Computer Networks 64 (2014) 308–321

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Adaptive congestion control framework and a simple implementation on high bandwidth-delay product networks

Min Wang [a,b], Junfeng Wang [a,*], Sunyoung Han [c]

[a] College of Computer Science, Sichuan University, Chengdu 610064, PR China
[b] College of Computer Science and Information Technology, Yunnan Normal University, Kunming 650092, PR China
[c] Department of Computer Science and Engineering, Konkuk University, Seoul 143-701, Republic of Korea

ARTICLE INFO

Article history:
Received 3 August 2013
Received in revised form 20 February 2014
Accepted 23 February 2014
Available online 1 March 2014

Keywords:
Adaptive congestion control
Bandwidth-delay product
Delay-based approach
Loss-based approach
TCP performance

ABSTRACT

As new link technologies and sub-networks proliferate and evolve, a large number of TCP variants have been developed for different types of the network environments. They can lead to major performance gains by taking advantage of local characteristics of the specific network. However, these TCP variants could not be automatically chosen according to the lower network environments. In this paper, we propose the ACCF, an adaptive congestion control framework, which can automatically transition among existing congestion control mechanisms according to the change of the network status. Then we perform a simple implementation of ACCF over the networks with high bandwidth-delay product (BDP). It can switch the congestion control approaches between the delay-based ones and the loss-based ones according to the network status. Extensive experiments are conducted based on network simulators as well as over real wired networks on different time periods of the day. For the simulation measures, the experimental results show that the performance of ACCF is significantly improved as compared to other state-of-the-art algorithms in term of throughput, fairness and TCP-friendliness. For the real network tests, the experimental results show that ACCF achieves speedup ratios up to 225.83% compared with average throughput of other TCP congestion control algorithms.

1. Introduction

Almost all Internet applications rely on the Transmission Control Protocol (TCP) [1] to deliver data reliably across the network. Although it was not part of TCP's primary design, the most vital element of TCP is congestion control, which determines TCP's performance characteristics. This area has been continuing to draw research and engineering effort because new link technologies and sub-networks have proliferated and evolved. For example, the past few years have seen an increase in wireless networks with variable bottleneck rates; datacenter networks with high rates, short delays, and correlations in offered load; paths with excessive buffering (now called "bufferbloat"); cellular wireless networks with highly variable, self-inflicted packet delays; links with non-congestive stochastic loss; and networks with large bandwidth-delay product (BDP). In these conditions, standard TCP's behavior could be sub-optimal and even erroneous [2–5] mainly because it is expected to operate in a diverse set of networks with different characteristics, making standard TCP a "Jack of all trades, master of none" protocol. And standard TCP could not adapt its congestion control algorithms to new scenarios, which constrains architectural evolution, as noted in an earlier position paper [6]. In this work, we aim to perform the adaptive congestion control for different network status.

* Corresponding author. Tel./fax: +86 028 85412159.
E-mail addresses: honghewangmin@sina.com.cn (M. Wang), wangjf@scu.edu.cn (J. Wang), syhan@cclab.konkuk.ac.kr (S. Han).

http://dx.doi.org/10.1016/j.comnet.2014.02.021
1389-1286/

Limiting TCP to a specific network and taking advantage of local characteristics of that network can lead to major performance gains. So far, a large number of TCP variants have been developed for different types of network environments. For example, TCP New Reno [7], TCP SACK [8] and TCP Vegas [9] are for low speed networks; BIC TCP [10], Cubic TCP [11], H-TCP [12], Illinois TCP [13] and Compound TCP [14] are for the networks with high BDP; TCP Westwood [15] is for wireless networks; TCP Peach [16] is for the satellite networks; DCTCP [17] is for data center networks. These variants can outperform TCP in the specific networks, while they might not be applicable in other networks. There are different congestion-control-mechanisms used in the TCP variants which are suitable for the similar networks. For instance, there are loss-based approaches, delay-based approaches and the synergy of loss-based and delay-based approach for the networks with large BDP [13]. Different congestion control approaches could achieve different performance gains.

Some of TCP variants have been implemented in the Linux kernel or Windows System, e.g. BIC TCP, Cubic TCP, TCP Vegas, TCP Illinois and TCP Westwood are implemented in Linux kernel, and Compound TCP is implemented in Windows Vista. However, these TCP variants could not be automatically chosen according to the lower network state. Some default congestion control modules, such as TCP Reno, BIC and Cubic, are used in different Linux kernel versions. Other congestion control approaches could not be used unless users deploy them manually. It is impractical to choose an appropriate congestion control module for every endpoint. Additionally, the lower network state, such as the available bandwidth and buffer size of the routers, is time varying. All this would make the automatic choice a challenge.

In [18,19], novel AQM (active queue management) algorithms have been proposed to adapt to changing link rates and traffic conditions. However, these algorithms can only be adopted in AQM networks that have not been widely deployed because of implementation difficulties. Therefore, the algorithms could not be widely used in current networks. It is noted that TCP stacks have adapted in some respects to the changing Internet; for example, increasing bandwidth-delay product have produced efforts to increase the initial congestion window [2,20], including recent proposals [21,22] for this quantity to automatically increase on the timescale of months or years. Nevertheless, the performance improvement only by changing the protocol parameters is still limited. In Remy [23], TCP's entire congestion control algorithm has adapted in response to empirical variations in underlying networks. Unfortunately, this approach seems impractical in that it requires prior knowledge or assumptions about the network and an objective specified by the protocol designer. Monia Ghobadi et al. [24] propose OpenTCP as a TCP adaptation framework that works in SDNs (Software Defined Networks). It requires the assistance of the controllers, and thus could not be adopted in Internet in the short term.

In this paper, we propose an adaptive congestion control framework, named ACCF, which can automatically transition among existing congestion control mechanisms according to the network status. ACCF is orthogonal to previous works improving TCP's performance. It is not meant to be a partial improvement of TCP. Instead, it complements previous efforts by making it easy to switch between different congestion control mechanisms automatically based on network conditions. For instance, with ACCF, one can utilize the congestion control approaches of either DCTCP or Cubic TCP in a data center environment. The decision on which approach to use is made according to current network status. As a preliminary attempt, we apply ACCF to the high BDP networks and adaptively choose the appropriate congestion control approaches from existing ones. In the example case, we focus on the efficient and fair data transfer over high BDP networks with various network devices and links, and we believe ACCF can satisfy the following requirements.

(1) It can efficiently use the network bandwidth in a wide spectrum of network status, including different network bandwidth and diverse router buffer size.
(2) It can have good intra-protocol fairness, especially when the competing flows have different RTTs.
(3) It also can have good TCP friendliness.

Our main contributions are as follows. Firstly, the flexible and scalable architecture of ACCF is presented, including three components: network parameter update, network status estimation and congestion control mechanism adaptation. Secondly, the preliminary application of ACCF is investigated over the high BDP networks to verify its feasibility.

The rest of this paper is organized as follows. In Section 2, a brief overview of related work is presented. Section 3 details the main mechanisms of the proposed framework. In Section 4, simulation-based experiment results and performance measured over real networks are presented. Then further work is discussed in Section 5. Finally, Section 6 concludes the paper.

2. Related work

In this section, we present a brief overview of adaptive approaches and existing TCP variants for the high BDP networks.

2.1. Protocols with adaptive approaches

S. Floyd et al. [18] propose an adaptive RED (Random Early Detection) algorithm that tunes RED's parameters to adjust to current traffic conditions. CoDel [19] is proposed as an innovative approach to AQM, which adapts to changing link rates and is suitable for deployment. These algorithms can only be applied in AQM networks, and thus could not be widely used in current networks.

RFC6928 [21] proposes an experiment to increase the permitted TCP initial window (IW) from between 2 and 4 segments, as specified in RFC 3390 [25], to 10 segments, with a fallback to the existing recommendation when performance issues are detected. Several large scale experiments show that the higher initial window improves the