

Scalable Quantitative Delay Guarantee Support in DiffServ Networks Through NSIS

Jian Zhang, Maxweel Carmo, Marilia Curado, Jorge Sá Silva, and Fernando Boavida*

Laboratory of Communications and Telematics (LCT), University of Coimbra
CISUC-DEI, Polo II, 3030-290 Coimbra, Portugal
{zhang, maxweel, marilia, sasilva and boavida}@dei.uc.pt

Abstract. This paper investigates the issue of enabling scalable quantitative delay guarantee support in DiffServ networks through NSIS. A NSIS QoS Model is utilized to add the admission control framework to the DiffServ architecture and reservation-based admission control algorithms are designed for ingress and interior nodes respectively for enabling quantitative delay guarantees in a DiffServ domain. Due to the NSIS protocol suite can support aggregate reservations effectively and the admission control algorithms are distributed at the ingress and interior nodes, our approach can enable the quantitative end-to-end delay guarantees in a DiffServ domain while still maintaining its simplicity and scalability.

1 Introduction

The next generation Internet will provide advanced features, such as the Quality of Service (QoS) guarantees, to end-users and their applications. The DiffServ QoS architecture [2], which realizes the service differentiation by deploying a small number of pre-defined and agreed forwarding behaviors (i.e., Per-Hop Behavior (PHB)) at a DiffServ network has been paid many attentions due to its simplicity and scalability. However, DiffServ does not offer any explicit resource reservation mechanism and can only provide some level of qualitative service differentiation. Meanwhile, delay-sensitive applications, such as Voice over IP (VoIP) or Video-on-Demand (VoD), require quantitative QoS guarantees to maintain the timely arrival of their packets. Thus, to effectively support the delay-sensitive applications in a DiffServ network, extra mechanisms and algorithms, capable of offering the quantitative QoS guarantees while preserving the simplicity and scalability nature of DiffServ, must be developed. This is a challenging task for the networking researchers.

Currently, The IETF Next Steps in Signaling (NSIS) working group is working on a more generic signaling architecture than RSVP for the Internet, which

* This work has been partly supported by the European Commission under IST project EuQoS

consists of two distinct signaling layers: NTLP (NSIS Transport Layer Protocol) and NSLP (NSIS Signaling Layer Protocol). For the QoS signaling purpose, a IETF draft QoS-NSLP (NSLP for Quality-of-Service signalling) [4] has been proposed that provides a general model for each network to implement a specific QoS Model appropriate to the network technology in use and supports aggregate reservations. Hence, NSIS is an attractive signaling approach for the DiffServ.

This paper investigates the issue of enabling quantitative delay guarantee support in a DiffServ network while still maintaining its simplicity and scalability by using NSIS and distributed admission control algorithms. In particular, a NSIS QoS Model of DiffServ is utilized to signal the token bucket parameters and requested quantitative delay guarantees of incoming traffics to the edge and interior nodes of a DiffServ network, where the edge nodes maintain per-flow QoS-NSLP and reservation states whereas the interior nodes maintain only per-class states. Moreover, distributed admission control algorithms are designed for the ingress and interior nodes of the DiffServ network, respectively, based on the delay bound proposed in [3]. Since per-class aggregate reservations are fulfilled and the admission control algorithms are distributed at the ingress and interior nodes, this approach can effectively enable the quantitative delay guarantees in the DiffServ architecture while still preserving the simplicity and scalability.

The rest of the paper is organized as follows. In Section 2, we discuss related work. In Section 3, we describe our approach for scalable quantitative delay guarantee support in DiffServ networks. In Section 4, we present some concluding remarks.

2 Related Work

Provisioning quantitative QoS guarantees in DiffServ networks while maintaining the simplicity and scalability nature of DiffServ paradigm has been a challenging task for networking researchers due to the fact that the DiffServ model lacks a standard definition of signaling mechanisms and an admission control framework. A bandwidth broker, which is a central server (per domain) that arbitrates access to a statically provisioned logical partition of a network's resources, was exploited to add the admission control to DiffServ networks in [5]. All session requests are directed to the bandwidth broker, which holds a map of the network and keeps track of utilization of each resource in the network. However, due to the centralized processing nature of bandwidth brokers, a bandwidth broker has to deal with all session set up requests to its domain, which makes it exhibit the same lack of scaling (in the complexity growth sense) as IntServ.

Liao *et al.* studies the issue of provisioning quantitative differentiated services in DiffServ by proposing a set of dynamic node and core provisioning algorithms for interior nodes and core networks, respectively [6]. The node provisioning algorithm prevents transient violations of service level agreements by predicting the onset of service level violations based on a multi-class virtual queue measurement technique, and by automatically adjusting the service weights of weighted fair queueing schedulers at core routers. Persistent service level violations are re-

ported to the core provisioning algorithm, which dimensions traffic aggregates at the network ingress edge. Note that the per-class quantitative delay guarantees provided in [6] are only in the scope of one node, i.e., the quantitative delay guarantee of a service class bounds only the packet delay across one network node. In contrast, in this paper we address the issue of provisioning end-to-end (or edge-to-edge) quantitative delay guarantees across the whole DiffServ network, which is more complex than the scenario in [6].

3 Scalable Quantitative Delay Guarantee Support in DiffServ Networks

This section first describes the network architecture and service model used in this paper and then our approach of enabling scalable quantitative delay guarantee support in DiffServ networks is presented.

3.1 Network Architecture and Service Model

We assume a DiffServ framework where all nodes in the network are NSIS aware and output-buffered, implementing class-based priority scheduling. At least 2 classes of edge-to-edge flows are considered there and we refer to one of the classes as *priority* class (e.g., Expedited Forwarding class). At each node in the DiffServ network, packets belonging to the priority class are queued in a separate priority queue which is served at strict non-preemptive priority over any other queue. Moreover, we assume that any flow i belonging to the *priority* class is shaped to conform to a leaky bucket with parameters (r_i, b_i) when it arrives at the ingress node of the DiffServ. Furthermore, flow i indicates its traffic parameters (r_i, b_i) and its QoS requirements (in this paper, i.e., the requested end-to-end delay guarantee D_i across the DiffServ) to the ingress node. Note that various alternatives (e.g. SIP, RSVP, NSIS, etc) can be used by incoming flows to express their traffic parameters and QoS requirements to the ingress node and no assumptions are made about that here. Then, NSIS signaling messages will be exchanged in the DiffServ network to first discover the data path which an incoming flow will take to pass through the network, then to check whether there are enough resources available along each link in the data path to meet the requested delay guarantee and to reserve the appropriate resources when they are available.

3.2 Distributed Admission Control Algorithms

The distributed admission control algorithms used by the ingress and interior nodes are derived here. First, we define some parameters as follows. MTU is to denote the maximum size of any packet in the network, h_{max} to denote the maximum number of hops any flow in the network could traverse, C_l to denote the capacity of line l , S_l to denote the set of all priority flows constituting the priority aggregate on link l , P_l to denote the maximum rate with which

the priority traffic aggregate is injected to link l and ρ to denote the ratio of the capacity of any link in the network to be devoted to the priority class. According to [3], a bound on the worst-case end-to-end delay for the priority class traffic is $D = \frac{h_{max}}{1-(h_{max}-1)u\rho}(\Delta + u\tau)$ provided that the three inequalities hold: $\rho < \min_l \frac{P_l}{(P_l - C_l)(h_{max}-1) + C_l}$, $\sum_{i \in S_l} r_i \leq \rho C_l$ and $\sum_{i \in S_l} b_i \leq \tau C_l$, where $u = \max_l \frac{P_l - C_l}{P_l - \rho C_l}$, $\Delta = \max_l \frac{MTU}{C_l}$ and τ is the parameter to be set to bound the sum of the token bucket depths burst at any link relative to the link capacity. Note that, in many cases, the depth of the leaky bucket of a flow depends linearly on the rate of the flow, such that $b_i \leq \tau_0 r_i$ for every flow i and for some τ_0 . In such cases, we set $\tau = \rho \tau_0$.

Given the network topology of a DiffServ domain, we can set the value for ρ according to the above inequalities and set the value for τ based on the estimated burstiness of priority class traffic. Moreover, the value of Δ and the value of u can be determined based on the traffic dynamics and the considered network topology. Note that, if there is no information about the value of P_l , it can be set as the sum of the bit rates of all incoming links to link l . Then, when a new flow (flow i) belonging to the priority class arrives, the ingress node will first check whether this flow can be accepted to its outgoing link. If the flow is accepted, it will send a NSIS signaling message to discover the data path which flow i will pass through and during the course of the path discover, the interior nodes in the data path will perform their admission control algorithm successively to check whether or not flow i can be accepted to the links it will traverse. The admission control algorithm for the interior node (including the ingress node), whose output link is link l , is designed as follows: if both $R_{curr} + r_i \leq \rho C_l$ and $B_{curr} + b_i \leq \tau C_l$ hold, flow i can be accepted to link l , otherwise, flow i is rejected, where $R_{curr} = \sum_{j \in S_l} r_j$ and $B_{curr} = \sum_{j \in S_l} b_j$. If flow i is accepted by all interior nodes in its data path and the number of hops it will traverse in the network is h , the ingress node will perform the following admission control algorithm to decide whether this flow will be accepted or rejected to the DiffServ network: if

$$D_i \geq \frac{h}{1 - (h-1)u\rho}(\Delta + u\tau) \quad (1)$$

holds, flow i is accepted, otherwise, it is rejected, where D_i is the end-to-end delay guarantee requested by flow i . Of course, if flow i is rejected by any interior node in its data path, it will be rejected to the network by the ingress node without executing the ingress node's admission control algorithm.

It can be observed from Eq. (1) that the right item of (1) is from the delay bound in [3] except that h_{max} is replaced by h due to the fact that the exact number of hops every flow will pass through can now be obtained via the NSIS signaling exchange. Since flow i is accepted by all interior nodes in its data path means that all the above inequalities has already been satisfied by the selection of ρ for the network, the packet delay d_i of flow i will be bounded by the value of $\frac{h}{1-(h-1)u\rho}(\Delta + u\tau)$, i.e., $d_i \leq \frac{h}{1-(h-1)u\rho}(\Delta + u\tau)$. Now, if flow i is accepted to the network by the ingress node, we can obtain $d_i \leq \frac{h}{1-(h-1)u\rho}(\Delta + u\tau) \leq D_i$, i.e., the requested worst-case end-to-end delay guarantee of flow i is satisfied.

3.3 NSIS QoS Model

The NSIS QoS Model for enabling quantitative delay guarantees in DiffServ networks is presented below.

First of all, the NSIS QoS Model allows external traffics to express their traffic parameters and quantitative QoS requirements (e.g. quantitative bandwidth, delay, jitter or loss guarantee) to the ingress nodes of a DiffServ domain. Secondly, the NSIS QoS Model supports two sets of admission control algorithms (measurement-based and reservation-based admission control) to satisfy the quantitative QoS requirements of accepted flows. In this paper, the reservation-based admission control algorithms are designed to illustrate the usage of the NSIS QoS Model for enabling quantitative delay guarantees in a DiffServ domain.

In particular, at a DiffServ domain where all nodes deploying this NSIS QoS Model, the edge nodes will store and maintain per-flow NTLP, QoS-NSLP and QoS Model related reservation states. The interior nodes will be NTLP stateless, which means no NTLP states need to be stored, and be either QoS-NSLP stateless (for measurement-based admission control operation), or are reduced-state nodes storing per PHB aggregated QoS-NSLP and reservation states (for reservation-based admission control operation). As described in [4], four message types: RESERVE, QUERY, RESPONSE and NOTIFY have currently defined to support the QoS signaling operations, which consist of three types of QoS-NSLP objects including the QoS specification (QSPEC) object. QSPEC object describes the actual resources that are required and depend on the QoS Model being used. The QSPEC object of the NSIS QoS Model presented here contains three fields: the QoS Description, the Per Hop Reservation Control Information (PHR Control Information) and the Per Domain Reservation Control Information (PDR Control Information). In particular, the QoS Description field contains the following parameters: **<QoS Description>** = **<Token Bucket>** **<PHB Class>** **<Bandwidth>** **<Path Latency>** **<Path Jitter>** **<Packet Loss Ratio>** **<Packet Error Ratio>**, where the bit formats of all above parameters conform to the bit formats specified by the QoS-NSLP QSPEC template [1]. These parameters describe the characteristics of incoming traffics and the quantitative QoS guarantees requested to the DiffServ domain.

3.4 Scalable Quantitative Delay Guarantee Support Through NSIS

Our approach for the scalable quantitative delay guarantee support in a DiffServ domain is illustrated here. First of all, we assume that a new flow (flow i) belonging to the priority class indicates its token bucket parameters (r_i and b_i) and requested end-to-end delay guarantee D_i to one ingress node of a DiffServ domain. Then, if the flow can be accepted to the outgoing link of the ingress node, it will encapsulate parameters r_i and b_i into the **<Token Bucket>** field of the QoS Description and send a QUERY message to discover the data path that flow i will take and to retrieve the number of hops it will pass through. Next, each interior node at the data path of flow i will execute the admission

control algorithm in Section 3.2 to check whether it can be accepted to the outgoing link of the interior node. When the QUERY message reaches the egress node, the egress node will report the QUERY result to the ingress node via the RESPONSE message which will however be bypassed by those interior nodes. If the QUERY result is positive, the ingress node will perform the admission control algorithm of Eq. (1) to check whether the requested end-to-end delay guarantee D_i can be satisfied or not. If D_i can be met by the DiffServ domain, the ingress node will send the RESERVE message to reserve the pertinent resources for flow i along the data path already discovered, in this paper, i.e., each interior node (including the ingress node) will add r_i and b_i to its R_{curr} and B_{curr} of the priority class, respectively. After the ingress node receives the positive RESERVE report from the egress node, it will mark the packets of flow i as priority class (here, i.e., EF class) and admit them into the DiffServ network. Otherwise, flow i is rejected.

4 Conclusions

This paper investigates the issue of enabling scalable quantitative delay guarantee support in DiffServ networks through NSIS. A NSIS QoS Model for DiffServ networks is utilized to signal the token bucket parameters and requested quantitative delay guarantees of incoming traffics to the edge and interior nodes in a DiffServ network, where the edge nodes maintain per-flow QoS-NSLP and reservation states whereas the interior nodes maintain only per-class states. Moreover, distributed admission control algorithms are designed for the ingress and interior nodes of the DiffServ network, respectively, based on the delay bound proposed in [3]. Due to the NSIS protocol suite can support aggregate reservations effectively and the admission control algorithms of our approach are distributed at the ingress and interior nodes, our approach can enable the quantitative end-to-end delay guarantees in a DiffServ domain while still maintaining its simplicity and scalability.

References

1. J. Ash, A. Bader and C. Kappler. QoS-NSLP QSPEC Template, Internet draft, work in progress, Internet Engineering Task Force, October 2005.
2. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss. An Architecture for Differentiated Service, RFC 2475, Internet Engineering Task Force, December 1998.
3. A. Charny and J.Y. L. Boudec. Delay Bounds in a Network with Aggregate Scheduling, Proc. of QoSIS 2000, LNCS 1922, pp. 1-13, 2000.
4. J. Manner, G. Karagiannis, A. McDonald, S. Van den Bosch. NSLP for Quality-of-Service signalling, Internet draft, work in progress, Internet Engineering Task Force, October 2005.
5. K. Nichols, V. Jacobson and L. Zhang. A two-bit differentiated services architecture for the Internet, RFC 2638, Internet Engineering Task Force, July 1999.
6. R.-F. Liao and A. T. Campbell. Dynamic Core Provisioning for Quantitative Differentiated Services, IEEE Transactions on Networking, 2004.