Available online at www.sciencedirect.com
SCIENCE DIRECT

Computer Communications 28 (2005) 287–299

computer communications
www.elsevier.com/locate/comcom

Delay bounds for FIFO aggregates: a case study

Luciano Lenzini*, Enzo Mingozzi, Giovanni Stea

Dipartimento di Ingegneria dell'Informazione, University of Pisa, Via Diotisalvi 2. I-56122. Pisa, Italy

Received 14 October 2004; accepted 14 October 2004
Available online 14 November 2004

Abstract

In a DiffServ architecture, packets with the same marking are treated as an aggregate at core routers, independently of the flow they belong to. Nevertheless, for the purpose of QoS provisioning, derivation of upper bounds on the delay of individual flows is of great importance. In this paper, we consider a case study network, composed by a tandem of rate-latency servers that is traversed by a tagged flow. At each different node, the tagged flow is multiplexed into a FIFO buffer with a different interfering flow. The tagged flow and the interfering flows are all leaky-bucket constrained at the network entry. We introduce a novel methodology based on well-known results on FIFO multiplexing from Network Calculus, by means of which we derive an end-to-end delay bound for tagged flow traffic. The delay bound assesses the contribution to the delay due to the interference of other flows precisely, and to the best of our knowledge, it is better than any other applicable result available from the literature. Furthermore, we utilize the delay bound formula to quantify the level of overprovisioning required in order to achieve delay bounds comparable to those of a flow-aware architecture.

Keywords: DiffServ; FIFO aggregation; Network calculus; Quality of service; End-to-end delay bounds

1. Introduction

In a nearby future, the Internet will be used as an infrastructure for providing a wide variety of services to its users. New services, such as video streaming, IP telephony and interactive video conference, requiring real-time constraints, are already being deployed and are likely to gain more widespread diffusion. Different services require different Quality of Service (QoS): for instance, a guaranteed delivery service model, enabling packets to reach their destination within pre-specified time constraints, needs to be developed beside the traditional best-effort delivery. The Differentiated Services (DS) architecture, proposed within the IETF [1], is aimed at providing differentiated QoS within the Internet at a manageable complexity. According to DS, packets from different flows are marked at the DS domain ingress as belonging to a small number of different QoS classes, called Behavior Aggregates (BAs), each one receiving a differentiated delivery service within the network. Packets belonging to a BA are then treated at core routers according to a specified per-hop behavior (PHB), independently of the flow they belong to. Currently, the Expedited Forwarding (EF) PHB is specified [2,3] for providing delay guarantees. Practical implementations of the EF PHB assume that all EF traffic is shaped and policed at the DS domain ingress, and then shares a single FIFO queue at each core router, which is either serviced at the highest priority or is guaranteed a large bandwidth share in a weighted fair scheduler. In this last case, capacity is statically reserved to the aggregate traffic at core routers, whereas appropriate admission control is performed at the DS domain edges, to provide specific QoS guarantees to the aggregate.

Regarding DiffServ, two different approaches have been developed: *relative* Differentiated Services, which is aimed at providing differentiated performance to the different aggregates, without taking into account explicit per-flow guarantees [4,5]. On the other hand, *absolute* Differentiated Services is aimed at providing end-to-end *absolute* performance guarantees to single flows, without the need of per-flow state in the core [6]. This last approach follows

* Corresponding author. Tel.: +39 050 2217 666; fax: +3 9050 2217 600.

E-mail addresses: l.lenzini@iet.unipi.it (L. Lenzini), e.mingozzi@iet.unipi.it (E. Mingozzi), g.stea@iet.unipi.it (G. Stea).

0140-3664/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2004.10.003

from considering that single Internet users would require individual, per-flow (rather than per-aggregate) end-to-end QoS guarantees. Within this last framework, analytical derivation of end-to-end delay bounds for individual flows is of great importance, since they can be used as the base for call admission control. In [7], per-flow delay bounds were derived for a generic network as a function of the utilization factor, the maximum hop count, and the parameters of the ingress shapers, without any assumption on the topology. However, in a network domain employing centralized resource management, it is reasonable to assume that the management entity is aware of all the requests at the domain edge, as well as of the domain topology: this is the case, for example, of the Bandwidth Broker (BB) service proposed in [8]. However, how the knowledge of the network topology and current load can be exploited in order to derive delay bounds useful for performing careful admission control is an open issue.

In this paper, we consider a case study network, composed by a tandem of rate-latency servers that is traversed by a tagged flow that is leaky-bucket constrained. The tagged flow is multiplexed into a FIFO buffer with another leaky-bucket constrained flow at each node. Our purpose is to derive an end-to-end delay bound for the traffic of the tagged flow. To this aim, we base our work on well-known results on FIFO multiplexing that have been derived by means of network calculus, and we describe a methodology that appropriately combines these results to derive a delay bound for the case study. To the best of our knowledge, the derived delay bound is better than any other applicable result from the literature. Furthermore, we show that a possible interpretation of the obtained result, is that the so-called 'pay bursts only once' property would no longer hold for the tagged flow in the case study network, reflecting the fact that sometimes paying the burst more than once, but at a higher rate, is better than paying it only once at a lower rate.

The remainder of the paper is organized as follows. In Section 2 the key concepts and some basic results of network calculus are given. By showing the limits of current results related to delay bounds for FIFO aggregates, Section 3 lays the foundation for our study, whose results are presented in Section 4. Section 5 addresses the related work, while Section 6 introduces some numerical results useful to get an insight into the analytical results. Finally, Section 7 draws some conclusions.

2. Network calculus fundamentals

Network calculus is a theory for deterministic network analysis. In next subsections, key concepts and some well-known results from network calculus are given. An introduction to the subject can be found in [9], whereas a comprehensive treatment can be found in two books [10,11].

2.1. Service curves

The concept of service curve is introduced as a general means to model a network element. Through its service curve, the network element is characterized, independently of specific internals, in terms of input and output flow relationships, i.e. how the element transforms an arriving stream of packets into a departing stream. To this aim, data flows are described by means of the cumulative function $R(t)$, defined as the number of bits seen on the flow in time interval $[0,t]$. Function $R(t)$ is always wide-sense increasing, that is $R(S) \le R(t)$ if and only if $s \le t$. Furthermore, it is assumed that $R(t)=0$ for $t \le 0$.

Specifically, let $A(t)$ and $D(t)$ be the cumulative functions characterizing the same data flow before entering a network element, and after having departed, respectively. Then, the network element can be modeled by the service curve $\beta(t)$ if

$$D(t) \ge \inf_{0 \le s \le t} \{A(t-s) + \beta(s)\} \tag{1}$$

for any $t \ge 0$. The flow is said to be guaranteed the minimum service curve β. The infimum on the right side of (1), as a function of t, is called the min-plus convolution of A and β, and is denoted by $(A \otimes \beta)(t)$. Min-plus convolution has several important properties, including being commutative and associative, from which a fundamental result of network calculus follows, that is: the service curve of a feed-forward sequence of network elements traversed by a data flow is obtained by convolving the service curves of each of the network elements.

Several network elements, such as delay elements, links, and regulators, can be modeled by corresponding service curves. More interestingly, it has been shown that many schedulers proposed for ATM or the Internet integrated services can be modeled by a family of simple service curves called the rate-latency service curves, defined as

$$\beta_{\rho,\theta}(t) = \rho[t-\theta]^+ \tag{2}$$

for some $\rho \ge 0$ (the rate) and $\theta \ge 0$ (the latency). Notation $[x]^+$ denotes max $\{0,x\}$.

2.2. Arrival curves

Differentiated and integrated services assume that traffic flows are constrained. In network calculus this feature is modeled by introducing the concept of arrival curve. A wide-sense increasing function α is said to be an arrival curve (or, equivalently, an envelope) for a flow characterized by a cumulative function R if it is $R(t)\text{-}R(\tau) \le \alpha(t-\tau)$, for all $\tau \le t$. As an example, a flow regulated by a *leaky bucket* network regulator, with rate ρ and burst size σ, is constrained by the arrival curve

$$\gamma_{\rho,\sigma}(t) = (\sigma + \rho t)1_{\{t>0\}}. \tag{3}$$

The indicator function $1_{\{\text{expr}\}}$ is equal to 1 if *expr* is true, and 0 otherwise.