INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Enhanced Queue Management Scheme for TFRC Over Wired Networks

Ramanjaneya Reddy Nalavala [1]* Chenna Reddy Pakanati [2] Padmavathamma Mokkala [3]

[1]*Vaagdevi Institute of Technology and Science, Proddatur, Andhra Pradesh, India*
[2] *Jawaharlal Nehru Technological University Anantapur, Andhra Pradesh, India*
[3]*Sri Venkateswara University, Tirupati, Andhra Pradesh, India*
* Corresponding author's Email: ramanji.nalavala@gmail.com

Abstract: Multiple senders share bandwidth and send their data in high speed over the wired network. This may lead to congestion in the network. Congestion control plays a prominent role in network resource management. Effective mechanisms are needed to control the congestion in the network. In Transport layer, TCP Friendly Rate Control (TFRC) has been widely used for advanced multimedia streaming application over wired networks to give best smoothing sending rate. However, there exists packet loss due to high traffic in the network. To alleviate the barriers of congestion and reduce packet loss in the network, one needs to implement best queue management mechanisms to help the TFRC. In this paper, we have applied enhanced queue management scheme to TFRC. The simulation results have shown that this strategy reduces packet loss and increases throughput.

Keywords: Congestion control, Queue management, RED, TCP, TFRC.

1. Introduction

Internet runs on TCP/IP protocol suite**.** Due to various real time application requirements, some protocols have been added to TCP/IP protocol suite. One of the important layers in internet protocol suite is Transport layer, to support end to end communication between the two hosts in the network. Quality of Service is one of the primary issues in transport protocols. Different applications require different data transmission rates over internet. These applications transmit with high data rate which leads to congestion in the network. Congestion control is a primary concern in a computer network. Many congestion control protocols are used to avoid or reduce congestion in the network.

Internet Engineering Task Force (IETF) standards concerning end-to-end congestion control focus either on specific protocols [1] [2] or on the syntax and semantics of communications between the end nodes and routers about congestion information [3] or desired quality-of service.

Various congestion control protocols have been developed in transport layer. TCP (Transmission Control Protocol) [4] is not only a reliable protocol but also controls congestion which has preserved the stability of the Internet. Congestion affects the network efficiency, which was successfully addressed by TCP. Key essential of the TCP for stability of the network is its mechanism of congestion control and reliability.

The trend has changed and majority of the streaming and real-time applications are starting to use UDP where TCP is suitable for the task. UDP doesn't control congestion and is unreliable protocol [5]. However, there is a rapid increase of multimedia applications over the internet for which TCP and UDP are not well suited. Hence we require protocols which have congestion control mechanism which compete with TCP and UDP without compromising quality of service. This is the primary motivation to design a new protocol named TFRC. TCP Friendly Rate Control Protocol (TFRC) is proposed [6] which adjusts its sending rate based on TCP Reno Eq. (1).

DOI: 10.22266/ijies2017.0228.03

$$X = \frac{S}{RTT\sqrt{\frac{2bp}{3}} + 3 * RTO\sqrt{\frac{3bp}{8}}\,p(1+32p^2)} \qquad (1)$$

Where X is the expected sending rate in bytes per second, S is the packet size in bytes, *RTT* is the round-trip time, *b* is the number of packets that are acknowledged by one ACK, *p* is the loss event rate, and *RTO* is the retransmit timeout value. This equation maintains a steady state sending rate to network to avoid the abrupt fluctuations.

By setting b value to 1 and retransmission time out value to 4 times of round trip time, then TFRC sending rate X is a function of packet size which is represented as Eq. (2).

$$X = \frac{S}{RTT\sqrt{\frac{2p}{3}} + 12 * RTT\sqrt{\frac{3p}{8}}\,p(1+32p^2)} \qquad (2)$$

Based on the above equation, TFRC congestion control mechanism works as follows.

- ✓ At first step, the loss event rate will be measured by receiver and pass this information to sender.
- ✓ Sender measures round trip time based on received information
- ✓ Sender measures its sending rate as a function of loss event rate and round-trip time
- ✓ Sender maintains its sending rate based on its calculated rate from the equation

Finally TFRC sending rate is primarily a function of loss event rate and round trip time. The performance of TFRC [7] is analyzed and is compared with TCP and UDP in wired environment. TFRC performance is better than TCP and UDP in terms of throughput, packet loss ratio and end to end delay.

To obtain high throughput for online multimedia streaming applications, a more sophisticated congestion control mechanism and best queue management is required. This encourages the author to propose a new technique for finding average queue length and a new methodology introduced for auto tuning of parameters to adapt dynamic real time situations. The main advantages of this proposed method is TFRC with enhanced queue management results in achieving high throughput and less packet loss ratio.

This paper is organized as follows. Section 2 describes related work done so far on TFRC. Section 3 presents the proposed methodology for enhancing the queue management. In Section 4, Simulation environment is described. Metrics are reported in Section 5. Simulation results and discussions of using proposed technique to improve throughput of TFRC are included in Section 6. Section 7 presents conclusion.

2. Related works

Several attempts were made to analyze and enhance the performance of TFRC. In [8], Authors suggested a new slow start mechanism and bandwidth estimation methods to improve the performance of TFRC in terms of better RTT fairness and reduce burst packet losses. In [9], Authors proposed a new receiver based retransmission scheme for TFRC to to improve the effective retransmission ratio according to the forward path delay variance. This scheme is more suitable for fluctuating networks. In [10], Authors proposed an enhanced TFRC for high quality video streaming over high bandwidth delay product networks. This scheme reduces the packet losses of slow-start, and provides RTT-fairness. In [11], an enhanced TFRC control algorithm called NSTC, which introduces available bandwidth variation as network identification and limits sending rate in different network states. This scheme enhances TFRC performance in terms of decreased packet loss and comprehensively improves QoS and friendliness

To control and avoid congestion [12] in the network, Queue management is a primary aspect in transport layer protocol. Many transport layer protocols uses one of popular active queue management algorithm called Random Early Detection (RED) [13], which drops packets with certain probability which is computed based on an exponential weighted moving average (EWMA). Mostly, TCP Friendly Rate Control (TFRC) protocol uses RED algorithm as its queue management. In recent years, RED had undergone a number of changes with effect of dynamic requirements.

Feng et al. [14] proposed a suitable varying mechanism for RED parameter named maximum marking probability ($\max_p$) which significantly reduce packet loss over congested links. In this study, by changing $\max_p$ parameter according to the observed traffic by constant factor α and β depending on which threshold it crosses better results are obtained. Sally Floyd et al [15] proposed Adaptive RED which is an extension of RED, in which RED parameter sensitivity is reduced.

Ahmed E kamal, Manzoor Murshed [16] proposed dynamic threshold adjustment to RED, to maximize throughput and minimize packet dropping and delay. Alpaben K. Patel, Jyothi Divecha [17] introduced modified exponential weighted moving average, which is very effective in detecting small and abrupt shifts in monitoring process mean.

Arjuna Sathiaseelan, Gorry Fairhurst, [18] proposed the introduction of congestion control for multimedia traffic to ensure the stability of the next generation Internet. TFRC algorithm was first specified in RFC 3448 [19]. S.Floyd, E.Kohler [20] proposed TFRC-SP (Small-Packet TFRC) a variant of TFRC which supports fixed sending rate by using variable sized small packets. The design goal for TFRC-SP is to achieve the same bandwidth in bps as a TCP flow using packets of up to 1500 bytes. Pedro Reviriego Vasallo[21] proposed an extension to the TFRC protocol in order to support variable packet size flows. In this paper, numerator of TFRC equation has changed to MTU, so that it is suitable for variable packet size flows. Mohammad A. Talaat, Gamal M. Attiya, and Magdi A. Koutb [22] predicted that Video traffic is booming over Internet and to be the prevailing traffic type in the coming few years. TFRC is the most promising candidate congestion control algorithm over Internet that handles such type of traffic appropriately satisfying its QoS requirements. Agnieszka Chodorek and Robert R. Chodorek [23] suggest that although TFRC protocol is suitable for multimedia transmission it can be improved. They proposed to substitute the original TFRC throughput equation with a linear throughput equation. Results allow us to believe that the proposed linear equation is more suitable for multimedia transmission than the equation originally included in the RFC 3448. In [24], authors proposed a new congestion control framework called UTFRC, which is more suitable for multimedia streaming applications than classical TFRC. In [25], authors compared and analyzed the performance of TFRC with interoperation of TCP and UDP.

Most researchers used EWMA technique to find average queue length in queue management. But EWMA cannot perform well in predicting and adapting large shifts. Less research work has done on finding new average queue length, which is more adaptable to all situations. So there is a need to find a new technique for finding average queue length, which is suitable to all situations.

3. Proposed system

One of the popular active queue management is RED. This algorithm is based on some parameters(min_{th}, max_{th}, avg, max_{p}...etc.) to identify whether the packets are in the queue or dropped based on its average queue length measured by exponential weighted moving average(EWMA) and its probability metrics. RED algorithm uses EWMA principle to calculate its average queue length.
The average queue length formula in RED is represented as Eq. (3)

$$avg_i = (1-w_q)avg_{i-1} + w_q\, q_i \qquad (3)$$

variables:
w_q is queue weight, w_q ϵ(o,1]
avg represents average queue size
q represents current queue size.

Many researchers tuned the parameters of RED to increase its efficiency, but less work has been done on finding effective average queue length using EWMA with unpredictable network traffic and sudden changes in the intervals. EWMA can perform well to detect small interval changes based on its traffic, but it can't react to detect large shifts. Shewhart control formula can perform well to detect large interval changes than small intervals. So it is best to combine both of them to get benefits of both small and large shifts due to traffic changes.

Moving towards in this direction, we applied modified exponential weighed moving average formula for finding average queue length in RED algorithm. This modified exponential weighted moving average formula combines the features of Shewhart and EWMA in a simple way to detect both small and large traffic interval changes than earlier. Shewhart-EWMA resolves the issues when it is designed to detect all the changes. Shewhart-EWMA formula for finding average queue length is

$$avg_i = (1-w_q)avg_{i-1} + w_q\, q_i + (q_i - q_{i-1}) \qquad (4)$$

Also auto tuned min_{th} and max_{th} formulas are newly added based on increase in average queue size.

```
If $((max_{th}-min_{th})<(avg_i - avg_{i-1}))$
begin
$min_{th} = min_{th} + (q_i - q_{i-1}) + \alpha$          (5)
$max_{th} = min_{th}*3;$
end
```

variables:
w_q is queue weight, w_q ϵ(o,1]
avg represents average queue size

Application of all modifications to RED algorithm is depicted in Fig. 1:

```
Initialization of variables
Avg=0; count=-1;
Calculate average queue length for each packet arrival based on modified ewma
If (!queue empty)
$avg_i = (1-w_q)avg_{i-1} + w_q q_i + (q_i – q_{i-1})$
m-f(time-q_time)
$avg=(1-w_q)^m avg$
if(($max_{th}$-$min_{th}$)<($avg_i – avg_{i-1}$ ))
begin
$min_{th} = min_{th} +(q_i – q_{i-1})+ \alpha$
$max_{th} = min_{th}*3$;
end
if $min_{th}$ <=avg<$max_{th}$
count++;
Compute probabilities with following info
Every interval seconds
If(avg>target and $max_p$<=0.5)
$max_p = max_p + \alpha$
else
if(avg<target and $max_p$>=0.01)
$max_p = max_p *\beta$;
$p_b =max_p(avg-min_{th} )/(max_{th} –min_{th})$
$p_{a =}$ $p_b /(1-count*p_b)$
mark the arriving the packet with prob $p_a$
count=0;
else if $max_{th}$ <avg
mark the arriving packet; count=0;
else
 count=-1;
when queue is emplty
q_time=time;
variable description:
$min_{th:}$ minimum threshold  for queue
$max_{th}$ : maximum threshold for queue
$max_p$: maximum value  for $p_b$
avg: average queue size
time: current time
$p_z$: current packet marking probability
$w_q$: queue weight
q_time: start of the queue idle time
count: Number of packets marked
q: current queue size
```

Figure.1 Enhanced Pseudo code for RED algorithm

4. Simulation Environment

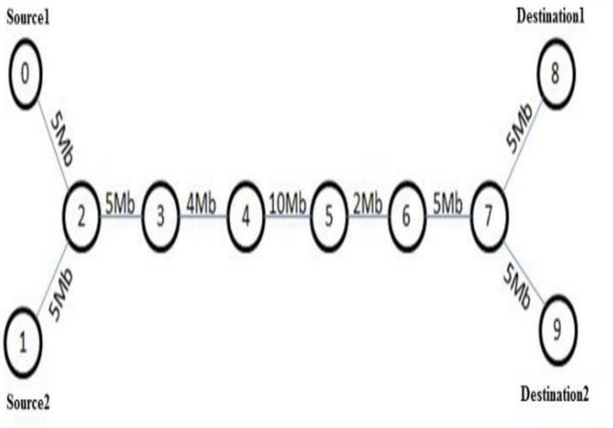

Figure.2 Dumbbell topology

Network simulator NS2.35 is one of the popular simulation tools which used to understand and predict the protocols behavior in networks. Dumbbell topology with multiple bottle neck links consisting of 10 nodes with different bandwidths like 5Mbps, 4Mbps, 10Mbps, and 2 Mbps with transmission delay 10ms is used. For both cases, TFRC packet size is fixed at 1000 bytes and total simulation time is 150 sec. NS2.35 is used as network simulator.

5. Performance metrics

Throughput:
Throughput is the rate at which a network sends or receives data. It is rated in terms of bits per second (bit/sec).

Packet Loss Rate:
Packet loss rate is the ratio between number of packets dropped or lost and number of packets sent through the network.

End-to-End delay:
The end-to-end-delay is averaged over all surviving data packets from the sources to the destinations.

Packet Delivery Ratio:
It is the ratio of data packets delivered to the destination to those generated by the sources.

6. Results and analysis

In this, we compare performance of modified TFRC and classical TFRC via throughput and packet loss rate.

Simulation results and analysis:

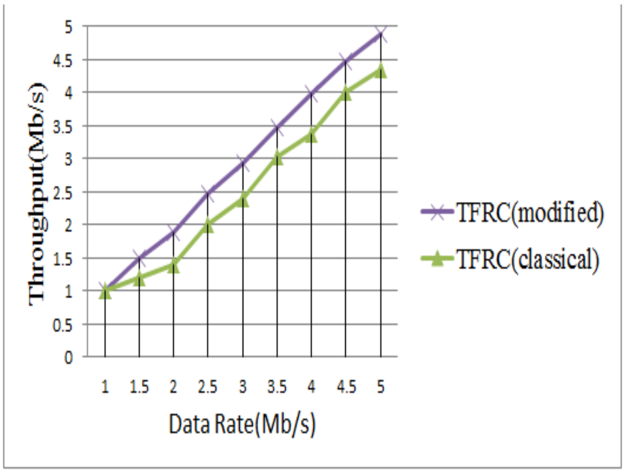

Figure.3 Comparison of throughput

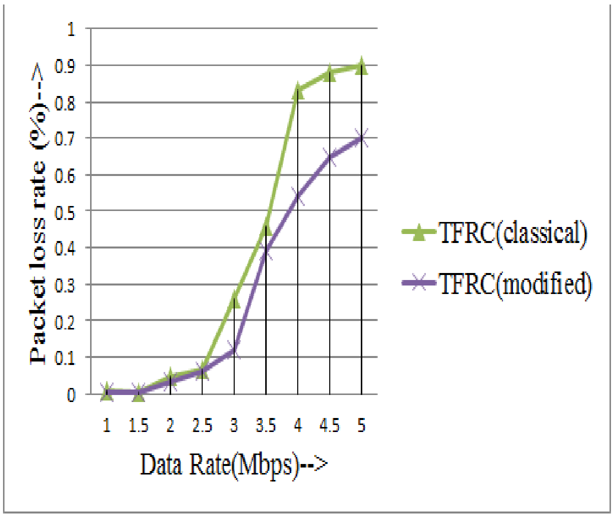

Figure.4 Comparison of packet loss ratio

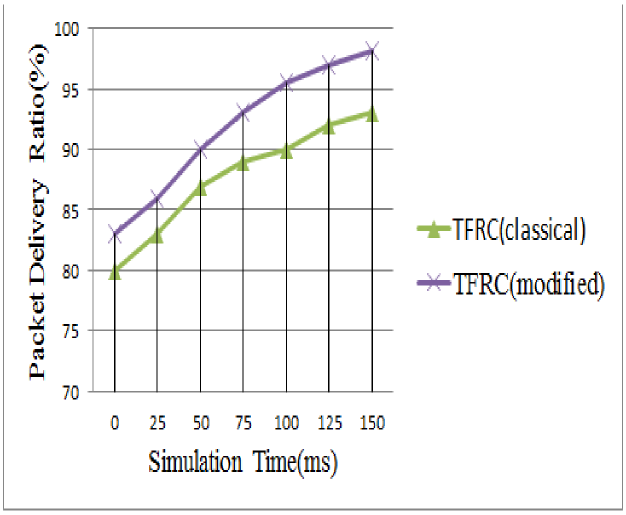

Figure.5 Comparison of Packet delivery ratio

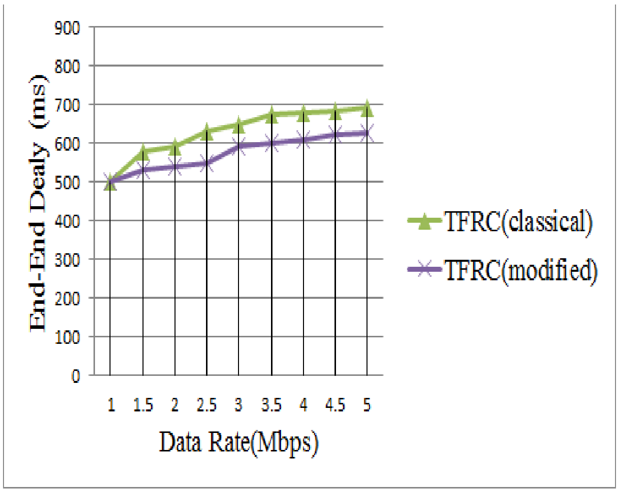

Figure.6 Comparison of End- End delay

Fig 3 depicts that throughput efficiency of TFRC with enhance queue management. With increasing simulation time, TFRC with enhance queue management (modified TFRC) gave good throughput than classical TFRC. When comparing the performance of modified TFRC and classical TFRC in terms of throughput, modified TFRC performs 20% more throughput than classical TFRC. Fig 4 illustrates that comparison of packet loss ratio with TFRC modified and classical. With creasing data rate, TFRC with enhance queue management gives low packet loss ratio than classical TFRC. When comparing the performance of modified TFRC and classical TFRC in terms of packet loss ratio, modified TFRC performs 18% less packet loss ratio than classical TFRC. Fig 5 shows result of packet delivery ratio under the comparison of classical TFRC and modified TFRC. Here easily identifies that our proposed TFRC performs outstanding as comparing with existing TFRC by 20%. Fig 6 shows comparison of modified TFRC and classical TFRC in terms of End-End delay. Modified TFRC performs less average end- end delay over classical TFRC by 16%. Further the proposed scheme gives good throughput when the traffic is unpredictable. Because the enhanced queue length predicts well about network traffic and it can adapt to the network fluctuations.

7. Conclusion

This paper presents an enhanced queue management algorithm and is applied to TFRC to alleviate the problems of congestion control and avoidance. This enhanced scheme helps to improve the performance of average queue length in RED algorithm and simulation results have shown that the modified TFRC produces increased throughput with less packet loss rate than existing. In future work we have planned to measure and increase throughput of TFRC in adversities like, measuring round trip time during dynamically changing bandwidth.

References

[1] M. Allman, V. Paxson, W. Stevens, "TCP Congestion Control", RFC 2581, April 1999.

[2] A. Mankin, A. Romanow, S. Bradner, V. Paxson, "IETF Criteria for Evaluating Reliable Multicast Transport and Application Protocols", RFC 2357, June 1998.

[3] K. Ramakrishnan, S. Floyd, "A Proposal to add Explicit Congestion Notification (ECN) to IP", RFC 2481, January 1999

[4] J. Postel, "Transmission Control Protocol (TCP): Protocol Specification", IETF RFC 793, September – 1981

[5] J. Postel, "User Datagram Protocol", RFC 768, August – 1980

[6] J. P. S. Floyd, et all, Protocol Specification: "TCP Friendly Rate Control (TFRC)", IETF RFC 5348, September – 2008

[7] P.Sreenivasa Rao, M. Janani and P. Chenna Reddy "TFRC for congestion control in wired Environment" *International Journal of Inventive Engineering and Sciences* ISSN:2319-9598, Volume-2, Issue- 1, pp. 13-15 December 2013.

[8] S. Lee and K. Chung, "Enhanced TFRC to improve the quality of multimedia streaming service," in *Proc. ICT Convergence*, pp. 373–378, 2012.

[9] B.H Oh, J. Han, K. Kim and J. Lee, "A New Receiver-Based Retransmission Scheme with TFRC ",*IEEE communications letters*,Vol 16, No 12, pp. 2091-2094, December 2012.

[10] S. Lee, H. Roh, H. Lee and K. Chung, "Enhanced TFRC for high quality video streaming over high bandwidth delay product networks," *Journal of Communications and Networks*, vol.16, no.3, pp.344-354,June 2014

[11] Z. Song, Y. Zhang, M. Zhou, "enhanced tfrc congestion control mechanism based on refined network states division", in *IEEE Proceedings of ICSIP*, pp. 938-942, Sep 2015.

[12] V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM Computer Communations. Rev.*, vol. 18, no. 4, pp. 314–329, Aug. 1998

[13] S. Floyd, V. Jacobson, "Random Early Detection Gateways for congestion avoidance" *–IEEE/ACM-Transactions on networking*, Vol.1, No. 4, pp.397-413, August-1993

[14] W. Feng, D. D. Kandlur, D. Saha and K. G. Shin, "A Self-Configuring RED Gateway", *Proceedings of IEEE INFOCOM* , pp. 1320-1328, 1999

[15] S. Floyd, G. Ramakrishna, and S. Shenker, "Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management", *Technical Report*, pp.1-12, August 2001

[16] A.E.Kamal, Md. Manzoor Murshed, "Adaptive RED with Dynamic Threshold Adjustment"- *Research report*, pp. 1-42, 2005

[17] A. K. Patel, Jyothi Divecha, "Modified exponential weighted moving average(EWMA) control chart for analytical process data", *JCEMS*- Vol 2(1) pp: 12-20, January 2011.

[18] A. Sathiaseelan, G. Fairhurst, "TCP-Friendly Rate Control (TFRC) for bursty media flows", *Computer Communications*, Volume 34, Issue 15, , pp. 1836-1847, September 2011

[19] M. Handley, et all, Protocol Specification: TCP Friendly Rate Control (TFRC), IETF RFC 3448, January – 2003

[20] S. Floyd, E. Kohler, "TFRC-SP variant", IETF RFC 4828,April 2007.

[21] P.R. Vasallo,-ICSI- "VariablePacket Size Equation-Based Congestion Control", *ICSI technical report*, pp. 1-11, 2000.

[22] M. A. Talaat, et all, "Enhanced TCP-friendly rate control for supporting video traffic over internet", *CJECE*, Volume 36, Issue 3, pp. 135-140, 2013,

[23] A. Chodorek and Robert R. Chodorek, "Streaming Video over TFRC with Linear Throughput Equation", *Advances in Electronics and Telecommunications*, Volume 1 Issue 2, pp. 26-29, November 2010

[24] A. Sterca; H. Hellwagner; F. Boian; A. Vancea, "Media-Friendly and TCP-Friendly Rate Control Protocols for Multimedia Streaming ". *IEEE Transactions on Circuits and Systems for Video Technology*, Volume: 26, Issue: 8 pp. 1516-1531, August 2016

[25] P. Chenna Reddy; M. Padmavathamma; N. Ramanjaneya Reddy, " Performance Comparison of TCP,UDP and TFRC in Wired Networks", *IEEE International Conference on Computational Intelligence and Communication Technology(CICT)*, pp. 257-263,2015