The Jitter Time Stamp Approach for Clock Recovery of Real-Time Variable Bit Rate Traffic

Weilian Su and Ian F. Akyildiz

Broadband and Wireless Networking Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Tel: (404)-894-5141; Fax: (404) 894-7883
Email: {weilian,ian}@ee.gatech.edu

Abstract—When multimedia streams arrive at the receiver, their temporal relationships may be distorted due to jitter. Assuming media stream is packetized, the jitter is then the packet’s arrival time deviation from its expected arrival time. There are various ways to reduce jitter which include synchronization at the application layer, or synchronization at the asynchronous transfer mode (ATM) adaptation layer (AAL). The new source rate recovery scheme called Jitter Time Stamp (JTS) provides synchronization at the ATM adaptation layer 2 (AAL2) which is used to carry variable bit rate (VBR) traffic such as compressed voice and video. JTS is implemented, and experiments have shown that it is able to recover the source rate.

Keywords—ATM Networks, Constant Bit Rate (CBR), Variable Bit Rate (VBR), ATM Adaptation Layer 2 (AAL2), Timing Recovery, Synchronization.

I. INTRODUCTION

Clock recovery at the ATM adaptation layer 1 (AAL1) utilizes the synchronous residual time stamp (SRTS) method [14] which is designed for CBR traffic carried by AAL1 segments. Note that we use the term packet(s) instead of AAL segment(s) throughout the paper. The SRTS method maps the service clock to the network clock to determine the packet’s residual timing information. This information is used by the receiver with the network clock to reconstruct the source rate.

The pulse stuffing and the absence of a network clock in SRTS [14], [13] contribute additionally to the jitter, which is the packet’s arrival time deviation from its expected arrival time.

Pulse stuffing introduces jitter even at undesirable low frequencies depending on the pulse stuffing ratio between the network clock and the source clock. Research has been conducted to determine pulse stuffing ratios which minimize the waiting time jitter [12] (or pulse stuffing jitter in conventional stuffing synchronizer). Since the receiver depends on the network clock to reconstruct the source clock, a slight deviation of the network clock will create noise in the reconstructed source clock.

A new scheme is proposed in [15] to reconstruct the source frequency in the AAL1 layer. The scheme uses a buffer control technique which changes the bandwidth of the low pass filter according to the filling level of the buffer and the statistical characteristic of the jitter.

As of date, to our knowledge, there is no source rate recovery scheme for AAL2 [10]. AAL2 is mainly used to carry VBR traffic such as compressed voice and video. As mentioned before, the SRTS method is used for constant bit rate (CBR) traffic and is not well-suited for variable bit rate (VBR) traffic, because VBR has variable ON and OFF burst lengths. Since compressed multimedia traffic is inherently bursty, the best way to transmit it is through VBR connections. By this way, more users can be supported at the same time and the bandwidth gain increases significantly. In particular, the bandwidth gain of multiplexing VBR sources into AAL2 packets has been demonstrated in [18], [19], [20], [21].

In this paper, we introduce a new source rate recovery scheme, called Jitter Time Stamp (JTS) approach which will be used in AAL2 layer. Recovering the source rate at the AAL2 layer has the following advantages:

- The computational load of the application layer can be reduced,
- No need for traffic smoothing since traffic is sent as VBR,
- VBR video coding is allowed and relatively stable picture quality can be achieved, and
- VBR sources can be multiplexed to achieve high bandwidth gain while QoS requirements can still be satisfied.

This work was supported by NASA-Ames under Contract # NAG 2-1262
This paper is organized as follows: In Section II we present an overview of the new source rate recovery scheme called JTS. In Section II.A we describe the initialization steps necessary to obtain the round trip delays measured at the source and receiver, and the reference network delay. In Section II.B we explain how packets obtain their time indication (TI) at the source while in Section II.C we describe how timing packets are dejittered at the receiver. The packet(s) with timing information will be referred as timing packet(s) throughout the paper. Also, the terms data and timing packet will be used instead of packet’s payload and timing packet’s payload, respectively, throughout the paper. Due to the changes in the network condition and drifting reference clocks, packets’ jitters are biased. This situation is also addressed. After incoming timing packets are dejittered, the source rate is recovered with the source rate recovery scheme. In Section III we evaluate the performance of the JTS scheme. In Section IV we conclude the paper.

II. THE JITTER TIME STAMP (JTS) APPROACH

The Jitter Time Stamp (JTS) approach has the following procedures:

- **Initialization**
- **Source Behavior**: How timing information and sequence numbers are inserted into data packets, which are sent to the receiver. These information are needed by the receiver to reconstruct the VBR traffic’s source rate.
- **Receiver Behavior**: How VBR traffic’s source rate is reconstructed at the receiver with the timing information, sequence number, estimated jitter, and the waiting time for the next packet from the source.

A. Initialization

Before VBR traffic is sent to the receiver, initialization between source and receiver is performed. Both the source and receiver send a packet to acquire the round trip delay of the network. $D_S$ and $D_R$ are the round trip delays measured at the source and receiver, respectively, in terms of reference clock cycles. The source and receiver then communicate their round trip delays, $D_S$ and $D_R$, and select the reference network delay, $D_{ref}$.

$$D_{ref} = 1/2 \cdot \max\{D_S, D_R\}$$  \hspace{1cm} (1)

The VBR traffic has the typical ON (active) and OFF (passive) periods. We assume packets are sent at source rate, $SRA$, during the active period. The source rate for active period, $SRA$, is also sent to the receiver by the source as

$$SRA = \frac{\gamma}{\eta} \text{ (packets/sec)}$$  \hspace{1cm} (2)

where $\gamma$ is the source rate in bytes/sec and $\eta$ is the number of information bytes per packet. For example, a voice talkspurt (an active period) is packetized with $\eta$ information bytes per packet and sent at $SRA$ while otherwise nothing is sent. Each packet is assumed a fixed packet size.

The receiver can then use the $SRA$ value to calculate the reference clock frequency $f_r$ used at the AAL level. $f_r$ is the reference clock frequency in Hz which one packet is processed with each $f_r$ clock tick. For example, if $SRA$ is 400 packets/second, $f_r$ is 400 Hz. $f_r$ is also the source clock used by the source to send a packet from the application layer to AAL with each clock tick. The period of the reference clock frequency $f_r$ is then

$$T_r = \frac{1}{f_r} \text{ (sec)}$$  \hspace{1cm} (3)

Also, the value $N$ is sent to the receiver by the source. $N$ is the number of reference clock cycles per $C_s$ counter increment as shown in Figure 1. It also indicates how often packets are inserted with timing information which is explained in Section II.B.

B. Source Behavior

VBR traffic has typical ON (active) and OFF (passive) periods. During an active period, the VBR traffic is packetized. Every packet generated by the source will carry a sequence number (SN). Before we proceed with the description of the source behavior, we present some definitions.

**Definition 1. Timing Insertion.** Every $N^{th}$ packet contains the timing information. Such packet is referred as a timing packet as previously mentioned. Insertion of this timing information at every $N^{th}$ packet is called timing insertion.

As shown in Figure 1, the value of $C_s$ counter is latched when timing packet ($N^{th}$ packet) arrives. The latched value is the time indication (TI) for the $N^{th}$ packet. The number of bits used for $C_s$ counter is

$$b \geq \lceil \log_2 \left( \frac{J_{max}}{T_r \cdot N} \cdot 2 \right) \rceil$$  \hspace{1cm} (4)

where $N$ is the number of reference clock cycles per $C_s$ counter increment, $T_r$ is calculated by equation (3), and $J_{max}$ is the network’s maximum jitter. The value 2 on the right hand side of equation (4) is needed, because the time indication (TI) must be able to represent twice the network’s maximum jitter in order for the receiver to estimate the received packet’s jitter with counters. Equation

$$T_r = \frac{1}{f_r} \text{ (sec)}$$  \hspace{1cm} (3)
(4) must be satisfied in order for the dejittering process at the receiver to work properly.

**Definition 2. Timing Insertion Accuracy.** A timing packet could arrive before the $C_s$ counter changes to the next value since $C_s$ counter changes value every $N$ reference clock cycles. The timing insertion accuracy is the number of reference clock cycles away since when the $C_s$ counter has last changed value when a timing packet arrives and obtains the time indication ($TI$).

Since the choice of $N$ in equation (4) affects the timing insertion accuracy, an error similar to quantization error may occur. If the packet’s arrival time is asynchronous, i.e., when packet arrival pulse is not aligned with rising edge of the reference clock, then this error $\varepsilon_{\text{insertion}_A}$ is constrained by

$$\varepsilon_{\text{insertion}_A} \leq N \cdot T_r$$

(5)

Since the reference clock $f_r$ is the same as the source clock, the packet’s arrival time is synchronous to the reference clock. As a result, the error $\varepsilon_{\text{insertion}_S}$ for the synchronous case is

$$\varepsilon_{\text{insertion}_S} \leq (N - 1) \cdot T_r$$

(6)

**C. Receiver Behavior**

As previously mentioned in Section I, data and timing data are referred as the packet’s payload and the timing packet’s payload. When VBR traffic in an active period is packetized and sent through an ATM network, the packets’ arrival time at the receiver may deviate from their expected arrival time. The receiver estimates the timing packets’ jitter by using the timing packets’ timing information and their measured arrival time which are latched values of two counters. The timing information represents only the variable part of the network delay since only varying network delay causes jitter to packets. At the receiver, the source rate is recovered by using the timing packets’ timing information, sequence number, estimated jitter, and the waiting time for the next packet.

The JTS recovers the source rate rather than the source clock to maintain a temporal relationship between consecutive packets of a real-time VBR stream. Unlike the CBR traffic, the VBR traffic has a difference in meaning between source rate and source clock. Source clock is a clock used by the application to send data to the AAL, i.e., the reference clock $f_r$ with each clock tick a packet is sent to the AAL. Source rate is the number of packets sent to AAL per second since the VBR traffic consists of ON and OFF periods.

A detailed description of the receiver behavior will be discussed in the following sections.

**C.1 The Packet Arrival**

As shown in Figure 2, when a packet arrives at the receiver, the “sort packet” block uses the received packet’s sequence number ($SN$) to find the correct position $\omega$ in the data buffer $B_{\text{data}}$ for inserting the new data when the received packet is arrived out of order otherwise the correct position $\omega$ is at the tail of the data buffer $B_{\text{data}}$. After the correct position $\omega$ is found, the data and sequence number ($SN$) are extracted from the packet and stored at position $\omega$ in the data buffer $B_{\text{data}}$ and the sequence number buffer $B_{SN}$ as shown in Figure 2. When the received packet is a timing packet, the adjusted arrival time (AdAT) will be stored at position $\omega$ in the time buffer $B_{\text{time}}$ after the timing packet’s timing information is processed as described by the following section.
C.2 The Timing Packet Arrival

A pulse will also be generated by the receiver when the received packet is a timing packet as shown in Figure 2. After which the counter $C_{tc}$ that ticks every reference clock $f_r$ cycle and counter $C_r$ that ticks every $N$ reference clock $f_r$ cycles are latched as shown in Figure 2. The latched value of the counter $C_{tc}$ is $\tau_{1C}$ and counter $C_r$ is $\tau_{NC}$. Both $\tau_{NC}$ and $\tau_{1C}$ values are the arrival time of the timing packet while $\tau_{1C}$ is more precise in measurement than $\tau_{NC}$ because $\tau_{1C}$ has precision of one reference clock cycle while $\tau_{NC}$ has $\lceil\frac{D_{ref}}{N}\rceil$. The number of bits for counter $C_r$ is $b$, i.e., the same size as time indication $(TI)$, while counter $C_{tc}$ is

$$b_{1C} \geq \left\lceil \log_2(\frac{J_{max}}{T_r}) \cdot 2 \right\rceil$$

where $J_{max}$ and $T_r$ are described in Section II.B.

Also the time indication $(TI)$ is extracted from the timing packet in the “TI extraction and EAT calculation” block as shown in Figure 2. The $TI$ is used to calculate the expected arrival time $(EAT)$ by

$$EAT = (TI + \left\lceil \frac{D_{ref}}{N} \right\rceil) \mod 2^b$$

where $D_{ref}$ is determined by equation (1), $b$ is the number of bits reserved for time indication $(TI)$, and $N$ are described by timing insertion.

In addition, the value of counter $C_{tc}$ is latched at every $N$ reference clock $f_r$ cycles (or whenever counter $C_r$ increments its value) as shown in Figure 2. The latched value is $\tau_{1C}$ which is used to determine $\tau_{NC}$’s error in measuring the timing packet’s arrival time.

Once the expected arrival time $(EAT)$, $\tau_{1C}$, $\tau_{NC}$, and $\tau_{NC}$ are calculated by equation (8) and obtained by latching counters $C_{tc}$ and $C_{r}$, respectively, the timing packet’s estimated jitter $J_i$ is calculated in the “estimate jitter” block as shown in Figure 2 by

$$J_i = J_{main} + J_{tail}$$

where subscript $i$ of $J_i$ represents the $i^{th}$ timing packet’s jitter. $J_{main}$ is calculated by

$$J_{main} = N \cdot \left\lceil (EAT - \tau_{NC}) \mod 2^b \right\rceil$$
The network condition might change over time, and the reference clocks at the source and receiver might drift from each other over time. These are classified as drift bias \( \delta_k \) from the \( 1^{st} \) mean network delay at \( k_{th} \) network condition as shown in Figure 3. So, it is important to find the jitter bias \( \mu_k \) at which the reference network delay \( D_{ref} \) is away from the \( k_{th} \) mean network delay at \( k_{th} \) network condition,

\[
\mu_k = \frac{\sum_{i=M_{k-1}+1}^{M_k} J_i}{M_k - (M_{k-1} + 1)}
\]

where \( M_k \) are constants for \( k = 2, ..., \infty \). \( M_k \) could have different values depending on how often the receiver wants to update \( \mu_k \). \( \delta_k \) is then

\[
\delta_k = \mu_1 - \mu_k
\]

where \( J_i \) is the estimated jitter calculated by equation (9), and \( \delta_k \) and \( \mu_1 \) are calculated by equations (14) and (12), respectively. After substituting equation (14) into equation (15), the adjusted jitter \( \tilde{J}_i \) is obtained as

\[
\tilde{J}_i = \left\{ \begin{array}{ll}
J_i - \mu_1, & 1^{st} \text{ network condition} \\
J_i - \mu_k, & k^{th} \text{ network condition}
\end{array} \right.
\]

In order to use the new jitter bias, \( \mu_k \), to calculate the bias adjusted jitter \( \tilde{J}_i \) from equation (15) for the \( k^{th} \) network condition, the number of estimated jitter \( J_i \) calculated by equation (9) used to calculate \( \mu_k \) in equation (13) must be large. This is to avoid the additional error when calculating the bias adjusted jitter \( \tilde{J}_i \) by equation (15) while \( \mu_k \) is approaching its mean value in equation (13).

After the estimated jitter \( J_i \) calculated by equation (9) is adjusted for the bias by the “adjust the jitter bias” block as shown in Figure 2, the adjusted jitter \( \tilde{J}_i \) calculated by equation (15) is used to calculate the adjusted arrival time (AdAT) by

\[
\text{AdAT} = \text{AdAT} + \tilde{J}_i
\]
\[ AdAT = (\tau_{1C} - \tilde{J}_i) \mod 2^{b_{1C}} \]  
(16)

where \( \tau_{1C} \) is obtained by latching the counter \( C_{1C} \) when the timing packet arrives while \( b_{1C} \) is calculated by equation (7). After the adjusted arrival time \( (AdAT) \) is calculated by equation (16), it is stored at position \( \omega \) as described by Section II.C.1 in the time buffer \( B_{time} \).

C.3 To Control Playout

After a packet arrives at the receiver as shown in Figure 2, a waiting time counter is initialized and turned ON. The waiting time \( W_{next} \) for the next packet to arrive is measured in units of reference clock cycles. If each received packet arrives within \( \beta_{wait} \) where \( W_{next} \) is less than \( \beta_{wait} \), i.e., the threshold which indicates an active period, the waiting counter resets itself and \( W_{next} \) is set to zero. Otherwise, a playout signal is sent to the “recover source rate” block as shown in Figure 2.

When the data buffer \( B_{data} \) is empty, the waiting counter resets and turns OFF while \( W_{next} \) and \( S_{\Delta_t} \), i.e., the sum of playout time difference \( \Delta_t \) as calculated by equation (17), are set to zero.

\[ S_{\Delta_t} = \sum_{t=1}^{[T/2]} \Delta_t \]  
(17)

where \( T \) is the number of timing information \( AdAT \), i.e., the adjusted arrival time, in the time buffer \( B_{time} \), and \( \Delta_t \) is calculated by

\[ \Delta_t = (\zeta_{t+1} - \zeta_t) \mod 2^{b_{1C}} \]  
(18)

where \( \zeta_{t+1} \) and \( \zeta_t \) are the adjusted arrival time \( (AdAT) \) in the time buffer \( B_{time} \) which correspond to \( (t+1)^{th} \) and \( t^{th} \) timing data in the data buffer \( B_{data} \), respectively, and \( b_{1C} \) is calculated by equation (7). Throughout the rest of the paper, the subscript \( t \) of \( \Delta_t \) will be dropped to represent the playout time difference between \( \zeta_2 \) and \( \zeta_1 \), i.e., \( AdAT \) corresponding to the \( 2^{nd} \) and \( 1^{st} \) timing data in the data buffer \( B_{data} \).

\[ \Delta = (\zeta_2 - \zeta_1) \mod 2^{b_{1C}} \]  
(19)

If \( S_{\Delta_t} \) is greater than \( \alpha_{jitter} \), i.e., the threshold which indicates that the source might be sending packets at \( SRA \) which is calculated by equation (2) during the active period, then the constant rate indicator (CRI) is set. When CRI is set, a playout signal is sent to the “recover source rate” block as shown in Figure 2. A choice of \( \alpha_{jitter} \) and \( \beta_{wait} \) should have values close to \( J_{max} \), so a change in the network’s condition will not affect the source rate recovery scheme.

C.4 To Recover the Source Rate

After the playout signal is received by the “recover source rate” block as shown in Figure 2, the “recover source rate” block calculates the number of pulses \( \lambda \) which will be generated within the playout time difference \( \Delta \) which is calculated by equation (19).

\[ \lambda = \frac{SN_2 - SN_1}{N} \]  
(20)

where \( SN_1 \) and \( SN_2 \) are the sequence numbers corresponding to \( \zeta_1 \) and \( \zeta_2 \).

Since the source rate recovery scheme uses the received \( 1^{st} \) and \( 2^{nd} \) timing data’s sequence numbers and their adjusted arrival time \( (AdAT) \), i.e., \( \zeta_1 \) and \( \zeta_2 \), to recover the source rate, packet loss does not affect the source rate recovery scheme.

For example, \( \zeta_1 \) and \( \zeta_2 \) point to the first and second timing data in data buffer \( B_{data} \), and \( N \) as described by timing insertion is equal to 5. Also, a timing data is assumed missing. If the first and second timing data have sequence numbers 1 and 11, respectively, and their playout time difference \( \Delta \) is 10 reference clock cycles, two pulses will be generated with equal interval within 10 reference clock cycles. If the timing data is not missing, \( \lambda \) is then equal to one. If \( \Delta \) is 5 reference clock cycles, one pulse will be generated within 5 reference clock cycles. In 10 reference clock cycles, two pulses are generated which is the same as if a timing data is missing. As a result, the source rate recovery scheme is robust to packet loss.

Let \( f_{signal} \) be the generated pulses, and it serves as the reference signal to the phase lock loop (PLL) as shown in Figure 2. The PLL’s output signal \( \hat{f}_s \) which is the reconstructed source clock is

\[ \hat{f}_s = N \cdot f_{signal} \ (Hz) \]  
(21)

where the PLL has a multiplying factor of \( N \). Note that the reconstructed source clock has ON/OFF periods of pulses, because \( f_{signal} \) is generated based on equation (19), equation (20), and playout signal from “control playout” block as shown in Figure 2. As a result, it is different than the reference clock which runs on constant frequency.

The reconstructed source clock \( \hat{f}_s \) is used to service the data buffer \( B_{data} \) as shown in Figure 2. \( \eta \) octets are sent to the application layer with each \( \hat{f}_s \) clock tick since a data is \( \eta \) octets long. Sequence numbers in the sequence number buffer \( B_{SN} \) are also discarded by \( \hat{f}_s \) where each clock tick discards one sequence number.

Whenever a timing data is at the head of the data buffer \( B_{data} \) while data is being sent to the application layer by
the reconstructed source clock \( \hat{f}_s \), the head timing information \( AdAT \) of the time buffer \( B_{time} \) is discarded. A new playout time difference \( \Delta \) and a new number of pulses \( \lambda \) within \( \Delta \) are calculated by equation (19) and (20), respectively, for the new \( \zeta_1 \) and \( \zeta_2 \) pair.

### III. PERFORMANCE EVALUATION

A burst is defined as a duration of data being sent to the receiver. The VBR traffic could be classified as two types, dependent and independent burst traffic. The dependent burst traffic is when the size of the burst varies while the interburst time is constant, e.g., a video stream. The interburst time is the time between two bursts of data. The independent burst traffic is when both the size of the burst and the interburst time vary. Telephony is an example of this. Independent burst traffic is more tolerable to the playout delay of a burst. For example, a voice talkspurt could be delayed while the listener does not find it annoying. On the other hand, a video stream burst is very sensitive to the playout delay. If a frame is delayed, it will cause jitter movements in the video. The CBR traffic is also another example of dependent burst traffic, where the size of burst and interburst time are constant.

The JTS scheme does not assume any specific network model which it could be applied. Only packets’ jitter have an effect on JTS. Packets are encapsulated into ATM cells which are used to deliver the packets to the receiver. As a result, a jitter distribution of ATM cells through an ATM network is used to test JTS. It could be characterized as geometrically distributed when the number of nodes between source and receiver is large and the background traffic is heavy [17]. Background traffic is the traffic which competes for the same resources as the application stream when it is sent from source to receiver.

Simulations have been performed to test the source rate recovery scheme. \( D_{\text{ref}} \) calculated by equation (1), i.e., the reference network delay, is set to zero to allow for the maximum time which the source rate recovery scheme takes to determine the mean network jitter. \( \alpha_{\text{jitter}} \) and \( \beta_{\text{wait}} \) are set to 100 milliseconds. \( J_{\text{max}} \) is also set to 100 milliseconds. For simplicity, no PLL is used during any simulation. The reconstructed source clock is \( N \cdot f_{\text{signal}} \) without going through the PLL. The choice of PLL is avoided here, so only the performance of the source rate recovery scheme is analyzed without taken into account of the loop gain and filtering effects. Also, source packets’ arrival time at the source end is assumed to be synchronous to the reference clock, and timing packet arrives when \( C_s \) counter changes value. So, \( \text{timing insertion error, } \varepsilon_{\text{insertionS}}, \) is zero.

Each ATM cell contains an AAL, AAL1 or AAL2, packet. Note that the JTS is mainly developed for AAL2 to handle real-time VBR traffic, but it is not limited to AAL2 since JTS recovers source rate by maintaining temporal relationships between consecutive packets of a time-sensitive data stream transported across an asynchronous network. As a result, the source traffic is transmitted by AAL packets. Each timing AAL packet contains one byte of time indication (TT), so \( b \) is 8. Note that equation (4) only gives the lower bound for \( b \). All AAL packets also contain 7 bits of sequence number (SN). The application’s
traffic is modeled as ON and OFF. Traffic is sent at \( SRA \) which is calculated by equation (2) when ON, and nothing is sent when otherwise. To test the performance of the new source rate recovery scheme, a performance comparison between SRTS and JTS is done with CBR traffic. Performance analysis of JTS with VBR voice and video traffic is also done. For all simulations, only the 1st network condition is used where \( M_1 \) is set equal to the number of AAL packets generated for the source traffic. Also, the source and receiver are implemented in software. The simulation model is illustrated in Figure 4. The source traffic is packetized into AAL packets which are encapsulated into ATM cells. So, one AAL packet generation is equivalent to one ATM cell generation from the source’s point of view. The ATM cells are sent through a 44.736 Mbps ATM virtual circuit (VC). The background traffic is 155.52 Mbps. The network jitter is modeled as geometrically distributed with probability of success equal to 0.3. The jitter distribution is calculated based on [17] where the source stream is competing with the background traffic. The source stream is running at 44,736,000 bits/sec while the background traffic is at 155,520,000 bits/sec. The performance of SRTS under ideal situations can be found in [14] [13] [12].

A jitter spectrum comparison between SRTS and the new timing scheme (JTS), Figure 5, shows that JTS is about 1 dB higher than SRTS for all frequency when \( N \) equals to 8. \( N \) is the number of AAL packets when there is timing information. Also, JTS lacks SRTS’s harmonic spikes with first harmonic at 64 Hz. As \( N \) gets larger, JTS’s jitter spectrum is much worse than SRTS. On the other hand, if \( N \) gets smaller, JTS’s jitter spectrum is much better than SRTS. The jitter spectrums of JTS and SRTS do not have a high dB role-off, because a simple low pass filter with a single pole at 0.95 is used instead of a PLL in [14]. A first order approximation of the simple low pass filter’s cut-off frequency is 118 Hz. An off-the-shelf PLL has a cutoff frequency of around 100 Hz. In order to minimize the effects of SRTS’s harmonic spikes, a specialized PLL with low cutoff frequency of around 20 Hz is needed. This would result in higher cost to build the receiver. Since SRTS requires 4 bits of timing information for every 8 AAL packets, the best fit in performance and cell formatting to JTS is when \( N \) equals to 8. JTS will use 8 bits of timing information for every 8 AAL packets.

Unlike the SRTS which encodes the source rate by a reference clock which is derived from the network clock and transmits the source rate to the receiver, JTS recovers the source rate by removing AAL packet’s network jitter,
removing clock drift between source and receiver, and detecting ON/OFF periods of the VBR traffic at the receiver. As a result, JTS does not depend on the network clock. The receiver is driven by the reference clock frequency \( f_r \) which is derived from \( SRA \). \( SRA \) is given by equation (2).

### B. JTS with VBR Voice Traffic

A 8 KB/sec voice signal is packetized into AAL2 packets where each AAL2 packet contains 20-octet of voice information. The \( SRA \) is 400 AAL2 packets per second. The voice signal is modeled as ON and OFF. The ON and OFF durations are exponentially distributed with average ON period of 400 ms and average OFF period of 600 ms. The voice signal is transported through a 44.736 Mbps VC. Jitter is geometrically distributed with probability of success equal to 0.3 since the background network traffic is assumed to be 155.52 Mbps.

From Figure 6, the playout time of voice bursts at the receiver is behind the original, because talkspurts at the receiver are allowed to be delayed. By delaying the playout time of the talkspurts, data associated with the talkspurts could arrive in time for playout to minimize jitter within voice bursts. Users will not notice the small initial playout delay of bursts, but they are aware of the jitter within the bursts. The probability density function of reconstructed source rate’s error, reconstructed source rate minus actual source rate, is plotted in Figure 7. Since \( \sigma_{\text{jitter}} \) is set to 100 milliseconds, error above 40 AAL2 packets per second are due to playout delay of each talkspurt. The average error from the actual source rate is -0.9307 AAL2 packet/second for \( N \) equals to 8. The average value is negative, because the reconstructed source rate is trailing the actual source rate on average by 0.9307 AAL2 packet/second.

### C. JTS with VBR video Traffic

A generalized ON/OFF video traffic which bursts every 63 ms has a random active duration uniformly distributed from 10 to 40 ms. When ON, the video traffic is sent at 8 KB/sec. A 8 KB/sec video rate is used, so a comparison between VBR voice and VBR video could be done. The purpose is to see the effect of the VBR video type traffic on the new source rate recovery scheme. The video traffic is packetized into 20-octet AAL2 packets. It is sent at 400 AAL2 packets per second when ON. Like VBR voice, the video traffic is transported through a 44.736 Mbps VC with 155.52 Mbps background network traffic. The jitter distribution is also geometrically distributed with probability of success equal to 0.3.

A plot of the reconstructed source rate versus the actual source rate in Figure 8 shows that the reconstructed source rate is following the actual source rate. The probability density function of the error from the actual source rate is in Figure 9. The average error is -2.5509 AAL2 packets/second for \( N \) equals to 8. As \( N \) increases, the average error also increases. When \( N \) decreases, the average error also decreases. The average error for VBR video traffic is higher than VBR voice traffic, because VBR video traffic bursts periodically and expects to be played out periodically. On the other hand, the playout time of each voice talkspurt could be relaxed without distorting the perceptual quality of the voice signal. Comparing the pdf of VBR voice, Figure 7, with VBR video, Figure 9, VBR video traffic does not have error greater than 40 AAL2 packets per second. This is consistent with the fact that VBR video is a dependent VBR traffic where each burst must be
played out periodically and could not be delayed. Hence, error greater than 40 AAL2 packets per second does not exist for VBR video traffic.

IV. CONCLUSION

JTS is able to reconstruct CBR and VBR traffic. JTS has a higher noise level than SRTS when \( N \) is equal to 8, but it does not have SRTS’s harmonic spikes. When \( N \) decreases, the noise performance is better than SRTS. JTS is able to reconstruct VBR voice and video traffic with the reconstructed source rate trailing the actual source rate. On average the reconstructed source rate trails the actual source rate by 0.9307 and 2.5509 AAL2 packets/second for VBR voice and VBR video traffic, respectively, when \( N \) is set to 8.

Note that the maximum jitter which any packet received for all simulations is below \( \alpha_{\text{jitter}} \) and \( \beta_{\text{wait}} \), so unstable situation did not occur. When significant amount of packets experience jitter which are greater than \( \alpha_{\text{jitter}} \) and \( \beta_{\text{wait}} \), it will cause unstable situation in source synchronization such as buffer underflow and significant packet drop. Hence, \( \alpha_{\text{jitter}} \) and \( \beta_{\text{wait}} \) must be chosen with care. The size of AAL packets and the choice of \( N \) also affect the reconstructed source rate’s performance. They should be selected to satisfy the traffic’s quality of service (QoS) requirements. In summary, QoS requirements could be adjusted by modifying \( \beta_{\text{wait}}, \alpha_{\text{jitter}}, \text{size of the AAL packets and} \: N \).

REFERENCES


