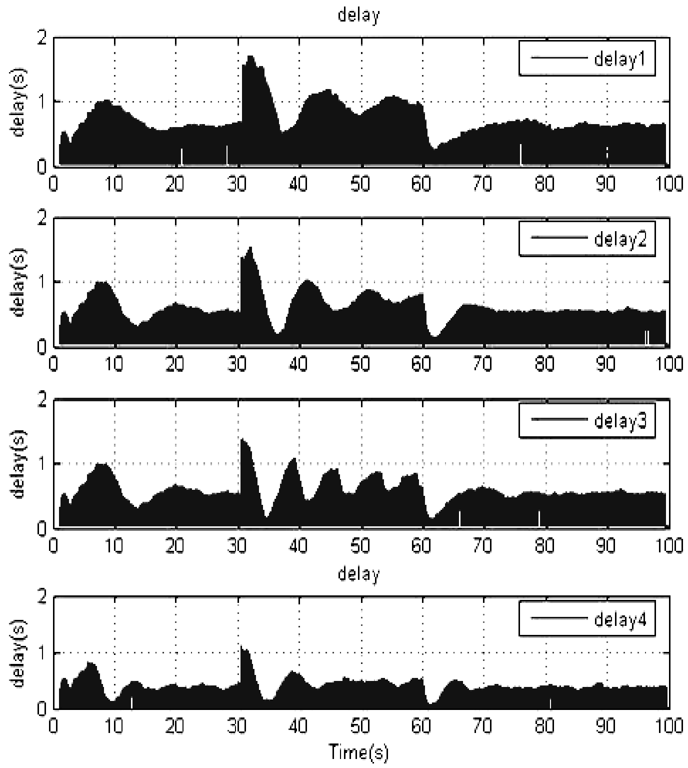

Figure 4. Comparison of Delay for Different Algorithm

AVG1, AVG2, AVG3 and AVG4 respectively indicates the change of average length of queue for RED, Gental-RED, Adapative-RED and IRED four algorithms in figure 5. It is shown that amplitude of fluctation of AVG curves obtained by RED algorithm greatly exceed the Gentle-RED ,A-RED and IRED and the fluctuation of average queue get by IRED is smallest and jitter rate is lowest.

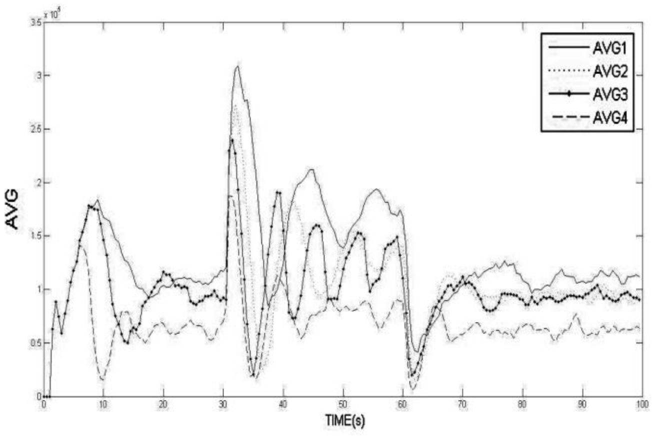

Figure 5. Comparison of avg for Different Algorithm

To better simulate the performance of algorithm under different load conditions, we respectively set scenario with low and high load and collect data. The core of congestion control algorithm is to control the length of cache, so we choose the average queue length as a reference value. The experimental data including time and length of average is displayed in table 1and table 2 using different algorithm under various load status.

Table 1. The Comparison of avg under Low Network Load

RED		Gentle-RED		Adaptive-RED		S-RED	
时间（s）	avg	时间（s）	avg	时间（s）	avg	时间（s）	avg
1	0	1	0	1	0	1	0
2	88552.18	2	88552.18	2	88552.18	2	88552.18
3	58801.63	3	58801.63	3	58801.63	3	58801.63
4	89797.66	4	89797.66	4	89797.66	4	89797.66
5	117665.1	5	117665.1	5	117665.1	5	117665.1
6	145049.2	6	145385.5	6	145385.5	6	140979.4
7	164205.6	7	164920.5	7	164920.5	7	133402.2
8	176593.8	8	177255.2	8	177255.2	8	79678.4
9	184250.8	9	174237.2	9	174237.2	9	26700.48
10	167275.2	10	146571.1	10	146571.1	10	15271.23

Table 2. The Comparison of avg under High Network Load

RED		Gentle-RED		Adaptive-RED		S-RED	
时间（s）	avg	时间（s）	avg	时间（s）	avg	时间（s）	avg
30	116996.1	30	92290.35	30	92290.35	30	70109.75
31	217033.4	31	229863.1	31	229863.1	31	187320
32	305973.2	32	273284.2	32	227136.1	32	174056.6
33	290183.4	33	233213.2	33	151880.5	33	101926
34	277547.7	34	167589	34	67171.1	34	39462.23
35	229412.7	35	88031.56	35	20312.07	35	20262.97
36	172971.5	36	37986.34	36	58387.55	36	22968.41
37	100692.4	37	32338.76	37	103119.8	37	60946.11
38	92756.94	38	58038.78	38	158804.8	38	93173.96
39	99309.5	39	119359.2	39	190695.5	39	113044.5
40	125131.2	40	154479.2	40	154636	40	104851.9

A good congestion control algorithm is to achieve the balance between high throughput and low delay, so we can not sacrifice throughput in pursuit of low latency. Th1, Th2, Th3 and Th4 respectively indicate the change of throughput for RED, Gentle-RED, Adaptive-RED and S-RED four algorithms in figure 6. It is shown that the throughput of the four algorithms is very close.It is proved S-RED is effective which obtain low latency and realize high link utilization at the same time.

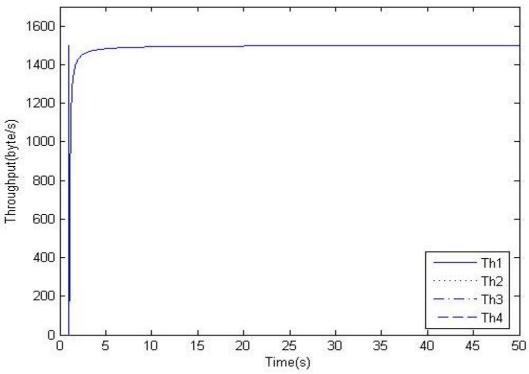

Figure 6. Comparison of throughput for Different Algorithm

Simulation results shows that the S-RED algorithm proposed can effectively improve problems ofparameters and load sensitivity in RED and simplify the parameter setting. Algorithm can realize smaller delay in section of bottleneck link, at the same time reduce variation range of average queue and jitter amplitudeand maintain close throughput to other classic algorithm.

5. Conclusion

On the basis of the study and analysis of RED algorithms, we found that algorithm is sensitive to key parameters and network load. It is main reason leading to unstable performance of the RED algorithm. In order to solve these problems,S-RED is proposed, which effectively improve the performance of the algorithm by incorporating quadratic function to calculation of the packet loss probability and reducing appropriately the number of parameters,

Finally, a positive result is get by NS2 simulation experiments.S-RED effectively controls the variety range of the average queue length and reduces the roundtrip delay of bottleneck link. It is proved thatS-RED can get stability of algorithm performance, besides, keep close throughput to other classic algorithms and greatly improve quality of service.

References

[1] W. Luo, M. Lin, C. Yan and B. Ping, "A Survey of Congestion Control in the Internet", Chinese Journal of Computers, vol. 24, no. 1, **(2001)**, pp. 1-17.
[2] J. Liu and X. H. Tong, "The TCP congestion control algorithm", Computer engineering and design, vol. 32, no. 7, **(2011)**, pp. 2309-2313.
[3] T. Fukatus, T. Kiura and M. Hirafuji, "A web-based sensor system with distributed data processing approach via web application", Computer Standards &Interface, vol. 33, no. 6, **(2011)**, pp. 565-573.
[4] B. Jian-min, "Introduction to Internet of things technology and application", Beijing: people's posts and telecommunications publishingClub, **(2011)**.
[5] J. X. Wu, X. Zhao and Y. X. Zhang, "Active queue management algorithm research", Control engineering, vol. 15, no. 2, **(2008)**, pp. 178-181.
[6] M. Sabato, B. Mario and G. Franco, "Design validation and experimental testing of a robust AQM control', Control Engineering Practice, vol. 19, no. 3, **(2009)**, pp. 1-6.
[7] S. Floyd and V. Jacobson, "Random early detectiongateway s for congestionavoidance", IEEE Transactionson Networking, vol. 1, no. 4, **(1993)**, pp. 397- 413.
[8] S. Floyd, "Recommendation on using the "Gentle_variant of RED algorithm", [EB/OL].http: / /www.icir.org /floyd /red /gentle.html., **(2008)**.
[9] S. Floyd, R. Gummadi and S. Shenker, "Adaptive RED: An Algorithm for Increasing the Robustness of RED'sActive Queue Management Technical Report", **(2001)**.
[10] V. Misra, W. B. Gong and D. Towsley, "Fluid-based Analysis of a Network of AQM RoutersSupporting TCP Flows with an Application to RED", ACM/SIGCOMM, **(2000)**, pp. 151-160.
[11] F. Y. Ren, C. Lin and F. B. Wang, "Stability of RED Algorithm: Analysis Based on Nonlinear Control Theory", Chinese Journal of Computers, vol. 25, no. 12, **(2002)**, pp. 1302-1307.
[12] USC/ISI, "The NS simulator and the documentation [EB/OL]", http://www.isi.edu/nsnam/ns/, **(2013)**.

Authors

Zhao Yu-hong, She received B.E.degree from School of Information Engineering, Inner Mongolia University of Science and Technology in 1998.She received M.S. degree from Inner Mongolia University of Science and Technology in 2003. Now she is studying in University of Science and Technology Beijing for Ph.Ddegree. Her main research areas include computer network and complex system control.(Email:zhaoyuhong35@163.com)

Zheng Xue-feng, He received M.S degree from Beijing Iron and Steel academy in 1982. He is a senior professor and Ph.D Supervisor of the School of Computer and Communication Engineering, University of Science and Technology Beijing. His main research interests include computer network and information security.(Email: zxfustb@126.com)

Tu Xu-yan, He received M.S degree from Institute of Automation ,Chinese Academy of Sciences in 1962. He is currently a professor and Ph.D Supervisor of the School of Computer and Communication Engineering, University of Science and Technology Beijing. His main research interests artificial intelligence and artificial life.(Email:tuxuyan@126.com)