Computer Networks 108 (2016) 160–170

Contents lists available at ScienceDirect

Computer Networks

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

Analysis of ECN/RED and SAP-LAW with simultaneous TCP and UDP traffic

Armir Bujari[a], Andrea Marin[b], Claudio E. Palazzi[a,*], Sabina Rossi[b]

[a] *Università degli Studi di Padova, Italy*
[b] *Università Ca' Foscari Venezia, Italy*

ARTICLE INFO

Article history:
Received 15 December 2015
Revised 10 July 2016
Accepted 15 August 2016
Available online 20 August 2016

Keywords:
ECN/RED
SAP-LAW
Queueing theory
Stochastic processes
Congestion control
Transport protocols

ABSTRACT

Internetworking often requires a large amount of users to share a common gateway to obtain connectivity to the Internet. Congestion avoidance mechanisms are used to prevent the saturation of the gateway which represents a bottleneck of the system. The most popular congestion avoidance mechanisms are the Explicit Congestion Notification (ECN) and the Random Early Detection (RED). Recently, a new method for the congestion avoidance has been proposed: the Smart Access Point with Limited Advertised Window (SAP-LAW). The main idea is to hijack the acknowledge packets in the TCP connections in order to artificially reduce the advertised destination window according to some bandwidth allocation policy. Therefore, the flux control mechanism is artificially exploited to control the congestion at the bottleneck. The advantage of this approach is that it does not require any modification in the TCP implementations at the clients. In this paper, we propose stochastic models for the ECN/RED and SAP-LAW mechanisms in order to compare their performances under different scenarios. The models are studied in mean field regime, i.e., under a great number of TCP connections and UDP based transmissions. Augmenting previous work on ECN/RED, we consider the presence of UDP traffic with bursts, and short lived TCP connections. The models for SAP-LAW are totally new. The comparison is performed in terms of different performance indices including average queue length, system throughput and expected queuing time.

1. Introduction

The explosive growth of computer networks has made the analysis of protocols designed to share one gateway among many devices more and more important. Differently from the early usage of the Internet, now the requirements in terms of bandwidth and responsiveness of many applications is very high and it is often the case that if these requirements are not met then the application usability is not acceptable. A popular and remarkable example is given by online games whose requirements in terms of response time are very strict.

When several devices share the same gateway, we have to address the problem of allocating the resources, i.e., the bandwidth. The mechanisms that are implemented to this end and that we consider in this paper are based on the transport protocols TCP and UDP. As it is well known [1], TCP is used mainly for reliable data transfer (including web browsing), and UDP in multimedia traffic or for applications with stringent requirements on responsiveness such as online gaming (obviously there are exceptions). The reason why real time applications usually prefer UDP, is that it is a connectionless protocol, and it does not manage the retransmission of lost packets, guaranteeing minimum delays in data arrivals. On the contrary, TCP is a connection-oriented transport protocol: it provides reliable delivery and high throughput. For this reason it is particularly suited for applications whose predominant activity is the downloading or the uploading of content, as their main requirement is the complete and correct arrival of data.

TCP and UDP do not have an intrinsic mechanism to regulate the bandwidth when a gateway is shared, but the TCP (in its various versions) admits congestion and flux controls, while the UDP does not implement any of them. Every TCP connection keeps track of two window sizes: its own sending window and the destination window. Congestion control is obtained by allowing the sender window size to increase when packets are properly sent and to decrease when the resources are not enough to satisfy the bandwidth request. Flux control is obtained by choosing a sending rate which is given by the minimum between the sender window size and the destination window size.

Popular approaches that use the built-in congestion mechanism to regulate the TCP transmission rates across a gateway are the

* Corresponding author.
E-mail addresses: abujari@math.unipd.it (A. Bujari), marin@dais.unive.it (A. Marin), cpalazzi@math.unipd.it (C.E. Palazzi), srossi@dais.unive.it (S. Rossi).

http://dx.doi.org/10.1016/j.comnet.2016.08.016
1389-1286/

Random Early Detection (RED) [2,3] and the Explicit Congestion Notification (ECN) [4,5] while the Smart Access Point with Limited Advertised Window (SAP-LAW) [6,7] is based on a flux control mechanism.

The SAP-LAW has been introduced in the context of integrated home server with shared access point. Indeed, due to the rapid evolution of commercial mobile devices, as well as the development of more and more advanced home entertainment systems, ensuring good performances to the applications sharing the same bottleneck has become a hard challenge [8].

In this paper, we study what happens when TCP and UDP are sharing the same link whose congestion is regulated by one of the mechanisms mentioned above: the ECN/RED and the SAP-LAW. On one side, the UDP-based flows seem to be aggressive towards the TCP-based ones since UDP does not provide any flux or congestion control mechanism. However, it has been shown that the TCP's congestion control mechanism causes serious problems to the critical deterioration of UDP-based applications [6,7].

Even if a larger and larger bandwidth is offered today to the user, the coexistence of TCP and UDP continues to present problems: due to its congestion control functionality, TCP continuously probes for higher data rates, queuing packets on the buffer associated with the bottleneck of the connection and, consequently, causing delays for real-time applications. In literature we can find different solutions proposed to overcome the problem, and, in particular, different TCP variants have been proposed, since the UDP is simpler and there are not many variable aspects [9–13]. A large number of papers have studied the performance of TCP and ECN, see [14–19] as a not exhaustive list, but the main focus is on the behaviour of a single TCP connection under various scenarios. More similar to the analysis carried out here is that proposed in [20] where the authors consider the overall system performance in the mean field approximation for ECN/RED. The theoretical results are then extended in [21]. Starting from these two papers we extend the considered model in several directions, as discussed later on, and we propose new ones for SAP-LAW. The mean field theory [21–23] is an important approach to study large stochastic models with a deterministic approximation thus overcoming the well-known problem of the state space explosion.

From the modelling point of view, the contributions of this paper are:

- The extension of the mean field model presented in [20] in order to encompass the presence of UDP traffic.
- The introduction of a totally new model to analyse the SAP-LAW congestion mechanism. This model is completely different from those previously known in the literature such as [20,21] since, according to SAP-LAW, each TCP connections does not necessary use its full congestion window size for each time slot. Therefore, a more detailed state representation is required that poses some computational challenges that we address by resorting to a state space aggregation.
- Another novelty is the relaxation of the common assumption [6,20,21] that TCP connections are greedy, i.e., they always have packets to send so that the sender window is always completely used at each time slot. We will study models that allow TCP connections to send a geometrically distributed random number of packets and then finish their life-cycle. We show that this extension is substantial for comparing the ECN/RED and SAP-LAW algorithms since the latter may allocate some bandwidth to TCP connections that are still in the slow start phase hence cause a waste of bandwidth. We show that our model is able to estimate the throughput reduction given a certain level of greediness of the TCP connections for the SAP-LAW.

Finally, we use these models to compare the performances of ECN/RED and SAP-LAW in the presence of UDP traffic. We consider both the cases of bursty and smooth UDP traffic, and the cases of greedy or non-greedy TCP connections.

The paper is structured as follows. Section 2 explains the ECN/RED and the SAP-LAW congestion mechanisms and Section 3 gives some background about the mean field theory. In Section 4 we introduce our models and give the iterative schemes for the computation of the performance indices. Section 5 briefly discusses the validation of the models and in Section 6 we compare the performance indices of the two congestion mechanisms under different scenarios. Finally, Section 7 concludes the paper.

2. ENC/RED and SAP-LAW congestion control mechanisms

In this section we review the two congestion control mechanisms that we study in this paper: the ECN/RED and the SAP-LAW.

2.1. ECN/RED

ECN [4] is an extension of TCP that avoids packet drops due to channel congestion. This scheme is used in conjunction with RED (Random Early Detection) gateways, marking the packets instead of dropping them, when the queue size is inside a given interval. In particular, RED fixes a minimum and a maximum queue threshold: when the number of packets in the queue is less then the minimum, the packets are not marked, when it is between the minimum and the maximum threshold, it marks the packets with a probability that depends on the queue length. Finally, if the number of the packets is larger than the maximum, all arriving packets are marked or dropped. In this paper we approximate this behaviour by using an exponential law to decide the packet marking probability [24]. The weak point of this strategy is that it does not take into consideration the different requirements of real time applications, for whom packet arrival delays may have catastrophic effects on the usability (e.g., for video streaming or online gaming).

2.2. SAP-LAW

SAP-LAW is a solution proposed in [6] with the specific aim of reducing the problems caused by the simultaneous presence of both TCP-based and UDP-based applications. The basic idea is to find a trade-off between throughput and time delays by working on a dynamic modification of the TCP flow sending rate, in order to avoid the over utilization of the buffer while at the same time guaranteeing a full utilization of the bandwidth. The key factor is to determine the upper bound for the TCP flow sending rate as a function of the amount of UDP traffic. In this way, we are always sure to reserve enough bandwidth to the real time applications activity, without limiting too much the TCP flows. The general formula proposed in [6] is:

$$maxTCP_{\text{traffic}}(t) = \frac{(C^* - UDP_{\text{traffic}}(t))}{\#TCP_{\text{flows}}(t)},$$

where $UDP_{\text{traffic}}(t)$ is the amount of bandwidth occupied by the real time applications at time t, C^* is the capacity of the bottleneck link, and $\#TCP_{\text{flows}}(t)$ is the number of TCP connections at time t. It is noteworthy to point out, that the formula above computes the maximum amount of bandwidth share that can be allocated to a TCP connection while the resulting amount of data is computed by multiplying the computed bandwidth share with the flows current round trip time (RTT) measurement [25–27]. Work [25] proposes an extension to SAP-LAW, addressing the RTT fairness problem amongst TCP flows. We do not contemplate for this feature in the modelling for SAP-LAW and leave it as a potential future work.