On Implementation Architecture for Achieving QoS Provisioning in Integrated Services Networks

Dapeng Wu* Yiwei Thomas Hou† Zhi-Li Zhang‡ H. Jonathan Chao§
Takeo Hamada¶ Tomohiko Taniguchi‖

Abstract

This paper presents an implementation architecture based on per flow queueing that is capable of achieving QoS provisioning for future integrated services networks consisting of the guaranteed service (GS), the controlled-load (CL), and the best-effort (BE) service classes. We propose several novel traffic management mechanisms, including Adaptive Rate allocation for Controlled-load (ARC), a hybrid model-based and measurement-based admission control algorithm for GS and CL flows, and a Quasi-Pushout Plus (QPO+) packet discarding mechanism. Simulation results show that our architecture and algorithms provide hard QoS guarantee to GS flows under all conditions, consistent (soft) QoS to CL flows under both light and heavy load conditions, and effective control of negative impact from non-conforming CL flows. Our architecture and algorithms also resolve several issues associated with the traditional class-based approach.

1 Introduction

One of the most challenging problems for the next generation Internet is to support diverse multimedia applications with quality of service (QoS) guarantees. To address this challenge, the Internet Engineering Task Force (IETF) Integrated Services Working Group has specified three service classes, namely, the *guaranteed service* (GS) [9], the *controlled-load service* (CL) [10], and the *best-effort service*. The GS guarantees that packets will arrive within the guaranteed delivery time, provided that the flow's traffic conforms to its specified traffic parameters [9]. That is, GS does not control the minimal or average delay of a packet; it merely controls the maximal queueing delay. The CL service is intended to support a broad class of applications which have been developed for use in today's Internet, but are sensitive to heavy load conditions [10]. The CL service does not specify any target QoS parameters. Instead, acceptance of a request for CL is defined to imply a commitment by the network to provide the requester with a service closely approximating the QoS the same flow would receive under lightly loaded conditions. The best-effort (BE) service class offers the same type of service under the current Internet architecture. That is, the network makes effort to deliver data packets but makes no guarantees.

To support the diverse QoS requirements from the GS, the CL, and the BE services in integrated services networks, new network architecture and traffic management algorithms must be in place. Such architecture and algorithms should meet the following performance evaluation criteria as specified by IETF.

Criterion 1 (C1): For GS, IETF requires that the architecture and algorithms of each switch must ensure that the delay bounds are never violated and packets are not lost if a source's traffic conforms to its traffic descriptors [9].

Criterion 2 (C2): For CL service, an architecture and algorithms should provide a flow, under all load conditions, with a QoS closely similar to the QoS that the same flow would receive under lightly loaded network conditions [10].

Criterion 3 (C3): The network architecture and traffic managements algorithms must be capable of controlling non-conforming GS/CL flows by minimizing their negative impact on other conforming GS/CL flows and BE flows [9, 10].

Previous work on integrated services networks has been focused on class-based queueing architecture [2, 5]. However, when class-based approach is used to support CL service, there are several problems as follows. First of all, it is not clear, under class-based approach, how to effectively isolate non-conforming flows and minimize their negative impact on other conforming GS and CL flows (i.e. criterion C3). Secondly, the class-based approach requires to classify all incoming CL flows, each of which may have different traffic behavior and QoS requirements, into a limited set of classes. Therefore, it is impossible to provide a flexible QoS support for each individual CL flow based on its unique traffic behavior and specific QoS requirements. Finally, it is impossible for a class-based approach to enforce fair rate allocation for CL flows.

Recent market demand has put QoS support as the key feature in differentiating network products from various vendors. Furthermore, due to advances in silicon technology, hardware implementation of sophisticated per flow based traffic management algorithms no longer poses any major cost constraint [1]. Such market demand and hardware capabilities enable us to design per

*D. Wu is with Polytechnic University, Brooklyn, NY, USA.
†Y. T. Hou is with Fujitsu Laboratories of America, Sunnyvale, CA, USA.
‡Z.-L. Zhang is on the faculty of the Dept. of Computer Science, University of Minnesota, Minneapolis, MN, USA.
§H. J. Chao is on the faculty of the Dept. of Electrical Engineering, Polytechnic University, Brooklyn, NY, USA.
¶T. Hamada is with Fujitsu Laboratories of America, Sunnyvale, CA, USA.
‖T. Taniguchi is with Fujitsu Laboratories of America, Sunnyvale, CA, USA.

0-7803-5284-X/99/$10.00 © 1999 IEEE.

flow based traffic management mechanisms to control QoS with substantially improved performance than traditional class-based approach for the next generation switches/routers. This paper presents a novel architecture and several traffic management algorithms based on per flow queueing that not only satisfy the three criteria to support integrated traffic of the GS, the CL, and the BE services, but also resolve the several problems associated with the traditional class-based approach.

Our network architecture strives to offer a good balance between traffic isolation and buffer sharing. We make three separate buffer partitions for the GS, the CL, and the BE flows, respectively, and one separate partition for non-conforming GS/CL packets. Per flow queueing with weighted fair queueing (WFQ) scheduling is employed for GS and CL flows, while shared queueing with FIFO is employed for BE flows and non-conforming GS/CL packets. We propose an Adaptive Rate allocation for Controlled-load (ARC) algorithm to provide soft bandwidth allocation to CL flows while enforcing a guaranteed rate allocation to each GS flow. We present a hybrid call admission control (CAC) algorithm consisting of model-based CAC for GS flows and measurement-based CAC for CL flows. Finally, we design a packet discarding algorithm, called quasi-pushout plus (QPO+), to effectively control non-conforming CL flows. Our simulation results show that our architecture offers guaranteed QoS to GS flows under all conditions (C1), consistent (soft) QoS to CL traffic under both light load and heavy load conditions (C2), and minimal negative impact on conforming flows should there be any non-conforming behavior from CL flows (C3). Furthermore, our architecture and traffic management algorithms have resolved the several problems associated with the traditional class-based approach.

The remainder of this paper is organized as follows. Section 2 presents our network node architecture. In Section 3, we presents our traffic management algorithms. Section 4 uses simulation results to demonstrate the performance of our network architecture and traffic management algorithms. Section 5 concludes this paper.

2 Network Node Architecture

We assume that each switch employs output port buffering. Figure 1 shows our architecture for the GS, the CL, and the BE traffic at each output port of a network node. Under our architecture (Fig. 1), we partition each output port buffer pool into four parts: one for GS flows, one for CL flows, one for BE traffic, and one for non-conforming GS or CL packets.

Within the same buffer partition for GS or CL flows, we employ per flow queueing for each individual GS or CL flow. Furthermore, a GS (or a CL) flow can share buffering with other GS (or CL) flows within their own buffer partition while there is no buffer sharing across partitions. That is, there is no buffer sharing between GS and CL flows. We believe this approach offers an excellent balance between traffic isolation and buffer sharing.

For BE buffer partition, we employ a common FIFO shared queue. This is because there is no QoS commitment of any kind to each individual BE flow.

For admitted GS or CL flows equipped with policing mechanism, packets not conforming to traffic parameters

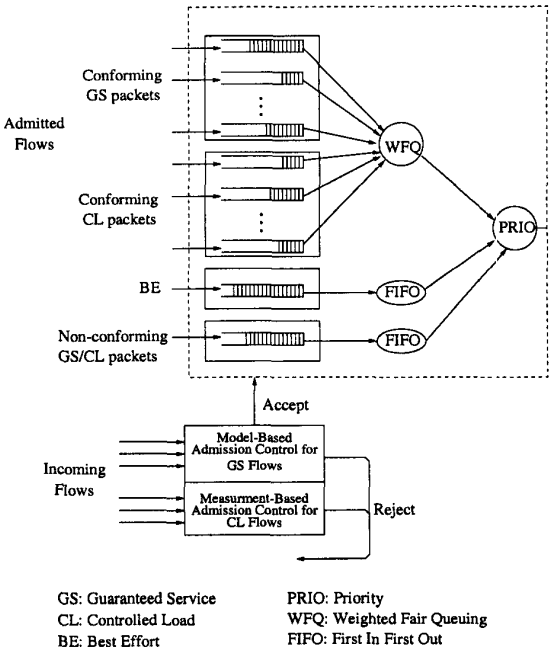

Figure 1: A network node architecture.

will be tagged at the network access point [10]. We propose to use one separate buffer for such non-conforming GS or CL packets and give them the lowest service priority so that they will have minimal negative impact on BE traffic [9, 10].

Under the above buffering architecture, we design our per flow based traffic management algorithms with the aim of achieving the three criteria for the GS, the CL, and the BE services and solve the several problems associated with the class-based approach.

3 Traffic Management Algorithms

We organize this section as follows. Section 3.1 presents rate and buffer allocation schemes for GS and CL flows. In Section 3.2, we show our hybrid CAC algorithm. Section 3.3 presents packet discarding mechanisms.

3.1 Resource Allocation for GS/CL Flows

For GS flows, we employ a simple calculation to allocate bandwidth and buffer and provide a deterministic QoS guarantee (i.e. hard delay bound for each packet and zero packet loss rate) for each flow. On the other hand, for CL flows, we can choose a much less conservative approach, since it only requires soft QoS guarantees. We show how to estimate the effective bandwidth of a CL flow by measuring the entropy of such flow. To support the link sharing between the GS and the CL flows, we present a novel rate assignment strategy called ARC (short for Adaptive Rate allocation for Controlled-load) to provide hard bandwidth guarantee to GS flows under all conditions and consistent (or soft) bandwidth allocation to CL flows. Also shown in Fig. 1 is a hierarchical packet scheduling architecture where a priority link scheduler is shared among a weighted fair queue-

ing (WFQ) for GS and CL flows, a FIFO for BE flows, and a FIFO for non-conforming GS/CL packets. Service priority is first given to the WFQ scheduler, and then to BE FIFO scheduler. The FIFO scheduler for non-conforming GS/CL packets has the lowest priority in receiving service.

The reason why we use per flow queueing and WFQ scheduler for CL flows is based on the results in [8], where it has been shown that GPS (fluid model of WFQ) scheduling is able to provide a flexible QoS support (both loss and delay requirement) and enforce bandwidth allocation for each individual flow. In order words, per flow queueing with a WFQ scheduler in our architecture solves the last two problems associated with the traditional class-based approach.

Model-Based Rate Calculation for GS Flows

According to [9], the end-to-end queueing delay bound for a GS flow j is given as follows.

$$D_j \leq \begin{cases} \frac{\sigma_j - M_j}{p_j - \rho_j} \cdot (\frac{p_j}{R_j} - 1) + \frac{M_j + Ctot_j}{R_j} + Dtot_j & \\ \qquad\qquad \text{if } \rho_j \leq R_j < p_j; & \\ \frac{M_j + Ctot_j}{R_j} + Dtot_j \qquad \text{if } \rho_j \leq p_j \leq R_j. & \end{cases} \tag{1}$$

where
σ_j: the leaky bucket size for flow j;
ρ_j: the token generating rate for flow j;
p_j: the peak rate of flow j;
R_j: the allocated bandwidth for flow j;
M_j: the maximum datagram size of flow j;
$Ctot_j$: the rate-dependent error term for flow j;
$Dtot_j$: the rate-independent error term for flow j.

Therefore, for a given delay requirement for a GS flow j, its required rate R_j^{GS} can be obtained from Eq. (1).

Buffer Allocation for GS Flows

To guarantee zero packet loss for GS, appropriate buffer must be allocated for each GS flow. We use the result in [4] to allocate buffer for GS flows. For flow $j \in GS$, the required buffer allocation at the l^{th} switch along the path is given by

$$b_j^{(l)} = M_j + \frac{(p_j - X)(\sigma_j - M_j)}{(p_j - \rho_j)} + \sum_{k=1}^{l} [\frac{C_j^{(k)}}{R_j} + D_j^{(k)}] \cdot X$$

where

$$X = \begin{cases} \rho_j & \text{if } \frac{\sigma_j - M_j}{p_j - \rho_j} \leq \sum_{k=1}^{l} [\frac{C_j^{(k)}}{R_j} + D_j^{(k)}]; \\ R_j & \text{if } (\frac{\sigma_j - M_j}{p_j - \rho_j} > \sum_{k=1}^{l} [\frac{C_j^{(k)}}{R_j} + D_j^{(k)}]) \text{ and } \\ & \quad (p_j > R_j); \\ p_j & \text{otherwise.} \end{cases}$$

In the above equations, $C_j^{(k)}$ and $D_j^{(k)}$ are the rate-dependent error term and the rate-independent error term at the k^{th} switch for flow $j \in GS$, respectively. $Ctot_j$ and $Dtot_j$ are the sum of $C_j^{(k)}$ and the sum of $D_j^{(k)}$ along the path of flow $j \in GS$, respectively.

Measurement-Based Rate Estimation for CL

Unlike GS flows, CL flows do not have hard delay requirements and therefore do not require hard bandwidth and buffer guarantee. Instead, CL flows only require soft bandwidth support from the network for consistent performance under light and heavy load conditions. Therefore, we can adapt more efficient bandwidth allocation based on the measurement of a CL flow's actual traffic behavior (instead of relying on a model with rigid parameters).

To measure the the effective bandwidth for CL flows, we divide time axis into small fixed interval d and denote t_B be the time required to accumulate a total of B bits for a particular CL flow. Clearly, t_B is a variable depending on the particular incoming CL flow traffic behavior. We also introduce a threshold T_{max} to set up an upper bound on the measurement interval and take the minimum of t_B and T_{max} as our measurement window T. That is, $T = \min\{t_B, T_{max}\}$.

Denote M the total number of d's within a measurement window T, i.e. $M = \lceil \frac{T}{d} \rceil$. Let $A_i^T(k)$, $1 \leq k \leq M$ be the number of bits arrived in the kth measurement interval. We first estimate the scaled cumulant generating function (SCGF) $\Lambda(\delta)$.

$$\Lambda^T(\delta_i) = \frac{1}{T} \log \frac{1}{M} \sum_{k=1}^{M} e^{\delta_i A_i^T(k)}$$

where $\delta_i = \frac{-(\log \varepsilon_i - \log \gamma_i)}{b}$, b is the size of the CL buffer partition, ε_i is the packet loss rate requested by source i, and γ_i is the probability that flow i is non-empty. Let λ_p^i be the peak rate of flow i. Then, we can obtain the effective bandwidth of CL flow i by

$$\alpha(\delta_i) = \min\{\lambda_p^i, \frac{\Lambda^T(\delta_i)}{\delta_i}\} .$$

In our measurement, we only measure the number of packets in bits that have successfully entered the buffer partition, *excluding* discarded packets. This is because discarded packets will not be served by the scheduler, and thus it is only necessary to consider the packets that have successfully entered the buffer and allocate appropriate rate for their service. Furthermore, we find that such measurement technique has the additional advantage of preventing non-conforming flows from unfairly increasing its rate share in the scheduler by sending more packets.

We assume the requirement for packet loss rate ε_i is available. For CL flows, user are not required to explicitly request such QoS parameter. We can start with a small value for ε_i, which represents a conservative admission control, and then increase ε_i if experience indicates that a less conservative admission control would be adequate [3].

Rate Assignment for GS and CL Flows

To provide hard rate guarantee to each GS flow and soft rate guarantee to each CL flow, we employ the following weight assignment strategy in the WFQ scheduler. When the sum of guaranteed rates from GS flows (calculated from Eq. (1)) and the estimated rates from

CL flows is less than the link capacity, we use these rates directly in the WFQ for the corresponding GS or CL flows and the delay requirement for each GS flow is always guaranteed. On the other hand, if the sum of calculated GS rates and measured CL rates is greater than the link capacity, we shall still use the calculated rate for each GS flow as the weight for such flow in the WFQ scheduler but use a down-scaled version of the estimated rate for a CL flow (by a factor of remaining capacity divided by the sum of estimated CL rates) as the weight for the corresponding CL flow in the WFQ scheduler. We name this rate assignment ARC, for Adaptive Rate assignment for Controlled-load.

Algorithm 1 ARC

For an admitted CL flow i, its rate R_i^{CL} is given by

$$R_i^{CL} = \begin{cases} \alpha(\delta_i) & \text{if } \sum_{i\in CL}\alpha(\delta_i) + \sum_{j\in GS} R_j^{GS} \le r; \\ \alpha(\delta_i)\cdot \frac{(r-\sum_{j\in GS} R_j^{GS})}{\sum_{i\in CL}\alpha(\delta_i)} & \text{if } \sum_{i\in CL}\alpha(\delta_i) + \sum_{j\in GS} R_j^{GS} > r. \end{cases}$$

where $\alpha(\delta_i)$ is the measured effective bandwidth of the CL flow i, and r is the link rate. □

Once we use the R_j^{GS}, $j \in GS$, and R_i^{CL}, $i \in CL$ as the weight for the respective flow j or i in our WFQ scheduler, we have the following property on rate allocation for GS and CL flows. The rate allocation for each GS flow is no less than its calculated guaranteed rate while the rate allocation for each CL flow may have occasional fluctuations (due to on-line measurement of each CL flow traffic behavior), which is understood to be a soft bandwidth guarantee [10].

3.2 Admission Control Algorithm

The objective of admission control is to guarantee QoS requirements of existing flows (i.e. hard QoS guarantees to admitted GS flows and consistent (or soft) performance to admitted CL flows) while maximizing network utilization. We design a simple hybrid admission control algorithm which consists of model-based admission control for GS flows and measurement-based admission control for CL flows. There is no admission control for BE traffic and such type of flows are always admitted.

The following is our admission control algorithm for the GS and CL flows, where μ is target utilization and r is the link capacity.

Algorithm 2 Admission Control for GS and CL

Upon the receipt of a new flow request for GS
  if $(\sum_{j\in GS} R_j^{GS} + \sum_{i\in CL} R_i^{CL} + R_{new}^{GS} \le \mu \cdot r)$ and $(\sum_{j\in GS} b_j^{GS} + b_{new}^{GS} \le b^{GS})$
  /* b^{GS} is the size of GS buffer partition. */
    admit the new GS flow and stop;
  else
    reject the new GS flow and stop;

Upon the receipt of a new flow request for CL service
  if $(\sum_{j\in GS} R_j^{GS} + \sum_{i\in CL} R_i^{CL} + R_{new}^{CL} \le \mu \cdot r)$
  /* R_{new}^{CL} is the requested rate of the new CL flow. */
    admit the new CL flow and stop;
  else
    reject the new CL flow and stop. □

In Algorithm 2, we use the peak rate of a CL for admission control rather than the token generating rate ρ. This is because that our previous experience in [11] has shown that the ρ parameter can be less than the required rate and, therefore, the targeted QoS could be violated if we only reserve a bandwidth of ρ.

3.3 Packet Discarding Mechanisms

An arriving packet is allowed to enter the particular buffer partition only when there is enough remaining buffer space. Otherwise, we have to either discard the incoming packet or discard some other packet(s) in the buffer in order to make room for the incoming packet.

For GS buffer partition, since the admission control algorithm for an incoming GS flow includes buffer allocation, an admitted flow will have sufficient buffer space throughout its path. Therefore, there should not be any buffer overflow for GS buffer partition. In the worst case, should the network misbehave, we may employ simple tail-dropping for GS buffer partition.

For BE buffer partition, we use Flow Random Early Drop (FRED) (proposed in [6] to prevent non-adaptive BE flows from harming other TCP-like BE flows) for BE traffic.

For the non-conforming GS/CL packets buffer partition, we employ simple tail-dropping.

For CL flows, buffer partition may overflow since we do not reserve any buffer space for each CL flow and the traffic behavior of such flow is unpredictable. Furthermore, since the network cannot assume that every admitted CL flow is equipped with a policing mechanism at the network access point, some non-conforming CL flow without policing mechanism may keep sending non-conforming packets into the CL buffer partition instead of the non-conforming GS/CL packets buffering partition. To address this problem, we propose a powerful pushout mechanism, called *quasi-pushout plus* (QPO+) to pushout packets from the longest queue to non-conforming buffer whenever the corresponding buffer partition cannot accommodate a new packet. Such packet discarding scheme is only possible under per flow queueing architecture and can achieve fair buffer sharing among competing flows during congestion. Our QPO+ extends the quasi-pushout (QPO) mechanism by its ability to handle variable sized packets [7]. We show that our QPO+ mechanism is capable to protect the QoS guarantees to conforming flows by isolating and discarding packets from non-conforming flows. Note that our QPO+ also solves the first problem associated with the traditional class-based approach.

In our QPO+ mechanism, a register is used to estimate the longest queue (LQ) in the CL buffer partition and is only updated upon a packet's arrival or departure. The queue length of flow i, $QL[i]$, is measured in unit of bits. When a packet arrives and the remaining free buffer space cannot accommodate such packet, packets from the quasi-longest queue (LQ) will be transferred

to the non-conforming buffer partition (instead of being discarded) and make room for this incoming packet. The following algorithm shows how our QPO+ packet discarding scheme works. We use RB to denote the remaining free buffer space in the CL buffer partition.

Algorithm 3 QPO+ Mechanism for CL

When a packet of size P from flow i arrives at the output port of a switch,

```
if (RB ≥ P) {
    accept such packet and let it join flow i;
    QL[i] := QL[i] + P;
    RB := RB − P;
    }
else /* i.e. RB < P */ {
    if (LQ == i) or ((QL[LQ] + RB) < P)
        put this incoming packet into the
        non-conforming buffer partition;
    else {
        pop packets (with a sum of x bits) from the
        tail of queue LQ to the non-conforming
        buffer until (RB + x > P);[1]
        QL[LQ] := QL[LQ] − x;   RB := RB + x;
        accept the incoming packet and let it join
        flow i;
        QL[i] := QL[i] + P;
        RB := RB − P;
        }
    }
if (QL[LQ] < QL[i])
    LQ := i;   /* Input comparison */
```

When a packet of size P from flow j departs from the output port of a switch,

```
QL[j] := QL[j] − P;
RB := RB + P;
if (QL[LQ] < QL[j])
    LQ := j   /* Output comparison */
```

□

We would like to emphasize the following two points regarding QPO+ packet discarding mechanism. First of all, it should be clear that only under per flow queueing architecture can we employ such pushout packet discarding mechanism. Secondly, according to [10], network elements must not assume that data sources or upstream elements have taken action to "police" CL flows (i.e. limiting their traffic to conform to the flow's traffic descriptor). Therefore, each network element providing CL service must independently ensure that criterion C3 is met in the presence of non-conforming GS/CL traffic. Our simulations have shown that FIFO with tail dropping cannot prevent non-conforming traffic from affecting conforming flows. Only packet discarding mechanism on a per flow basis such as QPO+ can effectively control non-conforming flows when policing is not done. It has been shown in [6] that FIFO-based RED cannot effectively control non-conforming flows. In the simulation results, we shall further demonstrate that when non-conforming users are present in the network, only QPO+ can minimize its negative impact on other conforming flow while other packet discarding schemes (e.g. drop-tail) are unable to effectively control such non-conforming flows.

We stress that it is entirely feasible to implement our QPO+ mechanism in hardware for IP switch/router. Since the largest IP packet size is 1500 bytes and the smallest is 64 bytes (under Ethernet), in the worst-case, the incoming packet with the largest packet size will pushout at most 24 packets with the smallest packet size. Unlike ATM where there is a cycle time constraint (e.g. 2.83 μs for OC-3), there is no such cycle time for IP switch/router and the processing time of a packet is basically proportional to the duration of the packet. The longer the packet, the more time there will be available to do pushout. Therefore, our QPO+ scheme will not have a timing constraint bottleneck in hardware implementation.

4 Simulation Results

In this section, we implement our integrated services architecture and traffic management algorithms on our network simulator and use simulations to demonstrate the performance of our architecture and algorithms.

4.1 Simulation Settings

We use the *parking lot* network configuration for our simulation study (Fig. 2). On the connection level, we assume that a GS or CL flow's inter-arrival time is exponentially distributed with an average of 50 seconds, and the holding time is also exponentially distributed with an average of 100 seconds. The simulation parameters for the GS, the CL, and the BE service classes are shown in Table 1. For GS flows, we use the simple constant bit rate as their traffic pattern. For each BE flow, we use persistent TCP data traffic. For CL flows, we use an exponentially distributed on/off model with average $E(T_{on})$ and $E(T_{off})$ for on and off periods, respectively. During each on period, the packets are generated at peak rate r_p. Delay bound is obtained by the ratio of σ and ρ.

In Table 2, we list the simulation parameters at each end system and network components. Buffer size in Table 2 is the size of the entrance buffer before the leaky bucket. In our simulations, for GS flow j, $Ctot_j$ is assumed to be zero and $Dtot_j$ consists of only packet processing delays at all the switches along its path, i.e., $Dtot_j = Ltot_j \cdot D_j^{(k)} = Ltot_j \cdot 4\mu s$, where $Ltot_j$ is the number of switches along the path for flow j. We assume the propagation delay is $5\mu s$ per kilometer.

We organize our presentation as follows. Section 4.2 presents the performance of the GS, the CL, and the BE traffic under light and heavy load conditions and show that criteria C1 and C2 are satisfied. In Section 4.3, we show that our architecture and algorithms can effectively control non-conforming flows by minimizing their negative impact on other conforming flows (criterion C3). Sections 4.4 and 4.5 demonstrate the capabilities of ARC and QPO+, respectively.

[1] There is a subtle case we would like to address during this pushout process. It is possible that the longest queue i is only slightly (e.g. 1 bit) more than the queue j, to which the incoming packet belongs. If the incoming packet is large (i.e. maximum of 1500 bytes for Ethernet packet), the former longest queue i will become much shorter (at most 3000 bytes) than queue j after QPO+. We stress that the probability of such event is extremely small. On the other hand, the overall benefits of QPO+ far outweighs such minor drawback.

Table 1: Simulation parameters for each traffic class.

GS	Peak Rate r_p	1.5 Mbps
	Packet Size	1K bits
	Delay Bound	10 ms
CL	Peak Rate r_p	1.5 Mbps
	$E(T_{ON})$	2 ms
	$E(T_{OFF})$	2 ms
	Packet Size	1K bits
	Packet Loss Requirement	10^{-3}
	Delay Bound	20 ms
BE (TCP)	Peak Rate r_p (light load)	1 Mbps
	Peak Rate r_p (heavy load)	10 Mbps
	Mean Packet Processing Delay	300 μs
	Packet Processing Delay Variation	10 μs
	Packet Size	1K bits
	Maximum Receiver Window Size	64K bytes
	Default Timeout	500 ms
	Timer Granularity	500 ms
	TCP Version	Reno

Table 2: Simulation parameters at an end system and network components.

End System	GS	σ	15 packets
		ρ	1500 packets/s
		Buffer Size	10 packets
	CL	σ	20 packets
		ρ	1000 packets/s
		Buffer Size	10 packets
	TCP	Packet Processing Delay	500 μs
		Buffer Size	500 packets
Switch	Buffer Size	Conforming GS	250 packets
		Conforming CL	250 packets
		BE	1000 packets
		Non-conforming GS/CL	1000 packets
	Packet Processing Delay		4 μs
	B		100 Kbits
Link	Speed		10 Mbps
	Target Utilization		0.90
	Distance	End System to Switch	1 km
		Inter-Switch	1 km

Table 3: Number of GS, CL, and BE flows on each path under light and heavy load conditions in the parking lot network.

Traffic Type	Path	Number of Flows	
		Light Load	Heavy Load
GS	G1	1	1
	G2	1	1
	G3	1	1
	G4	1	1
CL	G1	1	2
	G2	1	2
	G3	1	2
	G4	1	2
BE (TCP)	G1	1	2
	G2	1	2
	G3	1	2
	G4	1	2

4.2 Performance Under Light and Heavy Load Conditions

Table 3 shows the number of flows on each path under light and heavy load conditions in our simulation. We repeat our simulations many times to obtain 95% confidence intervals. The 95% confidence intervals for the maximum end-to-end delay for GS and CL flows under light load are (1.521, 1.566) and (8.58, 9.09), respectively. The 95% confidence intervals for the maximum end-to-end delay for GS and CL flows under heavy load are (1.685, 1.733) and (14.05, 14.57), respectively. We find that the delays experienced by each GS and CL flows are bounded and are much less than the delay requirements for GS and CL flows, respectively. In Figs. 3 and 4, we plot the delay experienced by the GS flow and the CL flow traversing SW1 to SW5 (path G1) under light and heavy load, respectively. As shown in both figures, the delay experienced by this GS flow is bounded and is much less than its delay bound requirement (10 ms). For the CL flow, its delay is also bounded under both conditions and is less than its delay requirements (20 ms). As expected, there is some occasional delay increase for this CL flow under heavy load than under light load. Again, such increase is normal and is considered satisfying our performance objective for CL flows. Figure 5 shows the link utilization at Link45 during the light and heavy load conditions. There is no packet loss from any of the GS or CL flows under both light and heavy load conditions.

4.3 Control of Non-Conforming CL Flows

When CL sources or upstream elements do not have policing mechanism to control their traffic, the packets of a non-conforming CL flow may enter the CL buffer partition instead of the non-conforming GS/CL buffer partition [10]. We show that our architecture and algorithms can effectively control such non-conforming CL flows and thus achieve criterion C3.

The non-conforming flow is chosen to be a flow on path G4, which shares the bottleneck link Link45 with all other flows on paths G1, G2, and G3. The non-conforming flow submit a peak rate of 1.5 Mbps as its traffic parameter for admission control but transmits at

a peak rate of 10 Mbps. Since there is no policing mechanism for this flow, all packets from this flow enter the CL buffer partition.

Our simulations show that in the presence of non-conforming CL flow, the contracted QoS to those conforming GS/CL flows can still be guaranteed while the non-conforming flow is effectively isolated (due to per flow queueing) and suffers from large packet loss rate (due to QPO+ packet discarding). In particular, we plot the delay for the conforming GS and CL flows on path G1 in Fig. 6, which shows that the delays experienced by these GS and CL flows are bounded and are much less than their respective delay requirements. Furthermore the packet loss rate remains zero for all conforming flows during this simulation run. On the other hand, Fig. 7 shows the packet loss ratio experienced by the non-conforming CL flow. Furthermore, we find that the non-conforming CL flow does not have any significant effect on BE traffic either.

4.4 ARC or No ARC

To demonstrate the significance of our ARC algorithm described in Algorithm 1, we use the same simulation settings in Section 4.3. Here, instead of using ARC, we use calculated rate R_j, $j \in GS$ and measured rate $R_i = \alpha(\delta_i)$, $i \in CL$ directly as the weight in the WFQ scheduler.

Figures 8 and 9 show delay and loss of the same GS flow on path G1. Here, the delay bound of 10 ms is violated and there is also packet loss for this GS flow, while the delay for the same GS is bounded (see Fig. 6) with zero packet loss when ARC is employed.

4.5 QPO+ vs. Tail-dropping

We compare the performance of QPO+ with tail-dropping packet discarding scheme. Again, we use the same simulation settings in Section 4.3, except we discard the incoming packet when the buffer partition is full (tail-dropping) instead of QPO+. Poisson call arrival is not used, and instead, we just run the simulation for 300 seconds.

Figure 10 shows that under tail-dropping, even conforming CL flow experiences large packet loss, while the same conforming CL flow experienced zero packet loss under QPO+ in Section 4.3.

5 Concluding Remarks

This paper presents a framework of network architecture and traffic management algorithms to provide QoS provisioning for integrated traffic of the GS, the CL, and the BE services for the future integrated services networks. Our architecture are shown to be capable of meeting the performance criteria for integrated services networks and resolving several problems associated with the traditional class-based approach.

References

[1] H. J. Chao, "A delay-bound guarantee packet scheduler using a RAM-based searching engine," Pending for U.S. patent, file on Nov. 5, 1997.

[2] D. Clark, S. Shenker, and L. Zhang, "Supporting real-time applications in an integrated services packet network: architecture and mechanism," *Proc. ACM SIGCOMM,* Aug. 1992.

[3] S. Floyd, "Comments on measurement-based admissions control for controlled-load service," *Technical Report,* Lawrence Berkeley National Laboratory, July 1996.

[4] L. Georgiadis, R. Guerin, V. Peris and R. Rajan, "Efficient support of delay and rate guarantee," *Proc. ACM SIGCOMM'96,* Aug. 1996.

[5] S. Jamin, P. B. Danzig, S. Shenker, and L. Zhang, "A measurement-based admission control algorithm for integrated services packet networks," *IEEE/ACM Trans. on Networking,* vol. 5, no. 1, pp. 56–70, Feb. 1997.

[6] D. Lin and R. Morris, "Dynamics of Random Early Detection," *Proc. ACM SIGCOMM'97.*

[7] Y. S. Lin and C. B. Shung, "Quasi-Pushout cell discarding," *IEEE Commun.. Letters,* pp. 146–148, Sept. 1997.

[8] F. Lo Presti, Z.-L. Zhang, and D. Towsley, "Bounds, approximations and applications for a two-queue GPS system," *Proc. IEEE INFOCOM'96,* pp. 1310–1317, March 1996.

[9] S. Shenker, C. Partridge, and R. Guerin, "Specification of guaranteed quality of service," *RFC 2212,* Internet Engineering Task Force, Sept. 1997.

[10] J. Wroclawski, "Specification of the controlled-load network element service," *RFC 2211,* Internet Engineering Task Force, Sept. 1997.

[11] D. Wu and H. J. Chao, "Efficient bandwidth allocation and call admission control for VBR service using UPC parameters," *Proc. IEEE INFOCOM'99,* March 1999.

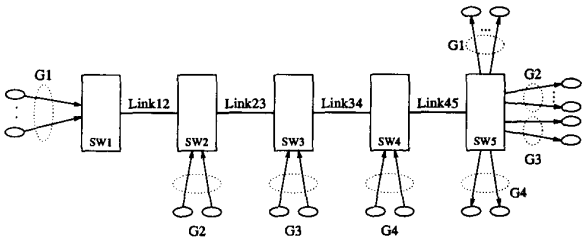

Figure 2: A parking lot network.

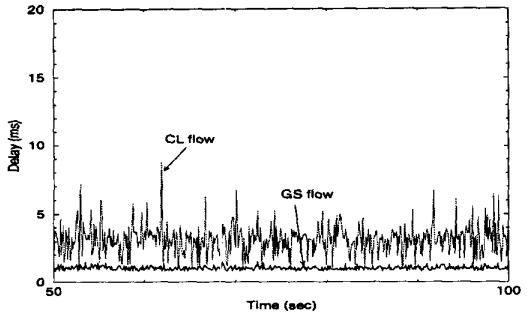

Figure 3: End-to-end delay of a GS flow and a CL flow under light load in the parking lot network.

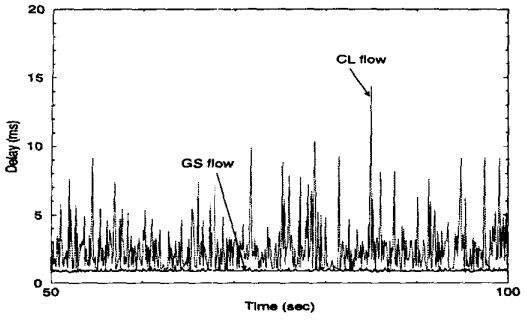

Figure 4: End-to-end delay of a GS flow and a CL flow under heavy load in the parking lot network.

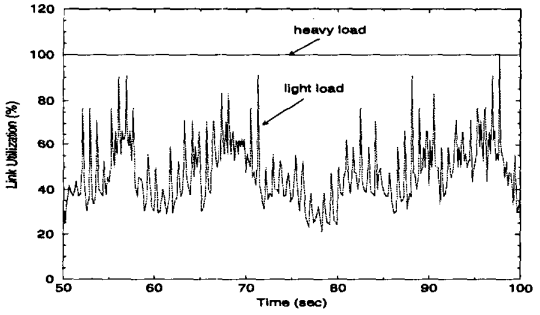

Figure 5: Link utilization of Link45 under light and heavy load in the parking lot network.

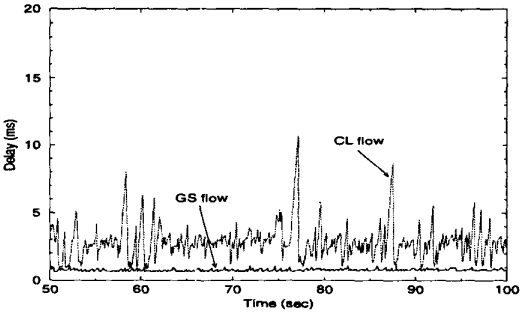

Figure 6: End-to-end delay for behaving GS and CL Flows in parking lot network.

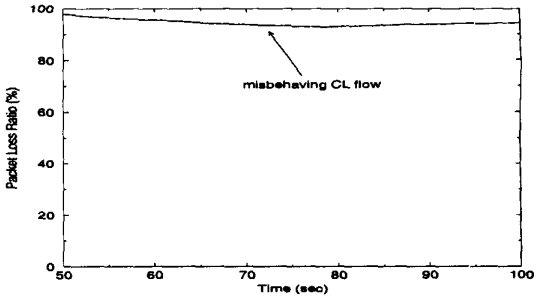

Figure 7: Packet Loss Ratio for the non-conforming CL flow in parking lot network.

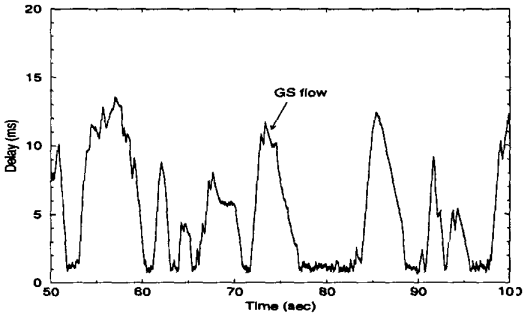

Figure 8: End-to-end delay for a GS Flow in parking lot network when ARC is not used.

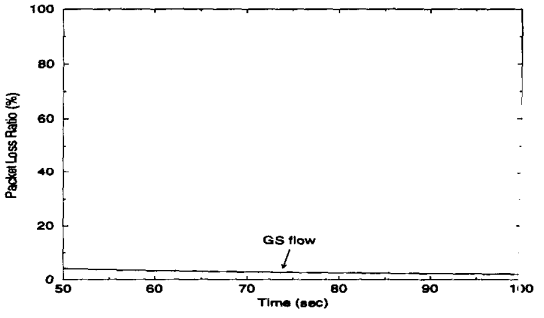

Figure 9: Packet loss ratio for a GS Flow in parking lot network when ARC is not used.

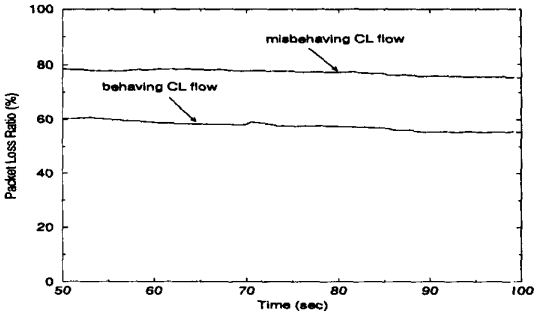

Figure 10: Packet loss ratio for behaving and misbehaving CL Flows in parking lot network using tail-dropping mechanism.