ELSEVIER

Information Sciences 141 (2002) 61–79

INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL

www.elsevier.com/locate/ins

Schedulable region for VBR media transmission with optimal resource allocation and utilization ☆

Ray-I Chang *, Meng-Chang Chen, Jan-Ming Ho, Ming-Tat Ko

Department of Information Management, National Central University, Chung-Li, and Academia Sinica, Institute of Information Science, Nankang, Taipei 115, Taiwan, ROC

Received 17 November 1999; received in revised form 23 December 2000; accepted 8 May 2001

Abstract

Media data are variable-bit-rate (VBR) in nature due to the coding and compression technologies applied. As VBR streams are complicated for network management, different approaches were proposed to shape the VBR stream as a transmission schedule with smoothed traffic burst. In this paper, instead of giving a fixed schedule result, a novel traffic shaping scheme is proposed to decide a schedulable region for all optimal transmission schedules that provides the minimal allocation and maximal utilization of system resources (such as network bandwidth, initial delay and client buffer). Experiments have shown that our obtained shaping results show dramatic improvements than that of conventional approaches in both the client buffer size and the network idle rate achieved. Based on the schedulable region provided, the ready time and deadline for each media packet can be precisely specified to support real-time network scheduling and error control. It allows users to determine their own optimal schedules under various quality-of-service (QoS) requirements and resource constraints.

Keywords: Traffic engineering; Traffic shaping/smoothing; Multimedia; Real-time networking

☆ This work was partially supported by NSC under grants NSC88-2213-E-001-011 and NSC88-2213-E-001-012.

* Corresponding author. Tel.: +886-2-2788-3799; fax: +886-2-2782-4814.
E-mail address: william@iis.sinica.edu.tw (R.-I. Chang).

0020-0255/02/$ - see front matter
PII: S0020-0255(01)00191-8

1. Introduction

Recently, multimedia applications such as digital library, home shopping, distance learning, video-on-demand (VOD), and video-conferencing (VC) have attracted great attentions. In these applications, media data such as audio and video are transmitted from server to clients via network according to some transmission schedules. Different from the conventional data streams, end-to-end quality-of-service (QoS) is necessary for media transmission to provide jitter-free playback. As network resources are allocated exclusively in fixed-size chunks to serve different data streams, it is simple to support constant-bit-rate (CBR) transmission. Grossglauser and Keshav [12] have investigated the performance of CBR traffic in a large-scale network with many connections and switches. They concluded that the network queuing delay for CBR transmission is less than one cell time per switch even under heavy loading. However, media streams are notably variable-bit-rate (VBR) in nature due to the coding and compression technologies applied [11,12]. For example, in an MPEG-1 movie, the average frame size is usually less than 25% of its maximal frame size. The conventional network service model that allocates a CBR channel to transmit the VBR stream by stream's peak data rate would be a waste of bandwidth. In past years, different traffic shaping approaches were proposed to reduce the traffic burst. Instead of giving a fixed transmission schedule, we try to decide the upper bound and the lower bound for all transmission schedules that provides the optimal resource allocation and utilization in this paper. It is called the schedulable region of optimal transmission schedules (or *schedulable region*, for short). Based on the schedulable region provided, the ready time and deadline for each media packet can be precisely specified to support real-time network scheduling and error control. It allows users to determine their own optimal schedules under various QoS requirements and resource constraints.

In a multimedia system, we generally measure the performance of a transmission schedule by the following four indices: peak bandwidth, network utilization, initial delay time, and client buffer size.

- *Peak bandwidth* is the maximum network bandwidth allocated for media transmission. A user request is admitted if the peak bandwidth required is smaller than the available bandwidth of the current network.
- *Network utilization* is the ratio of the total bandwidth consumed to the total bandwidth allocated. Generally, the higher the network utilization means more users can be served at the same time.
- *Initial delay* is the length of time interval from the time that the client sends the media request to the time that the client starts playing the received data. It is an important QoS indicator for users.
- *Client buffer* acts as a reservoir to regulate the difference between the transmission rate and the playback rate. It is an important resource for users to prevent playback jitters, i.e. buffer overflow and underflow.

While serving a VBR media stream, a good transmission schedule is designed to minimize the peak bandwidth, initial delay and buffer size required to keep the network utilization as large as possible. Moreover, end-to-end QoS of media transmission needs to be guaranteed for supporting jitter-free playback [6,16,17]. Recently, different approaches are proposed to shape the traffic burst for high network utilization, smaller buffer size, and short initial delay. In [19], the constant-rate transmission and transport (CRTT) method was presented to transmit VBR media data by a constant bandwidth. By given the available transmission bandwidth and initial delay, CRTT minimized the required buffer size by the dynamic programming technique. Although the admission control and transmission schedule were simple, CRTT had the drawback of requiring large buffer and delay. To reduce the required buffer, a piecewise CRTT (PCRTT) method [19] was introduced to evenly divide the media stream into sub-streams and applied CRTT to each sub-stream. Based on the similar idea, the renegotiated CBR (RCBR) method [13] was proposed to use the average data rates in different sub-streams. Given initial delay and client buffer, the minimum changes bandwidth allocation (MCBA) [9] and critical bandwidth allocation (CBA) [10] methods were proposed to minimize the number of bandwidth changes and the peak bandwidth required, respectively. In [20], the minimum variability bandwidth allocation (MVBA) method was proposed to minimize the bandwidth variation for media transmission by the shortest-path algorithm [18].

Note that, although previous traffic shaping methods had reduced some problem parameters in media transmission, they did not achieve the optimized schedule results that minimize the initial delay, the client buffer and the bandwidth utilization at the same time. For example, in [20], the allocated initial delay and the bandwidth utilization were not optimized as discussed in [3,4,26]. In this paper, a novel traffic shaping approach is presented to optimize both the resource allocation and utilization for VBR media transmission. Instead of giving a fixed result, our approach provides the schedulable region for all optimal transmission schedules. We have proved that all the schedule results presented in this given region have the minimal initial delay and client buffer for the network channel applied. The remainder of this paper is illustrated as follows. We introduce the VBR media transmission problem in Section 2. In Section 3, the proposed algorithm is proposed to identify the schedulable region for all optimal transmission schedules. Experiments are shown in Section 4. Section 5 shows the conclusion remarks.

2. VBR media transmission

In this paper, we consider the end-to-end transmission of a pre-stored VBR media stream. While a user request is presented, media data are first retrieved

from the storage sub-system by following the disk retrieval scheduler [2,7,8,24]. The network transmission scheduler then transmits the retrieved data from server to client at the proper time. On the client side, incoming data are temporarily stored in the client buffer and consumed frame-by-frame periodically by the playback scheduler. If a frame arrives late or is incomplete at its playback time, unpleasant jittery effects would be perceived by the audience. To avoid jittery playback, the transmission schedule must always be ahead of its related playback schedule so that the client buffer would not be underflow. On the other hand, the transmission schedule must avoid sending more data to the client buffer than the total data that the buffer can store. Otherwise, the overflow condition is occurred and the client results in loss of data. It will require an extra bandwidth for retransmission. While serving a media stream, a good transmission schedule is designed to minimize resource allocation and maximize resource utilization without playback jitter. In this paper, to concentrate on the formalization of the media transmission problem, the disk retrieval scheduler is assumed to always retrieve sufficient data before they request by the network transmission scheduler [24,27]. More detail descriptions for the design and implementation of a multimedia disk retrieval scheduler are shown in [28].

A media stream V can be represented by a set of frames $\{f_0, f_1, \ldots, f_{n-1}\}$ where n is the number of frames and f_i is the ith frame. We assume that the media stream is played at $t = 0$ and the time to play the ith frame is $i \times T_f$ where T_f is the playback time interval between adjacent frames. (For example, $T_f = 1/30$ s in a MPEG video stream [15].) In this paper, without loss of generality, we let $T_f = 1$ (unit time). The ith accumulative frames size of V is $F_i = F_{i-1} + f_i$ where the initial value $F_k = 0$ for $k < 0$. The media stream size is the total frame size $F_k = |V|$ for $k > n - 2$. As the client plays the media stream frame-by-frame periodically (f_i is consumed at the ith frame time), the playback schedule can be denoted by its accumulative playback function $F(\cdot)$ in the following:

$$F(t) = F_x = \sum_{i=0}^{x} f_i \quad \forall x \leqslant t < (x+1). \tag{1}$$

Note that $F(\cdot)$ is a nonnegative stair function with jumps at time t for $t = 0, 1, \ldots, n-1$. The low corner and the up corner at time t are $F(t)^- = F(t-1)$ and $F(t)^+ = F(t)$, respectively.

Based on the same idea, we define the transmission schedule $G(\cdot)$ as a function that cumulates the amount of media data received at the client. Assume that the media data are transmitted by rate $r(t)$ at time t, the transmission schedule is defined as the integration function of $r(t)$ as follows:

$$G(t) = \int_{x=0}^{t} r(x)\,\mathrm{d}x. \tag{2}$$

Note that this function is continuous and monotonically non-decreasing. The peak bandwidth of the network channel allocated for media transmission is

$$\text{Peak bandwidth} : r = \mathbf{max}\{r(t) \,|\, \forall t\}. \tag{3}$$

Let $t_s = \mathbf{min}\{t \,|\, \forall r(t) > 0\}$ and $t_e = \mathbf{max}\{t \,|\, \forall r(t) > 0\}$ be the start time and the end time of the transmission schedule $G(\cdot)$, respectively. The value $t_c = t_e - t_s$ is the connection time of the network channel allocated. We can compute the network utilization of the allocated channel as follows:

$$\begin{aligned} &\text{Network utilization} : u = |V|/(r \times t_c)100\%, \\ &\text{Network idle rate} = 100\% - u. \end{aligned} \tag{4}$$

According to the definition, $G(t)$ is the amount of data sent by the server up to time t. Assume that there is no transmission error and the network delay is zero. $G(t)$ also represents the amount of data received by the client up to time t. If the client starts the playback of the media stream at time 0, the value of the initial delay is shown as follows:

$$\text{Initial delay} : d = -t_s. \tag{5}$$

As the media data must be transmitted before be received and played, the start time $t_s < 0$ [12,25]. Note that, at the client, $G(t)$ and $F(t)$ represent the cumulated data received and consumed up to time t respectively. The buffer occupancy $b(t) = G(t) - F(t)$ is the amount of transmitted data temporarily stored in the client buffer at time t. The minimal client buffer size required and its utilization for media transmission and playback can be computed as follows:

$$\begin{aligned} &\text{Client buffer} : b = \mathbf{max}\{b(t) \,|\, \forall t\}, \\ &\text{Buffer utilization} = \left(\sum_{t=-d}^{n-1} b(t) \Big/ \sum_{t=-d}^{n-1} b \right) 100\%. \end{aligned} \tag{6}$$

Obviously, b is no smaller than the maximum frame size, and is no larger than the stream size. An example to illustrate the cumulative playback function, the cumulative playback function, the initial delay and the buffer size is shown in Fig. 1.

In this paper, a transmission schedule is said to be feasible if it can provide the jitter-free playback. By definition, a jitter-free transmission schedule demands a complete media frame before its playback. The cumulative transmission function $G(t)$ must not to be larger than the cumulative playback function $F(t)$. Besides, the buffer occupancy must not be smaller than the specified buffer size. Define $H(t)$ as the upper bound of $G(t)$ and, in this paper, $H(t) = \mathbf{min}\{|V|, F(t)^- + b\}$. A feasible transmission schedule satisfies the following conditions:

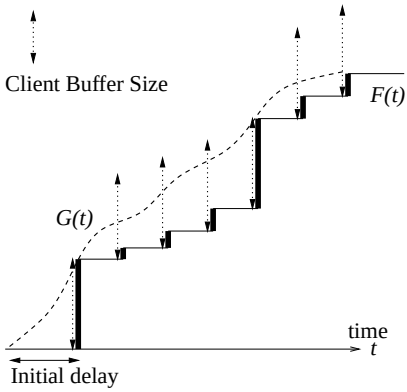


Fig. 1. An example to illustrate the relations among the cumulative transmission function, the cumulative playback function, the initial delay and the client buffer size.

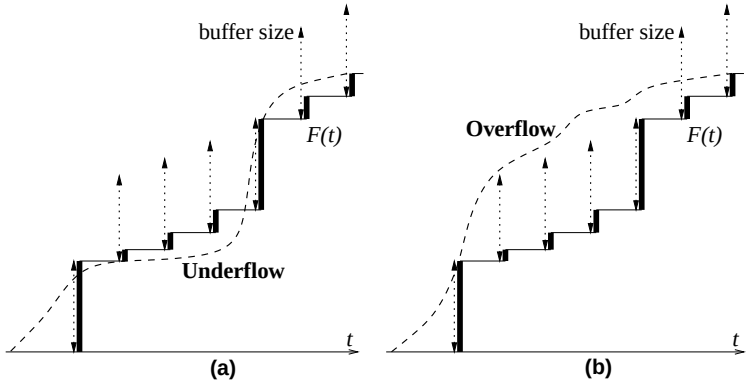


Fig. 2. (a) The underflow (or starvation) condition and (b) the overflow condition for the cumulative transmission function with a bounded client buffer.

$$F(t) \leqslant G(t) \leqslant H(t). \tag{7}$$

Fig. 2(a) and (b) shows the underflow condition and the overflow condition of media transmission, respectively, with a bounded buffer at the client.

3. Schedulable region of optimal transmission schedules

When designing a transmission schedule, two important resources are considered: network bandwidth and client buffer. In this paper, a transmission schedule is said to be optimal if it allocates the minimal resources (both net-

work bandwidth and client buffer) and has the maximal resource utilization. From the definitions shown in Eqs. (4) and (5), the network utilization is maximized only if the initial delay is minimal. An optimal transmission schedule must decide the minimal initial delay for media playback. In this section, we introduce Algorithm-L to decide the minimal resource allocated for media transmission. Then, Algorithm-A is proposed to maximize the resource utilization. Based on these two algorithms, the schedulable region for all the optimal transmission schedules is given to assign the ready time and the deadline to each packet. Finally, a smoothed optimal transmission schedule is presented.

3.1. Minimal resource allocation

Given a media stream V, we first introduce Algorithm-L to decide the amounts of resource required for media transmission. Let the available network bandwidth be r. The transmission schedule $L(\cdot)$ obtained is shown as follows:

$$\begin{aligned} &L(n-1) = |V| = F(n-1), \\ &L(t) = \mathbf{max}\{F(t), L(t+1) - r\} \quad \forall 0 \leqslant t < (n-1). \end{aligned} \tag{8}$$

An example to illustrate the computation of Algorithm-L is shown in Fig. 3. Note that the media data are transmitted and stored into the client buffer as late as possible. Therefore, at any time t, Algorithm-L decides the minimal buffer occupancy for guaranteeing jitter-free playback. It achieves the minimal client buffer size $b = \mathbf{max}\{L(t) - F(t) \,|\, \forall t\}$ and the minimum initial delay $d = L(0)/r$ for the available network bandwidth r. Besides, given any transmission schedule $G(\cdot)$ with the peak transmission rate r, we have $L(t) \leqslant G(t)$ for

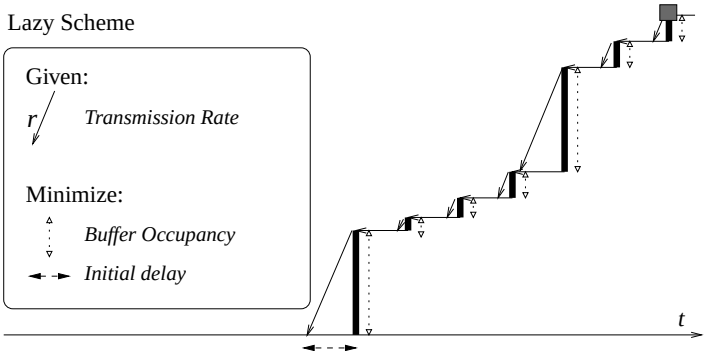


Fig. 3. An example to illustrate the operations of Algorithm-L that minimizes the resource allocation.

any time t. The achieved result $L(\cdot)$ is called the minimal r-bounded transmission schedule.

Lemma 1. *$L(\cdot)$ is the minimal r-bounded transmission schedule.*

Proof. Suppose the contrary. Let $G(\cdot)$ be a r-bounded transmission schedule, for which there exists a time index x such that $L(x) > G(x)$. Let y be the smallest time index that satisfies $x < y$ and $L(y) = F(y) = L(x) + r \times (y - x)$. From the definition of $L(\cdot)$ shown in Eq. (8), $L(y) = F(y)$ if $L(y + 1) - r \leqslant F(y)$. The value y is existed (at least, we have the initial value $L(n - 1) = F(n - 1)$). As $G(\cdot)$ is r-bounded, the relation $G(y) \leqslant G(x) + r \times (y - x) < L(x) + r \times (y - x)$ is true [1]. That implies $G(y) < F(y)$. The underflow condition of the client buffer is occurred and $G(\cdot)$ is not a feasible transmission schedule. It is a contradiction and the lemma is proved. □

Lemma 2. *$L(\cdot)$ has the minimal buffer size and initial delay for all r-bounded transmission schedules.*

Proof. Since $L(\cdot)$ is the minimal r-bounded transmission schedule, it sends the minimal amount of data to the client buffer for guaranteeing jitter-free playback. At any time t, we have $L(t) \leqslant G(t)$ where $G(\cdot)$ is any other r-bounded transmission schedule. As $L(0) \leqslant G(0)$, the initial delay is $L(0)/r \leqslant G(0)/r$. $L(\cdot)$ has the minimal initial delay for all r-bounded transmission schedules. Moreover, their buffer occupancies have the relation $L(t) - F(t) \leqslant G(t) - F(t)$. It implies that $L(\cdot)$ has the minimal buffer size (the required buffer size $\mathbf{max}\{L(t) - F(t) \,|\, \forall t\} \leqslant \mathbf{max}\{G(t) - F(t) \,|\, \forall t\}$). □

3.2. Maximal resource utilization

In this paper, Algorithm-L is introduced to decide the minimal amounts of system resource required for guaranteeing jitter-free playback. Given the system resources allocated, Algorithm-A is then introduced to maximize the utilization of bounded resources. The obtained transmission schedule $A(\cdot)$ is shown as follows:

$$A(t) = \mathbf{min}\{H(t), A(t-1) + r\} \quad \forall(-D) < t \leqslant (n-1). \tag{9}$$

The initial value is $A(-D) = 0$ where $d \leqslant D$. Without loss of generality, we let the value of D be the minimal initial delay d decided by Algorithm-L. It implies $A(0) = L(0)$. An example to illustrate the computation of Algorithm-A is shown in Fig. 4. Note that, as the media data are transmitted to the client as early as possible, this algorithm can maximize the utilization of given resources – the network bandwidth r and the client buffer b. Furthermore, the obtained transmission schedule is robust against network errors [21]. For any other

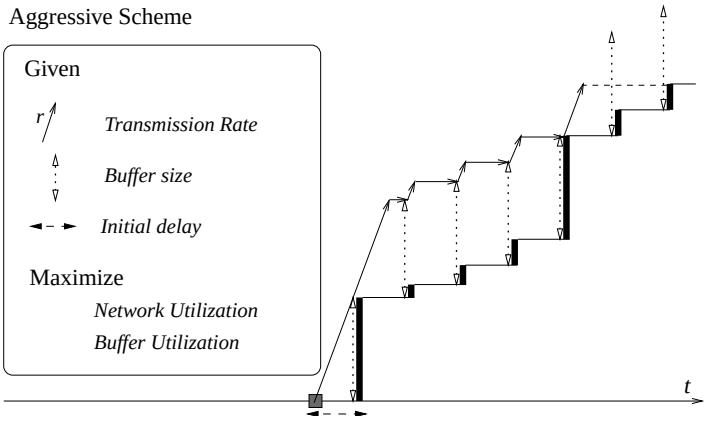


Fig. 4. An example to illustrate the operations of Algorithm-A that maximizes the resource utilization.

transmission schedule $G(\cdot)$ that has the same transmission bandwidth r and client buffer b, we can prove $G(t) \leqslant A(t)$ for any time t. The obtained result $A(\cdot)$ is called the maximal (r, b)-bounded transmission schedule.

Lemma 3. *$A(\cdot)$ is the maximal (r, b)-bounded transmission schedule.*

Proof. Suppose the contrary. Let $G(\cdot)$ be a (r, b)-bounded transmission schedule, for which there exists a time index x such that $G(x) > A(x)$. Let y be the largest time index that satisfies $y < x$ and $A(y) = H(y) = A(x) - r \times (x - y)$. From the definition of $A(\cdot)$ shown in Eq. (9), $A(y) = H(y)$ if $H(y) \leqslant A(y-1) + r$. The value y is not existed only when the peak bandwidth r is fully utilized in $A(\cdot)$. It implies $G(\cdot) \leqslant A(\cdot)$ (they are (r, b)-bounded). Assume that y is existed. As $G(\cdot)$ is a r-bounded, the relation $A(x) - r \times (x - y) < G(x) - r \times (x - y) \leqslant G(y)$ is true. That implies $H(y) < G(y)$. The overflow condition of the client buffer is occurred and $G(\cdot)$ is not a feasible transmission schedule. It is a contradiction and the lemma is proved. □

Lemma 4. *$A(\cdot)$ has the maximal utilization in buffer size and network bandwidth for all (r, b)-bounded transmission schedules.*

Proof. Since $A(\cdot)$ is the maximal (r, b)-bounded transmission schedule, the maximal amount of data is sent to the client buffer at any time t. Thus, given any (r, b)-bounded transmission schedule $G(\cdot)$, we have $G(t) \leqslant A(t)$ and the buffer occupancy $G(t) - F(t) \leqslant A(t) - F(t)$ at any time t. From the definition shown in Eq. (6), $A(\cdot)$ has the maximal utilization in buffer size. Based on the same idea, we have $G(t_e) \leqslant |V| = A(t_e)$ where t_e is the end time of schedule $A(\cdot)$.

Let t'_e be the end time of schedule $G(\cdot)$. We have $G(t_e) \leqslant |V| = G(t'_e)$. It implies $t_e \leqslant t'_e$ (t_e is the minimal value of end time for all (r,b)-bounded transmission schedules). From the definition shown in Eq. (4), $A(\cdot)$ has maximized the utilization of network bandwidth. □

3.3. Schedulable region for optimal resource allocation and utilization

We have shown that, given any (r,b)-bounded transmission schedule $G(\cdot)$, $G(t) \leqslant A(t)$ for all t. Let (r,b) be the minimal transmission bandwidth r and client buffer b obtained by Algorithm-L. The maximal (r,b)-bounded transmission schedule $A(\cdot)$ can determine the upper bound of the transmission schedules that optimize both the resource allocation and utilization. It has the minimal end time t_e for all (r,b)-bounded transmission schedules. In this section, by applying Algorithm-L and the minimal end time t_e, we determine the minimal (r,b)-bounded transmission schedule $R(\cdot)$ as follows:

$$R(t) = \mathbf{max}\{F(t), R(t+1) - r\} \quad \forall 0 \leqslant t < (n-1). \tag{10}$$

The initial value $R(t_e) = |V| = A(t_e)$. Given any transmission schedule $G(\cdot)$ that have the optimal resource allocation and utilization, we can prove that $R(t) \leqslant G(t)$ for all t. $R(\cdot)$ determines the lower bound of the transmission schedules that optimize both the resource allocation and utilization.

Lemma 5. *$R(\cdot)$ is the minimal (r,b)-bounded transmission schedule.*

Proof. It can be proved by a similar method shown in Lemmas 1 and 3. □

Fig. 5 shows the upper bound $A(\cdot)$ and the lower bound $R(\cdot)$ for all the optimal transmission schedules with the same peak bandwidth r, buffer size b, initial delay d and network utilization u. Instead of giving a fixed schedule result, our approach gives the upper bound and the lower bound of the optimal transmission schedules. It allows users to determine their own optimal schedules under various QoS requirements and resource constraints to support differentiated services. For example, if we want to smooth the variance of transmission bandwidths applied, we can use the MVBA algorithm [20] to the upper bound and the lower bound of the optimal transmission schedules. As shown in Fig. 5, the obtained result not only provides the minimal bandwidth variance but also has the optimal resource allocation and utilization. It is better from the original MVBA algorithm that does not guarantee the optimum of initial delay and network utilization. The same idea can be applied to minimize the number of bandwidth changes by MCBA [9].

Note that, at any time t, $A(t)$ represent the maximal amount of media data that could be received by client without buffer overflow. The minimal amount of media data that should be received is $R(t)$. Let the media stream V be packed

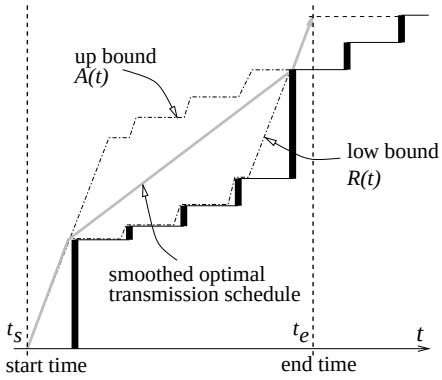


Fig. 5. An example that uses MVBA to the upper bound and the lower bound of the optimal transmission schedules. The result obtained not only has the minimal bandwidth variance but also has the optimal resource allocation and utilization.

as packets $\{p_0, p_1, \ldots\}$ for network transmission, and the cumulative size $P_x = p_0 + p_1 + \cdots + p_x$. When a transmission schedule $G(\cdot)$ is specified, packet p_x is transmitted/received at time t. Based on the upper bound $A(\cdot)$ and the lower bound $R(\cdot)$, we can specify the schedulable region (s_x, e_x) for each data packet p_x as follows:

$$\begin{aligned} s_x &= \mathbf{max}\{t \,|\, A(t) = P_x\}, \\ e_x &= \mathbf{max}\{t \,|\, R(t) = P_x\}. \end{aligned} \tag{11}$$

As shown in Fig. 6, the ready time s_x is the earliest time that p_x could start its transmission. The deadline e_x is the latest time that p_x should be received. By

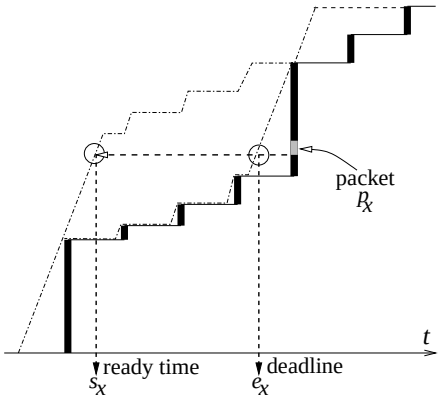


Fig. 6. We can utilize $A(\cdot)$ and $R(\cdot)$ to specify the schedulable region (ready time, deadline) $= (s_x, e_x)$ for each data packet p_x.

pre-specifying the real-time constraints (ready time and deadline) for each media packet, a better network scheduling and error control algorithm can be provided to transmit multiple media streams.

4. Experiment results

In this paper, different MPEG-encoded VBR media traces [22,23] are examined to evaluate the effectiveness of the proposed algorithm. The statistics of the test streams are listed in Table 1. It includes their frame numbers, the frame rates (number of frames played per second), the frame sizes (maximum, average, and standard deviation), and the group-of-picture sizes (maximum, average, and standard deviation). For each media stream, different performance parameters (such as buffer size, initial delay, network bandwidth, and network idle rate) are considered. The first video trace examined is a 2-h long MPEG-encoded movie *Star War*. The cumulative playback function of *Star War* is shown in Fig. 7(a) by its first 100 frames. Fig. 7(b) presents how the required buffer size and the obtained network idle rate change with the increase of allocated network bandwidth. As presented in [5], there is a tradeoff between the allocated buffer size and network bandwidth (called the bandwidth-buffer-tradeoff function). The required buffer size is piecewise-linearly and monotonically decreasing when the allocated bandwidth is increased. Note that, our algorithm is optimal in resource allocation. As shown in Fig. 7(b), it requires only 8 MB buffer to transmit *Star War* by 0.44 Mbps network bandwidth and 20 ms initial delay. Our algorithm is much better than CRTT [19] that requires over 10 GB client buffer to transmit *Star War* with jitter-free playback.

The second and the third test examples are two nearly 90 min long video traces: *Princess Bride* and *CNN News*. Both of them are encoded by Futuretel hardware MPEG coder with the same frame rate 30 fps (frame-per-second) and average frame size 4.89 KB. As the hardware coder uses variable distortion coding to maintain its target rate, the average group-of-picture (GoP) size is the same and the standard deviation of the GoP size is small. We show the

Table 1
Statistics of the media streams used in our experiments

Stream name	Frame number	Frames rate (fps)	Frame size (KB)			GoP size (KB)		
			Max	Avg	S.D.	Max	Avg	S.D.
Star War	174136	24	22.62	1.90	2.3	118.1	23.4	9.2
CNN News	164748	30	30.11	4.89	3.7	94.0	75.0	2.3
Princess Bride	167766	30	29.73	4.89	4.8	102.0	75.0	2.7
Lecture	16316	30	6.14	1.37	1.6	34.8	21.0	4.2
Advertisements	16316	30	10.08	1.86	1.9	124.1	28.5	13.0

fps: frame-per-second; GoP: group-of-picture.

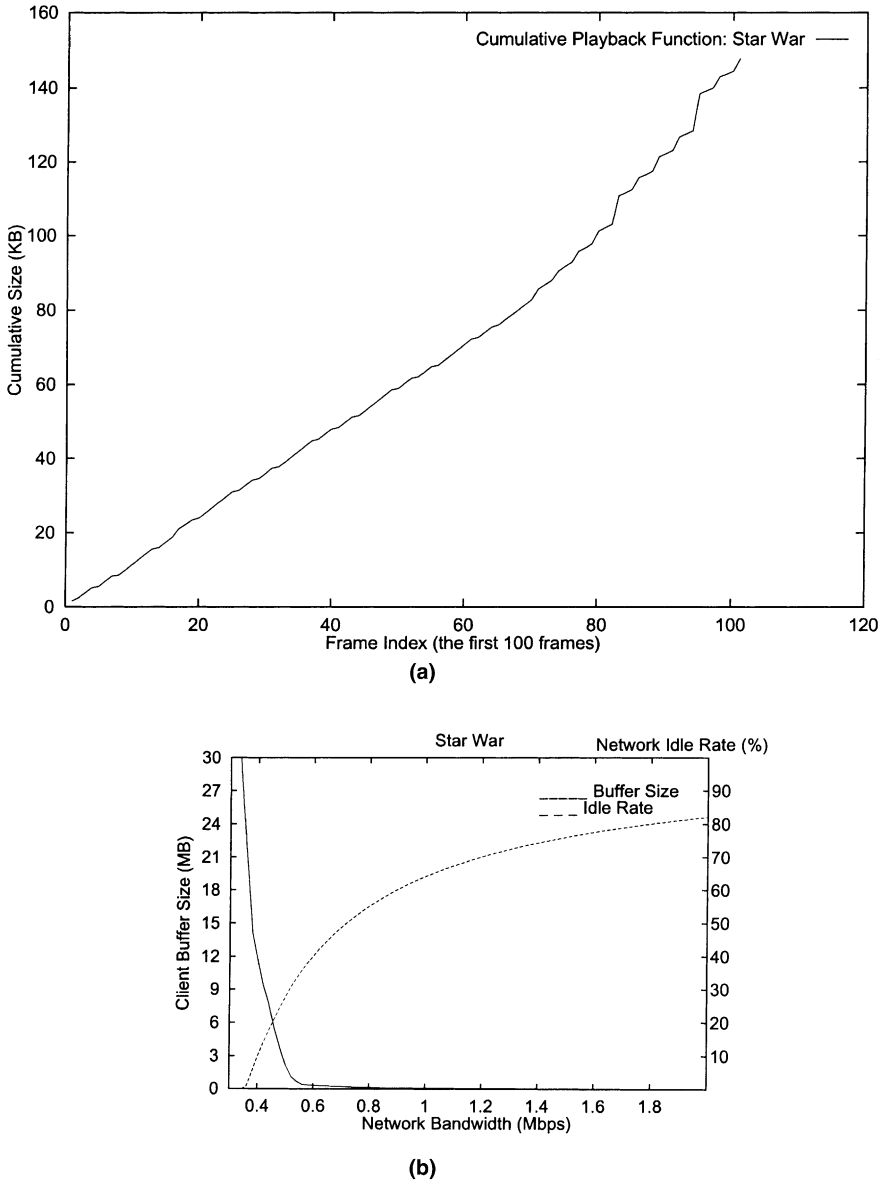


Fig. 7. *Star War*: (a) The cumulative playback function of the first 100 frames. (b) The optimal network bandwidth, buffer size and network idle rate obtained by our proposed algorithm.

cumulative playback function of the first 100 frames of *CNN News* in Fig. 8(a). The optimal resource allocation and utilization obtained are shown in Fig. 8(b). As the variance of frame sizes is small, the bandwidth-buffer-tradeoff

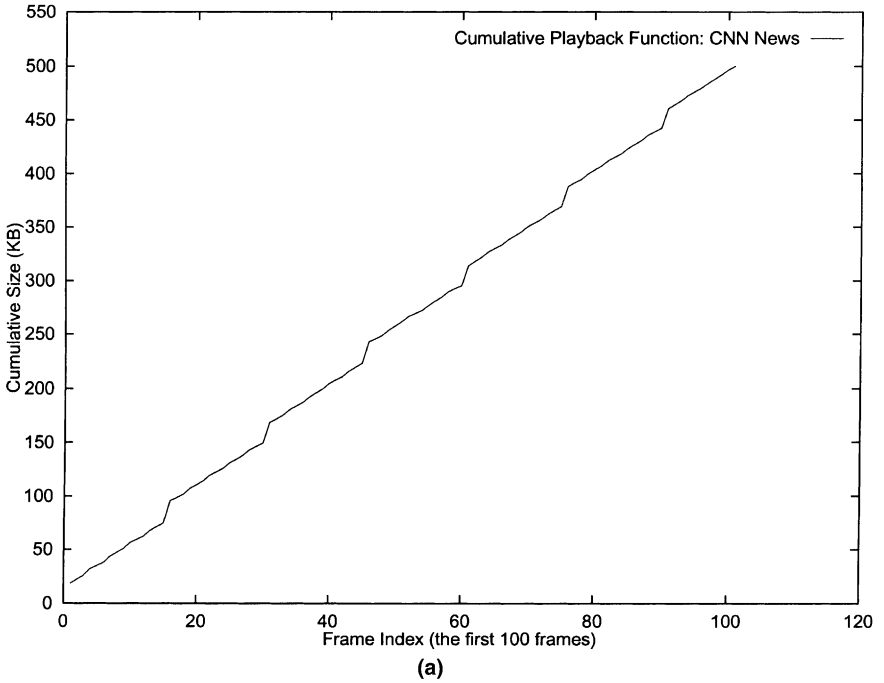
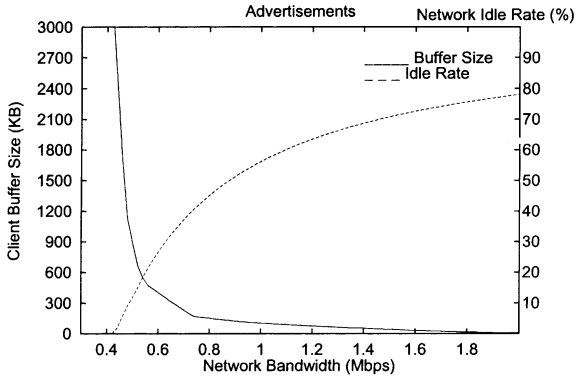


Fig. 8. *CNN News*: (a) The cumulative playback function of the first 100 frames. (b) The optimal network bandwidth, buffer size and network idle rate obtained by our proposed algorithm.

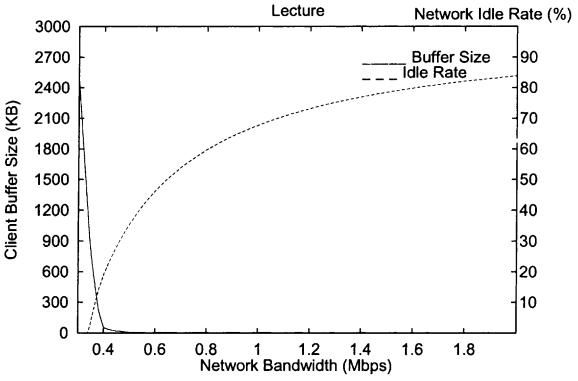
function is almost linear. Besides, the network idle rate is near zero when the transmission bandwidth is lower than the average stream rate 1.15 Mbps. For example, by applying the average stream rate as the transmission bandwidth,

the client can continuously play *CNN News* with 38 KB memory buffer and 150 ms initial delay. On an OC-3 link, we can support over 110 users to watch *CNN News* at the same time by the transmission schedule obtained. Comparing *Princess Bride* and *CNN News*, the hardware coded streams have almost identical statistics. The obtained results, i.e. the required transmission bandwidth, network utilization, buffer size and initial delay, are almost the same.

Different from the hardware coder, a software coder may introduce a high variance in GoP sizes due to the video contents presented. In this paper, our next two test examples are video traces *Advertisements* and *Lecture* encoded by the UCB software MPEG coder [14]. Both of them are about 10 min long with very different contents. *Lecture* shows a speaker and slides with zooming and



(a)



(b)

Fig. 9. The optimal network bandwidth, buffer size and network idle rate obtained by our proposed algorithm for (a) *Advertisements* and (b) *Lecture*.

panning. As the frame contents are similar, the variation of GoP sizes is small. On the contrary, *Advertisements* contains a sequence of advertisements. It has the different frame contents from one scene to another scene. Therefore, the variation of GoP sizes is large. Fig. 9(a) and (b) shows the optimal resource allocation and utilization obtained for *Advertisements* and *Lecture*, respectively. Our experiments conclude that the GoP sizes and their variations may affect the resources required for media transmission. The network bandwidth and the network idle rate required for transmitting high-variance *Advertisements* is larger than that required for transmitting low-variance *Lecture* under the same client buffer size. The relations between the required buffer size and the obtained network idle rate for different streams, high-variance *Advertisements* and low-variance *Lecture*, are compared in Fig. 10.

Although our approach has already proved optimal in resource allocation and utilization, we would like to compare our optimal schedule results to the schedule results obtained by previous approaches to measure the improvements achieved. In Fig. 11, we compare the proposed algorithm to CRTT by the client buffer required for transmitting *Advertisement* under different initial delays. Experiments show that, only when the initial delay is over 50 s, CRTT can provide the similar buffer size as our approach obtained. Our improvement in required buffer size is dramatic. Fig. 12 shows the comparisons of our proposed algorithm and MVBA by network idle rate obtained under different initial delays. Note that, as MVBA requires pre-specifying buffer size and initial delay to decide the related network bandwidth, it does not provide a way to minimize both buffer size and initial delay at the same time. To do the fair comparisons, the optimal buffer size and initial delay obtained by our approach

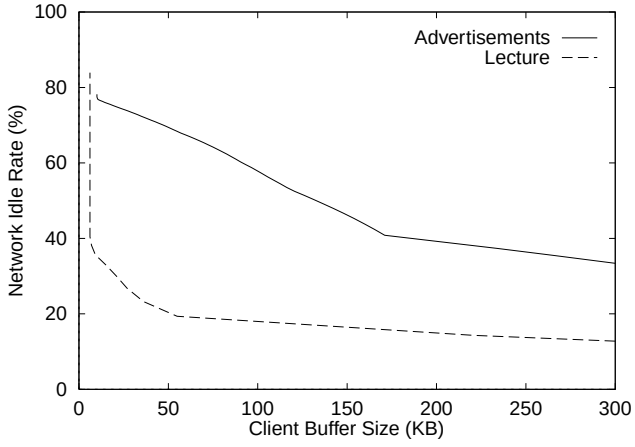


Fig. 10. The relations between the required buffer size and the obtained network idle rate for different streams, *Advertisements* and *Lecture*.

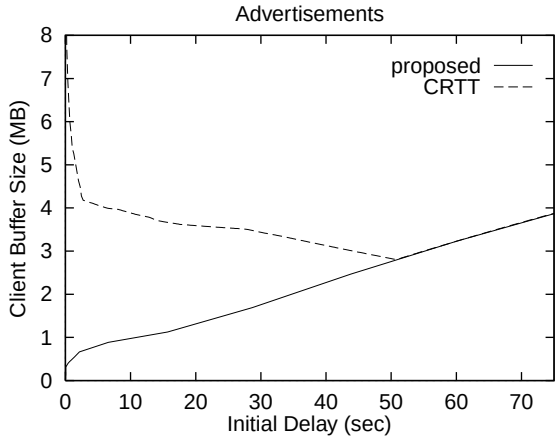


Fig. 11. A comparison of the client buffer size required for our proposed algorithm and CRTT with different initial delay.

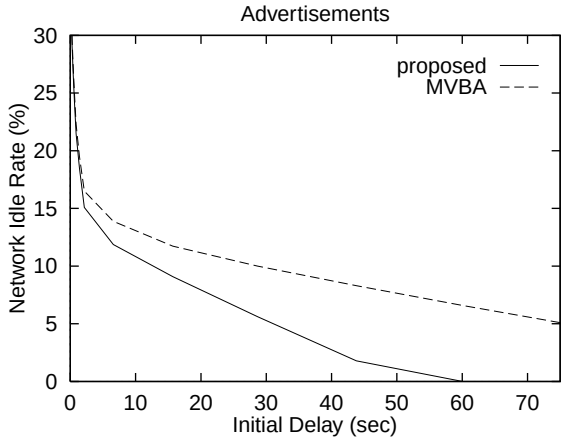


Fig. 12. A comparison of the network idle rate required for our proposed algorithm and MVBA with different initial delay.

are assigned as the given parameters to MVBA. Experiments show that our approach is better than the MVBA approach in network utilization obtained.

5. Conclusion

In this paper, a traffic shaping scheme is introduced to decide the suitable transmission schedules. Instead of giving a fixed schedule result, our approach

decides the schedulable region for all the transmission schedules that have the optimal allocation and utilization in system resources. Experiments have shown that our algorithm can achieve a dramatic improvement than the conventional approaches in both the buffer size and the network idle rate. Based on the schedulable region provided, the ready time and deadline are precisely specified to each media packet to support real-time network scheduling and error control. The proposed approach is shown to be practical, efficient, and flexible in supporting continuous media transmission.

References

[1] T.M. Apostol, Math. Anal. (1973).
[2] E. Chang, A. Zakhor, Scalable video data placement on parallel disk arrays, in: IS & T/SPIE Symposium on Electronic Imaging Science and Technology, Febraury 1994.
[3] R.I. Chang, M.C. Chen, J.M. Ho M.T. Ko, Designing the on-off CBR transmission schedule for jitter-free VBR media playback in real-time networks, in: IEEE RTCSA, 1997, pp. 1–9.
[4] R.I. Chang, M.C. Chen, J.M. Ho, M.T. Ko, Optimizations of stored VBR video transmission on CBR channel, in: SPIE VVDC, 1997, pp. 382–392.
[5] R.I. Chang, M.C. Chen, M.T. Ko, J.M. Ho, Characterize the minimum required resources for admission control of pre-recorded VBR video transmission by an $O(n \log n)$, in: IEEE IC3N, 1998.
[6] R.I. Chang, M.C. Chen, J.M. Ho, M.T. Ko, An effective and efficient traffic smoothing scheme for delivery of online VBR media streams, in: IEEE INFOCOM, 1999.
[7] R.I. Chang, W.K. Shih, R.C. Chang, A new real-time disk scheduling algorithm and its application to multimedia systems, in: 5th IEEE IDMS, Lecture Notes on Computer Science, 1998.
[8] M. Chen, D.D. Kandlur, P.S. Yu, Optimization of the grouped sweeping scheduling (GSS) with heterogeneous multimedia streams, in: ACM Multimedia Conference, 1993, pp. 235–242.
[9] W. Feng, F. Jahanian, S. Sechrest, Optimal buffering for the delivery of compressed prerecorded video, in: IASTED International Conference on Networks, January 1996.
[10] W. Feng, S. Sechrest, Smoothing and buffering for delivery of prerecorded compressed video, in: IS&T/SPIE Multimedia Computing and Networking, February 1995, pp. 234–244.
[11] M. Garrett, W. Willinger, Analysis, modeling and generation of self-similar VBR video traffic, in: ACM SIGCOMM, August 1994, pp. 269–280.
[12] M. Grossglauser, S. Keshav, On CBR service, in: IEEE INFOCOM, March 1996.
[13] M. Grossglauser, S. Keshav, D. Tse, RCBR: a simple and efficient service for multiple time-scale traffic, in: ACM SIGCOMM, August 1995.
[14] E.W. Knightly, D.E. Wrege, J. Liebeherr, H. Zhang, Fundamental limits and tradeoffs of providing deterministic guarantees to VBR video traffic, in: ACM SIGMETRICS, May 1995, pp. 98–107.
[15] M. Krunz, H. Hughes, A traffic model for mpeg-coded VBR streams, in: ACM SIGMETRICS, May 1995, pp. 47–55.
[16] S.S. Lam, S. Chow, D.K.Y. Yau, An algorithm for lossless smoothing of MPEG video, in: ACM SIGCOMM, 1994.
[17] T. Ott, T.V. Lakshman, A. Tabatabai, A scheme for smoothing delay-sensitive traffic offered to ATM networks, in: IEEE INFOCOM, 1992.
[18] A.R. Reibman, A.W. Berger, Traffic descriptors for VBR video teleconferencing over ATM networks, in: IEEE/ACM Trans. Networking, June 1995.

[19] J.M. McManus, K.W. Ross, Video on demand over ATM: Constant-rate transmission and transport, in: IEEE INFOCOM, March 1996.
[20] J.D. Salehi, Z.L. Zhang, J.F. Kurose, and D. Towsley, Supporting stored video: reducing rate variability and end-to-end resource requirements through optimal smoothing, in: Proceedings ACM SIGMETRICS, 1996.
[21] K. Sohraby, On the theory of general on-off source with applications in high-speed networks, in: IEEE INFOCOM, 1993, pp. 401–410.
[22] FTP ftp::/thumper.bellcore.com.
[23] URL http::/www.eeb.ele.tue.nl/mpeg/.
[24] Y.C. Wang, S.L. Tsao, R.I. Chang, M.C. Chen, J.M. Ho, M.T. Ko, A fast data placement scheme for video server with zoned-disks, in: SPIE VVDC, 1997, pp. 92–102.
[25] H. Zhang, E.W. Knightly, A new approach to support delay-sensitive VBR video in packet-switched networks, in: IEEE NOSSDAV, 1995.
[26] J. Zhang, J. Hui, Traffic characteristics and smoothness criteria in VBR video traffic smoothing, in: IEEE ICMCS, June 1997, pp. 3–11.
[27] R.I. Chang, W.K. Shih, R.C. Chang, Real-time disk scheduling for multimedia applications with deadline-modification-scan scheme, Real-time Syst. 19 (2) (2000) 149–168.
[28] R.I. Chang, C.L. Chen, W.K. Shih, R.C. Chang, Design and implementation of a real-time disk scheduling algorithm by the multi-segment cascade scheme, in: IEEE RTSS (WIP) 1999, pp. 1–5.