

CONGESTION CONTROL IN REAL TIME APPLICATIONS OVER WIRELESS NETWORKS

S.Vishnu Vardhan*

K.V. Bhaskar Rao*

Dr. P. Chenna Reddy*

Abstract:

The amount of traffic generated by Real Time Applications has increased substantially over the years. RTA will face congestion where there is any form of bottleneck restricting traffic. This will result in packet loss or delayed traffic which is unacceptable for RTAs. Therefore it is desirable for RTAs to implement congestion control mechanism to improve the stability of networks.

TCP Friendly Rate Control (RFC 3448) is a congestion control algorithm that provides a smooth transmission rate for continuous flow of data is available at sender. So it does not suitable for RTAs where data rate changes abruptly. RFC 5348 works based on variable bit rate for RTAs. TFRC RFC5348 has been proven to be fair when competing with TCP flows over congested links, but it lacks quality-of-service parameters to improve the performance of real-time traffic. In this work we use NS2, the network simulator for simulation of TFRC. TFRC is simulated in wireless environment, limitations are identified and modifications are proposed.

Keywords—congestion;RTAs; TFRC; TCP; video conference

* Computer Science and Engineering, JNTUA College of Engineering, Pulivendula.

1. INTRODUCTION

The Transmission Control Protocol (TCP) is connection oriented protocol that provides reliable and ordered delivery of packet and also provides end-to-end congestion control mechanism. Data transfer applications such as HTTP, FTP and SMTP are based on TCP. But over Internet, the use of RTAs such as VoIP, video conferences, instant messaging is constantly growing. Estimates show that these streaming media accounted for 30% of overall internet traffic.

The user datagram protocol (UDP) is one of the protocols of the Internet protocol suite. Using UDP, programs on networked computers can send datagram's to one another. UDP applications can send data at constant bit rate where it does not guarantee reliability or ordering in the way that TCP does. It is one of the non TCP based protocol. This non TCP flow cannot adjust their flow rate when congestion is detected where it continue to send at original rate. So these non TCP applications do not have congestion control mechanism and do not share bandwidth fairly with TCP based applications.

A new congestion control protocol for datagram transport was defined i.e., TFRC standardized by Internet Engineering Task Force (IETF). It provides a smooth transmission rate for real-time applications. It is reasonably fair when competing for bandwidth with TCP flows. In RFC 3448 a TFRC algorithm is defined based on model of TCP and it is designed for continues flow traffic. This is not suitable for the applications whose transmission is variable bit rate. Datagram congestion control protocol (DCCP) is recently standardized protocol. DCCP supports multiple congestion control algorithms and these will be selected through its Congestion Control ID (CCID). Three CCIDs are now standardized by IETF. CCID2 is a window based congestion control algorithm similar to TCP, CCID3 is a TCP-Friendly Rate Control algorithm, and CCID4 is a TCP-Friendly Rate Control for Small Packets (TFRC-SP).

In this paper, we present results of experimental evaluation of the performance DCCP (CCID3) TCP-Friendly rate control over wireless environments.

2. HOW TFRC WORKS

TFRC functionalities are located at the receiver's side. TFRC adjusts its sending rate based on the complex TCP throughput equation shown below:

$$T(\text{Bps}) = \frac{s}{RTT \sqrt{\frac{2p}{8}} + RTO \left(3 \sqrt{\frac{3p}{8}} \right) p(1+32p^2)} =$$

Where:

- s is segment size in bytes,
- RTT is round trip time in seconds,
- RTO is the TCP retransmission timeout value in seconds,
- P is the loss event rate.

To calculate loss event rate, receiver needs to find loss event of one or more packets lost or marked in particular RTT. Timestamp along with RTT is used by receiver to determine losses belong to same loss event or not. RTT is used to determine when to send feedback packets. Loss event rate and RTT is then fed to TCP throughput equation at senders end to calculate the TCP friendly rate. Sender then adjusts its sending rate according to this calculated rate. TFRC provides smooth sending rate while still providing sufficient responsiveness to competing traffic

A. Improvements of TFRC

The original specification of TFRC-RFC 3448 is suited for many multimedia streaming applications when continuous flow of data is available at sender. It is not useful for the applications like voice over IP (VoIP) and video conference characterized by periods of higher (but limited) transmission rate, separated by periods in which much less (which we call 'data-limited'), or without transmission of data (which we call 'idle periods'). This type of applications required variable media rate at senders.

TFRC(RFC 3448) starts transmission with slow start maximum of 1 packet per RTT and terminated with the first loss event. It adjusts its sending rate according to the calculated throughput equation by receiver. During slow start for the first few RTTs sending rate will be less than media rate and this delay is called start-up delay. During an idle period (no data can be transferred) for a period of 4RTTs, TFRC reduces the allowed sending rate by one half for every no feedback timer expiry (4RTT). The rate can be reduced to minimum of 2 packets for each

RTT. When the application restarts (starts sending data) TFRC must slow start back from idle period rate to the given rate. It takes time to recover from idle period to maintain media rate.

DCCP provides features of unreliable flow of datagram's with acknowledgements, reliable handshake for connection setup and teardown. DCCP working group also updated the algorithms in RFC 3448, leading to RFC 5348. It is intended to provide better support to streaming media applications like voice over IP (VoIP) and video conference during data limited and idle periods. RFC 5348 can share bandwidth fairly with the TCP based applications.

In TFRC (RFC 5348) during slow-start phase, the initial rate of the sender was increased to 4 packets per RTT. The behavior after an idle period was updated in the absence of loss, the sending rate is not reduced below 4 packets per RTT or equal to the initial rate. After idle and data-limited period double sending rate is not limited by receiver rate.

B. QOS requirements of VoIP and Video conference

Voice over IP and video conference both are two-way interactive applications that function within time frame that the user senses as immediate (or) current.

Three quality factors are required for both VoIP and video conference.

- Loss should be no more than 1 percent.
- One-way latency should be no more than 150ms.
- Jitter should be no more than 30ms.

3. METHODOLOGY

To test and compare the performance of DCCP (TFRC) protocol, the network simulator NS-2, version 2.34 is used. The network model used in simulation is composed by mobile nodes and links that are considered wireless. Each node considered as communication end-point is host and a forwarding unit is router.

In addition to NS-2, a set of tools, mainly Bash scripts and AWK filters, to post-process the output trace files generated by the simulator are developed. In order to evaluate the performance, multiple experiments have been set up.

A. Metrics

Packet Delivery Ratio: This is the ratio of total no of packets successfully received by the destination nodes to number of packets sent by the source nodes throughout the simulation.

End-to-End Packet Delay: It is the cumulative statistical measure of the delays experienced by packets traveling between source and destination.

4. SIMULATION RESULTS:

A. Fairness of TFRC with TCP

In this section TFRC (RFC5348) is compared with TCP by operating both at the same time. Packet size of TFRC and TCP is same i.e., 1000 bytes. Here 6 connections for TFRC and 1 connection for TCP in congested network.

Table 1. Comparison of TFRC with TCP

| TFRC rate | TFRC | | PDR | TCP | | PDR |
|-----------|------|----------|-------|------|----------|-------|
| | Sent | Received | | Sent | Received | |
| 128 Kbps | 8307 | 7956 | 0.957 | 1723 | 1709 | 0.991 |
| 256Kbps | 8615 | 7984 | 0.926 | 1566 | 1515 | 0.967 |
| 512Kbps | 9124 | 8084 | 0.886 | 1358 | 1325 | 0.975 |
| 768Kbps | 9478 | 8077 | 0.851 | 1391 | 1354 | 0.973 |
| 1Mb | 9613 | 8071 | 0.839 | 1389 | 1351 | 0.972 |

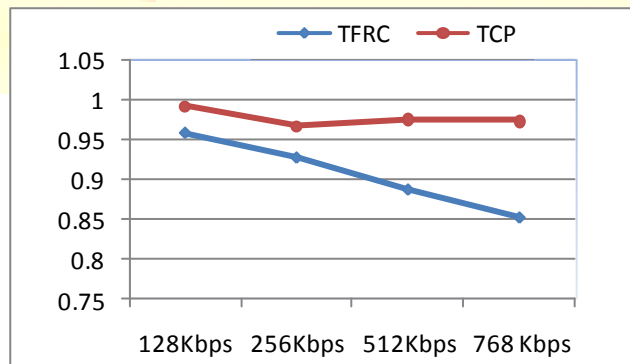


Figure1.1:Comparison of Packet delivery ratio of TFRC and TCP

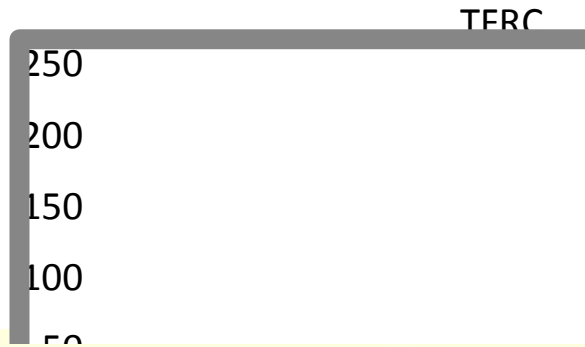


Figure1.2:End-to-End Packet Delay of TFRC and TCP

The results show that TFRC shares bandwidth fairly with the TCP. The packet delivery ratio metric is plotted in Fig. 1.1 On X-axis (TFRC rate) are represented. Table 1.1 presents the more detailed results. End-to-End delay results are shown in Fig. 1.2. The interesting observation to be made is TFRC also reduces the data rate in reaction to congestion and hence is fair to TCP.

B. Performance of TFRC

The following results show the Packet Delivery Ratio and End-to-End Packet Delay of TFRC and UDP at constant bit rates of 1Mb, 5Mb and 10Mb of continuous flow in wireless environment with packet size 1000.

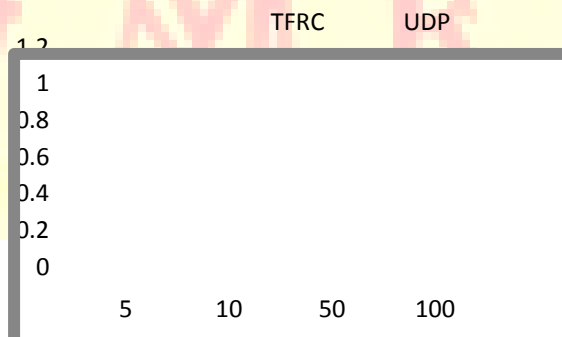


Figure2.1: Packet Delivery Ratio at 1Mb data

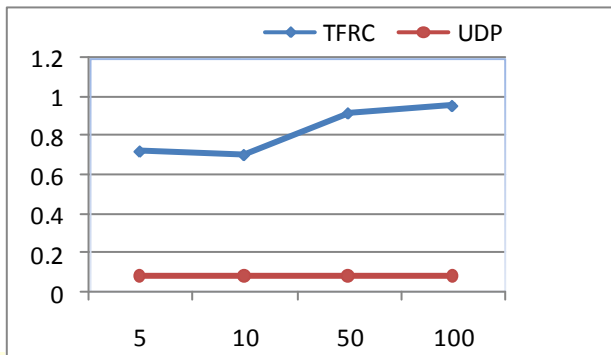


Figure1.2: Packet Delivery Ratio at 5Mb data rate

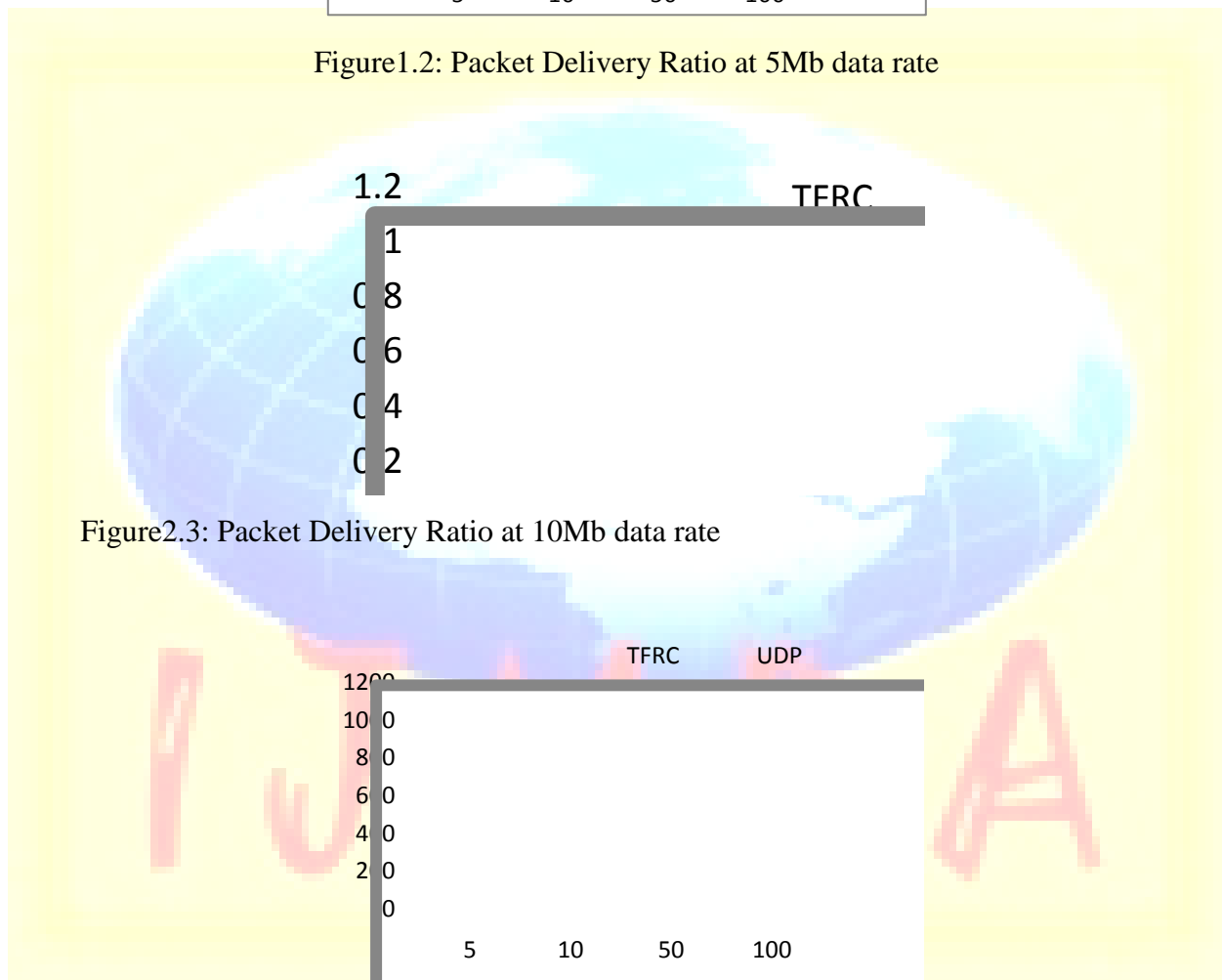


Figure2.3: Packet Delivery Ratio at 10Mb data rate

Figure2.4: End-to-End Packet Delay at 1Mb data rate

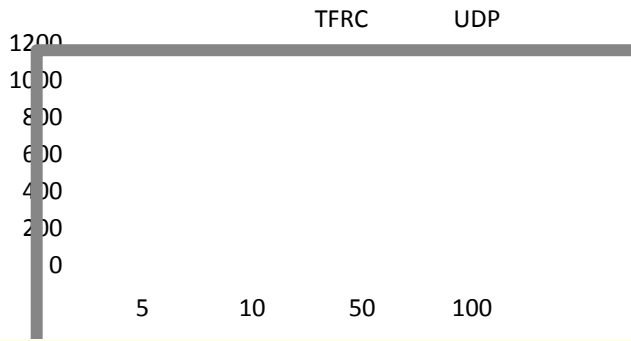


Figure2.2: End-to-End Packet Delay at 5Mb data rate

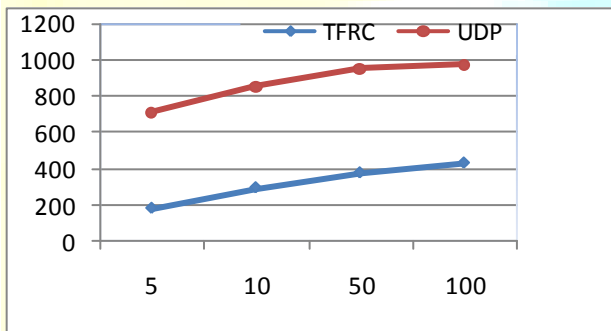


Figure2.6: End-to-End Packet Delay at 10Mb data rate

Table2.1: Packet Delivery Ratio of TFRC and UDP at rate 1Mb

| | Simulation time | Sent | Received | Packet delivery ratio |
|------|-----------------|-------|----------|-----------------------|
| TFRC | 5 ms | 630 | 452 | 0.717 |
| | 10 ms | 1148 | 940 | 0.818 |
| | 50 ms | 5078 | 4878 | 0.960 |
| | 100 ms | 9999 | 9752 | 0.975 |
| UDP | 5 ms | 1250 | 496 | 0.398 |
| | 10 ms | 2502 | 1004 | 0.401 |
| | 50 ms | 12500 | 5060 | 0.404 |
| | 100 ms | 25002 | 10129 | 0.4051 |

Table2.2: Packet Delivery Ratio of TFRC and UDP at rate 5Mb

| | Simulation time | Sent | Received | Packet delivery ratio |
|------|-----------------|--------|----------|-----------------------|
| TFRC | 5 ms | 635 | 459 | 0.722 |
| | 10 ms | 1328 | 935 | 0.704 |
| | 50 ms | 5272 | 4840 | 0.918 |
| | 100 ms | 10190 | 9745 | 0.956 |
| UDP | 5 ms | 6252 | 496 | 0.079 |
| | 10 ms | 12502 | 1004 | 0.0803 |
| | 50 ms | 62500 | 5060 | 0.0809 |
| | 100 ms | 125002 | 10129 | 0.0810 |

Table2.3: Packet Delivery Ratio of TFRC and UDP at rate 10Mb

| | Simulation time | Sent | Received | Packet delivery ratio |
|------|-----------------|--------|----------|-----------------------|
| TFRC | 5 ms | 635 | 459 | 0.722 |
| | 10 ms | 1328 | 935 | 0.704 |
| | 50 ms | 5272 | 4840 | 0.918 |
| | 100 ms | 10190 | 9743 | 0.956 |
| UDP | 5 ms | 12502 | 496 | 0.0396 |
| | 10 ms | 25002 | 1004 | 0.0401 |
| | 50 ms | 125002 | 5060 | 0.0404 |
| | 100 ms | 250002 | 10129 | 0.0405 |

B. Video performance

This section analyzes the performance of Video quality using variable bit rate. It is initially set to 1Mbps, at 20 seconds set rate is minimum rate of 100kbps, at 50 seconds set rate is 1.5Mbps and at 80 seconds set rate is 512Kbps. A packet size of 1000 bytes is taken.

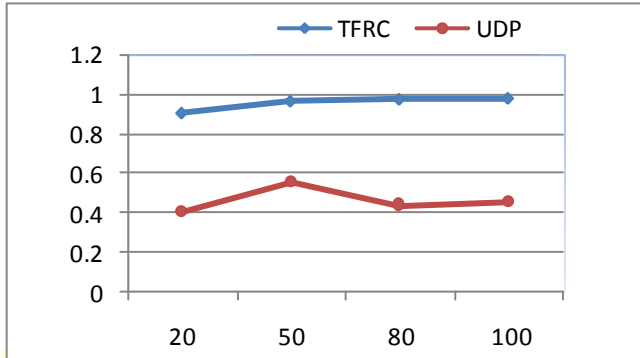


Figure3.1: Packet Delivery Ratio of TFRC and UDP

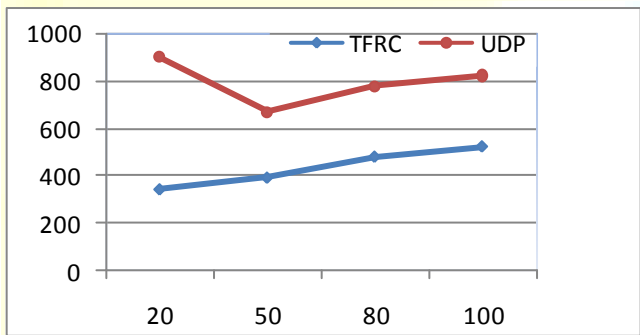


Figure3.2: End-to-End Packet Delay of TFRC and UDP

The following tables show the packet delivery ratio of TFRC and UDP with two connections in more detailed results.

Table3.1: Packet Delivery Ratio of TFRC and UDP at variable bit rate

| | Simulation time | Sent | Received | Packet delivery ratio |
|------|-----------------|-------|----------|-----------------------|
| TFRC | 20 ms | 2124 | 1916 | 0.902 |
| | 50 ms | 5040 | 4837 | 0.959 |
| | 80 ms | 8026 | 7772 | 0.968 |
| | 100 ms | 9997 | 9734 | 0.973 |
| UDP | 20 ms | 4991 | 2020 | 0.404 |
| | 50 ms | 9132 | 5041 | 0.552 |
| | 80 ms | 18496 | 8082 | 0.436 |
| | 100 ms | 22276 | 101109 | 0.453 |

5. CONCLUSION

TFRC operated along with TCP can share band width fairly. The performance results show that smooth transmission of sending rate of TFRC(5348) compare with UDP, where it continue to send at original rate. The specification in RFC 3448 poorly supported for Real Time applications where padding can be used to guarantee the required media rate for RTA applications. Where RFC5348 does not require padding, which consumes unnecessary network capacity. RFC 5348 also increases the sending rate compared to RFC 3448. We therefore expect this new standard to further encourage the use of a standards-based congestion control for Real Time Applications.

REFERENCES

- [1] M. Handley, S. Floyd, J. Padhye, J. Widmer, TCP Friendly Rate Control (TFRC): Protocol Specification, IETF RFC 3448, January 2003.
- [2] M. Handley, S. Floyd, J. Padhye, J. Widmer, TCP-Friendly Rate Control (TFRC): Protocol Specification, IETF RFC 5348, April 2008.
- [3] QoS requirements of VoIP and video conference by Cisco available <<http://www.ciscopress.com/articles>>
- [4] J. Postel, User Datagram Protocol, IETF RFC 768, August 1980.
- [5] M. Allman, V. Paxson, W. Stevens TCP Congestion Control, IETF RFC5681, September 2009.
- [6] S. Floyd, E. Kohler, J. Padhye, Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC), IETF RFC 4342, March 2006.
- [7] B. Zhou, C. Fu, V. Li, TFRC Veno: an Enhancement of TCP Friendly Rate Control over Wired/Wireless Networks, ICNP, October 2007, pp. 216–225.
- [8] S. Floyd, M. Handley, E. Kohler, Problem Statement for the Datagram Congestion Control Protocol (DCCP), IETF RFC 4336, March 2006.
- [9] M. Chen, A. Zakhor, Rate control for streaming video over wireless, IEEE INFOCOM, Hong Kong, March 2004.
- [10] S.Floyd; Congestion Control Principles; RFC 2914, BCP: 41; September 2000.

AUTHORS PROFILE



S. Vishnu Vardhan received the B.Tech degree in Computer Science and Engineering from JNT University, Anantapur in 2009. Currently he is pursuing M.Tech in Computer Science and Engineering at JNTUA College of Engineering, Pulivendula. His research interests include Congestion control, Transport layer protocols, wireless sensor networks.



Dr. P. Chenna Reddy did his B.Tech from S.V. University College of Engineering, Tirupati, M.Tech & Ph.D from JNTU, Hyderabad. He is currently working as Associate Professor at JNTUA College of Engineering, Pulivendula. He has 15 years of teaching experience . He has published several papers in reputed journals and conferences. His areas of interest are Computer Networks and related fields. He is currently working on Bio-inspired networking.