AN ALGORITHM FOR COMPUTING THE BURSTINESS CURVE FOR VBR MPEG-2 VIDEO SOURCES

Subir Varma

Emerging Technologies, Dept. C70A/673
Networking Systems Architecture, IBM Corp., NC 27709
email: *subirv@vnet.ibm.com*, Ph. No.: (919) 254-7493

ABSTRACT

The buffer and bandwidth resources that a video source requests from an integrated services network, is governed by its burstiness curve. Hence it is extremely important to obtain efficient algorithms for computing it. In this paper we give an efficient algorithm to compute the burstiness curve for VBR MPEG-2 sources, which utilize the MPEG-2 Transport Systems (TS) layer. We do so by taking advantage of the fact that the rate of the MPEG-2 TS stream is restricted to be piecewise constant.

1. Introduction

Consider a video source with rate $r(t), 0 \leq t \leq T$, such that for each interval $[t_1, t_2]$, the following equation is satisfied:

Defn of leaky bucket:
$$\int_{t_1}^{t_2} r(t) \leq \sigma + (t_2 - t_1)\rho, \quad 0 \leq t_1 \leq t_2 \leq T$$

The source is then said to be (σ, ρ) conformant. Unfortunately, as several researchers have pointed out [Gr], [LoVa], [Va], there is no single pair (σ, ρ) that uniquely characterizes a source. In fact for every value of ρ in the range $0 \leq \rho \leq \max_{0 \leq t \leq T} r(t)$, there is a σ such that the source is (σ, ρ) conformant. Hence the set of conformant (σ, ρ) pairs describe a curve which has been referred to as the *burstiness curve* by Low and Varaiya [LoVa]. Integrated service networks such as ATM use the burstiness curve to reserve buffer and bandwidth resources within the network. In general, the bandwidth requirements are proportional to ρ, while the buffer requirements are proportional to σ. Hence by varying ρ and σ, the server can realize a bandwidth vs buffer trade-off in the network. At the time when the server gets ready to transmit the video into the network, it can use the burstiness curve to request appropriate QOS values [Va]. As an example of how this may be done. consider the case when the peak rate of the source is $r_{\max}$. In order to save bandwidth resources, if the source wishes to reduce its peak rate to $r'_{\max} < r_{\max}$, then it incurs an additional delay of $\frac{\sigma(r'_{\max})}{r'_{\max}}$, where the value $\sigma(r'_{\max})$ can be obtained from the burstiness curve.

A brute-force way to obtain burstiness curve is based on the following result due to Cruz [Cr]:

$$\sigma = \max_{0 \leq s \leq t \leq T} [\int_s^t r(u)du - \rho(t-s)]$$

It follows from this that if the source is fed into a single server queue with deterministic rate ρ, then the max queue length over the interval $[0, T]$ is the corresponding σ. However this procedure can be very time consuming especially if the packet size is 53 bytes, as for ATM.

The objective of this paper is to present an efficient algorithm for computing the burstiness curve for VBR MPEG-2 sources that utilize the MPEG-2 Transport Systems (TS) layer. MPEG-2 is an emerging coding standard that is gaining widespread acceptance in the new Video-on-Demand (VOD) marketplace. The TS layer, which lies between the codec and the networking layers, provides additional functionality such as clock recovery and stream synchronization. However it restricts the rate of the output stream to be piecewise constant. We take advantage of this property and obtain an efficient algorithm to compute the burstiness curve for these sources, which does not require detailed simulation.

2. The Algorithm

The strategy that we will pursue to obtain the burstiness curve is to analytically derive the maximum queue length for a set of values of ρ. Furthermore we show that the entire curve can be obtained by connecting these points with linear segments.

Let $m(t), 0 \leq t \leq T$ be the rate function of the MPEG-2 source in the interval $[0, T]$ such that $m(t) = r_i, t \in [x_{i-1}, x_i], 1 \leq i \leq N$. Also let $x_0 = 0, x_N = T$ and

ule (Transport Rate) $m(t)$ changes at the boundaries x_i

$$\underbrace{\sum_{i=1}^{N}(x_i - x_{i-1})}_{\text{sum of all the intervals}} = T$$

The interpretation is that the time interval $[0, T]$ is divided into N parts, such that the rate of $m(t)$ over the i^{th} interval is r_i. Let $r_{\min}$ and $r_{\max}$ be the minimum rate and maximum rates, so that

$$r_{\min} = \min_{1 \leq i \leq N} r_i \quad \text{and} \quad r_{\max} = \max_{1 \leq i \leq N} r_i$$

The total amount of data contained in the MPEG-2 source M, is given by

$$M = \sum_{i=1}^{N}(x_i - x_{i-1})r_i$$

It is clear that $\sigma(0) = M$, since if the video source is sent into a single server queue with rate 0, then the maximum queue length is the same as the amount of data.

We start by showing that the equation for the burstiness curve in the interval $\rho \in [0, r_{\min})$ is given by

$$\sigma = M - \rho T, \quad 0 \leq \rho < r_{\min}. \tag{1}$$

Obvious since in that case the queue increases monotonically

Consider a single server queue with rate $\rho \in [0, r_{\min})$. If the video source is sent into this queue, then the queue length increases monotonically from zero at $t = 0$ to $M - \rho T$ at $t = T$, since at all times the rate of the queue is less than the rate of the video source. Also note that for $\rho \geq r_{\max}$, we have that $\sigma = 0$, since the rate of the queue greater than the input rate at all times, so that no queue is allowed to build up. This implies that the burstiness curve lies in the interval $\rho \in [0, r_{\max}]$.

$s_i(\rho_k)$ = start [illegible] i for shaper with rate ρ_k
$t_i(\rho_k)$ = stop time for busy period i for shaper with rate ρ_k

For the case when $\rho \geq r_{\min}$, the analysis is complicated by the fact that there are several busy periods in the interval $[0, T]$. Hence in order to obtain the maximum queue length, we need to consider every busy period and calculate the maximum queue length achieved in each. Our task is simplified by the following result:

Proposition: *The burstiness curve is piecewise linear and convex.*

Proof: Note that as the value of ρ increases from 0 to $r_{\max}$, the number of busy periods increases from 1 to L, where L is the number of packets. Consider the case when the maximum queue size occurs in the i^{th} busy period. Assuming that the i^{th} busy period started at $t = s_i$, the maximum queue length σ is thus given by

: # pkts in the Transport Stream

$$\sigma = \int_{s_i}^{x_p} m(t)dt - \rho(x_p - s_i) \tag{2}$$

The equation of the burstiness curve is given by (2), and changes under the following two circumstances as ρ increases:

(1): The maximum now occurs at time x_p' within the same busy period. In this case necassarily $x_p' < x_p$ so that the slope increases.

(2): The maximum occurs at time x_p' in some other busy period (which starts at s_i'). Once again, this is only possible if $x_p' - s_i' < x_p - s_i$ so that the slope of the burstiness curve increases. Since an increase in ρ leads to an increase in the slope of the burstiness curve, it is convex. ∎

By virtue of this proposition, we only need to compute points on the burstiness curve at places where the slope changes, since the entire curve can be obtained by joining these points with linear segments. The set of these points is defined as $\{\rho_k\}_{k=0}^{S}$. Note that $\rho_0 = 0, \rho_1 = r_{\min}$ and $\rho_S = r_{\max}$.

The following is a brief summary of the algorithm: Let us assume that we have obtained the burstiness curve in the interval $[0, \rho_k)$. In order to extend it to the interval $[\rho_k, \rho_{k+1})$, we proceed in the following three steps:

- Identify all the busy periods for $\rho = \rho_k$. Assume that there are $n(\rho_k)$ such busy periods, and the i^{th} one spans the interval $[s_i(\rho_k), t_i(\rho_k)], 1 \leq i \leq n(\rho_k)$.
- In each busy period, obtain the maximum queue length, say $q_i(\rho_k), 1 \leq i \leq n(\rho_k)$. Also compute the values of ρ at which the maximum shifts to some other point within the busy period as well the value at which the busy period splits into two or more busy periods. Lastly compute the value of ρ at which the maximum queue length shifts from one busy period to another. Then ρ_{k+1} is the minimum of these three values of ρ. Moreover, the equation of the straight line between ρ_k and ρ_{k+1} can be obtained by formulae that are given in the detailed description.

A detailed description of the algorithm now follows:

(0) : Initialize the algorithm by setting $\rho' = r_{\min}$, $b = 0$, $s_0(\rho') = 0$ and $t_0(\rho') = 0$.

(1) : In the rate function $m(t), 0 \leq t \leq T$, find the smallest time $x_i > t_b(\rho')$ such that the corresponding rate r_{i+1} (in the interval $[x_i, x_{i+1})$) satisfies $r_{i+1} > \rho'$. If no such x_i exists, then all the busy periods have been identified for $\rho = \rho'$, hence go to step (4). Otherwise set $b \leftarrow b + 1$ and $s_b(\rho') = x_i$.

(2a) : For $j = i+1, \ldots, h$, compute the quantities $p_{ij}^b(\rho')$ given by

$$p_{ij}^b(\rho') = \sum_{a=i+1}^{j} (x_a - x_{a-1})r_a - (x_j - x_i)\rho' \tag{3}$$

Note that either,

- h is the smallest integer greater that i, that satisfies $p_{ih}^b(\rho') < 0$, or
- $x_h = T$.

The maximum queue size over the b^{th} busy period is given by

$$q_b(\rho') = \max_{i+1 \leq j \leq h} p_{ij}^b(\rho') \tag{4}$$

Let the maximum be achieved for indices $(i(b), j(b))$. If $p_{ih}^b(\rho') < 0$, then the time at which the busy period ends, $t_b(\rho')$ is given by,

$$t_b(\rho') = \frac{\sum_{a=i+1}^{h-1}(x_a - x_{a-1})r_a + x_i\rho' - x_{h-1}r_h}{\rho' - r_h}, \tag{5}$$

otherwise, set $t_b(\rho') = T$ if $x_h = T$.

(2b) : Let $\rho'_{b1} > \rho'$ be smallest value of ρ at which the b^{th} busy period splits apart into two or more busy periods. In order to compute this quantity, first compute the quantities $\rho_{ij}^{b1}, j = i+1, \ldots, h-1$ for the b^{th} busy period given by

$$\rho_{ij}^{b1} = \frac{\sum_{a=i+1}^{j}(x_a - x_{a-1})r_a}{(x_j - x_i)}$$

Then,

$$\rho'_{b1} = \min[r_{\max}, \min_{i+1 \leq j \leq (h-1)} \rho_{ij}^{b1}] \tag{6}$$

(2c) : Let $\rho'_{b2} > \rho'$ be the smallest value of ρ at which the maximum in the b^{th} busy period shifts. In order to compute this quantity, first compute the quantities $\rho_{ij}^{b2}, j = i+1, \ldots, j(b) - 1$ for the b^{th} busy period given by

$$\rho_{ij}^{b2} = \frac{\sum_{a=i+1}^{j(b)}(x_a - x_{a-1})r_a - \sum_{a=i+1}^{j}(x_a - x_{a-1})r_a}{(x_{j(b)} - x_i) - (x_j - x_i)}$$

Note that it is sufficient to compute ρ_{ij}^{b2} for the intervals $[x_{j-1}, x_j]$ for which the corresponding rate r_j exceeds the rate in the interval $[x_{j(b)-1}, x_{j(b)}]$. Then,

$$\rho'_{b2} = \min[r_{\max}, \min_{i+1 \leq j \leq (j(b)-1)} \rho_{ij}^{b2}] \tag{7}$$

(3) : If $t_b(\rho') = T$, then go to step (4), otherwise go back to step (1).

(4) : Set $n(\rho') = b$. If $n(\rho') = 0$ then STOP. Otherwise compute the max queue length achieved for $\rho = \rho'$ as

$$q(\rho') = \max_{1 \leq b \leq n(\rho')} q_b(\rho')$$

If the maximum is achieved at $b = B$, then identify $i(B), j(B)$, such that $q_B(\rho_k) = p^B_{i(B)j(B)}(\rho')$. Then the burstiness curve satisfies the following equation for $\rho \in [\rho', \rho'')$, where ρ'' is computed in the next step.

$$\sigma(\rho) = \sum_{a=i(B)+1}^{j(B)} (x_a - x_{a-1})r_a - (x_{j(B)} - x_{i(B)})\rho, \quad \rho' \leq \rho < \rho'' \tag{8}$$

(4b) : Let

$$\rho'_1 = \min_{1 \leq b \leq n(\rho')} \rho'_{b1}$$

and

$$\rho'_2 = \min_{1 \leq b \leq n(\rho')} \rho'_{b2}$$

Let ρ'_3 be the minimum value of ρ at which the maximum shifts from the B^{th} busy period, then

$$\rho'_3 = \min_{1 \leq b \leq n(\rho')} \frac{\sum_{a=i(B)+1}^{j(B)}(x_a - x_{a-1})r_a - \sum_{a=i(b)+1}^{j(b)}(x_a - x_{a-1})r_a}{(x_{j(B)} - x_{i(B)}) - (x_{j(b)} - x_{i(b)})}$$

Then

$$\rho'' = \min(\rho'_1, \rho'_2, \rho'_3)$$

(6) : If $\rho'' = r_{\max}$, then STOP. Otherwise replace ρ' by ρ'', set $b = 0$, $s_0(\rho') = 0$, $t_0(\rho') = 0$ and go back to step (1).

We now provide justifications for the steps in this algorithm:

- In step (0) we initialize the algorithm at $\rho' = r_{\min}$, since equation (1) describes the burstiness curve in the interval $[0, r_{\min})$.
- In step (1) while locating the start of the next busy period, we can skip the intervals over which $r_i < \rho'$, since there will not be any build up of the queue at those times.
- The quantities $p^b_{ij}(\rho')$ in step (2) are the queue sizes at the end of the interval $[x_i, x_j]$. Note that it is sufficient to obtain the queue sizes at the time instants at which the rate $m(t)$ changes, since the maximum queue size always occurs at these rate change instants. Note that if $p^b_{ih}(\rho') < 0$, then $t_b(\rho')$ satisfies the equation

$$\sum_{a=i+1}^{h-1} (x_a - x_{a-1})r_a + (t_b(\rho') - x_{h-1})r_h = (t_b(\rho') - x_i)\rho'$$

which leads to (5).

- In step (2b), the quantity ρ_{ij}^{b1} satisfies the equation $p_{ij}^{b}(\rho_{ij}^{b1}) = 0$, and so is interpreted as the minimum value of ρ at which the queue size at the end of the interval $[x_i, x_j]$ becomes zero, i.e., the b^{th} busy period ends at $t = x_j$.
- In step (2c), the quantity ρ_{ij}^{b2} is the point at which the straight lines

$$\sigma = \sum_{a=i+1}^{j(b)} (x_a - x_{a-1})r_a - (x_{j(b)} - x_i)\rho$$

$$\sigma = \sum_{a=i+1}^{j} (x_a - x_{a-1})r_a - (x_j - x_i)\rho$$

  meet.
- In step (4), the summation is the maximum queue size achieved in the B^{th} busy period.

3. Conclusions

In this paper we provided an efficient algorithm to compute the burstiness curve for VBR MPEG-2 video sources. It can be used to pre-process MPEG-2 video traces and the burstiness curve can be stored as part of the meta-data associated with the actual trace data.

Even though the algorithm computes the entire curve in the interval $[0, r_{\max}]$, it can be easily modified to compute it in a smaller interval. For example the smallest value of ρ which would be requested from a network is the long term average, hence the algorithm can be started there rather than at the origin. Similarly is can be stopped before going all the way to $r_{\max}$.

REFERENCES

[Cr] R. L. Cruz, "A calculus for network delay, Part I: Network elements in isolation," *IEEE Transactions on Information Theory*, Vol 37, No. 1, pp. 114-131, (Jan, 1991).

[Gr] M. Graf, "Traffic shaping of VBR video in ATM end systems," *Proc. 4th Open Workshop on High Speed Networks*, Brest, France, (Sept, 1994).

[LoVa] S. Low and P. Varaiya, "A simple theory of traffic and resource allocation in ATM," *INFOCOM'91*, pp. 1633-1637, (1991).

[Va] S. Varma, "Specifying traffic descriptor and QOS parameters for video sources," *In Preparation*, (1994).