Performance Comparison and Analysis of TCP-Friendly Rate Control (TFRC) over the Network

by

Mohamad Izril bin Ishak
(0930210338)

A thesis submitted in partial fulfilment of the requirements for the degree of
Master of Science (Computer Engineering)

School of Computer and Communication Engineering
UNIVERSITI MALAYSIA PERLIS

2013

ACKNOWLEDGEMENT

First and foremost, I want to express my gratitude and thankful to Allah SWT for giving his special bless to me to be a Master student and finish up my thesis. Then, I would like to acknowledge my beloved parents, my wife and baby, my family and my entire friends for all the guidance, support, encouragement and pray. Without all of these peoples, I am not being able to finish my study.

My special thanks to the best supervisor, Dr. Ong Bi Lynn for all the advice, encouragement and supervision throughout these 4 years. Without her supervision, I wouldn't have the chance to be in network field and gain new experience and knowledge. Not forgotten Mr. Mohd Alif Hasmani as my co-supervisor for the advices and guidance.

I would also like to express my gratitude to the School of Computer and Communication Engineering for their concern on the student's project and providing the facilities and guidance to carry out the project.

My appreciation also goes to all those who have shared their knowledge and opinion with me. To all the individuals who had helped me during this project, I thank you all. Thanks to all of you, this project had given me a lot of knowledge and experiences that is very useful to me.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACM	Association for Computing Machinery
CRC	Cyclic redundancy codes
DARPA	Defense Advanced Research Projects Agency
IEEE	Institute of Electrical and Electronics Engineer
IP	Internet Protocol
ISN	Initial sequence number
MSS	Maximum segment size
NS2	Network Simulator 2
PPP	Point-to-Point Protocol
RTT	Round trip time
SIMD	Square Increase/Multiplicative Decrease
Tcl	Tool command language
TCP	Transport Control Protocol
TFRC	TCP-Friendly Rate Control
UDP	User Datagram Protocol
VoIP	Voice over IP

Perbandingan Prestasi dan Analisis *TCP-Friendly Rate Control* (TFRC) Melalui Rangkaian

ABSTRAK

Projek ini mencadangkan satu Perbandingan Prestasi dan Analisis *TCP-Friendly Rate Control* (TFRC) Melalui Rangkaian. Secara umumnya, TFRC digunakan untuk membawa data masa nyata. Secara tradisinya, penghantaran data masa nyata dicapai dengan menggunakan *User Datagram Protocol* (UDP). Isu-isu yang mencabar untuk menggunakan *Transport Control Protocol* (TCP) bagi aplikasi masa nyata adalah keterhadan *bandwidth* dan ketinggalan (*lagging*). Selain itu, telah diketahui secara meluas bahawa TCP kurang sesuai untuk aplikasi penstriman multimedia masa nyata. Setelah dikenal pasti isu-isu ini, projek ini dijalankan bertujuan untuk mengkaji dan menganalisis kemungkinan untuk meningkatkan prestasi aplikasi data masa nyata menggunakan TFRC yang direka untuk penstriman jangka panjang. Seperti diketahui, terdapat banyak kelebihan TFRC seperti kaedah kawalan kesesakan dan jaminan kadar celus yang diperlukan. Oleh itu, projek ini akan meninjau dan mencadangkan kelebihan TFRC dalam menghantar dan menerima data masa nyata. Adalah dicadangkan bahawa penyiasatan TFRC untuk penghantaran data masa nyata dijalankan menggunakan teknik simulasi. Dalam simulasi, model yang dibangunkan mewakili aplikasi masa nyata untuk menganalisa prestasi TFRC yang telah diubahsuai. Mekanisme TFRC yang diubahsuai diperkenalkan sebagai skim cadangan TFRC dan dinilai melalui eksperimen simulasi. Kajian simulasi projek ini adalah untuk menganalisis dan menilai beberapa parameter mekanisme TFRC yang sedia ada. Analisa prestasi dan eksperimen simulasi menunjukkan bahawa skim cadangan menunjukkan prestasi yang lebih baik berbanding dengan mekanisme lain yang terlibat. Keputusan menggunakan skim yang dicadangkan mengatasi mekanisme lain sehingga 5% untuk kecekapan pemprosesan dan sehingga 20% ke atas kehilangan paket. Akhir sekali, kajian ini mempunyai kemungkinan kerja yang berterusan untuk dijalankan dalam persekitaran tanpa wayar dan IPv6.

Performance Comparison and Analysis of TCP-Friendly Rate Control (TFRC) over the Network

ABSTRACT

This project proposes a Performance Comparison and Analysis of TCP-Friendly Rate Control (TFRC) over the network. Generally, TFRC is used for carrying real-time data. Traditionally, real-time data transmission has been accomplished over the user datagram protocol (UDP). The challenging issues of using Transport Control Protocol (TCP) are restrictive bandwidth and lagging. Additionally, it is widely known that TCP is ill-suited to real-time multimedia streaming application. Having known these issues, this project aims to study the possibilities for analysis and improvement of real-time data application using TFRC which is designed for long-lived streams. As known, there are many advantages of TFRC, which are congestion control and guaranteeing the required throughput. Therefore, this project explores and proposes the advantages of TFRC in transmitting and receiving real-time data. It is proposed that the investigation of TFRC in real-time data is conducted using simulation technique. During the simulation, the developed model is representing the real-time application for analyzing the performance of modified TFRC. The modified TFRC mechanism is introduced as TFRC proposed scheme and evaluated through the simulation experiment. The simulation study of this project is to analyze and evaluate some parameters of existing TFRC mechanisms. The performance analysis and simulation experiment show that the proposed scheme performs better compared to the other mechanisms involved. The results of applying the proposed scheme outperform the other mechanisms up to 5% for throughput efficiency and up to 20% on packet loss. Finally, this study has possibilities of the ongoing work to be carried for implementing in wireless environment and IPv6.

CHAPTER 1

INTRODUCTION

1.1 Overview

Ever since man developed his communication skills, he has embarked on a journey of technological developments. These communication skills have been developed to such an extent that the information pass without much delay and lagging. Yahoo Messenger and Skype are such a communication applications that provide real-time medium for human to communicate. With the rapid increase of multimedia traffic and the deployment of real-time audio/video streaming applications, the current Internet has seen an exponential increase in the percentage of non-TCP traffic. These multimedia streaming applications do not share the available bandwidth fairly with applications built on TCP. This evolution can lead to a congestion collapse and starvation for TCP traffic.

The capabilities of TCP to transmit real-time packet cannot be refuted. TCP has congestion control and timer management mechanisms. In order to be fair to TCP flows in best-efforts IP networks, the congestion control algorithms must be TCP-friendly which is TFRC. This project takes the opportunity to manipulate these capabilities to give benefits for real-time communication impressively. This is because TFRC provides congestion control mechanism and media flows with smooth transport.

Previous researches [1], [2], [3], and [4] have shown that TCP can perform better using rate-based TCP-friendly Congestion Control (TFRC) for real-time data transmission. This protocol is developed to support the deployment of multimedia applications in the Internet. The purpose of this protocol is to ensure that the application's traffic shares the network in the fairly and friendly manners with the dominant TCP traffic.

Therefore, in this project some exploration of techniques to enhance TFRC is made to be used for real-time streaming data and protocol such as voice communication impressively. Using these techniques, there is an aim to ensure that TFRC is suitable for real-time data applications. With the best of knowledge and through the studies of previous researches, there is higher confident level that this project can achieve the goals.

1.2 Problem Statement

In order to derive an acceptable TCP-friendly transmission, the TFRC sender adjusts its transmission rate based on the measured loss rate and RTT. Using TCP Reno throughput model, a control equation has been derived for the use of the adjustment of sending rate to achieve TCP-friendliness. The slow start technique is used at the beginning of the transmission phase. During the transmission, TFRC tries to increase its sending rate by multiplicatively at every RTT until it detects a loss. Packet losses are identified by gaps in the sequence number of the transmitted packet at the receiver module. The more the packet loss, the less the quality of service offered [2].

The loss of packet affects the throughput efficiency of data transmission. The lower the throughput efficiency, the lower the quality of data received [5]. Throughput efficiency depends on the transmission rate of the links over which the data flows. In particular, a link with a high transmission rate may nonetheless be the bottleneck link if many other data flows are also passing information through that link. Therefore, an analysis and study for increasing throughput efficiency are developed.

Moreover, throughout research observations, most of the researchers focus on new algorithm development to improve TFRC. This matter may be referred to papers, articles and journals published. Hence, combining current enhanced mechanisms may reduce the cost of study and may produce better result of improvement.

1.3 Research Objectives

This research has an aim to study the performance of TFRC mechanisms over the network. The results of the study give further aims which are to reduce the packet loss during the transmission of real-time data. This aim raises three research objectives:

a) To propose a new mechanism for TFRC protocol using combination of existing enhanced mechanisms of TFRC. The development is carried out by studying the algorithm and mechanism used in each of the enhanced protocol.

b) To enhance the performance of TFRC for transmit real-time data application by reducing the packet loss and increasing throughput efficiency. The improvement is carried out by modifying the original TFRC sending rate algorithm.

c) To analyze the proposed protocol by evaluating the selected parameters of each mechanism involved using simulation experiment technique.

1.4 Significance of the Research

Major contributions of this research include:

a) The TFRC Proposed Scheme is designed by combining TFRC SIMD and TFRC Veno. The combination of the famous enhanced mechanisms shows better performance than the original mechanisms. This approach may be used by future researchers as a guideline for improving TFRC in different environment.

b) An enhance equation to improve the performance of TFRC is developed. The development for ensuring smooth sending rate is done by substituting the value of α in Veno with the value from SIMD while the value of β is taken from the experimental analysis.

c) The result of analysis has shown that the proposed scheme reduces the packet loss. Furthermore, both the performance analysis and simulation experiment agree that proposed scheme mechanism has higher throughput efficiency. Thus, the proposed scheme is proposed to be implemented in current TFRC. Hence, the performance of real-time data application is increased.

1.5 Research Scope

This project is specifically interested in studying ways to reduce the packet loss. The reason is because lower packet loss causes higher throughput efficiency. Higher throughput efficiency ensures better Internet service quality.

The mechanisms selected for comparison is limited to TFRC SIMD, TFRC Veno and TFRC proposed scheme while the original TFRC becomes the analysis benchmark of experiment. The parameters examined in this research are limited to packet loss, delay and throughput efficiency. From these parameters, there is expansion of discussions on how much improvement or degradation form in the experiments.

Moreover, the evaluation of TFRC performance is running on wired environment and settings. Again, the simulation experiment never runs using the proposed scheme under wireless environment.

Having known the scope of research study, the overview of the thesis is presented in the next section. The thesis overview gives the general outlook of the content of this thesis.

1.6 Organization of Thesis

This project presents the performance evaluation and analysis of various TFRC mechanisms over the Internet. In particular, this project describes the improvement of existing mechanisms which aim to reduce the packet loss. This report consists of five chapters. The first chapter presents introduction to the project and the background of problem on why is the study is being conducted. It also gives the objectives and scope of the study. Chapter 2 reviews the background and related works on all major research issues covered in the thesis. Chapter 3 discusses on the framework of this project in details including equation enhancement and network simulation topology model. The modification and implementation of the proposed mechanism of protocol is presented. Chapter 4 contains the results of performance analysis of selected parameters and mechanisms. The results of the evaluation are compared in order to prove that the proposed mechanism reduces the packet loss. Chapter 5 concludes the thesis. The contributions and future directions of this research are presented later.

CHAPTER 2

LITERATURE REVIEW

The focus of this research is comparing the equation-based TCP-friendly over the network. This chapter presents an overview of the research background and the issues to be discussed throughout this thesis. Section 2.1 describes about TCP. Then in Section 2.2 the concept of TFRC is explained. Section 2.3 introduces the performance evaluation technique. The related works of the research areas is comprehensively discussed in Section 2.4. Finally, Section 2.5 summarizes the whole chapter.

2.1 Transmission Control Protocol (TCP)

TCP is one of the core protocols in TCP/IP networks. TCP/IP is made up of two acronyms, TCP and IP (Internet protocol). It is a connection oriented reliable protocol working at the transport layer. Specifically, TCP/IP handles network communication between network nodes (computers, or nodes, connected to the net).

2.1.1 Internet Protocol

The Internet Protocol is designed for use in interconnected systems of packet-switches computer communications networks. The IP provides for transmitting blocks of data called datagram from sources to destinations, where sources and destinations are hosts identified by fixed length addresses [6].

Fig. 2.1 shows the IP (version 4) datagram format which is used in current implementation of Internet connectivity. The header of IP datagram is 20 bytes by assuming no options. If the datagram carries a TCP segment, each non-fragmented datagram carries a total of 40 bytes of header (where 20 bytes of IP header plus 20 bytes of TCP header) along with the Layer 5 (TCP application-layer) message.

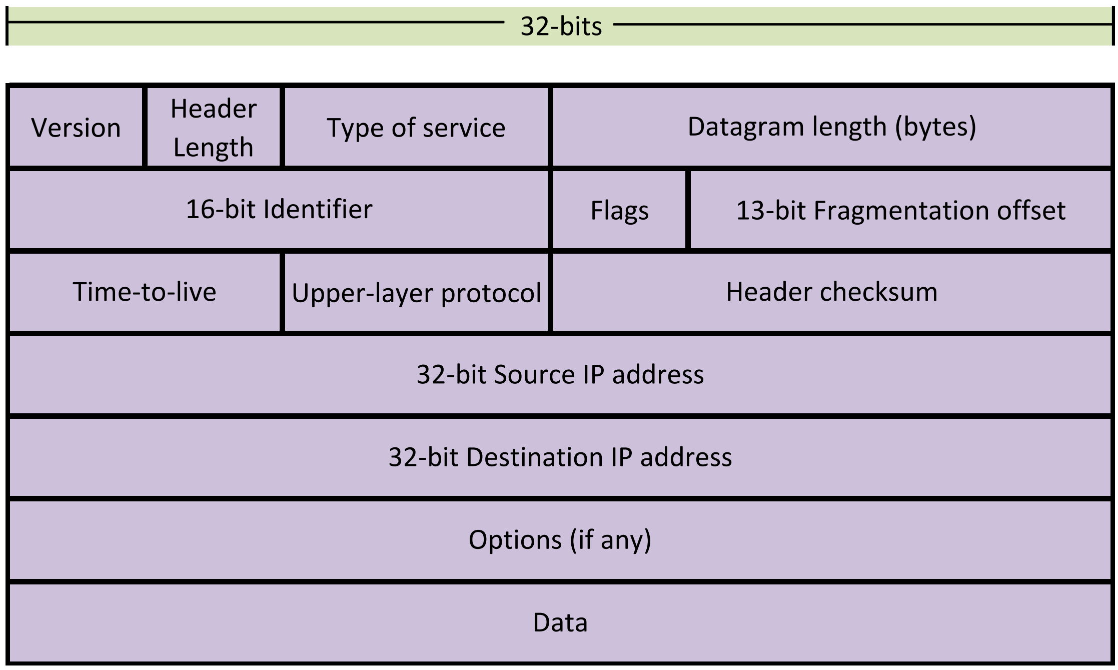

Figure 2.1: IPv4 datagram format

2.1.2 TCP Segment Structure

TCP segment consists of header fields and data field. The data field contains a segment of application data. Fig. 2.2 shows the structure of the TCP segment. The header includes source and destination port numbers. This port numbers are used for multiplexing/demultiplexing data from/to upper-layer applications.

TCP sender and receiver use sequence number field and acknowledgement number field in implementing a reliable data transfer service while the receive window is used for flow control.

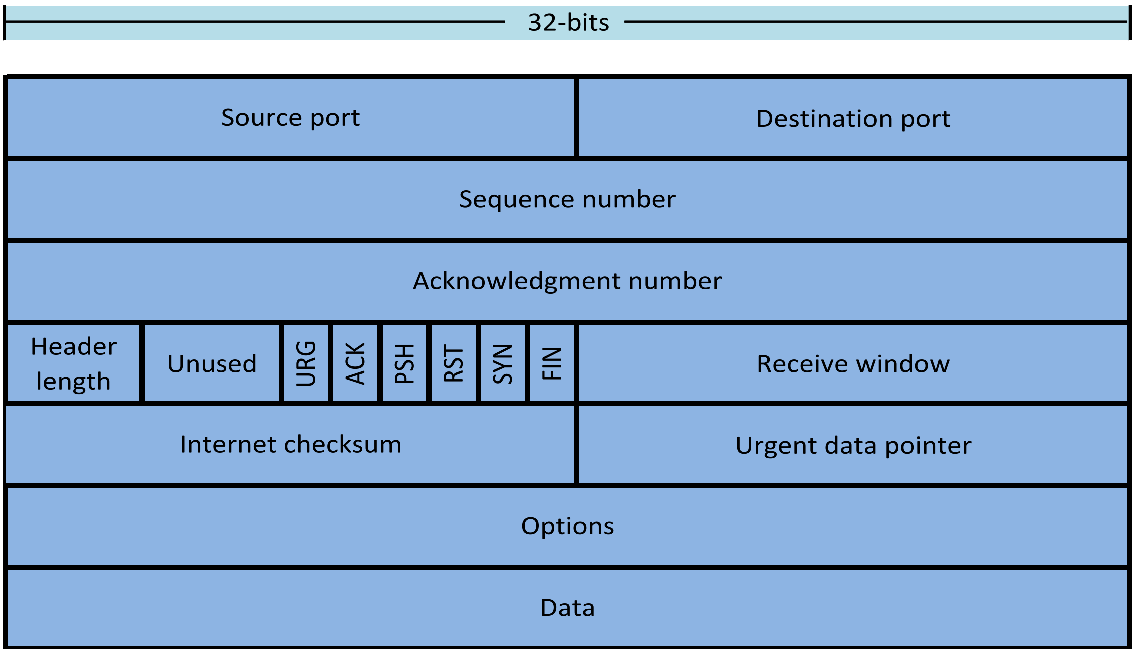

Figure 2.2: TCP segment structure

2.1.3 TCP Connection Establishment

For connection establishment, TCP uses a three-way handshake. This is called three-way handshake because three segments are sent between the two hosts which are client and server. The client has the job of initiating contact with the server. In order for the server to be able to react to the client's initial contact, the server has to be ready. With the server process running, the client process can initiate a TCP connection to the server as illustrated by Fig. 2.3.

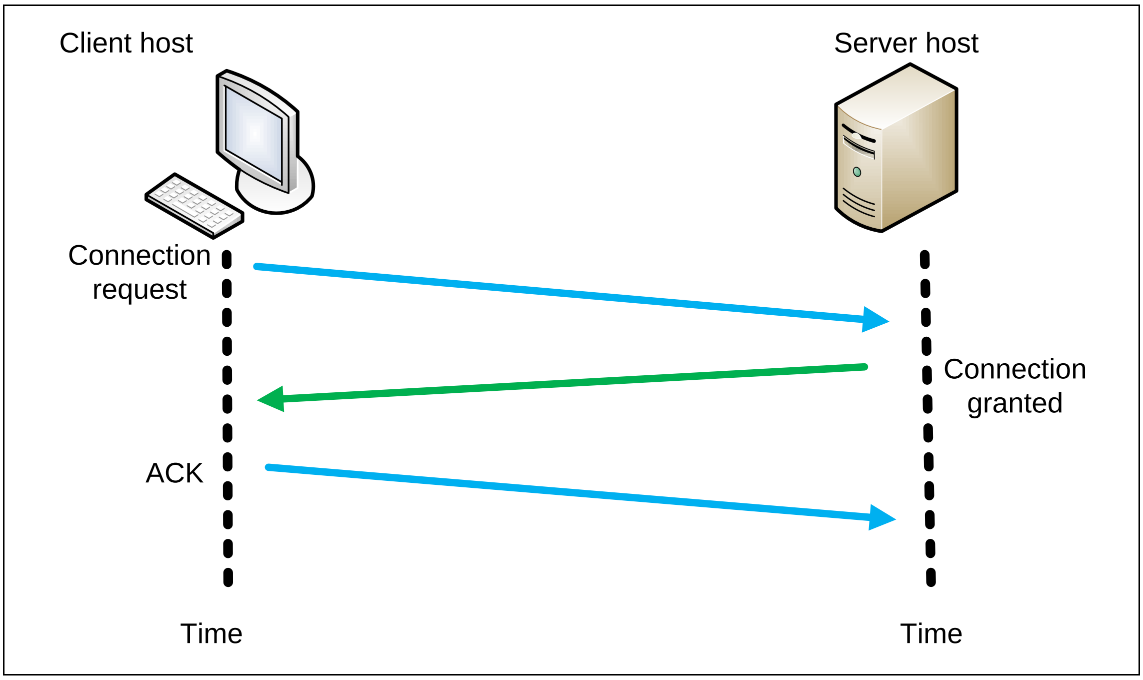

Figure 2.3: TCP connection establishment

A special TCP segment is sent by client-side TCP to the server-side TCP where the content has no application-layer data. One of the flag bits in the segment's header (see Fig. 2.2) which is SYN bit is set to 1. This segment is encapsulated within an IP datagram and sent to the server. Once this segment arriving at the server host, the server extracts the TCP-SYN segment, allocates the TCP buffers and variables to the connection and sends a connection-granted segment to the client TCP. Upon receiving the SYNACK segment, the client allocates buffers and variables to the connection and sends the last segment as acknowledgement to the server's connection-granted segment.

2.1.4 TCP Reliable Transmission

Generally, IP service is unreliable where IP neither guarantee datagram delivery nor guarantee in-order delivery of datagram as well as the integrity of the data in datagram. This situation can affects transport-layer segments which are carried by IP datagram across the network. As a prevention manner, TCP creates a reliable data transfer service on top of IP's unreliable best-effort service. This for ensures that the data stream is uncorrupted, without gaps, without duplication and in sequence. The byte stream received by the end system is exactly the same byte stream sent by the other side of connection.

In order to ensure reliable transmission, there are three major events related need to be focused. The first major event is data received from application above. Once TCP received data from the application, TCP encapsulates the data in a segment and passes the segment to the IP. Each segment includes a sequence number that is the byte-stream number of the first data byte in the segment. The second major event is timeout. TCP responds to the timeout event by retransmitting the segment that caused the timeout and then restarts the timer. The third major event that must be focused is an acknowledgment. TCP uses cumulative acknowledgements where the receiver acknowledges the receipt of all bytes.

Other things that must be handled by TCP for guaranteeing the reliable transmission are timeout interval, fast retransmit and retransmission missing segment method. All of these things are not discussed in this thesis report.

2.1.5 TCP Flow Control

Recall from previous section, TCP places the correct received bytes and in sequence in the receive buffer. The associated application process which needs to use this data accesses the buffer but not necessarily at the instant the data arrives. Bear in mind that the application may be busy to attempt to read the data due to some other task. The data still remains in the buffer even long after it has arrived. In this case, the connection's receive buffer can be easily overflowed by the sender due to sending too much data too quickly.

In order to eliminate the possibility of the sender overflowing the receiver's buffer, TCP provides a flow-control service to its applications. Flow control is a speed-matching service-matching the rate at which the sender is sending against the rate at which receiving application is reading [7].

To perform this flow control, the sender needs to maintain a variable called the receive window. This window is used to generally tell the sender the available buffer space at the receiver. Because TCP is full-duplex, the sender at each side of the connection maintains a distinct receive window. If the segment is fast enough from the buffer, the buffer will overflow and the segment will get dropped.

2.2 TCP-Friendly Control Rate (TFRC) Protocol

TFRC is a congestion control mechanism for unicast flows operating in a best-effort Internet environment. It is reasonably fair when competing for bandwidth with TCP flows, but has a much lower variation of throughput over time compared with TCP, making it more suitable for applications such as telephony or streaming media where a relatively smooth sending rate is of importance [8]. The word "fair" here means TFRC's long-term throughput has to be similar to TCP and no need to fit perfectly to the behaviour of TCP.

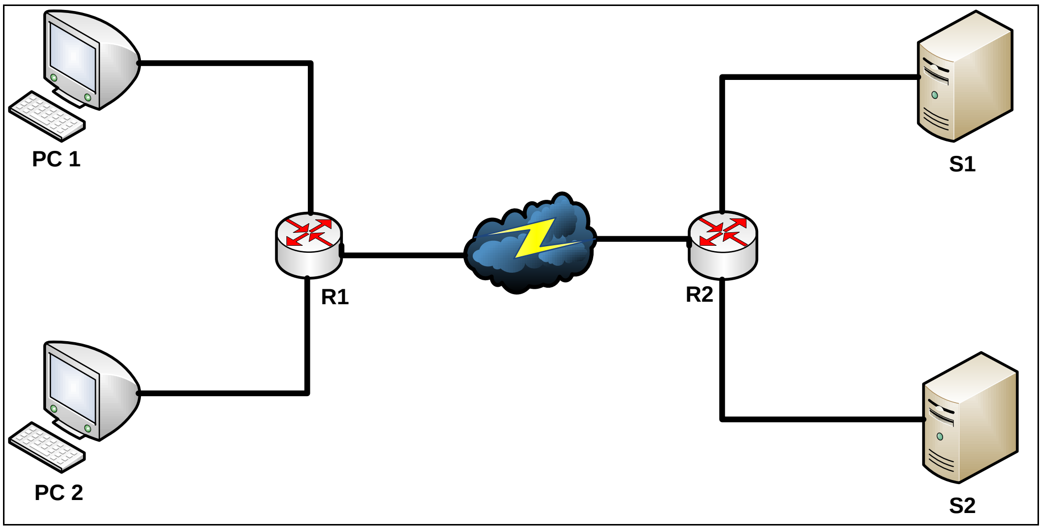

Figure 2.4: TCP implementation scenario

Referring to Fig. 2.4, the connection between R1 and R2 may not able to support incoming data packet from servers (S1 and S2). Thus the buffer at R2 starts to fill in the packets even it has to thrown away if the buffers overflow. This creates the situation called congestion. The information transmitted by the application such as multimedia streaming and voice over IP (VoIP) is critical of the times. These applications need smooth data rate to transmit their continuous packet stream.

Due to this stipulation, User Datagram Protocol (UDP) is used instead of TCP because UDP is well suited for applications that need fast data delivery. The major problem of UDP is its not interested on congestion-awareness mechanisms which may results by high delays and lower throughputs.

The solution to overcome this problem is using TFRC. TFRC is designed for applications that use a fixed packet size and vary their sending rate in packets per second in response to congestion [8]. For the congestion control mechanism, TFRC directly uses a throughput equation to allow sending rate as a function of the loss event rate and round-trip time.

2.2.1 Basic Mechanism of TFRC

A brief mechanism of TFRC can be illustrated in Fig. 2.5. Generally, TFRC has several mechanisms to ensure the competition of data flows is fair. Fundamentally, TFRC's congestion mechanism works as follows:

- The loss event rate is measured and this information is feed backed to the sender
- This feedback messages is also used by the sender to measure the RTT
- All the info then fed into TFRC's throughput equation for producing the acceptable transmitting rate
- The transmit rate is adjusted by the sender in order to match the rate which is calculated before

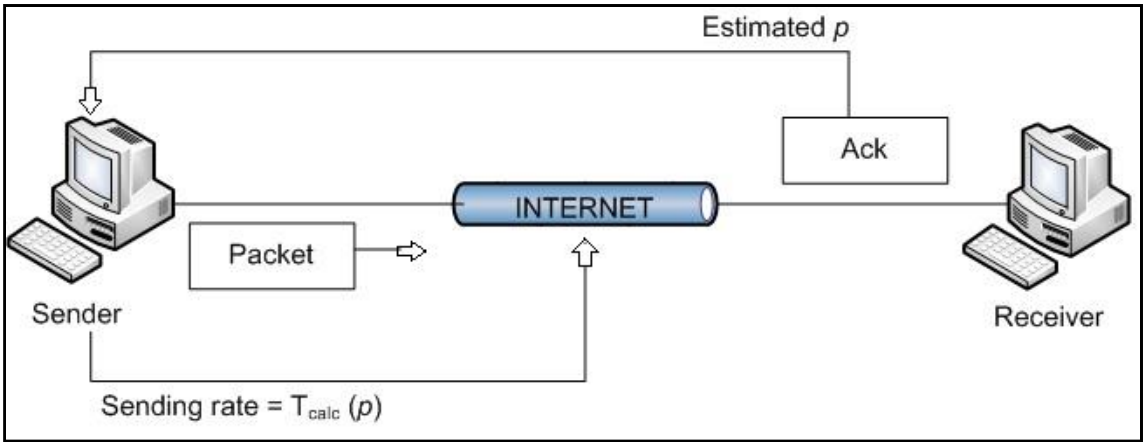

Figure 2.5: Basic mechanism of TFRC

Once the packet losses occur, the receiver estimates the packet loss rate p and acknowledges it to the sender. Then using TCP throughput model, a control equation has been derived for the use of the adjustment of sending rate to achieve TCP-friendliness. The control equation which based on TCP Reno equation is [8]: