Available online at www.sciencedirect.com
SCIENCE DIRECT
Computer Communications 29 (2006) 1868–1878
computer communications
www.elsevier.com/locate/comcom

A rate-based drop policy for punishing unresponsive flows ☆

Ikjun Yeom*

Department of Computer Science, Korea Advanced Institute of Science and Technology, Daejeon, South Korea

Received 15 March 2004; revised 17 May 2005; accepted 17 May 2005
Available online 22 June 2005

Abstract

This paper proposes a drop policy for punishing unresponsive flows. The proposed drop policy uses the difference in traffic arrival patterns of responsive and unresponsive flows, and discards more packets from unresponsive flows. The drop probability of an arriving packet at a router is determined only by the temporal arrival rate of the router at that time, and the router does not need to identify the flow to which the packet belongs. The main advantage of this policy is that it performs well even in the presence of a large number of unresponsive flows since it does not attempt to distinguish high-bandwidth flows. Extensive simulations are presented to show that the proposed policy effectively punishes unresponsive flows by dropping more packets from them in various network situations.

Keywords: Drop policy; Unresponsive flows

1. Introduction

As multimedia/real-time applications become popular, the load of UDP traffic in the Internet increases. Since most of these applications are sensitive to delay and require a certain amount of bandwidth, they send packets at fixed rate, called CBR (Constant Bit Rate), and do not respond to congestion (unresponsiveness). The growth of CBR or unresponsive traffic may cause severe congestion and congestion collapse from undelivered packets. Congestion collapse from undelivered packets happens when network resource is occupied by packets, which eventually, cannot be delivered to their final destinations [1]. This degrades overall performance of the network.

To avoid severe congestion and congestion collapse caused by CBR or unresponsive traffic, recently, two approaches have been studied: (a) application-level congestion control for UDP flows at end systems, and (b) regulating unresponsive flows at edge and/or core routers. (a) aims to provide TCP-like congestion control scheme for UDP applications without degradation of their performance [2–4]. However, there is a limitation of this approach in the sense that it requires users' willingness to deploy it. (b) is to provide scheduling schemes or drop policies to regulate unresponsive flows sending packets more than the fair amount of bandwidth in order to realize fair bandwidth sharing among responsive and unresponsive flows [5–12]. Some of studies [5,9] also claim to give penalty to unresponsive flows in order to encourage them to deploy end-to-end congestion control mechanisms. In this paper, we focus on punishing unresponsive flows at core routers.

The key part of the previous research on fair bandwidth sharing and punishing unresponsive flows is how to identify unresponsive flows occupying excessive bandwidth. Currently, there are several algorithms proposed to identify high bandwidth flows while minimizing overhead. The proposed algorithms use various approaches such as multiple levels of hashing [7], partial flow state [5,10,11] and queue occupancy [6,12]. Once we identify such flows, then it is relatively easy to regulate or punish them. It is observed that these approaches perform well when there are a few unresponsive high-bandwidth flows. In the presence of such multiple flows, however, it becomes harder to identify them. The situation is getting worse when a large number of low-bandwidth unresponsive flows are aggregated. In this situation, the previous approaches are likely to fail to define target flows to be penalized since the individual flows stay within the fair shared amount of bandwidth. Even though each individual flow does not claim more than the fair

☆ This work was supported by University IT Research Center Project.
* Tel.: +82 428 693 544; fax: +82 428 693 510.
E-mail address: yeom@cs.kaist.ac.kr.

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.05.012

shared amount of bandwidth, however, the aggregated flows behave as a single high-bandwidth flow, and it has the potential to cause congestion collapse due to its unresponsiveness.

In this paper, we propose a new drop policy for punishing unresponsive traffic without flow identification. As we discussed above, efforts for flow identification may not work when there exist large number of unresponsive flows. To cope with this situation, the proposed policy does not attempt to distinguish unresponsive flows to punish them. Instead, the policy tries to determine, upon receiving a packet, whether the packet belongs to unresponsive 'traffic' or a specific flow.

To determine unresponsiveness of a packet without flow information, we use changes of arrival rate. Suppose that there are a large number of responsive and unresponsive flows at a router. Responsive flows such as TCP usually increase their sending rate when the network is not congested, and decrease their sending rate responding to network congestion mostly detected by packet losses. The arrival rate of responsive flows keeps changing, mostly oscillating, over time while unresponsive flows keep their sending rate regardless of the presence of network congestion. At a moment, changes of the arrival rate of the aggregate traffic mainly depend on changes of the arrival rate of responsive flows since unresponsive flows maintain their sending rate constantly. When more packets from responsive flows arrive, the arrival rate of the aggregate traffic increases, and vice versa. Hence, the probability that an arriving packet belongs to an unresponsive flow is higher when the arrival rate of the router is low rather than when the arrival rate is high. Then, it is possible to give a higher drop rate to unresponsive flows by dropping more packets when the arrival rate is low.

The proposed policy measures the maximum and minimum arrival rates and derives a drop function in terms of arrival rate. The drop function is inversely proportional to arrival rate and operates in the range between the measured maximum and minimum rates. Upon receiving a packet, it discards the packet with the probability determined by the arrival rate at that moment. Consequently, packets received when arrival rate is lower are more likely to be dropped.

Of course, the arrival rate of the aggregate traffic may also change due to (a) changes of overall amount of responsive and/or unresponsive traffic; and (b) changes of responsive traffic responding to congestion. Here we reasonably assume that changes due to (a) is observed in relatively longer period of time compared to changes due to (b) since responsive flows respond to congestion within one round-trip time (RTT), which is mostly less than several hundreds milliseconds. To deal with changes due to (a), the operation range is adjusted with observation period longer than several RTTs.

The main advantage of this approach is that we can give a preference in the drop rate to responsive flows even in the presence of a large number of unresponsive flows since it does not need any per-flow state and any efforts to identify responsive or unresponsive flows. It measures only the temporal arrival rate of the aggregate traffic. This makes it scalable enough to be deployed in Internet core routers. We evaluate the proposed drop policy through extensive simulations using ns-2 [15]. The results show that it effectively provides higher drop probability to unresponsive flows than to responsive flows in various network scenarios.

The rest of this paper is organized as follows. In Section 2, we describe the proposed drop policy in detail. Performance evaluations through simulations are presented in Section 3. In Section 4, we summarize related work on queue management for punishing unresponsive flows and compare them with the proposed scheme. Finally, we present conclusions in Section 5.

2. A rate-based drop policy

In this section, we propose a new drop policy for punishing unresponsive traffic. The objective of this policy is to discard more packets from unresponsive flows than from responsive flows without any per-flow state maintenance and also without any efforts to identify the flow to which a packet belongs so that it performs well even when a large number of low-bandwidth unresponsive flows are aggregated.

2.1. Motivation and background

Consider Fig. 1 illustrating packet arrivals. Fig. 1(a) and (b) shows a CBR packet arrival pattern and a burstiness of TCP packet arrivals, respectively. A superposition of them is shown in Fig. 1(c). When CBR and TCP traffic are superposed, as shown in the figures, it is expected that the arrival rate measured upon a TCP packet arrival is likely to be higher than the arrival rate upon a CBR packet arrival.

We can also make the similar observation through a simple simulation presented in Fig. 2. In the simulation, there are 30 TCP and 30 UDP-CBR flows competing a 3 Mbps single bottleneck link. The packet size is 1 KB, and the total sending rate of 30 UDP flows is set to 1.5 Mbps. In Fig. 2(a), we present the arrival rate of the aggregated traffic

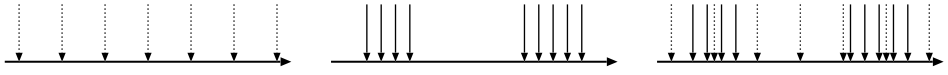

Fig. 1. Traffic superposition.