Design and Analysis of Traffic Control for Guaranteed Services on the Internet

PI-CHUNG WANG, CHIA-TAI CHAN[*] AND YAW-CHUNG CHEN
Department of Computer Science and Information Engineering
National Chiao Tung University
Hsinchu, Taiwan 300, R.O.C.
E-mail: pcwang@csie.nctu.edu.tw
[*]Telecommunication Lab.
Chunghwa Telecom Co., Ltd.
Hsinchu, Taiwan 300, R.O.C.

Real-time communications services will be one of the most promising applications on the Internet. Real-time traffic usually utilizes a significant amount of resources while traversing through the network. To provide guaranteed service for delay-sensitive real-time applications, both resource reservation and effective traffic control are necessary. The resource reservation protocol (RSVP) was developed to deal with the former problem while various approaches have been proposed to solve the latter one. It is crucial for guaranteed services to maintain satisfactory performance through effective traffic control, which must be implemented in a feasible way and be able to work with RSVP. In this article, we present a flexible and effective traffic-control approach with a resource reservation scheme to accommodate delay-sensitive services on the Internet. We demonstrate that our approach features proven bounded end-to-end queueing delay for delay-sensitive applications and good resilience for resource allocation. In addition, our packet transmission discipline achieves significant improvement in end-to-end delay control.

Keywords: internet, real-time traffic, delay-sensitive, RSVP, QoS, traffic scheduling

1. INTRODUCTION

The Internet is a heterogeneous network consisting of different subnetworks or autonomous systems, which are interconnected via gateways, routers, switches, and various transmission facilities. During the 1990s, the Internet grew rapidly as a global multimedia communications environment; and this trend necessitates guaranteed service for many applications, especially real-time data transport. In the presence of the congestion, the throughput of each traffic flow may degrade significantly. Here we use the term “flow” to refer to the packet stream that is transported from a source to a destination host. None of the needed quality guarantees can be fulfilled with the existing IP. To accommodate real-time services, we must allocate resources as well as provide an effective traffic control mechanism in each intermediate node. The main objective of this work is to design a control mechanism for bounded end-to-end delay and loss-free packet transmission to support delay-sensitive real-time traffic.

Received March 31, 1999; revised November 1, 1999; accepted November 26, 1999.
Communicated by Chyi-Nan Chen.

Transmission service disciplines can be categorized as work-conserving and nonwork-conserving. With the former, transmission is never idle as long as there is a packet waiting to be sent; examples are, Weight Fair Queueing (WFQ) [9], Virtual Clock [10], packet-by-packet Gener-alized Processor Sharing (PGPS) [11, 12], Carry-Over Round Robin [22] and Self-Clocked Fair Queueing (SCFQ) [13]. In the latter type, the server may be idle if no packets are eligible, even when there are packets to be sent; examples are Stop-and-Go framing (S&G) [5, 6], Hierarchical Round Robin (HRR) [7] and Rate Control Static Priority (RCSP) [8]. With a nonwork-conserving disci-pline, each packet is assigned, either explicitly or implicitly, an eligible time.

Work-conserving disciplines have less average queueing delay but more end-to-end delay-jitter than nonwork-conserving disciplines. With the work-conserving disciplines, traffic needs to be characterized based on each connection to fulfill both end-to-end delay bound and buffer space requirements. Since the traffic pattern may be blocked inside the network due to resource contention, it may be difficult to receive the sending traffic pattern at the destination. The nonwork-conserving disciplines feature certain advantages which support guaranteed service on the Internet. These advantages include feasible of end-to-end delay analysis and calculation of the required buffer space. On the other hand, there are some disadvantages: first, there is lower bandwidth utilization because transmission may be idle even when there is packet waiting for transmission. This under-utilized resource can be used by the best-effort services in the integrated service environment. Second, the performance of the best-effort services may be degraded. Based on the above pros and cons of the nonwork-conserving disciplines and the feasibility of implementation, it is appropriate to adopt the nonwork-conserving disciplines to provide guaranteed service.

Two well-known nonwork-conserving disciplines are the S&G frame strategy and RCSP. S&G is the most popular scheduler that naturally fits moving-window shapers. It features better end-to-end delay bound and a shorter delay-jitter bound than HRR, but bandwidth management is inflexible and the assigned frame size can rarely meet the applications' real needs [14, 15]. While RCSP provides nearly optimal packet delay jitter-control for real-time traffic, it has to pass the packet eligible time downstream node by node and synchronize the clock time of all the intermediate nodes precisely, which is an extremely difficult task on the Internet. These problems will be further discussed below.

The factors discussed above motivated us to design a traffic control method which can accommodate guaranteed service on the Internet. We adopted a framing strategy coupled with resource allocation in RSVP. Our approach uses a nonwork-conserving discipline with priority sorting to achieve the delay-performance guarantee. The merits of our proposed transmission control strategy are that it improves bandwidth utilization and offers feasible implementation. The organization of the rest of this paper is as follows. Section 2 presents the proposed resource reservation scheme for use on the Internet. In section 3, the proposed traffic control strategy is given. System simulation and performance evaluation are discussed in section 4. Section 5 concludes this paper.

2. RESOURCE RESERVATION BEHAVIOR

To fulfill the requirement of end-to-end guaranteed service on the Internet, resource reservation establishment instead of the traditional call-setup is necessary. The resource reservation protocol (RSVP) [3] is designed for the Internet and is used by a host to request a specific QoS from the network. There are three basic design principles behind RSVP. First, it is receiver-initiated, so a service request is generated by a receiver and then transmitted to the sender. The service requests, carried in RESV messages, travel in the reverse direction of the source-destination paths, as shown in Fig. 1. Second, the protocol is based on the notion of soft-state, which needs to be periodically refresh by the receivers. Rather than rely on the network to detect and respond failures, receivers in RSVP have to resend their service requests periodically; if a failure occurs, a refresh request will re-establish an appro-priate state. This is an important design principle because the reservation establishment messages need to be idempotent. That is, a service request must result in the same state installed in a router whether the request is a new or a retransmitted one. Third, the protocol itself is independent of the service model. RSVP carries client service requests to the routers, but it does not need to un-derstand the content of those requests. This allows RSVP to remain unchanged when the service model is extended.

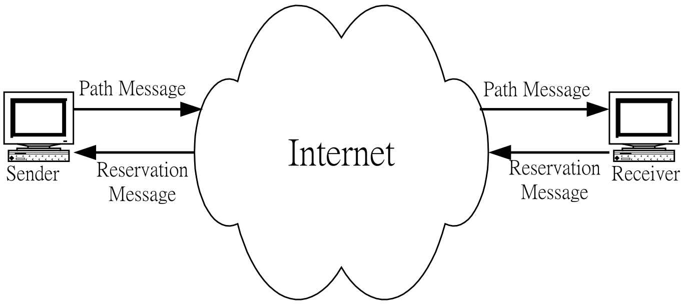


Fig. 1. Resource reservation protocol with the OPWA framework.

Since RSVP is a one-pass mechanism, a reservation request cannot, in any reasonable manner, allow a receiver to specify the end-to-end delay or the jitter bound. This is because the network does not know whether the desired end-to-end service objectives can be met by the resulted set-ting of reservations. As a result, the reservation request can only meaningfully specify the desired per-link service.

To solve the above problem, an improved type of one-pass service known as One Pass With Ad-vertising (OPWA) was proposed in [19]. With OPWA, an advertisement message is appended to a Path message so that each node can export characterization parameters to specify the quality of ser-vice (QoS) level which can be provided to the flows. The receiver uses the received information in the advertisement message to decide on the quantity of resources to be reserved and predicts the end-to-end QoS. Then, the receiver can construct, or dynamically adjust, an appropriate reservation request.

Generally, the end-to-end delay bound, excluding the processing overhead in the end systems, is the sum of the per-link delay bounds. How to decide on the amount of resources to reserve at an intermediate node still needs further study. In any case, the resource allocation method tried to increase the level of network utilization in order to satisfy the service requirement. We present a basic allocation scheme here. First, we divide the resources provided by an intermediate node into several levels and assume that 64Kbps is the unit of resource allocation. There may be two situations: one is that every node can provide sufficient resources, and the other is that some nodes are unable to provide the required resources. As shown in Fig. 2, each rectangle in a node represents one resource unit, and the shaded parts are the resources available in each node. The dotted line indicates that the required resources are equally divided into shares by each node. If the total resources provided are still insuf-ficient, the reservation will fail. Otherwise, we can allocate more resources for these nodes with less contention. The allocation algorithm is as follow:

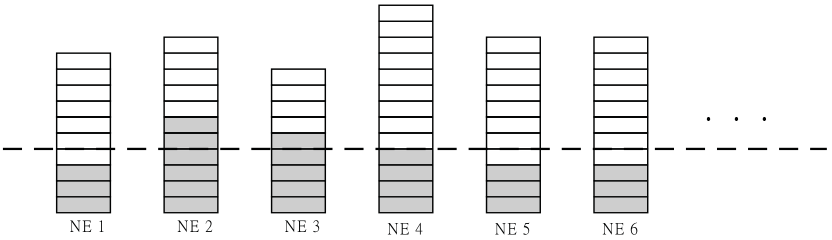


Fig. 2. Resource allocation in the network.

Definition:

L: sum of difference between equal share and provided resources for all nodes which do not have enough resources.

O: sum of difference between equal share and provided resources for all nodes which have enough resources.

S: equally divided resources

R_i: excessive resources provided, where $R_1 > R_2 > .... > R_j$

N_i: number of sources with excess resources R_i

Algorighm

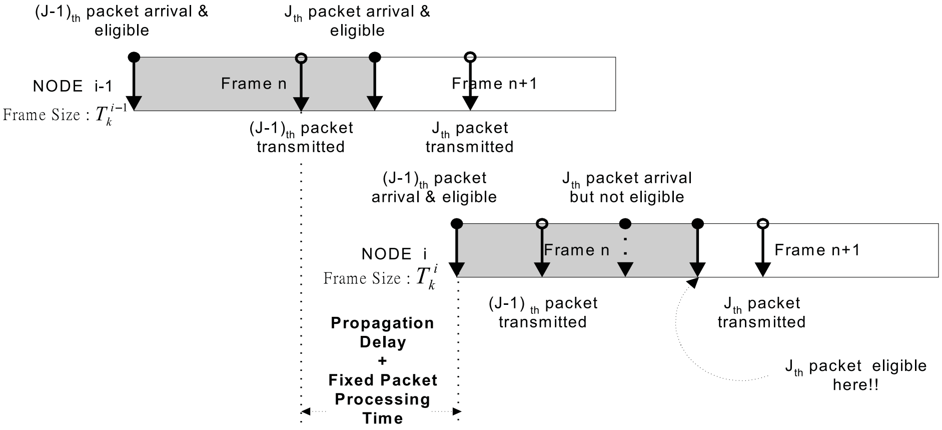
```
Calculate $S$.
If all nodes can satisfy the requirement, then allocate $S$ at each node.
Else do
    calculate $L$ and $O$.
    If ($O < L$), then it fails to satisfy the requirement.
    Else do
        we sort the nodes based on $R$ where $R_1 > R_2 > ... > R_j$. Nodes with equal amounts of
        excess resources have the same priority.
        Allocate resources from the nodes which do not have enough resources.
        For nodes with the highest priority do
            If ($O \leq (R_1 - R_2) \times N_1$), allocate $\left\lceil \frac{O}{N_1} \right\rceil$.
            Else, allocate ($R_1 - R_2$), update the priority and set $O = O - (R_1 - R_2) \times N_1$. Repeat
the previous step until $O \leq zero$.
```

3. TRAFFIC CONTROL STRATEGY FOR DELAY-SENSITIVE TRAFFIC

To accommodate the hard real-time traffic and delay-sensitive services on the Internet, an effec-tive traffic control mechanism must provide a guaranteed end-to-end delay bound, tolerable delay jitter, and a stringent packet-loss ratio. The proposed service discipline is based on the following considerations. One is the resolution of service contention for incoming smoothed traffic. The other is that the service discipline must maintain a traffic pattern which is as close to the original smoothed pattern as possible. It, thus, has to obey the following rules:

(1) According to different service types, the service contention between incoming packets with various delay requirements may tolerate varying queueing delays. In order to meet the delay requirements for each service type, priority ordering of transmission service is set to prefer the smallest frame size if the delay requirement is still the same. However, this priority ordering scheme can be set in a flexibly manner according to the real service needs.
(2) Without proper packet transmission scheduling in a node, consecutive packets may be in-ter-clustered at downstream nodes and may cause even larger bursts. To keep the input traffic patterns as smooth as possible, just one packet can be transmitted within a specific time frame interval for each flow.
(3) Transmission will be idle if there is no eligible packet in the queue. An eligible packet is defined as a packet that is competing with other packets for the next transmission.
(4) The time frame size is derived from the expected packet inter-arrival time. Each flow reserves resources and obtains a guaranteed bandwidth (R) with a time frame size of T, i.e., $R = L / T$, where L is the maximum packet size. The source traffic for each flow is regulated according to its time frame size T.

Assume that the frame size of the k_{th} connection at the $(i-1)_{th}$ node is T_k^{i-1}. As shown in Fig. 3, the $(j-1)_{th}$ and the j_{th} packet from the k_{th} connection arrive at the $(i-1)_{th}$ node with inter-arrival time T_k^{i-1}, where the packet arrival time is defined as the arrival time of its last bit



$(J-1)_{th}$ packet arrival & eligible
J_{th} packet arrival & eligible
NODE i-1
Frame Size : T_k^{i-1}
Frame n
Frame n+1
$(J-1)_{th}$ packet transmitted
J_{th} packet transmitted
$(J-1)_{th}$ packet arrival & eligible
J_{th} packet arrival but not eligible
NODE i
Frame Size : T_k^i
Frame n
Frame n+1
$(J-1)_{th}$ packet transmitted
J_{th} packet transmitted
Propagation Delay + Fixed Packet Processing Time
J_{th} packet eligible here!!

Fig. 3. Traffic scheduling discipline (I).

plus the fixed packet processing time. Here, both packets will be eligible for immediate transmission. However, due to contention, the departure delay of the $(j-1)_{th}$ packet is close to T_k^{i-1} while the departure delay for the j_{th} packet may be close to 0. When the j_{th} packet arrives at the i_{th} node, the inter-arrival time between the j_{th} and the $(j-1)_{th}$ packet will be smaller than T_k^{i-1}. Once the $(j-1)_{th}$ packet becomes immediately eligible, the j_{th} packet at the i_{th} node must wait for eligibil-ity until the the $(j-1)_{th}$ packet is eligible plus the time frame size T_k^i. Moreover, if the eligibility waiting time of the $(j-1)_{th}$ packet is W, then the j_{th} packet should wait $W + T_k^i$ for eligibility, as shown in Fig. 4.

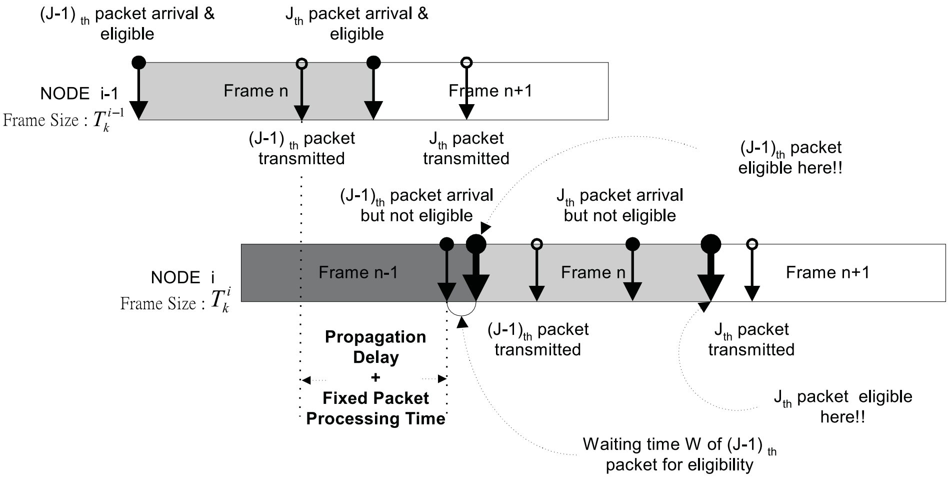

(J-1) th packet arrival & eligible
Jth packet arrival & eligible
NODE i-1
Frame Size : T_k^{i-1}
Frame n
Frame n+1
(J-1) th packet transmitted
Jth packet transmitted
(J-1)th packet eligible here!!
(J-1)th packet arrival but not eligible
Jth packet arrival but not eligible
NODE i
Frame Size : T_k^i
Frame n-1
Frame n
Frame n+1
Propagation Delay + Fixed Packet Processing Time
(J-1)th packet transmitted
Jth packet transmitted
Jth packet eligible here!!
Waiting time W of (J-1) th packet for eligibility

Fig. 4. Traffic scheduling discipline (II).

Based on our proposed method, we can show that a packet of the k_{th} connection with time frame size T_k^i can be transmitted within a T_k^i time interval after it becomes eligible at the i_{th} node. The explanation for this is as follows: the packet scheduler processes the incoming smoothed traffic of each flow which contains at most one packet within a time frame. Therefore, the maximum number of packet arrivals during the time interval T_k^i from these flows with a time frame size less or equal to T_k^i is $N = \sum_{j=1}^{i} n_k^j \times \left(\left\lceil \frac{T_k^i}{T_k^j} \right\rceil\right)$, where n_k^j is the total number of flows with frame size $T_k^j\,(T_k^j \le T_k^i)$. Since the admission control will not accept excessive input traffic, the inequality must hold for any k. It can be shown that an incoming packet with time frame size T_k^i and eligible time t can be transmitted within a T_k^i time interval. This is true for every k because k can be arbitrarily chosen.

Given that the k_{th} flow has a time frame size of T_k^i, we use a timer $T_{p,k}(i)$ to maintain a logical time framing, and a packet-transfer indicator $P_{tran,k}(i)$ to indicate whether a packet is eligible or not. $P_{tran,k}(i)$ is set to 1 once a packet becomes eligible and is reset to 1 if the current clock time exceeds the time variable $T_{p,k}(i)$. Therefore, the packet will become eligible only when $P_{tran,k}(i) = 0$. According to rule 2, when a packet arrives, it checks whether the packet-transfer indicator $P_{tran,k}(i)$ is equal to 0 or not; if it is, then the incoming packet will become eligible immediately, and vice versa. Therefore, the packet scheduler can regulate and maintain a smooth traffic pattern for each flow. The algorithm for the packet scheduler is described below.

Algorithm for Packet Scheduler:
Declaration:

N: the total number of flows at the i_{th} node
$T_{p,k}(i)$: the timer of the k_{th} flow at the i_{th} node $(1 \le k \le N)$
$P_{tran,k}(i)$: the packet transmission status bit, where $P_{tran,\,i}(k)$ is set to 1 if a packet transferred
T_k^i: the time frame size of the k_{th} connection at i_{th} node $(1 \le k \le N)$.
L_k: the k_{th} queue length. L_k is equal to 0 if the queue is empty and is equal to 1 otherwise
clock: current clock time in the switch

```
Begin
  while (bandwidth resource (time frame size $T_k^i$) of $k_{th}$ flow is reserved) do
    $T_{p,k}(i) = 0$;
  while (buffer is not empty) do
    for all ($L_k \neq 0$) do
      if ($T_{p,k}(i) \le clock$) then
        $T_{p,k}(i) = T_k^i + clock$;
        $P_{tran,k}(i) = 0$;
      endif
     for all (($L_k \neq 0$) and ($P_{tran,k}(i) < 1$)) do;
        $P_{tran,k}(i) = 1$;
         insert the HOL of $k_{th}$ queue to the SPC;
     endfor
  endwhile
End
```

Before analyzing the proposed strategy, we will introduce some formal notations. Given a flow k with time frame size T_k^i in the i_{th} node, flow k passed through H hops. Let the sum of the processing time and the propagation delay in each link be τ_l, where $l = 0... H$ and the access link of the connection is represented by l_0; it becomes clear that the end-to-end queueing delay of the j_{th} packet for a given flow k is $QD_k^j = \sum_{i=1}^{H} Q_k^j(i)$, where $Q_k^j(i)$ is the queueing delay of the j_{th} packet in the i_{th} node $(0 \le Q_k^{j,i} < 2T_k)$. It is worth noting that $Q_k^j(i)$ may exceed T_k^i. A theorem for the traffic property under the proposed traffic control strategy is given as follows:

Theorem 1: For any flow k with time frame size T_k^i, the end-to-end queueing delay $QD_k^j = \sum_{i=1}^{H} Q_k^j(i)$ is bounded by $[T_k^H + \sum_{i=2}^{H} T_k^i]$, where $Q_k^j(i)$ is the queueing delay of the j_{th} packet in i_{th} node $(0 \le Q_k^j(i) < 2T_k^i)$.

Proof: Let us define $W_k^j(i)$ as the service waiting delay of the j_{th} packet in the k_{th} flow at the i_{th} node after it becomes eligible. The inequality,

$$0 \le W_k^j(i) \le Min(T_k^i, Q_k^j(i)), where \begin{cases} j = 1,2\cdots \\ i = 1\cdots H \end{cases}, \tag{1}$$

must hold for any packet in the k_{th} flow.

The difference between $Q_k^j(i)$ and $W_k^j(i)$ is the time interval, $q_k^j(i)$, for the j_{th} packet to become eligible after it arrives at the i_{th} node, where

$$Q_k^j(i) = q_k^j(i) + W_k^j(i) \text{ and} \tag{2}$$

$$0 \le q_k^j(i) \le T_k^i. \tag{3}$$

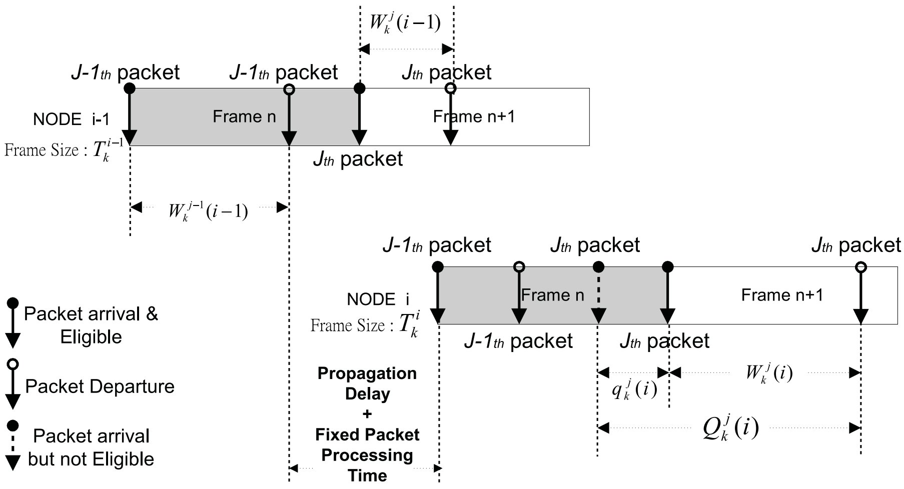


Fig. 5. Analysis of the packet queueing delay on the node.

As shown in Fig. 5, the equation

$$q_k^j(i) = \begin{cases} 0 & if(\left\lfloor T_k^{i-1} - W_k^{j-1}(i-1) \right\rfloor + W_k^j(i-1) \le 0) \\ T_k^i - \left[T_k^{i-1} - W_k^{j-1}(i-1)\right] + W_k^j(i-1) & Otherwise \end{cases} \tag{4}$$

must hold for any packet of the k_{th} flow. If the j_{th} packet becomes eligible immediately after it arrives (i.e., $\left[T_k^{i-1} - W_k^{j-1}(i-1)\right] + W_k^j(i-1) \le 0$), then the queueing delay of the j_{th} packet ($Q_k^j(i)$) must be smaller than T_k^i in the i_{th} node. On the other hand (i.e., $q_i^j \ne 0$), the queueing delay of the j_{th} packet may be larger than T_k^i in the i_{th} node. When the first packet of a flow arrives, since the packet-transfer indicator $P_{tran,k}(i)$ is equal to 0, it will become eligible immediately, i.e., $q_k^1(i) = 0$, where $1 \le i \le H$.

From the above expressions, the end-to-end queueing delay of the j_{th} packet $QD_k^j = \sum_{i=1}^{H} Q_k^j(i)$ can be written as

$$\begin{aligned} QD_k^j &= \sum_{i=1}^{H}(q_k^j(i)+W_k^j(i)) \\ &= W_k^j(1)+q_k^j(1)+W_k^j(2)+\cdots+q_k^j(H)+W_k^j(H) \\ &\le W_k^j(1)+\left[T_k^2-(T_k^1-W_k^{j-1}(1))+W_k^j(1)\right]+ \\ &\quad W_k^j(2)+\left[T_k^3-(T_k^2-W_k^{j-1}(2))+W_k^j(2)\right]+ \\ &\quad W_k^j(3)+\cdots+W_k^j(H-1)+\left[T_k^H-(T_k^{H-1}-W_k^{j-1}(H-1))+W_k^j(H-1)\right]+W_H^j. \end{aligned} \tag{5}$$

By eliminating the same terms in (5), we obtain $QD_k^j \le T_k^H - T_k^1 + \sum_{i=1}^{H} W_k^{j-1}(i)$. From inequality (1), we have

$$QD_k^j \le T_k^H + \sum_{i=2}^{H} T_k^i \tag{6}$$

This completes the proof. Q.E.D.

Therefore, the end-to-end delay of the j_{th} packet for a given flow k is

$$D_k^j = \tau^k + QD_k^j \tag{7}$$

where $\tau^k(\tau^k = \sum_{i=0}^{H}\tau_i)$ is the total propagation delay plus the processing time and QD_k^j is the total queueing delay of the j_{th} packet. To further analyze of the smoothness property, we will now investi-gate the relationship between the incoming packets. Given a source traffic with a time frame size T_k^i, where $T_k^i = \max(T_k^j)$ $1 \le j \le H$, the transmission rate of the source must be less than *maxium packet size*/T_k^i. Owing to service contention among different flows, transmission of packets may be delayed. This will cause the time interval between two consecutive packets in the same flow to be less then T_k^i. More precisely, it seems possible that two consecutive packets will be inter-clustered back to back. According to the packet scheduling algorithm, traffic smoothness can be maintained in each flow. Traffic in a flow with a maximum time frame size of T_k^i always remains smooth throughout the Internet. If there are N flows in the node, it is possible that the k_{th} flow ($1 \le k \le N$) will cause the queue length to be greater than 1 under the proposed service discipline. Assume that j_{th} and $(j+1)_{th}$ are two consecutive packets; the arrival of the $(j+2)_{th}$ packet will not cause the length in the k_{th} queue to increase. Since the source rate is bounded by the largest time frame size, the j_{th} incoming packet will always be transmitted before the $(j+2)_{th}$ packet arrives. Buffer space sufficient to hold 2 packets is large enough for each flow to avoid overflow. Thus, a total buffer space of $2N \times$ *the maximum packet size* is sufficient in each output port using our proposed method.

Obviously, the end-to-end queueing delay is reduced from $2HT$, which is provided by S&G, to no more than $T_k^H + \sum_{i=2}^{H} T_k^i$, and there is much more flexibility. The major advantage as compared with S&G is the significant improvement in both the maximum and the average end-to-end queue-ing delays as well as the accommodation of an arbitrary time frame size. Clearly, our method can pro-vide not only guaranteed delay performance for delay-sensitive traffic on the Internet, but also flexible resource allocation for RSVP. The main reason for the bounded queueing delay guarantee is the fact that the strict transmission control keeps the traffic patterns smooth throughout the Internet. Since best-effort service can tolerate longer delay time compared with delay-sensitive traffic, the utilization performance can be improved by accounting for the provision of best-effort service.

4. SYSTEM SIMULATION AND PERFORMANCE EVALUATION

4.1 System Simulation Model

Assumptions regarding the network and traffic flow are given in this section along with the default values of the parameters used in the system simulation. For delay-sensitive traffic, there is a strong correlation between successive packet arrivals. The assumptions about the characteristics of the traffic source significantly influence the performance such as in packet-loss and packet-delay. The two-phase burst/silence (or ON/OFF) model was used in our simulation. During ON periods, the packets arrive at the peak rate, and a fixed number of packets are generated by the source. The duration of a burst is assumed to have a geometric distribution while the duration of a period of silence is assumed to have a negative-exponential distribution. Our simple model consists of four nodes, 1 end host and 56 end systems that function as input sources, as shown in Fig. 6. Let these 57 flows be established in the system model: Source 0 connects to node 1 and then goes through node 2 to node 4. Source i ($i \neq 0$) connects to the $\left(\left\lceil {}^{i}\!/_{14} \right\rceil\right)_{th}$ node and goes through the $\left(\left\lceil {}^{i}\!/_{14} \right\rceil + 1\right)_{th}$ node. All the above nodes were interconnected with 155.52 Mbps link capacity. For simplicity, we assumed that the traffic of source 0 was a stored MPEG video with a 3 Mbits/s average rate and a burst duration of around 100 Kbits. The other 56 sources were servers which had higher service priority. The packet size was limited to 1.5 Kbytes (Ethernet MTU), and the peak rate was 10 Mbits/s. Each source had the same time frame size T_i = 15 packet time; *i.e., the packet time was the transmission time of the maximum-size packet*, $0 \leq i \leq 57$. To compare the performance with that of S&G, the time frame size of source 0 at each node was the same. The performance measurement considered here was the bounded end-to-end queueing delay and the bandwidth utilization of source 0 in the worst case, i.e., where source 0 had the lowest priority during service contention.

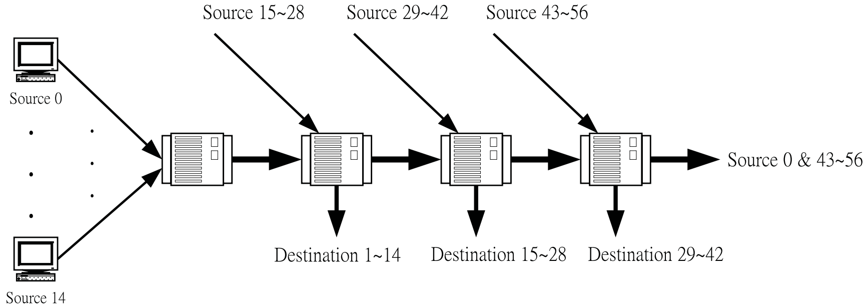


Fig. 6. The system model for delay-sensitive traffic control in the Internet.

4.2 Numerical Results

Proposed strategy versus S&G in end-to-end queueing delay and bandwidth utilization.

Average end-to-end queueing delay versus traffic intensity: The results for the average end-to-end queueing delay for different packet schedulers under different traffic intensities are presented here. The traffic intensity varied from 0.1 to 0.9. From the simulation results shown in Fig. 7, the proposed traffic control featured lower average end-to-end delay. The major reason is that an incoming packet became eligible immediately if there no packet was transmitted during the last time frame interval. This led to a shorter queueing delay and reduced the mismatch time that may occur with S&G. The end-to-end queueing delay achieved using our proposed strategy was better than that obtained using S&G. Also, the queueing delay for any flow never exceeded its upper bound even when traffic intensity was a high as 0.9.

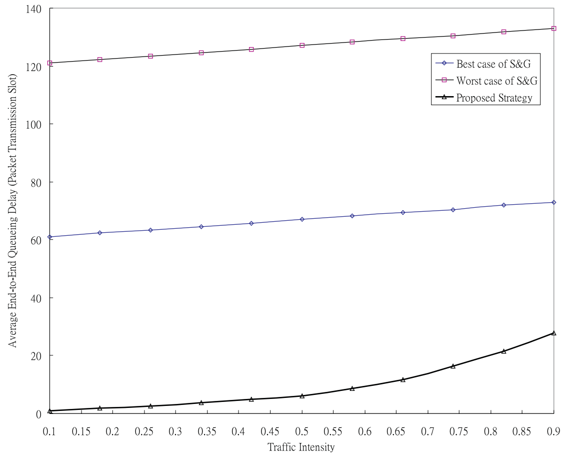


Fig. 7. Average end-to-end packet queueing delay of source 0 under the proposed traffic control and S&G.

End-to-end packet queueing delay distribution: As discussed earlier, the average queueing delay under work-conserving disciplines is shorter than that under nonworking-conserving disciplines, such as PGPS and FIFO. However, the end-to-end delay distribution of work-conserving dis-ciplines has a much wider range. Since bounded delay-jitter allows the destination-end system to determine the proper buffer space for eliminating jitter, it may lead to buffer overflow. In order to evaluate the end-to-end queueing delay under heavy loads, the traffic intensity is set to be nearly 1.0 in each node. The following results show that the proposed scheme provides guaranteed service for delay-sensitive traffic.

In Fig. 8, we can observe the results for both the best case and the worst case. In the former, the mis-match time at each node is 0. But in the worst case, the mismatch time is T. In the best case, the end-to-end queueing delay is bounded by $HT - T$ and $HT + T$. But in the worst case, the end-to-end queueing delay is bounded by $2HT - T$ and $2HT + T$. The above distribution is based on 58% traffic intensity. If we increase the load to 90%, delay jitter using S&G decreases because source 0 has the lowest priority, and because its departure time simply becomes larger. However, there is no significant change under our proposed strategy, as shown in Fig. 9.

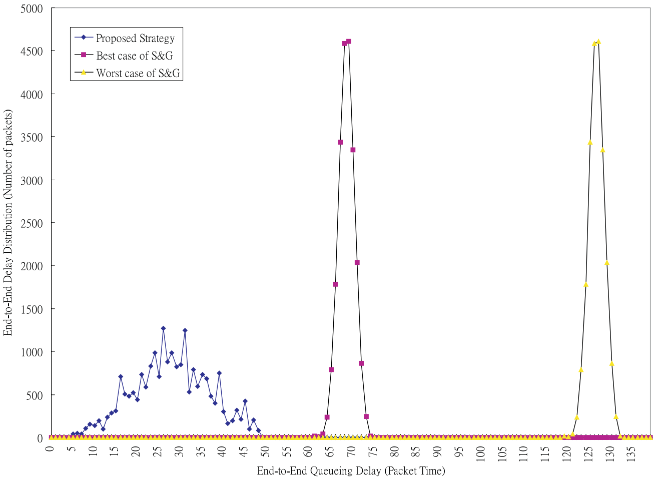


Fig. 8. End-to-end queueing delay distribution of source 0 using the proposed strategy and the S&G strategy (Traffic Intensity = 58%).

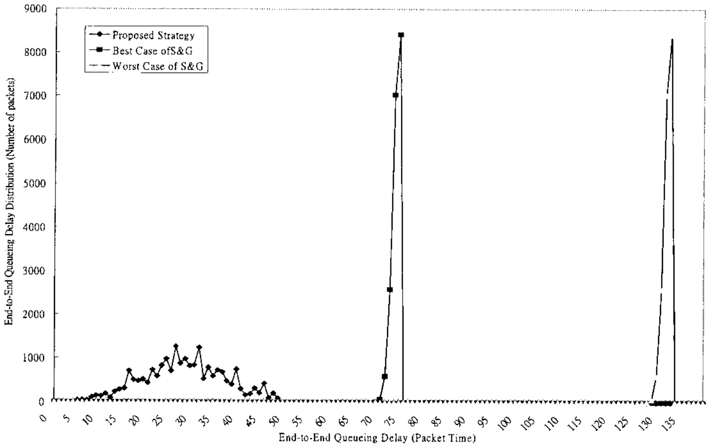


Fig. 9. End-to-end queueing delay distribution of source 0 under the proposed strategy and S&G strategy (Traffic Intensity = 90%).

Bandwidth Utilization: As shown in Fig. 10, bandwidth utilization under S&G is much lower than that under our proposed strategy, especially when the traffic intensity is low. However, utilization under S&G improves when the traffic intensity increases. Since higher utilization can be achieved using the proposed strategy, the bandwidth available for best-effort service will naturally be lower.

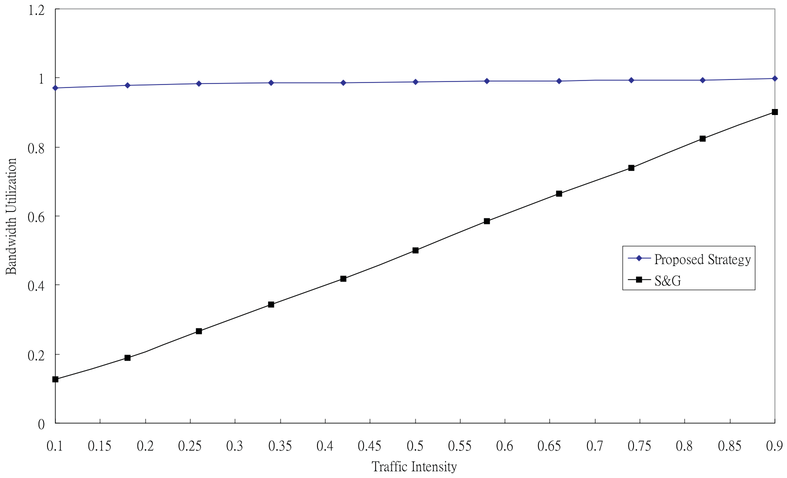


Fig. 10. Bandwidth utilization under the proposed strategy and the S&G strategy.

4.3 Numerical Analysis

The S&G framing strategy guarantees bounded end-to-end delay and packet loss free transmis-sion for each flow. However, it leads to greater synchronization delay since the framing of incoming and departing frames must be synchronized at each intermediate node. In addition, the S&G policy has to incorporate multiple frame sizes according to different delay requirements. Each frame size must be multiplicative of the basic frame size. Its bandwidth management is inflexible, and the assigned frame size rarely satisfies the practical needs. Our proposed strategy removes the extra waiting time by node synchronization and uses various frame sizes based on the packet generation rate for each flow; hence, flows are allowed to have arbitrary frame sizes in a node. This increases the flexibility in resource reservation. Furthermore, bandwidth utilization is improved.

5. CONCLUSIONS

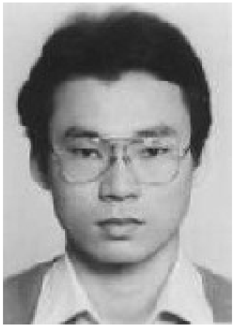
In this work, we have proposed a simple traffic control technique combined with a resource reservation scheme to guarantee the performance of delay-sensitive traffic on the Internet. It has been suggested that resource control, access control and transmission control can be combined to obtain a more effective traffic control method. In the proposed strategy, the objective of the transmission discipline is achieved by regulating the traffic flow, which can be kept as close to smooth input traffic as possible throughout the Internet. The proposed strategy reduces the average end-to-end delay by removing extra waiting time in node synchronization. However, the total memory requirement of each output port is directly proportional to the total number of active flows, which may result in a large memory buffer and efficient buffer management. By using the proposed method, a remarkable bounded end-to-end delay can be achieved, and QoS guaranteed delay-sensitive service can be provided on the Internet.

REFERENCES

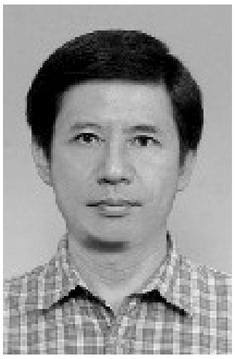
1. S. Shenker and J. Wroclawski, "Network element service specification template," *Internet Draft*, 1996, <draft-ietf-intserv-svc-template-03.txt>.
2. S. Shenker, C. Partridge, and R. Guerin, "Specification of guaranteed quality of service," *Internet Draft*, 1996, <draft-ietf-intserv-guaranteed-svc-06.txt>.
3. R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource reservation protocol (RSVP)-version 1functional specification," *Internet Draft*, 1997, <draft-ietf-revp-spec-13.ps>.
4. L. Georgiadis, R. Guerin, V. Peris, and R. Rajan, "Efficient support of delay and rate guarantees in an internet," *ACM Special Interest Group on Data Communications* '96, 1996, pp. 106-116.
5. S. Golestani, "A framing strategy for congestion management," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 7, 1991, pp. 1064-1077.
6. S. Golestani, "Congestion-free for communication in high speed packet network," *IEEE Transactions on Communications*, Vol. 39, No. 12, 1991, pp. 1802-1812.
7. C. Kalmanek, H. KanaKia, and S. Keshav, "Rate controlled services for very high-speed networks," *IEEE Global Telcommunication Conference*, 1990, pp. 300.3.1-300.3.9.
8. H. Zhang and D. Ferrari, "Rate-controlled static priority queueing," *IEEE Conference on Computer Communications* '93, 1993, pp. 227-236.
9. A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *Journal of Internetworking Research and Experience*, Vol. 1, 1990, pp. 3-26.
10. L. Zhang, "Virtual clock: a new traffic control algorithm for packet switching networks," in *Proceedings of ACM Special Interest Group on Data Communications* '90, 1990, pp. 19-29.
11. A. Parekh and R. Gallager, "A generalized processor sharing approach to flow control in integrated services net-works: The single-node case," *IEEE Transactions on Networking*, Vol. 1, No. 3, 1993, pp. 344-357.
12. A. Parekh and R. Gallager, "A generalized processor sharing approach to flow control in integrated services net-works: The multiple node case," *IEEE Transactions on Networking*, Vol. 2, No. 2, 1994, pp. 137-150.
13. S. Golestani, "A self-clocked fair queueing scheme for broadband applications," *IEEE Conference on Computer Communications* '94, 1994, pp. 636-646.
14. H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," in *Proceedings of the IEEE*, Vol. 83, No. 10, 1995, pp. 1374-1396.
15. H. Zhang, "Providing end-to-end performance guarantees using nonwork-conserving disciplines," *Computer Communications*, Vol. 18, No. 10, 1995, pp. 769-781.

16. H. J. Chao and N. Uzun, “Priority management in ATM switching nodes,” *IEEE/ACM Transactions on Networking*, Vol. 3, No. 6, 1995, pp. 418-427.
17. H. J. Chao and N. Uzun, “A VLSI sequencer chip for ATM traffic shaper and queue management,” *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 11, 1992, pp. 1634-1643.
18. M. W. Garrett, “A service architecture for ATM: from applications to scheduling,” *IEEE Network*, Vol. 10, No. 3, 1996, pp. 6-14.
19. S. Shenker and Lee Breslau, “Two issues in reservation establishment,” in *Proceedings of ACM Special Interest Group on Data Communications* ’95, Vol. 25, No. 4, 1995, pp. 14-26.
20. D. Ferrari, “Client requirements for real-time communication services,” *IEEE Communications Magazine*, Vol. 28, No. 11, 1990, pp. 65-72.
21. D. Ferrari and D. Verma, “A scheme for real-time channel establishment in wide-area networks,” *IEEE Journal on Selected Areas in Communications*, Vol. 8, No. 3, 1990, pp. 368-379.
22. D. Saha, S. Mukherjee, and S. K. Tripathi, “Carry-over round robin: a simple cell scheduling mechanism for ATM networks,” *IEEE/ACM Transactions on Networking*, Vol. 6, No. 6, 1998, pp. 779-796.

Pi-Chung Wang (王丕中) received both his B.S. degree and M.S. degree in Computer Science and Information Engineering from National Chiao Tung University in 1994 and 1996. He is currently pursuing his Ph.D. degree in computer science. His research interests include Internet multimedia communications and traffic control on high speed networks.



Chia-Tai Chan (詹家泰) received his Ph.D. degree in computer science from National Chiao Tung University, Hsinchu, Taiwan in 1998. He is now with the Telecommunication Laboratories Chunghwa Telecom Co., Ltd. His current research interests include the design, analysis and traffic engineering of broadband multiservice networks.



Yaw-Chung Chen (陳耀宗) received his Ph.D. degree in computer science from Northwestern University, Evanston, Illinois in 1987. During 1987-1990, he worked at AT&T Bell Laboratories as a Member of Technical Staff. Since 1990, he has been an associate professor in the Department of Computer Science and Information Engineering of National Chiao Tung University. His research interests include multimedia communications, high speed networking, and ATM traffic management. He is a member of IEEE and ACM.