Description of the Flow-to-VC Mapping Control Module in DIANA´s RSVP over ATM Architecture

Egil Aarstad
Telenor Research & Development, P.O. Box 83, 2027 Kjeller, Norway
E-mail Egil.Aarstad@telenor.com

Lars Burgstahler, Martin Lorang
University of Stuttgart, IND, Pfaffenwaldring 47, 70569 Stuttgart, Germany
E-mail {lorang, burgstah}@ind.uni-stuttgart.de

Abstract

This paper describes the Flow to VC mapping control module that is embedded in the RSVP over ATM implementation of the ACTS project DIANA. As part of an edge router at the ingress to the ATM network, this module provides signalling translation from RSVP to ATM, handles the admission of flows to ATM VCs and controls an IP over ATM queueing discipline. It is thus an example of a traffic descriptor and QoS parameter based resource reservation that strictly guarantees end-to-end QoS. In order to address scalability issues, a concept of massive aggregation of flows to VCs and a dynamic bandwidth management is applied to reduce the control overhead induced by signalling and the maintenance of per-flow state information. The resulting architecture is compared with the IP based Scalable Reservation Protocol SRP and the Differentiated Services implementation SIMA.

1 Introduction

The performance degradation experienced in the Internet due to frequent bandwidth bottlenecks is a major obstacle for introducing multimedia services on a large scale. This situation has triggered much effort to develop and deploy network mechanisms enabling QoS support for bandwidth and delay sensitive services. One project within the ACTS Research Programme addressing this area is DIANA, Demonstration of IP and ATM Interworking for real-time Applications. The objective of DIANA is to develop, integrate, validate and demonstrate mechanisms for QoS support and interworking in a heterogeneous network environment. The project partners are Telscom, Flextel, EPFL, University of Stuttgart, ASPA, NetModule, Finsiel, Nokia, Swisscom and Telenor.

The DIANA project has investigated a scenario based on Integrated Services [1] where the Resource Reservation Protocol (RSVP) [2] is deployed over an ATM core network [11], as depicted in Fig. 1. This is not an unlikely scenario given the widespread use of ATM as a backbone technology. However, edge routers are required at the border of the ATM core network handling mapping between IP flows and ATM Virtual Circuits in the user plane and corresponding translation between RSVP and ATM signalling in the control plane. DIANA implements the edge router on the Flextel ATM switching platform, and the resulting network element is called the DIANA Integration Unit (DIU).

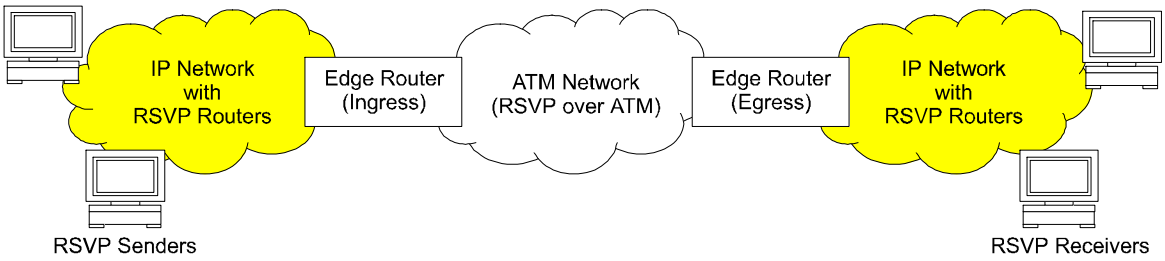

Fig. 1: RSVP over ATM: Edge routers enable QoS interworking between the RSVP/IP and ATM network domains

In the Integrated Services model reservation state maintenance together with packet classification and forwarding takes place on a per flow basis. Since this can lead to poor scalability in networks carrying numerous small flows, aggregation of QoS requests has been proposed [23, 24]. In DIANA, the IntServ/RSVP scalability issue has been addressed by developing aggregation functionality in the DIU allowing numerous small IP flows to share one ATM connection. Furthermore, an associated dynamic bandwidth management scheme for connections carrying aggregated traffic has been developed.

In addition to the RSVP over ATM implementation two other approaches for QoS support have been implemented on the Flextel switching platform: Simple Integrated Media Access (SIMA) [13] and Scalable Reservation Protocol (SRP) [14]. Both SIMA and SRP have been developed by partners in the DIANA project. By implementing RSVP over ATM, SIMA and SRP on the same platform, the project delivers an open experimental environment for investigating and comparing the strengths and weaknesses of each approach.

SIMA complies with the Differentiated Services (DiffServ) architecture [15] which defines a scalable framework for QoS support. At the border of a DiffServ domain packets are classified into a few Behaviour Aggregates and marked with a corresponding code-point in the IP packet header [16]. Inside the DiffServ domain each Behaviour Aggregate is forwarded on a per hop basis according to a specified Per Hop Behaviour (PHB). In this way service differentiation can be achieved without maintaining per flow state information. So far the IETF DiffServ Working Group has defined two types of PHB, Assured Forwarding [17] and Expedited Forwarding [18].

On the other hand, SIMA implements a Per Hop Behaviour called Dynamic Real-Time Non-Real-Time PHB Group [19]. A SIMA sender can emit packets marked as either real-time (rt) or non-real-time (nrt) packets. In access nodes, packets are classified and marked as belonging to one of several priority levels depending on the ratio between the momentary

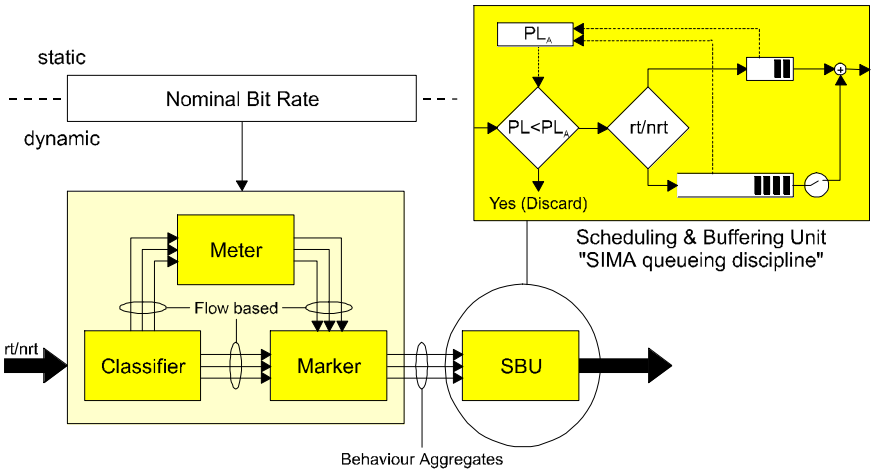

Fig. 2: The Simple Integrated Media Access (SIMA) architecture

sending rate and a contracted Nominal Bit Rate.

The SIMA Per Hop Behaviour is implemented in the Scheduling & Buffering Unit (SBU) in the core nodes, see Fig. 2. If a packet arrives at a core node with a Priority Level (PL) less than the threshold level PL_A, it will be discarded. Otherwise the packet will be queued, either in a real-time or non-real-time queue depending on the packet type. The SBU scheduling discipline is based on strict delay priority, thus the non-real-time queue will be served only when the real-time queue is empty. The priority threshold level PL_A changes dynamically as a function of the congestion level in the node, i.e. according to the buffer occupancy levels in the rt- and nrt-queues. If the congestion level increases, the priority level threshold is increased, thereby allowing a larger portion of packets to be discarded.

The Scalable Reservation Protocol is a novel approach for QoS support in IP networks providing dynamic resource reservation as in Integrated Services while maintaining scalability similar to Differentiated Services. The scalability is achieved by making reservations on a per hop basis and only for traffic aggregates.

Fig. 3 shows the main components of the SRP architecture. Simply explained, an SRP sender can mark packets either as Reserved (Resv), Request (Req) or Best-Effort (BE) packets. Reserved packets correspond to packets for which resources have been reserved in the routers. These packets are forwarded unchanged through the network. On the other hand, Request packets, which correspond to packets requesting resource reservation, will only be admitted and forwarded as Request packets if sufficient free resources are available in the routers. In this case the routers allocate resources to the Request packets. Otherwise the Request packets will be downgraded to Best-Effort. A receiver returns feedback to the sender about the amount of Reserved and Request packets received. From this information the sender estimates the amount of resources that have been established for the Request packets and will start emitting Reserved packets accordingly. In other words SRP provides a dynamic feedback control loop for establishing resource reservations in the network. The SRP implementation makes use of the Differentiated Services framework for packet marking and Per Hop Behaviour.

This paper focuses on the Flow to VC mapping control module embedded in the RSVP over ATM implementation. Its protocol and process architecture is explained in section 2. The flow mapping and signalling translation functions are implemented by a IP flow-to-VC Mapping module (F2VM), whereas operations related to ATM VCs management are implemented in a VC Set-up and Modification Module (VSMM). An overview of these modules is given in section 3. The associated traffic control and resource allocation functions are described in section 4. The dynamic bandwidth management scheme has been analysed with respect to trade-off between signalling load and bandwidth utilisation. The results of this analysis is presented in section 5. Finally, the RSVP over ATM architecture is compared with

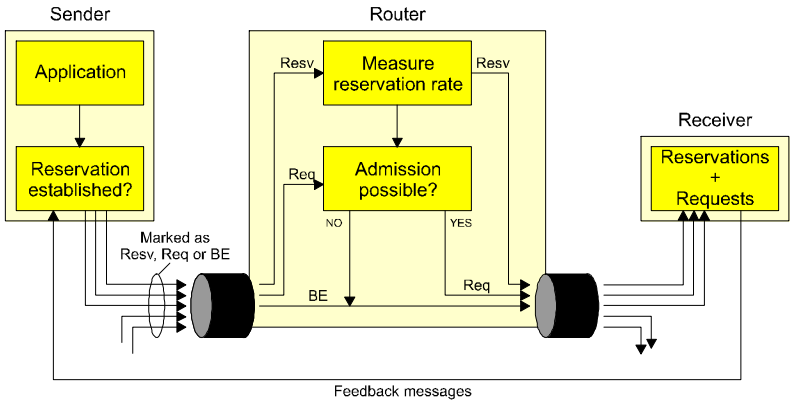

Fig. 3: The Scalable Reservation Protocol (SRP) architecture

the SIMA and SRP alternative.

2 Protocol and Process Architecture

RSVP over ATM architecture [2, 3, 4, 7, 8, 9, 10, 11, 12] is an example of a traffic descriptor and QoS parameter based resource reservation that strictly guarantees end-to-end QoS. In order to address scalability issues, DIANA realises a concept of massive aggregation of flows to VCs similar to [23, 24,] to reduce the control overhead induced by signalling and the maintenance of per-flow state information in the core network.

Therefore, DIANA´s RSVP over ATM traffic control architecture is implemented in a router that represents the border between an IP network and an ATM network, at a point where a sufficiently large number of IP flows can be aggregated and transferred on a common VC towards the egress of the ATM network. The router prototype implementation unifies RSVP signalling, Integrated Services traffic control, flow-to-VC mapping, CLIP IP over ATM address resolution [22] and UNI 4.0 [21] signalling in a working ensemble, see Fig. 4.

The controlling RSVP demon process starts interacting with ATM traffic control when traffic control related RSVP messages arrive. For this purpose, RSVP has been enhanced with an ATM specific Link Layer Dependent Adaptation Layer (LLDAL). DIANA´s LLDAL implementation is separated into a traffic control adaptation to RSVP and a demon process ATMTCD that controls most of the interworking tasks (address resolution, signalling translation) and traffic management functions (flow-to-VC mapping, dynamic bandwidth management, queue management & schedulers).

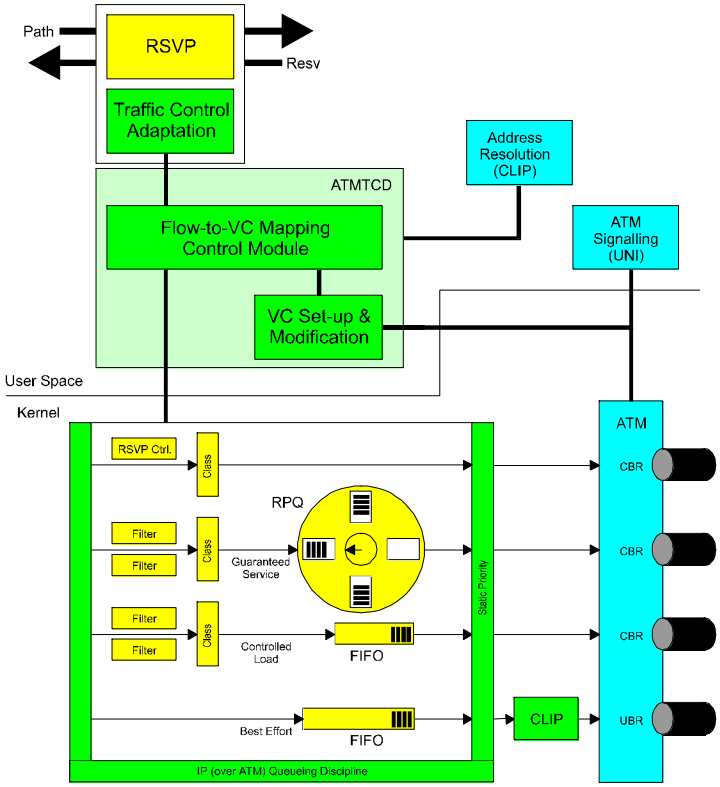

Fig. 4: DIANA´s RSVP over ATM traffic control architecture

3 Overview of the ATMTCD

The ATM traffic control demon process (ATMTCD) is not only in charge of handling communication with RSVP and the Linux kernel traffic control but also in controlling its subordinate modules Flow-to-VC Mapping Control Module (F2VM) and VC Set-up and Modification Module (VSMM) which carry out actions related with the mapping from an IP flow to an ATM VC and the dynamic set-up and modification of VCs following the rules of a dynamic threshold based bandwidth management scheme.

3.1 The Flow-to-VC Mapping Control Module

As its name suggests, the Flow-to-VC Mapping Control Module (F2VM) is a set of functions that carry out actions related with the mapping from an IP flow to an ATM VC. F2VM is an integral part of the ATMTCD. More specifically, it controls an architecture comprising the following interworking and traffic control functions:

Aggregation filters and flow records
The F2VM creates, modifies or deletes entries in its Flow-to-VC mapping table. Those entries are called F2VM records. Such a record collects information about a single VC and all the flows mapped to that VC. Of course only flows whose egress Integration Unit IP addresses match can be mapped to the same record. Moreover, an aggregation filter at the entry of each record ensures that only flow requests that also match the record´s aggregation criterion are mapped to the VC associated with that record.

ATM address resolution and signalling translation:
Whenever the F2VM cannot assign an incoming RSVP reservation request to one of the existing F2VM records, i.e. if none of the next RSVP hop fields and aggregation filters (see below) matches, the F2VM creates a new record, starts the address resolution procedure, and, if this succeeds, calculates the ATM traffic control parameters and finally triggers the ATM signalling demon to set-up a new ATM VC. When the ATM connection is ready, ATMTCD sends a confirmation back to the RSVP demon which then forwards the reservation message towards the sender.

Asynchronous ATM VC set-up and modification
In order to reduce the control overhead induced by signalling for a single flow with a possibly

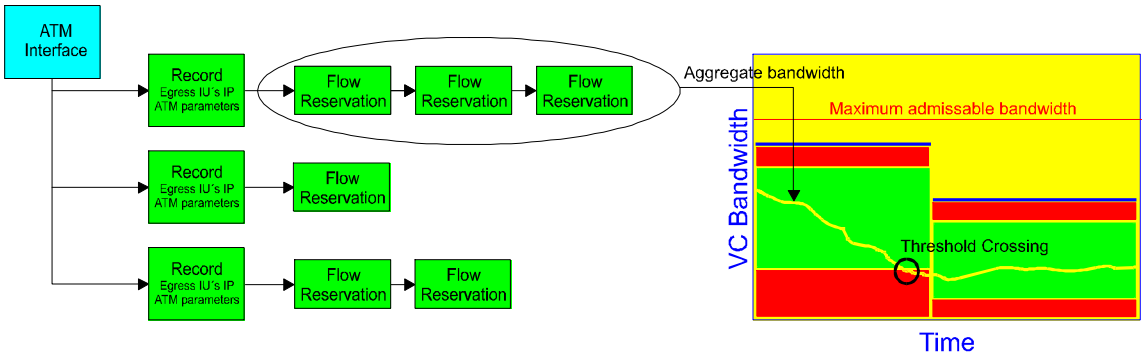

Fig 5: Flow-to-VC mapping and dynamic bandwidth management

low bandwidth demand and short lifetime, a dynamic bandwidth management scheme re-negotiates reserved ATM bandwidth [25] of a record only when the resource calculation for the flows mapped to that record yields a bandwidth that crosses a re-negotiation threshold. Those thresholds will be tuned in a way that avoids re-negotiation to an extent that can be justified by economic considerations in terms of over-allocated bandwidth and signalling processing overhead. This means that new flows can often be granted prior to re-negotiation by exploiting the safety margins. In this case, re-negotiation can be carried out asynchronously. As explained in more detail below, the VSMM is able to handle set-up and modification requests asynchronously and is thus a pre-requisite for the dynamic bandwidth management scheme. The behaviour of the system is sketched in Fig. 5.

Queueing and scheduling
The RSVP over ATM process ensemble introduced in Fig. 4 controls an IP over ATM priority queueing discipline that includes a RPQ scheduler [27] ensuring isolation of Guaranteed Service flows with different delay requirements if they are aggregated to the same VCs.

For Controlled Load service flows that do not include delay bound information, separate FIFO queues are used. In addition, a default queue for best-effort traffic is foreseen. Unlike the other classes, the default best-effort class does not filter traffic. Furthermore, it is the only „class“ that does not have a VC of its own but uses CLIP in a standard fashion. More details are given in section 4. The queueing discipline concept depicted in Fig. 4 is conformant to the generic Linux kernel traffic control framework [26].

As an additional feature, the queueing discipline assigns dedicated RSVP over ATM control CBR-VCs to protect RSVP signalling messages that would otherwise be forwarded via CLIP (UBR-VCs) from being dropped when congestion occurs.
The F2VM is in charge of installing the IP packet filters that allow to map packets to the classes that provide the forwarding behaviour specified through RSVP signalling.

3.2 The VC Setup and Modification Module

The VC Setup and Modification Module (VSMM) controls the setup, modification and release of ATM VCs. Within the ATMTCD, the VSMM interacts with the Flow-to-VC Mapping Module (F2VM) and the ATM API. Fig. 4 above shows how the VSMM is embedded in the Integration Unit’s architecture.

All functions that are directly related to VC manipulation work asynchronously to avoid that the ATMTCD has to wait for successful completion of VC setup, modification or release requests. Although USC´s RSVP demon does not support asynchronous operation with the LLDAL, asynchronous re-negotiation is still useful to increase an aggregate’s bandwidth once the calculated bandwidth enters a buffer zone without exceeding the actual VC bandwidth. For other requests can be accepted and the corresponding control processes started while previous requests are still being processed by remote switches, routers or the peer end system.

Whenever available, non-blocking API functions are used. Missing asynchronous API functionality is compensated using suitable UNIX interprocess communication mechanisms. In this way, the VSMM can often quickly respond to F2VM requests without awaiting the end of ATM processing. Regular polling ensures that ATM updates are taken into account later. As long as no updates occur, the ATMTCD can accomplish other tasks. Fig. 6 shows an example for a VC setup being done using non-blocking ATM API functions. As indicated, VSMM is able to handle several set-up requests quasi simultaneously. The final result, or more precisely, its impact on pending flow requests are communicated to the RSVP demon.

As child processes can use all the data already available in the parent process but work

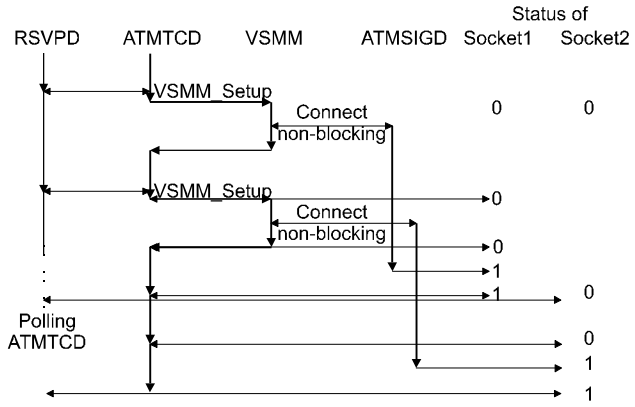

Fig. 6: ATM connection setup

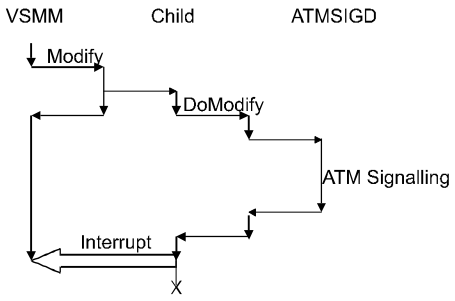

Fig. 7: ATM connection modification

independently and since the Linux ATM API does not monitor the result of an asynchronous VC modification, this method is used for re-negotiation. While the parent process can accept new requests, the child starts re-negotiation and waits for the result. After having delivered the result to the parent process, the child terminates itself. Fig. 7 illustrates the course of a re-negotiation. Several re-negotiations can be done quasi simultaneously, but for each one a new child process is launched.

4 Traffic Control and Resource Allocation Functions

IP network elements that support service guarantees have to be able to differentiate between flows and to forward packets in accordance with a traffic contract agreed upon between the network and the user. As a consequence, scarce resources, which means in this paper bandwidth and buffer, have to distributed to flows taking into account their specific requirements. Although the number of flows that require (strict) service guarantees does not increase as fast as the number of best-effort flows, this differentiation is likely not to scale well. Aggregation of flows to a single flow, e.g. as represented by an ATM VC, promises to solve that problem. However, when an aggregate is built up, resource allocation and scheduling has to make sure that the QoS requirements of each flow is respected.

Since the additional control overhead induced by scheduling, typically the ordering of packets in a sorted list, is significant, rotating priority queues (RPQ) and a corresponding resource allocation [27] have been integrated to DIANA´s IP over ATM queueing discipline to support Guaranteed Service. RPQ only approximates sorting of packets that is usually done based on transmission deadlines or similar time information and thus greatly simplifies scheduling, of course at the price of a reduced accuracy of its scheduling decisions and resource allocation. In this section, the performance of RPQ is investigated, partly based on the worst case analysis given in [27] and partly based on simulation. Besides, it is demonstrated how RPQ fits into DIANA´s flow aggregation concept.

The RPQ scheduler maintains P+1 priority ordered FIFO queues to handle P priorities. Those queues rotate at the end of a rotation interval Δ. This means that the queue with the previously highest priority is removed from the top and appended to the end of the priority sorted list of queues. Since packets with priority p, p = 1, ..., P are inserted into the queue with the current index p, i.e. nothing is inserted to the top queue anymore, and since resource allocation and policing ensures that none of the queues overflows, a deterministic, delay bounded service as required by Guaranteed Service [6] can be realised.

Provided the maximum traffic arrivals from a flow j is bounded by a right-continuous subadditive traffic constraint function $A_j(t)$, a given set of connections characterised by a tuple

consisting of this traffic constraint function and the specified delay bound $\{A_j(t),\ d_j\}$ is schedulable with a certain scheduling method if for all $t>0$ no packet exceeds its delay bound d_j.

In [27], schedulability conditions (1) have been derived based on a deterministic worst case analysis for each priority p (Below, p is used to number schedulability conditions). Ct denotes the service curve.

$$
\begin{aligned}
Ct &\geq \sum_{j\in C_1} A_j\left(t-d_1\right)+\max_{i\in C_q,q>1} s_i^{\max} && d_1 \leq t < d_2-\Delta \\
Ct &\geq \sum_{j\in C_1} A_j\left(t-d_1\right)+\sum_{q=2}^{P}\sum_{j\in C_q} A_j\left(t+\Delta-d_q\right)+\max_{i\in C_q,q>p} s_i^{\max} && d_p-\Delta \leq t < d_{p+1}-\Delta \\
& && 2 \leq p < P \\
Ct &\geq \sum_{j\in C_1} A_j(t-d_1)+\sum_{q=2}^{P}\sum_{j\in C_q} A_j\left(t+\Delta-d_q\right) && t \geq d_P-\Delta
\end{aligned}
\qquad (1)
$$

In the following, a fluid idealisation of those schedulability conditions is often used to derive guidelines for the use of RPQ, i.e. the terms

$$\max_{i\in C_q,q>1} s_i^{\max} \quad \text{and} \quad \max_{i\in C_q,q>p} s_i^{\max}$$

that account for the maximum packet transmission time of packets from flow i that cannot be pre-empted are ignored. Throughout this paper, traffic constraint functions of type (2) are used,

$$A_j(t)=\min(p_j t,\sigma_j+\rho_j t) \qquad (2)$$

where p_j and ρ_j denote rate parameters that can be interpreted as a the peak and mean rate of flow j, σ_j is a burst parameter. Hence, (2) specifies a traffic constraint function that is characterised by a finite peak rate p_j. This has some important implications:

With Guaranteed Service, the amount of data sent must not exceed $M+pt$ for all times t, with M denoting the maximum packet size and p the peak rate as indicated in the Integrated Services traffic descriptor TSpec. If one inserts p for p_j in (2), packet arrivals may exceed the traffic constraint function. As a consequence, (2) can only be interpreted as a fluid idealisation of the actual traffic constraint function in this case. Alternatively, p_j. could be set to the link rate but this impairs the efficiency of the RPQ resource allocation significantly since a source with lower p_j nestles more closely against a linear service curve which yields a lower delay.

The second implication of the choice (2) concerns the verification of the schedulability conditions (1). Since the schedulability conditions have to be met for all times t, the most critical time has to be found, i.e. the time at which delay is maximum. Due to the characteristics of the used traffic constraint functions, this is the time instant at which the slope of the sum of the traffic constraint functions A_j on the right side of (1) crosses the slope C of the service curve. If this does not happen during the time interval relevant for a schedulability condition, the left interval boundary is the most critical time if the slope of the sum is already smaller than C at that time or the right interval boundary in the opposite case, however, the latter condition is covered by the subsequent schedulability condition. Since the schedulability conditions are disjunct on the time axis, only one critical time per flow type has to be checked. For each of those critical times, a separate calculation is required since it is not clear in advance which of the critical times forms the bottleneck for the schedulability conditions as a whole.

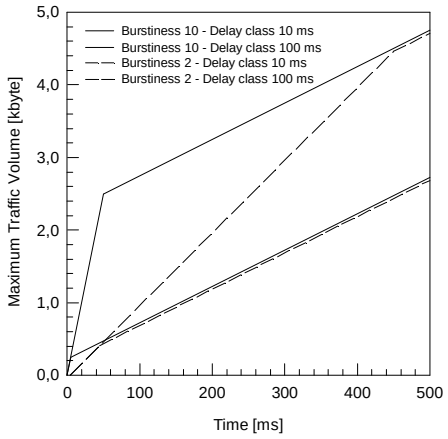

Fig. 8: Construction of traffic constraint functions for sources with variable burstiness

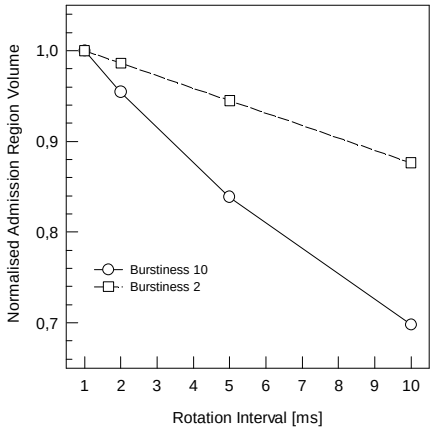

Fig. 9: Normalised volume of the admission region as a function of the rotation interval Δ

Finally, a traffic constraint function with finite peak rate mitigates the negative impact of a large rotation interval Δ on the required service rate *C*. This was a major concern and reason for an improvement of RPQ in [28]. The time instants at which the schedulability conditions are checked move to the right (as compared to a system with infinite peak rate, where there is only one critical time at the beginning of an interval) and t-Δ in (1) gets less sensitive to changes of Δ ($\Delta \leq d_1$). Fig. 9 demonstrates this effect with two source types that are constructed as depicted in Fig. 8 in a way that they differ in terms of their burstiness but their traffic constraint functions still converge. The exact parameters can be found in Table 1. Note

Table 1: Source parameters (t_q denotes the maximum duration of a burst)

Fig.	d_q/ms	p_q/C	ρ_q/C	t_q/ms	s_q/byte	No.	Remarks
8, 9	10	0.05	0.005	5	fluid	all	burstiness 10
	20	0.05	0.005	10	fluid	all	
	50	0.05	0.005	25	fluid	all	
	100	0.05	0.005	50	fluid	all	
8, 9	10	0.01	0.005	45	fluid	all	
	20	0.01	0.005	90	fluid	all	burstiness 2
	50	0.01	0.005	225	fluid	all	
	100	0.01	0.005	450	fluid	all	
10	10	0.05	0.005	5	fluid	60	load: 0.575 *C*
	20	0.05	0.005	10	fluid	19	max. load if only one source type
	50	0.05	0.005	25	fluid	22	is present: 0.3 *C*
	100	0.05	0.005	50	fluid	14	
11	20	0.01	0.005	90	fluid	76	load: 0.78 *C*
	50	0.01	0.005	100	fluid	40	max. load if only one source type
	100	0.01	0.005	110	fluid	40	is present: 0.61 *C*, 0.75 *C*, 0.95 *C*
12	10	0.05	0.005	5	125	60	ON-OFF sources with
	20	0.05	0.005	10	125	13	deterministic ON phase and
	50	0.05	0.005	25	125	23	and
	100	0.05	0.005	50	125	15	Erlang-10 distributed OFF phase
13	10	0.01	0.005	45	113	110	ON-OFF sources with
	20	0.01	0.005	90	113	11	deterministic ON phase and
	50	0.01	0.005	225	113	11	and
	100	0.01	0.005	450	113	33	Erlang-10 distributed OFF phase

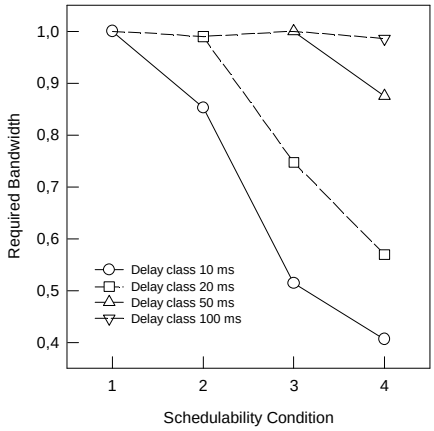

Fig. 10: Schedulability conditions for a balanced traffic mix (Δ = 1 ms)

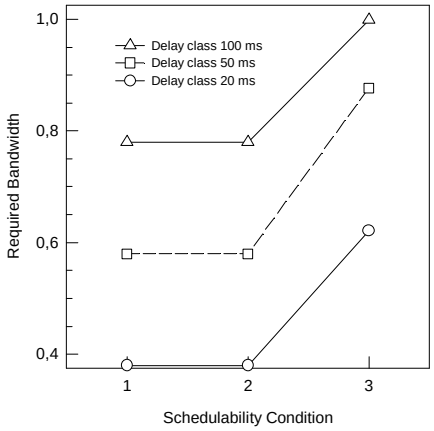

Fig. 11: Schedulability conditions for an unbalanced traffic mix (Δ = 10 ms)

that the admission region volume is normalised to the respective flow type maximum with a rotation interval of 1 ms.

In order to illustrate the efficiency of the RPQ system, various admission regions have been computed. Fig. 10 is obtained with a traffic mix that is composed of sources the critical times of which lie in different schedulability conditions. As a consequence, the resource demand is distributed almost equally among the different schedulability conditions. The admissible load reaches 0.575 whereas the presence of sources of a single type would approximately yield 0.03 (Table 1). Fig. 11 shows a counter-example where all critical times and thus their maximum demand of the sources involved lie in a single schedulability condition, in this case number 3. As a result, the maximum demand is not balanced and the lower schedulability conditions remain unsaturated.

Hence, if Δ approaches zero and if the packet terms in (1) are still ignored, the schedulability conditions (1) converge to

$$Ct \geq \sum_{j \in C_1} A_j\left(t - d_1\right) + \sum_{q=2}^{P} \sum_{j \in C_q} A_j\left(t - d_q\right) \tag{3}$$

Such a system would always outperform a FIFO based queueing system that separates flows with different delay requirements [29]. Although real systems may significantly deviate from this ideal case, the results shown in Fig. 10 show that a RPQ system that aggregates all Guaranteed Service flows with a reasonably small Δ (significantly smaller than d_1) is in most cases more efficient unless the delay requirements of a flow do not match the range or resolution (Δ) offered by the RPQ system and/or the maximum packet transmission time is large and - at the same time - the demand of the traffic mix under consideration cannot be distributed to the schedulability conditions in a balanced way.

Nevertheless, the results of simulative performance studies that are depicted in Fig. 12 and 13 show that a system with stochastic sources is usually far away from approaching the worst case scenario which the derivation of the schedulability conditions bases upon.

The results presented in Fig. 12 and 13 are obtained for traffic mixes that yield a maximum load among the source combinations that lie within the admission region calculated for the respective parameter set. This means that a guaranteed service with deterministic delay bounds and zero loss is only achievable with a generous reservation of resources. Liebeherr and Wrege propose an improvement to RPQ, the so-called RPQ+ [28], that uses a pair of queues per priority to improve the sorting of packets and to outperform a delay priority system even with

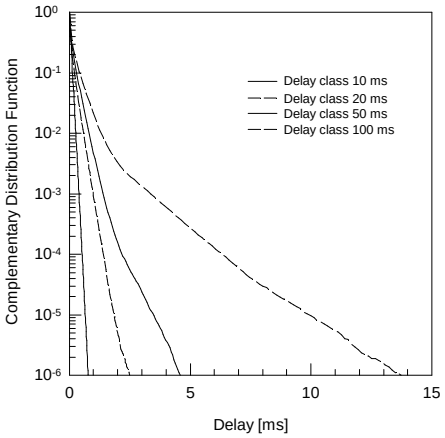

Fig. 12: Delay characteristics of RPQ with stochastic traffic sources (burstiness 10, Δ = 5 ms)

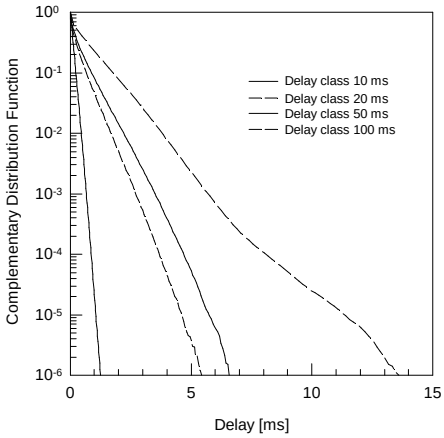

Fig. 13: Delay characteristics of RPQ with stochastic traffic sources (burstiness 2, Δ = 5 ms)

relatively high rotation intervals. Similar to RPQ, *P* schedulability conditions can be derived. However, in contrary to RPQ, those schedulability conditions are not disjunct, i.e. each schedulability condition covers the whole time axis. In addition, two different sums per schedulability condition have to be evaluated. Thus, 2*P* critical times per source type have to be checked providing the traffic constraint function introduced above is used - a non-negligible overhead.

A probably better strategy to make Guaranteed Service less expensive is to fill the unused ATM VC capacity with best-effort traffic but in a way that ensures that no queueing takes place at the ATM layer that would affect subsequent Guaranteed Service traffic. Whenever the RPQ head queue is empty, no Guaranteed Service traffic is in acute danger to offend its delay bound and best-effort traffic can be added (shaped to the VC or possibly VP bandwidth) until the next rotation occurs.

The Controlled Load service [5], the second Integrated Service, is less sensitive. Network elements implementing this service should be able to closely approximate the QoS that the same flow would receive from an unloaded service even if this is actually not the case. The F2VM currently only carries out simple calculation based on the mean rate as indicated in the RSVP traffic specification TSpec and configures a large FIFO queue with lower priority than the Guaranteed Service RPQ. If an effective bandwidth approach is employed instead, it has to be investigated whether aggregation criteria can be found that achieve a performance gain by grouping flows with similar characteristics and separate them from flows that would cause the admission region to significantly deviate from the ideal linear curve [30].

5 Potential of a Dynamic Bandwidth Management Scheme

When establishing a Virtual Connection for flow aggregation it can be difficult to select a suitable connection bandwidth because of unpredictable traffic demand. A small fixed VC bandwidth might be used up quickly, requiring a new connection to handle more flows. This will reduce the efficiency of the aggregation scheme. On the other hand assigning a large fixed VC bandwidth may lead to poor bandwidth utilisation.

The problems resulting from a fixed VC bandwidth can be minimised using mechanisms for VC bandwidth re-negotiation [25]. Modifying the connection bandwidth depending on the actual traffic demand eliminates the need for accurate bandwidth predictions. The VC

bandwidth can be dynamically changed as flows are added to or removed from the flow aggregate. However, to modify the VC bandwidth for every change in the aggregate flow bandwidth might impose a high signalling load. The implemented scheme seeks to avoid this by increasing and decreasing the VC bandwidth according to a threshold based algorithm. The algorithm is similar to one proposed for dynamically changing the bandwidth of Virtual Paths depending on the traffic demand at the VC level [20].

Fig. 5 illustrates how the dynamic bandwidth management algorithm operates on a Virtual Connection. A Buffer Zone is defined below the VC bandwidth and a Work Zone is defined below the Buffer Zone. The Work Zone is limited by upper and lower bandwidth thresholds. When the VC bandwidth is modified the threshold values will change accordingly.

As long as the aggregated flow bandwidth fluctuates within the Work Zone no changes are made to the VC bandwidth. However, if the aggregated flow bandwidth moves out of the Work Zone by crossing the upper or lower Work Zone threshold, bandwidth re-negotiation will be initiated. The new VC bandwidth is selected such that the aggregated flow bandwidth will be in the centre of the new Work Zone. Fig. 5 illustrates a reduction in the VC bandwidth.

The purpose of the Buffer Zone is to allow flows triggering bandwidth modification to be accepted without waiting for the re-negotiation process to be completed. The width of the Buffer Zone should be set equal to the largest flow bandwidth suitable for aggregation.

The width of the Work Zone is a very significant parameter influencing the dynamic operation of the bandwidth re-negotiation process. If a narrow Work Zone is selected, threshold crossings and corresponding bandwidth re-negotiation will occur frequently. However, the amount of bandwidth over-allocation will be low, since the VC bandwidth will track the aggregated flow bandwidth closely. On the other hand, selecting a large width for the Work Zone will reduce the frequency of threshold crossings and thereby reduce the number of re-negotiations. A larger Work Zone will of course increase the amount of bandwidth over-allocation. Thus there exists a trade-off between bandwidth over-allocation and signalling load.

To evaluate quantitatively the trade-off between VC bandwidth over-allocation and signalling load, a Markov chain model for the flow aggregation process *W* has been employed.. The model assumes homogeneous flow traffic generated by *M* on/off sources. Furthermore, it is assumed that the WZ thresholds are restricted to a finite number of equidistant values $WZ_0,\ldots,WZ_{N+1}$. Thus there will be only a finite number of possible Work Zones. Work Zone *i* has an upper and lower threshold equal to WZ_{i+1}, and WZ_{i-1} respectively. Fig. 14 shows the relation between Work Zone *i* and the aggregated flow state process.

As indicated in the figure Work Zone *i* covers *(i-1)K* to *(i+1)K-1* aggregated flows, so the work zone width is equal to *2K-1*. Entering Work Zone *i* takes place through state *iK*. Furthermore, Work Zone *i* can be left through state *(i-1)K* or *(i+1)K* which corresponds to entering Work Zone *i-1* or *i+1* respectively.

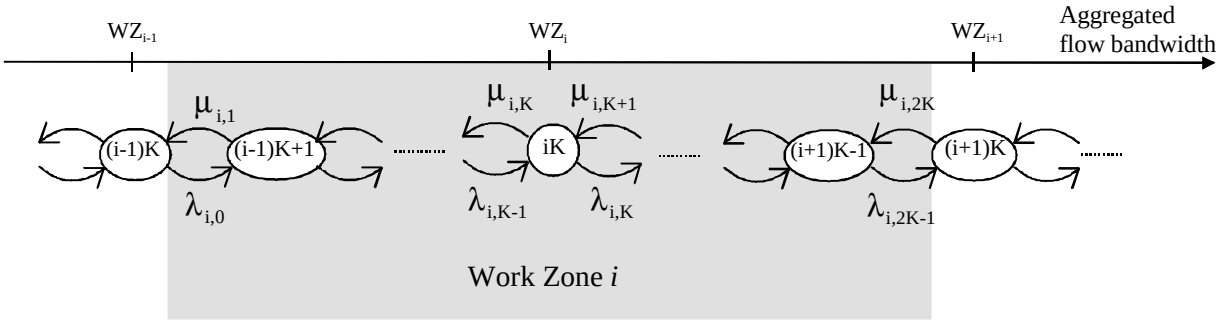

Fig. 14: The relation between Work Zone i and the aggregated flow state process. Work Zone i is limited by bandwidth thresholds WZ_{i-1}, and WZ_{i+1}

The model allows calculating the dynamic characteristics of the Work Zone process. For Work Zone *i* the probabilities of performing a transition into one of the adjacent work zones are calculated as absorption probabilities in an auxiliary Markov chain defined by making state *(i-1)K* and *(i+1)K* in Fig. 14 absorbing. The mean sojourn time ω_i in Work Zone *i* is calculated as the mean time until absorption in the auxiliary Markov chain. The stationary distribution $\{\pi_i\}$ of the Work Zone transition process is determined from the transition probabilities. Then the stationary probability σ_i of finding the Work Zone process in state *i* can be approximated as:

$$\sigma_i = \frac{\pi_i \omega_i}{\sum_k \pi_k \omega_k} \tag{4}$$

The average re-negotiation frequency is given by:

$$\bar{f} = \sum_i \sigma_i \frac{1}{\omega_i} = \frac{1}{\sum_k \pi_k \omega_k} \tag{5}$$

An expression for the average amount of bandwidth over-allocation can also be found. The analysis is treated in more detail in [31].

Fig. 15 compares the average re-negotiation rate and bandwidth over-allocation as a function of the Work Zone width. The Work Zone width is expressed in number of flows (*2K-1*). The average re-negotiation rate is expressed as a percentage of the average rate of state change in the aggregated flow process, whereas the bandwidth over-allocation is expressed as the percentage over-allocation relative to the aggregated flow bandwidth. In this example the Buffer Zone is set to zero.

As expected, the over-allocation increases linearly with the Work Zone width (the over-allocation is approximately equal to half of the Work Zone width). For a Work Zone width equal to 1 the VC bandwidth is re-negotiated for every change in the aggregated flow bandwidth (100%) while the over-allocation is zero. However, the re-negotiation frequency drops quickly as the Work Zone width increases. For example, with a Work Zone width equal to 5, the re-negotiation frequency has dropped to 13%. At this point the average VC bandwidth over-allocation has increased to 8%. This shows that the dynamic management scheme can potentially reduce the signalling load significantly for a moderate level of bandwidth-over-allocation. However, to determine the optimum Work Zone width the relative

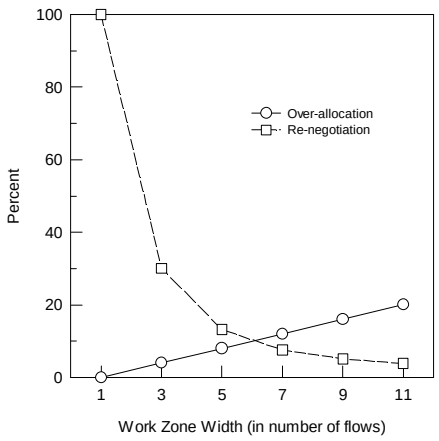

Fig. 15: Relative bandwidth over-allocation and re-negotiation frequency as a function of the Work Zone width

cost of bandwidth and signalling must be taken into account.

6 Conclusions

The ACTS project DIANA has implemented RSVP over ATM on an experimental IP routing and ATM switching platform. The platform can act as an edge router at the ingress of an ATM network to provide QoS integration between IP and ATM. This paper describes the Flow to VC mapping control module that is embedded in the RSVP over ATM implementation which can be seen as an example of a traffic descriptor and QoS parameter based resource reservation. In order to make this connection oriented approach scalable, the F2VM aggregates flows to a limited number of VCs. Simulations have been carried out to illustrate that the implemented traffic control functionality has the potential to ensure that the QoS experienced by an individual flow within an aggregate is not affected.

A threshold based scheme for dynamic management of bandwidth on VCs carrying aggregated flow traffic completes the presented architecture. This scheme uses bandwidth re-negotiation mechanisms to change the VC bandwidth according to the flow traffic demand. An analytical model for the scheme is presented. The model is used for analysing the trade-off between bandwidth over-allocation and re-negotiation frequency. Numerical results show that the average re-negotiation frequency drops rapidly for a small increase in bandwidth over-allocation. This implies that the dynamic management scheme can significantly reduce the signalling load for a moderate level of bandwidth over-allocation.

The RSVP over ATM prototype implementation allows the evaluation of several aspects like QoS translation, buffer management, scheduling, resource management and aggregation policies. These are all important research issues not only relating to RSVP and ATM but to many of the proposed approaches for QoS support. The results obtained can provide valuable information for network operators helping them to optimise their network architectures.

The following conclusions about the implemented QoS approaches can be drawn based on architectural considerations. Both the SIMA and SRP supports packet forwarding of traffic aggregates on a per-hop basis. This results in simple implementation and good scalability. Being a DiffServ implementation SIMA does not support reservation states in core nodes. Adequate QoS operation may therefore rely on proper Traffic Engineering, for example in combination with MPLS. With implicit signalling SRP offers better control of QoS and network resources than SIMA. However, the reservation scheme depends on the dynamic properties of the sender-receiver feedback loop, which makes this architecture best suited for non-real-time data services requiring a certain throughput.

The RSVP over ATM architecture offers strict service guarantees. Using ATM at the link layer establishes a close connection between routing and resource reservations. This allows very good control of QoS. The architecture applies sophisticated traffic control methods and is therefore more complex to implement. Flow aggregation should be exploited to minimize the impact of poor scalability. RSVP over ATM is ideally suited for real-time services with strict delay and loss requirements.

References

[1] Braden, R., Clark, D., and S. Shenker, *Integrated Services in the Internet Architecture: an Overview.* IETF Request for Comments, RFC 1633, June 1994.

[2] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog and S. Jamin, *Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification.* IETF Request for Comments, RFC 2205, September 1997.

[3] R. Braden and L. Zhang, *Resource ReSerVation Protocol (RSVP) - Version 1 Message Processing Rules.* IETF Request for Comments, RFC 2209, September 1997.

[4] J. Wroclawski, *The Use of RSVP with Integrated Services.* IETF Request for Comments, RFC 2210, September 1997.
[5] J. Wroclawski, *Specification of the Controlled-Load Network Element Service.* IETF Request for Comments, RFC 2211, September 1997.
[6] S. Shenker, C. Partridge and R. Guerin, *Specification of Guaranteed Quality of Service.* IETF Request for Comments, RFC 2212, September 1997.
[7] S. Shenker and J. Wroclawski, *General Characterization Parameters for Integrated Service Network Elements.* IETF Request for Comments, RFC 2215, September 1997.
[8] L. Berger *RSVP over ATM Implementation Guidelines.* IETF Request for Comments, RFC 2379, August 1998.
[9] L. Berger, *RSVP over ATM Implementation Requirements.* IETF Request for Comments, RFC 2380, August 1998.
[10] M. W. Garrett and M. Borden, *Interoperation of Controlled-Load Service and Guaranteed Service with ATM*. IETF Request for Comments, RFC 2381, August 1998.
[11] E. Crawley, *A Framework for Integrated Services and RSVP over ATM*. IETF Request for Comments, RFC 2382, August 1998.
[12] A. Demirtjis et al., *RSVP and ATM Signalling*. ATM Forum Contribution 96-0258, January 1996.
[13] Nokia Research Center, *SIMA project web site*. <http://www-nrc.nokia.com/sima/>, valid October 1999.
[14] T. Ferrari, W. Almesberger and J.-Y. Le Boudec, „SRP: a Scalable Reservation Protocol for the Internet“, in *Proceedings of IWQoS´98*, pp. 107-116, Napa, CA, May 1998.
[15] S. Blake et al., *An Architecture for Differentiated Services*. IETF Request for Comments, RFC 2475, December 1998.
[16] K. Nichols, S. Blake, F. Baker, D. Black, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*. IETF Request for Comments, RFC 2474, December 1998.
[17] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, *Assured Forwarding PHB Group*. IETF Request for Comments, RFC 2597, June 1999.
[18] V. Jacobson, K. Nichols, K. Poduri, *An Expedited Forwarding PHB Group*. IETF Request for Comments, RFC 2598, June 1999.
[19] M. Loukola, K. Kilkki and J. Ruutu, *Dynamic RT/NRT PHB*. Work in Progress, IETF Differentiated Services Working Group, <draft-loukola-dynamic-00.txt>, November 1998.
[20] M. Verdier, D. Griffin and P. Georgatsos, „Dynamic Bandwidth Management in ATM Networks“, in *Proceedings of the 4th EUNICE Open Summer School on Network Management and Operation*, Munich, Germany, August 31 - September 3, 1998.
[21] ATM Forum, *ATM User-Network Interface (UNI) Signalling Specification - Version 4.0.* ATM Forum, AF-SIG 0061.000, July 1996.
[22] M. Laubach, *Classical IP and ARP over ATM*. IETF Request for Comments, RFC 1577, January 1994.
[23] S. Berson and S. Vincent, *Aggregation of Internet Integrated Services State.* Work in Progress, IETF Integrated Services Working Group, <draft-berson-classy-approach-01.ps >, November 1997.
[24] R. Guerin, S. Blake and S. Herzog, *Aggregating RSVP-based QoS Requests.* Work in Progress, IETF Integrated Services Working Group, <draft-guerin-aggreg-rsvp-00.txt>, November 1997.
[25] ITU-T, *Peak cell rate modification by the connection owner.* ITU-T Recommendation Q.2963.1, July 1996.
[26] W. Almesberger, „Linux Network Traffic Control - Implementation Overview“, in *Proceedings of the 5th Annual Linux Expo*, pp. 153-164, Raleigh, NC, May 1999.
[27] J. Liebeherr, D. E. Wrege and D. Ferrari, „Exact Admission Control for Networks with a Bounded Delay Service“, in *IEEE/ACM Transactions on Networking*, Vol. 4, No. 6, pp. 885-901, December 1996.
[28] J. Liebeherr and D. E. Wrege, „Priority Queue Schedulers with Approximate Sorting in Output-Buffered Switches“, in *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 6, pp. 1127-1144, June 1999.
[29] G. Urvoy, Y. Dallery and G. Hébuterne, „CAC Procedures For Delay-constrained VBR Sources“, in *Sixth IFIP Workshop on Performance Evaluation of ATM Networks (IFIP ATM´98), Participants Proceedings, Part2 - Research Papers*, pp. 39/1-39/10, Ilkley, West Yorkshire, U.K., July 1998.
[30] J. Y. Hui, „Resource Allocation for Broadband Networks“, in *IEEE Journal on Selected Areas in Communications*, Vol. 6, No. 9, pp. 1598-1608, December 1988.
[31] L. Burgstahler (ed.), *Prototype Implementation of a RSVP/IP and ATM Network Integration Unit*. ACTS project DIANA, Deliverable 3, August 1999.