

Reservation Mechanisms for Efficient Resource Management in Internetworks

S. Damaskos

GMD-FOKUS,

Hardenbergplatz 2, 10623 Berlin, Germany

Email: damaskos@fokus.gmd.de

H. F. Salama

Center for Advanced Computing and Communication

North Carolina State University, Raleigh, NC 27695, USA

Email: hfsalama@eos.ncsu.edu

Abstract

Distributed multimedia applications with quality of service requirements that must be guaranteed by the underlying network are evolving at a fast pace. Resource reservation protocols are essential to enable efficient resource management in internetworks. We survey existing resource reservation protocols and the reservation styles they support. Current protocols do not permit different reservation styles to be combined within the same session. We show, using simulation of realistic scenarios that allowing different reservation styles to coexist within the same session yields more efficient management of resources and provides more satisfactory service to the applications. We then show the need for shared dynamic reservation styles which allow the users to switch between sources without the possibility of service interruption. A complete mechanism for implementing the shared dynamic reservation style is proposed, along with three techniques to enhance its effectiveness in managing the network resources. The effectiveness of these three techniques is evaluated using simulation.

1 Introduction

Distributed multimedia applications such as remote education, computer-supported cooperative work, and multimedia conferencing are evolving at a fast pace. These applications require guaranteed service, and therefore need more sophisticated networks than the classical data transfer type applications for which best-effort service is sufficient. The requirements of an application are expressed in terms of Quality of Service (QoS) parameters that must be guaranteed by the underlying network. In addition, distributed multimedia applications typically involve more than two users and thus expect the underlying network to support multicast (point-to-multipoint and multipoint-to-multipoint) communication in order to be capable of efficiently managing the available resources.

A multicast routing protocol, a resource reservation protocol, and an admission control mechanism are needed for properly setting up a multicast session for an application with QoS requirements. Several groups of researchers are currently working on the design of standard resource reservation protocols. Resource reservation is straight forward in the connection-oriented ATM networks. The situation is more complicated, however, in case of the already existing giant Internet. It is a point-to-point best-effort connectionless data network. The Internet Engineering Task Force (IETF) is putting a lot of effort to enable the Internet to support applications with QoS requirements without changing its existing model [1]. The first protocol for reserving resources on the Internet was the stream protocol (ST) [2] which evolved to ST2 [3] and eventually became ST2+ [4]. Another protocol that is currently being designed for the Internet is the Resource ReSerVation Protocol (RSVP) [5]. These protocols propose the use of different reservation styles with each reservation style being suitable for a certain class of applications.

In this paper, we propose techniques to improve the performance of resource reservation protocols, i.e. to make them capable of achieving more efficient management of the network resources and providing better service to the users. We use simulation to show how effective the proposed techniques are. The remainder of the paper is organized as follows: Section 2 describes the different reservation styles. Section 3 provides a brief summary of the existing resource reservation protocols. In section 4, we show the effects of allowing different styles to be used for reservations within the same multicast session. In section 5, we discuss the need for dynamic reservation styles and propose an efficient mechanism for implementing dynamic reservation. Section 6 concludes the paper.

2 Reservation Styles

Resource reservation protocols enable a communication network to provide real-time multicast service for applications with QoS requirements. These applications may be of different natures. For example, in a remote lecture there are multiple receivers and only one source, while in computer-supported cooperative work each participant may be a source and a receiver at the same time. The traffic characteristics (e.g. rate and burstiness) also differ from one media stream to another. Resource reservation protocols define a set of reservation styles to accommodate the requirements of the different applications and media streams. An application may choose one or more than one reservation styles depending on its nature and the characteristics of the media involved in that application. Figure 1 shows a classification of reservation styles.

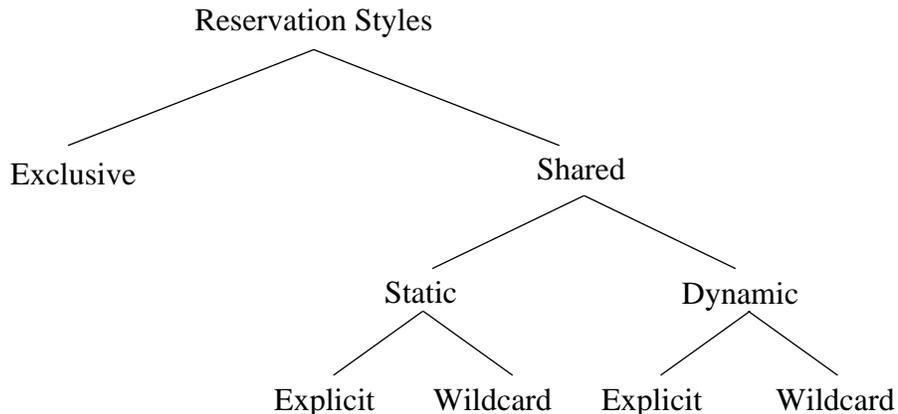


Figure 1: Reservation styles for resource reservation protocols

When using the exclusive reservation style, a node creates a distinct reservation for the traffic arriving from a designated source. The reserved resources can not be shared with traffic arriving from any other sources. The shared reservation style, on the other hand, allows the reserved resources to be shared among a set of sources. These sources represent the scope of the reservation. This set must be specified by the receiver in case of a shared explicit reservation. If the receiver chooses a wildcard reservation, however, then the reserved resources will be shared among all existing sources for that session. The wildcard style is useful when either the sources transmitting to a session are not known in advance, or when the sources transmitting to a session are dynamically changing. For example, if a source joins a session after a wildcard reservation has been established, it will be dynamically connected to the already existing reservation tree.

With the shared dynamic reservation style, once a reservation has been set up from a set of sources, S , to a receiver, that receiver is allowed to dynamically specify which sources' packets it would like to receive. The specified set of sources are called the active sources, $A \subseteq S$. Thus the receiver can switch from a set of active sources to another set of active sources at any time without having to tear down old reservations and to construct new ones. Thus shared dynamic reservations avoid resorting to admission control every time the receiver switches from one set of active sources to another. If the receiver of a dynamic reservation is not allowed to specify more than one active source at any time, then the specified source has all the reserved resources for itself alone (similar to exclusive reservation). In case of shared static reservations, however, the receiver doesn't have the luxury of switching between sources. Thus shared static reservations are useful if there is some floor control mechanism that decides which sources are allowed to transmit at a certain point in time or if only a small subset of the sources sharing a reservation are active at any time.

3 Resource Reservation Protocols

ST2 [3], and RSVP [5] are two protocols being designed to be deployed on the Internet. In this section, we briefly present the distinguishing features of these two protocols.

ST2 is a connection-oriented internetworking protocol. It coexists at the same layer with connectionless IP. It consists of two subprotocols: SCMP which is used to establish hard-state connections and to allocate resources to those connections, and ST which is responsible for the transmission of data over the established connections. ST2 implements a source-initiated resource reservation scheme. The source forwards a reservation request to all receivers. The intermediate nodes on the paths to the receivers reserve the required resources and forward the reservation request further downstream towards the receivers. Routing decisions are made during the reservation time. The receivers may accept or reject the reservation request, by sending a reservation accept/reject message back to the source. ST2 uses its own mechanism for group management and not the standard Internet Group Management Protocol (IGMP). To address a multicast group, ST2 has to construct an explicit list containing the addresses of all receivers. ST2+ [4], the successor of ST2, adds a receiver-based join mechanism that partially relieves the sources from the responsibility of tracking all receivers. IBM's implementation of ST2 applies a network layer scaling scheme which allows the support of heterogenous receivers [6].

RSVP is proposed to enable network resources to be reserved on top of a connectionless network protocol, e.g. IP. It is independent of other architectural components. For example, unlike ST2, the

construction of the multicast routing tree is completely independent of resource reservation. Resource reservation is receiver-initiated in RSVP. This permits RSVP to efficiently accommodate heterogeneous receivers, to support dynamic group membership, and to be consistent with IP multicast. RSVP is used to forward a receiver-initiated reservation request upstream to all nodes along the reverse data paths to the sources, and to reserve the requested resources along these paths. Resource reservations are represented by soft-state, so they must be refreshed periodically to prevent time-out and deletion.

A detailed comparison of ST2 and RSVP can be found in [7] and [8].

Table 1 summarizes the reservation styles supported by RSVP and ST2+. RSVP supports multicast sessions containing multiple sources, and it allows a receiver to select either the exclusive style or the shared static style when requesting a reservation. If the receiver selects the shared static style, it has to specify either the explicit or the wildcard scope. ST2+ only supports exclusive reservations and shared static reservations with wildcard scope.¹

	Exclusive Reservation	Shared Static Reservation		Shared Dynamic Reservation	
		Explicit	Wildcard	Explicit	Wildcard
ST2+	yes	no	yes	no	no
RSVP	yes	yes	yes	no	no

Table 1: Reservation styles supported by ST2+ and RSVP

RSVP doesn't permit exclusive reservations and shared static reservations to coexist for the same session. In ST2+, if a stream is a member of a group (of streams) it must use shared reservations, otherwise it only uses exclusive reservations. Thus neither RSVP nor ST2+ permit combining reservation styles. There are situations, however, where allowing such a combination may be beneficial. For example, a multimedia conference, involving the transmission of audio and video streams, is an application where combining more than one reservation styles may lead to more efficient management of the available resources. Shared static reservations are sufficient for the audio streams, because usually one participant speaks at a time. Exclusive reservations are necessary for the video streams, however, if the receiver wants to see all other participants. In section 4 we study

1. A wildcard scope in case of ST2 includes all streams associated with the same group. A group consists of a set of streams, and is identified by a group name.

the effect of allowing/forbidding the combination of different reservation styles on the network's efficiency in managing its resources.

Neither ST2+ nor RSVP define shared dynamic reservation styles. The current RSVP documents briefly describes a shared dynamic reservation style with wildcard scope as an open issue. This reservation style will not be included in RSVP version 1, due to its complexity and the incompleteness of its specification, but it may be considered further for future releases. In section 5, we propose a mechanism to implement shared dynamic reservations which allows the use of either the explicit or the wildcard scope. This mechanism as well as the ability to combine various reservation styles, described above, are part of the Internet Resource Management Protocol (IRMP) [9] which is a receiver-initiated hard-state resource reservation protocol.

4 Combining Reservation Styles

In section 3, we provided an example of a conferencing application which involves multiple media. These media may be of different natures, and thus different reservation styles may be needed to accommodate the requirements of each as well as to achieve efficient management of the network resources. If different reservation styles are not allowed to coexist within the same session, then the application has to use the same style for all media streams. In this section, we study both possibilities to determine whether or not there is an actual need for combining reservation styles within the same session.

For our analysis, we simulated a realistic scenario of a multimedia conference involving multiple users. Only one participant may be active (e.g. speaking) at any time. Each participant transmits two streams: an audio stream and a video stream. When a participant is active, he/she transmits high-quality video at a rate of R_V and audio at a rate of R_A . An inactive participant transmits only low-quality presence video at a rate of R_P and no audio. We used a receiver-initiated approach to reservation similar to RSVP and IRMP. Half of the participants do not have the capability to process video, and thus they only request to receive audio streams. These are the "incapable" participants. The other half are the "capable" participants. They request to receive the audio stream and high-quality video from the active participant, as well as the low-quality presence videos from all other participants. We didn't allow the use of shared dynamic reservations in this experiment. Three alternate approaches for reserving the resources necessary for the success of the multimedia conference were considered. All three approaches are capable of providing satisfactory performance to the participants. The approaches are:

- **Combining Exclusive Reservations and Static Shared Reservations (COMB):** Each incapable participant requests a static shared explicit reservation of R_A , to be shared among the audio streams from all other participants. Each capable participant requests exclusive reservations of R_P each for the presence video streams from each of the other participants. In addition, each of the capable participants requests a static shared explicit reservation of $R_V + R_A$ to be shared among the video streams from all other participants. This reservation is needed to carry the high-quality video and the audio streams from the active participant.
- **Exclusive Reservations Only (ERO):** Each participant requests exclusive reservations of R_A each for the audio streams from each of the other participants, and each capable participant requests exclusive reservations of R_V each for the video streams from each of the other participants.
- **Static Shared Reservations Only (SSRO):** Each incapable participant requests a static shared explicit reservation of R_A , to be shared among the audio streams from all other participants, while each capable participant requests a static wildcard reservation. The bandwidth requested for this reservation is $R_{WC} = R_V + R_A + (N - 2) * R_P$, where N is the number of participants in the multimedia conference. The components R_V and R_A are for the video and audio streams from the active participant, while the component $(N - 2) * R_P$ is the bandwidth required to receive the presence videos from all other participants.

We ran this experiment on networks created using Waxman’s random generator [10]. The parameters of the generator were adjusted to yield 100-node networks with an average node degree of 3.5, which is approximately the connectivity of the Internet. The costs of all links were set to unity. We used DVMRP [11] to construct the multicast trees from each participant to all other participants in the multimedia conference. The rates R_V , R_A , and R_P were set to 1.5 Mbps, 256 Kbps, and 256 Kbps respectively. This experiment, and the experiments in section 5, were run repeatedly until confidence intervals of less than 5%, using the 95% confidence level, were achieved for all measured quantities.

We used the cost of a multimedia conference as a criterion for evaluating the efficiency of each of the three approaches described above in managing the network resources. The cost of a multimedia conference is the bandwidth per link reserved for that conference, averaged over all links in the network. Figure 2 shows the cost of a multimedia conference versus the number of participants in

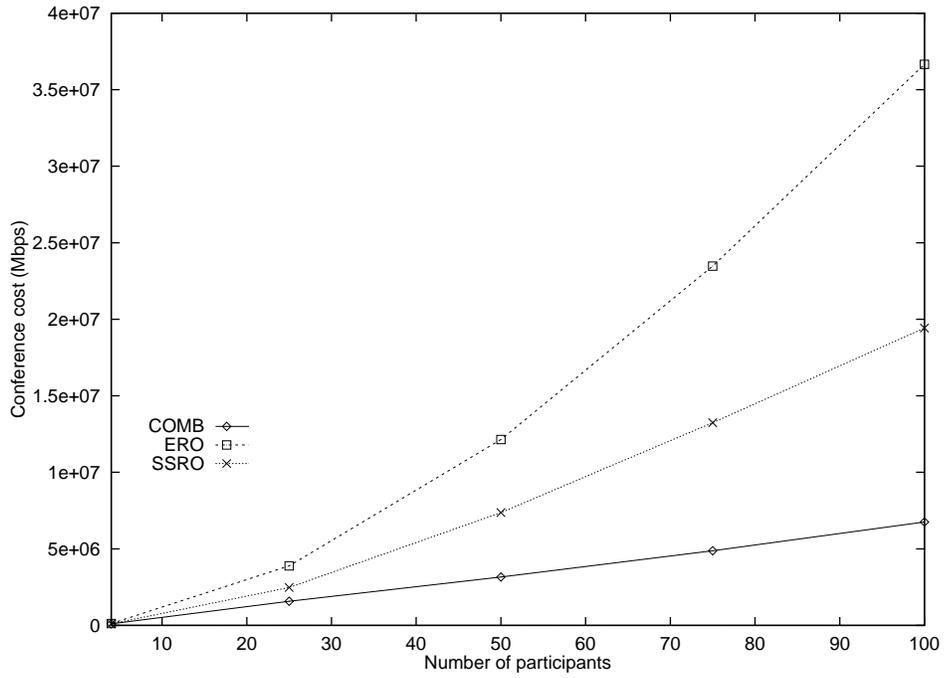


Figure 2: Cost of a multimedia conference, 100-node networks.

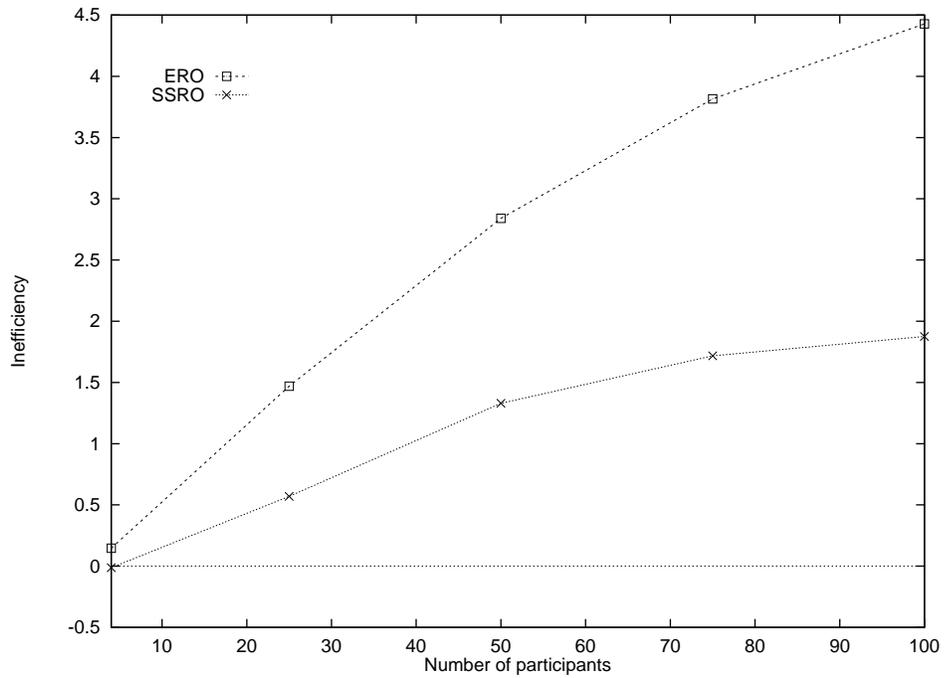


Figure 3: Inefficiency of ERO and SSRO relative to COMB, 100-node networks.

that conference, and Figure 3 shows the inefficiency of the ERO and SSRO approaches relative to COMB. It is defined as:

$$Inefficiency = \frac{Cost - Cost_{COMB}}{Cost_{COMB}}$$

The figures show that ERO's cost performance is always the worst. In ERO, each capable participant requests an exclusive reservation of 1.5 Mbps for each video stream from each other participant. This is a waste of resources, because at most one of these exclusive reservations will actually carry a traffic of 1.5 Mbps, namely the exclusive reservation to the active participant. All other reservations will only carry the presence video traffic of 256 Kbps. Similarly, each participant requests an exclusive reservation for each audio stream from each other participant. This is another factor contributing to ERO's inefficiency, because only the active participant transmits audio traffic at any time. ERO's inefficiency increases as the number of participants in the conference increases, because the number of unneeded exclusive reservations it constructs increases. When all 100 nodes have users participating in the conference, ERO yields conference costs that are more than 5 times the costs of COMB, and hence ERO's inefficiency exceeds 4 for large numbers of participants.

SSRO also yields worse cost performance than COMB. This is due to over-reservation: the bandwidth reserved on a link exceeds the requirements of the application. Over-reservation results in a satisfactory service to the application users, but poor management of the network resources. The appendix discusses the causes of over-reservation. The severity of over-reservation increases as the number of participants in the conference increases. That's why SSRO's costs are almost 3 times the costs of COMB, i.e an inefficiency of almost 2, when all 100 nodes have users participating in the conference.

In summary, ERO is capable of providing satisfactory service to the participants, but at the expense of very poor management of the network resources, because it constructs an exclusive reservation for each stream transmitting to the multimedia conference. SSRO also provides a satisfactory service to the participants at the expense of poor resource management due to over-reservation. COMB doesn't suffer from over-reservation, and it is therefore capable of providing efficient network management as well as satisfactory service to the participants. We therefore strongly recommend resource reservation protocols to allow different reservation styles to coexist within the same session.

5 The Dynamic Reservation Style

We introduced the shared dynamic reservation style in section 2. Once a dynamic reservation has been set up from a set of sources, S , to a given receiver, that receiver can switch from one set of active sources, A , to another at any time. This switching process adds extra complexity to the reservation protocol. Exclusive reservations are a simple alternative for the dynamic reservation style. One possibility is for the receiver to construct exclusive reservations to the sources of A , and then tear down these reservations and construct new ones whenever the set A changes. This approach is more efficient with respect to the network resources than constructing a dynamic reservation. If the load on the network is large, however, then the probability of a reservation request being rejected by admission control is also large. A dynamic reservation has to pass the admission control process only once during its set up, while the exclusive reservations have to pass the admission control process each time the receiver switches the active set of sources. We simulated several networks with heavy loads to determine the conditions when the use of the simple exclusive reservations is sufficient.

We ran this experiment on 20-node and 100-node networks with an average degree of 3.5. All link capacities were set to 155.2 Mbps (OC3), and admission control allowed reservations on a given link as long as the sum of the reserved bandwidths for all sessions doesn't exceed that link's capacity. We assigned an initial random load to each link in the network to represent the reserved bandwidth on the links. The load assigned to each link is a random variable uniformly distributed between L_{\min} and L_{\max} . First, we attempted to construct a dynamic shared reservation of 1.5 Mbps, and measured the probability of failure of such an attempt. A dynamic reservation request was considered successful only if all links on the paths from each source to the receiver pass the admission control criteria. In the second part of the experiment, we selected one source to be the only active source, and attempted to construct an exclusive reservation of 1.5 Mbps from the receiver to that source. If this attempt succeeded, we continued as follows: the load on each link was randomly modified within the bounds of L_{\min} and L_{\max} , then a new active source was randomly selected, the old exclusive reservation was torn down, and we attempted to construct a new exclusive reservation for the new active source. These steps were repeated, switching the active source each time, until one exclusive reservation request failed.

Figures 4 and 5 show the probability of failure of a multicast session versus the number of times the active source is switched (number of switchings) during the session for 20-node and 100-node networks. A multicast session is considered successful only if all exclusive reservations attempted

during that session are successfully established. The random link loads were distributed between 0 Mbps and 155.2 Mbps, the maximum link capacity. The switching operation doesn't affect the dynamic reservations, hence they are represented by the horizontal lines. As the dynamic reservations attempt to reach more sources, the number of links included in the reservation trees increases, and hence the probability of failure of the dynamic reservation requests increases. The performance in the case of exclusive reservation with switching is independent of the number of sources from which it selects the active source. Thus we only show one curve for that case. It is clear that as the number of switchings in a session increases, the probability of session failure also increases. Figure 5 shows that for more than 3 switchings the probability of failure of the dynamic reservations is smaller than that of the equivalent exclusive reservations, even when the scope of the dynamic reservation includes all other nodes in the network. Figure 6 shows a similar performance for the 100-node networks, except that the exclusive reservations yield less failures than the dynamic reservations with a large set of sources for multicast sessions with up to 11 switchings. Dynamic reservations with a small set of sources in their scopes fail less than the equivalent exclusive reservations even when the number of switchings is very small.

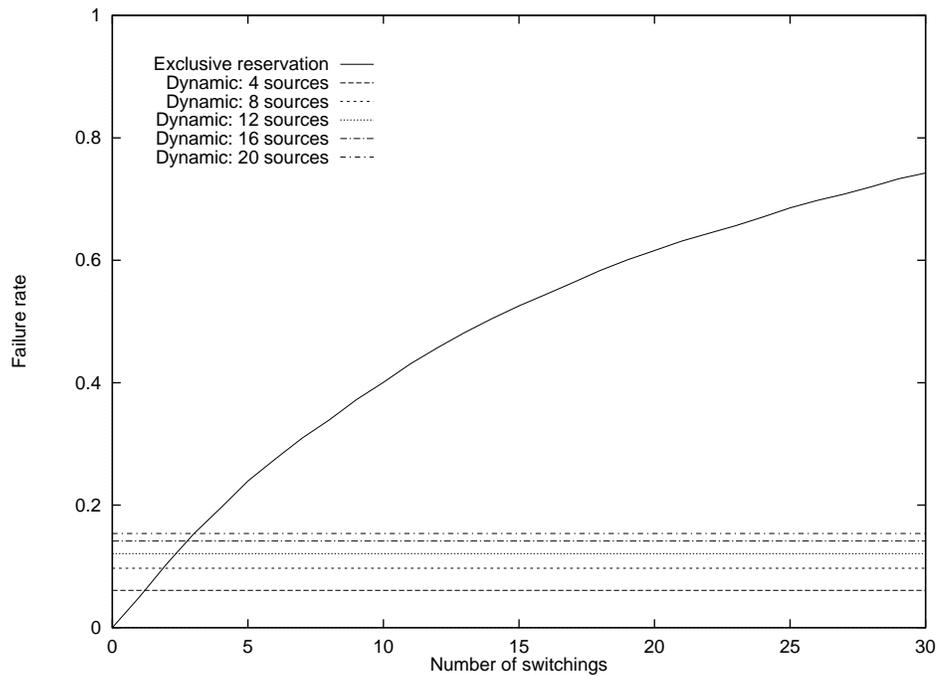


Figure 4: The probability of failure of exclusive reservations due to continuous switching in comparison to the probability of failure of a dynamic reservation, 20-node networks, $L_{\min} = 0$ Mbps, $L_{\max} = 155.2$ Mbps.

The failure probabilities given in figures 4 and 5 are high considering that the link loads are uniformly distributed across the entire range of a link's capacity, i.e. from 0 to 155.2 Mbps, and therefore the number of saturated links which may cause admission control failure is small. This is due to routing. The DVMRP protocol is unaware of the admission control policies, and therefore it doesn't attempt to avoid the saturated links when constructing a multicast tree for a given session. The resource reservation protocol then fails to reserve resources on the saturated links which are part of the multicast tree.

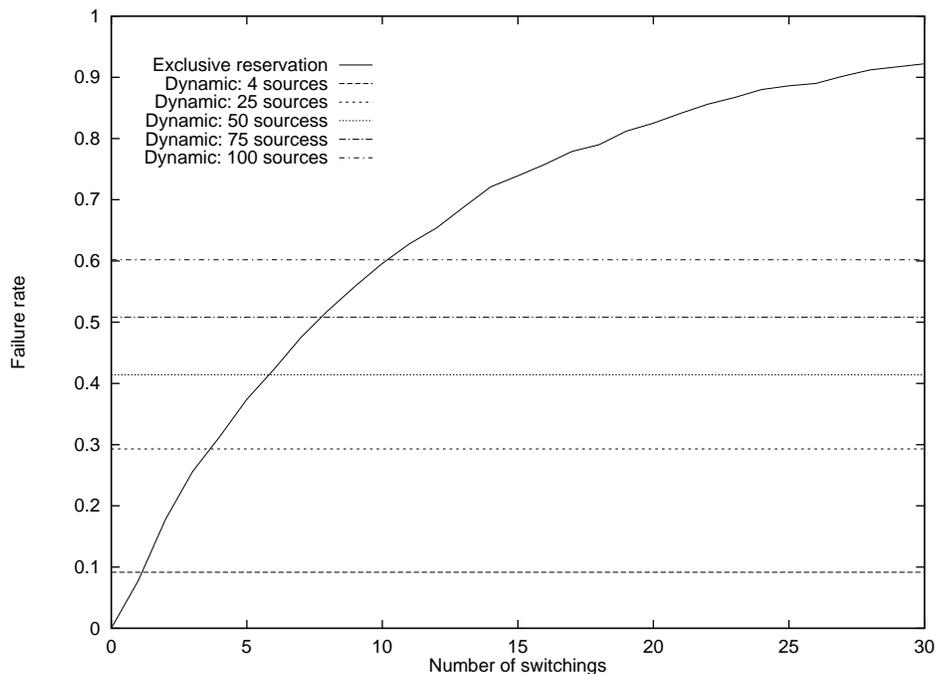


Figure 5: The probability of failure of exclusive reservations due to continuous switching in comparison to the probability of failure of a dynamic reservation, 100-node networks, $L_{\min} = 0$ Mbps, $L_{\max} = 155.2$ Mbps.

The two example cases shown in figures 4 and 5, and other experiments we ran with different loading conditions indicate that shared dynamic reservations can provide a stable service to applications in which the receiver changes the set of active sources frequently, e.g. conferencing type applications. Once the dynamic reservation is established, the user receives a guaranteed service. To construct and tear down exclusive reservations dynamically saves network resources, but the service provided to the receiver may be interrupted whenever the admission control mechanism rejects an exclusive reservation request due to the presence of saturated links on its path. Thus there

is a substantial need for efficient shared dynamic reservation styles. In the remainder of this section, we present a mechanism to implement shared dynamic reservations [9].

The shared dynamic reservation style we consider allows the use of either an explicit source list or the wildcard scope. It can coexist with other reservation styles within the same session. The set of active sources of a dynamic reservation contains no more than one source at any time. Merging reservations, as it is implemented for static shared reservations, is not applicable to dynamic reservations, even if they have overlapping or identical scopes. The reason is that different active sources may be selected for dynamic reservations from different receivers. Merging such reservations results in a reservation with only one active source, and streams from the other active sources may get lost. Prohibiting merging limits the efficiency of dynamic reservations in managing the network resources. We propose three techniques [9] to improve the efficiency of dynamic reservations:

- **Mapping:** When a shared dynamic reservation request arrives, check whether or not exclusive reservations exist for all sources listed in the scope of that request. If they do exist, check if the resources reserved for any of these exclusive reservations have to be increased in order to satisfy the requirements of the incoming dynamic reservation request. If the additional resources needed to update the exclusive reservations are less than the resources required to construct a dynamic reservation from scratch, then add these necessary resources to the exclusive reservations and don't construct a dynamic reservation.
- **Conversion:** When a shared dynamic reservation request arrives, locate all already existing shared dynamic reservations, if any, which have the same scope as the incoming request. Check if converting these shared dynamic reservations into exclusive reservations will save network resources. If yes, then tear down these reservations and establish an exclusive reservation for each of the sources listed in their scopes.
- **Reorganization:** After executing any exclusive or shared dynamic reservation request, check if the resources allocated to all exclusive and shared dynamic reservations exceed the resources that would have been allocated if only exclusive reservations were allowed. If this condition is true, select a shared dynamic reservation such that its scope holds more sources for which exclusive reservations exist than any other dynamic reservations. Convert that shared dynamic reservation into the corresponding exclusive reservations, then repeat the reorganization another time.

In order to verify the effectiveness of the three mechanisms described, we simulated a multicast session in which each participant is both a sender and a receiver. Each participant transmits high-quality video at a rate of R_V . Half of the participants wish to receive the video streams from all other participants and thus they construct an exclusive reservation for each video stream. The other half wish to receive one video stream only at any time. Therefore, each of these participants constructs a dynamic reservation with wildcard scope in order to be able to switch from one video stream to another at any time. Each participant randomly requests one of four bandwidth levels, $R_V/4$, $R_V/2$, $3R_V/4$, or R_V , for his/her reservations depending of his/her machine's capabilities. We ran this experiment on random 100-node networks with an average node degree of 3.5. All link costs were set to unity, and DVMRP was used to construct the multicast trees for all participant. R_V was set to 6 Mbps. The same experiment was repeated five times: with mapping only, with conversion only, with reorganization only, with all three improvement techniques, and without any improvement techniques.

The cost of a multicast session was used as a criterion for evaluating the effectiveness of each of the three techniques described above. The cost of a multicast session is defined as the bandwidth reserved per link for that session, averaged over all links in the network. Figure 6 shows the cost of a multicast session versus the number of participants. The figure shows that each of the three techniques alone can achieve a considerable improvement to the session cost as compared to the case when none of them is applied. The reorganization technique achieves better costs than the mapping and conversion techniques. Mapping comes in second place, because it occurs more frequently than the conversion technique. The resulting costs when applying all three techniques are not significantly better than the costs when applying reorganization alone. In short, all three techniques we proposed are quite effective in managing the network's resources.

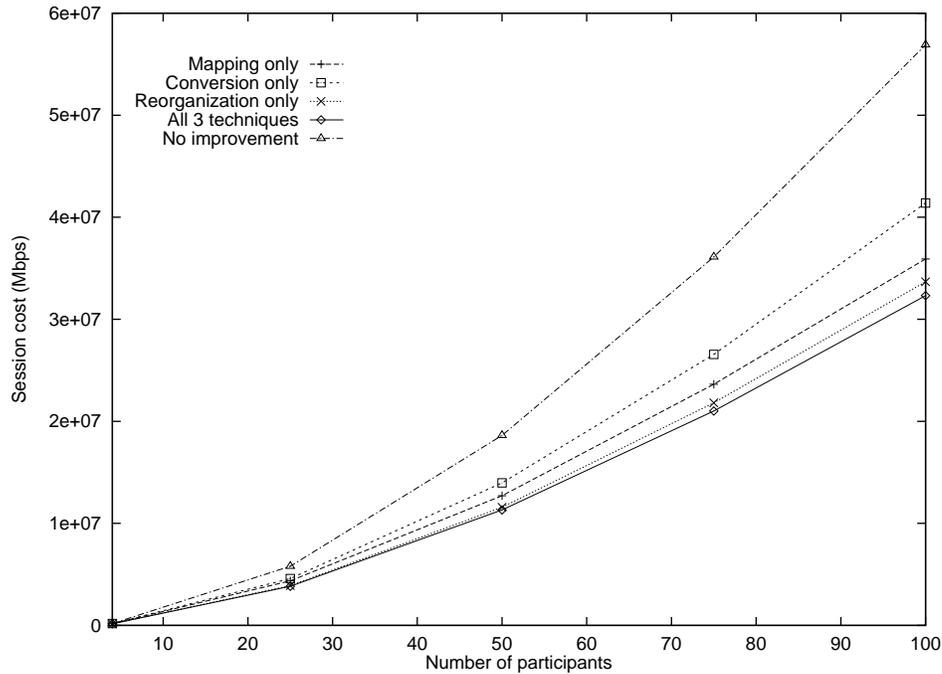


Figure 6: Effectiveness of the proposed techniques in the efficiency of shared dynamic reservations in managing the networks resources, 100-node networks.

6 Conclusions

Efficient resource reservation protocols are needed to enable current best-effort internetworks to support future’s distributed multimedia applications. In this paper, we proposed mechanisms to be used by resource reservation protocols to achieve more efficient resource management, and provide better service to the applications. We started with a brief discussion of the different reservation styles needed to accommodate the requirements of applications of diverse natures. Then we investigated two reservation protocols proposed for the Internet: ST2+ and RSVP. The two protocols are radically different from each other, but none of them allows different styles to be combined for reservations within the same session. There are numerous applications, however, which involve multiple media streams, and a different reservation style may be best suited for each of these media. We showed, by simulating a realistic scenario of a multimedia conference, that allowing different reservation styles to coexist within the same session yields both a more effective management of the available resources as well as a more satisfactory service to the application users.

We then turned our attention to the shared dynamic reservation style, and proposed a mechanism for implementing it. The proposed mechanism allows the use of both the shared explicit and the wildcard scopes. On the contrary to shared static reservations, dynamic reservations can not be merged. This limits their efficiency in managing the network resources. We presented three techniques to be applied along with dynamic reservations to improve their performance: mapping, conversion, and reorganization. Simulation results show that each of the three techniques is quite effective in reducing the resources consumed by a dynamic reservation. The reorganization technique is most effective followed by mapping while conversion comes last.

In summary, incorporating the shared dynamic reservation style and allowing different reservation styles to coexist improves the performance of resource reservation protocols considerably.

7 References

- [1] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," Internet RFC 1633, June 1994
- [2] J. Forgie, "ST - A Proposed Internet Stream Protocol," Internet Experimental Notes IEN-119, September 1979.
- [3] C. Topolcic, "Experimental Internet Stream Protocol: Version 2 (ST-II)," Internet RFC 1190, October 1990.
- [4] L. Delgrossi and L. Berger, "Internet Stream Protocol: Version ST2+," Internet RFC 1819, August 1995.
- [5] R. Braden et. al., "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification," Internet Draft: draft-ietf-rsvp-spec-07.ps, July 1995.
- [6] H. Wittig, J. Winckler, and J. Sandvoss, "Network Layer Scaling: Congestion Control in Multimedia Communication with Heterogenous Networks and Receivers," Proceedings of the International COST 237 Workshop, pages 274-293, November 1994.
- [7] D. Mitzel, D. Estrin, S. Shenker, and L. Zhang, "An Architectural Comparison of ST-II and RSVP," Proceedings of INFOCOM'94.
- [8] L. Delgrossi, R. Herrtwich, C. Vogt, and L. Wolf, "Reservation Protocols for Internetworks: A Comparison of ST-II and RSVP", Proceedings of the Fourth International Workshop on Network and Operating System Support for Digital Audio and Video, November 1993.
- [9] S. Damaskos, "Internetwork Resource Management Protocol: IRMP, Functional Specification," Work in progress.
- [10] B. Waxman, "Routing of Multipoint Connections," IEEE Journal on Selected Areas in Communications, Volume 6, Pages 1617-1622, 1988.
- [11] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol," Internet RFC 1075, November 1988.

Appendix

The over-reservation phenomenon occurs with static shared reservations, and causes poor management of the network resources. This appendix presents an example of a multimedia conference in which over-reservation occurs. We use the same conditions and terminology of section 4. Figure 7 shows a conference with 5 participants, P_1 through P_5 , in an 8-node network. We adapt the SSRO approach and assume that P_1 is a capable participant. P_1 requests a wildcard reservation of:

$$R_{WC} = 1.5 + 0.256 + (5 - 2) * 0.256 = 2.524 \text{ Mbps}$$

The request is forwarded upstream to the I_1 which reserves 2.524 Mbps on the link from I_1 to P_1 . I_1 is a branching point for this reservation request. It forwards the two copies of the reservation request further upstream to I_2 and I_3 . When I_2 receives this reservation request for $R_{WC} = 2.524$ Mbps, It calculates the total bandwidth requirement of all sources upstream from it:

$$R_{Total,I2} = R_{V,P2} + R_{A,P2} + R_{V,P3} + R_{A,P3} = 2R_V + 2R_A = 3.512 \text{ Mbps}$$

The bandwidth that I_2 actually reserves on the link from I_2 to I_1 is:

$$R_{Actual,I2,I1} = \min(R_{WC}, R_{Total,I2}) = 2.524 \text{ Mbps}$$

The reservation required to satisfy the requirements of the application on that link is, however:

$$R_{Satisfy,I2,I1} = R_V + R_A + 2R_P = 2.268 \text{ Mbps}$$

$R_A + R_V$ represent the shared reservations needed to carry the audio and video streams in case that either P_2 or P_3 is the active participant, and the term $2R_P$ represents the two exclusive reservations needed to carry the presence video from P_2 and P_3 . The application does not know in advance who the active participant is, and therefore it establishes exclusive reservations for the presence videos to all participants including the active one. That's why we use $2R_P$ and not only R_P . $R_{Satisfy,I2,I1} < R_{Actual,I2,I1}$ and thus there is an over-reservation on the link from I_2 to I_1 . As the number of partic-

Participants in a conference increases, the number of branching points on the paths of a static shared reservation to the sources in its scope increases, and over-reservation occurs more frequently.

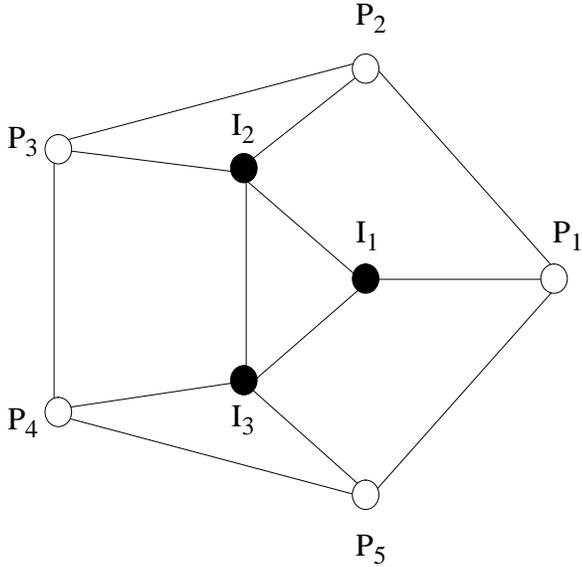


Figure 7: An example multimedia conference illustrating the over-reservation phenomenon.