Effective Utilization of Router Buffer by Threshold Parameter Setting Approach in RED

Kiran Chhabra
Research Scholar
Computer Science & Engineering
Dr. C. V. Raman University,
Bilaspur (C. G.)

Manali Kshirsagar
Professor,Computer Technology, Yashwantrao Chawan College of Engineering
Nagpur (MS)

Arun Zadgaonkar
Ex. Vice Chancellor
Dr. C. V. Raman University,
Kargi Road
Bilaspur (C. G.)

ABSTRACT

Active Queue Management (AQM) provide solution to Network congestion of Internet. Random Early Detection(RED) is the first well known AQM recommended by Internet Engineering Task Force (IETF) used for congestion avoidance for last three decades. RED has some disadvantages like hard tuning of control parameters, inefficient congestion notification, insensitivity to variation of traffic load. In this research work above issues are addressed and a mixed approach of threshold parameter tuning in terms of router buffer space and inclusion of traffic load in congestion notification along with average size is used. The approach is given the name LTRED, here L is for length of buffer and T stands for Threshold. In this research work, impact of variation of router queue size in terms of bandwidth under different traffic load scenario is observed and compared with standard AQM's like RED, ARED and AVQ. Extensive simulations using ns-2 simulator demonstrates that LTRED outperforms others in terms of effective utilization of router buffer space, less packet loss, high goodput and high link utilization.

Keywords

Active Queue Management (AQM), Average queue size, Congestion Avoidance, Network Simulator (ns), Random Early Detection (RED).

1. INTRODUCTION

Network performance is affected by congestion phenomenon [3]. It occurs on shared networks when multiple user contend for access to the same resource like buffer space, queue and bandwidth. To handle congestion AQM approach is suggested by IETF [1] for preventing packet loss due to buffer overflow. Basic idea behind an AQM algorithms is to convey congestion notification early enough to the senders so that senders are able to reduce their transmission rates before queue overflows and prevents many packet losses in coming time. IETF recommended RED [2] as the first AQM which is widely used.

RED was proposed by Sally Floyd et.al in 1993 [2]. In RED algorithms in order to anticipate congestion buffer queue size is averaged using Exponential Weighted Moving Average (EWMA) given by equation (1). In order to find sustained congestion not the transient one average of current queue size is calculated. Two thresholds Minimum i.e. Min_T and Maximum threshold i.e. Max_T are used for packet dropping. After calculation of average queue size it is compared with thresholds if average queue size is less than Min_T every packet is enqued and if average greater than Max_T then every incoming packet is discarded. When average size lies between two thresholds packets are randomly dropped using drop probability calculated by equation (2) & (3).

$$a_{i+1} \longleftarrow (1 - w)a_i + wq_i \qquad (1)$$

$$P_b \longleftarrow P_m(a-Min_T) /(Max_T - Min_T) \qquad (2)$$

$$P_a \longleftarrow P_b /(1-c \times Pb) \qquad (3)$$

Where

$\mathbf{a_{i+1}}$	:	average queue size at (i + 1)th time
$\mathbf{a_i}$	:	average queue size at ith time
w	:	moving weighted average constant
$\mathbf{q_i}$	:	current queue size
Min_T	:	Minimum threshold for queue
Max_T	:	Maximum threshold for queue
$\mathbf{P_m}$	:	Maximum value for P_b
$\mathbf{P_a}$	:	current packet marking probability
$\mathbf{P_b}$	:	temporary probability used in calculation
c	:	packets since last discarded packet

It is now widely accepted that RED's controlled queue performance is better than drop-tail queue however, the inherent design of RED makes it difficult to parameterize RED queues to give good performance under different network scenarios. Hence, having been widely used in combination with TCP for several years, RED has not found acceptance in the internet research community. It has some disadvantages like hard parameter setting, congestion depend on parameters, insensitivity towards input traffic and there is no matching between average queue length and current queue length. In recent years research activities have came out to solve the short comings of RED. The existing schemes use various factors on metrics to detect congestion. The schemes are based on congestion metrics like queue length [5,7, 12,13], load both queue and load [6,8]. Some have used control theory principles [4,11] and some researchers also used artificial intelligence tools [9] to detect congestion.

This paper has been organized in the following manner section 2 gives the proposed work and explained about approach used, section 3 gives simulation results and section 4 deals with conclusions and final section gives all the references made in completing the present work.

2. PROPOSED WORK

RED suffers from drawbacks like hard parameters setting and its performance degrades during varying network conditions, its unstable behaviour is observed by wild oscillations in current queue length and more packet losses occurred during congestion time. Packet dropping decision is taken on the basis of value of average queue size whether it is below, above or between the two thresholds. Hence, careful setting of threshold value is important. We have set Min_T = 40% of buffer size and Max_T = 70% of buffer size, so that buffer packet loss should be controlled properly. Some research work has been already done earlier[14].

Traffic load variation affects the current queue length in order to cope-up with the traffic load changes which is reflected by change in current queue size and to find out sustained congestion and not transient one packet dropping decision is supported by above these two factors. Here the early congestion notification is done on the basis of not only average queue size but also current queue size.

2.1 Proposed Algorithm

For each packet arrival following steps are carried out :

(1) Average queue length is calculated using equation (1).

(2) If current queue length is greater than 75% of buffer value increment average queue value by 5% of current queue value.

(3) If average queue value is less than Min_T value enqueue the packet.

(4) If, current queue value greater than 85% of queue buffer value do following step:-

```
    if average value is greater than 60% of
    buffer value, drop the packet.
    else
    if average is greater than $Min_T$, drop
    the packet according to RED Algorithm.
```

(5) If average value greater than Max_T, drop the packet.

3. SIMULATION RESULTS AND DISCUSSION

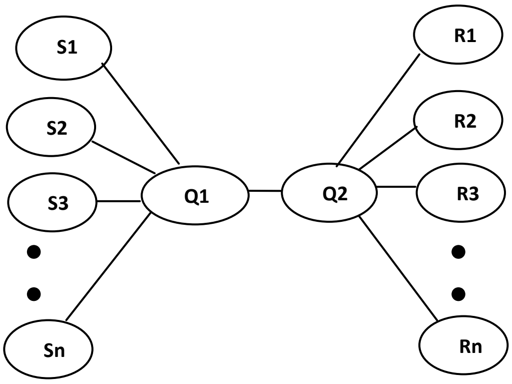


Fig 1: Network Topology

We have implemented the proposed work using network simulator ns-2[15]. We have compared our approach with standard AQM’s like RED [2], ARED [7] and AVQ [5]. Basic network technology which is used is shown in figure-1. Here S1,S2, R1 and R2 have bandwidth of 2Mb and delay of 10 ms, S3 to Sn and R3 to Rn have bandwidth of 10Mb and delay from 14ms to 15ms. Q1 to Q2 is the bottleneck router which has bandwidth variation from 0.8Mb to 1.8Mb having delay of 20ms.

Table 1. Different Bandwidths Values

Bandwidth	RTT	Buffer Size
0.8 Mb	75 ms	30 Packets
1.1 Mb	70.9 ms	40 Packets
1.45 Mb	68.3 ms	50 Packets
1.8 Mb	66.7 ms	60 Packets

The parameters used for RED and ARED are Min_T = 0.15 x buffer size, Max_T =3x Min_T,. For LTRED, Min_T =40% of buffer size and Max_T =70% of buffer size. Other parameters for RED, ARED and LTRED are w=0.002, P_m=0.1. For AVQ, the parameters are default parameters used in ns-2. TCP Reno is used for all AQM and mean packet size is 500. Two FTP sessions randomly starts in between to 0 to 0.01 seconds and last to end that is 30 seconds. In middle of the simulation another m FTP session would randomly start in between 10.0 to 10.1 seconds and last to the end which is to simulate change of network conditions. We have observed results for source varying m from 30 to 60 sources to show the improvement of proposed method and we have drawn different graphs for all the approach used for bandwidth=1.8Mb and buffer size=60. We have observed performance in case of packet loss, packets delivered, goodput values, average queue size and current queue size.

We have made tables for all the four cases of bandwidth variation and queue size (Table-1) and drawn graphs for only one case i.e. the last one. The tables number 02 to 05 give the percentage value for packets lost, packets delivered, good put values and link utilization indicating that LTRED outperforms all the three cases. In case of LTRED delay is more compared to all the other cases.

The graphs drawn for different cases are from figure 2 to 10. Figure 2 indicates number of packets lost for sources from 30 to 60 showing that number of packets lost are minimum in case of LTRED which indicates that for the case of LTRED re-transmission of packets will also be less as the packets are not to be transmitted again utilizing the link for the new packets. Figure 3 gives number of packets delivered through out the simulation time and shows that it is maximum in case of LTRED which indicates that throughput as well as bandwidth utilisation is maximum in case of LTRED. Figure 4 gives us packet arrival ratio and it indicates that apart from AVQ, LTRED packet arrival is less indicating effective congestion indication. Figure 5 indicates current queue size for all the four cases, which shows that in case of LTRED buffer utilization is maximum which indicates under utilization of link is not there. Figure 6 indicates link utilization or bandwidth utilization, which is also maximum in case of LTRED. Figure 7 indicates average queue size for RED, ARED & LTRED showing stable behaviour in case of LTRED. From Figure 8 to 10 shows average queue size and current queue size behaviour for RED, ARED & LTRED. From these figures, it is clear that mismatch behaviour of both queue sizes is reduced very much in case of LTRED.

Table 2 : Different Values for Bandwidth = 0.8 MB and sources from 30-60

Queue Size =30							
AQM type	**Packets Delivered %**	**Packets Lost %**	**Packets Goodput %**	**Packets Retransmission %**	**Packets yet to be transmitted %**	**Link Delay %**	**Link Utilization %**
S/30							
RED	86.43	14.14	77.71	61.65	38.35	0.1031	90.31
ARED	85.28	14.45	77.21	55.87	44.13	0.106	91.56
AVQ	84.58	15.39	73.10	74.56	25.43	0.0673	87.47
LTRED	86.76	12.91	80.04	52.09	47.91	0.13	91.65
S/40							
RED	82.61	17.09	72.36	59.94	40.06	0.1054	90.36
ARED	83.24	16.54	72.00	67.95	32.05	0.1062	91.3
AVQ	82.77	17.16	75.64	41.50	58.50	0.0672	87.66
LTRED	84.22	15.47	76.66	48.85	51.15	0.13	91.83
S/50							
RED	80.92	18.83	70.39	55.94	44.06	0.1053	90.45
ARED	80.87	19.00	69.49	59.86	40.14	0.109	91.69
AVQ	80.61	19.35	72.75	40.66	59.34	0.0672	87.84
LTRED	82.22	17.43	74.32	45.32	54.68	0.132	91.74
S/60							
RED	78.69	21.12	70.15	40.44	59.56	0.1074	91.06
ARED	78.73	21.06	71.31	35.24	64.76	0.111	91.54
AVQ	78.80	21.18	69.79	42.53	57.46	0.0674	88.03
LTRED	80.56	19.13	71.79	45.81	54.19	0.131	92.49

Table 3 : Different Values for Bandwidth = 1.1 MB and sources from 30-60

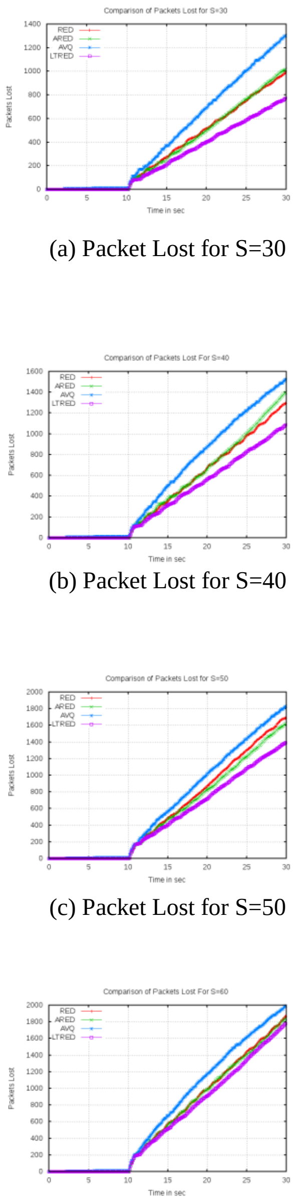
Queue Size =40							
AQM type	**Packets Delivered %**	**Packets Lost %**	**Packets Goodput %**	**Packets Retransmission %**	**Packets yet to be transmitted %**	**Link Delay %**	**Link Utilization %**
S/30							
RED	87.89	11.79	78.32	81.11	18.88	0.087	88.92
ARED	87.35	12,43	79.65	61.88	38.11	0.085	89.10
AVQ	86.65	13.34	78.41	61.79	38.21	0.051	86.85
LTRED	89.32	10.26	79.53	95.45	4.55	0.12	91.55
S/40							
RED	85.51	14.53	77.04	56.11	43.89	0.088	89.90
ARED	85.39	14.44	76.88	58.91	41.08	0.086	89.52
AVQ	85.18	14.81	77.22	53.74	46.26	0.053	87
LTRED	86.75	12.89	78.28	65.72	34.27	0.12	91.67
S/50							
RED	83.32	16.32	73.32	61.26	38.74	0.089	89.05
ARED	83.56	16.32	74.07	61.84	38.15	0.091	89.97
AVQ	82.71	17.26	72.53	58.96	41.03	0.053	87.1
LTRED	84.86	14.82	74.97	66.67	33.33	0.123	91.8
S/60							
RED	82.01	17.75	70.95	63.74	36.26	0.901	88.83
ARED	81.62	18.21	70.39	61.64	38.35	0.092	90.02
AVQ	82.23	17.62	72.93	52.81	47.19	0.053	87.23
LTRED	83.93	16.31	75.41	52.27	47.73	0.125	91.92

Table 4 : Different Values for Bandwidth = 1.45MB and sources from 30-60

Queue Size =50							
AQM type	**Packets Delivered %**	**Packets Lost %**	**Packets Goodput %**	**Packets Retransmission %**	**Packets yet to be transmitted %**	**Link Delay %**	**Link Utilization %**
S/30							
RED	90.71	9.10	84.14	72.25	27.74	0.081	90.70
ARED	87.35	12.43	79.65	67.06	32.93	0.085	90.73
AVQ	89.10	10.88	84.12	45.81	54.19	0.05	86.86
LTRED	91.60	8.21	84.02	92.24	7.76	0.106	91.47
S/40							
RED	87.94	11.86	80.23	64.99	35.00	0.083	90.91
ARED	85.39	14.44	76.88	67.07	32.93	0.086	91.00
AVQ	86.79	13.19	81.30	41.67	58.33	0.049	86.96
LTRED	88.80	11.02	79.06	88.36	11.64	0.109	91.57
S/50							
RED	85.92	13.84	75.04	78.65	21.34	0.087	91.10
ARED	83.56	16.32	74.07	48.04	51.96	0.091	91.13
AVQ	85.74	14.24	80.25	38.57	61.43	0.05	87.05
LTRED	86.82	12.84	78.76	62.54	37.54	0.111	91.66
S/60							
RED	84.84	15.90	73.84	62.90	37.09	0.087	91.14
ARED	81.62	18.21	70.39	49.72	50.28	0.092	91.23
AVQ	83.36	16.63	75.95	44.54	55.46	0.05	87.15
LTRED	85.55	14.11	77.15	59.51	40.49	0.111	91.75

Table 5 : Different Values for Bandwidth = 1.8 MB and sources from 30-60

Queue Size=60							
AQM type	**Packets Delivered %**	**Packets Lost %**	**Packets Goodput %**	**Packets Retransmission %**	**Packets yet to be transmitted %**	**Link Delay %**	**Link Utilization %**
S/30							
RED	91.76	7.92	87.15	58.19	41.81	0.070	91.25
ARED	91.31	8.53	85.56	67.43	32.57	0.068	90.80
AVQ	89.72	10.27	82.54	69.86	30.14	0.038	87.24
LTRED	93.22	6.45	87.34	90.86	9.14	0.09	91.82
S/40							
RED	89.35	10.40	81.01	80.01	19.99	0.073	91.53
ARED	88.85	11.02	82.07	61.47	38.53	0.068	91.34
AVQ	87.69	12.30	81.35	51.54	48.47	0.033	87.31
LTRED	90.72	8.95	83.61	79.42	20.58	0.095	91.84
S/50							
RED	87.31	12.51	76.79	84.01	15.99	0.074	91.72
ARED	87.32	12.50	79.78	60.24	39.76	0.074	91.63
AVQ	85.81	14.18	78.40	52.27	47.74	0.032	87.4
LTRED	88.45	11.26	80.09	74.16	25.84	0.098	91.96
S/60							
RED	85.69	14.28	76.83	62.03	37.97	0.075	91.89
ARED	85.50	14.31	75.86	67.37	32.63	0.073	91.91
AVQ	84.93	15.07	79.51	35.99	64.01	0.032	87.48
LTRED	87.20	12.43	77.65	76.80	23.19	0.10	92.04



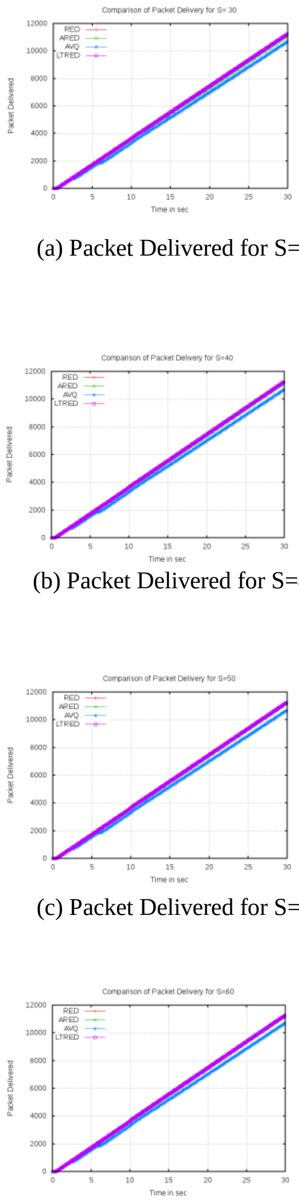
(a) Packet Lost for S=30

(b) Packet Lost for S=40

(c) Packet Lost for S=50

(d) Packet Lost for S=60

Fig. 2: Packet Lost



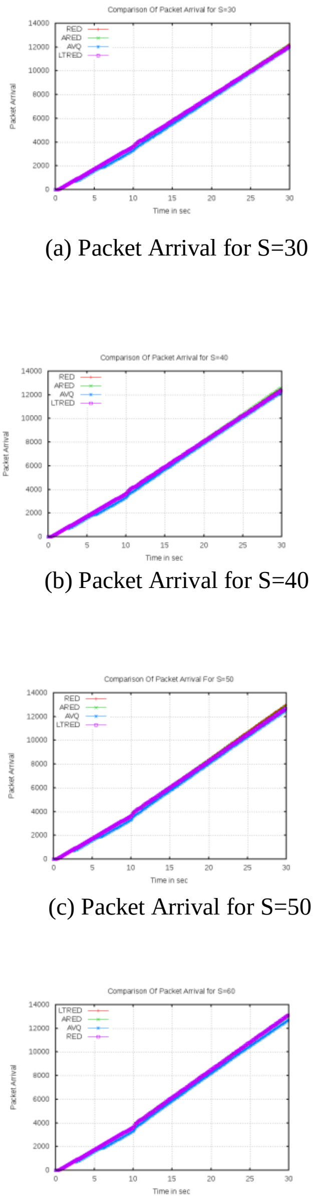
(a) Packet Delivered for S=30

(b) Packet Delivered for S=40

(c) Packet Delivered for S=50

(d) Packet Delivered for S=60

Fig. 3 : Packets Delivered



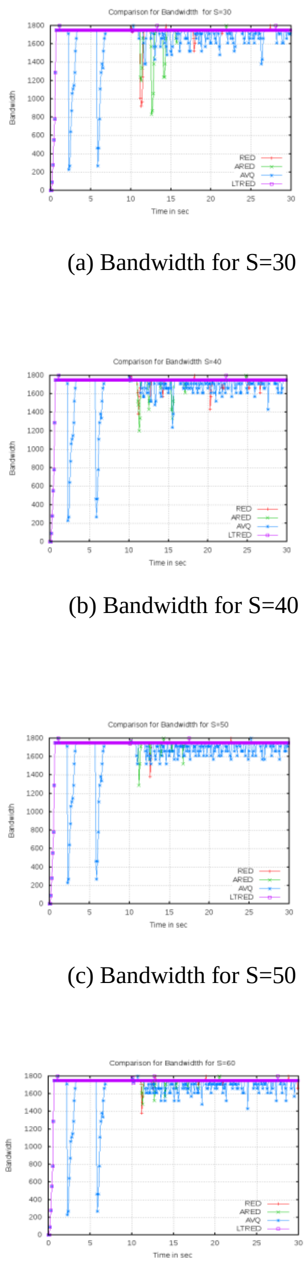
(a) Packet Arrival for S=30

(b) Packet Arrival for S=40

(c) Packet Arrival for S=50

(d) Packet Arrival for S=60

Fig.4 : Packets Arrival



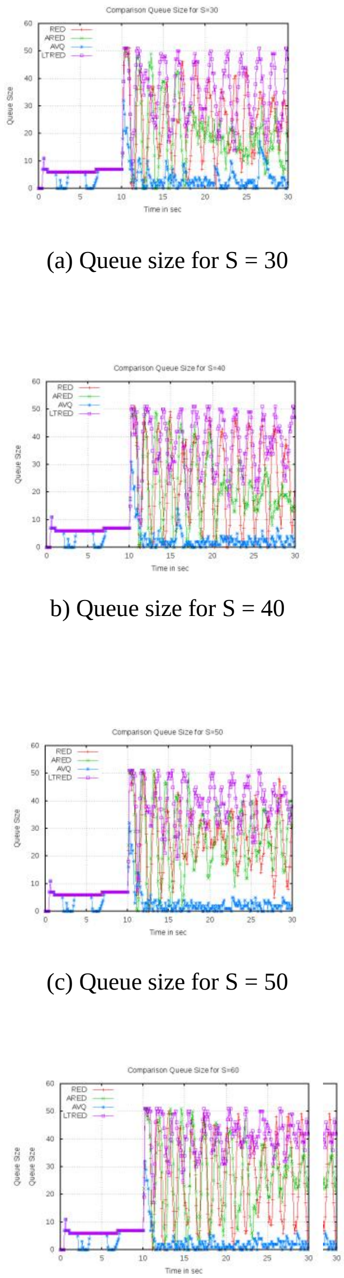
(a) Bandwidth for S=30

(b) Bandwidth for S=40

(c) Bandwidth for S=50

(d) Bandwidth for S=60

Fig 5: Bandwidth



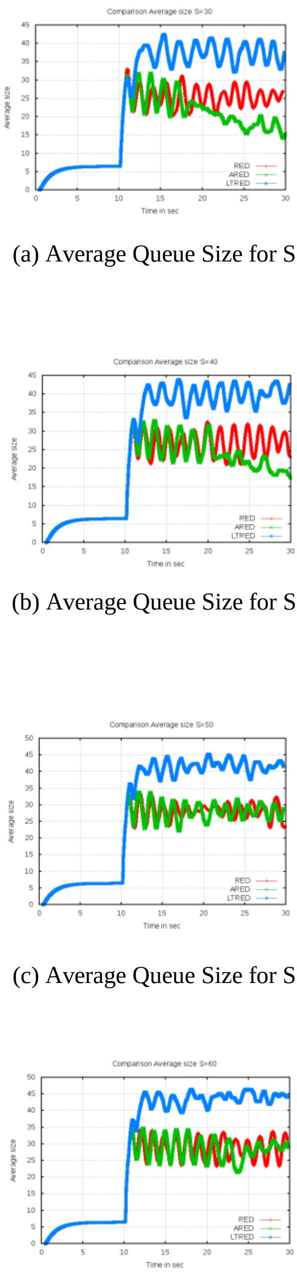
(a) Queue size for S = 30

b) Queue size for S = 40

(c) Queue size for S = 50

(d) Queue size for S = 60

Fig 6 : Queue Size



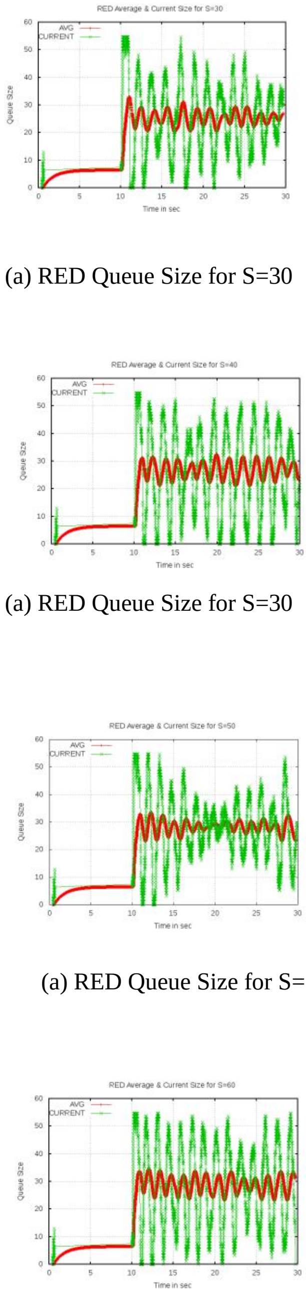
(a) Average Queue Size for S=30

(b) Average Queue Size for S=40

(c) Average Queue Size for S=50

(d) Average Queue Size for S=60

Fig 7 : Average Queue Size



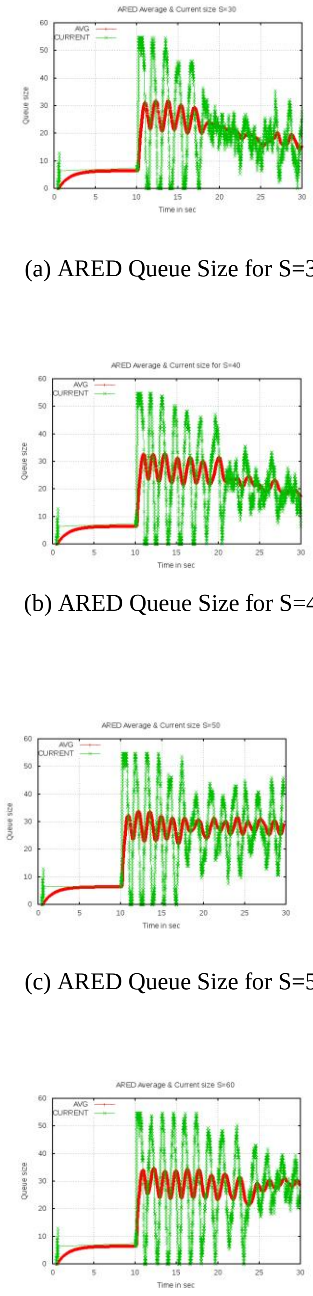
(a) RED Queue Size for S=30

(a) RED Queue Size for S=30

(a) RED Queue Size for S=30

(a) RED Queue Size for S=30

Fig 8 : RED Queue Size



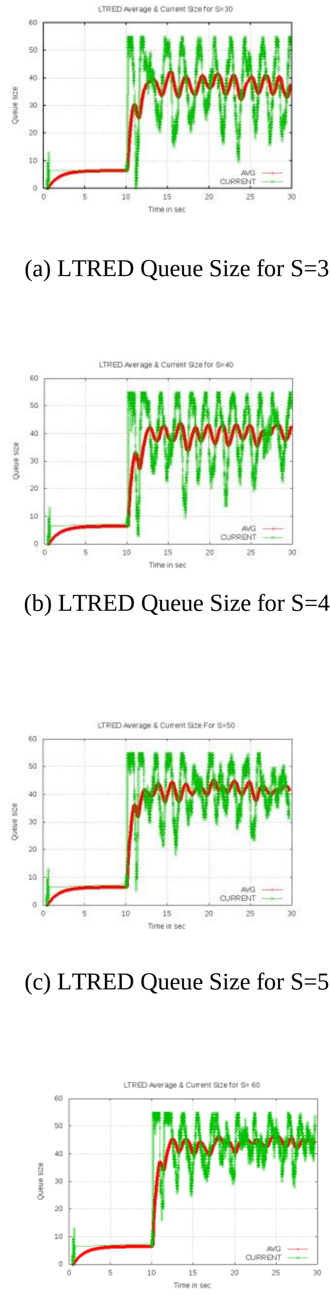
(a) ARED Queue Size for S=30

(b) ARED Queue Size for S=40

(c) ARED Queue Size for S=50

(d) ARED Queue Size for S=60

Fig 9 : ARED Queue Size



(a) LTRED Queue Size for S=30

(b) LTRED Queue Size for S=40

(c) LTRED Queue Size for S=50

(d) LTRED Queue Size for S=60

Fig 10 : LTRED Queue Size.

4. CONCLUSION

In this reason a novel algorithm LTRED is proposed. By making small changes to RED algorithm, the performance improvement aim is achieved. The performance metrics which are evaluated to show the advantages of LTRED are low packet loss ratio, high goodput value, better congestion indication, reduced oscillations in queue length and efficient utilisation of router buffer space. The algorithm is compared with standard AQM like RED, ARED and AVQ and LTRED gives better performance as compared to other which is illustrated through tables and graphs. Performance of LTRED is observed under varying bandwidth, queue size and load variation.

5. REFERENCES

[1] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski and L. Zhang RFC 2309: Recommendations on Queue Management in April (1998)

[2] S. Floyd and V. Jacobson, "Random early detection gateway for Congestion avoidance, "IEEE/ACM Transaction on Networking, vol. 1, no.4, pp.397-413, August (1993)

[3] Seunwan Ryu, Christopher Rump, And Chunming Qiao "Advances in Internet Congestion Control", Volume 5, No.1 http://www.comsoc.org/pubs/surveys, third quarter (2003)

[4] J. Sun, K. Ko, G Chen, M Zukermam," PD-RED: To Improve the performance Of RED", IEEE Communication letters 7(8) 406-408, (2003)

[5] Srisankar S. Kunniyur, R. Srikant, " An Adaptive Virtual Queue [AVQ] for Active Queue Management", IEEE/ACM Transactions on Networking, April (2004)

[6] Athuraliya, D.E Lapsley, S.H Low" Random Exponential Marking for internet congestion control" IEEE Transactions on Network, June (2001)

[7] S. Floyd, R. Gummadi, S. Shenkar,"Adaptive RED: An algorithm for Increasing the robustness of RED's active Queue Management", Berkely CA, [online] http://www.icir.org/floyd/red.html(2001)

[8] C.Wang, J.Liu, B.Li, K.Sohraby, YT. Hou, "LRED: a robust and responsive AQM algorithm using packet loss ratio measurement," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 1, pp. 29-43, 2007.

[9] B. Hariri, N. Sadati, "NN-RED: an AQM mechanism, based on neural networks," Electronics Letters, vol. 43, no. 19, pp. 1053-1055, 2007.

[10] M. Li and W. Zhao, "Representation of a stochastic traffic bound", IEEE Transactions on Parallel and Distributed Systems, vol. 21, no 9 pp. 1368-1372, (2010).

[11] Hao Wang, Zilong Ye, Bo Wang," Using auto-tuning proportional integral probability to improve random early detection" Communication Technology (ICCT), IEEE 13th International Conference, (2011).

[12] AH. Ismail, A. El-Sayed, Z. Elsaghir, IZ. Morsi, "Enhanced Random Early Detection (ENRED)," International Journal of computer Applications, vol. 92, no. 9, pp-25-88, 2014.

[13] Shahram Jamali, Neda Alipasandi, and Bita Alipasandi, "An Improvement over Random Early Detection Algorithm: A Self-Tuning Approach" Journal of Electrical and Computer Engineering Innovations, JECEI, Vol.2, No. 2, 2014.

[14] Kiran Chhabra, Manali Kshirsagar, Arun Zadgaonkar, "Effect of Load and Threshold Variation on Performance of RED : Random Early Detection" International Journal of Science and Research, vol. 4, issue 6,ISSN (online) :2319-7064 June (2015).

[15] "ns [network simulator]", [Online] Available http://www.isi.edu/nsnam/ns, (1999)