SRBQ and RSVPRAgg: A Comparative Study

Rui Prior[1], Susana Sargento[1,2], Pedro Brandão[1], Sérgio Crisóstomo[1]

[1]DCC-FC & LIACC, University of Porto, Portugal
[2]Institute of Telecommunications, University of Aveiro, Portugal
{rprior, ssargento, pbrandao, slc}@ncc.up.pt

Abstract This paper presents a comparative evaluation of the Scalable Reservation-Based QoS (SRBQ) and the RSVP Reservation Aggregation (RSVPRAgg) architectures, both designed to provide QoS levels similar to RSVP/IntServ without the scalability concerns that prevent its usage in high-speed core networks. The comparative analysis, based on simulation results, shows that SRBQ provides the same QoS guarantees of RSVPRAgg, with significantly increased network resource utilisation and a small penalty in signalling processing overhead. This stems from the fact that although based on end-to-end reservations, SRBQ makes use of techniques and algorithms that reduce the computational complexity of signalling processing, increasing its scalability.

1 Introduction

With the goal of benefiting from the virtues of both IntServ [1] and DiffServ [2] and mitigating their respective problems, several architectures have been proposed in the literature. None of them, however, ensures simultaneously the strict and differentiated QoS support and the maximisation of the usage of network resources without scalability concerns. One of the most promising architectures [3] is based on the aggregation of per-flow reservations, where the RSVP is extended to allow RSVP signalling messages to be hidden inside an aggregate. In the simplest case, reservations of aggregate bandwidth are performed between ingress and egress routers of a network domain; these reservations are updated in bulks much larger than the individual flow's bandwidth. Whenever a flow requests admission in an aggregate region, the edge routers of the region check if there is enough bandwidth to accept the flow on the aggregate. If enough resources are available, the flow is accepted without any need for signalling the core routers. Otherwise, the core routers will be signalled in an attempt to increase the aggregate's bandwidth. If this attempt succeeds, the flow will be admitted; otherwise, it will be rejected. This architecture benefits from the fact that signalling messages are only exchanged when the aggregate's bandwidth needs to be updated. Unfortunately, the decrease in signalling rate is accompanied by a decrease in resource utilisation.

In order to address the requirements of end-to-end QoS support without resource utilisation and scalability concerns, we developed a new architecture, Scalable Reservation-Based QoS (SRBQ) [4]. The underlying architecture of SRBQ is based on DiffServ, with the addition of signalling-based reservations subject to admission control. The network is partitioned into domains, consisting of core and edge routers; access domains contain also access routers. Flows are aggregated according to service classes, mapped to DiffServ PHBs (Per-Hop Behaviors), and packet classification and scheduling are based on the DS field of the packet headers. Besides Best-Effort (BE), SRBQ supports a Guaranteed Service (GS) class, providing strict QoS guarantees, and one or more Controlled Load (CL) classes, emulating lightly-loaded BE networks, based on the Assured Forwarding (AF) PHB. SRBQ's queuing model is compatible with DiffServ; the two models may coexist in the same network. The main scheduler is priority-based: the highest priority belongs to the GS class, which must be shaped by a token-bucket; below, there is a class for signalling, which must be rate-controlled; the CL class(es) come next, with optional rate-control; finally, at the bottom priority is the BE class.

In SRBQ, all nodes perform signalling and support the previously described queuing model. Access routers perform per-flow policing for the CL class and per-flow ingress shaping for the GS class. Edge routers perform aggregate policing and DSCP remarking. Core routers perform no policing.

Flows are subject to admission control, performed at every node. GS flows are characterised by token-buckets; CL flows are characterised by 3 rate water-marks, corresponding to different drop priorities. A scalable hop-by-hop signalling protocol was developed to perform unidirectional, sender initiated, soft-state reservations. Several techniques and algorithms have been developed aiming at the minimisation of the computational complexity and, therefore, the improvement of the signalling scalability. More specifically, a label switching mechanism, was developed to allow direct access to the reservation structures, avoiding expensive lookups in flow reservation tables. The labels are installed at reservation setup time, and all subsequent signalling messages use them. Moreover, a scalable implementation of expiration timers for soft reservations, with a complexity that is low and independent from the number of flows, was also developed. In terms of QoS guarantees, [5] has shown that our architecture is able to support strict and soft QoS guarantees to each flow, irrespectively of the behaviour of the other flows in the same and in different classes, with resource utilisation similar to that of IntServ, but increased scalability.

The rest of the paper is orgnanised as follows. In section 2 we compare both architectures in terms of QoS guarantees and resource utilisation, evaluating their relative merits and shortcomings, as well as their suitability to replace the reference RSVP/IntServ architecture, which suffers from scalability problems that disallow its usage in high traffic core networks. The results indicate that both the Scalable Reservation-Based QoS (SRBQ) [4] and the RSVP Reservation Aggregation (RSVPRAgg) [3] models provide adequate QoS levels and may, therefore, be used in place of RSVP/IntServ. Section 3 presents the most important conclusions and points out some topics for future work.

2 Comparison between SRBQ and RSVPRAgg

Both the SRBQ and RSRVPRAgg models aim at providing QoS levels comparable to RSVP/IntServ, but in a scalable manner. Both of them make use of flow aggregation in order to achieve scalability in packet classification and scheduling, using the DSCP field of the IP header to this end, in a DiffServ-like approach. The main differences between these architectures stem from the different approaches to signalling. In RSVPRAgg, reservations at the core are performed in an aggregate basis and their bandwidth is updated in bulk quantities, reducing the amount of state stored and the number of signalling messages processed. The unused bandwidth of all aggregates, however, adds up, leading to poor resource utilisation. Large bulk sizes aggravate this problem, but are needed for signalling to be scalable. Additionally, at edge routers of transit domains, where traffic is still very high, per-flow signalling (and, in some cases, classification and scheduling) is performed, imposing a scalability limitation. In SRBQ, the end-to-end character of reservation signalling is preserved, and scalability is achieved by using highly efficient techniques (like label switching and efficient timers). The amount of state stored is not really a problem [5], and resource usage is always optimal. Overall, the signalling processing overhead of these models is expected to be comparable.

Both architectures were implemented in the ns-2 network simulator. Although not scalable, an existing implementation of the RSVP/IntServ model is also used as a reference for QoS results. It is important to keep in mind that ns-2 has some limitations, the most significant of which is the inability to simulate and measure processing delays.

These models have some adjustable parameters. In RSVP and RSVPRAgg, the R parameter (average refresh period) used was the default of 30 s. The reservation expiration timer in SRBQ was chosen so that refreshes would be sent every 32 s, the closest value. Controlled Load (CL) flows in SRBQ are characterised by 3 rate water-marks; their target utilisation values were adjusted to 0.999, 1.0 and 3.0 times the bandwidth assigned to the CL class in order to ensure that, using the reservation parameters given below for each set of simulations, admission control would be performed based on the second water-mark. In the RSVPRAgg model there are two tunable parameters related to aggregate bandwidth management [6]: we used a value of 15 s for the bulk release delay timer, related to hysteresis, and a value of 5 s for the hold timer which prevents repeated failed attempts at increasing the bandwidth of a given aggregate. Simulations with the RSVPRAgg model were performed with two different bulk sizes: 300 kbps and 600 kbps. The admission control used in all models is parameter-based (PBAC).

A mapping between the QoS architectures needs to be performed for an accurate comparison. The aggregation regions of RSVPRAgg and the non-aggregated RSVP regions correspond to the core and access domains of SRBQ, respectively; the aggregators and deaggregators in RSVPRAgg correspond to edge routers in SRBQ. Given this mapping, the simulations used the same topology, depicted in figure 1. It includes 1 transit (TD) and 6 access (AD) domains. Each terminal simulates a set of terminals. The bandwidth of the connections in the transit domain, and in the interconnections between the transit and the access domains, is 10 Mbps. The propagation delay is 2 ms in the transit domain connections and 1 ms in the interconnections between the access and the transit domain.

The simulated scenario contains a class for signalling traffic, CL and BE classes. At each link, the bandwidth assigned to signalling is 1 Mbps. Note that, although this seems very high, the unused signalling bandwidth is used for BE traffic. The bandwidth assigned to the CL class is 7 Mbps. The remaining bandwidth, as well as unused CL and signalling bandwidth, is used for BE traffic.

Each terminal of the access domains on the left side generates a set of flows belonging to the CL and BE classes. The destination of each flow is randomly chosen among the terminals in the right side access domains. With 3 source and 3 destination edge routers in the core domain, the number of required end-to-end aggregates in the domain is 9. Traffic belonging to the CL class is a mixture of different types of flows: CBR, exponential on-off and Pareto on-off. These flows are initiated according to a Poisson process with a certain mean time interval between calls (MTBC), and flows' durations are exponentially distributed. Filler traffic in the BE class is composed by on-off Pareto and FTP flows.

All simulations presented in this paper are run for 5,400 simulation seconds, and data for the first 1,800 seconds is discarded. All values presented are an average of, at least, 5 simulation runs with different random seeds. The simulation results are presented in the next sub-sections.

2.1 QoS, utilisation and signalling

In the first set of simulations we used, in the CL class, only 64 kbps constant bit-rate (CBR) flows with a packet size of 500 bytes. This is done to completely avoid the unfairness originated by disproportionate rejection rates between flows of different types in different models. The average flow duration is 120 s, and the mean time between calls is adjusted to vary the offered load between 0.8 and 1.2 times the bandwidth allocated to the CL class at the core link. The results from this set of simulations are presented in figure 2.

In all models, the mean delay is not much higher than the sum of transmission and propagation delays (12.08 ms), meaning that the time spent in queues is low. Nevertheless it is lower in SRBQ, as is jitter (not shown). These results are probably due to the use of WFQ in RSVP and in RSVPRAgg outside the aggregation domain. In all models presented there are no losses, since the reserved bandwidth of the CBR flows is equal to the maximum required bandwidth, being sufficient to accommodate the accepted flows.

Regarding the utilisation of bandwidth allocated to the CL class, it is much higher in SRBQ (similar to RSVP) than in RSVPRAgg. In RSVPRAgg, the utilisation is very noticeably lower with a bulk size of 600 kbps than with a bulk size of 300 kbps. Notice that a bulk size of 600 kbps is less than a factor of 10 higher than the flow rates. This suggests that the use of larger bulk sizes in order to increase the scalability would lead

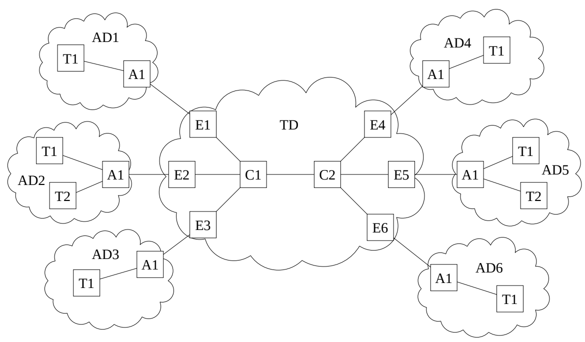
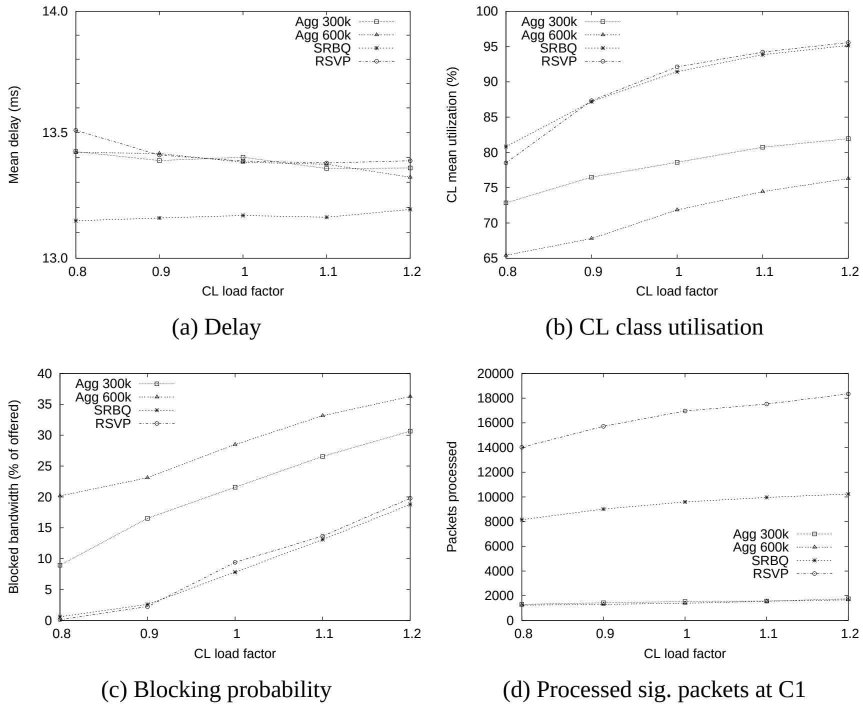


Figure 1. Simulation topology



(a) Delay

(b) CL class utilisation

(c) Blocking probability

(d) Processed sig. packets at C1

Figure 2. Performance of SRBQ, RSVPRAgg and RSVP with a single type of fbws

to very poor network resource utilisation. Corresponding to the lower utilisation figures, the blocked bandwidth in the RSVPRAgg model is higher than in SRBQ, and is higher when using larger bulk sizes. In SRBQ the blocked bandwidth figures are similar to regular RSVP, since in both end-to-end reservations are accepted up to the bandwidth reserved for the CL class, contrasting to the RSVPRAgg model in which end-to-end reservations are only accepted up to the reserved rate of the corresponding aggregate.

We also evaluated the number of signalling messages processed at core node 1 (see figure 1). This number is much lower in RSVPRAgg (about 1500 packets on average during the 3600 useful simulation seconds) than in SRBQ or RSVP (respectively, about 9000 and 16000 packets processed under the same conditions). This is an obvious result, since at interior nodes only aggregate messages are processed in RSVPRAgg. The almost twofold difference between SRBQ and RSVP is due to the fact that in RSVP both Path and Resv refreshes are needed. From the number of processed messages only, the RSVPRAgg model would be the clear winner in terms of signalling processing scalability. Keep in mind, though, that one big strength of the SRBQ model is the use of low complexity, highly efficient algorithms (labels, timers, etc.), which translate in much less CPU time used to process each message.

We performed a similar set of simulations using a mixture of different flow types (CBR, exponential on-off and Pareto on-off). The results of these simulations have shown that both models provide adequate QoS to the different flow types, except for

Type	Avg. rate (kbps)	Pkt. size (Bytes)	On (ms)	Off (ms)	Pk. rate (kbps)	Token Bucket		Watermarks (kbps)			MTBC (s)	Avg. dur. (s)
						R (kbps)	B (Bytes)	1	2	3		
cbr64cl	64	500				64	1500	64	64.064	72	11	120
exp1cl	48	500	200	200	96	64	15000	32	64	96	11	120
exp2cl	48	500	var	var	var	64	15000	32	64	96	11	120

Table 1. Flow characteristics for the isolation test

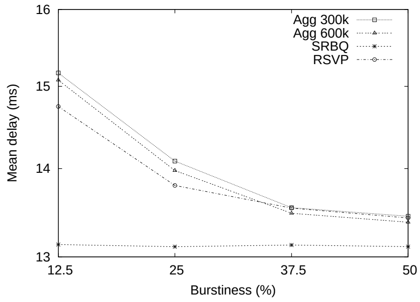
Pareto on-off flows which suffer higher delay and very significant losses in RSVPRAgg (about 10%, compared to about 0.003% in SRBQ). The higher delay is inflicted by WFQ outside the aggregation domain, while the packet losses occur mostly at the aggregator due to policing. Both of these problems stem from the fact that, having a heavy-tailed distribution with infinite variance, Pareto flows are not well suited for the token-bucket characterisation. SRBQ's rate water-marks characterisation is more tolerant of this type of flow. The per-class utilisation curves were similar to the previous ones, but lowered by about 10%.

2.2 Flow isolation

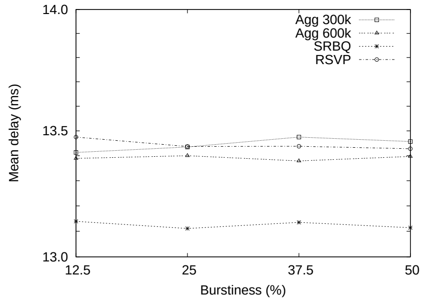
With this set of simulations we evaluate the behaviour of the different models in the presence of misbehaved flows, that is, flows that transmit at rates much higher than they reserved for considerable periods of time. We measure not only the quality of service received by these bursty flows but also the impact in the other flows. Three flow types were used in this test (table 1): (1) CBR flows (cbr64cl) that are considered well behaved flows; (2) on-off exponential flows (exp1cl) with a burstiness of 50% (average busy and idle times of 200 ms) and a peak rate of 96 kbps, that are considered nearly well behaved flows, since they send at a rate a little higher than the reserved; and (3) on-off exponential flows (exp2cl) with varying burstiness and peak rate that are considered misbehaved flows, since they send at a rate much larger than the reserved one for considerable periods of time. Their burstiness is variable, from 50% to 12.5%, varying their peak rate between 96 kbps (average busy and idle times of 200 ms) and 384 kbps (average busy and idle times of 50 ms and 350 ms, respectively). Notice that the sum of the average idle and busy times remains constant (400 ms), as does the average rate. It is the high mismatch between the requested rate and the peak transmission rate that turns exp2cl flows into misbehaved ones.

Each of the 4 transmitting terminals generates flows with these parameters, in the CL class. The total mean offered load at the core link is, therefore, 120% of the bandwidth allocated to CL, in terms of reserved rate.

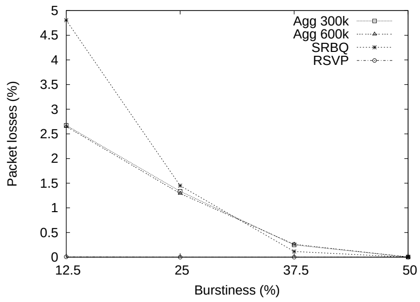
Figure 3 shows some results from this set of simulations. As observed, the mean delay for misbehaved flows is not affected by their burstiness in SRBQ (where it is mostly the sum of transmission and propagation delays), contrary to the other models. This is due to the fact that in the SRBQ model all CL data packets share the same queue. Notice however, that in this model, the traffic is policed before entering the domain. On the other hand, we may observe that in all models highly bursty flows have no noticeable impact in the delay of the low burstiness flows; the delay of CBR flows (not shown) is not affected either. The same kind of results are obtained for jitter, not shown here due to space limitations, but the difference is even higher: contrasting to the approximately



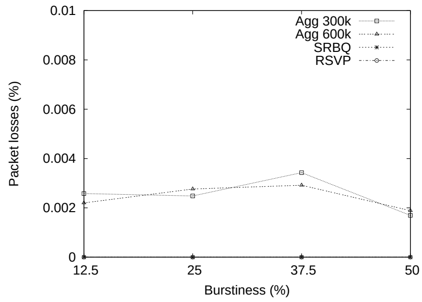
(a) Delay for high bustiness fbws



(b) Delay for low burstiness fbws



(c) Losses for high burstiness fbws



(d) Losses for low burstiness fbws

Figure 3. QoS results for low and high burstiness exponential fbws

constant value of 0.6 ms in SRBQ, jitter for the aggregation model varies from more than 10 ms with a burstiness of 12.5% to little more than 1 ms with a burstiness of 50%. Jitter for low burstiness flows is constant in all models, being about 0.6 ms in SRBQ and just above 1 ms in RSVPRAgg.

Misbehaved flows are heavily penalised in terms of packet losses. In SRBQ, packet losses for these flows almost reach 5% with a burstiness of 12.5%, while they are about 2.6% in RSVPRAgg. The high loss values are due to the fact that these flows transmit at rates much higher than they reserved for considerable periods of time. A relatively large bucket size absorbs these bursts up some level in RSVPRAgg, but in SRBQ the reservations for CL traffic have no bucket parameter, only 3 rate water-marks, of which even the third one is much lower (96 kbps) than the peak transmission rate (384 kbps). Clearly, these flows are violating their contracts, so measures must be taken against them to prevent QoS degradation for other flows. SRBQ is inflicting higher penalisations, better protecting other flows. Loss figures for other flows are not affected by the misbehaved ones. They are very low in RSVPRAgg and null in SRBQ for low burstiness exponential flows and are null in all models for CBR flows (not shown).

These results show that all models are able to provide adequate QoS levels to flows respecting their traffic contracts even in presence of misbehaved flows. The service of these flows is degraded in order to protect the well behaved flows: in SRBQ this degradation is only in terms of packet losses; in RSVPRAgg they are less penalised in terms of losses, but also have increased delay and jitter.

3 Conclusions and Future Work

In this paper we performed a comparative evaluation of two QoS architectures, SRBQ and RSVPRAgg, aimed at providing QoS levels similar to those provided by the well-known RSVP/IntServ architecture, but which are scalable enough for use in high traffic core networks. Several sets of simulations were performed for both models in order to evaluate different relevant aspects of the architectures (QoS parameters, flow isolation and resource utilisation). From the simulation results discussed in the previous section we may state that both the RSVPRAgg and SRBQ models provide adequate QoS levels and flow isolation in the CL class and are, therefore, real alternatives to RSVP/IntServ. Regarding scalability, SRBQ and RSVPRAgg are similar in terms of packet classification and scheduling procedures at the core, both using a scalable DiffServ-like approach. In terms of the raw number signalling messages processed at core nodes, RSVPRAgg wins by a wide margin. This is due to the fact that signalling scalability in SRBQ is not obtained by performing it at an aggregate level with bulk updates; instead, SRBQ makes use of highly efficient techniques and algorithms while keeping the end-to-end character of signalling. Due to these different approaches, utilisation figures are significantly better in SRBQ: there is no trade-off with signalling scalability like in RSVPRAgg, and resource utilisation is optimal under all conditions.

As future work, we plan to evaluate the possibilities for interoperability between SRBQ and other QoS architectures. We also plan to perform simulations with less synthetic, more realistic generation of flows, based on flow data collected in a real network. Finally, we expect to implement prototypes of both architectures in order to evaluate performance parameters which are not provided by ns-2, namely those regarding processing power required.

References

1. Braden, R., Clarck, D., Shenker, S.: Integrated Services in the Internet Architecture: an Overview. RFC 1633, Internet Engineering Task Force (1994)
2. Blake, S., Blake, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An Architecture for Differentiated Services. RFC 2475, Internet Engineering Task Force (1998)
3. Baker, F., Iturralde, C., Faucheur, F.L., Davie, B.: Aggregation of RSVP for IPv4 and IPv6 Reservations. RFC 3175, Internet Engineering Task Force (2001)
4. Prior, R., Sargento, S., Crisóstomo, S., Brandão, P.: End-to-end Quality of Service with Scalable Reservations. In: Proceedings of the 11th International Conference on Telecommunication System, Modeling and Analysis. (2003)
5. Prior, R., Sargento, S., Brandão, P., Crisóstomo, S.: Efficient Reservation-Based QoS Architecture. In: Interactive Multimedia on Next Generation Networks. Volume 2899 of Lecture Notes in Computer Science., Springer-Verlag (2003) 161–181
6. Prior, R., Sargento, S., Brandão, P., Crisóstomo, S.: Performance Evaluation of the RSVP Reservation Aggregation Model. In: Proceedings of the 7th IEEE International Conference on High Speed Networks and Multimedia Communications (to appear). (2004)