Syracuse University

SURFACE

Electrical Engineering and Computer Science College of Engineering and Computer Science

2001

Interdomain Resource Reservation via Third-Party Agent

Haci Ali Mantar
Syracuse University

Junseok Hwang
Syracuse University

Ibrahim Taner Okumus
Syracuse University

Steve J. Chapin
Syracuse University, chapin@syr.edu

Follow this and additional works at: https://surface.syr.edu/eecs

Part of the Computer Sciences Commons

Recommended Citation

Mantar, Haci Ali; Hwang, Junseok; Okumus, Ibrahim Taner; and Chapin, Steve J., "Interdomain Resource Reservation via Third-Party Agent" (2001). *Electrical Engineering and Computer Science*. 58.
https://surface.syr.edu/eecs/58

This Article is brought to you for free and open access by the College of Engineering and Computer Science at SURFACE. It has been accepted for inclusion in Electrical Engineering and Computer Science by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.

Interdomain Resource Reservation via Third-Party Agent

Haci Ali Mantar, Junseok Hwang, Ibrahim Taner Okumus, Steve J. Chapin
Syracuse University
Syracuse,NY 13244

Abstract

This paper describes interdomain resource reservation through a third-party agent called a Bandwidth Management Point (BMP). The BMP of each domain is responsible for admission control, dynamic bandwidth provisioning, DSCP assignment, policy control, etc. We also propose that each domain be free to choose its own intradomain resource reservation protocol.This model solves two significant problems of today's Internet: interdomain dynamic resource provisioning, and scalability of the backbone.

The BMP makes bulk reservations with each possible destination domain on behalf of hosts in its domain, and end hosts can join or leave the reservation without being involved with communication protocol between the domains. Reservations are based on the destination domain IP prefix and DSCP, and are dynamically updated according to aggregated traffic demand. Thus, unlike RSVP, the number of control states that backbone routers keep and the number of reservation setup messages between domains are reduced linearly with the number of domains. Since individual hosts are not involved with the interdomain reservation process, there is a corresponding saving in time.

Keywords: Scalability, Resource Reservation, Aggregation, QoS, Diffserv.

1 Introduction

Quality of Services (QoS) is becoming one of the most significant challenges faced in the internet as the use of real time applications such as IP telephony, interactive games, teleconferencing, video, audio, etc., continues to grow rapidly. These applications require some end-to-end QoS support, meaning that the internet is able to provide different service constraints (delay, reliability, BW, jitter, etc.) that are appropriate for different applications.

Currently, RSVP/IntServ is the only accepted protocol that supports these requirements. It requires all the nodes along a path to store three states for each session: classification, scheduling, and control. The number of states in routers is proportional to the number of flows, thus this model has serious scalability problems in the core routers if there is a large number of flows. Because of these problems, IETF came up with Differentiated Services (Diffserv), which pushes complexity to the edges of domains and maintains simplicity within the network core. Diffserv eliminates classification and scheduling problems in the core of the network by assigning packets to a limited number of Differentiated Services Code Points (DSCP) in the packets' IP header. However, since Diffserv does not have a signaling protocol, meaning that there is no mechanism to control injected traffic to a domain, it cannot guarantee end-to-end QoS. Thus, in order to have end-to-end QoS for Diffserv networks, a control path mechanism must be specified.

We propose interdomain resource reservation model through a third-party agent called a Bandwidth Management Point (BMP) [1, 2]. BMP is similar to Bandwidth Broker (BB), which was first introduced by Jacobson et al. in [12]. BMPs play the main role in the control path of Diffserv architecture. Each Diffserv domain has a single BMP that is in charge of the domain. Each domain is free to choose its own resource reservation protocol and to manage resources independently of other domains as long as it meets the requirements of bilateral agreement with its neighboring domains. In short, our model works as follows: Each BMP makes pipe-type reservations to possible destination regions (AS, sub-domain, or a set of regions that are identified by CIDR [14]) on behalf of its customers. Pipes are identified by a DSCP and destination IP prefix. The BMP adjusts the size of the pipe based on aggregated demand from its customer rather than an individual host demand. When an end host wants to make a reservation to another host in the same or different domain, it contacts its BMP. The BMP grants a reservation internally and externally if the end host has a right to make a reservation. Thus, end hosts can join and leave the pipe without making an interdomain reservation.

The rest of the paper is organized as follows: Section 2 gives related work, Section 3 describes the operation of our model, Section 4 describes dynamic resource provisioning, and in Section 5 the enhancement of our model is given.

2 Related work

Several studies are related to this subject. Pan et al. [7] proposed the Border Gateway Reservation Protocol (BGRP),whose main idea is to build a sink tree for each destination domain. Reservations from different domains that have the same sink tree are aggregated along the path. The number of states stored in core routers is proportional to the number of sink trees, which correspond to the number of destination domains. This mechanism solves many problems that RSVP exposes. This model is different from our work in two ways. First, dynamic bilateral agreement between domains is not managed by a single third-agent but by border routers. Second, dynamic interdomain resource management is not yet clearly done.

Third-party, agent-based approaches are presented in

[3, 11, 16], where signaling with Diffserv and business aspects of the Internet problems are solved. In [11] each domain has a Bandwidth Broker (BB) that is responsible for full control of its domain. Interdomain reservations are made by negotiation between BBs, and resource usage and charging depend on Service Level Agreements (SLAs) between BBs. In this approach, each domain makes an SLA with its adjacent domain without providing any information about destinations. It is difficult for a BB to grant reservations to other domains, because it has no information that will make sure that there are sufficient resources along the path in other domains further downstream [3]. Thus, this model still has weaknesses in providing end-to-end QoS, especially for Expedited Forwarding (EF) traffic. Also, aggregation (how border routers condition, how forwarding will be done) are still open issues.

Hwang et al. proposed the BMP architecture [1, 2]. In [1] network resources could be managed and controlled dynamically based on the market mechanism. By using the proposed BMP architecture [2], Hwang et al. also formatted an optimization model to control the various Diffserv resources. This work identified control scalability as an issue in implementing market-managed inter-connecting networks.

A fundamental work of BB, SIBBS (Simple Interdomain Bandwidth Broker Signalling) [13] was done by the QBone Signaling Design Team. In this work end-to-end QoS can be achieved either by end-to-end notification for each individual demand or by establishing a core tunnel between domains. As highlighted in the document, end-to-end notification has a scalability problem. The core tunneling approach does not have scalability problems, but since this is not yet completed, there are several open issues that are given in the document. The goal of our work is to extend SIBBS and to solve the research questions posed there.

3 Operation of the Model

We have mentioned that having an SLA between neighboring domains without distinguishing destination addresses makes it difficult to grant a reliable end-to-end QoS commitment. Therefore, in this model domains do provide destination information to downstream domains. Each BMP makes a pipe reservation to possible destination domains on behalf of its customers. Pipes are identified by a destination domain IP prefix and by DSCP.

In Figure 1, $S = S1, S2..Sn$ represents source domains, and $H = Hi1, Hi2, ...Hin$ represents hosts in each Si domain. $T = T1, T2, ..Tm$ represents transit domains, and D represents destination domain. For simplicity, we assume that there are no end hosts located in transit domain and that requests are directed from Si to D. We also assume that requests are based on bandwidth demands.

As shown Figure 1, source domains $(S1, S2, .Sn)$ establish a pipe to transit domain (T1) for the traffic destined for D. The BMP of T1 aggregates its customers' $(S1, S2, .Sn)$ requests and establishes a pipe to T2; similarly, the BMP of T2 establishes a pipe to D for traffic coming from its customers $(T1, T3, T4, Tm)$. Each BMP dynamically modifies the pipe to the downstream domain based on aggregated demands from its customers. In Section 5 we will explain pipe modification in more detail.

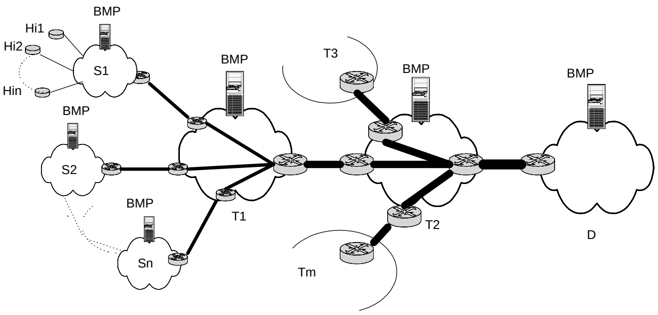

Figure 1: Example of architecture

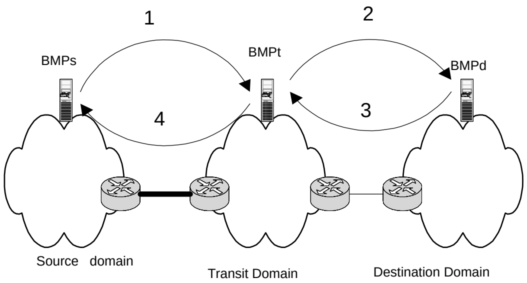

Figure 2: Interdomain pipe set up architecture

Now, each BMP can guarantee a certain level of service to upstream BMPs based on the services provided by downstream BMPs for the traffic destined for D. If a host requests a reservation to D from the BMP of S1, the BMP of S1 can make the decision without noticing further downstream BMP. Similarly, when the BMP of T1 gets requests from any of its customer BMPs to modify, remove, or establish a new reservation, it can handle those requests within its domain, because it has a certain level of commitment from the downstream domain (T2) for traffic destined for D. In short, each BMP can handle requests received from an upstream domain without noticing downstream domains. To do that, the BMP makes the size of the pipe greater than the actual usage rate by taking current, historical, and near-future customer requests into account. Once the aggregated demand is out of the range of certain levels, a BMP attempts to adjust it. In the following subsection we define a way to set up pipes.

3.1 Interdomain Pipe Setup Protocol

Our pipe setup protocol is similar to interdomain signaling protocol in the current version of BB [13]. It aims to extend the current BB protocol to make more scalable and powerful rather than to replace it. Typical interdomain control messages include resource allocation requests (RAR), resource allocation answers (RAA), cancels (CANCEL), and cancel acknowledgements (CANCEL ACK), as recommended by BB [13]. We have assumed that BMPs communicate with each other via TCP connections.

The reservation steps (Figure 2) between domains can be explained as follows: Suppose the BMPs wants to establish a pipe to BMPd. (Numbers in parentheses are keyed to the numbers in Figure 2.)

- BMPs sends RAR (1) message to BMPt asking to establish a pipe to BMPd with specified size and traffic parameters.
- BMPt checks whether BMPs is authenticated to make this reservation.
- Checks whether there is already a pipe with available room to handle the RAR (1) from ingress router to that destination. If so, it accepts the RAR (1), and sends RAA (4) to BMPs.
- If there is a pipe from the ingress router to the destination domain that is too small, BMPt sends RAR (2) to BMPd to increase the size of pipe.
- If the pipe to the destination domain is not extendible, or if there is no pipe, it determines the egress router from the BGP routing table, and checks for an available intradomain path between ingress and egress routers from the IGP table, then checks whether there is an available pipe between egress and destination regions. If there is, and it can handle the RAR (1), the BMPt accepts the RAR and sends RAA (4) to BMPs; if there is no available room in the pipe, it sends RAR (2) to BMPd.
- If there is no pipe between the egress and the destination domain, it sends RAR (2) to BMPd to establish a pipe.
- If BMPt gets positive RAA (3), it sends positive RAA (4) to BMPs; otherwise, sends negative RAA (4).

If there is no established pipe between BMPt and BMPd, RAR (2) is sent to BMPd to ask for pipe reservation. BMPd does the following tasks:

- Checks whether BMPt is authenticated to make the reservation.
- Checks its available resources as to whether it can handle the RAR (2).
- If both of the above tests are positive, it sends RAA (3), if not, it sends negative RAA (3).

After BMP establishes a new pipe, or modifies an existing pipe, it sets up traffic conditioning parameters in the edge routers for checking commitments. All traffic in the same pipe is conditioned as a single flow source; there is no isolation between traffic in same the pipe. It is totally the responsibility of the upstream domain to condition fairly among different sources [19]. Thus, traffic from different sources can be aggregated into the same pipe as their destinations merge. A pipe might consist of several pipes from different sources to the same destination, but traffic conditioning functions are applied per pipe, because all the traffic in the pipe is considered as one source of traffic. Intermediate domains do not have to know where traffic comes from originally.

We do not specify any intradomain resource reservation protocol; each domain is free to choose its own internal resource allocation protocol, such as RSVP, as long as it meets interdomain agreements. Reservation requests are traversed transparently between the source and the destination domain without involving interdomain process. [5, 10, 11]

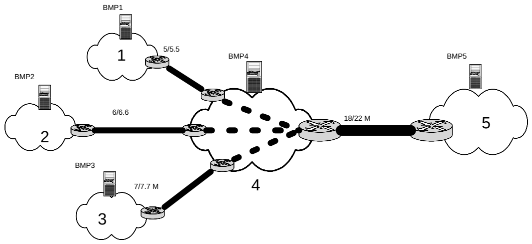

Figure 3: Example of model

3.2 An Example

In Figure 3, current usage BW from domains 1, 2, 3 to domain 5 are 5, 6, 7 Mbps, and reserved BW is 5.5, 6.6, 7.7 Mbps, respectively. Also, the usage and reserved BW from domain 4 to domain 5 are 18 and 22 Mbps, respectively.

Let's say a host in domain 1 wants to make a reservation that requires 40 Kbps to another host in domain 5. The host sends RAR to BMP1 asking for 40 Kbps BW. Since there is already an existing pipe to that domain which has available room to handle this request, BMP1 accepts the RAR without going through the interdomain process. Now, suppose that the load of pipe between domain 1 and domain 4 for traffic destined for D exceeds the high threshold (e.g., 95% of its pipe capacity), BMP1 sends RAR to BMP4 asking to increase its pipe size from 5.5 to 6.5. Since the pipe between domain 4 and domain 5 has available capacity to handle this request, BMP4 accepts the request without going beyond its domain. Now let's say that in a certain time interval BMP1 receives 50 new reservation requests with an average of 40 Kbps bandwidth requirement. On the other hand, 45 reservations are terminated with an average of 40 Kbps. Since the size of the pipe did not exceed specified levels (low threshold, high threshold), BMP1 will handle all the changes without requesting any modification from BMP4. Similarly, let's say that after some time, BMP1 increased the size its pipe from 6.5 Mbps to 8.5, BMP3 decreased the size of its pipe from 7.7 to 5, and BMP2 increased the size of its pipe from 6.6 to 7.6. BMP4 will not renegotiate with the BMP5 for these changes, because it never exceeds the size of the pipe. So, as shown in the example, BMPs can handle thousands of reservations without going through interdomain communication, or with a few modifications. This ability makes this model very scalable.

4 Dynamic Provisioning

In the previous section, we explained how BMP establishes a pipe between two regions, we assumed that the pipe is established, and we explained the mechanism with which BMP dynamically modifies the size of the pipe. Two issues that need to be considered when modifying the pipe are the efficient use of resources and scalability. Network resources can be used more efficiently if reservations are made based on individual demands. However, this has two detrimental effects. First, each request will expose interdomain reservation setup delay. Second, in the core of the network there will be a large number of setup messages that cause processing and bandwidth overhead. If the BMP chooses a size much greater than it actually uses, there will be no scalability problem; however, the network resources will be under-utilized. Thus, there is a trade-off between the number of setup messages and efficient resource usage. A key issue is therefore to determine an appropriate size of pipe modification.

The boundaries of a pipe are defined by low threshold (LT) and high threshold (HT) where HT=$R \times H$ and LT=$R \times L$. H and L are constant, and $L < H < 1$ (e.g, H=0.95, L=0.75). R is the current size of the pipe and R_a is denoted as the actual usage level of the pipe. R_{new}, HT_{new}, LT_{new} are modified values of R, HT, and LT, respectively. The modification of pipe size is done according to the following algorithm. For simplicity, we assume that BMP calculates actual traffic rate based on traffic token bucket parameters.

```
if($R_a > HT$) then
    choose $\alpha$
    $R_{new} \longleftarrow R + \alpha \times T$
endif

if ($R_a < LT$) then
    $R_{new} \longleftarrow Ra \times D$
endif
    $HT_{new} \longleftarrow R_{new} \times H$
    $LT_{new} \longleftarrow R_{new} \times L$
```

As shown above, whenever a BMP detects that utilization of the pipe exceeds HT, it attempts to increase the size of the pipe. When utilization drops to below LT, it attempts to decrease the size of the pipe.

The next task is to determine how much the size of the pipe should be increased and decreased. Increasing amount is based on past values. Let us denote t_n, t_{n-1}, t_{n-2}....t_1, for past adjustment time, and R_n, R_{n-1},..R_1 as the usage rate of the pipe at adjustment time. The size of the pipe for the next time interval T can be predicted from past values. Prediction can be done in several ways, including ARIMA [18], Hidden Markov Model [16], or neural network [16]. We chose a very simple algorithm which is based on exponential averaging, similar to the mechanism used by Jacobson for estimating round-trip time [17]. We denote α_p and α_c for slope of incoming traffic in the previous and current intervals repectively.

$$\alpha_p = (R_{n-1} - R_{n-2})/(t_{n-1} - t_{n-2}) \tag{1}$$

$$\alpha_c = (R_n - R_{n-1})/(t_n - t_{n-1}) \tag{2}$$

$$\alpha = \lambda \times \alpha_c + (1 - \lambda) \times \alpha_p \tag{3}$$

where λ is the gain ($0 < \lambda < 1$).

To decrease the size of the pipe, BMP does not need a prediction, since the value is deterministic, it just decreases to $Rnew = R_a \times D$, where D is constant and $D > 1$ (e.g. $D = 1.05$). We use D in order to avoid losing packets in the case of burstness.

Since the interdomain renegotiation process is slow, the HT is used to accept incoming requests during renegotiation time. The larger value of $R - HT$ is the larger amount of traffic to be observed during renegotiation time.

In order to avoid an oscillation problem, the new value of LT must be smaller than the previous value of HT.

$$LT_{new} < HT \Rightarrow L < R \times H / R_{new} \tag{4}$$

Figure 4 illustrate this algorithm more clearly.

5 BMP Scalability

5.1 Setup Protocol States Scalability

The setup protocol state consists of admission and policy control. In RSVP architecture [8, 9], each individual reservation setup request is passed to all the routers along the path for admission control and then passed to policy control modules managed by local policy or by a policy server. Thus, setup messages in RSVP cause various problems. First, each flow exposes a substantial delay before the setup is complete. Second, the performance of core routers is affected by the processing of a large number of setup messages. Third, large numbers of setup messages spends extra BW.

In our model, routers are not involved in the admission and policy control process, since both are made by the BMP of each domain. BMP knows in advance all the resource availability within its domain and the policy information of its customers, thus to a make decision it does not have to interfere with routers. It might communicate with routers to update its database, but this is not done based on individual demand. Request messages are directed to BMP without interacting with any other components. When BMP receives a reservation request message, it checks its policy database and the resource availability in the pipe to that destination. If any of these fail, the request is rejected; if they succeed, the new reservation is multiplexed into the existing pipe that carries traffic of the same class to the destination region.

The BMP establishes and modifies the size of a pipe more than its actual usage rate by considering near-future incoming requests, therefore any incoming end host requests can be multiplexed into the pipe without going through the interdomain slow process. Thus, interdomain control messages are based on aggregated demand, which has several advantages. First, avoiding the slow process in the interdomain results in a short connection setup time. Second, having an aggregated type of reservation reduces the processing scalability problems and bandwidth overhead in the interdomain caused by the large number of individual setup requests. When flows are of short duration, it is possible that many flows come and go within a short time. Requesting reservations for each of these individual flows separately is a burden for core routers. As shown in Figure 4 and Figure 5, The BMP does not change the size of the pipe as long as the utilization of the pipe is between thresholds.

We now illustrate the scalability of setup messages for both RSVP and BMP in the following scenario. For simplicity, we consider two domains, source (A), and destination (B), and we assume that individual reservation requests from A to B arrive according to the Poisson process with rate λ, and that the reservation lifetime is exponentially distributed with mean $1/\mu$. Figure 4 shows the traffic demand from A to B in terms of BW as well as the pipe size and its changes. In Figure 5 we show the number

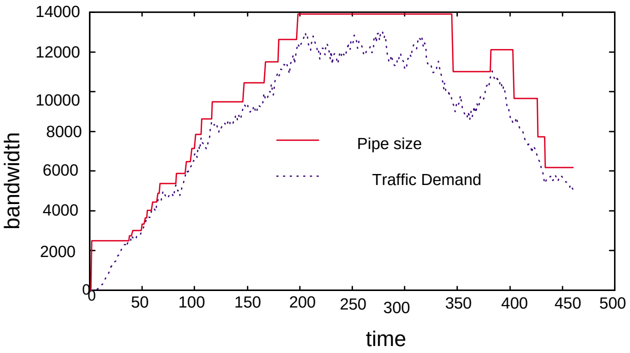

Figure 4: Traffic demand between two domains and the pipe size changes

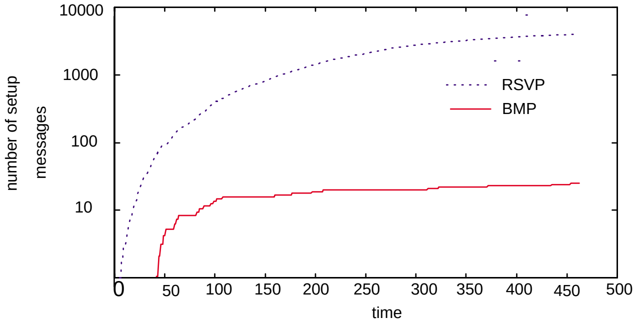

Figure 5: The number of setup messages between domains

of interdomain setup messages for RSVP and BMP. RSVP setup messages include reservation setup and reservation remove messages. BMP messages include the pipe size of increase and decrease messages.

5.2 Reservation States in Core Routers

In our model, routers keep the single state for all individual flows in the pipe, destination IP prefix, and DSCP. As we mentioned before, pipes from different sources can be merged if their destination address is the same. In this case, downstream routers simply maintain an aggregated pipe state. The number of states in the core routers are proportional to the number of destination regions. If the pipe is established between two ASs, the number of states the core routers need to maintain is proportional to the numbers of the ASs and DSCP. In some cases, it might be more reasonable to establish a pipe between stub domains, the region represented by CIDR [14], or between two virtual private networks (VPN). In this case we still get very good performance, as shown in Figure 8, because each of these specified regions contains large number of end systems.

Unlike all other aggregation models [4, 5, 10], in the entry and exit points of pipes there is no encapsulation and decapsulation associated with additional processing and bandwidth overhead.

In Figure 6 there are two ISP (ISP1, ISP2), each having n networks connected and each network having m hosts equally, with the number of DSCP being c. We assume that pipes are established based on networks' IP prefix. Also, for simple analysis we assume that requests are between

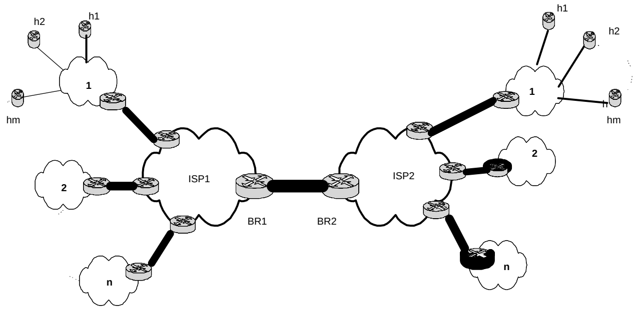

Figure 6: A network model

hosts of ISP1 and ISP2, and that all systems are multicast. The rate of reservation requests for each host from other domain's hosts is Poisson distributed with rate λ, and the reservation duration is exponential distributed with the mean of $1/\mu$

According to the above assumption, in RSVP the average number of sessions for each host is λ/μ, since the system is multicasting. If a host has at least one session connected, there will be a reservation state in BR1 and BR2 for it. According to Poisson distribution, the probability of there being at least one session for a host is:

$$(1 - \exp^{-\lambda/\mu}) \tag{5}$$

and the average number of reservation states in routers are binomially distributed with:

$$2mn(1 - \exp^{-\lambda/\mu}) \tag{6}$$

In a pipe based model, the average number of hosts in a pipe is:

$$m\lambda/\mu c \tag{7}$$

Similarly, the probability of a network having a pipe of a particular class is:

$$1 - \exp^{-m\lambda/\mu c} \tag{8}$$

and the average number of pipes in routers is:

$$2nc(1 - \exp^{-m\lambda/\mu c}) \tag{9}$$

$$Gain = m(1 - \exp^{-\lambda/\mu})/c(1 - \exp^{-m\lambda/\mu c}) \tag{10}$$

As shown in Figure 7, when $\lambda/\mu \to \infty$ the gain is m/c. The number of DSCP (c) is limited; for example today there are only 13 classes are being used, but the number of hosts connected to a network might be much higher. Similar results were obtained by BGRP [7] also.

In the case where pipes are established based on destination domain prefix, the gain will be higher. In the above case, we have only two ISPs, so the gain will be nm/c.

Figure 8 shows the simulation results for $m = 200$, $c = 5$, $n = 25$ and $\lambda/\mu = 0.65$

6 Conclusion

In this paper we have described interdomain resource reservation via BMP, which is responsible for both internal and external resource control. We do not specify intradomain reservation protocol, as each domain is free to

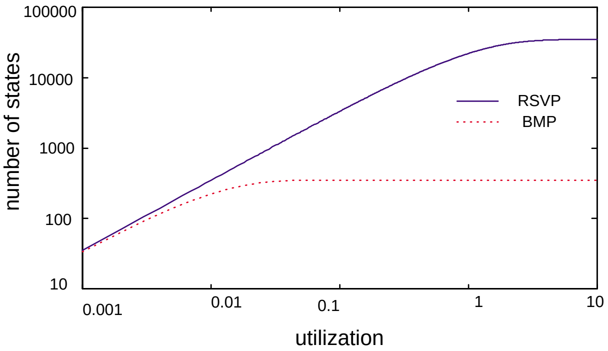

Figure 7: The number of reservations in RSVP and BMP with respect to utilization factor (λ/μ)

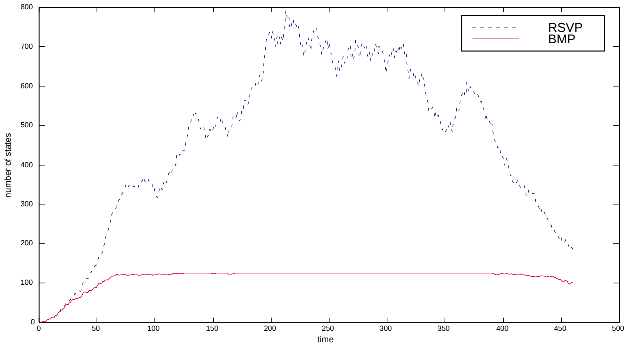

Figure 8: The number of reservations in routers for RSVP and BMP

choose its own protocol as long as it meets interdomain agreements. We have focused on both control and forwarding path scalability. Each BMP establishes pipes to possible destinations with a size greater than actual traffic rates. Thus, individual hosts can join and leave without going through the interdomain's slow process as long as the size of the pipe is between specified levels. By doing this, we succeeded in the following ways. First, the number of setup messages in the interdomain was substantially reduced. Second, the number of interdomain refresh messages is proportional to the number of pipes. Third, since individual requests are not involved in the interdomain process, the connection setup time is very short.

In the forwarding path we took advantage of Diffserv scalability. The number of states in core routers is proportional to the number of pipes relative to the number of destination regions. In the future, we will investigate interdomain policing and traffic engineering.

References

[1] J. Hwang. A Market-Based Model for the Bandwidth Management of IntServ-DiffServ QoS Interconnection, Ph.D. Thesis. University of Pittsburgh, 2000.

[2] J. Hwand, H. Kim and M. Weiss.“Interprovider Diffserv Interconnection Management Models in the Internet Bandwidth Comodity Markets,” *Telematics and Informatics*, Special Issues on EC, 2001.

[3] O. Scheln, S. Pink. “Aggregating Resource Reservations over Multiple Routing Domains.” *Proceedings of IFIP IWQoS’98*, Napa Valley, California, May 1998.

[4] S. Berson and S. Vincent. “Aggregation of Internet Integrated Services State.” *Proceedings of IWQoS*, May 1998.‘

[5] F. Baker. “Aggregation of RSVP for IP4 and IP6 Reservations,” Internet Draft, Internet Engineering Task Force, June 1999. Work in progress.

[6] R. Braden, Ed., L. Zhang, S. Berson, S. Herzog, and S. Jamin. “Resource ReSerVation Protocol (RSVP),” RFC2205, IETF, Sept. 1997.

[7] P. Pan, E, Hahne, H. Schulzrinne, “BGRP: A Tree-Based Aggregation Protocol for Interdomain Reservations,” Bell-Lab Technical Memorandum, and Columbia Technical Report CUCS-029-99. December 1999.

[8] S. Herzog. “Building Blocks for Accounting and Access Control.” RSVP: USC/ISI, November 1995.

[9] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. “A New Resource ReserVation Protocol.” IEEE Network, September 1993.

[10] R. Guerin, S. Blake, S. Herzog. eeee“Aggregating RSVP-Based QoS Requests.” Internet Draft, IETF, May 1998. Work in progress.

[11] A. Terzis, L. Wang, J. Ogawa, L. Zhang. “A Two-Tier Resource Management Model for the Internet.” Global Internet, Dec. 1999.

[12] K. Nichols, V. Jacobson “A Two-Bit Differentiated Services Architecture for” the Internet, 1998.

[13] QBone Bandwidth Broker Architecture. Work in Progress. February 2001.

[14] V. Fuller, T. Li, J. Yu, K. Varadhan, Classless Inter-Domain Routing (CIDR), rfc1519.

[15] L. Berger, “RSVP refresh reduction extensions,” Internet Draft, IETF, July 1999. Work in progress.

[16] Yehuda Afek, Menashe Cohen, Eyal Haalman, and Yishay Mansour. Dynamic Bandwidth Allocation Policies. INFOCOM 1996.

[17] V. Jacobson. “Congestion avoidance and control,” *Proc. ACM SigComm*, August 1988.

[18] G. Box and G. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, CA. 1970.

[19] A Framework for Integrated Services Operations over Diffserv Networks rfc2998.