Flow-Based Admission Control Algorithm in the DiffServ-Aware ATM-Based MPLS Network

Gyu Myoung Lee, Jun Kyun Choi, Mun Kee Choi, Man Seop Lee, and Sang-Gug Jong

This paper proposes a flow-based admission control algorithm through an Asynchronous Transfer Mode (ATM) based Multi-Protocol Label Switching (MPLS) network for multiple service class environments of Integrated Service (IntServ) and Differentiated Service (DiffServ). We propose the Integrated Packet Scheduler to accommodate IntServ and Best Effort traffic through the DiffServ-aware MPLS core network.

The numerical results of the proposed algorithm achieve reliable delay-bounded Quality of Service (QoS) performance and reduce the blocking probability of high priority service in the DiffServ model. We show the performance behaviors of IntServ traffic negotiated by end users when their packets are delivered through the DiffServ-aware MPLS core network. We also show that ATM shortcut connections are well tuned with guaranteed QoS service. We validate the proposed method by numerical analysis of its performance in such areas as throughput, end-to-end delay and path utilization.

I. INTRODUCTION

One of the today's most pressing challenges in designing IP networks is to meet users' Quality of Service (QoS) requirements. The need to support QoS-sensitive applications has led to the development of new architectures. Internet architecture that guarantees some QoS can be modeled by Integrated Services (IntServ) and the Differentiated Services (DiffServ) [1]-[4].

In the Resource Reservation Protocol (RSVP), the IntServ model is used for signaling QoS requests from application to network. With the DiffServ model, user flows are aggregated into a small set of Class of Services (CoSs). Since IntServ and DiffServ models focus, respectively, on reservation and scalable service differentiation, it is advantageous to combine both for an overall solution: a scalable and guaranteed end-to-end IntServ service model and DiffServ core network with individual QoS for flows.

Integrating different types of traffic in a single network requires an admission control mechanism which operates according to a resource reservation mechanism. We propose an optimal admission control algorithm which uses flow-based classification to meet users' QoS requirements. In addition, we consider the Asynchronous Transfer Mode (ATM) shortcut connection in an ATM-based Multi-Protocol Label Switching (MPLS) network to reduce end-to-end delay [5]-[8].

In this paper, we present our design of an ATM-based MPLS core network for accommodation of the IntServ and DiffServ models. These models require signaling support for the association of the desired category. The label and each packet belonging to a stream needs to carry the information of the desired service category.

Our flow-based admission control algorithm for the Int-

Manuscript received July 3, 2001; revised Oct. 24, 2001.

The research featured in this paper is sponsored by Korea Science and Engineering Foundation (KOSEF), and partially supported by Korean Ministry of Information and Communication (MIC).

Gyu Myoung Lee (phone: +82 42 866 6122, e-mail: gmlee@icu.ac.kr), Jun Kyun Choi (e-mail: jkchoi@icu.ac.kr), Mun Kee Choi (e-mail: mkchoi@icu.ac.kr), and Man Seop Lee (e-mail: leems@icu.ac.kr) are with the Information and Communications University, Daejeon, Korea.

Sang-Gug Jong (e-mail: sgjong@kt.co.kr) is with Korea Telecom, Daejeon, Korea.

Serv/DiffServ model on the ATM-based MPLS network optimizes the network by using admission control according to traffic classification, which is suitable for multiple service class environments. The proposed flow-based admission control considers QoS and buffer statistics. This algorithm makes an admission decision in accordance with the conventional measurement-based admission control. Consequently, this algorithm can achieve two objectives: reliable delay bound QoS and high resource utilization. Also, this reduces the blocking probability of high priority service class flow in the operating DiffServ. The proposed approaches guarantee a hard QoS that satisfies all performance parameters (bandwidth, latency, jitter, etc.) using an ATM shortcut connection for IntServ guaranteed service class. They also increase resource utilization according to per class QoS conditions for other service classes. In the numerical analysis, we analyze loss probability and delay as well as priority queuing behavior. We also examine the relationship between blocking probability and the actual load. If the blocking probability increases, the actual load decreases and network efficiency improves. However, the flow acceptance rate decreases. In addition, we apply the ATM shortcut connection for the guaranteed service class. An ATM shortcut has a number of advantages: higher throughput, shorter end-to-end delay, and reduced router load. Numerical results of the proposed shortcut algorithm can improve their performance.

In Section II, the ATM-based MPLS network architecture and flow-based admission control algorithm are discussed. In Section III, numerical analyses for the proposed network model and admission control algorithm are taken. Finally, numerical results are presented in Section IV.

II. ATM-BASED MPLS NETWORK ARCHITECTURE AND FLOW-BASED ADMISSION CONTROL ALGORITHM

In this section, we discuss the DiffServ-aware MPLS/ATM core network for accommodation of the conventional IntServ and DiffServ. We also investigate integrated packet scheduling and suggest a flow-based admission control algorithm.

1. The DiffServ-Aware ATM-Based MPLS Network Architecture

Figure 1 illustrates the architectural model of the ATM-based MPLS network. Its overlay architecture is characterized by layering the ATM shortcut network and the MPLS core network with the DiffServ model. The MPLS core network is composed of Label Edge Routers (LERs) and Label Switching Routers (LSRs). The LER is located at the ATM edge switch as an MPLS-aware ingress/egress router. The LER performs label binding based on the Label Information Base (LIB). The LSR performs a label swapping function and reserves the resources to build a Label Switched Path (LSP) using a Constraint-based Routing Label Distribution Protocol (CR-LDP) or RSVP.

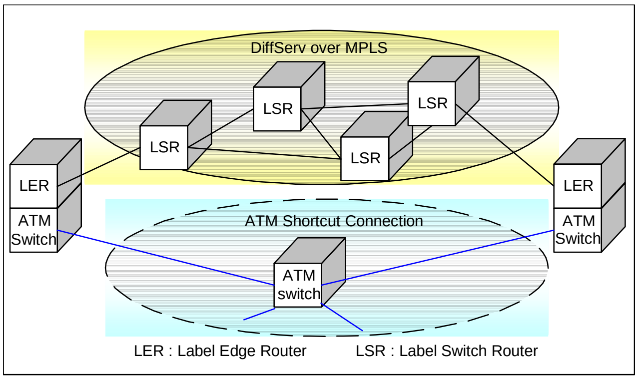


Fig. 1. The proposed ATM-based MPLS network architecture.

The advantage of the ATM-based MPLS core network architecture is that it combines both the IntServ and DiffServ network models for scalable service differentiation. Figure 2 shows the combined IntServ/DiffServ network model using the ATM-based MPLS network. It supports multiple traffic types in a single network. The core network is composed of ATM–based MPLS domains that support the DiffServ paradigm. The Access Network Domain considers three main service domains: the IntServ model, the DiffServ model, and the Best Effort service model. The ingress LER performs admission control and integrated packet scheduler functions. In the IntServ model, multiple classes of traffic can be assured of different QoS profiles. According to availability of resources, the network reserves the resources and sends back a positive or negative acknowledgement. In the DiffServ model, traffic is classified into different behavior aggregates. Packets, which enter into ingress LER at the border of the core network, are assigned a single Differentiated Service Code Point (DSCP). They are forwarded as per hop behaviors associated with their codepoints. In the Best Effort service model, it delivers packets to their destination without any bounds on delay, latency, jitter, etc.

The DiffServ model in the core network may not achieve the performance that can be obtained by the IntServ model at the edge of ATM network. The IntServ model takes care of end-to-end behavior in its intrinsic definition, while the DiffServ model basically specifies “local” behavior that must be somehow composed to achieve end-to-end significance. Our proposed network model considers the integrated solutions of the two approaches. This network model includes a scalable end-to-end IntServ model with acceptable service guarantees in the core network.

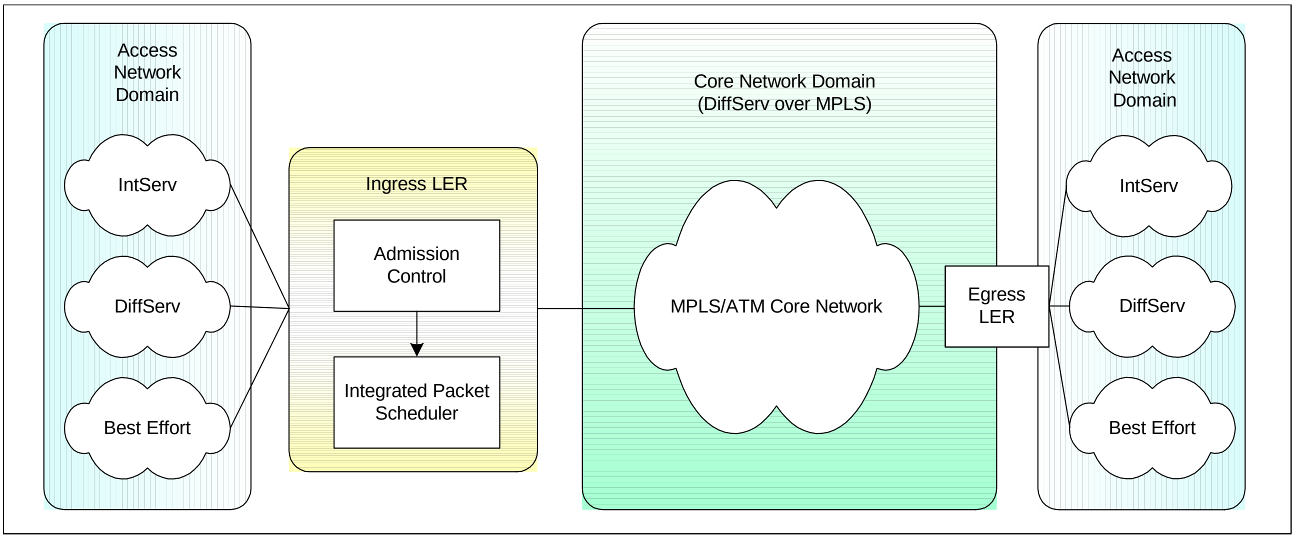


Fig. 2. The DiffServ/IntServ models on the ATM-based MPLS network.

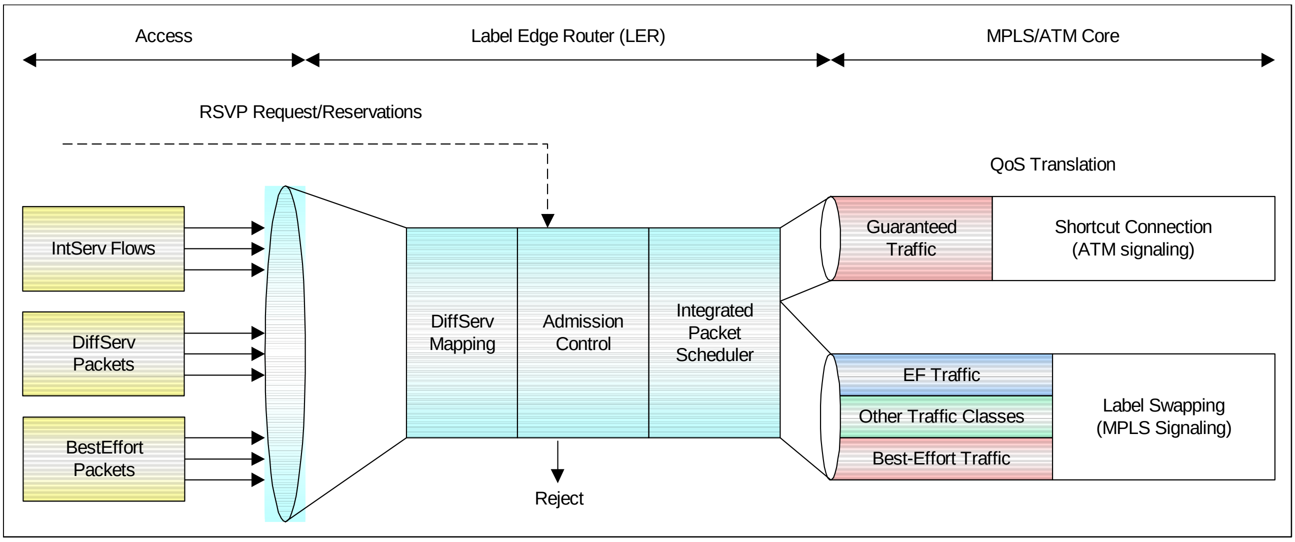


Fig. 3. The proposed traffic flow model of the MPLS network.

2. Traffic Flow Model of the ATM-Based MPLS Network

The QoS-enabled LERs should function like an IntServ/DiffServ and/or Best Effort capable router at the edge node and like a DiffServ router at the core network. The IntServ model makes a bandwidth reservation along the path and performs policing on the packets. The DiffServ model simplifies the forwarding functions in the core network and no policing occurs. The traffic flow model of the LER is proposed in Fig. 3. Incoming IP packets are classified as IntServ flows, DiffServ flows, or Best Effort flows. They are processed by the integrated packet scheduler using admission control. The integrated packet scheduler performs appropriate queuing disciplines based on service classes. By introducing flow concept, IP switching has been developed as a set of methods to reduce router workload and to offer Quality or Grade of Service in the network.

As illustrated in Fig. 3, we propose that flows can be handled in the core network by two methods. One uses the ATM shortcut connection. The ingress LER performs a QoS translation and maps the RSVP traffic parameters into ATM parameters. The other uses the Label swapping technology. For guaranteed traffic, we propose the ATM shortcut connection. An ATM shortcut connection can provide higher throughput, shorter end-to-end delay, reduced router load, and path utilization. In the ATM-based MPLS networks, a shortcut connection is set up using Virtual Path Identifier (VPI)/Virtual Channel Identifier (VCI) values which are allocated with proper admission control. For the guaranteed flows, the LER sets up an end-to-end shortcut connection.

3. DiffServ Model for the Integrated Packet Scheduler

The DiffServ in the MPLS network needs to carry packets at the desired service category. A separated LSP is created for each Forwarding Equivalence Class (FEC) and scheduling aggregate pair. Differentiation in treatment of packets from different behavior aggregates has to be implemented by mapping drop precedence. Thus, when the underlying technology is ATM, it can only support two levels of drop precedence. However, by marking the use of the EXP field in the "shim" header for the top label stack entry, support for all the drop precedence can be provided in MPLS clouds.

A "shim" header cannot be used with ATM because this would involve doing segmentation and re-assembly at each ATM-LSR in order to read the DSCP. Hence, the DSCP in the IP header is not accessible by the ATM hardware responsible for the forwarding. Therefore, two alternative solutions may be considered: either have some part of the ATM cell header mapped to the DSCP, or use an LDP.

In the first approach, the most likely solution is to use the VPI and part of the VCI of the ATM cell header as the label, and to use the remaining eight least significant bits of the VCI to map the DSCP. Then, all that is needed is a functional component in the interior DiffServ-enabled ATM LSRs to perform the appropriate traffic management mechanisms on the cells by interpreting the DSCP correctly with respect to the Per-Hop Behavior (PHB). In the second approach, which is more likely for future deployment, the DSCP is mapped to an LSP at the ingress of the MPLS domain. This means that for each DSCP value/PHB a separate LSP will be established for the same egress LSR. Therefore, if there are *n* classes and *m* egress LSRs, $n \times m$ LSPs and *n* labels for each of the *m* FECs need to be set up. The packets belonging to streams with the same DSCP and FEC will be forwarded on the same LSP. In other words, the label is regarded as the behavior aggregate selector.

In Fig. 4, the DiffServ model in the proposed MPLS router system is shown. This is similar to conventional DiffServ architecture but has many additional functions like service mapping. While incoming packets are composed of IntServ flows and the Best Effort flows, the LER is able to convert IntServ requests into DiffServ traffic classes. Table 1 shows the DSCP of incoming DiffServ packets according to class.

In Fig. 4, the Multi-Field (MF) classifier checks IntServ/DiffServ and Best Effort through IP flow classification. A traffic conditioner is a part of a network node that takes the node's ingress packets as its input and places the packets in its output in an order that best satisfies the forwarding requirements set

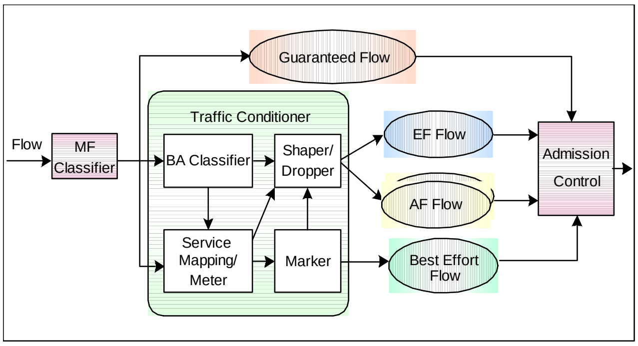


Fig. 4. The DiffServ model in the proposed MPLS router system.

Table 1. The DiffServ classes.

Service class	Control	Codepoint
Expedited Forwarding (EF)	Controlled	101100, …
Assured Forwarding (AF)	Low drop precedence	001010, 010010, 011010, 100010
	Medium drop precedence	001100, 010100, 011100, 100100
	High drop precedence	001110, 010110, 011110, 100110
Best Effort (BE)	Not controlled	000000

for the packets and uses the network's resources in the best possible way. The Behavior Aggregate (BA) classifier classifies packets based on the DSCP only.

If incoming packets are in the Best Effort service class, service mapping is performed according to applications. The low quality services of each group (i.e., multimedia conferencing, File Transfer and WWW navigation) can be mapped to a Best Effort service class. Applications that request QoS, such as video telephony, premium multimedia conferencing, on demand retrieval, and premium WWW navigation, are mapped to a guaranteed flow and will be assigned to one of the DiffServ classes. Figure 5 shows the DiffServ mapping algorithm in detail. Incoming packets are mapped into five service classes: Guaranteed, Expedited Forwarding (EF), two types of Assured Forwarding (AF), and Best Effort.

In the Integrated Packet Scheduler for admission control, there are usually multiple logical queues destined to each outgoing link. Figure 6 shows the queuing architecture of the proposed Integrated Scheduler for Guaranteed, EF, AF1, AF2, and Best Effort traffic. Each queue may contain packets from one flow or a class of flows. The basic operation of scheduling is to decide, at a particular moment, which packet in all these queues should be transmitted onto the outgoing link. The decision

```
/* go through IP flow classification */

if  Integrated Services
        /* traffic characteristic criteria */
   then
        if Guaranteed Services
              map the RSVP traffic parameters into ATM parameters
              /* prepare ATM shortcut connection */
        else if Controlled-Load Services
              /* goto DiffServ Traffic Conditioner */
              mark EF or AF1/AF2  PHBs
         else Best-Effort Services
               /* goto DiffServ Traffic Conditioner */
               mark BE PHBs

else if  Differentiated Services (marked packet)
        /* IP TOS Field */
   then
         if Expedited Fowarding (EF) PHBs
               mark EF PHB
         else if Assured Forwarding (AF) PHBs
               mark AF1 or AF2 PHB
         else Best-Effort (BE) PHBs
               mark BE PHB

else  Best-Effort Services
         /* service mapping */
    then
         if Guaranteed Services
              map the RSVP traffic parameters into ATM parameters
              /* prepare ATM shortcut connection */
         else other Services
               /* goto DiffServ Traffic Conditioner */
              mark EF/AF1/AF2 or BE PHB
```

Fig. 5. DiffServ mapping algorithm.

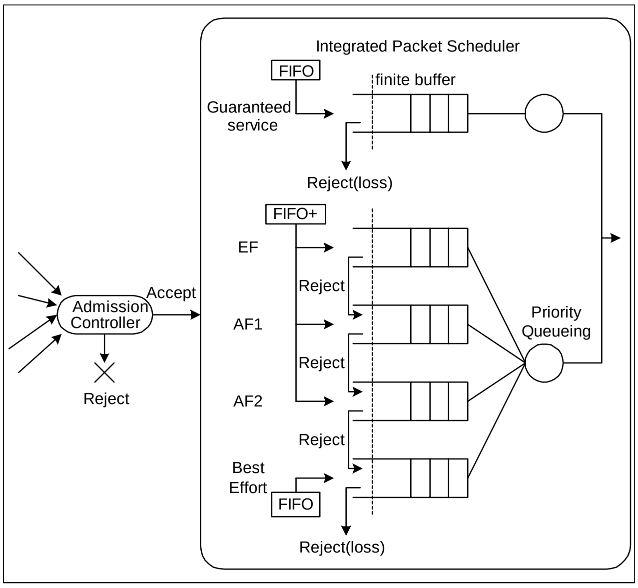


Fig. 6. The queueing model of the proposed integrated packet scheduler.

algorithm directly affects delay performance and the buffering strategy directly affects packet loss performance. The integrated packet scheduler is realized by applying a Priority Queuing (PQ) algorithm and an admission control function. For Guaranteed Service, we consider per flow queuing for each individual Guaranteed flow. There is no buffer sharing between Guaranteed flows and other traffic flows. The separate buffer for the Guaranteed flow offers an excellent balance between traffic isolation and buffer sharing.

For the EF and the AF flows, we adopt a First In First Out (FIFO) + queue [9]. The idea behind FIFO+ is to try to limit the accumulation of delay across hops and to bring down the worst-case delay. Each hop group flows into classes. Each class tracks its average queuing delay for the hop. For each packet, the hop computes the difference between the queuing delay the packet experiences and the average queuing delay for the packet's class. It then adds (or subtracts) this difference from an offset field in the packet's header. Over the hops, this offset will record how far ahead or behind the packet is from its class's average. Each hop is required to schedule a packet in its queues as if it had arrived at the packet's real arrival time plus the packet's offset.

It is important to keep in mind that FIFO+, unlike Weighted Fair Queuing (WFQ), is a statistical multiplexing technique. Because it does not provide strict isolation, FIFO+ may occasionally violate its delay requirements. Applications that require strict delay guarantees will have to be satisfied with the longer delay bounds of queuing schemes like WFQ. For Best Effort flows, we employ a common FIFO shared queue. There is no QoS commitment to each individual Best Effort flow.

We can define four priorities: EF, AF1, AF2, and BE. Incoming traffic is classified and stored at four separate queues. The advantage of the priority queue is the absolute preferential treatment that always gives top priority to mission critical traffic in the event of congestion.

4. Flow-Based Admission Control Algorithm

Admission control is the process that decides whether a newly arriving request for service from a network element can be granted. It must be performed on any service that wishes to offer absolute quantitative bounds on performance. The precise criteria for making the admission control decision are specific to each particular service with some statements of performance. The goal of the admission control algorithm is to meet users' quality of service requirements. The admission control algorithm is ultimately constrained by the service commitments [10]-[11].

Figure 7 shows the proposed admission control algorithm. The proposed flow-based admission control algorithm decides on two terms: QoS condition and resource condition. It can be classified into the conventional measurement-based admission control. The QoS parameters are offered on a per flow basis, corresponding to the number of related performance objectives.

The proposed flow-based admission control has the following assumptions. Each service class such as Guaranteed Service (GS), EF, AF1, AF2, Best Effort (BE) is divided at the edge node with the negotiated users' requirements of delay, loss, and bandwidth.

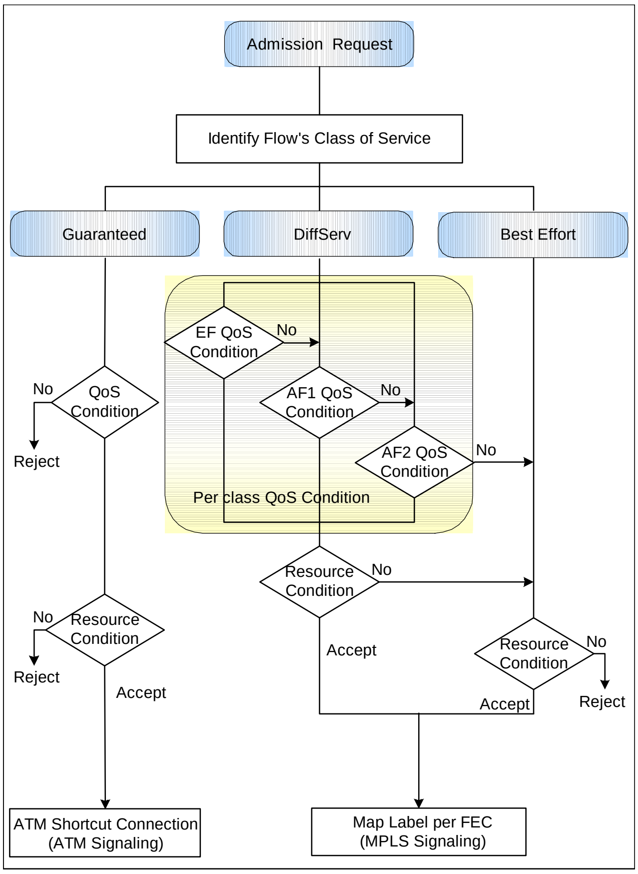


Fig. 7. The proposed flow-based admission control algorithm.

In the GS class, the edge node checks QoS and resource. It rejects connection admission when it can't meet service commitment. In the DiffServ service classes, we use a QoS condition per class.

To meet the QoS condition, the edge node decides whether the estimated overall aggregated traffic flow's delay $(\hat{r}_{overall})$ according to system queuing behavior is satisfied with a delay bound (D_{bound}) that users are requested. New connections are accepted only if they can be guaranteed to meet the requested performance bounds. Here, we define the estimated loss rate $(\hat{R}_{loss})$ with the users' required loss rate $(R_{required})$. The measured performance directly affects the flow-based admission control and resulting network utilization. This flow-based admission control algorithm can achieve two objectives: delay and utilization.

III. PERFORMANCE ANALYSIS

1. M/M/1/K with Bulk Arrival System Modeling

To analyze the proposed flow based traffic admission control algorithm, we model the queuing system with a finite buffer and bulk arrival. We investigate the queuing behavior with four priorities - EF, AF1, AF2, and BE - in terms of blocking probability, end-to-end throughput, and average message transfer delay.

If the packet arrivals that occur in disjointed time intervals are independent and identically distributed, we assume that the proposed system is modeled as an M/M/1/K queue with bulk arrival. This means that the aggregated arrival flow is modeled after a Poisson process and each packet flow is exponentially distributed. We also assume that message size is modeled by geometric distribution as a bulk arrival process since Internet traffic tends to be volatile. We assume the system can hold at most a total of K customers with finite storage. Figure 8 shows the state transition model of an M/M/1/K system with bulk arrivals, where λ is the mean arrival rate and μ is the average service rate.

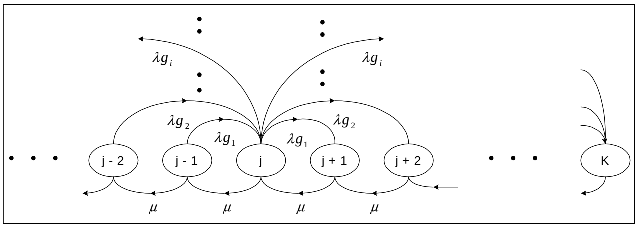


Fig. 8. The M/M/1/K state transition model with bulk arrival.

To model the bulk arrival process, g_i is given by

$$g_i \equiv P[\text{bulk size is } i] \tag{1}$$

where $\lambda g_1 + \lambda g_2 + \cdots + \lambda g_K = \lambda \sum_{i=1}^{K} g_i = \lambda$. Let us define the p_i, as the steady state probability of the i-th state, which can be obtained by

$$\left(\sum_{i=1}^{K-j} \lambda g_i + \mu \right) p_j = \mu p_{j+1} + \sum_{i=0}^{j-1} p_i \lambda g_{j-i} \qquad 1 \le j \le K \tag{2}$$

$$\mu p_1 = \lambda \sum_{i=1}^{K} g_i p_0 = \lambda p_0 . \tag{3}$$

Using the method of z-transforms, we have

$$\left(\sum_{i=1}^{K-j} \lambda g_i + \mu \right) \left(P(z) - p_0 \right) = \frac{\mu}{z} \left(P(z) - p_0 - p_1 z \right) + \lambda P(z) G(z), \tag{4}$$

where $P(z) \equiv \sum_{j=0}^{\infty} p_j z^j = \sum_{j=0}^{K} p_j z^j$

and $G(z) \equiv \sum_{i=0}^{\infty} g_i z^i = \sum_{i=1}^{K} g_i z^i$.

The system utilization factor ρ can be given in terms of the bulk arrival rate as

$$\rho = \frac{\lambda G'(z)\big|_{z=1}}{\mu} . \tag{5}$$

In the case of a fixed bulk size, g_i is given by

$$g_i = \begin{cases} 1 & i = r \\ 0 & i \neq r \end{cases} . \tag{6}$$

Then, the system can be modeled and analyzed by an M/E$_r$/1 queuing system with r stages of service time [12]. In the special case of $r = 1$, p_i, the steady state probability of the i-th state is given by

$$P_k = \begin{cases} \dfrac{1-\left(\dfrac{\lambda}{\mu}\right)}{1-\left(\dfrac{\lambda}{\mu}\right)^{K+1}}\left(\dfrac{\lambda}{\mu}\right)^k & 0 \le k \le K \\ 0 & otherwise. \end{cases} \tag{7}$$

Now, the loss probability can be calculated in p_K value since the system has a finite buffer of K. The average number of customers in the system $\overline{N}$ is given by

$$\overline{N} = \sum_{k=0}^{K} k p_k . \tag{8}$$

And $\overline{T}$, the average system transfer delay, can be obtained as follows:

$$\overline{T} = \frac{\overline{N}}{\lambda} = \frac{1}{\lambda}\sum_{k=0}^{K} k p_k . \tag{9}$$

2. Priority Queuing Analysis

We analyze the priority queuing model for DiffServ. We assume P classes (where $p = 1, 2, ..., P$) with non-preemptive priority. We also assume that messages with priority class p arrive at the Poisson process with rate λ_p. Then, the inter-arrival time distribution of messages with priority class p is given by $A_p(t) = 1 - e^{-\lambda_p t}$ $t \ge 0$ .With the non-preemptive priority queuing system $E[W_p]$, the average waiting time of messages of priority class p , is given by [12],

$$E[W_P] = E[T_0] + \sum_{p=1}^{P} E[T_p] + \sum_{p=1}^{P-1} E[T'_p], \tag{10}$$

where T_0 is a residual life time to complete the current service, T_p is the message service time of priority p, and T'_p is the message service time of higher priority during the waiting interval of messages with priority class p. Here, $E[T_0]$ is just the weighted sum over all priority classes, which is given by

$$E[T_0] = \frac{\lambda}{2} E(\tau^2) = \frac{1}{2}\sum_{p=1}^{P} \lambda_p E(\tau_p^2), \tag{11}$$

where the aggregate arrival rate is given by $\lambda \equiv \sum_{p=1}^{P} \lambda_p$, and τ_p is the message service time with priority p.

3. Blocking Probability and Throughput Analysis

We analyze the blocking probabilities in the proposed flow admission control algorithm. The blocking probability depends on the probability, P_{delay} , such that a delay time in the system is less than the negotiated threshold value and the probability, P_{loss} , such that the loss in the system is less than the threshold value, and the probability, $P_{resource}$, such that the buffer resource in the system is less than the threshold value, which can be represented by

$$P_{blocking} = 1 - P_{QoS} P_{resource} , \tag{12}$$

where P_{QoS} is the probability that is satisfied with the QoS condition. It can be modeled by

$$P_{QoS} = P_{delay} \cdot P_{loss} = P[\hat{T}_{overall} \le D_{bound}] \cdot P[\hat{R}_{loss} \le R_{required}] \tag{13}$$

where $\hat{T}_{overall}$ is the estimated overall delay and $\hat{R}_{loss}$ is the estimated loss ratio, D_{bound} is the negotiated delay bound and $R_{required}$ is the negotiated packet loss rate.

In (12), $P_{resource}$ is the probability that is satisfied with the resource condition. Let $\hat{\lambda}_{i,\tau}$ be the instantaneous estimated arrival rate of flow i at time τ . Assume that $\hat{\lambda}_{i,\tau}$'s are independent, identically distributed. Let $\sum_{i=1}^{n} \hat{\lambda}_{i,\tau}$ be the instantaneous estimated arrival rate of n flows at the specific link. While the packet arrival rate is specified as a function of the parameters, {peak rate, average rate, Maximum Burst Size (MBS)}, we calculate $P_{resource}$ approximately as

$$P_{resource} = P\left[\sum_{i=1}^{n} \hat{\lambda}_{i,\tau} \le \mu^* C^*\right], \quad (14)$$

where $\mu^* C^*$ is the actual available bandwidth, i.e., the packets serviced per second since μ^* is the actual service rate of a packet and C^* is the actual channel capacity in bit per second. With the relationship of the blocking probability, the actual throughput ρ_s can be obtained by

$$\rho_S = \frac{\lambda_S^*}{\mu C} = \frac{\lambda_S\left(1 - P_{blocking}\right)}{\mu C}, \quad (15)$$

where λ_s^* is also the actual arrival rate, i.e., the actual throughput is decreased by the blocking probability. It can be extended to calculate the performance of the queuing model for the integrated packet scheduler (Fig. 6). There are five queuing systems: Guaranteed Service (GS), Expedited Forwarding (EF), Assured Forwarding - level 1 (AF1), Assured Forwarding - level 2 (AF2) and Best Effort (BE). [1)]

$$\rho_S^{GS} = \frac{\lambda_S^{*GS}}{\mu C} = \frac{\lambda_S^{GS}\left(1 - P_{blocking}^{GS}\right)}{\mu C} = \frac{\lambda_S^{GS}\left(P_{delay}^{GS} P_{loss}^{GS} P_{resource}^{GS}\right)}{\mu C} \quad (16)$$

$$\rho_S^{C1} = \frac{\lambda_S^{*C1}}{\mu C} = \frac{\lambda_S^{C1}\left(1 - P_{blocking}^{EF} P_{blocking}^{DS}\right)}{\mu C} = \frac{\lambda_S^{EF}\left(P_{delay}^{EF} P_{loss}^{EF} P_{resource}^{DS}\right)}{\mu C} \quad (17)$$

$$\rho_S^{C2} = \frac{\lambda_S^{*C2}}{\mu C} = \frac{\lambda_S^{C2}\left(1 - P_{blocking}^{AF1} P_{blocking}^{DS}\right)}{\mu C}$$
$$= \frac{\left(\lambda_S^{EF}\left(1 - P_{delay}^{EF} P_{loss}^{EF}\right) + \lambda_S^{AF1}\right)\left(P_{delay}^{AF1} P_{loss}^{AF1} P_{resource}^{DS}\right)}{\mu C} \quad (18)$$

$$\rho_S^{C3} = \frac{\lambda_S^{*C3}}{\mu C} = \frac{\lambda_S^{C3}\left(1 - P_{blocking}^{AF2} P_{blocking}^{DS}\right)}{\mu C}$$
$$= \frac{\left(\left(\lambda_S^{EF}\left(1 - P_{delay}^{EF} P_{loss}^{EF}\right) + \lambda_S^{AF1}\right)\left(1 - P_{delay}^{AF1} P_{loss}^{AF1}\right) + \lambda_S^{AF2}\right)}{\mu C}$$
$$\times \frac{\left(P_{delay}^{AF2} P_{loss}^{AF2} P_{resource}^{DS}\right)}{\mu C} \quad (19)$$

$$\rho_S^{C4} = \frac{\lambda_S^{*C4}}{\mu C}$$
$$= \frac{\left\{\left(\left(\lambda_S^{C1} P_{delay}^{EF} P_{loss}^{EF}\right) + \left(\lambda_S^{C2} P_{delay}^{AF1} P_{loss}^{AF1}\right) + \left(\lambda_S^{C3} P_{delay}^{AF2} P_{loss}^{AF2}\right)\right)\right.}{\mu C}$$
$$\times \frac{\left.\left(1 - P_{resource}^{DS}\right) + \lambda_S^{C4}\right\}\left(P_{resource}^{BE}\right)}{\mu C} \quad (20)$$

1) The superscript denotes GS: Guaranteed service, EF: Expedited service, AF1: Assured Forwarding – level 1, AF2: Assured Forwarding – level 2, BE: Best Effort, DS: DiffServ, The superscript C1, C2, C3, and C4 mean the actual throughput of EF class. AF1 class, AF2 class, and BE class, respectively.

where the actual arrival rate of each class can be given as shown in Fig. 9.

$$\lambda_S^{C1} = \lambda_S^{EF} \quad (21)$$

$$\lambda_S^{C2} = \lambda_S^{EF} P_{reject}^{EF} + \lambda_S^{AF1} \quad (22)$$

$$\lambda_S^{C3} = \left(\lambda_S^{EF} P_{reject}^{EF} + \lambda_S^{AF1}\right) P_{reject}^{AF1} + \lambda_S^{AF2} \quad (23)$$

$$\lambda_S^{C4} = \left(\left(\lambda_S^{EF} P_{reject}^{EF} + \lambda_S^{AF1}\right) P_{reject}^{AF1} + \lambda_S^{AF2}\right) P_{reject}^{AF2} + \lambda_S^{BE}. \quad (24)$$

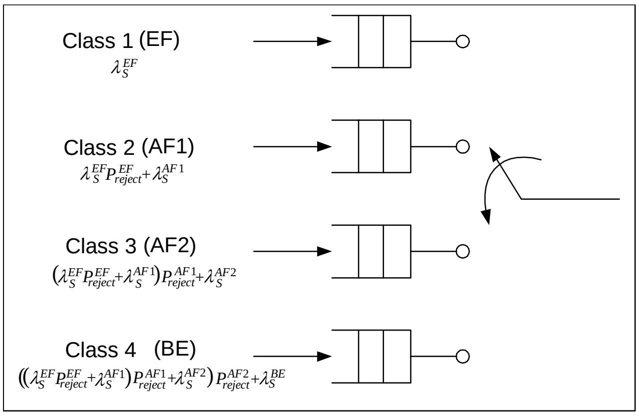


Fig. 9. Calculation of arrival rate of the proposed integrated packet scheduler.

4. End-to-End Performance Analysis

Now, we analyze the end-to-end performance of the throughput and transfer delay. In our analysis, the Markovian queuing network models are assumed for end-to-end data flows. All traffic flows are assumed to have behavior that is similar to Poisson arrival and exponential distribution.

First, we analyze the end-to-end throughput according to the flow blocking probability. We consider the fixed routing path γ between the source and destination node for data transfer. Now, the flow utilization ρ_γ for the routes γ is given by

$$\rho_\gamma = \frac{\lambda_\gamma^*}{\mu C} = \frac{\lambda_\gamma\left(1 - B\right)}{\mu C} \quad (25)$$

where λ_γ is the total number of packets per second that arrive at the original node on the route γ, λ_γ^* is the number of packets per second that are actually delivered to the destination node, B is the probability that the flow is blocked on route γ, $1/\mu$ is the average packet length, and C is the capacity in number of bits per second. Now, the flow blocking probability B is given by

$$B = 1 - \prod_{i \in \gamma} (1 - L_i) = 1 - \prod_{i=1}^{h} (1 - L_i), \tag{26}$$

where L_i is the probability that the *i*-th link is blocked on the route γ and h is the hop distance of route γ. The link blocking probability depends on the probabilities that the link is overflowed and out of service. Here, the link blocking probabilities L_i are recursively calculated by

$$\begin{aligned} L_i(\beta_i, \rho_i, C) &= \beta_i + (1 - \beta_i) P_{over}(\rho_i, C_U) \\ &\cong \beta_i + (1 - \beta_i) Er(\rho_i, C) \end{aligned} \tag{27}$$

since the *i*-th link utilization is given by $\rho_i = \dfrac{\lambda_i (1 - L_i)}{\mu C}$. β_i is the probability that link i is out of service (e.g., link or channel errors, routing failure, etc.), and $P_{over}(\,)$ is the probability that the given virtual circuits are overflowed. It could be approximately calculated by the Erlang loss formula denoted by $Er(\,)$ [12].

Now, we analyze the connection behavior of route γ on the saturated condition. Here, we assume that all the cascading circuits consisting of the route γ have a same bandwidth. As the flow traffic increases, the link utilization on route γ converges to overall end-to-end flow utilization. In a saturated condition, the *i*-th link utilization's ρ_i could be calculated by

$$\rho_i = \frac{\lambda_{i \in \gamma}(1 - L_i) + \lambda_{i \notin \gamma}}{\mu C} \;\rightarrow\; \rho_\gamma = (1 - L_i)\rho_\gamma + \rho_{i \notin \gamma} \quad \text{for } i = 1, \ldots, h \tag{28}$$

where $\rho_{i \notin \gamma} = \lambda_{i \notin \gamma} / \mu C$, $\lambda_{i \in \gamma}$ is the number of packets per second which arrive at the *i*-th link on the route γ, and $\lambda_{i \notin \gamma}$ is the number of packets per second which arrives at the *i*-th link except from the given route γ. From (28), it results in a saturated condition that

$$\rho_\gamma = \frac{\lambda_{i \notin \gamma}}{\mu C L_i} \qquad \text{for } i = 1, \ldots, h\,. \tag{29}$$

For the transfer delay analysis, we utilize the Markovian network model of a cascaded M/M/1 queuing circuit. The average flow transfer delay could be calculated by

$$\overline{T_T} = \frac{1}{\lambda_\gamma (1 - B)} \sum_{i \in \gamma} \lambda_i^* \overline{T_i} = \frac{1}{\lambda_\gamma (1 - B)} \sum_{i=1}^{h} \lambda_i (1 - L_i) \overline{T_i}\,, \tag{30}$$

where $\overline{T_i} = 1 / (\mu_i C_i - \lambda_i (1 - L_i))$. The overall average end-to-end delay can be calculated by summation of the average system delay and average flow transfer delay, that is

$$\overline{T_{overall}} = \overline{T_T} + \overline{T_C}\,. \tag{31}$$

Now, we calculate the number of lost messages on the route γ since they may be caused by link blocking or out of resources. It could be measured by the average number of messages that are not correctly delivered to the destination, which is denoted by $\overline{N_{loss}}$, and given by

$$\overline{N_{loss}} = \lambda_\gamma \left(\overline{T_T} B + 2\overline{T_T} (1 - B) B_D \right) \tag{32}$$

where B_D is the probability that a destination user is blocked since there are no available resources.

IV. NUMERICAL RESULTS

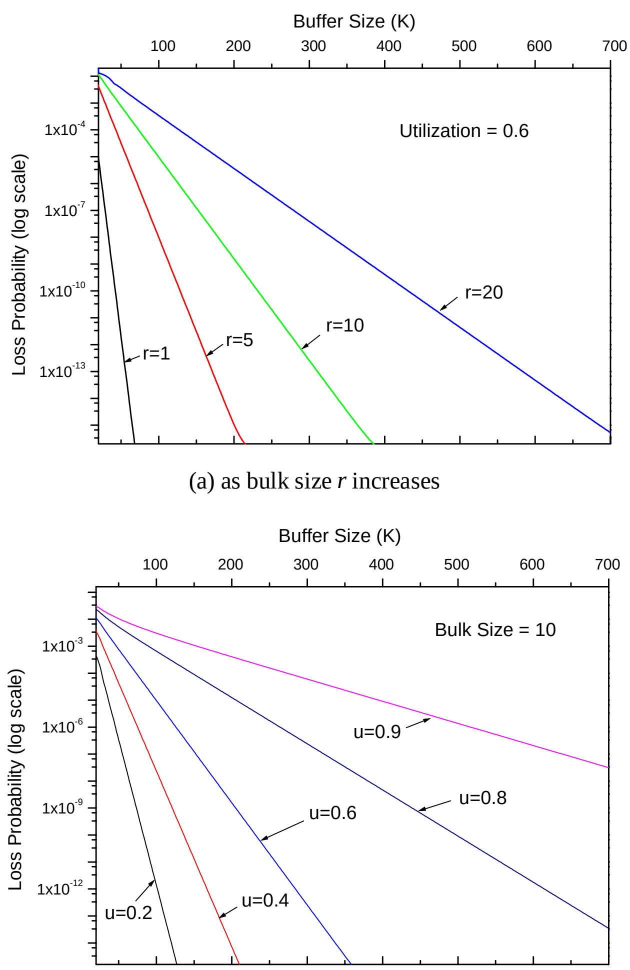
We present numerical results for our proposed system including the integrated packet scheduler. To get the numerical results, we consider the following assumptions. For the five queues model of the GS queue and priority queues (EF queue, AF1 queue, AF2 queue, BE queue), each queue has the same buffer size and the traffic activities of each class are statistically the same. We assume that the link bandwidth is 155Mbps and the mean service time is 2.73 μs. The basic unit of the packet size is 53octets for the bulk arrival model.

We first show the loss probability in M/M/1/K with the bulk arrival system. Next, the average time delay and the average waiting time in the priority queuing system is represented. We also present the blocking probability and actual load considering our flow-based admission control algorithm and end-to-end performance result.

1. Loss Probability

The loss probability strongly depends on the buffer size *K*. The loss behaviors due to buffer overflow can occur in the GS queue and BE queue. Figure 10 shows the behavior of loss probability versus buffer size. Loss probability decreases as the buffer size *K* increases. Figure 10(a) shows that the loss probabilities are sensitive to bulk size. We observe that the loss probabilities are about 10^{-16} for r =5, 10^{-9} for r =10, and 10^{-5} for *r*=20 in the same buffer size (*K*=200).

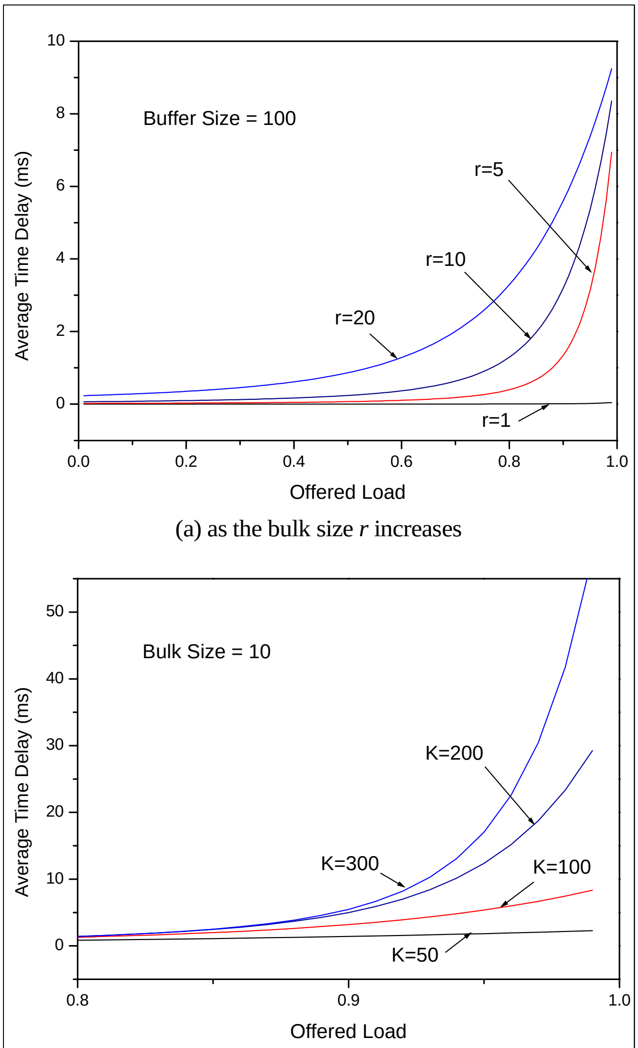
In Fig. 10(b), we observe the behavior of loss probability as the offered utilization increases. This figure shows that the loss probability is sensitive to the utilization *u*. We observe that the loss probabilities are about 10^{-3} for *u*=0.9, 10^{-5} for *u*=0.8, 10^{-9} for *u*=0.6, and 10^{-15} for *u*=0.4 in the same buffer size (*K*=200).



(a) as bulk size *r* increases

(b) as the offered utilization *u* increases

Fig. 10. Behaviors of loss probability versus buffer size.



(a) as the bulk size *r* increases

(b) as the buffer size *K* increases

Fig. 11. The behaviors of average time delay versus offered load.

2. Average Time Delay

Next, we calculate the average time delay in M/M/1/K with the bulk arrival system. Figure 11 shows the behavior of the average transfer delay versus the offered load. This performance behavior is applied to all traffic classes. The overall delay behavior is very similar to the conventional queuing behavior. Figure 11(a) shows the average time delay versus the offered load as the bulk size increases. The average time delay sharply increases for a high loading condition as the bulk size increases. This indicates that the bulk arrival has a great influence on system performance. Figure 11(b) gives the average time delay depending on the offered load for the buffer size difference when the bulk size is fixed. This figure demonstrates that buffer size change has only a little influence on the average time delay. The average time delay scarcely increases until the offered load reaches 0.8.

3. Priority Queuing System Performance

In the numerical results, the following service classes are defined as EF, AF1, AF2, and BE. As shown in Fig. 9, each service class queuing model is represented by the relationship of service classes and blocking probabilities according to our flow-based admission control algorithm. We assume that the arrival rates are the same, that is, $\lambda_S^{C1} = \lambda_S^{C2} = \lambda_S^{C3} = \lambda_S^{C4}$. Figure 12 shows the behavior of the average waiting time versus the offered load for the different priority classes. As we can see, class 1 traffic results in the shortest average waiting time.

4. Normalized End-to-End Throughput and Transfer Delay

Figure 13 depicts the behavior of normalized throughput versus the link blocking probability. The capacity of data flows are

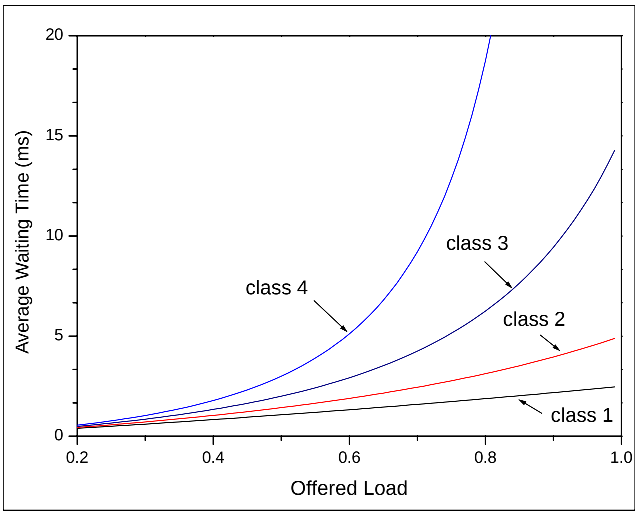


Fig. 12. Average waiting time versus offered load for the priority queueing model.

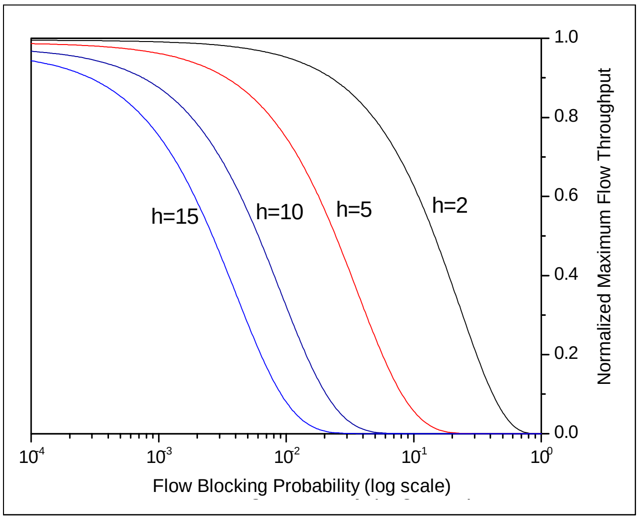


Fig. 14. Normalized maximum flow throughput versus flow blocking probability (h = hop count).

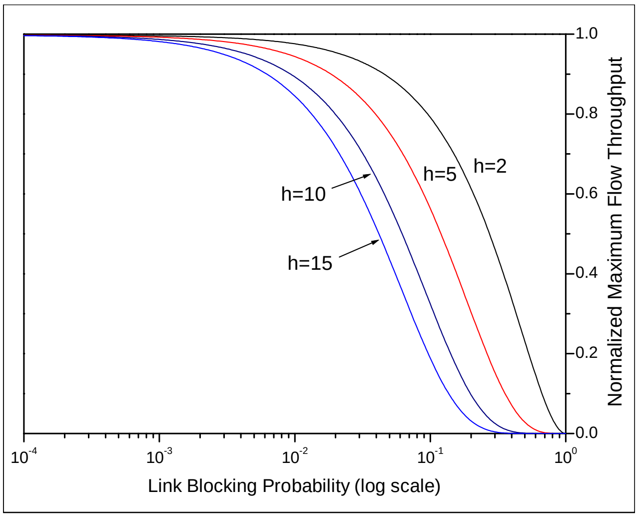


Fig. 13. Normalized maximum flow throughput versus link blocking probability (h = hop count).

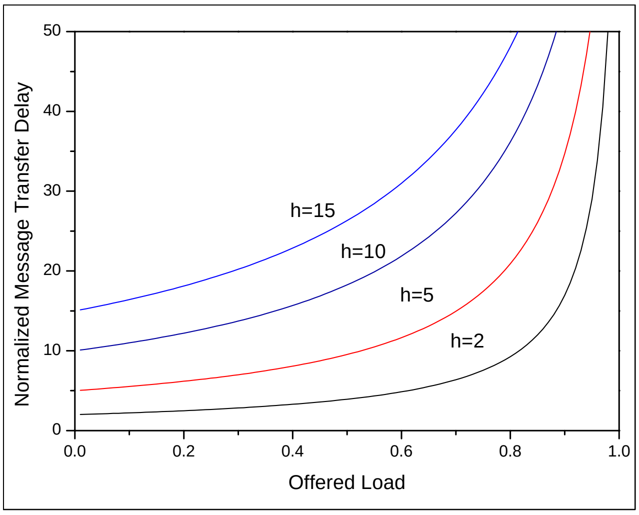


Fig. 15. The behaviors of normalized message transfer delay versus offered load.

significantly reduced while both the hop distance and the link blocking probability increase. Figure 14 shows the behavior of normalized throughput versus the flow blocking probability. It notes that the flow blocking probability can be given in the flow admission control algorithm shown in Fig. 7. The normalized throughput of data flows is reduced according to both the hop distance and the flow blocking probability, but the flow blocking probability affects the throughput behavior more. Figure 15 shows the behavior of normalized mean data transfer delay versus the offered load when the flow blocking probability is 10^{-3}. This figure also shows the behavior of the data channel as the hop distance increases. The normalized mean data transfer delay has an influence on the hop count and offered load. It shows that message transfer delay is reduced for the short hop distance.

V. CONCLUSIONS

In this paper, we proposed a flow-based admission control algorithm to integrate DiffServ and IntServ models on ATM based MPLS networks. We demonstrated that the proposed algorithm is one of the best solutions [13], [14] for IntServ/DiffServ provisioning as well as guaranteed QoS provisioning. The proposed network supports the virtual shortcut connection or MPLS label swapping using an underlying ATM capability. This has a number of advantages on throughput, end-to-end delay, and flow utilization.

The proposed DiffServ mapping and integrated packet scheduler are the essential part of the flow-based admission control algorithm. Accordingly, we also analyzed the behavior of normalized throughput versus flow blocking probability. The proposed approaches guarantee a hard QoS using an ATM shortcut connection for the IntServ guaranteed service class and increase resource utilization according to the per class QoS condition. We analyzed loss probability and average time delay with the proposed priority queuing model. The results show that loss probability is closely related to buffer size and bulk size. The mean transfer delay is related to priority and hop count. As the flow blocking probability increases, the behavior of the normalized throughput is obtained. However, we did not analyze the detailed blocking probability according to traffic characteristics. Therefore, it is necessary to conduct further performance analysis.

The numerical results show good performance behavior on normalized throughput and delay performance. In the proposed priority queuing system of the IntServ/DiffServ model, the average delay of class 4 traffic increases by about three times that of class 1 traffic. When the offered load is less than 0.6, the blocking probability of guaranteed service is less than 10%. As the link blocking probabilities increase about 10%, the flow blocking probabilities for data flows increase about 10^{-2} for hop count h = 2 and about 10^{-1} for hop count h = 5.

REFERENCES

[1] G. Eichler, H. Hussmann, G. Mamals, C. Prehofer, and S. Salsano, "Implementing Integrated and Differentiated Services for the Internet with ATM Networks: a Practical Approach," *IEEE Comm. Magazine*, vol. 38, Jan. 2000, pp. 132-141.

[2] Paul P. White, "RSVP and Integrated Services in the Internet: a Tutorial," *IEEE Comm. Magazine*, vol. 35, May 1997, pp. 100-106.

[3] 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, IETF, Dec. 1998.

[4] P. Giacomazzi and L. Musumeci, "Transport of IP Controlled-Load Service over ATM Networks," *IEEE Network*, vol. 13, Jan-Feb. 1999, pp. 36-47.

[5] Arun Wiswanthan, Nancy Feldman, Zheng Wang, and Ross Callon, "Evolution of Multi-Protocol Label Switching," *IEEE Comm. Magazine*, vol. 36, May 1998, pp. 165-173.

[6] Anoop Ghanwani, Bilel Jamoussi, Don Fedyk, Peter Ashwood-Smith, Li Li, and Nancy Feldman, "Traffic Engineering Standards in IP Networks Using MPLS," *IEEE Comm. Magazine*, vol. 37, Dec. 1999, pp. 49-53.

[7] B. Davie, J. Lawrence, K. McCloghrie, Y. Resen, G. Swallow, Y. Rekhter, and P. Doolan, "MPLS Using LDP and ATM VC Switching," RFC3035, IETF, Jan. 2001.

[8] J.K. Choi, S.H. Kim, N.Kim, S.W.Shon, and M.K. Choi, "Performance Benefits of Virtual Path Tunneling for Control and Management Flows in the Broadband ATM Networks," *ETRI J.*, vol. 21, no. 4, Dec. 1999, pp. 23-28.

[9] D.D. Clark, S. Shenker, and L. Zhang, "Supporting Real-Time Applications in an Integrated Service Packet Network: Architecture and Mechanism," *Proc. ACM SIGCOMM'92*, Aug. 1992, pp. 14-26.

[10] Sugih Jamin, Scott J.Shenker, and Peter B.Danzig, "Comparison of Measurement-Based Admission Control Algorithms for Controlled-Load Service," *IEEE INFOCOM'97*, vol. 3, Apr. 1997, pp. 973-980.

[11] Sugih Jamin, Peter B. Danzig, Scott J. Shenker, and Lixia Zhang, "A Measurement-Based Admission Control Algorithm for Integrated Service Packet Networks," *IEEE/ACM Transactions on Networking*, vol 5, no 1, Feb. 1997, pp. 56-70.

[12] L. Kleinrock, *Queuing Systems Volume 1: Computer Applications*, John Wiley & Sons, 1974.

[13] Ilias Andrikopoulos, George Pavlou, "Supporting Differentiated Services in MPLS Networks," *IEEE*, *IWQoS'99*, 1999, pp. 207-215.

[14] T.V. Do, G. Eichler, H. Hussmann, B. Mamais, C. Prehofer, S. Salsano, J.J. Tchouto, C. Tittel, P. Todorova, and I.S. Venieris, "ELISA: European Linkage between Internet Integrated and Differentiated Services over ATM," *IEEE Workshop, QoS Support for Real Time Internet Apps*, June 2-4, 1999.

Gyu Myoung Lee was born in Chinju, Korea, in 1972. He received his BS degree in electronic and electrical engineering from Hong Ik University, Seoul, Korea, in 1999 and his MS degree in the School of Engineering from ICU (Information and Communications University), Daejeon, Korea in 2000. He is currently a PhD student in the network group, ICU. His research interests include signaling and control architecture for IP-based optical networks and QoS-based traffic control technology.

Jun Kyun Choi received his BS degree in electronics from Seoul National University in 1982 and MS and PhD degrees from KAIST in 1985 and 1988. He worked for ETRI during 1986-1997. He is currently working as an Associate Professor in Information and Communications University, Daejeon, Korea. His research interests include high-speed network architecture and protocol.

Mun Kee Choi received his BS degree in applied mathematics from Seoul National University in 1974 and MS degree in industrial engineering from KAIST in 1978. He obtained his PhD degree in operations research from North Carolina State University, in 1989. He worked for ETRI System Technology Division from 1978 to 1999. He is currently working as a Professor in the School of Management, ICU. His current research interests include high speed network architectures, business on networks and GRID technology.

Man Seop Lee received his BS and MS degrees from Busan National University in 1976 and 1978 and his PhD degree from KAIST in 1991, all in electrical engineering. During the period of 1979 to 1998, he worked in the area of optical transmission technology and served as Director of Transmission Technology Department in ETRI. He has been an Associate Professor in Information and Communications University (ICU) since 1998 and now serves as Director of Venture Business Incubation Center of ICU. His research interests include 40Gb/s optical transmission technology, PON technology in access area and optical wireless LAN technology. He is also Chairman of the Subcommittee of Transmission Technology Group in KICS.

Sang-Gug Jong was born in Seoul, Korea, 1956. He received his BS and PhD degrees in electronics engineering from Kyung Hee University, Seoul, Korea, in 1980 and 1994 and a DESS degree in Electronics Engineering from Paris 6 University, Paris, France, in 1985. From 1987 to 2000 he worked as the Director of the Training Center of Korea Telecom. Since 2000 he has been with the Telecommunication Network Laboratory in KT as Director. His current research interests include high speed internet access, traffic engineering, and Mobile communications.