

Chapter 21

Fuzzy Logic–Based Round Trip Time Scaling and Scheduling in High Speed TCP Stacks

V. Kavidha

Dr. Sivanthi Aditanar College of Engineering, India

V. Sadasivam

Manonmaniam Sundarnar University, India

ABSTRACT

Network management and control is a complex problem that requires intelligent, control methodologies to obtain satisfactory performance. Round trip time (RTT) scaling mechanism has been introduced for changing congestion window and to perform congestion control satisfactorily in all circumstances. This paper presents a fuzzy RTT scaling (FRTTS) scheme that performs RTT scaling and RTT scheduling for different high speed transmission control protocol (TCP) networks. In this scheme, RTT samples are allocated requesting application by using RTT scheduling factor and RTT samples are scaled for an application by using RTT scaling factor. A RTT scaler placed at the end node performs the RTT scheduling as well as RTT scaling. We also apply a FRTTS scheme on different high speed TCP's namely high-TCP (H-TCP) and scalable-TCP(S-TCP) and demonstrates that it provides better performance than non fuzzy scheme. The scheme has been extensively simulated to test the performance in terms of flow rate, RTT flows, packet size and congestion window size. The results show that FRTTS scheme provides better performance than non fuzzy scheme which employs dynamic RTT scheduling and RTT scaling.

INTRODUCTION

The basic design property limits the scalability of the existing TCP congestion control algorithm on bandwidths with a high bandwidth delay product (BDP). The linear rate of increase for the congestion window by one packet per RTT means, the

following back off factor of the congestion window can take a substantial period of time before the window recovers. Xu et al. (2004) propose the Linux operating system an experimental algorithm called BIC-TCP. Cai et al. (2009) have proposed a number of proposals to modify the TCP congestion control algorithm and to improve performance in

DOI: 10.4018/978-1-4666-7456-1.ch021

high bandwidth delay product paths. All of these are still experimental and pending evaluation as they change the congestion control in new and significant ways and their effects on the network are not well understood.

TCP is the standard Internet used by the majority of applications. Padhye et al. (2000) constructed two fundamental functions reliability and congestion control which motivated wide acceptance of TCP. TCP congestion control mechanism is based on a sliding window mechanism and employs an additive increase multiplicative decrease (AIMD) algorithm with the purpose of transmission to the available network resources. Based on acknowledgement (ACK) feedback from the receiver, the sender detects packet losses and consequently adjusts its sending rate. TCP treats all packet losses as if they were caused by congestion, and as a consequence drastic reduction of sending rate is the only reaction on the packet loss detection. At connection startup, the available path capacity is probed with an exponential window increase (by one for every ACK), giving rise to what is referred to as slow start phase. Leith et al. (2008) constructed a protocol which enters the congestion avoidance phase, where the congestion window value is increased approximately linearly by one for every RTT.

Fuzzy logic has been used in congestion control algorithms such as Fuzzy leaky bucket algorithm (FLBA) and fuzzy explicit marking (FEM). Each of which showed the superiority of fuzzy approach proposed by Fengyuan et al. (2002) versus the non-fuzzy one. Our main contribution in this work is to provide a FRTTS algorithm which is more practical to deploy and maintains higher utilization and fairness, in the face of flows with widely varying RTTs and in the presence of short lived connections, in contrast with the traditional fuzzy logic systems. H-TCP, S-TCP, BIC, and FAST have proposed by David et al. (2006) to address the under-utilization problem of TCP in high speed and long distance networks. These high speed TCP variants constructed by Sumitha et al. (2006)

modify the congestion avoidance algorithms of TCP to be more aggressive in high speed networks. They still use the same fast retransmit and fast recovery algorithms as TCP, which enables TCP to detect and recover from packet loss earlier than the timeout period. However, it is well known that the fast retransmit and fast recovery algorithms of TCP may misinterpret RTT scheduling as packet loss, and therefore cause TCP to perform poorly in networks with severe RTT scheduling. Li et al. (2007) used high speed TCP variants with the same fast retransmit and fast recovery algorithms and may not achieve the expected high throughput when RTT scheduling occurs.

A number of solutions proposed by Kelly (2003) are H-TCP, S-TCP and FAST TCP. Roughly these protocols can be divided into two classes: loss based and delay based. Loss based versions share similar features with traditional TCP whereas delay based TCP is an extension of TCP Vegas (Christiansen, 2001). There is a considerable research regarding the modeling and analysis of high speed TCP versions. It is widely accepted that one of the most important issues with these protocols is operability and employability. This directly leads to the question of fairness. FAST TCP a delay-based TCP version that is designed as a transport protocol for next generation networks, especially high bandwidth delay product networks.

We introduce FRTTS scheme which is designed with fuzzy rules to perform RTT scaling and RTT scheduling in high speed TCP networks. The scheme operates at the base layer of the network and the data in knowledge base is updated periodically depending upon the running applications parameters. RTT scheduling factor and RTT scaling factor are computed adaptively based on four fuzzy parameters, RTT flows, flow rate, packet size and congestion window size.

The rest of the paper is organized as follows. The fuzzy approach is composed of three steps fuzzification, inference with rule base and defuzzification is introduced. RTT scheduling and RTT scaling algorithm are introduced based on

12 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/fuzzy-logic-based-round-trip-time-scaling-and-scheduling-in-high-speed-tcp-stacks/124512

Related Content

The Contemporary Ethical and Privacy Issues of Smart Medical Fields

Victor Chang, Yujie Shiand Yan Zhang (2019). *International Journal of Strategic Engineering* (pp. 35-43).

www.irma-international.org/article/the-contemporary-ethical-and-privacy-issues-of-smart-medical-fields/230936

Indigenous knowledge and Globalization in Bangladesh: NGOs' Capacity for Social Capital and Community Development

M. Rezaul Islam (2019). *Social Research Methodology and New Techniques in Analysis, Interpretation, and Writing* (pp. 49-74).

www.irma-international.org/chapter/indigenous-knowledge-and-globalization-in-bangladesh/220331

Nanoparticles for the Prevention and Treatment of Bacterial Biofilms on Orthopedic Implants

Priyanka Prabhuand Vishvesh Joshi (2024). *Biomedical Research Developments for Improved Healthcare* (pp. 208-245).

www.irma-international.org/chapter/nanoparticles-for-the-prevention-and-treatment-of-bacterial-biofilms-on-orthopedic-implants/341070

Research Methodologies, Data Collection, and Analysis at MailChimp: A Case Study

Gregg Bernsteinand Laurissa Wolfram-Hvass (2015). *Contemporary Research Methods and Data Analytics in the News Industry* (pp. 170-189).

www.irma-international.org/chapter/research-methodologies-data-collection-and-analysis-at-mailchimp/132913

Summer Mentoring Experiences for Students in Online Doctoral Programs: The Llano Estacado Writers' Alliance

Julie Smit, Elizabeth Jones, Michael Ladickand Mellinee Lesley (2021). *Handbook of Research on Developing Students' Scholarly Dispositions in Higher Education* (pp. 109-131).

www.irma-international.org/chapter/summer-mentoring-experiences-for-students-in-online-doctoral-programs/285114