Low pass filter/over drop avoidance (LPF/ODA): an algorithm to improve the response time of RED gateways

Bing Zheng[1,†] and Mohammed Atiquzzaman[2,*,‡]

[1] *New Focus Inc., 2584 Junction Ave., San Jose, CA 95136, U.S.A.*
[2] *School of Computer Science, University of Oklahoma, Norman, OK 73019, U.S.A.*

SUMMARY

The low pass filter (LPF) algorithm, which is used to calculate the average queue length in the random early detection (RED) algorithm, results in a poor response time when RED recovers from congestion. In this paper, we propose a new algorithm, called low pass filter/over drop avoidance (LPF/ODA), to improve the response time of RED. LPF/ODA is based on reducing the time required by RED to recover from long-term congestion. Simulation results have been used to show that the proposed LPF/ODA algorithm significantly improves the response time, throughput and delay of RED gateways. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: active queue management; next generation routers; random early detection; performance enhancement

1. INTRODUCTION

Random early detection (RED) [1], an active queue management (AQM) scheme, has been recommended by IETF for next generation Internet gateways. RED was originally designed to solve the lock-out and full queue problem associated with tail drop queues, with less attention to the response time. However, today's Internet is characterized by short-term traffic (such as http) which requires short response times from AQM schemes.

RED uses four parameters and one state variable to regulate its performance. The state variable is the *average queue length* which is defined as $\mathrm{avg} = (1 - w)\mathrm{avg} + wq$, where w is a weight parameter with a value much less than 1, and q is the instantaneous queue size of the gateway buffer. It works as a low pass filter (LPF) [1] by filtering out transients in the instantaneous queue length. *However, the LPF characteristic of calculating avg results in poor response to changes in long-term congestion*, and limits the benefits of RED for interactive applications (such as http) which require short response times [2, 3].

*Correspondence to: M. Atiquzzaman, School of Computer Science, University of Oklahoma, Norman, OK 73019, U.S.A.
[†] E-mail: zhengbin@ieee.org
[‡] E-mail: atiq@ou.edu

The slow response time of RED is due to the fact that after long-term congestion at an RED gateway, a considerable amount of time is required for avg to drop to a level where it does not unnecessarily drop packets when the instantaneous queue size is low [2]. Although a large value of w can improve the response time, it will adversely affect the performance of RED by tracking short-term congestion (which also violates the AQM requirements [4]). The *objective of this paper is to redefine the LPF algorithm of RED with a view to increasing the response time of RED, but without violating the rules of AQM.*

In this paper, we propose the *low pass filter/over drop avoidance* (*LPF/ODA*) algorithm to calculate the average queue size in RED. We have shown that LPF/ODA maintains the advantages of RED in tracking long-term congestion, while improving the response time by reducing the number of packets dropped at the end of long-term congestion. The main *contribution of this paper is a new algorithm to reduce the response time of RED gateways.*

The rest of the paper is organized as follows. The proposed algorithm is described in Section 2, followed by simulation results in Section 3 which compare the performance of LPF/ODA and LPF. Concluding remarks are presented in Section 4.

2. LOW PASS FILTER/OVER DROP AVOIDANCE (LPF/ODA) ALGORITHM

It has been observed that RED gateways alternate between long-term congestion and congestion-free periods [5]. Consider an RED gateway whose instantaneous queue length is shown in Figure 1, where we can see a congestion-free period between two long-term congestion periods. Although the instantaneous queue length falls rapidly at the end of the first long-term congestion period, *the average queue length using LPF* [1] *falls very slowly. This will cause packets to be unnecessarily dropped with a high probability even during the congestion free period*, resulting in slow recovery of the throughput and poor response time. We state below the requirements of the original RED algorithm, along with *how well the LPF algorithm satisfies the requirements*:

- *Trace long-term congestion* while accommodating short-term congestion: The LPF algorithm satisfies this criterion.
- *Quickly respond* to the end of long-term congestion in order to avoid dropping of excessive packets: **The LPF algorithm of original RED does not satisfy this requirement very well**.
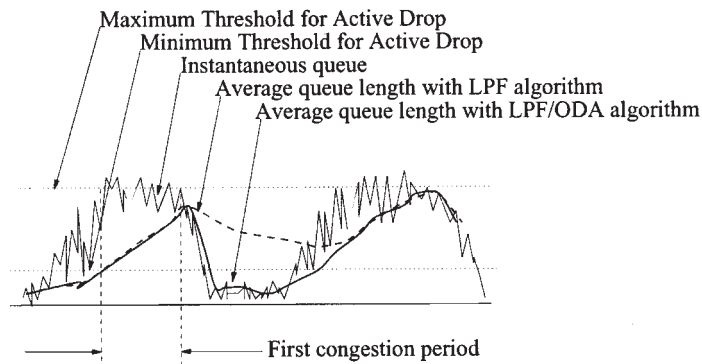

Figure 1. Comparison of average queue length with LPF and LPF/ODA algorithms.

Our proposed *LPF/ODA* algorithm solves the above limitation of the LPF algorithm in calculating the average queue length. LPF/ODA alternates between two phases: *active drop* and *over drop avoidance.* The LPF/ODA algorithm to calculate avg is given below.

Step 1: During long-term congestion, calculate the average queue length using the LPF algorithm. During this period, the RED queue is in the *active drop* phase because it randomly drops incoming packets.

Step 2: At the end of long-term congestion (indicated by the instantaneous queue being at a 'low level' for a considerable period of time and the average queue length is higher than RED's minimum threshold min_{th}) [1], halve the value of the average queue length to avoid over dropping incoming packets. During this period, the RED queue with LPF/ODA is at ODA phase, resulting in an *exponential reduction of the average queue size.* (Note that the LPF algorithm maintains a high value of the average queue length and unnecessarily drops incoming packets.)

Step 3: When the average queue length drops below min_{th} (because of ODA phase in step 2) after a long-term congestion, calculate the average queue length using the LPF algorithm.

From above description, during long-term congestion, the average queue length is calculated by $avg = (1 - w)avg + wq$ in LPF/ODA; if long-term congestion is gone and the avg is greater than min_{th}, then the avg will be halved; if long-term congestion is gone and the avg is smaller than min_{th}, then the avg is still calculated by $avg = (1 - w)avg + wq$. The pseudocode of the LPF/ODA is given in Figure 2.

As shown in Figure 2, the congestion detection in LPF/ODA is similar to the regular LPF in the presence of long-term congestion. Therefore, it will not introduce any difficulty when TCP connections are slow starting with very high congestion windows. From the pseudocode, it is seen that if the queue is in long-term congestion during the slow-start period, it will still be traced as in regular LPF. If the queue is not in long-term congestion, it is not necessary to trace the burst level congestion.

The LPF/ODA algorithm results in the RED queue alternating between the active drop and over drop avoidance phases. During the *active drop* phase (i.e. when the RED queue is experiencing long-term congestion), the LPF model tracks long-term congestion while accommodating short-term congestion; active packet drop is performed depending on the value of the average queue length. During the *ODA* phase (i.e. at the end of long-term

```
FOR each packet arrival
    IF long term congestion THEN
        avg<=(1-w)avg+wq
    ELSE
        IF avg >= minth THEN
            avg<=0.5 avg
        ELSE
            avg<=(1-w)avg+wq
        ENDIF
    ENDIF
```

Figure 2. Pseudocode for LPF/ODA algorithms.

congestion), halving the average queue size to quickly catch up with the end of long-term congestion avoids unnecessary dropping of packets. *The performance of the LPF/ODA algorithm is therefore, similar to that of the LPF algorithm during 'congestion periods' (as required by IETF), while having faster response time than LPF during 'congestion-free periods'*. The LPF/ODA algorithm retains the advantages of RED by using the LPF algorithm, while at the same time overcomes the limitation (poor response time) of the LPF algorithm. Because of fewer packet drops, the LPF/ODA algorithm also improves the throughput and delay of RED.

3. SIMULATION RESULTS

To compare the performance of LPF and LPF/ODA, simulations were carried out with OPNET 5.1D. OPNET is a widely used industry-standard simulation platform. The simulation topology (as in References [1, 2]) is shown in Figure 3, and the RED configuration parameters were [6]: $min_{th} = 5$ packets, $max_{th} = 20$ packets, $p_{max} = 0.1$ and $w = 0.07$. w was set to 0.07 to ensure that the router queue is under active queue management. Since w will decide how the average queue length follows the traffic congestion, if w is too small, the average queue length will be far behind the congestion, resulting in active queue management being ineffective. Poisson traffic was used in simulation. The links connecting the three servers to the gateway were 100 Mbps each, with propagation delays of 1, 5 and 3 ms, respectively. To ensure a fair comparison, the values of propagation delays were set to the same value as in Reference [1]. The bottleneck link (from gateway to client) was 10 Mbps with a propagation delay of 5 ms.

In our simulation, the instantaneous queue length of less than five packets was considered as *low level*. Detection of low level for three consecutive packet arrivals was taken as *the end of long-term congestion. Response time* is defined as the time required by the gateway to react to the end of long-term congestion. It is measured by the time from the end of long-term congestion to the start of the gateway recovering its throughput.

We compare the performance of LPF and LPF/ODA in Figures 4–7 using *normalized throughput, instantaneous queue size, queuing delay and packet drops as the performance measure*. We focus on the simulation time in the time interval 133–166 s which represents a congestion-free period between two periods (0–133 and 166–300 s) of long-term congestion.
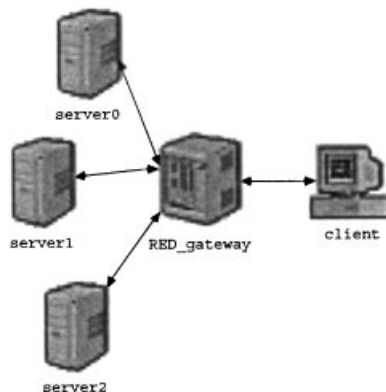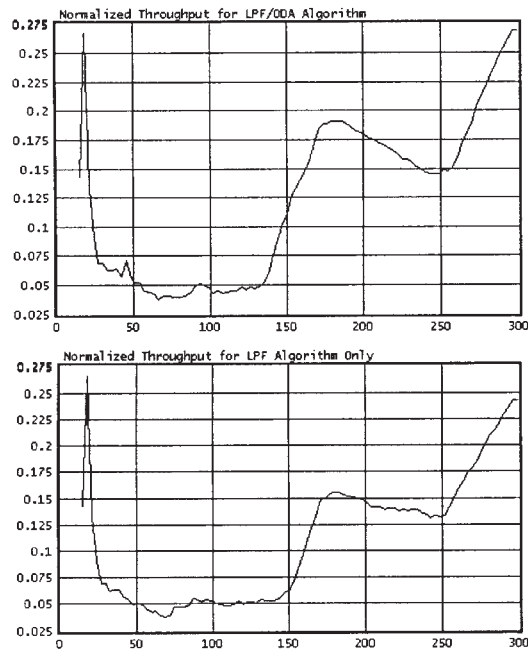

Figure 3. Simulation configuration.

Figure 4. Comparison of *normalized throughput* of the LPF/ODA and LPF algorithms.
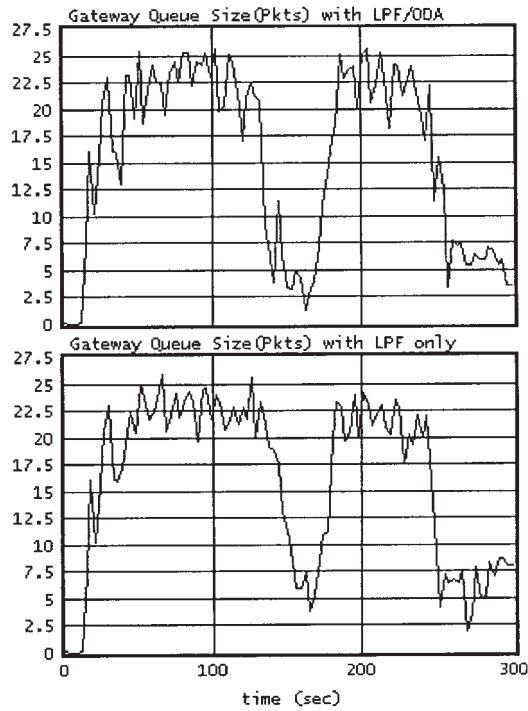
Figure 5. Comparison of *instantaneous queue size* of LPF/ODA and LPF algorithms.
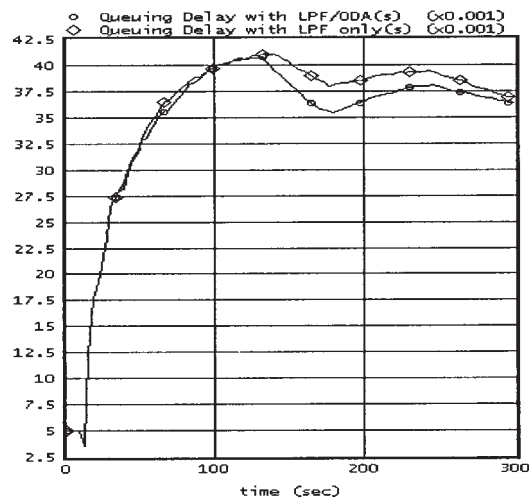
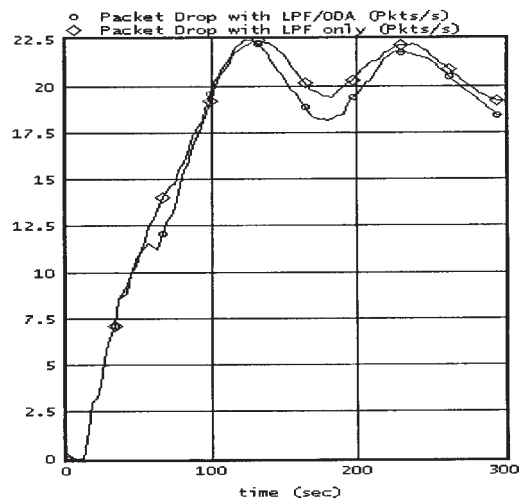Figure 6. Comparison of *queuing delay* of LPF/ODA and LPF algorithms.

Figure 7. Comparison of *packet drop* for LPF/ODA and LPF algorithms. Note that LPF requires 13 s more to reduce its packet drop to the same value as LPF/ODA.

Figures 4 and 5 show the normalized throughput and instantaneous queue size, respectively, for the LPF and LPF/ODA algorithms. It is seen that the throughput of both the algorithms are same until 133 s. This is due to the fact that the gateway goes through the active drop phase for both the algorithms. The instantaneous queue size starts falling at time 133 s indicating the end of the first long-term congestion. The LPF/ODA algorithm triggers the over drop avoidance phase as soon as the instantaneous queue level drops to low level (around five packets). The packet drop decreases when the instantaneous queue size reaches the low level as seen in Figure 7.

As a result, as shown in Figures 4 and 6, the throughput starts increasing and the queuing delay starts decreasing at 136 s. The response time of the LPF/ODA algorithm is therefore approximately $136 - 133 = 3$ s.

When the LPF algorithm is used at the RED gateway, the packet drop is high (due to active packet drop) even after the end of the first long-term congestion at 133 s. As a result, the throughput is low until 146 s, resulting in the response time of LPF to be $146 - 133 = 13$ s. *LPF, therefore, requires about* $13 - 3 = 10$ s *longer for the throughput to reach the same value as LPF/ODA.* At the end of long-term congestion, *LPF/ODA algorithm responds* $\frac{13}{3} = 4.3$ *times faster than the LPF algorithm.*

During the congestion-free period (133–166 s), LPF/ODA always has a much higher throughput than LPF. For example, at 150 s, the throughput of LPF/ODA is 0.1125 which is 80% higher than the value of 0.0625 for LPF. At time 166 s, where the congestion-free period ends and the second long-term congestion period starts, LPF/ODA has a throughput 0.15 which is 25% higher than the value of 0.12 for LPF.

A second long-term congestion occurs between 166 and 250 s. During this period, as expected, LPF/ODA switches to the active drop phase, and exhibits similar performance as that of LPF.

From the above discussion, besides tracking long-term congestion with active packets drops, compared with regular LPF algorithm, it is seen that *our proposed LPF/ODA algorithm possesses a distinct advantages* to quickly reacts to the end of long-term congestion thereby avoiding unnecessary packet drops. As a result, *our LPF/ODA algorithm significantly improves the response time of RED gateways at the end of long-term congestion*. Considering the large amount of traffic in today's Internet, this improvement at RED gateways could result in noticeable improvement in the performance of next generation Internet applications.

4. CONCLUSIONS

We have proposed the LPF/ODA algorithm to calculate the average queue length of an RED gateway. In the proposed LPF/ODA algorithm, an RED gateway alternates between two phases: active drop phase and over drop avoidance phase. The active drop phase enables an RED gateway to trace long-term congestion while filtering out short-term congestion. The over drop avoidance phase allows the gateway to react promptly when it leaves the active drop. *LPF/ODA therefore responds faster to a change in the long-term congestion as compared to the response time of a conventional RED. Results have shown that LPF/ODA significantly improves the response time of RED gateways, resulting in an improvement in throughput and delay.*

REFERENCES

1. Floyd S, Jacobson V. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking* 1993; **1**(4):397–413.
2. May M, Diot C, Lyles B, Bolot J. Influence of active queue management parameters on aggregate traffic performance. *Research Report*, Institut National de Recherche en Informatique et en Automatique, April 2000.
3. Christiansen M, Jeffay K, Ott D, Smith FD. Tuning RED for web traffic. *Proceedings of ACM SIGCOMM2000*, Stockholum, Sweden, August 2000.
4. Braden B, Clark D, Crowcroft J, Davie B, Deering S, Estrin D, Floyd S, Jacobson V, Minshall G, Partridge C, Peterson L, Ramakrishnan K, Shenker S, Wroclawski J, Zhang L. Recommendations on queue management and congestion avoidance in the Internet. *RFC* 2309, April 1998.

5. Jacobson V. Notes on using RED for queue management and congestion avoidance. www.aciri.org/floyd/red.html, June 1998.
6. Floyd S. RED: Discussion of setting parameters. www.aciri.org/floyd/REDparameters.txt, November 1997.

AUTHORS' BIOGRAPHIES

Bing Zheng received his PhD degree in electrical engineering in August 2001, from University of Dayton. He received a BSc in physics in 1984 and an MSc in electro-optics in 1990. Before joining the Department of Electrical and Computer Engineering of University of Dayton in August 1997, he was an associate professor working in tunable WDM technology, optical fibre technology, integrated optoelectronic technology, and flat panel display technology with the University of Electronic Science and Technology of China (UESTC). His working areas include TCP, IP, ATM, multimedia transmission, VHDL/FPGA/Digital system design, tunable WDM technology, fibre optical technology, and photonic/optoelectronic technology, etc.

Mohammed Atiquzzaman received the MSc and PhD degrees in electrical engineering and electronics from the University of Manchester Institute of Science and Technology. Currently he is faculty member in the School of Computer Science the University of Oklahoma, USA.

He has served as senior editor of the IEEE Communications Magazine, and currently serves on the editorial boards of the Computer Communications journal, Telecommunications Systems journal, Real Time Imaging journal and International Journal on Wireless and Optical Communications. He has guest edited special issues on many topics in various journals. He is the co-chair of the IEEE Workshop on High Performance Switching and Routing, co-chair of the SPIE QoS over Next Generation Internet Conference. He has also served in the technical programme committee of many national and international conferences including IEEE INFOCOM, IEEE Globecom and IEEE Annual Conference on Local Computer Networks. He is a senior member of the IEEE. He is the co-author of the book 'Performance of TCP/IP over ATM.'

His current research interests are in Radio Frequency transmission over Optical Networks, Next Generation Internet, Broadband Networks, Multimedia, multiprocessor systems and image processing. He has over 100 referred publications in the above areas, most of which can be accessed at www.cs.ou.edu/~atiq. He can be contacted at atiq@ou.edu.