Aggregation Policies over RSVP Tunnels

Muhammad Ali Malik[*,†] Salil S. Kanhere[*] Mahbub Hassan[*,†]
[*]School of Computer Science and Engineering
University of New South Wales, Sydney, NSW
2052, Australia
{mamalik,salilk,mahbub}@cse.unsw.edu.au

[†]National ICT Australia
Bay 15, Australian Technology Park
Eveleigh, NSW 1430, Australia

***Abstract*— IETF's integrated service with resource aggregation techniques provide a mechanism to reduce the RSVP states kept at router. Therefore, it reduces the number of RSVP refresh messages exchanged across routers. But the high frequency of setup messages exchanged between the edge routers can hinders the performance of aggregation. In this paper, we look at the management of the aggregation of resources over IP-in-IP tunnel with a view of reducing the setup messages. We develop and compare the cost functions of three aggregation control policies: the Temporal Operating Policy, the Cardinal Operating Policy, and the Resource Threshold Operating Policy. We also figure out the conditions under which the operating costs of these policies are minimum. Furthermore, we compare the performance of these policies under different conditions and shown that the Temporal Operating Policy will be more cost effective than the other two policies.**

I. INTRODUCTION

Among resource management mechanism, those offering the finest grain of traffic control work on a per-flow basis such as embodied by the IETF's Integrated Services Protocol (Int-Serv) [1]. However, flow-based resource reservation schemes can introduce an enormous load on the nodes along the path of a flow. To this end, aggregation of resources has been proposed as a mechanism to significantly reduce the memory overheads placed on core [2] and tunnel routers [3]. Further, the authors proposed mechanisms for aggregation over label switched paths [4], multiple domains [5], and by reservation agents [6].

However, the performance of aggregation depends on many factors, the most important of which is the session duration of the underlying flows. The aggregation scheme performs poor in case of short duration of sessions. This is because adding or dropping users' sessions often force the resource reserved by aggregation to swing quickly and with high variance. This could introduce a large number of setup messages, thus losing the advantage of scalability promised by aggregation. By setup messages, we mean RSVP new, update and tear-down messages in this paper. The proposed aggregation techniques [2]–[6] do not necessarily result in any decrease in the number of setup messages over the tunnel.

The large number of setup messages can clearly introduce an enormous burden on the intervening routers. This is because processing of setup message requires various stages such as new reservation session lookup, setting up new reservation states and setting the internal traffic control [7]. Feher et al [8] carried out an intensive performance analysis of different routers handling resource reservation messages. They observed the RSVP message processing time increases linearly with the session load and data traffic load. Further, setup messages incur bandwidth overhead. Since the setup messages move in the network as data packets belonging to a certain service class, the bandwidth available to other packets of that class decreases proportionally to the frequency of setup messages, as showed in [8].

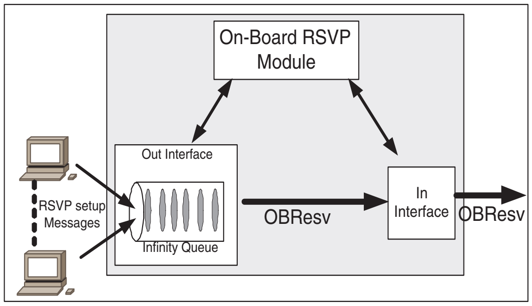

Fig. 1. Illustration of reservation messages merging

One simple way to address this issue is to reserve the maximum possible bandwidth over the tunnel. This removes the need for sending setup reservation request messages over the tunnel. Since this solution will result in the waste of bandwidth, in [9] we proposed a scalable resource reservation protocol based on RSVP [10]. Our protocol, known as On-Board RSVP [9], is designed to provide QoS guarantee to an on-board IP network, which is running NEMO basic protocol [11]. To improve the performance of aggregation, On-board RSVP [9] proposed a mechanism to compress multiple individual outgoing unicast setup messages into a single message. The devices where compressions take place are called compression points. For example, in case of IP-in-IP tunnels, the tunnel edge routers will act as the compression points. Fig. 1 illustrates the operations of compression at the outgoing interface of the tunnel edge router with respect to data flows. In Fig. 1, the On-Board RSVP module running at the edge router intercepts multiple outgoing RSVP setup messages during a time period determined by deployed control policies. The On-Board RSVP module then compresses these multiple setup messages into a single OBResv message and sends it to the incoming interface.

The number of messages compressed during the time period is determined by the deployed aggregation control policies

0-7803-9152-7/05/$20.00 © 2005 IEEE

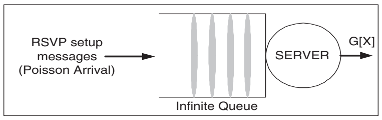

Fig. 2. $M/G^{[X]}/1$ Queueing Model

discussed in [9], which are: (1) Resource Threshold Operating Policy (ROP), (2) Temporal Operating Policy (TOP), and (3) Cardinal Operating Policy (COP). The aggregation technique employ by On-Board RSVP is modeled as $M/G^{[X]}/1$ queue with waiting period. $M/G^{[X]}/1$ with waiting period is different to known $M/G/1$ with vacation model [12] in three ways: (a) waiting period start as soon as the serve switches on for the service whereas vacation period start after the service time, (b) at the time, server switches on it serve only those messages which are present in the queue, (c) server serves all messages simultaneously. Therefore to evaluate the optimal parameters of these policies, we modified $M/G/1$ with T [13], K [14] and D [15] policies for TOP, COP and ROP respectively.

In this paper we develop an analytical model for On-Board RSVP's aggregation technique. We also investigate the optimal control parameters, which minimises the total operating cost, of these policies. The rest of the paper is organised as follows. In Section II we model the aggregation technique of On-Board RSVP by $M/G^{[X]}/1$ with waiting period queue and also describe the working of control policies. Following this in Section III, we develop the cost function to evaluate the optimal control parameter for the respective policies. The control policies are compared in different conditions in Section IV. Finally, Section V concludes the paper and outlines the future work.

II. MODELING OF AGGREGATION TECHNIQUE

The On-Board RSVP at tunnel edge router interface with infinite buffer is modeled as $M/G^{[X]}/1$ queue as shown in Fig. 2. The server here represents On-Board RSVP module. We assume setup messages arrive at the queue according to a Poisson process with rate λ, wait in queue for some time and are then served in a batch (altogether), after which they leave the system. The length of time by which messages are buffered in the queue is called *waiting period* W_p. At the end of the waiting period, the server begins to batch serve those messages which have arrived during the waiting period. Any new message which turn up in the service time are buffered in the queue and are served after the next waiting period. In case no messages arrive during the waiting interval, the next waiting period starts.

The batch service time in this model is to create (compress individual RSVP setup messages) and to send On-Board RSVP message. The time needed for the service of any batch has a general distribution G with a mean of:

$$S_p = cN + M \tag{1}$$

where c is the mean time needed by the server to process a single setup message, N is the number of setup messages that arrive during the waiting period and M is the mean transmission time of a single compressed setup message. To avoid an overlap between the waiting period and service time, we assume that the service time is less than the waiting period.

A. CONTROL POLICIES

The duration of the waiting period depends on the control policies. One such policy referred to as Resource Threshold Operating Policy (ROP), was briefly discussed in [16]. In this scheme, the waiting period ends only when the total bandwidth requested by individual messages changes by more than a threshold value D. In this policy we don't have control over the number of messages served. The processing and signaling overheads do not depend on the bandwidth requested by the message but rather depend on the number of messages. As an alternative to ROP, we proposed two new control policies based on number of messages: (1) Temporal Operating Policy (TOP) and (2) Cardinal Operating Policy (COP). TOP is based on time interval, where the waiting period is of a fixed time T. Unlike TOP, where the waiting period is fixed duration of time, in COP the waiting period finishes only when K number of messages are accumulated up in the queue.

III. PERFORMANCE ANALYSIS OF CONTROL POLICIES

In this section, we develop optimal control parameters of these policies by defining the cost function of On-Board RSVP's aggregation technique. The cost associated with running of control policies depends on the following parameters: (1) cost incurred due to time spent by the message in queue C_ω, (2) cost incurred due to the processing and signaling overheads on tunnel routers and links respectively when an On-Board RSVP message is sent C_φ, (3) cost at the edge routers because of processing individual RSVP messages to a single On-Board RSVP message C_p. As a result, we can express the generalised cost function C_c for the control policies:

$$C_c = C_\omega \omega + C_\varphi \varphi + C_p P \tag{2}$$

Where :

ω: expected time spent by the message in queue.

φ: expected number of On-Board RSVP messages sent by the server in unit time.

P: expected amount of time needed by the server in creating and sending On-Board RSVP message.

In the coming subsequent subsections, we drive the optimal parameters for the control policies under different conditions.

A. TEMPORAL OPERATING POLICY (TOP)

In TOP, the system has fixed duration T of waiting period. Therefore at fixed time T, server processes all messages simultaneously. The expected time spent by the message in the queue is the expectation of the waiting period, as follows:

$$\omega = E(T) = \frac{T}{2} \tag{3}$$

The expected number of On-Board RSVP message sent is the number of waiting period per unit time and the probability of at least one arrival during the waiting period. Thus,

$$\varphi = \frac{P\{N(T+t) - N(t) > 0\}}{T} = \frac{(1 - e^{-\lambda T})}{T} \tag{4}$$

The expected amount of time spent by the server on serving the messages in TOP is defined as:

$$P = S_p = c(\lambda T) + M \tag{5}$$

So, we obtain the cost function of TOP as:

$$C_c(T) = C_\omega . \frac{T}{2} + C_\varphi \frac{(1 - e^{-\lambda T})}{T} + C_P(c\lambda T + M), \quad T \geq 1 \tag{6}$$

1) OPTIMAL CONTROL FOR TOP: To find out the optimal value of T to minimise the cost function of TOP, the first derivative is examined to investigate the existence of any minima for $C_c(T)$. After simplification, the first derivative of $C_c(T)$ is given by,

$$\frac{d(C_c(T))}{dT} = \frac{C_\omega}{2} - C_\varphi \left(\frac{1}{T^2} - \frac{\lambda e^{-\lambda T}}{T} - \frac{e^{-\lambda T}}{T^2} \right) + c\lambda C_P \tag{7}$$

Since the function $C_c(T)$ is strictly positive and approaches infinity as T approaches infinity, we have the following two cases of $\frac{d(C_c(T))}{dT}$:

$$\frac{d(C_c(T))}{dT} = \begin{cases} +ve, & \frac{C_\omega x + C_P y}{C_\varphi} > z \\ 0, & \frac{C_\omega x + C_P y}{C_\varphi} = z \end{cases} \tag{8}$$

where $x = \frac{1}{2}$, $y = c\lambda$, and $z = \left(\frac{1}{T^2} - \frac{\lambda e^{-\lambda T}}{T} - \frac{e^{-\lambda T}}{T^2} \right)$. For positive $\frac{d(C_c(T))}{dT}$, $C_c(T)$ is an increasing function for $T \geq 1$; for this case minimum value of T, which is T^*, is at 1. For zero $\frac{d(C_c(T))}{dT}$, the function has a minimum at $T^* = a$.

Aggregation introduces tradeoff. Aggregation can significantly reduce the processing and signaling overhead cost C_φ but on the expense of C_ω and C_P. By doing numerical investigation, we also figure out that T^* increase in C_φ but decrease in the sum of C_ω and C_P. We assume C_ω and C_P to be equal because their sum is important rather than individual costs. Therefore in Table I we set the cost ratio $\frac{(C_\omega + C_P)}{C_\varphi}$ to 0.5, 1, 10, 20 to cover 4 levels of cost relationships.

TABLE I

COST PARAMETERS VALUES CONSIDERED

Case	$C_\omega = C_P$	C_φ	Cost Ratio
1	40	40	0.5
2	20	40	1
3	20	400	10
4	20	800	20

Fig. 3 shows $C_c(T)$ versus T for (1) cost ratio and (2) different arrival rates to explain two conditions described in Equation 8. Fig. 3 reveals the function $C_c(T)$ is an increasing function for the case when cost ratio is less than or equal to 1 irrespective of arrival rate λ. As stated before $T^* = 1$ in this case. Whereas, the cost function is convex on interval $[1, \infty]$ when cost ratio is greater than 1. In this case the value of T^* decreases in λ but increases with the cost ratio. For example $T^* = 5.2$ and $T^* = 2.8$ for $\lambda = 1$ and $\lambda = 10$ respectively for the cost ratio 10. On the other hand, T^* is increased by 2.4 times when the cost ratio increases from 10 to 20 (refer to Fig. 3 A). This shows that the waiting period should be increased as QoS demand increases to aggregate large number of QoS requests in a single message.

B. CARDINAL OPERATING POLICY (COP)

In COP, the waiting period does not have fixed duration but the number of arrivals K in waiting period is constant. Applying the memoryless property of the Poisson process, the expected length $E[W_L]$ of the waiting period is the sum of K exponential random variables each having a mean of $\frac{1}{\lambda}$. Thus, the expected wait time ω is given by:

$$\omega = E[E[W_L]] = \frac{K}{2\lambda} \tag{9}$$

Similarly for COP, φ is also the number of waiting period in a unit time, which is given by:

$$\varphi = \frac{\lambda}{K} \tag{10}$$

The expected service time in COP is simply:

$$P = S_P = cK + M \tag{11}$$

The cost function for COP is therefore as follows:

$$C_c(K) = C_\omega(\frac{K}{2\lambda}) + C_\varphi(\frac{\lambda}{K}) + C_P(cK + M), \quad K = 1, 2, ... \tag{12}$$

1) OPTIMAL CONTROL FOR COP: To determine the optimal value of K to minimise the cost function for COP, the first derivative of $C_c(K)$ is examined to investigate the existence of K^*, a positive integer which minimises (optimises) $C_c(K)$. Differentiate $C_c(K)$ with K and setting the result equals to zero yields:

$$K^* = \lambda \sqrt{\frac{2C_\varphi}{C_w + 2c\lambda C_p}} \tag{13}$$

Differentiate $C_c(K)$ with K twice and then substitute K^* to obtain:

$$\frac{d^2(C_c(K))}{dK^2} = \frac{2C_\varphi(C_w + 2c\lambda C_P)^{3/2}}{\lambda^3 (2C_\varphi)^{3/2}} \ > 0 \tag{14}$$

which implies that $C_c(K)$ is a concave upward (convex) function and achieves a global minimum at K^*. Similarly as T^*, from Equation 13, it can be observed that K^* also increase in C_φ and decrease in the product of C_w and C_P. Fig. 4 plots the cost function $C_c(K)$ as a function of K for different cost ratio cases (refer to Table I) at arrival rate $\lambda = 1$ and 10. In this

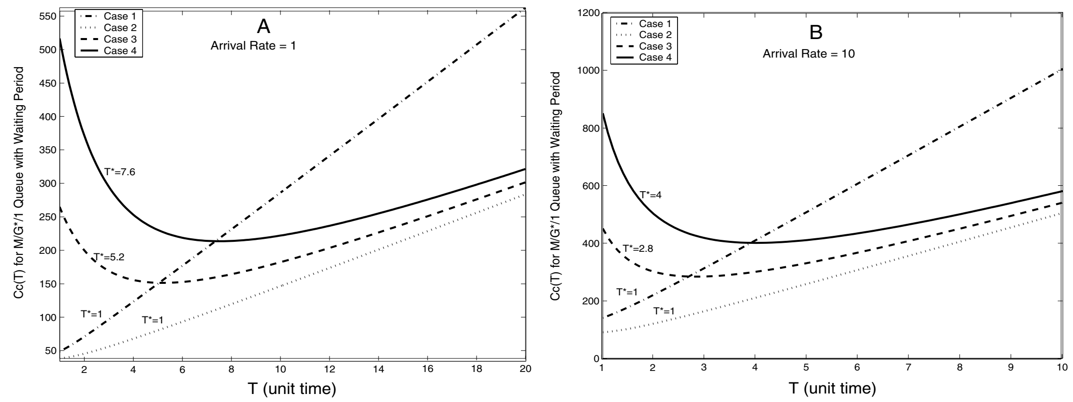

Fig. 3. Illustration of $C_c(T)$ for the two conditions of $\frac{d(C_c(T))}{dT}$ for case $1-4$, $c=0.2$ and $M=0.01$

examples, the value of K^* is rounded off to closest integer. Fig.4 also shows that K^* significantly dependent on cost ratio and λ; it increases as cost ratio and arrival rate increases. Except for cost ratio 0.5, K^* increases by 1 as arrival rate increase from 1 to 10. Similarly, maximum value of $K^*$is at higher cost ratio i.e., 20 in both graphs. This concludes that it is better to increase the aggregate level (number of messages to aggregate) in case of high QoS requests.

C. RESOURCE THRESHOLD OPERATING POLICY (ROP)

In ROP, the waiting period depends on the total bandwidth requested by individual setup messages. We are assuming the ith arriving setup message increases the total bandwidth requested (e.g., new and update RSVP messages) by Y_i or decreases the total bandwidth requested (e.g., teardown RSVP messages) by X_i. In this policy, $\{Y_i|i \geq 1\}$ and $\{X_i|i \geq 1\}$ are i.i.d and are independent of the arrival process. We are assuming that Y_i and X_i are a Poisson process with mean intensity of λ_{inc} and λ_{dec} respectively. The net expected bandwidth request is also a Poisson process with a mean of:

$$\lambda_{req} = |\lambda_{inc} - \lambda_{dec}| \tag{15}$$

Similarly as for COP, the expected length $E[W_L]$ of the waiting period in ROP is given by:

$$E[W_L] = \frac{1}{\lambda_{req}} \tag{16}$$

Thus,

$$\omega = E[E[W_L]] = \frac{D}{2\lambda_{req}} \tag{17}$$

φ in this policy is as follows:

$$\varphi = \frac{\lambda_{req}}{D} \tag{18}$$

The expected number of arrivals in waiting period of ROP is defined by:

$$E[L_a] = \frac{\lambda D}{\lambda_{req}} \tag{19}$$

Therefore the expected service time is:

$$P = S_P = \frac{c\lambda D}{\lambda_{req}} + M \tag{20}$$

The cost function for ROP is as follows:

$$C_c(D) = C_\omega \frac{D}{2\lambda_{req}} + C_\varphi \frac{\lambda_{req}}{D} + C_P \left(\frac{c\lambda D}{\lambda_{req}} + M \right) \tag{21}$$

1) OPTIMAL CONTROL FOR ROP: Differentiate $C_c(D)$ with D and setting the result to zero yields:

$$D^* = \lambda_{req} \sqrt{\frac{2C_\varphi}{C_\omega + 2c\lambda C_P}} \tag{22}$$

Since the second derivative of the cost function is positive, therefore the function $C_c(D)$ achieves a global minimum at D^*. The ROP is unstable in case $\lambda_{req} = 0$. In this case the messages have to wait in queue for indefinite time. By analyzing Equation 22 we note the following conditions about D^*: (a) it increases with high bandwidth changes and cost C_φ but (b) slightly decreases with more QoS request messages.

IV. COMPARISON OF THE POLICIES

In this section, we will compare the minimum expected costs $C_c(T^*)$, $C_c(K^*)$ and $C_c(D^*)$ for TOP, COP and ROP respectively. We perform a sensitivity analysis of the above minimum expected costs on changes in the arrival rate λ and cost C_φ. Fig. 5 plots the minimum expected costs for the policies verses arrival rate λ using the values of case no. 2 (refer to Table I). The plot shows that $C_c(K^*)$ increases linearly, whereas the arrival rate does not have significant effect on ROP. COP performs better than ROP when $\lambda \leq \lambda_{req}$. The performance of TOP is better than ROP and COP except for the higher rate when ROP and TOP performs equally.

In similar analysis, Fig. 6 shows the relationship between the minimum expected costs and cost because of sending of On-Board RSVP message (C_φ). Similarly to the above case, TOP performs better than COP and ROP but this time COP is a better choice than ROP. The difference in minimum expected cost between the policies increases with C_φ. Overall, TOP is

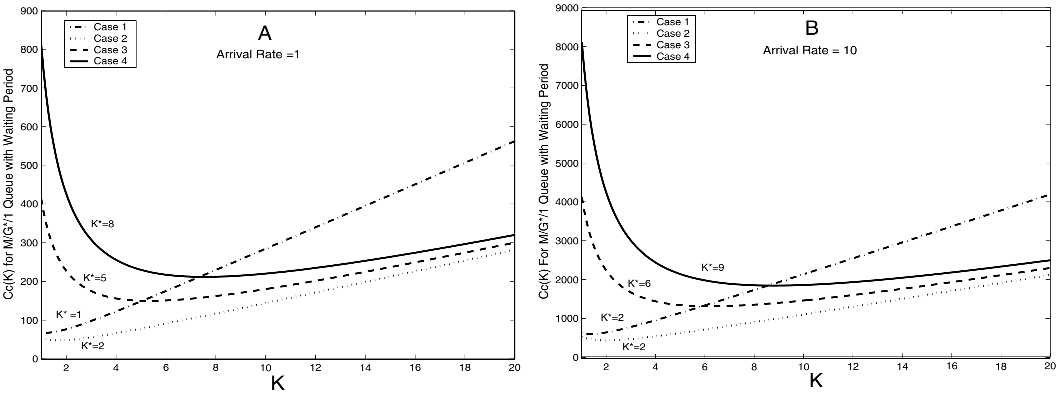

Fig. 4. $C_c(K)$ for $K \geq 1$ corresponding to case $1-4$, $c = 0.2$ and $M = 0.01$

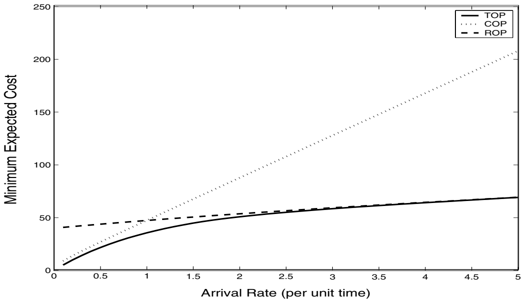

Fig. 5. Minimum Expected Costs verses Arrival Rate corresponding to case no. 2 and $\lambda_{req} = 1$

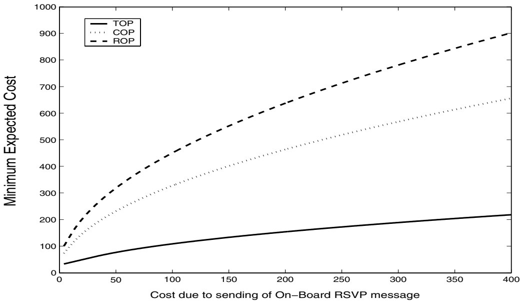

Fig. 6. Minimum Expected Costs verses C_φ corresponding to $\lambda = 5$ and $\lambda_{req} = 7$

an ideal situations where C_φ is significant such as the network involving mobility and wireless link.

V. CONCLUSION

In this paper, we considered three different control policies to improve the performance of resource aggregation and also investigate the conditions under which aggregation can simultaneously achieve high utilization and scalability in case of high QoS traffic. We derived models for the cost functions of these policies and conditions under which each of the policies is individually optimal. We also compared these policies in different set of conditions such as arrival rate of QoS request and the cost of sending the resource request messages and derived that the Temporal Operating Policy is cost effective than the Cardinal Operating Policy and the Resource Threshold Operating Policy. Our study will enable telecommunication carrier to optimise the cost of running resource reservation protocol over the network. There could be other situations involving signaling protocols where similar techniques could be put in practice to improve the performance of the network.

REFERENCES

[1] D. Clark R. Braden and S. Shenker. Integrated services in the internet architecture: An overview. RFC 1633, July 1994.
[2] F. Baker, C. Iturralde, F. Le Faucheur, and B. Davie. Aggregation of RSVP for IPv4 and IPv6 Reservations. RFC 3175, 2001.
[3] A. Terzis, J. Krawczyk, J. Wroclawski, and L. Zhand. RSVP Operation Over IP Tunnels. RFC 2746, January 2000.
[4] D. Gan T. Li V. Srinivasan. G. Swallow D. Awduche, L. Berger. Rsvp-te:extensions to rsvp for lsp tunnels. RFC 3209, 2001.
[5] E. Hahne P. Pan and H. Schulzrinne. Bgrp: A tree-based aggregation protocol for interdomain reservations. 2:157–167, June 2000.
[6] O. Schelen and S. Pink. Aggregation resource reservation over multiple routing domains. pages 107–116, May 1998.
[7] K. Nemeth I.Cseleny, G. Feher. Benchmarking of signalling based resource reservation in the internet. *Networking 2000 IFIP-TC6/European Commission International Conference*, May 2002.
[8] Istvan Cselenyi Gabor Feher, Krisztian Nemeth. Performance evaluation framework for IP resource reservation signalling. *Performance Evaluation*, 48:131–156, 2002.
[9] Muhammad Ali Malik, Salil S. Kanhere, Mahbub Hassan, and Boualem Benatallah. On-Board RSVP: An Extension of RSVP to Support Real-Time Services in On-Board IP Network. In *Chapter in Springer LNCS series*, volume 3326/2004, pages 264–275, 2004.
[10] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A new resource ReSerVation Protocol. pages 8–18, September 1993.
[11] A. Petrescu P. Thubert V. Devarapalli, R. Wakikawa. Network Mobility (NEMO) Basic Support Protocol. RFC 3963, January 2005.
[12] Dimitri Berksekas and Robert Gallager. *Data Networks*. PRENTICE HALL, Englewood Cliffs, New Jersey 07632, 1992.
[13] D. P. Heyman. T-policy for the M/G/1 queue. *Management Science*, 23:775–778, 1977.
[14] P. Naor M. Yadin. Queueing systems with a removable service station. *Operations Research Quarterly*, 14:393–405, 1963.
[15] H. Tijms K. R. Balachandran. On the D-policy for the M/G/1 queue. *Management Science*, 21:1073–1078, 1975.
[16] 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.