Computer Networks 57 (2013) 3463–3478

Contents lists available at ScienceDirect

Computer Networks

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

Congestion control of TCP flows in Internet routers by means of index policy ☆

K. Avrachenkov [a], U. Ayesta [b,c,d,g], J. Doncel [c,d,f,*], P. Jacko [e,f]

[a] INRIA Sophia Antipolis, Valbonne, France
[b] IKERBASQUE, Basque Foundation for Science, 48011 Bilbao, Spain
[c] Universite de Toulouse, LAAS, F-31400 Toulouse, France
[d] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
[e] Department of Management Science & LANCS Initiative, Lancaster University, Lancaster, UK
[f] BCAM-Basque Center for Applied Mathematics, Bilbao, Spain
[g] UPV/EHU, University of the Basque Country, 20018 Donostia, Spain

ARTICLE INFO

Article history:
Received 13 September 2012
Received in revised form 20 June 2013
Accepted 3 August 2013
Available online 11 August 2013

Keywords:
Active Queue Management (AQM)
Markov decision process
TCP modeling
Index policies
Whittle index

ABSTRACT

In this paper we address the problem of fast and fair transmission of flows in a router, which is a fundamental issue in networks like the Internet. We model the interaction between a source using the Transmission Control Protocol (TCP) and a bottleneck router with the objective of designing optimal packet admission controls in the router queue. We focus on the relaxed version of the problem obtained by relaxing the fixed buffer capacity constraint that must be satisfied at all time epoch. The relaxation allows us to reduce the multi-flow problem into a family of single-flow problems, for which we can analyze both theoretically and numerically the existence of optimal control policies of special structure. In particular, we show that for a variety of parameters, TCP flows can be optimally controlled in routers by so-called index policies, but not always by threshold policies. We have also implemented the index policy in Network Simulator-3 and tested in a simple topology their applicability in real networks. The simulation results show that the index policy achieves a wide range of desirable properties with respect to fairness between different TCP versions, across users with different round-trip-time and minimum buffer required to achieve full utility of the queue.

1. Introduction

This paper deals with congestion control and buffer management, two of the most classical research problems in networking. The objective of congestion control is to control the traffic injected in the network in order to avoid congestive collapse. Most traffic in the Internet is governed by TCP/IP (Transmission Control Protocol and Internet Protocol) [1,2]. TCP protocol tries to adjust the sending rate of a source to match the available bandwidth along the path. In the absence of congestion signals from the network TCP increases congestion window gradually in time, and upon the reception of a congestion signal TCP reduces the congestion window, typically by a multiplicative factor. Buffer management determines how congestion signals are generated. Congestion signals can be either packet losses or ECN (Explicit Congestion Notifications) [3]. At the present state of the Internet, nearly all congestion signals are generated by packet losses. Packets can be dropped either when the router buffer is full or when an AQM (Active Queue Management) scheme is employed [4].

☆ The authors are grateful to the two anonymous referees for valuable suggestions that helped to improve the presentation of the paper. This research was initiated during the stay of J. Doncel at BCAM during May–September 2011 funded by the BCAM Internship program. This research was partially supported by grant MTM2010-17405 (Ministerio de Ciencia e Innovación, Spain), SA-2012/00331 of the Department of Industry, Innovation, Trade and Tourism (Basque Government), and ADR Semantic Networking of INRIA Alcatel-Lucent joint laboratory.

* Corresponding author at: LAAS-CNRS, Toulouse, France. Tel.: +33 652 985 962.

E-mail address: jdoncel@laas.fr (J. Doncel).

1389-1286/$ - see front matter Crown Copyright © 2013 Published by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comnet.2013.08.001

In this paper we develop a rigorous mathematical framework to model the interaction between a TCP source and a bottleneck queue with the objective of designing optimal packet admission controls in the bottleneck queue. The TCP sources follow the general family of *Additive Increase Multiplicative Decrease* pattern that TCP versions like New Reno or SACK follow [5]. A TCP source is thus characterized by the decrease factor, which determines the multiplicative decrease of the congestion window in the event of a congestion notification (in TCP New Reno the decrease factor takes the value 1/2).

The objective is to design a packet admission control strategy to use the resources efficiently and provide satisfactory user experience. Mathematically we formulate the problem using the Markov Decision Process (MDP) [6] as a dynamic resource allocation problem, which extends the restless bandit model introduced in [7]. The router aims at maximizing the total aggregated utility. As utility function we adopt the parameterized family of generalized α-fair utility functions, which depending on the value of α permit to recover a wide variety of utilities such as max–min, maximum throughput and proportional fair [8,9]. The fixed bandwidth capacity constraint that must be satisfied at all time epochs makes that the problem can be solved analytically only in simplistic scenarios. However, the problem becomes tractable if the fixed capacity constraint is relaxed so that the bandwidth allocation must be satisfied only on average, known as the Whittle relaxation. This relaxation allows to see congestion control at the router as a family of per-flow admission control problems, thus reducing the complexity of a multi-flow problem into a family of single-flow problems. In our main contribution, for the single-flow problem we analyze both theoretically and numerically the existence of optimal control policies of special structure. In particular, we show that for a variety of parameters, TCP flows can be optimally controlled in routers by so-called index policy, but not always by threshold policy.

This solution approach based on the Whittle relaxation has gained notorious success in recent years and has been for instance used in scheduling in wireless networks, dynamic/stochastic knapsack problems, online marketing, or military applications. We refer to [10] for a recent account on the methodology and applications. The interest in the approach is justified by mathematical results that show that, under some additional assumptions, the heuristic is asymptotically optimal [11]. In practice it has been reported on various occasions that the heuristic provides a close-to-optimal solution to the problem studied [10, Chapter 6].

The index policy requires that the TCP sources are assigned a dynamic index that depends on their current congestion window and the TCP variant implemented at the source. Such a *transmission index* measures the efficiency of transmission, and therefore can be interpreted as a priority for transmission. When a packet arrives and the router wants to send a congestion notification (i.e., the buffer is full or the implemented AQM requires to drop or mark a packet), the router will choose the packet with *smallest index*. In the event that more than one packet share the same smallest index, the packet that has been the longest in the queue is chosen.

Building on the cooperative nature of the TCP protocol, we assume that, apart from adjusting the window according to the congestion signals received from the network, the source writes the current index into the packet header, which are read by the routers. Alternatively, index policy can be implemented only in the routers, which however requires to identify or infer the flows, their TCP variants, and the current congestion window of each flow.

Note that our scheme requires as much or less information exchanges than the related congestion control protocols XCP [12], ACP [13] and RCP [14]. Moreover, these congestion control protocols aim at achieving max–min fairness, whereas our scheme allows to achieve several notions of fairness by using the generalized α-fairness framework. Max–min is a particular case of α-fairness when the parameter α goes to infinity. The most important TCP variants are FAST TCP [15], TCP CUBIC [16] and TCP Compound [17], which are implemented in Linux and Windows, respectively. The main originality of our approach with respect to previously existing variants is that our index protocol aims at achieving efficiency and fairness on shorter-term time scales, by maximizing the mean fairness of the actual rate instead of the fairness of the mean rates.

We have implemented our solution in NS-3 [18] and performed extensive simulations in a benchmark topology to explore and validate the properties of the algorithm, and to assess the improvement with respect to *DropTail router* (that drops packets when the buffer is full) and *RED router* (that has implemented the Random Early Detection (RED) AQM mechanism). In the simulations we focus on the case $\alpha = 1$, since it was shown that the current Internet (with DropTail routers) maximizes the aggregate sum of logarithmic utilities of the time-average transmission rates [19].

The simulation results show that the algorithm has several desirable properties with respect to fairness and efficiency:

- We achieve inter protocol fairness between different versions of AIMD TCP.
- We achieve fairness with respect to the round-trip-time on the scale of congestion periods.
- We improve fairness across users with different round-trip-time.
- We improve fairness across users compared to DropTail and RED.
- In a router implementing index policies, a smaller buffer size is needed to get the same throughput.

We believe that our approach opens a new avenue of research to design in a combined manner congestion control and queue management policies. This paper represents a first attempt, and briefly discuss in Section 8 how to assess several aspects that we did not consider in this paper, as, for instance, the performance of the algorithm in the presence of short-lived TCP connections, or when implemented in many routers in the network. We also believe that we provide fundamental framework and ideas useful or