31
Sharing Resources through Advance Reservation Agents

Olov Schelén and Stephen Pink
Dept. of Computer Science and Electrical Engineering
Luleå University of Technology, S - 971 87 Luleå, Sweden
email: {olov, steve}@cdt.luth.se, *tel:+46 920 91428, fax: +46 920 72801*

Abstract

We propose an architecture where clients can make advance reservations through agents responsible for advance admission control. The agents allocate resources in the routers just before they are needed for packet forwarding. In this paper we show that network resources can be shared between immediate and advance reservations without being pre-partitioned. The admission control schemes for immediate and advance reservations still operate with little interaction.

Admission control decisions for immediate reservations use information about resources to be allocated for advance reservations in the near future. An important parameter in the admission control algorithm is the so called lookahead time, i.e., the point at which we actually start making resources available for approaching advance reservations by rejecting immediate requests. In our model, preemption of immediate reservations is made in cases where the admission control cannot make resources available through rejection of immediate requests. The risk of preemption can be varied by changing the lookahead time when making immediate admission control.

We explore, with simulations, the effects of providing advance reservations according to this model. The results show the cost in terms of resource utilization, rejection probability and preemption probability.

Keywords

QoS, advance reservations, resource sharing, agents, admission control

1 INTRODUCTION

Real-time applications such as audio and video conferencing may need resource reservations in the network to perform well. However, many of the current resource reservation proposals allow only immediate reservations, i.e., reservations at the time the session begins, while real-time events often are

*This work was supported by a grant from the Center for Distance Spanning Technology (CDT), Luleå, Sweden

Building QoS into Distributed Systems A. Campbell & K. Nahrstedt (Eds.)

scheduled and advertized in advance. Clients making immediate reservations for scheduled events will have to judge the appropriate time for making reservation requests in order to be accepted in due time. When resources are scarce, clients may start making reservations earlier than needed, leading to suboptimal scheduling of network resources. By offering reservations in advance, these problems can be alleviated.

In (Schelén & Pink 1997), we present an architecture where admission control for advance reservations is performed by *agents* on behalf of the routers in their routing domains. By participating in a link-state routing protocol, the agents know the topology and the static link capacities of their domains and can therefore select suitable routes and reserve resources over the links in advance. When the reserved resources are needed for packet forwarding, the reservation agents will allocate resources in the routers along the selected path and then periodically refresh those resource allocations through the duration of the reservation.

Clients can make end-to-end reservations by contacting any agent. Requests involving several routing domains are handled by cooperation between the involved reservation agents. An advantage with an agent-based approach is that advance reservations can be provided without having to maintain reservation state in the routers through periods when the reservations have no effect on packet forwarding. Furthermore, the agent-driven approach does not require the reserving endpoints to be present until the session for which resources have been reserved is starting. Reservations can also be made for remote locations, thereby supporting third party reservations and nomadic computing, i.e., reservations for places where the host will be moving in the future.

In this paper we will show that network resources can be allocated through advance reservation agents. In particular we will show that pre-partitioning of resources is not necessary and that admission control for immediate and advance reservations still can be performed with little interaction.

Admission control for both immediate and advance reservations could be performed by reservation agents provided that the admission decision is based on *traffic specifications* given by the clients and static link resources only. This is the case for admission control schemes offering some kind of performance bound (Firoiu, Kurose & Towsley 1996) (Shenker, Partridge & Guerin 1997), enforced by the packet scheduler (e.g., WFQ or EDF). However, there are other proposals where immediate admission is based on *measurements* of the used resources rather than the aggregate of the requested resources (Jamin, Danzig, Shenker & Zhang 1995) (Wroclawski 1996). We believe that the signaling overhead for having traffic measurement based admission control in agents may be prohibitive, and can be better served by having admission control in the routers (where traffic measurement can be done directly). Also, to facilitate the integration of advance reservation support with current technologies for immediate reservations, advance reservations should be managed separately from immediate reservations.

One way of separating the admission control for immediate and advance reservations would be to partition the resources so that one part is available for immediate reservations and another part for advance reservations However, this would result in two logically separate networks with bad utilization due to the problem of partitioning resources according to current demand. Therefore, we have chosen to share resources between immediate and advance admission control. To allocate resources safely we have extended the immediate admission control to consider the near future concerning the resources that will be allocated for advance reservations. There are three major consequences of these results. First, the resources can be managed efficiently without suffering from badly chosen resource partitions. Second, the immediate admission control scheme can be kept fast and simple by not having to consider advance reservations in the far future. Third, we have the architectural freedom of locating the admission control for immediate and advance reservations at separate places in the network without suffering undue communication overhead. Agents for advance reservations inform about resource allocations shortly before they are scheduled to happen.

In our model, resources for advance reservations will be made available by *rejecting* immediate requests for a period before resources are to be allocated, or ultimately by *preempting* service for flows that were granted immediate access. We show, with simulations, the effects of advance reservations in terms of resource utilization, rejection probability and preemption probability. We have simulated admission control for a single link.

2 SERVICE MODEL

- Durations must be specified for advance reservations. A duration includes the start time and the finish time for the requested service.
- Immediate reservations do not specify durations. Such requests are serviced immediately (if possible) and can be preempted later by flows that reserved in advance.
- Preemption means that a flow loses its service quality; however, it will still be serviced at a best-effort level.
- In order to manage resources safely there may be limits imposed on minimum and maximum bookahead times for advance reservations.
- The admission control for advance reservations may support finding the earliest point for which a requested service can be granted.

We have chosen *not* to include a service for immediate requests that are *guaranteed* never to be preempted. One reason is that we aim at keeping the preemption probability low, so that the user need not worry about preemption when asking for admission. Another more fundamental reason is that indefinite non-preemptible service would block resources indefinitely and no advance reservations of those resources could therefore be made. Furthermore, resources that are reserved in advance for some time in the far future cannot be given to indefinite sessions in the meantime unless we have the option of preempting those sessions when resources are needed. To provide indefinite sessions, it is necessary to set aside resources for this service exclusively. In the Tenet suite (Ferrari, Gupta & Ventre 1995), resources are partitioned to support advance reservations and non-preemptible immediate sessions simultaneously. However, there will be some fragmentation due to it being hard to maintain the partitioning such that the the demand for the two different services are met at each instant. We could support indefinite sessions by using such a partition in our architecture as well. However, in this paper, we will investigate whether or not we can avoid some problems with resource fragmentation by sharing resources between immediate and advance reservations and instead allow for statistically rare cases of preemption.

3 RESOURCE SHARING BETWEEN IMMEDIATE AND ADVANCE RESERVATIONS

When immediate reservations and advance reservations share resources and the duration of each immediate reservation is unknown, there is a risk of over-allocation that can be made less likely with preemption. The probability of preemption can be controlled by either monitoring immediate requests when admitting advance reservations, or vice versa, by monitoring advance reservations when admitting immediate requests.

When advance reservation requests are made far in advance, the information obtained from monitoring present traffic will not be useful for advance admission control (Degermark, Köhler, Pink & Schelén 1995). Most of the active flows would have finished at the crucial time anyway. It is only when advance reservations are requested with very short notice that it would be valuable to know about present traffic. Since advance reservations are provided to meet the demand for scheduled sessions primarily, we believe that most reservations will be requested on a notice much longer than the average session length. We may even impose this in the reservation agents by having a minimum bookahead time for advance reservations.

On the contrary, by monitoring advance reservations for the near future when admitting immediate requests, we can to some extent control the risk of preemption. If there are insufficient resources available in the near future, the request could be rejected instead of serviced. It is up to the provider to decide whether rejection or preemption is preferable in these cases. Rejecting immediate requests because the resources are reserved in the near future means that the overall utilization will go down to make room for the advance reservations. We denote the pre-allocation time, i.e., the time for starting to set aside resources for advance reservations as the ***lookahead time*** into the advance reservation state. Setting the lookahead time is an internal issue to the admission control scheme. We denote the time from making an advance reservation until the resources are to be available as the ***bookahead time.*** The bookahead time is set by the clients asking for admission and varies from request to request, possibly within some limits imposed by admission control.

4 SIMULATIONS

The objective of our simulations is to explore the effects of allowing reservations in advance. The performance measures are ***link utilization***, ***rejection probability***, and ***preemption probability.*** We vary the ***fraction*** of the total link resources used for ***advance reservations*** and the ***lookahead time*** into advance reservation state at admission control for immediate requests. Admission decisions are based on reservation specifications (e.g., traffic specifications) provided by the clients. For simplicity we start with peak rate resource allocation, using the requested bandwidth as the only parameter. This study is for admission control over a single link only.

4.1 Simulation parameters

- The duration of immediate requests, i.e., the call hold time, is exponentially distributed with a mean of 300 time units.
- The duration of advance requests, is also exponentially distributed with a mean of 300 time units.
- The bookahead time for advance requests, i.e., how far in advance the advance requests are made, is exponentially distributed with a mean of 20 000 time units, where the tail is cut at 60 000 units (i.e., there is a maximum bookahead time).
- The total link capacity is 45 000 resource units.
- The resources asked for in each request are uniformly distributed in [1..3000] resource units.
- The inter-arrival time of admission requests is exponentially distributed with a mean of 11.76 time units. This number was selected to obtain a reasonable rejection probability, where around 6% of the requests are rejected. The average offered load is 85% of total capacity.
- When preemption is performed, the immediate request with the largest requested capacity is picked as a victim (which would encourage making requests in advance when large capacity is needed).

The simulations are run many times for each parameter set, with different seeds for the random variables. The 90% confidential intervals are computed and presented as error bars in the plots (although the confidence is often so tight that the error bars appear as dots).

4.2 Advance reservation requests

We vary the fraction of the admission requests that are made in advance between zero and one. The upper bound on resources available for advance reservations equals the total link capacity, resulting in advance reservations being seldomly rejected when a small fraction of the requests are made in advance (Figure 1).

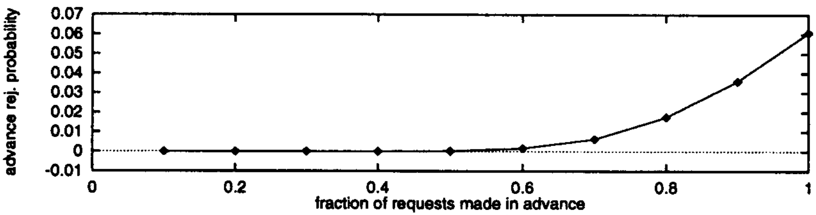

Figure 1 Rejection probability for advance requests

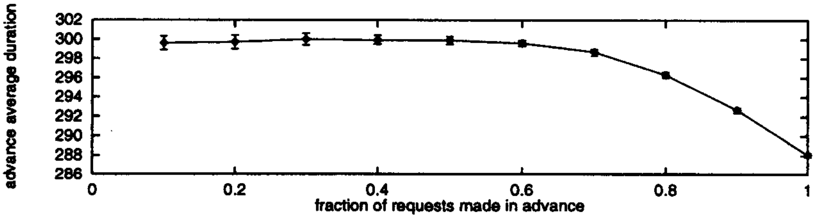

Figure 2 Average duration for accepted advance requests

Figure 2 shows that the average duration of the granted requests goes down as there is a higher fraction of advance reservations. When resources are scarce, advance reservations will result in some fragmentation in time, and there is a greater chance that requests for short durations are accepted. This effect does not occur for immediate requests since admission is made without considering durations (in fact, durations are unknown).

A similar effect can be observed concerning the average bandwidth given to the granted requests. If the offered load is increased, the average bandwidth for granted requests will go down as the rejection probability goes up. However, for advance reservations the rejection probability also depends on the bookahead time. Therefore, users requesting large bandwidth have a stronger motive for reserving well in advance to avoid rejection.

4.3 Total link utilization

Figure 3 shows that the total link utilization goes down as the fraction of advance reservations increases. There are two reasons for this. First, we must start allocating resources ahead of the starting point of the durations for advance reservations in order to avoid having to preempt immediate requests. The result depends on how far in advance we start setting-aside resources, i.e., how far we look ahead into advance reservation state when deciding if there is capacity available for an immediate reservation. Second, fragmentation in time is introduced when advance reservations are allowed. There is little chance that other advance requests ask for the small fragments that may be available between already granted advance reservations. Consequently, a decrease in utilization can be observed as the rate of advance reservations is increased, even though there is no pre-allocation, i.e., the lookahead is zero and preemption is the only means for making resources available.

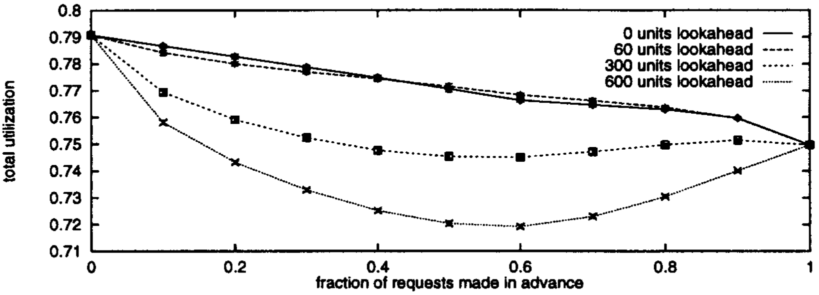

Figure 3 Total link utilization

This may sound like bad news for providing advance reservations, but in fact similar drops in total utilization would be observed as soon as there are scheduled events introduced in the network, even though users were to make only immediate reservations. Users would still be likely to allocate resources before events begin to make sure that they have resources allocated in time. The only difference is that the control is with the clients instead of with the network. The degree of over-allocation in time would be a consequence of social behavior in an environment with scarce resources. In some situations we can expect that effective utilization drops in an uncontrolled way.

Measurement-based immediate admission control could alleviate the problems of low utilization, but since we cannot get any useful measurements until the traffic starts, we must choose between staying with the traffic specification given by the client or gradually losing the resource allocation as the measurement procedure finds that there is no traffic. When there are scheduled events, advance reservations are likely to improve the service and the total utilization by ensuring that resources are allocated at an appropriate time.

4.4 Rejection and preemption probability

Figure 4 shows that the rejection probability for immediate reservations increases as the fraction of advance reservations is increased. However, the change in rejection probability for immediate requests is primarily an effect of picking the victims among a smaller set of immediate requests as the rate of advance reservations increases. If we instead monitor the total rejection probability (Figure 5) over both immediate and advance requests, we get much lower risk of rejection (note that the scales are different). Perhaps this is a better reflection of overall user satisfaction.

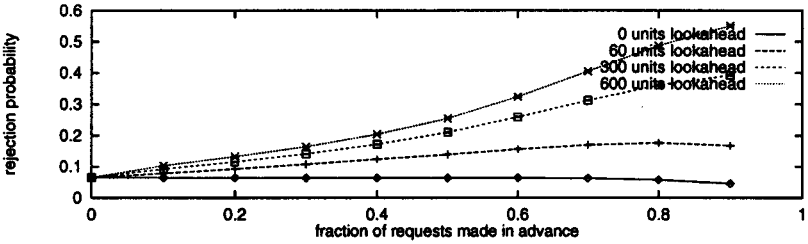

Figure 4 Rejection probability for immediate requests

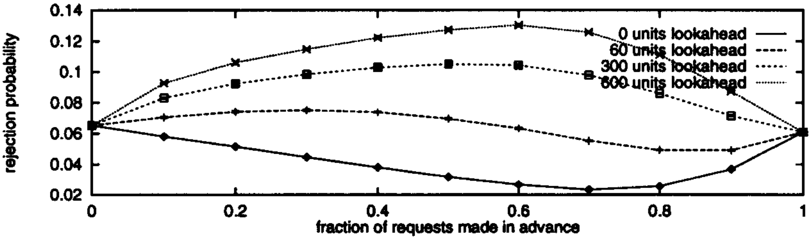

Figure 5 Total rejection probability (for immediate and advance requests)

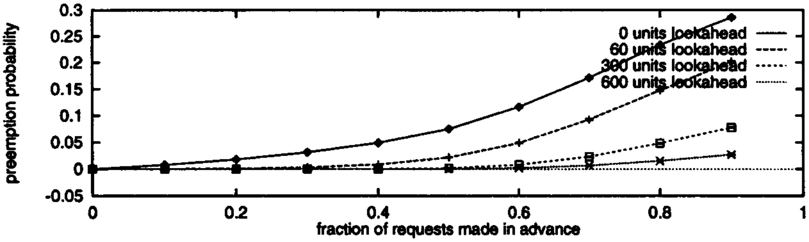

Figure 6 Preemption probability for immediate requests

We can see that the rejection probability is about the same when we have only immediate requests as it is when we have only advance requests, i.e., around 6 %. The decrease in total utilization (Figure 3) is explained by the fact that the average duration for accepted requests goes down (due to fragmentation in time) with increasing demand for advance reservations (Figure 2).

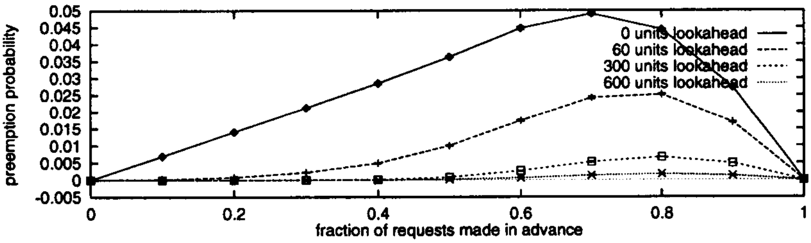

Figure 7 Total preemption probability (for immediate and advance requests)

Figure 6 shows that the preemption probability increases as the fraction of advance reservations is increased. Again, part of the explanation is that all preemption victims are picked within the immediate requests and as the fraction of immediate requests goes down we get a higher risk of preemption for this service class. In Figure 7, we show the preemption probability over the total number of served reservations.

It is clear that increasing the lookahead time makes admission control more conservative, resulting in lower preemption probability at the cost of lower utilization and higher rejection probability. By decreasing the lookahead time, we allow the system to be more overbooked resulting in higher preemption probability. The fact that the system is allowed to be overbooked explains why the total rejection probability (Figure 5) goes down when we use zero lookahead and there is a suitable fraction of advance reservations.

4.5 Lookahead time

Given a desirable target for preemption probability, the lookahead time can be conservatively computed by considering the bandwidth that must be allocated for each advance reservation and the amount of bandwidth that is returned per time unit by immediate reservations ending. However, at reasonable load it is likely that there are some unallocated resources that should be considered as well. In our simulations, between 75% and 79% of the resources are allocated on average, which explains why it is possible to allocate resources for advance reservations with a total preemption probability below 5%, even when using zero lookahead.

In previous examples we kept the lookahead times constant as the fraction of advance reservations was increased. We have also tried to use exponentially distributed lookahead times, but we found a strict performance degradation compared with constant lookahead time. If there are problems in meeting the requirements for high utilization and low preemption probability, we may be forced to have a limit imposed on resources available for reservations in advance. As an example, we now fix the advance request rate to 40% of the total number of requests (i.e., the average is bounded and there is no firm bound) and we plot the performance measures as functions of the lookahead time

Figures 8, 9, and 10, show the effects. It is clear that the preemption probability falls steeply as the lookahead is increased. In most systems, like cellular phone systems, preemption is considered bad and should be avoided. In a packet switched network, preemption is not as bad since the packets are still served best-effort. We believe, however, that the relative weight of having low preemption is still high. Provided that we have the average duration for resource reservations (assuming exponentially distributed session lengths), the fraction of advance reservations, and the individual weights (set by the provider) for utilization, blocking and preemption, we can find a suitable lookahead time.

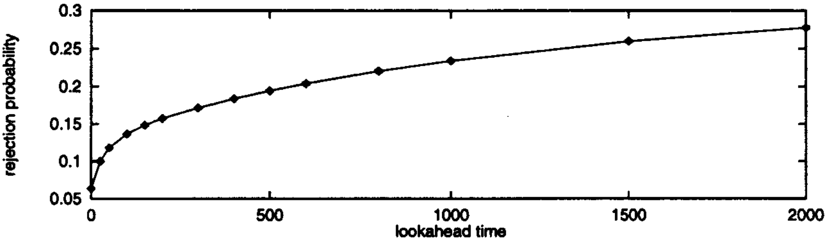

Figure 8 Rejection probability for immediate requests, at 40% advance request ratio

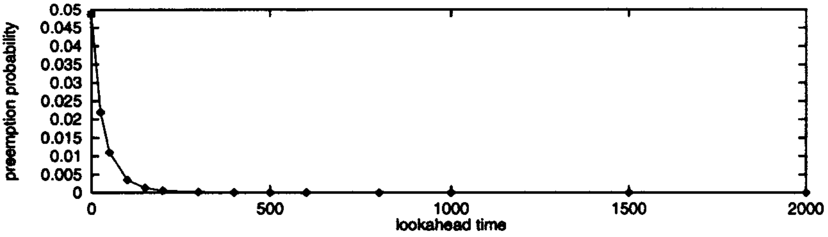

Figure 9 Preemption probability for immediate requests, at 40% advance request ratio

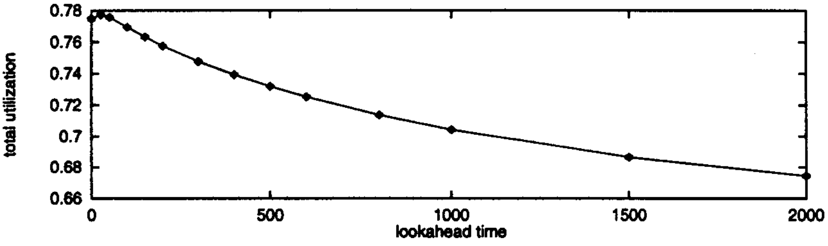

Figure 10 Utilization for immediate requests, at 40% advance request ratio

With a 40 % advance reservation request rate, we can, for example, use a lookahead of 50 time units, resulting in 77.6% of the resources being allocated under a 1.1% immediate request preemption probability (i.e., 0.44% total preemption probability). By using a lookahead of 100 time units, we get a utilization of 76.9% at a preemption probability of 0.35% for immediate requests. This is to be compared with 79% of the resources being allocated when there are no advance reservations. The offered load is in both cases 85% of the total capacity and the total rejection probability is around 7%.

If there were reservation agents without supporting lookahead in the immediate admission control algorithm, the results would be the same as having zero lookahead (the network must support preemption and non-blocking resource allocation that can be used by the advance reservation agent). In that case we have 5% preemption probability for immediate requests when 40% of the admission requests are made in advance.

5 CONCLUSION

Reservations for scheduled events run the danger of over-allocating resources independently of whether advance reservations are provided. By encouraging advance reservations for scheduled events, resources can be more accurately allocated over time compared to what would be the case if clients themselves had to decide when to allocate resources through immediate reservations, especially, when resources are scarce.

Our simulations show the effects of providing advance reservations. We assume allocating resources at some point for each reservation and the utilization is measured in terms of how much we can allocate. We do not consider drops in the link utilization due to reserved resources not being used.

In our model, resources are shared between advance and immediate admission control, i.e., there is no pre-partitioning of resources. The most important parameter in the admission control algorithm is the so called lookahead time, i.e., the point at which we actually start making the resources in the routers available for approaching advance reservations. The paper shows how a suitable lookahead time can be found through simulations.

The effects of advance reservations can be expressed in terms of rejection probability, preemption probability, and total link utilization. In the simulations presented we can allocate 79% of the link resources when there are no advance reservations and around 77% of the resources when 40% of the admission requests are made in advance, and there is a 0.4% preemption probability for immediate requests. The offered load is in both cases 85% of the total capacity and the total rejection probability is around 7%.

REFERENCES

Banerjea, A., Ferrari, D., Mah, B. A. & Moran, M. (1994), The tenet real-time protocol suite: Design, implementation, and experiences, Technical Report TR-94-059, University of California at Berkeley.

Clark, D. D., Shenker, S. & Zhang, L. (1992), Supporting real-time applications in an integrated services packet network: architecture and mechanism, *in* 'Proceedings of SIGCOMM', ACM, Baltimore, Maryland, pp. 14–26. *Computer Communication Review*, Volume 22, Number 4.

Degermark, M., Köhler, T., Pink, S. & Schelén, O. (1995), Advance reservations for predictive service, *in* 'NOSSDAV', Lecture Notes in Computer Science, Springer, Durham, New Hampshire, pp. 3–14.

Ferrari, D., Gupta, A. & Ventre, G. (1995), Distributed advance reservation of real-time connections, *in* 'NOSSDAV', Lecture Notes in Computer Science, Springer, Durham, New Hampshire, pp. 15–26.

Firoiu, V., Kurose, J. & Towsley, D. (1996), Efficient admission contròl for edf schedulers, *in* 'Proceedings of IEEE Infocom', San Fransisco, California.

Gupta, A., Howe, W., Moran, M. & Nguyen, Q. (1995), Resource sharing for multi-party real-time communication, *in* 'Proceedings of IEEE Infocom', Boston, Massachusetts.

Herzog, S. (1996), Rsvp extensions for policy control, Internet Draft, Internet Engineering Task Force.

Jamin, S., Danzig, P. B., Shenker, S. & Zhang, L. (1995), A measurement-based admission control algorithm for integrated services packet networks, *in* 'Proceedings of SIGCOMM', Cambridge, Massachusetts.

Mitzel, D. J., Estrin, D., Shenker, S. & Zhang, L. (1996), A study of reservation dynamics in integrated services packet networks, *in* 'Proceedings of IEEE Infocom', San Fransisco, California.

Reinhardt, W. (1994), Advance reservation of network resources for multimedia applications, *in* 'Proccedings of IWACA'.

Schelén, O. & Pink, S. (1997), An agent-based architecture for advance reservations, Luleå University of Technology. URL: *http://www.cdt.luth.se/~olov/publications.html*

Shenker, S., Partridge, C. & Guerin, R. (1997), Specification of guaranteed quality of service, Internet Draft, Internet Engineering Task Force.

Wolf, L., Delgrossi, L., Steinmetz, R., Schaller, S. & Wittig, H. (1995), Issues of reserving resources in advance, *in* 'NOSSDAV', Lecture Notes in Computer Science, Springer, Durham, New Hampshire, pp. 27–37.

Wroclawski, J. (1996), Specification of the controlled-load network element service, Internet Draft, Internet Engineering Task Force.

Zhang, L., Deering, S., Estrin, D., Shenker, S. & Zappala, D. (1993), 'RSVP: a new resource ReSerVation protocol', *IEEE Network Magazine* **7**(5), 8–18.