1998 International Computer Symposium
Workshop on Computer Networks, Internet, and Multimedia
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

A Characteristic Based Scheduling Scheme for VBR Traffic over ATM Networks

Shuo Cheng Hu, Yu Fu Wang, and Yaw Chung Chen

Department of Computer Science and Information Engineering
National Chiao Tung University, Hsinchu, Taiwan 300, R.O.C.
Tel: 886-35-731864, Fax: 886-35-727842, E-mail: {schu, yfwang, ycchen}@csie.nctu.edu.tw

ABSTRACT

In this work, we investigate an ATM cell scheduling scheme based on traffic characteristics and QoS requirements. This so-called characteristic-based scheduler (CBS) uses both time frame concept and priority queueing. The frame size of a virtual connection (VC) is determined according to its reserved bandwidth and Maximum Burst Size (MBS). From the analysis and simulation results, we show that CBS features isolation of flows, guaranteed end-to-end delay, high utilization, and simplicity of implementation. It is expected to accommodate the real-time VBR traffic with guaranteed QoS requirements.

1. INTRODUCTION

ATM is a connection-oriented protocol, in which each connection has its own characteristics and QOS requirement. There is a trade-off between the implementation complexity and the guaranteed QoS. For example, PGPS-like scheduling algorithms [2,3,4,5,6] allow the network to make a range of worst-case performance guarantees on both throughput and delay, whereas Round Robin algorithms [1,6] only let the network get a throughput guarantee. In fact, the implementation of the former is much more complex because it needs to calculate the virtual time and selects the highest priority packet in the system. We will investigate an approach that guarantees both throughput and delay with simpler implementation.

We propose a combined service discipline based on both Characteristic Based Scheduling (CBS) and traffic shaping by leaky bucket to provide an efficient and fair use of links for VBR traffic. CBS aims at the following goals: (1) Isolation of flows. (2) Guaranteed end-to-end delay: it provides end-to-end cell transfer delay guarantee for individual connections. (3) High utilization: CBS uses a work-conserving service discipline, so the server will be busy as long as there is a cell queued in the buffer. (4) Simplicity of implementation: in ATM networks, the time available for completing a scheduling decision is very short. For SONET OC-3 the cell transmission time is less than 2.8 μs, it is even shorter for higher transmission speed such as OC-12. Since CBS is a frame-based scheduler, it is easy to implement with satisfactory performance.

The rest of this paper is organized as follows. In Section 2, the CBS mechanism is discussed in detail. In Section 3, we present the theoretical analysis of CBS. In Section 4, the simulation is addressed, and results are discussed. Section 5 concludes the work.

2. CHARACTERISTIC BASED SCHEDULING SCHEME

We present the CBS architecture and propose a CAC mechanism to ensure that every flow is schedulable, then the CBS service discipline is discussed.

2.1 The CBS Architecture

The architecture of CBS is shown in Figure 1. The switch uses per VC queueing. Each VC queue maintains two variables, Bank Counter (BC) and Timer. Two parameters, the service credit Q and the service period T are associated with them respectively. These parameter values can be preset according to the traffic characteristics and the delay requirement. Timer will count up based on our defined rule. Once the value of Timer reaches T, BC is set to Q. With these two variables and two parameters, each backlogged VC queue can transmit an amount of Q during the period of T. The detail will be explained later.

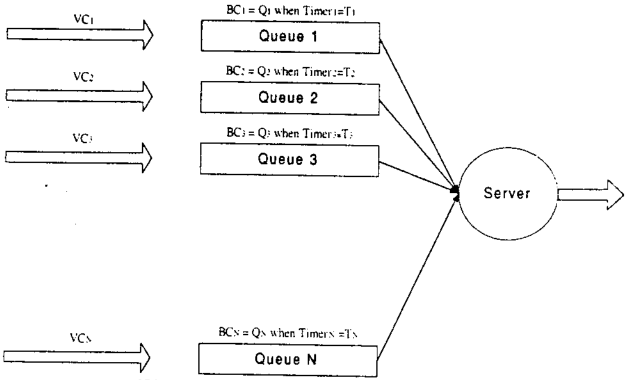

Figure 1 The CBS architecture.

2.2 Call Admission Control (CAC)

In call admission control (CAC), a connection request may specify the source traffic parameters and the QoS requirements. The source parameter used in our scheme is the maximum burst size (MBS). The required QoSs are maximum cell transfer delay (CTD) and sustained cell rate (SCR). According to these parameters, we can calculate the quantum Q (cells) of transmission during a specific period T (cell slots) for each VC queue in the switch.

Assume that N VCs have been admitted. Each VC_j has its time-frame size T_j and service quantum Q_j, $1 \le j \le N$. Suppose that $T_i < T_j$ and VC_i has higher service priority than VC_j for $i < j$. If a new connection VC_i with T_i and Q_i requests to enter the network, CAC must be executed node by node to decide whether the new request can be accepted. The first condition that must be satisfied is as follows:

$$Q_i / T_i + \sum_{k=1}^{N} Q_k / T_k \le 1 \qquad (1)$$

Here Equation (1) means that the total traffic load can not exceed the link capacity. It implies that there is available bandwidth for the new VC. While the second condition is

$$\sum_{\forall j, T_j \le T_i} \left\lceil \frac{T_i}{T_j} \right\rceil \times Q_j \le T_i \text{, or } \sum_{\forall j, T_j \le T_i} \left\lceil \frac{T_i}{T_j} \right\rceil \times Q_j > T_i$$

$$\text{and } \sum_{\forall l, T_l \le T_{i-1}} \left\lceil \frac{T_{i-1}}{T_l} \right\rceil \times Q_l \le T_{i-1} - Q_i \qquad (2)$$

Equation (2) is necessary to guarantee the queueing delay bound of a VC in an intermediate node. Due to the transmission contentions among various priority levels, the smaller the value of T, the higher the queue priority. To guarantee the bounded delay, the maximum number of cell arrivals within time-frame T_i from those connections with time-frame less or equal to T_i must be no greater than T_i.

However, the existed VC may violate the above condition, it is because that there are idle slot times during the period T_{i-1}. Thus, when a new connection request satisfies the above two conditions, the CAC procedure must further decide whether admitting the new connection causes the QoS violation of existed connections or not. The condition to be checked is shown below. If the new connection can be accepted by the CAC procedure we proposed, the end-to-end delay can be guaranteed.

$$\forall k, T_k \ge T_i \ \{$$

$$\sum_{\forall j, T_j \le T_i} \left\lceil \frac{T_i}{T_j} \right\rceil \times Q_j \le T_i \text{ or } \sum_{\forall j, T_j \le T_i} \left\lceil \frac{T_i}{T_j} \right\rceil \times Q_j > T_i \text{ and}$$

$$\sum_{\forall l, T_l \le T_{i-1}} \left\lceil \frac{T_{i-1}}{T_l} \right\rceil \times Q_l \le T_{i-1} - Q_i \ \} \qquad (3)$$

2.3 Service Discipline

First of all, since the shorter the T is, the higher the priority of queue will be. The service discipline is shown in Figure 2. The algorithm examines those nonempty queues from higher priority to lower priority and check whether the variable BC of a queue is greater than zero. If both conditions are satisfied, the queue is allowed to send at least one cell. The value of BC is the number of cells allowed to be sent. No sooner had the first cell satisfying the condition been selected, the examination stopped and the Timer was advanced, as shown in Figure 3.

On the other hand, if no cell can be found according to the rule, there may be two reasons: the variable BC is greater than zero but the queue is empty, or the queue is backlogged but the variable BC equals to zero. Since CBS is a work-conserving discipline, the server can not be idle if any cell exists in the buffer. The solution here is to advance the Timer. Only Timer of backlogged queues will be advanced until the earliest event happens, that is, the variable of BC is reset. Thus, queues with newly updated credit are allowed to send. The algorithm is illustrated in Figure 3. Since only backlogged queues can share the excessive bandwidth, so only these queues can advance their Timer.

Figure 4 is an example that explains the mechanism of advancing Timer. There are three sessions in the system. At time t, session 1 has the service credit but its queue is empty. While session 2 and 3 do not have any service credit, but their queues are backlogged. In such case, the procedure AdvancTimer() must be executed. As a result, Timer of session 1 does not act while Timer of session 2 and 3 is incremented by MinDifference, which is the minimum time duration from time t to the earliest event that the variable BC is reset to Q.

```
Procedure SelectCell()
Begin
   While (There is at least one backlogged queue)
   { // The order of priorities: $Queue_1$ > $Queue_2$ > ... > $Queue_N$
     For (i=1,2,3...N)
            { // There exists data ready to send
            if (Queue_i is backlogged)
            {       // There are credits to send data
                    if ( $BC_i > 0$ )
                    {       Send one cell;
                            $BC_i = BC_i - 1$;
                            Stop;
                    }
            }
        }
}
AdvanceTimer;
End;
```

Figure 2 The algorithm of cell scheduling.

```
AdvanceTimer()
Var
    Difference;     // if  (($Timer_i$ + Difference) == $T_i$)
    MinDifference;
    MaxValue;       // Initialized value of MinDifference
Begin
     MinDifference = MaxValue;
   For (i=1,2,3...N)
    {       if ( $Queue_i$ is backlogged)
            {          Difference = $T_i$ - $Timer_i$ ;
                       if ( Difference < MinDifference)
                              MinDifference = Difference;
            }
    }
    // Only the backlogged queue can advance Timer
    For (i=1,2,3...N)
    {       if ( $Queue_i$ is backlogged)
            {       $Timer_i$ = $Timer_i$ + MinDifference;
                    if  ( $Timer_i$ == $T_i$)
                    {       $BC_i = Q_i$;
                            $Timer_i$ = 0;
                    }
            }
    }
End;
```

Figure 3 The algorithm of advancing Timer.

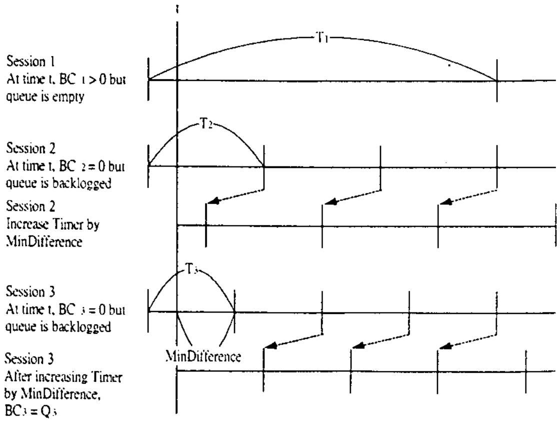

Figure 4 An example of advancing Timer.

3. PERFORMANCE ANALYSIS OF CBS

We now investigate the maximum buffer size and the guaranteed end-to-end delay each VC needs in each node. The following definitions must be provided first before we proceed the derivation.

Assume that servers are non-cut-through devices, let $A_i(\tau,t)$ denote the arrivals from session *i* during the interval $(\tau,t]$, and $W_i(\tau,t)$ the amount of service received by session *i* during the same interval. Let T_i^n and Q_i^n denote the n_{th} time frame and n_{th} service credit respectively for session *i*.

Definition 1: A busy period for session *i* is a minimum interval of time $(s,f]$. Where s is the time at which the first cell is received after an idle period or after the beginning of session *i*, and *f* is a time frame boundary when the buffer of session *i* is empty and no cell arrival occurs at that time.

Definition 2: An idle period for session *i* is a time interval that interleaves two busy periods, there is no cell transmission during an idle period.

Definition 3: A dense part of a busy period occurs either at the beginning or at the end of the previous part of a busy period for session *i*, it is a maximal interval of time $(s_d,f_d]$, $f_d\in nT_i$ and $n\in N$, such that for any $t\in(s_d,f_d)$, $t\in nT_i$ and $n\in N$,

$$A_i(s_d,t)>\left\lceil\frac{t-s_d}{T_i}\right\rceil\times Q_i\cdot \qquad (4)$$

Definition 4: A sparse part of a busy period occurs at either the beginning of the busy period or the end of the dense part within the busy period for session *i*. A sparse part is a maximal interval of time $(s_s,f_s]$, $f_s\in nT_i$ and $n\in N$, such that for any $t\in(s_s,f_s]$, $t\in nT_i$ and $n\in N$,

$$A_i(s_s,t)\le\left\lceil\frac{t-s_s}{T_i}\right\rceil\times Q_i\cdot \qquad (5)$$

Assume that a dense part within a busy period for session *i* starts at time s_d. Then the total service provided to session *i* during the interval $(s_d,t]$, $t\in nT_i$ and $n\in N$, satisfies the inequality

$$W_i(s_d,t)\ge\left\lceil\frac{t-s_d-T_i}{T_i}\right\rceil\times Q_i \qquad (6)$$

Now we present a delay bound for CBS. First we consider the behavior of a session in a single node, and subsequently we extend the analysis to a network. In both cases, we assume that the input traffic of the session is leaky-bucket smoothed and the allocated rate is no less than the average arrival rate. That is, if session *i* is under observation, its arrivals to the network during the interval $(\tau,t]$, $t\in nT_i$ and $n\in N$, satisfy the inequality

$$A_i(\tau,t)\le\sigma_i+\rho_i(t-\tau)\le\sigma_i+\left\lceil\frac{t-\tau}{T_i}\right\rceil\times Q_i \qquad (7)$$

where σ_i and ρ_i denote the burstiness and average cell rate, respectively.

3.1 Analysis for Single Node Case

Stiliadis and Varma have proposed a general mode, called Latency-Rate server, to analyze the scheduling algorithm [9]. We modify their model to analyze CBS as follows.

Let $B_i(t)$ represent the amount of session *i* traffic queued in the server at time *t*, that is

$$B_i(t)=A_i(0,t)-W_i(0,t) \qquad (8)$$

Lemma 1: If $B_i(t)$ is the backlog of session *i* at time *t*, $t\in nT_i$ and $n\in N$, then

$$B_i(t)\le\sigma_i+Q_i \qquad (9)$$

Proof:

Intuitively, the worst case occurs during the dense part within a busy period, it is the only case we will prove.

Assume that a dense part covered the time interval (ds,df), $ds\in T_i^n$ and time τ is the end of T_i^n. After time τ, only those cells arrived during last dense part will be served. Cells belonging to the previous part will complete the service by time τ.

From (6), (7), and (8)

$$\begin{aligned}B_i(t)&=A_i(cs,t)-W_i(cs,t)\\&\le\sigma_i+\left\lceil\frac{t-cs}{T_i}\right\rceil\times Q_i-\left\lceil\frac{t-cs-T_i}{T_i}\right\rceil\times Q_i\\&=\sigma_i+Q_i\end{aligned} \qquad (10)$$

Theorem 1: If $B_i(t)$ is the backlog of session *i* at time *t*, then

$$B_i(t)\le\sigma_i+2Q_i-\frac{Q_i^2}{T_i} \qquad (11)$$

Proof:

Lemma 1 shows that the maximum backlogged cells queued in the server is (σ_i+Q_i) at the end of the service period within the dense part.

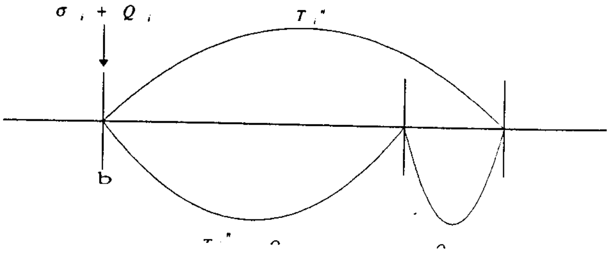

Figure 6 Analysis of backlogged traffic.

As shown in Figure 6, assume that period T_i^n starts at time *b*, at which the maximum backlogged cells queued in the server is σ_i+Q_i. The worst case occurs when there is no cells being served until time $(b+T_i^n-Q_i)$. Thus,

$$\begin{aligned}B_i(t)&\le\sigma_i+Q_i+\rho_i(T_i-Q_i)\\&=\sigma_i+Q_i+\frac{Q_i}{T_i}(T_i-Q_i)\\&=\sigma_i+2Q_i-\frac{Q_i^2}{T_i}\end{aligned} \qquad (12)$$

***Theorem 2*:** If D_i is the delay of a cell in session i, then

$$D_i \le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + 2T_i \tag{13}$$

***Proof*:**

Intuitively, the worst case occurs within the dense part during a busy period, which is the only case we will prove. Assume that a dense part covers the time interval *(ds,df)*, $ds \in T_i^n$ and time τ is the end of T_i^n. After time τ, only those cells arrived during the past dense part will be served. Cells that belong to the previous part will complete the service by time τ.

As shown in Figure 7, assume that the maximum delay D_i including $D1$ and $D2$ is caused by a cell that arrived at time $t^* \in T_i^m$. Let time τ be the end of T_i^m, $D1$ be the period from t to τ and $D2$, from τ to t^*+D_i. This means that the cell was served at time t^*+D_i. Hence, the amount of service offered to the session up to time t^*+D_i is equal to the amount of traffic that arrived to the session until time t^*, therefore

$$W_i(cs,t^*+D_i) = A_i(cs,t^*) \tag{14}$$

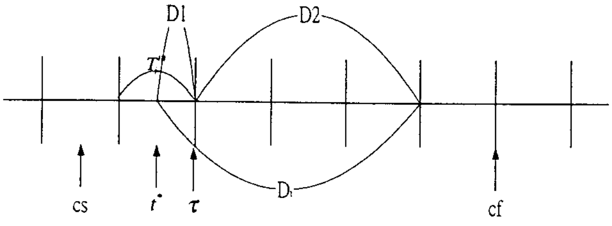

Figure 7 Analysis of delay .

From (6), with the total service provided to session i and the leaky-bucket constraint in (7), we have

$$\begin{aligned} \left\lceil \frac{t^*+D1+D2-cs-T_i}{T_i} \right\rceil \times Q_i &\le \sigma_i + \rho_i(t^*-cs) \\ &\le \sigma_i + \left\lceil \frac{t^*+D1-cs}{T_i} \right\rceil \times Q_i \end{aligned} \tag{15}$$

Simplifying (15), we get

$$D2 \le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + T_i \tag{16}$$

and it is clear that

$$D1 \le T_i \; . \tag{17}$$

From (16) and (17), we obtain

$$D_i = D1 + D2 \le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + 2T_i \tag{18}$$

3.2 Analysis of Multiple Nodes Case

We consider the dense part within a busy period for analyzing the properties of CBS in a network because it is the worst case. First, we will analyze the behavior of a session in two servers, and extend the result to multiple servers. Let $B_i^n(t)$ denote the amount of cells queued in the n_{th} server at time t, and $W_i^n(\tau,t)$ denote the amount of service offered by the n_{th} server during an interval (τ,t). Here the service offered by the network to session i during a time interval means the volume of session-i traffic transmitted by the last server during that time interval.

***Theorem 3*:** If $B_i^k(t)$ is the backlog of session i in the k_{th} node at time t, then

$$B_i^k(t) \le \sigma_i + (k+1)Q_i - \frac{Q_i^2}{T_i} \tag{19}$$

***Proof*:**

Assume that the service period of the second server is synchronized with the first one. Theorem 1 shows that the maximum backlogged cells queued in the first server is $\sigma_i + 2Q_i - Q_i^2/T_i$ at time τ, as shown in Figure 8. If session i is the only active connection in server 1 after time τ, the backlog will be served by server 1 by time $\tau+T_i$. The maximum backlog in server 2 occurs when there is no cells being served by server 2 until time $\tau+T_i$. The additional backlog, $\frac{Q_i}{T_i}\times T_i$, will be added to server 2. Thus,

$$B_i^2(t) \le \sigma_i + 2Q_i - \frac{Q_i^2}{T_i} + Q_i \tag{20}$$

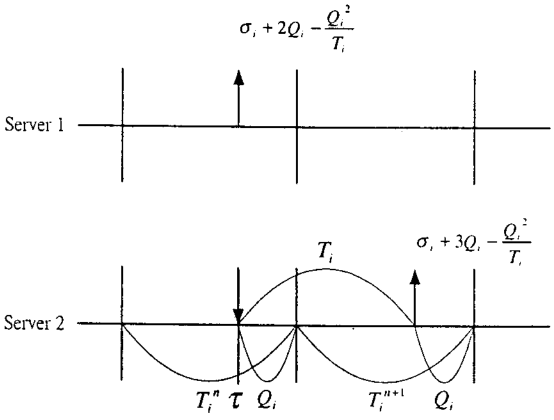

Figure 8 An example of two servers in series.

Equation (20) asserts that if one more server is inserted, one more Q_i will be added. If there are N servers in the network, the maximum backlog for session i in node k can be represented as follows:

$$\begin{aligned} B_i^k(t) &\le \sigma_i + 2Q_i - \frac{Q_i^2}{T_i} + (k-1)Q_i \\ &= \sigma_i + (k+1)Q_i - \frac{Q_i^2}{T_i} \end{aligned} \tag{21}$$

if $\sigma_i + (k+1)Q_i - Q_i^2/T_i$ can be served completely in node *(k-1)* by time $\tau + (k-1)T_i$.

***Theorem 4*:** If D_i is the delay of session i in a network consisting of N nodes in series, then

$$D_i \le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + (N+1)T_i \tag{22}$$

***Proof*:**

As shown in Figure 9, the service period of the second server is synchronized with the first one through an RM cell. Assume that a dense part covers the time interval *(ds1,df1)* in the first server. The traffic of the dense part in the first server leads to the dense part that covers the interval *(ds2,df2)* in the second server. Let $t \in nT_i$ and $n \in N$.

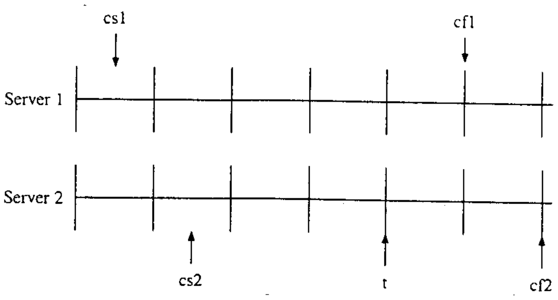

Figure 9 An example of two servers in series.

$$\begin{aligned} W_i(cs1,t) &= W_i^2(cs2,t) \\ &\le \left\lceil \frac{t - cs2 - T_i}{T_i} \right\rceil \times Q_i \qquad (23) \\ &= \left\lceil \frac{t - cs1 - T_i - T_i}{T_i} \right\rceil \times Q_i \end{aligned}$$

Equation (23) asserts that if one more server is added, then one more service latency will be needed. If there are N servers in the network, the total service provided to session *i* can be calculated as

$$W_i(cs1,t) \le \left\lceil \frac{t - cs1 - (N-1)T_i}{T_i} \right\rceil \times Q_i \qquad (24)$$

Denote D_i^n the maximum delay in node *n*. According to Equation (24) and Theorem 1, we can directly conclude that the maximum delay in node N is

$$\begin{aligned} D_i &= D_i^N \\ &\le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + 2T_i + (N-1)T_i \qquad (25) \\ &\le \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + (N+1)T_i \end{aligned}$$

The condition we have discussed is for general case. In fact, if any server whose Timer advances for session *i*, the delay will become even shorter.

3.3 Setting CBS Parameters

The traffic of session *i* is characterized by its maximum burst size (MBS) which is also its maximum frame size. QoS of session *i* is characterized by the maximum cell transfer delay (CTD) and the sustained cell rate (SCR). The worst end-to-end delay calculated in Theorem 4 must be no less than CTD. The inequality is as follows:

$$CTD \ge \left\lceil \frac{\sigma_i}{Q_i} \right\rceil \times T_i + (N+1)T_i \qquad (26)$$

The procedure to set Q and T is as follows:

- $Q = MBS$
- $T = \dfrac{CTD}{\left\lceil \frac{\sigma}{Q} \right\rceil + (N+1)}$, this is obtained from (26).
- If $1 \ge \frac{Q}{T} \ge SCR$ then Q and T obtained from above steps can be accepted. If $\frac{Q}{T} < SCR$, then $T = \frac{Q}{SCR}$ because the sustained cell rate must be satisfied. If none of two can be satisfied, it will be rejected.

As a result, the end-to-end cell delay can be bounded by CTD. If cells of a specific frame is generated one by one, its frame delay can also be bounded by CTD.

4. SIMULATION MODEL AND RESULTS

4.1 Configuration and Assumptions

We have a single node configuration as shown in Figure 10. All four hosts are connected to Switch 1 directly, also they have the same destination, Host 5. The single node configuration is simple, so we can simulate some properties of CBS easily.

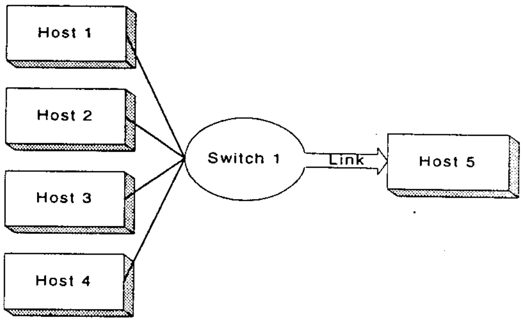

Figure 10 Single node configuration.

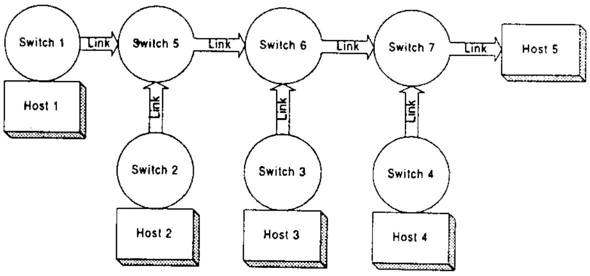

Figure 11 Parking lot configuration.

In order to simulate more realistic network environment, we also use a parking lot configuration, as shown in Figure 11. Assume that the edge device consist of a host and a switch. Host 1, 2, 3 and 4 are attached to Switch 1, 2, 3 and 4 respectively. There are two sessions in each host. Host 1 has session 1 and session 8, Host 2 has session 2 and session 7, Host 3 has session 3 and session 6, and Host 4 has session 4 and session 5. All these sessions have the same destination, Host 5. Since all sessions will pass through Switch 7, the congestion may happen there, and we can find the worst case of simulation result in that switch, call it the bottleneck switch. The links in the configuration have a bandwidth of 150 Mbps, and a propagation delay of 500 μs. We assume that queues are sufficiently large so that we can factor out effects due to the lack of buffer space.

4.2 Traffic Model

(1) Source Model:

The source model is based on a two-state discrete-time Markov chain On-Off model, as shown in Figure 12. Cells are generated whenever the Markov chain enters or returns to the "On" state, termed as busy period. Therefore, cells are generated with probability β when the source was previously in the "Off" state, and with probability $1-\alpha$ when the source was previously in the "On" state. When a source is in "Off" state, termed idle period, it does not generate any cell. In order to simulate the realistic traffic source, both idle and busy periods are assumed to be geometrically distributed.

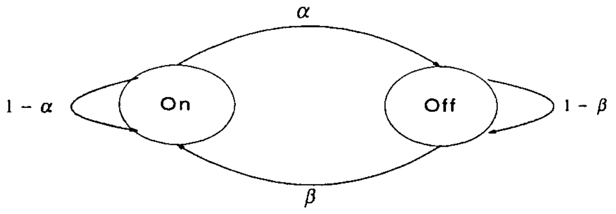

Figure 12 The on-off model.

As shown in Figure 13, we define some parameters to create the On-Off source model. These parameters include mean burst interarrival time (MEBIT), mean burst size (MEBS) and burstiness (b). When the source in "On" state generates cells with inter cell gap (ICG), the behavior of cell generation is in geometric distribution with mean value MEBS.

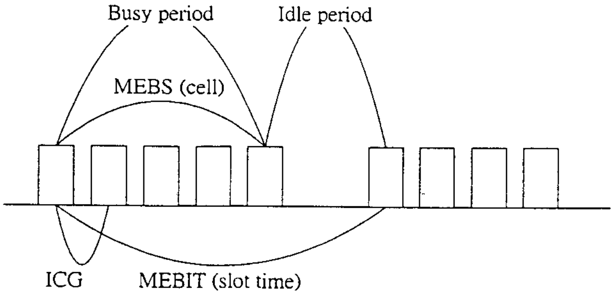

Figure 13 Modified packet train model.

The source behavior of multimedia traffic in the broadband network can be modeled as a superposition of On-Off sources. By properly choosing parameter values, we can model a variety of traffic sources. The traffic parameters of each source are listed in Table 1. Assume that the total traffic load is 0.86.

(2) Leaky Bucket :

Each session has a leaky bucket to shape the traffic. Only the cell with permitting token can enter into the input queue of the switch. The leaky bucket has two parameters, σ and ρ. The parameters of each leaky bucket associated with each session are shown in Table 2.

Session ID	MEBIT (slot time)	MEBS (cell)	Burstiness (b)
Session 1	300	30	100
Session 2	500	100	100
Session 3	700	90	100
Session 4	900	80	100
Session 5	1300	220	100
Session 6	1500	100	100
Session 7	2000	20	100
Session 8	2000	200	100

Table 1 Parameters of source traffic.

Session ID	σ	ρ
Session 1	30	30/300
Session 2	100	100/500
Session 3	90	90/700
Session 4	80	80/900
Session 5	220	220/1300
Session 6	100	100/1500
Session 7	20	20/2000
Session 8	200	200/2000

Table 2 Parameters of the leaky bucket.

(3) Specification of Parameters in CBS :

Table 3 lists the CBS parameters T and Q. These two parameters values for each session can be obtained in the switch with maximum burst size (MBS), maximum cell transfer delay (CTD) and sustained cell rate (SCR).

Session ID	T (slot time)	Q (cell)
Session 1	.300	30
Session 2	500	100
Session 3	700	90
Session 4	900	80
Session 5	1300	220
Session 6	1500	100
Session 7	2000	20
Session 8	2000	200

Table 3 Parameters of CBS.

4.3 Simulation Results

We obtain the cell queueing delay in the first switch under the single node configuration. On the other hand, we also observe the results such as end-to-end delay, buffer occupancy, and characteristics of both flow isolation and fair sharing under the parking lot configuration,

(1) Single Node Configuration:

In the single node configuration, we only analyze the result of the maximum end-to-end delay, because the other results can be derived from the parking lot configuration. As shown in Figure 14, the maximum end-to-end delay is derived from the simulation while CTD means the allowed maximal cell transfer delay. We observe that the maximum end-to-end delay in the first node is much less than CTD. We can conclude that each cell passes through the first node within the bounded delay under our control scheme.

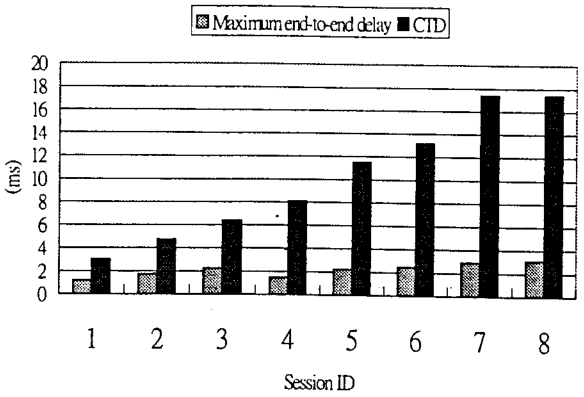

Figure 14 Maximum end-to-end delay in the single node configuration.

(2) Parking Lot Configuration

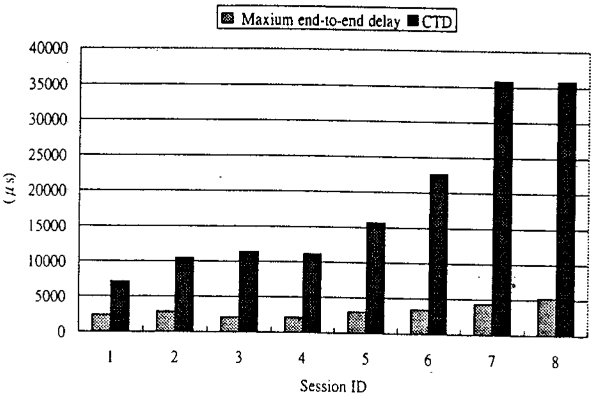

Figure 15 Maximum end-to-end delay.

In Figure 15, the maximum end-to-end delay is derived from the simulation while CTD means the allowed maximum cell transfer delay, we can see that the maximum end-to-end delay of each session is much less than CTD. From this simulation result, we can conclude that each cell can pass through the network within a bounded delay under our control scheme.

As depicted in Figure 16, we find that the buffer occupancy in the bottleneck switch is much smaller than expected. Only session 5 and session 8 have larger backlog because these two sessions have larger values of Q and σ.

In Figure 17, the actual rate is derived from the simulation. We assume that session 3 is a misbehaved flow, which sends data with much higher rate than it has requested. Owing to the control by our scheme, session 3 is only permitted to share the excess bandwidth without affecting other sessions. We can see that all sessions except session 3 still reach the requested rate.

In order to share the excess bandwidth fairly, we use advancing timer to achieve the objective in our scheme. In the simulation, we assume that the VC queue of each session in bottleneck switch is always backlogged .Figure 18 shows the comparison of the requested rate and the actual shared rate. We find that each session has almost the same percentage. It means that the excess bandwidth left by the idle connection is distributed in proportion to their reservation among all backlogged connections.

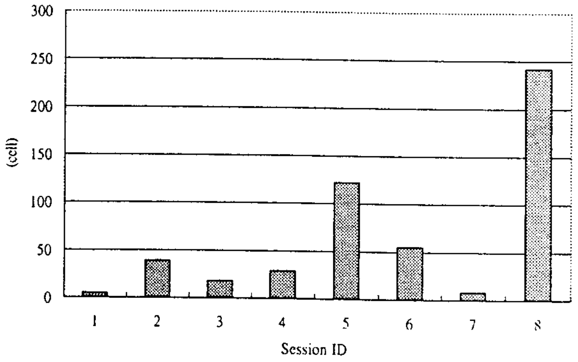

Figure 16 Buffer occupancy in the bottleneck switch.

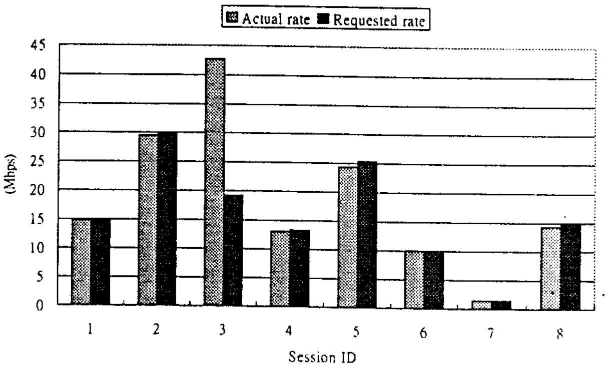

Figure 17 Actual rates acquired with misbehaved session.

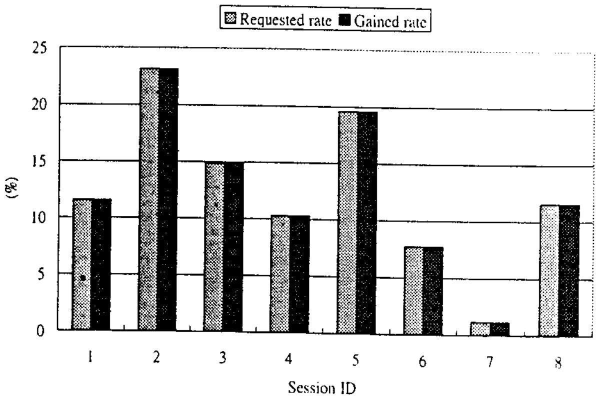

Figure 18 Comparison between requested rate and acquired rate.

Regarding the throughput under the condition that VC queues in the bottleneck switch are all backlogged. We monitor the system every 564 μs period, the longest T. As shown in Figure 19 and Figure 20, we find that the throughput of session 1 and session 4 fluctuates between two specific bounds. Since T of session 4 is larger than that of session 1, the fluctuation of session 4 is larger than that of session 1.

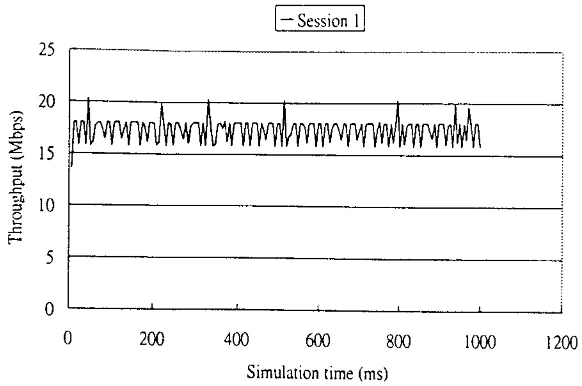

Figure 19 Throughput of session 1 in backlogged condition.

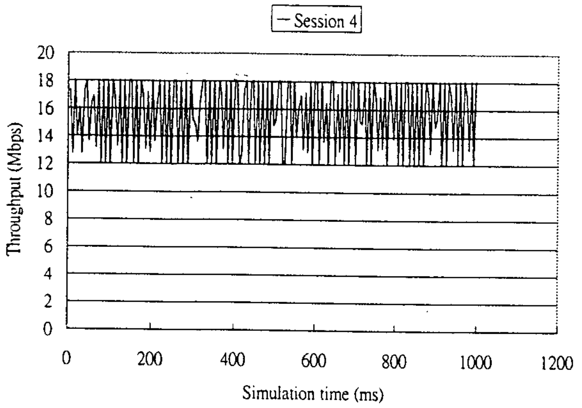

Figure 20 Throughput of session 4 in backlogged condition.

5. CONCLUSION AND FUTURE WORK

In this work, we propose a CBS scheme to schedule different traffic with various burst characteristics and delay requirements. We define T and Q based on MBS, CTD and SCR to achieve the real-time service requirements. The benefit of our scheme is two folds, the scheduler can support real-time services, and the implementation is simple.

Some features of our system are summarized as follows: First, according to the traffic characteristic and the guaranteed QoS, the parameters of CBS, T and Q, can be defined. Second, CAC is used to assure that the service of the new connection can be provided and the existed connections will not be affected by the new one. The scheme decides the service order based on the value of T. As to the quantum of service, T and Q are all needed. Therefore, the scheme, accompanied with leaky bucket has the properties of flow isolation and bounded end-to-end delay. Third, our scheme is work-conserving, so the utilization is higher than the non-work-conserving scheme. Fourth, we use the advancing timer that allows each connection to share the excessive bandwidth fairly based on its requested rate, so the scheme is good for VBR traffic. Since the traffic with burst characteristic and service discipline is frame-based, so the fairness can not be defined for any period but longest T. During the longest period, the throughput is regular and the properties of fairness is satisfactory.

In future work, traffic models that can provide a more realistic representation of ATM, such as MMPP and Regression Models [10], would be considered. We analyze the buffer occupancy in the scheme under the per-VC queuing condition. If an efficient buffer management is provided, the buffer size can be downsized significantly. Furthermore, in order to control the traffic more effectively, it is a challenge to find a way for deciding the CBS parameters precisely.

REFERENCES

[1] M. Shreedhar and George Varghese, "Efficient Fair Queuing Using Deficit Round Robin," IEEE/ACM Transactions on Networking, Vol. 4, No. 3, June 1996, pp.375-85.

[2] Lixia Zhang, "VirtualClock: A New Traffic Control Algorithm for Packet Switching Networks," ACM Trans. Comp. Sys. Vol. 9, May 1991, pp.101-24.

[3] Abhay K. Parekh and Robert G Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case," IEEE/ACM Transactions on Networking, VoL. 1, No. 3, June 1993, pp. 344-57.

[4] S. Jamaloddin Golestani, "A Self-Clock Fair Queueing Scheme for Broad Applications," Proc. IEEE INFOCOM 1994, April 1994, pp.636-46.

[5] J. C. R. Bennett and H. Zhang, "WF2Q: Worst-Case Fair Weighted Fair Queuing," Proc. IEEE INFOCOM '96, March 1996, pp.120-28.

[6] Hui Zang, "Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks," Proceedings of The IEEE, Vol. 83, No. 10, October 1995, pp.1374-96.

[7] Anujan Varma and Dimitrios Stiliadis, "Hardware Implementation of Fair Queuing Algorithms for Asynchronous Transfer Mode Networks," IEEE Communications Magazine, December 1997, pp.54-68.

[8] Y. C. Chen, C. T. Chan and S. C. Hu, "Design and Analysis of Delay Control for CBR Services in ATM networks," To appear on Computer Communications .

[9] Dimitrios Stiliadis and Anujan Varma, "Latency-Rate Servers: A General Model for Analysis of Traffic Scheduling Algorithms," Tech. Rep. UCSC-CRL-95-38, July 1995 , http: //www.cse.ucsc.edu/research/ hsnlab/publications/.

[10] Abdelnaser Adas, " Traffic Model in Broadband Networks," IEEE Communications Magazine, July 1997, pp.82-89.

[11] Raj Jain and Shawn A. Routehier, "Packet Trains-Measurements and a New Model for Computer Network Traffic," IEEE Journal on Selected Areas in Communication, VOL. SAC-4, NO. 6, September 1986, pp. 986-95.

[12] Simon S. Lam and Geoffrey G. Xie, "Burst scheduling: Architecture and algorithm for switching packet video," INFOCOM '95, April 1995, pp.940-95.