Running Integrated Services over Differentiated Service Networks: Quantitative Performance Measurements *

Haowei Bai[a], Mohammed Atiquzzaman[b], and William Ivancic[c]

[a]AES Research Lab
Honeywell Aerospace
3660 Technology Drive, Minneapolis, MN 55418, USA
[b]School of Computer Science
University of Oklahoma, Norman, OK 73019-6151, USA
[c]NASA Glenn Research Center
21000 Brookpark Rd. MS 54-8, Cleveland, OH 44135, USA

ABSTRACT

Integrated Services (IntServ) and Differentiated Services (DiffServ) are two of the current approaches to provide Quality of Service (QoS) guarantees in the next generation Internet. IntServ aims at providing guarantees to end applications (individual connections) which gives rise to scalability issues in the core of the network. On the contrary, DiffServ is designed to provide QoS to aggregates, and does not suffer from scalability. It is therefore, believed that the combination of IntServ at the edge and DiffServ at the core will be able to provide QoS guarantees to end applications.[1,2] Although there have been several proposals on how to perform mapping of services between IntServ and DiffServ, there hasn't been any study to quantitatively show the level of QoS that can be achieved when the two networks are connected. The *objective* of this paper is to quantitatively demonstrate the QoS guarantees that can be obtained by end applications when IntServ is run over DiffServ. We have used goodput, drop ratio and non-conformant ratio of packets from the different services and the queue size of DiffServ router to determine the QoS obtained by packets belonging to different traffic classes.

Keywords: QoS, DiffServ, IntServ, Internetworking, Next Generation Internet

1. INTRODUCTION

Internet was designed for non-real time applications and hence does not provide Quality of Service (QoS) guarantees to applications. With the proliferation of the Internet, there is a strong interest in providing QoS to real-time applications in the next generation Internet. QoS includes guaranteed bandwidth, bounded packet delay, jitter and packet loss. QoS is generally implemented by different classes of service contracts for different users. A service class may provide low-delay and low-jitter service for customers who are willing to pay a premium price to run real-time applications such as video conferencing. Another service class may provide predictable services for customers who are willing to pay for reliability. Finally, the *best-effort* service provided by the current Internet will remain for those customers who only need connectivity.

The Internet Engineering Task Force (IETF) has proposed a few models to meet the demand for QoS. Notable among them are the Integrated Services (IntServ)[3] and Differentiated Services (DiffServ)[4] models. The IntServ model is characterized by resource reservation; before data is transmitted, applications must set up paths and reserve resources along the path. IntServ aims to support applications with different levels of QoS within the TCP/IP (Transport Control Protocol/Internet Protocol) architecture. IntServ however, requires the core routers to remember the state of a large number of connections giving rise to scalability issues in the core of the network. It is therefore *suitable at the edge network* where the number of connections is limited.[2]

The DiffServ model is currently being standardized to provide service guarantees to aggregate traffic instead of individual connections. The model does not require significant changes to the existing Internet infrastructure

*The work reported in this project was supported by NASA grant no. NAG3-2318.
Further author information: Send email to bai_haowei@htc.honeywell.com; or atiq@ou.edu.

or protocol. The DiffServ model utilizes six bits in the Type of Service (TOS) field of the IP header to mark a packet for being eligible for a particular QoS. DiffServ does not suffer from scalability issues, and hence is *suitable at the core of the network*.[2] It is therefore believed that a significant part of the next generation Internet will consist of IntServ at the edge and DiffServ at the core of the network. As a result, architectures with IntServ at the edge and DiffServ at the core to provide QoS to end applications have been proposed at the IETF[1,2] and the European ACTS project ELISA.[5–7]

Interconnection of IntServ and DiffServ, in order to exploit the individual advantages of IntServ (per flow QoS guarantee) and DiffServ (good scalability in the backbone), requires a mapping from IntServ traffic flows to DiffServ classes to be performed at the ingress to the DiffServ network. Some preliminary work has been carried out in the area of interconnecting IntServ and DiffServ. Balmer et. al.[8] present a concept for the integration of IntServ and DiffServ, and describe a prototype implementation using commercial routers. Budiardjo et. al.[9] suggests some prelimiary ideas on a packet forwarding algorithm to forward packets from the Guaranteed Class traffic of IntServ to Expedited Forwarding class of DiffServ. Chahed et.al.[10] shows that packet loss in the DiffServ network can result in bursty loss to the IntServ applications. Detti et.al.[6] have proposed an architecture for supporting IntServ and DiffServ and have carried out a scalability analysis. Harju et.al[11] present results to determine performance differences between IntServ and DiffServ, as well as some characteristics about their combined use. Mamais et.al[12] propose a new DiffServ class for carrying RSVP signalling originating from the edge IntServ domain. However, *the above studies do not present any numerical result to evaluate the QoS guarantee that can be achieved by end applications*. The authors are not aware of any study which *quantitatively* shows the QoS that can be achieved by IntServ end applications when IntServ and DiffServ are interconnected.

The *objective* of this paper is to *quantitatively measure* the QoS guarantees that can be obtained by end applications when IntServ is run over DiffServ. In our study, to map services from IntServ to DiffServ, we have proposed and implemented a mapping function between the two domains. Traffic arriving from the IntServ domain are appropriately mapped into the corresponding Behavior Aggregates of DiffServ, and then marked with the appropriate Differentiated Service Code Point (DSCP) for routing in the DiffServ domain. To determine the QoS obtained by end IntServ applications, we have used *goodput* of applications, the *queue size* at the router, and *drop ratio* of packets as *the performance criteria*. To prove the effectiveness of the admission control mechanism, we also measured the *non-conformant ratio* (the ratio of non-conformant packets to the in-profile packets). Our simulation results show that, without making significant changes to the IntServ or DiffServ infrastructure or without any additional protocol, it is possible to provide QoS to end applications when IntServ runs over a DiffServ network where the DiffServ is considered a network element to the edge IntServ networks. We have also shown that Guaranteed Load traffic can be protected from other traffic types during network congestion.

The rest of this paper is organized as follows. In Sections 2 and 3, we briefly present the main features of IntServ and DiffServ, respectively. In Section 4, we describe our approach of mapping traffic from IntServ to DiffServ, and the simulation configuration of IntServ over DiffServ. In Section 5, we analyze the simulation results to demonstrate that QoS can be provided to end applications by a IntServ over DiffServ configuration. Concluding remarks are finally given in Section 6.

2. INTEGRATED SERVICES

The basic framework of integrated services[2] is implemented by four components: the *signaling protocol*, the *admission control routine*, the *classifier* and the *packet scheduler*. This model requires explicit signaling mechanism to convey information to routers so that they can provide the requested resources to flows that require them. RSVP is one of the most widely known example of such a signaling mechanism which will be described in detail in Section 2.1. In addition to the *best effort* service, the integrated services model provides two services as follows.

- *Guaranteed service*[13] for applications requiring firm bounds on end-to-end queueing delays.
- *Controlled-load service*[14] for applications requiring services closely equivalent to that provided to uncontrolled *best effort* traffic under unloaded (lightly loaded) network conditions.

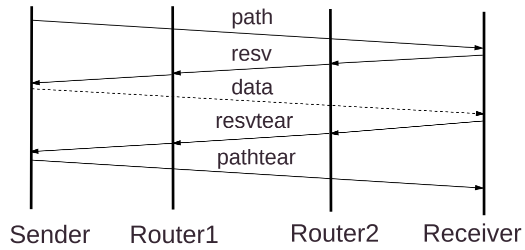

Figure 1. RSVP signaling for resource reservation.

The above two services will be discussed in Sections 2.2 and 2.3.

2.1. RSVP Signaling

RSVP is a signaling protocol to reserve network resources for applications. Figure 1 illustrates the setup and teardown procedures of the RSVP protocol.

The sender sends a **PATH** message to the receiver specifying the characteristic of the required traffic. Every intermediate router along the path forwards the **PATH** message to the next hop determined by the routing protocol. If the receiver agrees to the advertised flow, it sends a **RESV** message, which is forwarded hop by hop via RSVP capable routers towards the sender of the **PATH** message. Any intermediate router along the path may reject or accept the request. If the request is accepted, resources are allocated, and **RESV** message is forwarded. If the request is rejected, the router will send a **RESV-ERR** message back to the sender of the **RESV** message.

Receipt of a **RESV** message by the sender implies that resources have been reserved and data can be transmitted. To terminate a reservation, a **RESV-TEAR** message is transmitted to remove the resource allocation, and a **PATH-TEAR** message is sent to delete the path states in every router along the path.

2.2. Guaranteed Service

Guaranteed service guarantees that datagrams will arrive within the guaranteed delivery time and will not be discarded due to queue overflows, provided the flow's traffic stays within its specified traffic parameters.[13] The service provides assured level of bandwidth or link capacity for the data flow. It imposes a strict upper bound on the end-to-end queueing delay as data flows through the network. The packets encounter no queueing delay as long as they conform to the flow specifications. It means packets cannot be dropped due to buffer overflow and they are always guaranteed the required buffer space. The delay bound is usually large enough even to accommodate cases of long queueing delays.

2.3. Controlled-load Service

The controlled-load service does not accept or make use of specific target values for control parameters such as delay or loss. Instead, acceptance of a request for controlled-load service is defined to imply a commitment by the network elements to provide a service closely equivalent to that provided to uncontrolled (best effort) traffic under lightly loaded conditions.[14] The service aims at providing the same QoS under heavy loads as under unloaded conditions. Though there is no specified strict bound on delay, it ensures that a very high percentage of packets do not experience delays highly greater than the minimum transit delay due to propagation and router processing.

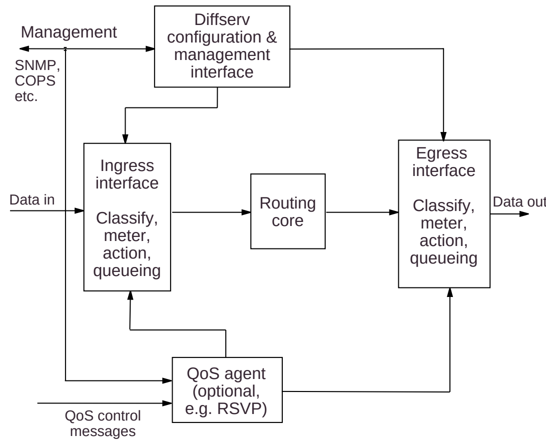

Figure 2. Major functional block diagram of a router.

3. DIFFERENTIATED SERVICES

The IntServ/RSVP architecture described in Section 2 can be used to provide QoS to applications. All the routers are required to be RSVP-aware and capable of performing admission control, MF classification and packet scheduling. These require maintaining of information for each flow at each router, giving rise to scalability concerns in large networks.[2] Because of the difficulty in implementing and deploying integrated services and RSVP, differentiated services is currently being developed by the IETF.[4]

Differentiated services (DiffServ) is intended to enable the deployment of scalable service discrimination in the Internet without the need for per-flow state and signaling at every hop. The premise of DiffServ networks is that routers in the core network handle packets from different traffic streams by forwarding them using different per-hop behaviors (PHBs). The PHB to be applied to a packet is indicated by a DiffServ Codepoint (DSCP) in the IP header of the packet.[15] The advantage of such a mechanism is that several different traffic streams can be aggregated to one of a small number of behavior aggregates (BA), each of which is forwarded using the same PHB at the router, thereby simplifying the processing and associated storage.[16] There is no signaling since QoS (Quality of Service) is invoked on a packet-by-packet basis.[16]

The DiffServ architecture is composed of a number of functional elements, including a small set of per-hop forwarding behaviors, packet classification functions, and traffic conditioning functions which includes metering, marking, shaping and policing. The functional block diagram of a typical DiffServ router is shown in Figure 2.[16] This architecture provides *Expedited Forwarding* (EF) service and *Assured Forwarding* (AF) service in addition to *best-effort* (BE) service as described below.

3.1. Expedited Forwarding (EF)

This service is also been described as *Premium Service*. The EF service provides a low loss, low latency, low jitter, assured bandwidth, end-to-end service.[17] Loss, latency and jitter are due to the queuing experienced by traffic while transiting the network. Therefore, providing low loss, latency and jitter for some traffic aggregate means there are no queues (or very small queues) for the traffic aggregate. At every transit node, the aggregate of the EF traffic's maximum arrival rate must be less than its configured minimum departure rate so that there is almost no queuing delay for these premium packets. Packets exceeding the peak rate are shaped by traffic conditioners to bring the traffic into conformance.

3.2. Assured Forwarding

This service provides a reliable services for customers even during network congestion. Classification and policing are first done at the edge routers of the DiffServ network. The assured service traffic is considered *in*-profile if

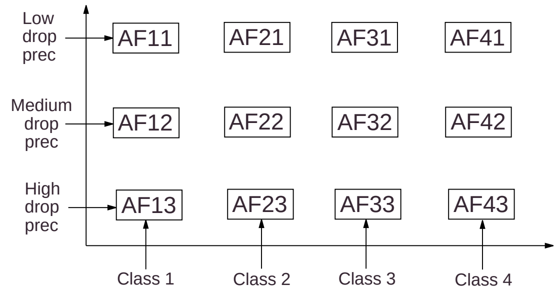

Figure 3. AF classes with drop precedence levels.

the traffic does not exceed the bit rate allocated for the service; otherwise, the excess packets are considered *out*-of-profile. The *in*-profile packets should be forwarded with high probability. However, the *out*-of-profile packets are delivered with lower priority than the *in*-profile packets. Since the network does not reorder packets that belong to the same microflow, all packets, irrespective of whether they are *in*-profile or *out*-of-profile, are put into an *assured queue* to avoid out-of-order delivery.

Assured Forwarding provides the delivery of packets in four independently forwarded AF classes. Each class is allocated a configurable minimum amount of buffer space and bandwidth. Each class is in turn divided into different levels of drop precedence. In the case of network congestion, the drop precedence determines the relative importance of the packets within the AF classes. Figure 3[18] shows four different AF classes with three levels of drop precedence.

3.3. Best Effort

This is the default service available in DiffServ, and is also deployed by the current Internet. It does not guarantee any bandwidth to customers, but can only get the available bandwidth. Packets are queued when buffers are available, and dropped when resources are over committed.

4. INTEGRATED SERVICES OVER DIFFERENTIATED SERVICE NETWORKS

In this section, we describe in detail the mapping strategy and simulation configuration that have been used in this paper to connect the IntServ and DiffServ domains.

4.1. Mapping Considerations for IntServ over DiffServ

In IntServ, resource reservations are made by requesting a service type specified by a set of parameters known as *Tspec* (Traffic Specification). Each set of parameters determines an appropriate priority level. When a connection with a certain priority level is mapped to the DiffServ domain, the following basic requirements should be satisfied.

- PHBs in the DiffServ domain must be appropriately selected for each requested service in the IntServ domain.
- The required policing, shaping and marking must be done at the edge router of the DiffServ domain.
- Taking into account the resource availability in DiffServ domain, admission control must be implemented for traffic arriving from the IntServ domain.

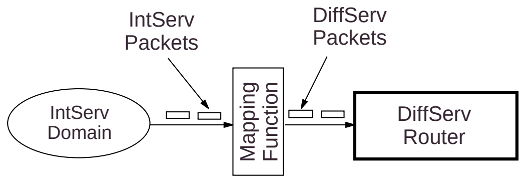

Figure 4. Mapping function for integrated service over differentiated service.

4.2. Mapping Function

The mapping function is used to assign an appropriate DSCP code to packets arriving from a flow specified by the *Tspec* parameters in the IntServ domain. This is to ensure that the appropriate QoS can be achieved for IntServ flows when running over a DiffServ domain. To achieve the above goal, we introduce a mapping function at the boundary router in the DiffServ domain as shown in Figure 4. Every packet in the flow from an IntServ domain has a *flow ID* indicated in the *flow-id* field in the IP (Internet Protocol) header. The *flow ID* attributed with the *Tspec* parameters is used to determine which flow the packet belongs to. Packets specified by *Tspec* parameters in IntServ domain are first mapped to the corresponding PHBs in the DiffServ domain by appropriately assigning a DSCP according to the mapping function. The packets are then routed in the DiffServ domain where they receive treatment based on their DSCP code. The packets are grouped into BAs in the DiffServ domain.

Table 1 shows an example mapping function which has been used in our simulation. As an instance, a flow in the IntServ domain specified by $r = 0.7$ Mb, $b = 5000$ bytes and *Flow ID*=0 is mapped to *EF* PHB (with corresponding DSCP code of 101110) in the DiffServ domain. r and b represent the token bucket rate and depth respectively.

Table 1. An example mapping function between InstServ and DiffServ which is used in our simulation.

Tspec	*Flow ID*	*PHB*	*DSCP*
r=0.7 Mb, b=5000 bytes	0	EF	101110
r=0.7 Mb, b=5000 bytes	1	EF	101110
r=0.5 Mb, b=8000 bytes	2	AF11	001010
r=0.5 Mb, b=8000 bytes	3	AF11	001010
r=0.5 Mb, b=8000 bytes	4	AF11	001010

The sender initially specifies its requested service using *Tspec*. Note that it is possible for different senders to use the same *Tspec*. However, they are differentiated by the *flow ID*. In addition, it is also possible that different flows can be mapped to the same PHB in DiffServ domain.

4.3. Simulation Configuration

To determine the QoS that can be provided to IntServ applications when they are run over DiffServ using the mapping function described in the previous section, we carried out simulation using the *ns* (Version 2.1b6) simulation tool from Berkeley.[19] The network configuration used in our simulation is shown in Figure 5.

All the links in Figure 5 are labeled with a (bandwidth, propagation delay) pair. The mapping function shown in Table 1 was implemented in the DiffServ edge router (see Figure 4). Constant Bit Rate (CBR) traffic was used for all IntServ sources in our simulation so that the relationship between the bandwidth utilization and bandwidth allocation can be evaluated. A admission control module (a part of edge router) was used to guarantee resource availability within DiffServ domain. To keep the figure simple, the admission control modules are not illustrated in Figure 5.

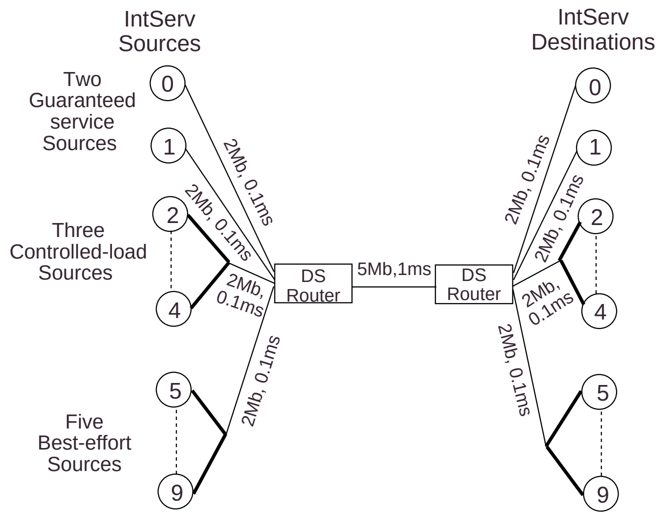

Figure 5. Network simulation configuration.

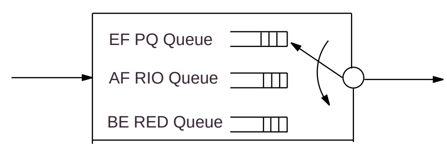

Figure 6. Queues inside the edge DiffServ router.

The EF queue in the DiffServ edge router was configured as a simple Priority Queue with Tail Drop. The AF queue was configured as a RIO queue, and the BE queue as a RED[20] queue as shown in Figure 6. The queue weights of EF, AF and BE queues were set to 0.4, 0.4 and 0.2 respectively. Since the bandwidth of the bottleneck link between the two DiffServ routers was 5 Mb, the above scheduling weights implies bandwidth allocations of 2 Mb, 2 Mb and 1 Mb for the EF, AF and BE links respectively.

5. SIMULATION RESULTS

In this section, results obtained from our simulation experiments are presented. The criteria used to evaluate our proposed strategy are first described followed by the explanations of our experimental setup and numerical results.

5.1. Performance Criteria

To determine the QoS obtained by end IntServ applications, we have used *goodput* of applications, the *queue size* at the router, and *drop ratio* of packets as the performance criteria. To prove the effectiveness of the admission control mechanism, we also measured the *non-conformant ratio* (the ratio of non-conformant packets to the in-profile packets). We present the results of measurements of the above quantities from our simulation experiments in Section 5.2.

5.2. QoS Obtained by Guaranteed Services

We use *three cases* (described in Sections 5.2.1, 5.2.2 and 5.2.3 respectively) to determine the QoS obtained by IntServ applications running over a DiffServ network. Table 2 shows the goodput and non-conformant ratio of each *Guaranteed service* source for the three different cases described in Section 5.2. Table 3 shows the drop ratio of the *Guaranteed service* sources measured at the scheduler for three cases. Figures 7, 8 and 9 show the queue size for each of the three case, from which the queuing delay and jitter can be evaluated. In the next three sections, we provide detailed explanation of these Tables.

Table 2. Goodput and Non-conformant Ratio of each *Guaranteed service* source.

Tspec	*Flow ID*	*Goodput (Kb/s)*			*Non-conformant Ratio*		
		Case 1	Case 2	Case 3	Case 1	Case 2	Case 3
r=0.7 Mb, b=5000 bytes	*0*	699.8250	699.8039	459.8790	0.00026	0.00026	0.00026
r=0.7 Mb, b=5000 bytes	*1*	699.8039	699.6359	1540.1400	0.00026	0.22258	0.00040

Table 3. Drop ratio of *Guaranteed service* traffic.

Type of traffic	*Case 1*	*Case 2*	*Case 3*
Guaranteed service Traffic	0.000000	0.000000	0.258934

5.2.1. Case 1: No congestion; no excessive traffic

The traffic generated by *Guaranteed service* sources (sources 0 and 1) were set to 0.7 Mb and 0.7 Mb, respectively. In this case, the traffic rate is equal to the bucket rate (0.7 Mb as shown in Table 1), which means that *there should not be any significant excessive IntServ traffic*. According to the network configuration described in Section 4.3, two *Guaranteed service* sources generate 1.4 Mb traffic which is less than the corresponding scheduled link bandwidth for *Guaranteed service* (EF in DiffServ domain) traffic (2 Mb). Under this scenario, *there should not be any significant congestion* at the edge DiffServ router.

This is an ideal case. As seen in Table 2, the goodput is almost equal to the corresponding source rate, and the non-conformant ratio is almost zero. Since there is no significant congestion, the drop ratio of each type of

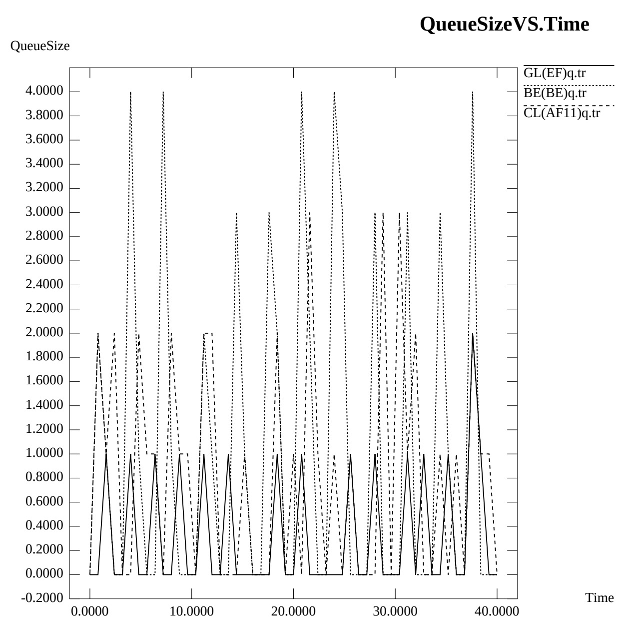

Figure 7. Queue size plots for *Case 1*.

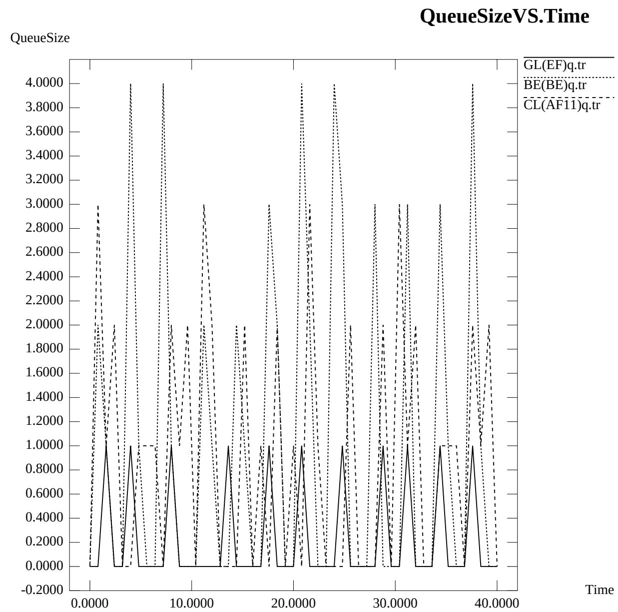

Figure 8. Queue size plots for *Case 2*.

source is zero as seen from Table 3. Figure 7 shows the queuing performance of each queue. Because this is an ideal case, the size of each queue is very small. Though the three queues have almost the same average size, we observe that the BE queue of IntServ (mapping to BE queue in DiffServ domain, according to the mapping function) has the largest jitter.

5.2.2. Case 2: No congestion; Guaranteed service source 1 generates excess traffic

The traffic generated by the *Guaranteed service* sources (sources 0 and 1) were set to 0.7 Mb and 0.9 Mb, respectively. In this case, the traffic rate of source 1 is greater than its corresponding bucket rate (0.7 Mb, shown in Table 1), which means that *source 1 generates excessive IntServ traffic*. According to the network configuration described in Section 4.3, two *Guaranteed service* sources generate 1.6 Mb traffic which is less than the corresponding scheduled link bandwidth for *Guaranteed service* (EF in DiffServ domain) traffic (2 Mb). Under this scenario, *there should not be any significant congestion* at the edge DiffServ router.

From Table 2, the goodput of source 0 for this case is equal to its source rate. However, the goodput of source 1 is equal to the corresponding token rate of 0.7 Mb, rather than its source rate of 0.9 Mb. Table 3 shows that the drop ratio of *Guaranteed service* is 0. The reason is that, in this case, there is no congestion for *Guaranteed service* traffic.

The non-conformant packet ratio of source 1, shown in Table 2, is increased as compared to case 1. It is because source 1 generates excessive traffic in this case. From Figure 8, we find that the average queue size of the *best effort* queue is far greater than the other two types of sources. In addition, the jitter of *best effort* traffic is also greater than the other two types of sources. The *Guaranteed service* traffic has the smallest average queue size and the smallest jitter. In addition, compared with Figure 7, the upper bound of the *Guaranteed service* queue is guaranteed, though source 1 generates more traffic than what it had reserved. This however, satisfies well the requirements of IntServ.[13]

5.2.3. Case 3: Guaranteed service gets into congestion; no excessive traffic

Traffic generated by *Guaranteed service* sources (sources 0 and 1) were set to 0.7 Mb and 2 Mb, respectively. To simulate a congested environment, we also set the token rate of sources 1 to 2 Mb. In this case, the traffic rate of source 1 is equal to its corresponding bucket rate (2 Mb), which means *there is no significant excessive*

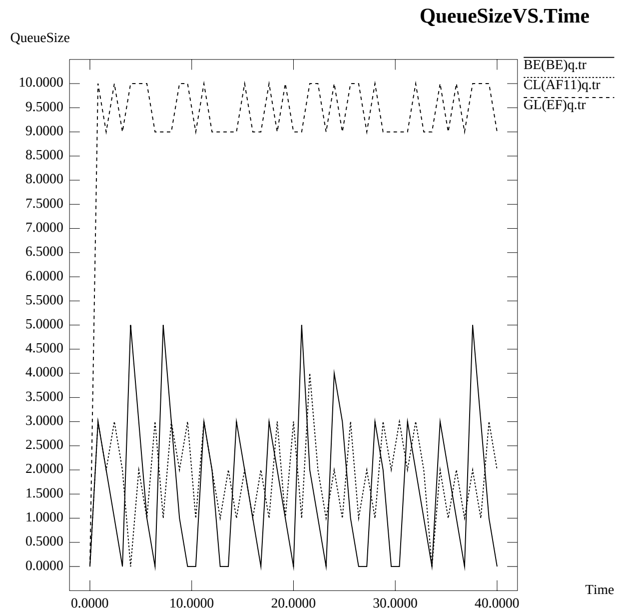

Figure 9. Queue size plots for *Case 3*.

IntServ traffic. According to the network configuration described in Section 4.3, two *Guaranteed service* sources generate 2.7 Mb traffic which is greater than the corresponding scheduled link bandwidth for *Guaranteed service* (EF in DiffServ domain) traffic (2 Mb). Under this scenario, *Guaranteed service traffic gets into congestion* at the edge DiffServ router.

Case 3 is used to evaluate our mapping function under congested environments. As expected, we find that the drop ratio (measured at scheduler) of the *Guaranteed service* traffic is increased, and the total goodput of the *Guaranteed service*, instead of being 2.7 Mb, is limited by the output link bandwidth assigned by the scheduler (2Mb). Since there is no excessive traffic, the non-conformant packet ratio of both the *Guaranteed service* sources are close to 0. From Figure 9, since we increased the token rate of one of the *Guaranteed service* sources, (source *1*), the upper bound of *Guaranteed service* is increased, which is reasonable. In addition, the *Guaranteed service* queue still has the smallest jitter.

5.3. QoS Obtained by Controlled-load Services

Because of the similarity between the results of *Guaranteed service* and *Controlled-load service, all our descriptions in Section 5.2 are focused on* Guaranteed service*. We only give out results for Controlled-load service without detailed explanations.*

We use *case 2* described in Section 5.2.2 as an example. As described in Section 4.3, we used three *Controlled-load service* sources in our simulation: sources *2*, *3* and *4*. The token bucket parameters are shown in Table 1. We set the source rate of sources *2* and *4* to 0.5 Mb, 0.5 Mb, respectively, and set the rate of source *3* to 0.7 Mb (greater than its token rate, 0.5 Mb). Therefore, *source* 3 *generates excessive traffic*. The total *Controlled-load service* traffic is 1.7 Mb, which is less than the scheduled link bandwidth; therefore, *there should not be any significant congestion*.

Table 4 shows the *goodput* and the *non-conformant ratio* of all the Controlled-load sources. Table 5 shows the *drop ratio* of Controlled-load sources measured at the scheduler. The queue size is shown in Figure 10. Note that though the non-conformant ratio of source 3 is much higher that the other two (shown in Table 4), the goodput of source 3 (shown in Table 4) is equal to its source rate (0.7 Mb). It is because the non-conformant packets are degraded and then forwarded, which is one of the forwarding schemes for non-conformant packets proposed in.[14]

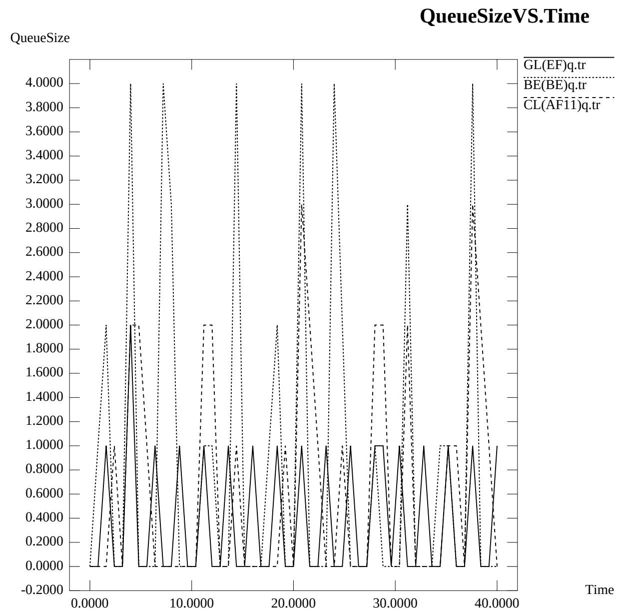

Figure 10. Queue size plots for *Controlled-load service*.

Table 4. Goodput and Non-conformant Ratio of each *Controlled-load service* source.

Tspec	*Flow ID*	*Case 2*	
		Goodput (Kb/s)	Non-conformant Ratio
r=0.5 Mb, b=8000 bytes	*2*	499.9889	0.00000
r=0.5 Mb, b=8000 bytes	*3*	700.0140	0.28593
r=0.5 Mb, b=8000 bytes	*4*	499.9889	0.00000

6. CONCLUSION

In this paper, we have evaluated the QoS that can be obtained by end applications when Integrated Services (IntServ) subnetworks are connected together using Differentiated Services (DiffServ) network. Traffic from various IntServ classes with different priorities are mapped into appropriate DiffServ services such that QoS can be guaranteed to individual applications.

Results show that QoS requirements of end applications can be met when IntServ is run over a well provisioned DiffServ. We have demonstrated this by measuring the *goodput* of end applications and *non-conformant ratio*, *queue size* and *drop ratio* of packets at the router. We found that the upper bound of queueing delay of *Guaranteed service* is guaranteed. Moreover, *Guaranteed Load* always has the smallest jitter, and is not affected by other traffic flows. *Controlled Load* has a smaller jitter and queue size than the *best effort* traffic. We have shown that reducing the priority of non-conformant packets before being forwarded through the network helps in protecting the QoS of conformant traffic.

Table 5. Drop ratio of *Controlled-load service* traffic.

Type of traffic	*Case 2*
Controlled-load Traffic	0.000000

ACKNOWLEDGMENTS

The authors thank NASA Glenn Research Center for support of this project.

REFERENCES

1. J. Wroclawski and A. Charny, "Integrated service mapping for differentiated services networks." draft-ietf-issll-ds-map-00.txt, March 2000.
2. Y. Bernet, R. Yavatkar, P. Ford, F. Baker, L. Zhang, M. Speer, R. Braden, B. Davie, J. Wroclawski, and E. Felstaine, "A framework for integrated services operation over diffserv networks." draft-ietf-issll-diffserv-rsvp-04.txt, March 2000.
3. R. Braden, D. Clark, and S. Shenker, "Integrated services in the internet architecture: an overview." RFC 1633, June 1994.
4. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services." RFC 2475, December 1998.
5. G. Eichler, H. Hussmann, G. Mamais, I. Venieris, C. Prehofer, and S. Salsano, "Implementing Integrated and Differentiated services for the Internet with ATM networks: A practical approach," *IEEEcommunication Magazine* **38**, pp. 132–141, Jan 2000.
6. A. Detti, M. Listanti, S. Salsano, and L. Veltri, "Supporting RSVP in a Differentiated service domain: An architectural framework and a scalability analysis," in *IEEE International Conference on Communications*, pp. 204–210, (Vancouver, Canada), June 6-10, 1999.
7. T. Do, G. Eichler, H. Hussmann, B. Koch, G. Mamais, C. Prehofer, S. Salsano, J. Tchouto, C. Tittle, P. Todorova, and I. Venieris, "ELISA: European linkage between Internet Integrated and Differentiated services over ATM," in *IEEE Workshop on QoS Support for Real-Time Intenet Applications*, (Vancouver, Canada), June 1999.
8. R. Balmer, F. Baumgarter, and T. Braun, "A concept for RSVP over diffserv," in *Proc. Nineth IC-CCN'2000*, (Las Vagas, NV), October 2000.
9. B. Budiardjo, B. Nazief, and D. Hartanto, "Integrated services to Differentiated services packet forwarding: Guaranteed service to expedited forwarding PHB," in *25th Annual IEEE Conference on Local Computer Networks*, pp. 324–325, (Tampa, FL), Nov 8-10, 2000.
10. T. Chahed, G. Hebuterne, and C. Fayet, "On mapping of QoS between Integrated services and Differentiated services," in *International Workshop on Quality of Service*, pp. 173–175, (Pittsburgh), June 5-7, 2000.
11. J. Harju and P. Kivimäki, "Co-operation and comparison of diffserv and intserv: Performance measurements," in *Proc.* 25^{th} *Conference on Local Computer Networks*, pp. 177–186, (Tampa, FL), November 2000.
12. G. Mamais, M. Markaki, G. Politis, and I. Venieris, "Efficient buffer management and scheduling in a combined IntServ and DiffServ architecture; a performance study," in *Second International Conference on ATM*, pp. 236–242, (Colamar, France), June 1999.
13. S. Shenker, C. Partridge, and R. Guerin, "Specification of guaranteed quality of service." RFC 2212, September 1997.
14. J. Wroclawski, "Specification of the controlled-load network element service." RFC 2211, September 1997.
15. K. Nichols, S.Blake, F. Baker, and D.Black, "Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers." RFC 2474, December 1998.
16. Y. Bernet, S. Blake, D. Grossman, and A. Smith, "An informal management model for diffserv routers." draft-ietf-diffserv-model-04.txt, July 2000.
17. V. Jacobson, K. Nichols, and K. Poduri, "An expedited forwarding PHB." RFC 2598, June 1999.
18. J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured forwarding PHB group." RFC 2597, June 1999.
19. V. P. U. Berkeley/LBNL, "ns v2.1b6: Network simulator." http://www-mash.cs.berkeley.edu/ns/, January 2000.
20. S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking* **1**, pp. 397–413, August 1993.