Measurement-Based Admission Control for Dynamic Multicast Groups in Diff-Serv Networks*

Elena Pagani and Gian Paolo Rossi

Dip. di Scienze dell'Informazione, Università degli Studi di Milano
via Comelico 39, I-20135 Milano, Italy
{pagani,rossi}@dsi.unimi.it

Abstract. An appealing approach to the admission control problem for traffic with QoS requirements consists in evaluating the resource availability by means of *measurement-based* techniques. Those techniques allow to provide QoS with minimal changes to the current network devices. In this work, we propose a mechanism to perform active measurement-based admission control for *multicast groups with dynamically joining receivers*. The proposed mechanism has been implemented in the NS-2 simulation framework, to evaluate its performance.

1 Introduction

The *Differentiated Service* model [1] has been proposed in the literature to provide QoS in a scalable manner. According to that model, *bandwidth broker* agents [4] exist that take in charge the traffic admission control functionalities. Yet, only a few practical implementations of the diff-serv model have been realized. Moreover, in the diff-serv model it is difficult to support multicast [1].

In this paper we describe the end-to-end *Call Admission Multicast Protocol* (CAMP) [5], that can be used to ensure bandwidth guarantees to multicast sessions in IP networks, thus providing them with the Premium Service [4]. CAMP is scalable, operates on a per-call basis and supports the group membership dynamics. It performs the functionalities of a *distributed bandwidth broker* (BB). To perform the admission control, CAMP adopts an active-measurement approach. We have implemented CAMP in the frame of the NS-2 simulation package to verify its effectiveness under different system conditions.

In the system model we consider, a QoS-sensitive application specifies to the underlying service provider, the QoS communication requirements and the behaviors of the data flow it is going to generate (*traffic profile*). We assume that a session announcement protocol (e.g., `sdr`) is available to announce the needed session information. We consider sources generating CBR traffic. All the recipients receive the same set of microflows; they have the same QoS requirements.

* This work was supported by the MURST under Contract no.MM09265173 "Techniques for end-to-end Quality-of-Service control in multi-domain IP networks".

E. Gregori et al. (Eds.): NETWORKING 2002, LNCS 2345, pp. 1184–1189, 2002.

We adopt the notation proposed in [1] for the *differentiated services* (diff-serv, DS) model. We consider the functional architecture of the BB in accordance with the proposal presented in [7]. The BB provides the applications with the interface to access the QoS services. When the QoS aggregate spans multiple domains, an inter-domain protocol is executed amongst peer BBs, to guarantee the proper configuration of the transit and destination domains.

2 Distributed Bandwidth Broker

In this section we outline the end-to-end *Call Admission Multicast Protocol* (Camp); greater details can be found in [5]. In Figure 1, we show the system architecture in which Camp works. Camp operates within the RTP/RTCP [6] protocol suite and performs the set-up of a RTP session. It receives from the application, via RTP, the profile of the data traffic that will be generated. Camp uses RTCP to monitor the QoS supplied to the recipients. Camp performs the admission control using an *active measurement* approach [2]. It generates *probing* traffic with the same profile as the data traffic generated by the application. Both the data and the `probe` packets are multicast. To this aim, we assume that both a membership protocol and a multicast routing protocol are available. The latter maintains a tree-based routing infrastructure connecting the multicast recipients. Camp is independent of both those protocols. The *probing* phase has

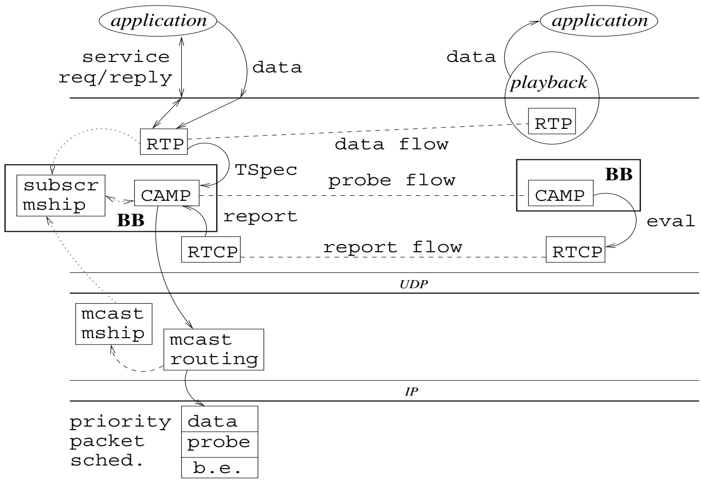

Fig. 1. Architecture of Camp-based end stations

the aim of evaluating whether the available bandwidth is sufficient to support the new traffic. With respect to the classification given in [2], we adopt *out-of-band* probing with *dropping* of the `probe` packets as the congestion signal. All the routers use a *priority* packet scheduling discipline: the `probe` packets are marked with a higher priority than the best effort packets, but a lower priority than the QoS packets. This priority assignement ensures that the probing traffic does not affect the established QoS sessions. On the other hand, `probe` packets can drain the available bandwidth for the new QoS session at the expenses of the best effort traffic.

To support multicast, two issues must be considered: (*i*) the receiver group membership can dynamically change; and (*ii*) different destinations can experiment different QoS in receiving the same traffic. To cope with problem (*ii*) above, the recipients that receive the probing traffic with low quality prune from the tree and refuse the service, by sending a refusing RTCP report to the source. When all the reports have been received by the source, if the service is accepted by at least one recipient, the source switches from the transmission of `probe` packets to the transmission of the data packets generated by the application, *without discontinuity*. The data packets are forwarded along the pruned tree.

We deal with the problem (*i*) above using a *proxy* mechanism. The source announces the multicast session via `sdr` and starts transmitting `probe` packets at the scheduled time, if at least one receiver is listening. A CAMP proxy is instantiated in a router either in the initialization phase, or when one or more new downstream output interfaces (*oif*s) appear in the router for the group (dynamic membership changes). The proxy remarks as `probe` packets all the incoming packets for the session, that must be forwarded to the probing *oif*s. This way, the data sent to the new destinations do not affect previously established sessions traversing the new branch. The proxy lifetime lasts until, for each probing *oif*, either it is pruned from the tree (as the result of a service refusal), or an acceptance report is received from it. In the latter case, if the initialization phase is ongoing, the report is forwarded to the source. The source CAMP entity switches to the transmission of the data as soon as it receives an accepting report. The proxy mechanism allows to hide the membership to the source.

3 Performance Evaluation

We have implemented the architecture shown in Figure 1, in the frame of the NS-2 simulation package [3]. The simulations have been performed with a meshed network of 64 nodes, connected by optical links of 2 Mbps bandwidth and variable length in the range 50 to 100 Km. Background, best effort traffic is uniformly distributed all over the network; best effort sources generate CBR traffic with a 0.66 Mbps rate. The size of best effort, `probe` and data packets is 512 bytes. We embedded a real RTP implementation into the RTP template of NS-2. The recipients dynamically join the group; we performed experiments with different join rates. The multicast tree is incrementally built as join events occur; the source is located in the tree root. The source does not know the group of recipients; it generates CBR traffic whose rate assumes different values in the range 0.4 Mbps to 1.9 Mbps. During the probing phase, the recipients compare the received rate with the source rate: if the difference is below a tolerated threshold, a recipient

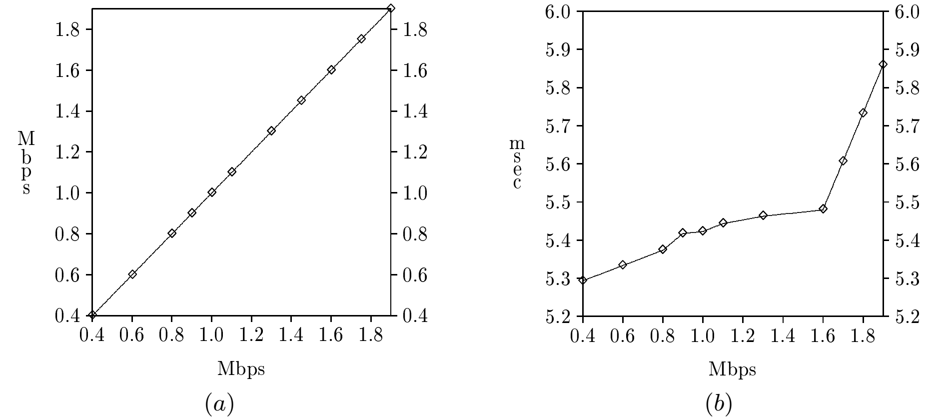

Fig. 2. (a) Throughput vs. offered load for $|G| = 10$. (b) Average end-to-end delay vs. offered load for $|G| = 10$

sends a positive report. We performed measures for different thresholds [5]. The recipient decision is sent within the first RTCP report a destination generates after the reception of a number of `probe` packets, i.e. of samples, sufficient to ensure an accurate measure of the available bandwidth by covering the rate of the slowest traffic source. We performed simulations with different sample sizes.

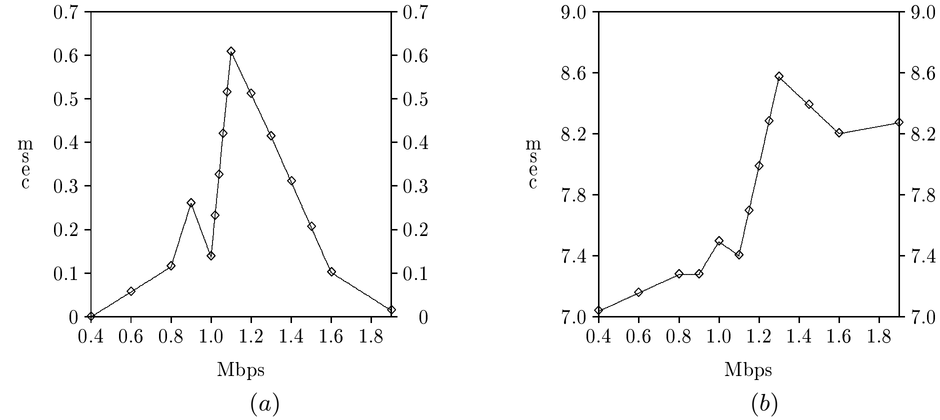

Fig. 3. (a) Jitter vs. offered load, for $|G| = 10$. (b) Fair delay vs. offered load, for $|G| = 10$

By performing simulations with different group cardinalities, we observed that the performance is almost independent of this parameter. We performed simulations with recipients that join the group with different rates. The proxy mechanism has proved to be effective in performing the admission control, and the measured performance is independent of the frequency with which recipients join the group. The results shown in this section have been obtained for a group of 10 recipients, acceptance threshold set to 5% of the source rate and frequency of the join requests arrival 1 sec. The measures have been taken after 20 sec. from the end of the set-up phase of the last grafted recipient.

Simulations indicate that Camp effectively performs the call admission control. The recipients accepting the transmissions receive at the correct data rate. In Figure 2, we report the throughput (a) and the end-to-end delay (b) averaged over all the recipients; no receiver has refused the service. The delay increases when the offered load approximates the link capacity, while it is independent of the interference of the best effort traffic. This indicates that the sessions characteristics are preserved from source to destination, independently of the other network load.

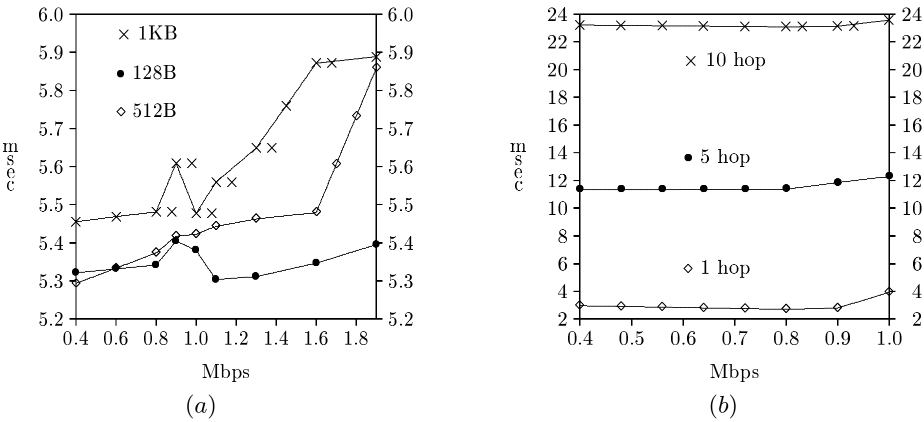

Fig. 4. (a) Average end-to-end delay vs. offered load as a function of the best effort packet size, for $|G| = 10$. (b) Average end-to-end delay vs. offered load as a function of the receivers distance from the source

The jitter has been computed according to the algorithm given in the RTP specification [6]; it is reported in Figure 3(a). The jitter behaviour indicates that at the receiver side a delivery agent must be used to perform the playback of the source transmission. The jitter shows a peak in correspondence with the maximum contention between the QoS and the best effort traffic. After that value, the QoS traffic pushes the best effort traffic away from the tree branches, and best effort packets start to be dropped from the queues.

The *fair delay* is the maximum difference between the end-to-end delays perceived by two different destinations. Its behaviour (Figure 3(b)) indicates that the destinations at a greater distance from the source greatly suffer the network congestion. In the worst case, this could result in a lower probability of successful service set-up for the farest destinations. However, we never observed service refusal.

The measures of the jitter and the fair delay show the effects of the presence of best effort packets at the core routers. As expected, the priority mechanism alone is not sufficient to ensure jitter control at the destinations. To highlight the impact of the best effort traffic over the QoS, we performed simulations with different best effort packet sizes. In Figure $4(a)$ we report the end-to-end delay observed by the QoS packets that compete with best effort traffic generated as before. As the links cannot be preempted once a packet transmission is ongoing, QoS packets arriving at a node could have to wait at most for a best effort packet transmission time before gaining the link, although they have the highest priority. The impact of the delay over the received rate is however negligible.

We performed an experiment with two sources: the former one has a 1 Mbps rate; we varied the rate of the latter source. In figure $4(b)$, we show the average delay measured with respect to the load generated by the second source and the distance of the recipients from the source. The contention probability amongst different sessions increases with the path length: it affects the queueing delays, thus altering the regular traffic profile. The impact on the received throughput is however negligible.

The achieved results show that the devised mechanism effectively performs admission control. Yet, further investigation has to be carried out concerning the interactions amongst several concurrent transmissions and their impact on the probability of successful service establishment.

References

1. Blake S., Black D., Carlson M., Davies E., Wang Z., Weiss W. "An Architecture for Differentiated Services". *RFC 2475*, Dec. 1998. Work in progress.
2. Breslau L., Knightly E., Shenker S., Stoica I., Zhang H. "Endpoint Admission Control: Architectural Issues and Performance". *Proc. SIGCOMM'00*, 2000, pp. 57-69.
3. Fall K., Varadhan K. "ns Notes and Documentation". The VINT Project, Jul. 1999, `http://www-mash.CS.Berkeley.EDU/ns/` .
4. Nichols K., Jacobson V., Zhang L. "A Two-bit Differentiated Services Architecture for the Internet". *Internet Draft, draft-nichols-diff-svc-arch-00.txt*, Nov. 1997. Work in progress.
5. Pagani E., Rossi G.P., Maggiorini D. "A Multicast Transport Service with Bandwidth Guarantees for Diff-Serv Networks". *Lecture Notes in Computer Science 1989*, Jan. 2001, pp. 129. Springer, Berlin.
6. Schulzrinne H., Casner S., Frederick R., Jacobson V. "RTP: A Transport Protocol for Real-Time Applications". *RFC 1889*, Jan. 1996. Work in progress.
7. Teitelbaum B., Chimento P. "QBone Bandwidth Broker Architecture". *Internet 2 QoS Working Group Draft*, Jun. 2000. Work in progress. `http://qbone.internet2.edu/bb/`