On-Board RSVP: An Extension of RSVP to Support Real-Time Services in On-Board IP Networks

Muhammad Ali Malik[1,2], Salil S. Kanhere[1], Mahbub Hassan[1,2], and Boualem Benatallah[1]

[1] School of Computer Science and Engineering, The university of New South Wales
Sydney, NSW 2052, Australia,
[2] National ICT Australia Ltd.** ,
Locked bag 9013
Alexandra, NSW 1435, Australia

Abstract. The extension of Internet services to public transport passengers is slowly becoming inevitable. To this end, it is envisaged that high-speed local area networks will be deployed on-board public transport vehicles (e.g., buses, trains, ships and planes). The on-board LAN will be connected to the Internet via an on-board mobile router (MR). The passengers simply connect their devices to the on-board LAN and start enjoying Internet services. The mobility of the entire on-board network including the passenger devices is managed transparently by the MR. However, the mobility of the router (and the entire IP subnet) gives rise to several unique challenges for achieving end-to-end resource reservation. In this paper we propose a novel extension for RSVP, which addresses these issues. The proposed On-Board RSVP protocol can effectively, transparently, and scalably support end-to-end resource reservation in on-board IP networks. A key feature of On-Board RSVP is that it retains the basic building blocks of the original RSVP, minimising the changes required to existing RSVP infrastructure. The high level of dynamism associated with the QoS resource demand in an on-board communication system results in excessive signaling and processing overhead at the MR and the intermediate routers along the end-to-end paths. To address this issue, we propose and discuss two new aggregation schemes for handling the large number of RSVP setup messages: Cardinal Operating Policy (COP) and Temporal Operating Policy (TOP).

1 Introduction

In recent years we have witnessed an explosive growth in the availability of interconnected computing devices (e.g., PDAs, laptops, and 3G mobile phones) and the deployment of more sophisticated wireless communication infrastructure

---

** National ICT Australia is funded through the Australian Government's backing Australia's ability initiative, in part through the Australian Research Council.

(e.g., advanced data communication satellites). In order to achieve a truly pervasive computing environment it is imperative that we introduce Internet services in public transport systems. An on-board communication solution will enable transport operators to deliver value-added work, communication and entertainment services to their passengers. Indeed, providing limited on-board services such as access to entertainment and news is already a reality [1]. In recent years, this new paradigm of *Networks in Motion* is fast becoming an active area of research and development, and several commercial and research projects have been initiated to build such systems [2–7]. The work in this paper is part of our larger goal of providing On-board Communication, Entertainment, And iNformation (OCEAN) [8] for public transport systems.

A typical on-board mobility architecture consists of three main components (see Fig.1) : high-speed on-board local area network (OBLAN), the Mobile Router (MR), and Mobile Wireless Internet Connection (MWIC) [1]. The OBLAN provides a local high-speed connectivity to the outside world for on-board passengers. The MR facilitates communication between the OBLAN and the global communication infrastructure (e.g., Internet). The QoS module attached to the MR is responsible for the management of all QoS functions such as admission control and resource reservation. The OBLAN may also be additionally equipped with data server and query manager to process on-board user requests. The MWIC connects the MR to a land-based wireless station (e.g., 3G cellular packet data service) or a satellite (e.g., Inmarsat Swift64 mobile packet data service) to maintain connectivity between the OBLAN and the outside world. The heart of the architecture is the MR [9] which provides global connectivity whereby all users can access information by simply plugging into the OBLAN.

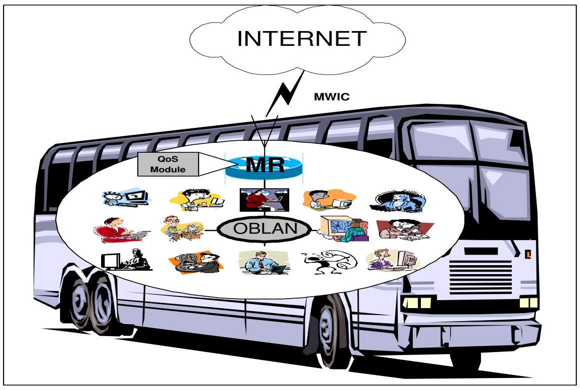


Fig. 1. Architecture of on-board Network

IP networks are considered a viable candidate for on-board computing environment due to their flexibility and cost effectiveness. A key characteristic of the on-board IP network is that the entire IP subnet consisting of the MR and its associated user devices, is mobile and may rapidly change its communication point to the outside world while moving. Existing IP protocols are not appropriate to cope with the requirements of the mobile networks. New extensions are required to provide continuous connectivity while MR changes point of association to the Internet. To this end, the Internet Engineering Task Force (IETF) has recently charted a new Working Group, called Network Mobility or NEMO to address the issue of mobility management for networks in motion. In the NEMO basic protocol [10] none of the user devices behind the MR are aware of the network's mobility. The MR is responsible for preserving session continuity as the network moves making the mobility of the network transparent and seamless to the on-board users. In other words, the on-board IP networks appears to be *static* to all user devices. This mobility transparency posse several new challenges for providing QoS support. The existing RSVP extensions for mobility are designed to work with mobile IP but not with the NEMO basic protocol. Further, since the MR is responsible for managing the mobility of the user devices, it is natural for the MR to handle the resource reservation on behalf of these devices.

Another challenging aspect of the on-board communication system is the high level of dynamism associated with the QoS resource demand at the MR. The number of user devices connected to the OBLAN in a public transport vehicle will invariably be very large. Further, this number is expected to change frequently as travelers depart the vehicle and new riders board. Last but not least, due to the wide variety of applications available to the users, there is a high likelihood that a single user will access different real-time services at different times. As a result the QoS requirements of a single user may also vary significantly over the duration of his trip. This high level of QoS dynamics will result in massive processing and signaling overhead of setup messages at MR and the other routers along the end-to-end paths from senders to receivers. Hence, it is extremely critical to address this scalability issue.

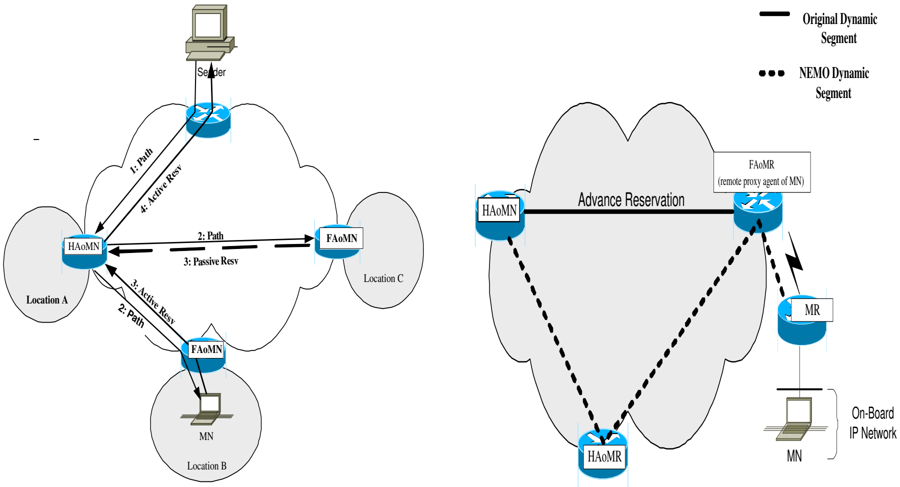
The rest of the paper is organized as follows. Section 2 provides an overview of existing RSVP protocols that have been proposed for mobile networks. We also elaborate on the unique challenges raised by on-board communication systems which render these protocols to be inapplicable in the on-board context. The scalability issues associated with the RSVP signaling overhead are also discussed in greater detail. To overcome these limitations, we proposed a new on-board RSVP protocol, which builds upon the building blocks of the traditional RSVP and extends it to cater for the requirements of mobile networks. Section 3 presents the conceptual architecture of the on-board RSVP protocol. In Section 4 we present the temporal and cardinal aggregation policies to tackle scalability issues. Finally, Section 5 concludes the paper and outlines future work.

2 Overview of Existing RSVP Protocols

The original RSVP proposed by Zhang et al [11] is a receiver initiated signaling protocol for the Integrated Services architecture [12] for establishing QoS paths between senders and receivers. If the sender/receiver is mobile, part of the end-to-end path changes over time. Resources in the old path have to be released, and required resources must be reserved along the new path each time the mobile host moves. Several extensions [13–16] of RSVP have been proposed to address host mobility and for gracefully integrating RSVP with Mobile IP. One common feature of these protocols is that they make advance reservations at future locations of mobile host.

We use MRSVP (a representative extension of RSVP for mobile communication) [13] as an example to illustrate the operation of RSVP extensions for unicast flows. In Fig.2 (a), we assume that a mobile node (MN) is receiving real-time data from a fixed sender. The path from sender to MN has a *fixed segment* from sender to HAoMN, and a *dynamic segment* from HAoMN to MN. The reservation over the dynamic segment is established through proxy agents at home (HAoMN) and foreign networks (FAoMNs) as shown in Fig.2 (a). The NEMO basic protocol proposed to handle mobility management for networks in motion requires the MR to maintain a bidirectional tunnel between its current location and its home network. All in-bound and out-bound on-board traffic is routed through this tunnel, making the mobility of the network transparent and seamless to the on-board users. The NEMO basic protocol bidirectional tunnel is not an issue since RSVP tunnel protocol [17] can provide signaling of RSVP messages in the tunnel. However, existing mobile RSVP protocols such as MRSVP posses certain fundamental drawbacks that make them inappropriate for on-board IP networks. Firstly, the dynamic segment (from HAoMN to MN) is now more complex in on-board IP networks due to the mobility of the router (MR). In NEMO basic protocol [10], the dynamic segment is broken into three sub-segments (see Fig.2 (b)), the first connecting the HAoMN and the HAoMR (home agent of mobile router), the second connecting the HAoMR and the FAoMR (foreign agent of MR), and the third, essentially a wireless link, connecting the MR to the FAoMR. Clearly MRSVP, is very limited in this environment, because it will reserve resources over the original dynamic segment (HAoMN and FAoMN), whereas all traffic mainly flows over the more circuitous NEMO basic protocol segment (see Fig.2 (b)).

Secondly, for MRSVP to function correctly (i.e., releasing resource over old path and converting passive reservation into active reservation over new path), MNs must be able to detect change of location (handoff). With NEMO, on-board MNs are incapable of detecting change of location (mobility is managed by MR). Thirdly, maintenance of individual RSVP flows (sessions) over the bi-directional tunnel between MR and HAoMR by periodic refresh messages can lead to serious scalability problems for the MR as well as the tunnel routers. Further the resources requested change dynamically with new sessions being established and existing ones being torn down quite frequently. Thus a change in the original reserved resources triggers the setup process [11]. Some of these issues, notably the scalability of refresh messages have been addressed elsewhere [18–20] in contexts which different from the on-board communication architecture. A scheme to aggregate the RSVP refresh messages has been proposed which essentially combines all the refresh messages for the various sessions over a single refresh



(a) (b)

Fig. 2. (a) Overview of MRSVP, (b) Dynamic segment problem of MRSVP

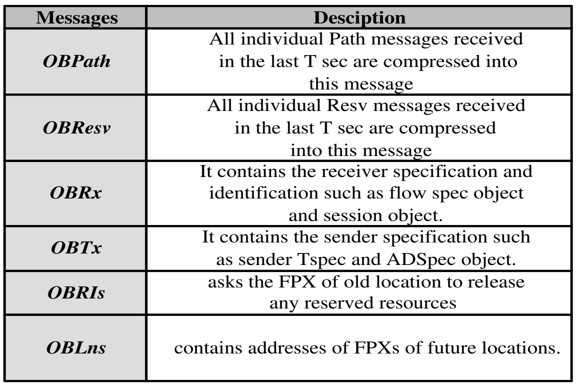
period into a single message. The aggregation of the refresh messages is much easier due to their periodic nature. However, handling the setup and tear down messages is much more difficult since they are triggered randomly based on user dynamics. This problem is further magnified due to the presence of a large number of users in the on-board system. A scheme to aggregate setup messages based on the bandwidth threshold has been briefly mentioned in [18]. However, to our knowledge, there has been no detailed analysis of aggregation schemes for setup messages.

3 Architecture and Operation of On-Board RSVP

Unlike most wireless end devices (e.g; mobile phones), whose mobility behavior is generally unpredictable, the routes of on-board IP network in public transport vehicle are known in advance. We can leverage this knowledge for making the advance reservations to the future locations. This important feature is used to design an effective and scalable On-Board RSVP protocol. This section presents the protocol architecture and operation of the proposed On-Board RSVP. We describe the QoS proxies, message formats, protocol operation, and handoff management.

3.1 QoS Proxies

On-Board RSVP requires three types of QoS proxies deployed at three different locations. A mobile proxy (MPX) is located at the on-board IP subnet, a home proxy (HPX) is located in the home network of the MR, and foreign proxies (FPXs) are located at subnets the MR visits during the trip.



Messages	Desciption
OBPath	All individual Path messages received in the last T sec are compressed into this message
OBResv	All individual Resv messages received in the last T sec are compressed into this message
OBRx	It contains the receiver specification and identification such as flow spec object and session object.
OBTx	It contains the sender specification such as sender Tspec and ADSpec object.
OBRls	asks the FPX of old location to release any reserved resources
OBLns	contains addresses of FPXs of future locations.

Fig. 3. On-Board RSVP messages

The main tasks performed by MPX are: (1) to compress multiple individual outgoing RSVP (Path and Resv) messages into a single message and de-compress incoming messages into individual RSVP messages, (2) establish active reservations from MR to home of MR, (3) to acquire addresses of FPXs (which are pre-allocated CoAs of MR) using mechanisms such as service location protocol (beyond the scope of this paper) at predefined future locations of MR and send these addresses to HPX. Task (1) addresses the scalability issue by reducing bandwidth overhead of RSVP signaling over the wireless connection, Task (2) allows reservation of resources over the correct path (NEMO dynamic segment shown in Figure 2(b)), and Task (3) helps HPX to establish passive reservation between home of MR and all future visiting locations. Passive reservations will be converted to active reservations when MR reaches a new location (similar to MRSVP). HPX has the following responsibilities: (1) compress multiple MR-bound RSVP (Path and Resv) messages into a single message and de-compress messages from MR into individual RSVP messages, (2) establish active reservations between home of MR and MR, and passive reservations between home of MR and FPXs. On behalf of the MR, FPXs establish passive reservations between future locations of MR and MR home using pre-allocated CoA of MR at their respective locations.

3.2 On-Board RSVP Messages

To facilitate the above tasks of QoS proxies, On-Board RSVP uses six new messages in addition to the existing RSVP messages respectively. These messages are briefly described in Fig.3. Fig.4 and Fig.5 shows the format of OBPath and OBResv messages respectively.

These additional objects in Fig.4 and Fig.5 help in compression and decompression mechanism. For compression of RSVP Path messages to OBPath message, the additional object ACC_Sender has the destination (receiver) addresses with the corresponding Sender_TSpec objects of the received Path mes-

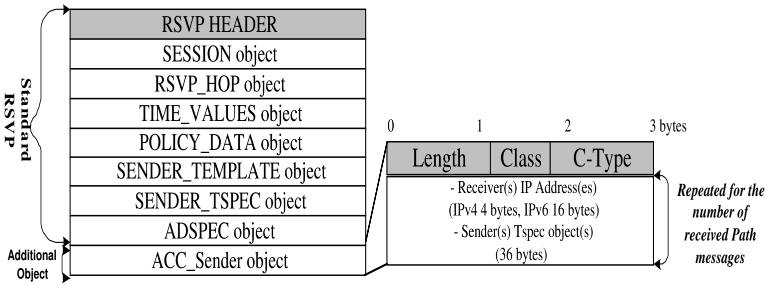


Fig. 4. OBPath message format

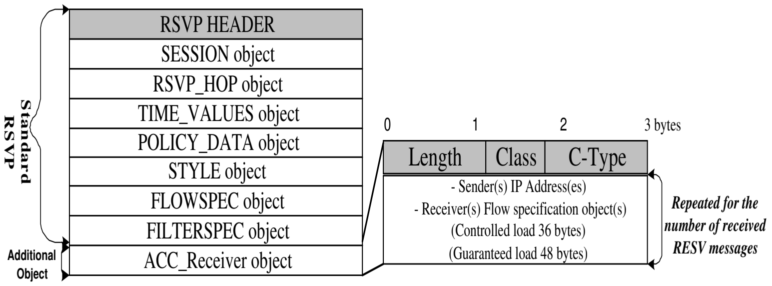


Fig. 5. OBResv message format

sages by the QoS proxy. Similarly for the compression of standard Resv messages to OBResv message, ACC_Receiver has the destination (sender) addresses with the corresponding flow specifications of the received Resv messages. The received Path and Resv messages are compressed by the QoS proxy depending on the aggregation policy in use. These aggregation policies will be explained in greater detail in the subsequent section.

To de-compress OBPath, the responsible QoS proxy will send the new standard RSVP Path messages to destinations using their respective Sender_TSpec specified in compress message. On the other hand, for the de-compression of OBResv, the responsible QoS proxy will send the standard Resv messages to the senders on behalf of the destinations (receivers) using their respective flow specifications stored in ACC_Receiver object.

3.3 Tunnel Operation

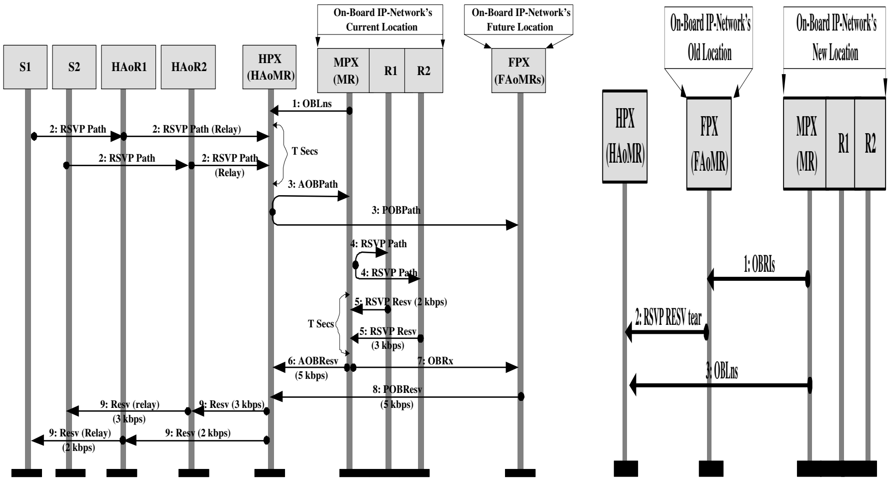
In RSVP tunnel protocol [17], data packets that require resource reservations within a tunnel are encapsulated by perpending an IP and UDP header and by using the UDP port number to distinguish packets of different RSVP flows. This is a layer violation problem because routers are designed to process data only at the network layer of OSI model. Other drawbacks with IP and UDP encapsulation are traffic control performance and IP level security problem. For the efficient operation of On-Board RSVP over bidirectional tunnel, the proposed protocol make use of flow label field of mobile IPv6 to distinguish between QoS and non-QoS data packets. Therefore in order to establish a proper On-Board RSVP session state in bidirectional tunnel routers, OBPath has a unique label in the sender template object and similarly OBResv must also contain that label in filter specification object. The packet classifier in the tunnel routers does not have to look at any port number but only at the pair of QoS proxy address and label in an encapsulated header.

3.4 Protocol Operation

Using proxies and messages described above, Fig.6 (a) illustrates the resource reservation process of On-Board RSVP. Initially, MPX sends the list of future locations (or addresses of FPXs) to HPX using the OBLns message (step 1). Let us assume that there are two senders, S1 and S2, sending data to two on-board receivers, R1 and R2 respectively. The periodic Path messages from senders go to the home agents of respective receivers, who then simply relay them to home network of MR (step 2). The HPX intercepts all these Path messages during a time period determined by the deployed aggregation policy. In this illustrative example, we assume that the TOP policy has been deployed wherein the HPX compresses all the Path messages in the last T sec, and sends a single compressed active OBPath (AOBPath) message to MPX and a passive OBPath (POBPath) to all FPXs contained in the OBLns message received earlier (step 3). Upon receiving the OBPath message, the MPX de-compresses it and delivers the individual Path messages to respective on-board receivers (step 4). The receivers respond with Resv messages with bandwidth requirements, in this case 2 Kbps and 3 Kbps (step 5). MPX compresses these two outgoing Resv messages into a single active OBResv (AOBResv) message and sends it to HPX over the wireless link (step 6). It also notifies the FPXs that the total bandwidth requirement is 5 Kbps using the OBRx message (step 7). The FPXs, upon receiving OBRx, establishes passive reservation to HPX using the passive OBResv (POBResv) message (step 8). Finally, the HPX de-compress active OBResv (AOBResv) message into individual Resv messages of 2 kbps and 3 Kbps and send them to S1 and S2 (step 9). In this case, the data packets, which require resources, are encapsulated by HPX with proper label (carried by OBPath and OBResv messages). While normal best effort oriented data packets are encapsulated with no label.

3.5 Handoff Management

Fig.6 (b) shows the dynamics of resource management of On-Board RSVP after a hand-off, i.e., after a MR changes its point of attachment from an old location to a new location. MPX sends an OBRIs message to its old foreign agent (step 1). Upon receiving the OBRls message from MR, MR's old foreign agent will send standard RSVP RESV tear message to explicitly free up the resources reserved by MR on routers along the path to the home agent of MR (step 2). Finally



(a) (b)

Fig. 6. (a) Resource reservation process, (b) Handoff scenario

MPX sends the updated list of its future FPXs (old FPX is removed from the list) to HPX (step 3).

4 Aggregation Schemes for Setup Messages

The QoS module at the MR and FAoMR are responsible for handling all the signaling messages. The refresh and setup/update messages are classified into two queues as shown in Fig.7. The incoming Path and Resv messages will be compared by the classifier to the existing Path state block and reservation state block respectively in order to determine whether they are new, updated, or refresh messages. Since new/update messages need different treatment to that of refresh messages, new and update messages are placed in the setup queue, while refresh messages are placed in the refresh queue.

One way to reduce the frequency of setup messages is the reservation threshold scheme [18]. In this scheme, the setup/update messages will only be sent when the reserved bandwidth changes by more than a certain threshold value R. We will call this policy as Resource Threshold Operating Policy (ROP). The main advantage of the ROP is that the granularity of bandwidth request in setup messages can be controlled by setting R. But this scheme suffers from three problems. Firstly, the ROP policy provides no control over the waiting time of the messages in queue. This time duration is entirely dependent on the value of the threshold and the bandwidth requirements of the new setup requests. In particular, if the bandwidth requirements of the sessions are considerably small, as compared to the value of the threshold, it can easily result in a considerably large waiting time. Secondly with ROP, there is no control over the level of aggregation

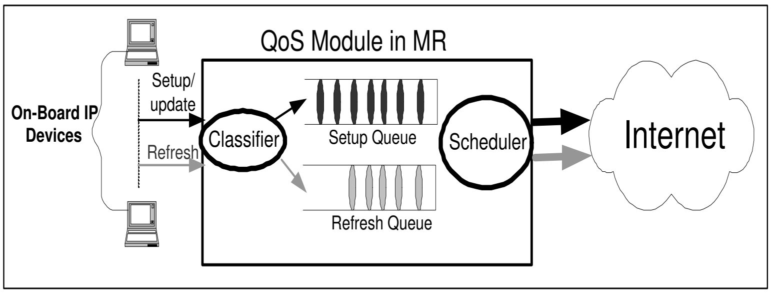


Fig. 7. Setup and refresh messages handling at MR

that can be achieved. In other words, we cannot control the number of messages that will be compressed. Lastly, ROP may result in unfairness in the waiting time duration for sessions with different resource requirements. For simplicity, assume that there are two categories of sessions with bandwidth requirements $r1$ and $r2$ respectively, where $r1$ is much less than R and $r2$ is approximately equal to R. One can readily see that sessions with bandwidth requirement $r1$ have to wait for a longer time in the queue before the threshold value is reached, as compared to the sessions with bandwidth requirement of $r2$.

As an alternative to ROP, we propose two new aggregation techniques to address the above problems: (1) Temporal Operating Policy (TOP), and (2) Cardinal Operating Policy (COP). The TOP is based on time interval. In this technique the QoS module takes a vacation of a fixed length T if there are no message(s) in queue. After the time period T, it scans the setup queue and if it finds message(s) waiting in the queue it starts the setup process (generating On-Board RSVP messages), otherwise it takes another vacation of length T. Unlike TOP, where the QoS module goes on vacation for a fixed duration of time, in COP the QoS module remains in the vacation state until a certain K request messages are accumulated in the queue. We assume that that there is no arrival during the setup process as the time for setup process is negligible. The problem of the waiting delay of setup messages can be managed by the TOP technique. Consequently, TOP may be used to aggregate real time sessions, which require defined minimum setup delay. The COP policy provides fine-grained control over the message aggregation factor by setting an appropriate value for K, and hence eliminates the second drawback of the ROP policy. As a result, with COP, it is possible to admit a defined number of sessions, which may be useful for situations where profit is based on number of admitted sessions. Contrary to ROP, the control parameters in TOP and COP are completely independent of the bandwidth requirement of sessions. As a result, these two schemes do not exhibit any unfairness towards the sessions with low bandwidth requirements, which is another limitation of ROP. However, unlike the ROP scheme, neither TOP nor COP provides any control over the granularity of the bandwidth aggregation.

In general, these aggregation policies will help to reduce the frequency of setup messages, which results in savings in terms of the signaling and processing overhead. However, since all these schemes aggregate several requests into one collective request, they will lead to a slight increase in the setup delay. It is important to note that the setup process delay is totally independent of the actual end-to-end delay experienced by flows. The control parameters of these policies can be adjusted based on passenger dynamics and the QoS requirements of the applications. For example, a non-real-time application may be able to sustain a longer setup delay. A hybrid QoS module could also be developed, wherein these three aggregation policies could be deployed simultaneously to cater to the varying requirements of different applications. However, there is a need to develop an analytical model for comparing the performance of these policies under different set of parameters such as setup delay and setup processing cost.

5 Conclusion and Future Work

We have identified three major limitations of existing RSVP protocols in the context of on-board IP networks. To overcome these limitations,, we have proposed an extension of RSVP, called On-Board RSVP. The architecture, protocol message formats and the operation of On-Board RSVP are described. One can readily see that On-Board RSVP requires minimum modifications to the existing RSVP protocol. The highly dynamic nature of the QoS reservations in an on-board communication system also introduces a scalability issue due to the large number of setup and tear-down messages that need to be processed by the MR and other routers along the bi-directional tunnel. To address this problem, we have also proposed two new aggregation schemes for setup messages : Cardinal Operation Policy (COP) and Temporal Operating Policy (TOP). As part of our on-going work, we are focusing on developing a mathematical model for analysing these schemes and comparing their performance characteristics under different scenarios.

Acknowledgment

The authors would like to acknowledge Dr. Debashish Saha and the entire OCEAN [8] team for their suggestions and feedback. The research was partly funded by Australian Research Council (ARC) Discovery Grant DP0452942.

References

1. K-D Lin and J-F Chang. Communications and entertainment onboard a high-speed public transport system. *IEEE Wireless Communication*, 9:84–89, February 2002.
2. PointShot Wireless: http://www.pointshotwireless.com.
3. Icomera: Real-Time Internet Services Onboard Trains: http://www.railway-technology.com/contractors/entertainment/icomera/.
4. Connexion by Boeing: http://www.connexionbyboeing.com.
5. Nautilus6 Working Group: http://www.nautilus6.org/.

6. In Motion Technology Inc.: http://www.inmotiontechnology.com.
7. Overdrive: Spectrum Efficient Uni-and Multicast Services over Dynamic Multi-Radio Networks in Vehicular Environments: http://www.comnets.rwth-aachen.de.
8. Project OCEAN: On-board Communication, Entertainment, And iNformation: http://www.ocean.cse.unsw.edu.au.
9. CISCO 3200 Series Wireless and Mobile Access Routers: http://www.cisco.com/en/us/products/hw/routers/ps272.
10. V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. NEMO Basic Protocol. Internet draft: http://www.ietf.org/internet-drafts/draft-ietf-nemo-basic-support-03.txt, June 2004.
11. L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A new resource ReSerVation Protocol. pages 8–18, September 1993.
12. R. Braden, D. Clark, and Shenker S. Integrated services in the internet architecture: an overview. RFC 1633, June 1994.
13. A. K. Talukdar, B. R. Badrinath, and A. Acharya. MRSVP: A Resource Reservation Protocol for an Integrated Services Network with Mobile Hosts. *Wireless Networks*, 7:5–19, 2001.
14. W-T. Chen and L-C. Huang. RSVP mobility support: a signaling protocol for integrated services Internet with mobile hosts. In *Proc.IEEE INFOCOM '00*, volume 3, pages 1280–1292, 2000.
15. C-C. Tseng, G-C. Lee, R-S. Liu, and T-P. Wang. HMRSVP: A Hierarchial Mobile RSVP Protocol. *Wireless Networks*, 9:95–102, March 2003.
16. A. Mahmoodian and G. Haring. Mobile RSVP with Dynamic Resource Sharing. In *Proc. IEEE Conference on Wireless Communications and Networking (WCNC'00)*, volume 2, pages 896–901, September 2000.
17. A. Terzis, J. Krawczyk, J. Wroclawski, and L. Zhand. RSVP Operation Over IP Tunnels. RFC 2746, January 2000.
18. A. Terzis, L. Zhang, and E. Hahne. Reservations for Aggregate Traffic: Experiences from an RSVP Tunnels Implementation. In *Proc. IEEE International Workshop on Quality of Service (IWQoS'98)*, May 1998.
19. F. Baker, C. Iturralde, F. Le Faucheur, and B. Davie. Aggregation of RSVP for IPv4 and IPv6 Reservations. RFC 3175, September 2001.
20. L.Berger, D. Gan, G. Swallow, P. Pan, F. Tommasi, and S. Molendini. RSVP Refresh Overhead Extensions. RFC 2961, April 2001.