Intelligent Voice Smoother for Silence-Suppressed Voice over Internet

Po L. Tien and Maria C. Yuang

***Abstract*—When transporting voice data with silence suppression over the Internet, the problem of jitter introduced from the network often renders the speech unintelligible. It is thus indispensable to offer intramedia synchronization to remove jitter while retaining minimal playout delay (PD). In this paper, we propose a neural network (NN)-based intravoice synchronization mechanism, called the intelligent voice smoother (IVoS). IVoS is composed of three components: 1) the smoother buffer; 2) the NN traffic predictor; and 3) the constant bit rate (CBR) enforcer. Newly arriving frames, assumed to follow a generic Markov modulated Bernoulli process (MMBP), are queued in the smoother buffer. The NN traffic predictor employs an online-trained back propagation NN (BPNN) to predict three traffic characteristics of every newly encountered talkspurt period. Based on the predicted characteristics, the CBR enforcer derives an adaptive buffering delay (ABD) by means of a near-optimal simple closed-form formula. It then imposes the delay on the playout of the first frame in the talkspurt period. The CBR enforcer in turn regulates CBR-based departures for the remaining frames of the talkspurt, aiming at assuring minimal mean and variance of distortion of talkspurts (DOT) and mean PD. Simulation results reveal that, compared to three other playout approaches, IVoS achieves superior playout, yielding negligible DOT and PD, irrespective of traffic variation.**

***Index Terms*—Back propagation neural network (BPNN), best effort service, constant bit rate (CBR), Internet, intramedia synchronization, jitter, Markov modulated Bernoulli process (MMBP), multimedia communications, silence suppression.**

I. Introduction

THE RECENT evolution in high-speed communication technology enables the development of distributed multimedia applications combining a variety of media data such as text, audio, graphics, images, voice, and full-motion video. These applications often require stringent quality of service (QoS) guarantees, such as bounded delay and jitter [1]. Moreover, the multicast backbone (MBone)/Internet [2]–[4] has been widely deployed to support diverse multicasting traffic for businesses and individuals. Its current transport protocols, however, were originally designed to offer the best effort service without performance guarantees. Consequently, the proliferation of these multimedia applications has imposed ever more strain on the MBone/Internet. In particular, in order to make more efficient use of scarce bandwidth, voice data has been considered to be transported via variable bit rate (VBR), using speech activity detection [5], [6]. As a result, the problem of unbounded jitter introduced from the network often renders the speech unacceptable or even unintelligible. It thus becomes essential to offer intramedia synchronization retaining distinctive QoS guarantees.

Essentially, voice applications can be broadly classified as either interactive, such as voice conversation of teleconferencing, or unidirectional (unicast or multicast), such as voice distribution services [7]. Serving dissimilar purposes, these two classes of applications differ in the PD requirement and the tolerance of playout impairments [8]. Two playout impairments considered include distortion of talkspurts (DOT) (or speech clipping) and distortion of silence (or variable speech burst delay). While the former is invariably significant, the latter is often imperceptible for most applications. Owing to the real-time nature, interactive voice applications are more sensitive to PD than playout impairments. On the contrary, unidirectional applications are rather susceptible to playout impairments, subject to reasonable PD. The main objective of the paper is to propose an adaptive playout mechanism satisfying any given set of QoS requirements in terms of PD and DOT.

Several existing intramedia synchronization methods which perform at end systems exhibit various performance merits. They can be categorized as: *static delay-based*, *dynamic feedback-based*, and *dynamic delay-based*. Static delay-based methods preserve playout continuity by buffering massive packets at receiving end systems [9] or delaying the playout time of the first packet received [4], [10], [11]. These methods have been shown to be feasible, but at the expense of a drastic increase in PD. On the other hand, dynamic feedback-based methods [12], [13] perform intramedia synchronization through adjusting the source generation rate by means of sending feedback from receiving end systems. While these methods are effective, they are unviable for most live-source applications. Unlike the two classes of methods described above, dynamic delay-based methods [14], [15] employ dynamic playout rates in accordance with a computed window (or threshold) which can be intelligently predicted in real time or analytically computed in advance. These methods have been shown to be viable, particularly for video streams which are dissimilar to voice streams by nature [7]. Two other dynamic delay-based methods [16], [17] attempted to preserve playout continuity via adaptive buffering of frames having been time-stamped at the sources. The methods are indeed feasible, but at the expense of drastic processing and framing overhead from frequent time stamping.

Manuscript received December 1, 1997; revised May 17, 1998. This work was supported in part by the Institute for Information Industry (III) under Contract 86-0040. This paper was presented in part at INFOCOM, 1998.

The authors are with the Department of Computer Science and Information Engineering, National Chiao Tung University, Taiwan, R.O.C.

Publisher Item Identifier S 0733-8716(99)00006-2.

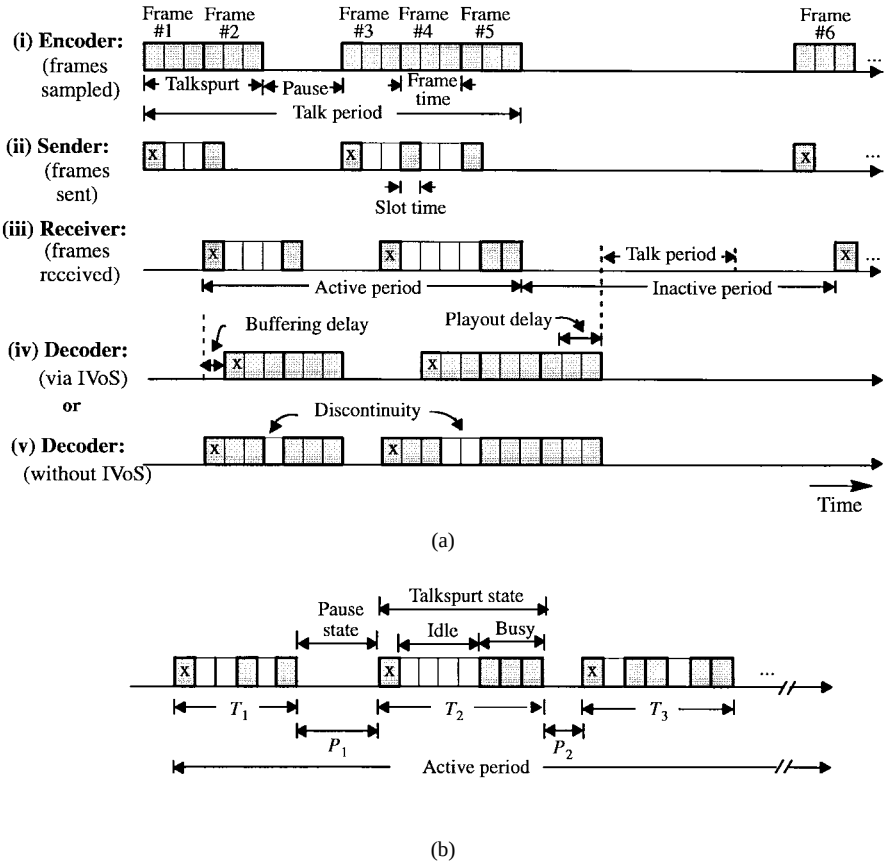


Fig. 1. Concept of IVoS where X: first frame of a talkspurt; P_i: pause i; T_i: talkspurt i. (a) An end-to-end voice flow scenario. (b) IVoS states.

In this paper, we propose an NN-based intravoice synchronization mechanism, called the intelligent voice smoother (IVoS), operating at the application layer of receiving end systems. IVoS is composed of three components: 1) the smoother buffer; 2) the neural network (NN) traffic predictor; and 3) the constant bit rate (CBR) enforcer. The inbound traffic to IVoS is modeled as a generic discrete-time Markov modulated Bernoulli process (MMBP) with unknown and varying probabilistic parameters.

Initially, newly arriving frames are queued in the smoother buffer. The NN traffic predictor employs an online-trained back propagation NN (BPNN) to predict three characteristics (talkspurt length, frame count, and last burst length) of the upcoming talkspurt period. Based on the predicted characteristics, the CBR enforcer imposes an adaptive buffering delay (ABD) derived from a near-optimal closed-form formula. The CBR enforcer, in turn, regulates CBR-based departures of frames within this talkspurt period, aiming at assuring minimal mean and variance of DOT and mean PD. Simulation results reveal that, compared to three other playout approaches, IVoS achieves superior playout, yielding negligible DOT and PD, irrespective of traffic variation.

The remainder of this paper is organized as follows. Section II presents the main concept and the inbound traffic model of IVoS. Section III details the system architecture, including its NN traffic predictor and the CBR enforcer. Section IV then demonstrates performance comparisons between IVoS and existing playout approaches. Finally, concluding remarks are given in Section V.

II. IVoS—Concept and Model

A. Concept

Generally, voice data are sampled and encoded as fixed-size frames. These frames with silence suppression are in turn sent over the MBONE/Internet [2]. Upon receiving frames which are assumed to arrive in accordance with a generic MMBP (described later) with unknown probabilistic parameters, IVoS determines the departure time at which frames are transferred from the IVoS smoother buffer to the decoder buffer from which frames are playbacked. An end-to-end flow scenario is illustrated in the time–space diagram shown in Fig. 1(a). In the figure, the conversation between the sender and receiver (with IVoS) is conducted in an alternating manner between an active period and an inactive period. IVoS is in an active period during receiving frames; otherwise it is in an inactive period. Furthermore, being in an active period, IVoS alternates between the *talkspurt* state, with frames intermittently received (during the *busy state*), and the *pause* state during which no frames appear. Moreover, the first frame in any talkspurt is tagged (marked $\times$ as shown in the figure) before being transmitted.

Accordingly, the goal of IVoS is the enforcement of CBR playout during the active period by dynamically adjusting

TABLE I
VARIABLES USED THROUGHOUT THE PAPER

Variable	Definition
MFR	Mean frame rate (frames/slot)
MBL	Mean burst length (or burstiness [7], p. 79)
$\tilde{t}$	Random variable of the talkspurt length
$\tilde{f}$	Random variable of the number of frames within a talkspurt
$\tilde{b}$	Random variable of the last burst length of a talkspurt
T_i	Actual length of talkspurt i
F_i	Actual number of frames in talkspurt i
B_i	Actual length of the last burst of talkspurt i
$\hat{T}_i$	Predicted length of talkspurt i
$\hat{F}_i$	Predicted number of frames in talkspurt i
$\hat{B}_i$	Predicted length of the last burst of talkspurt i
$A\tilde{B}D$	Random variable of (adaptive) buffering delay of a talkspurt
$D\tilde{O}T$	Random variable of Distortion of Talkspurt
$P\tilde{D}$	Random variable of Playout Delay
ABD_i	Adaptive buffering delay for talkspurt i
DOT_i	Distortion of Talkspurt for talkspurt i
PD_i	Playout Delay for talkspurt i
$\mathcal{F}$	Frame–to–slot ratio
θ	Locality parameter

the duration of pauses in an effort to compensate for jitter within talkspurts. For ease of illustration, and without loss of generality, we assume that the system remains in one active period for the entire duration of the connection throughout the rest of the paper. This is often the case for unidirection-based multimedia applications, such as distance learning.

The rationale of how IVoS achieves the aforementioned goal is also shown in Fig. 1(a). The time axis in IVoS is slottized by the processing of a single frame from the adjacent lower layer, i.e., the transport layer. We assume that, disregarding the framing overhead, voice frames are generated (playbacked) at the encoder (decoder) of the sending (receiving) end system at a rate of one-third of the processing rate of the transport layer [14]. Define $\mathcal{F}$ as the ratio of the generation or playout of a frame, referred to as the *frame time*, to the processing of a single frame at the transport layer, referred to as the *slot time*, i.e., $\mathcal{F} =$ frame time/slot time. For the example given in Fig. 1, $\mathcal{F} = 3$. For ease of description, all variables which are used throughout the paper are summarize in Table I.

Frames are finally received at IVoS at the receiving end system. Ideally, in a jitter-free network the interdeparture times of frames from the sender's application are the same as the interarrival times of frames at IVoS. In this case, frames are playbacked intelligibly at the maximum rate, i.e., one frame per every three time slots during talkspurts. Unfortunately, in reality, owing to delay jitter induced in the network, different frames yield different end-to-end delays which result in speech unintelligibility. Denote t_i^s and t_i^p as the sample and playout time of frame i, respectively. In addition, denote D_i as the end-to-end transfer delay of frame i. Accordingly, playout

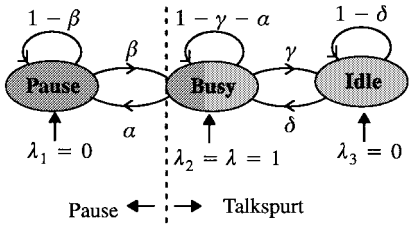


Fig. 2. Inbound traffic model.

discontinuity is quantified by DOT where the DOT for talkspurt j denoted as DOT_j is defined as

$$DOT_j = \frac{\sum_{i=R(j)+1}^{L(j)} \left[(t_i^p - t_{i-1}^p) - (t_i^s - t_{i-1}^s) \right]}{F_j - 1} = \frac{\sum_{i=R(j)+1}^{L(j)} \left[D_i - D_{i-1} \right]}{F_j - 1} \quad (1)$$

where $R(j)$ and $L(j)$ are the ordinal numbers of the first and last frame in talkspurt j and F_j is the total number of frames in talkspurt j. Moreover, playout discontinuity can be reduced at the expense of an increase in PD. Let PD_i denote the PD of talkspurt i defined as the elapsed time between the fastest possible departure and real departure of the last frame in talkspurt i.

Consequently, IVoS aims at achieving minimal $E[\tilde{DOT}]$ and $V[\tilde{DOT}]$ (zero in the case of CBR playout) while sustaining minimal $E[\tilde{PD}]$. It is worth noting that we do not consider frame loss probability, that is, the dropping of frames due to late arrival. Although discard of late frames assures delay-free playout, the price paid is deteriorating speech clipping [8]. Two issues have been considered in the design of IVoS: 1) how and what characteristics of future traffic to be predicated and 2) how to determine an ABD imposed on each talkspurt aimed at achieving a quasi-CBR playout during talkspurts. Before proposing solutions to these two issues, we first present the inbound traffic model and the architecture of IVoS in the next subsections.

B. Inbound Traffic Model

The inbound traffic to IVoS is modeled by a generic discrete-time MMBP [5], [6], [18] as shown in Fig. 2. The process alternates between the pause state and the talkspurt state. Within the talkspurt state, the process switches between the busy state, during which frames arrive in a burst, and the idle state during which no frame appears. The transition probabilities between states are given in the figure. For example, α defines the probability of switching from the busy to the pause state and β defines the opposite probability. The lengths of all three states are assumed to be geometrically distributed. Moreover, during the busy state, one frame always arrives (with probability $\lambda = 1$) per slot time and no frame is generated during the idle and the pause states.

It is worth noting that VBR voice sources have been modeled by such a three-state MMBP [6] with parameters matched to the nature of the voice applications. It has been shown [18] that multiplexing of MMBP's can be approximated by another MMBP with different transitional probabilities. Therefore, after the traffic has been multiplexed and demultiplexed over a network at the end system in IVoS, we adopt a *generic* MMBP with unknown and varying transitional probabilities which will be in turn predicted by the NN traffic predictor.

The steady-state probabilities of being at the three states, denoted as $\Pi_{\text{pause}}, \Pi_{\text{busy}}$, and Π_{idle} can be computed using $\Pi = \Pi P$ [19] where $\Pi = [\Pi_{\text{pause}}, \Pi_{\text{busy}}, \Pi_{\text{idle}}]$ and P is the state transition probability matrix of the MMBP. As a result

$$\Pi_{\text{pause}} = \frac{\alpha\delta}{\delta\beta + \gamma\beta + \alpha\delta}; \quad \Pi_{\text{busy}} = \frac{\delta\beta}{\delta\beta + \gamma\beta + \alpha\delta}; \quad \text{and}$$
$$\Pi_{\text{idle}} = \frac{\gamma\beta}{\delta\beta + \gamma\beta + \alpha\delta}. \tag{2}$$

Moreover, the mean frame rate (MFR) and mean burst length (MBL) can be directly expressed as functions of α, β, γ, and δ

$$MFR = \Pi_{\text{busy}} \times \lambda = \frac{\delta\beta \times \lambda}{\delta\beta + \gamma\beta + \alpha\delta} = \frac{\delta\beta}{\delta\beta + \gamma\beta + \alpha\beta} \quad \text{and}$$
$$MBL = \frac{1}{\alpha + \gamma}. \tag{3}$$

Next, it can be perceived that the ABD of each talkspurt is largely dependent on three variables: the talkspurt length ($\tilde{t}$), the frame count ($\tilde{f}$), and the last burst length ($\tilde{b}$) of talkspurts. We now examine the probability mass functions (pmf's) of these three random variables.

First, let $\tilde{c}$ denote the total number of cycles, from the busy to the idle and back to the busy state, exhibited in a talkspurt. The pmf of talkspurt length $\tilde{t}$ becomes

$$P[\tilde{t} = t] = \sum_{c=0}^{\lfloor t-1/2 \rfloor} P[\tilde{t} = t | \tilde{c} = c], \qquad t > 0 \tag{4}$$

with the conditional pmf given as shown in (5) at the bottom of this page, where $R_c^t = t - 2c - 1$. Second, the pmf of frame count $\tilde{f}$ can be expressed as

$$P[\tilde{f} = f] = \sum_{t=f}^{\infty} P[\tilde{t} = t, \tilde{f} = f], \qquad f > 0 \tag{6}$$

in which the joint pmf can be derived as shown in (7) at the bottom of this page. Finally, the pmf of the last burst length $\tilde{b}$ can be obtained from the joint pmf of the three random variables. That is

$$p[\tilde{b} = b] = \sum_{t=b}^{\infty} \sum_{f=b}^{t} P[\tilde{t} = t, \tilde{f} = f, \tilde{b} = b] \tag{8}$$

in which the joint pmf can be derived based on the same notion of the cycle introduced above, as shown in (9) at the bottom of this page. We notice that after the analytical computation of (8) and (9) the last burst length $\tilde{b}$, as was perceived, happens to be geometrically distributed. That is $P[\tilde{b} = b] = (1 - \alpha - \gamma)^{b-1}(\alpha + \gamma)$.

For ease of explanation, in Table II we summarize 81 types of traffic arrivals (nine different MFR's and MBL's) which are used throughout the rest of the paper. For all traffic arrivals we assume that due to the CBR nature ($1/\mathcal{F}$ frames per every slot time) the MFR during talkspurts is given as $1/\mathcal{F}$, which is equal to one-third in all cases. This, assuming a silence suppression rate of 40%, results in an MFR for the entire talk of $1/3 \times (1 - 40\%) = 1/5$. Moreover, notice that we employ the same α and β for all traffic types due to their being independent of playout intelligibility.

III. IVoS System Architecture

IVoS is composed of three major components (see Fig. 3): 1) the smoother buffer; 2) the NN traffic predictor; and 3) the CBR enforcer. Newly arriving frames are first placed in the smoother buffer in a first-come-first-served (FCFS) fashion. Each time, the reception of a marked frame, which corresponds to the initiation of a new talkspurt, triggers the NN traffic predictor to perform the prediction of three traffic

$$P[\tilde{t} = t | \tilde{c} = c] = \begin{cases} \alpha(1 - \alpha - \gamma)^{t-1}, & \text{if } c = 0; \\ \alpha(\gamma\delta)^c \sum_{i=0}^{R_c^t} \left[\binom{R_c^t - i + c}{c} \binom{i + c - 1}{c - 1} (1 - \alpha - \gamma)^{R_c^t - i} (1 - \delta)^i \right], & \text{if } c \geq 1 \end{cases} \tag{5}$$

$$P[\tilde{t} = t, \tilde{f} = f] = \begin{cases} \alpha(1 - \gamma - \alpha)^{t-1}, & \text{if } t = f; \\ \alpha \sum_{c=1}^{t-f} \left[\binom{f - 1}{c} \binom{t - f - 1}{c - 1} (1 - \alpha - \gamma)^{f-c-1} (1 - \delta)^{t-f-c} (\gamma\delta)^c \right], & \text{if } t > f \end{cases} \tag{7}$$

$$P[\tilde{t} = t, \tilde{f} = f, \tilde{b} = b] = \begin{cases} \alpha(1 - \alpha - \gamma)^{t-1}, & \text{if } t = f = b; \\ \alpha \sum_{c=1}^{t-f} \left[\binom{f - b - 1}{c - 1} \binom{t - f - 1}{c - 1} (1 - \alpha - \gamma)^{f-c-1} (1 - \delta)^{t-f-c} (\gamma\delta)^c \right], & \text{if } t > f > b \end{cases} \tag{9}$$

TABLE II
INBOUND TRAFFIC ARRIVALS USED THROUGHOUT THE PAPER

Traffic	Sending End		Receiving End				
	Sil. Supp.	Resulted	M M B P				*MBL*
Type	Rate (%)	*MFR*	α	β	δ	γ	
A_1,A_2,...,A_9	25	1/3 × 75%=0.25	0.1	0.2	$\frac{(MFR)\gamma\beta}{\beta-(MFR)\beta-(MFR)\alpha}$	$\frac{1}{MBL}-\alpha$	1,2,...,9
B_1,B_2,...,B_9	32.5	1/3 × 67.5%=0.225	0.1	0.2			1,2,...,9
C_1,C_2,...,C_9	40	1/3 × 60%=0.2	0.1	0.2			1,2,...,9
D_1,D_2,...,D_9	47.5	1/3 × 52.5%=0.175	0.1	0.2			1,2,...,9
E_1,E_2,...,E_9	55	1/3 × 45%=0.15	0.1	0.2			1,2,...,9
F_1,F_2,...,F_9	62.5	1/3 × 37.5%=0.125	0.1	0.2			1,2,...,9
G_1,G_2,...,G_9	70	1/3 × 30%=0.1	0.1	0.2			1,2,...,9
H_1,H_2,...,H_9	77.5	1/3 × 22.5%=0.075	0.1	0.2			1,2,...,9
I_1,I_2,...,I_9	85	1/3 × 15.5%=0.05	0.1	0.2			1,2,...,9

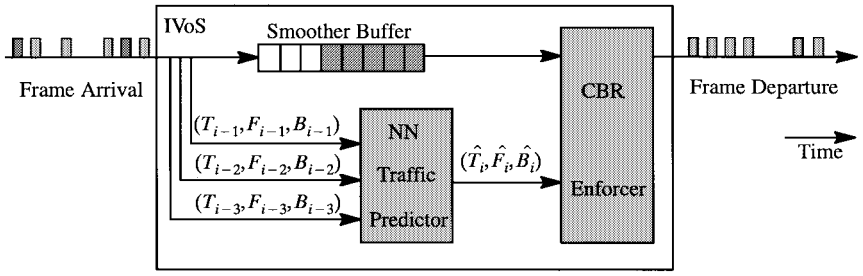


Fig. 3. System architecture of IVoS.

characteristics of the upcoming talkspurt i. They are the talkspurt length (T_i), frame count (F_i), and the last burst length (B_i). Based on the three predicted characteristics, the CBR enforcer then determines a dynamic delay to be imposed on the first frame, and regulates instant playout for all subsequent frames of the talkspurt. The same process repeats for the next talkspurt until the end of the talk. In the following sections the predictor and enforcer are described in detail.

A. NN Traffic Predictor

Substantially, we have discovered several strengths of NN's with respect to the training of traffic distributions. On the whole, while the offline learning of traffic distributions has been shown to be profoundly viable, the online training [20], [21] of highly bursty traffic is more challenging. The NN traffic predictor of IVoS employs an online-trained BPNN to predict T_i (talkspurt length), F_i (frame count), and B_i (last burst length) of the upcoming talkspurt i, based on the same characteristics taken from the past three talkspurt periods $i-1, i-2, i-3$. More explicitly, the NN is modeled, as shown in Fig. 4, as

$$\hat{M}(t_c, FTS) = NN_f[M(t_c, [PTS]_n), WG]. \quad (10)$$

In the equation, NN_f denotes the NN function and WG represents the weight matrix of the links between neurons. $M(t_c, [\mathrm{PTS}]_n)$ denotes the n sets of input vectors (T_i, F_i and $B_i, i = 1$ to n), representing the three characteristics respectively taken from n past talkspurt periods. $\hat{M}(t_c,\ FTS)$ denotes the output vector, representing the three traffic characteristics over the next talkspurt period. At the beginning of every FTS, in addition to predicting the three parameters of the future talkspurt as described above, the NN also performs

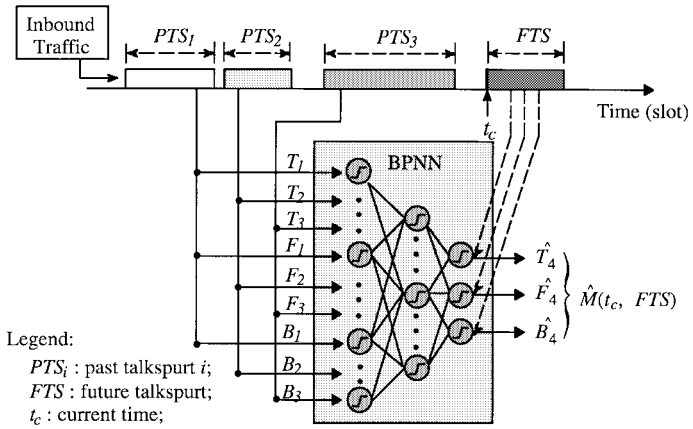


Fig. 4. NN traffic predictor.

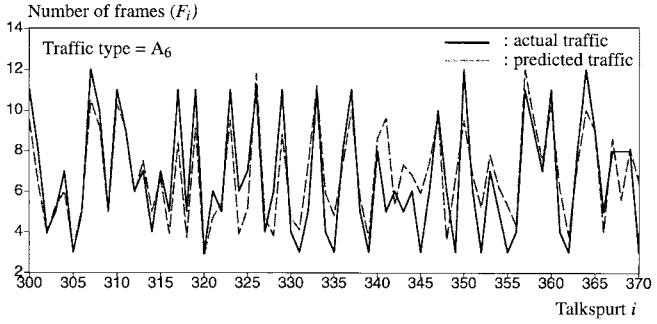
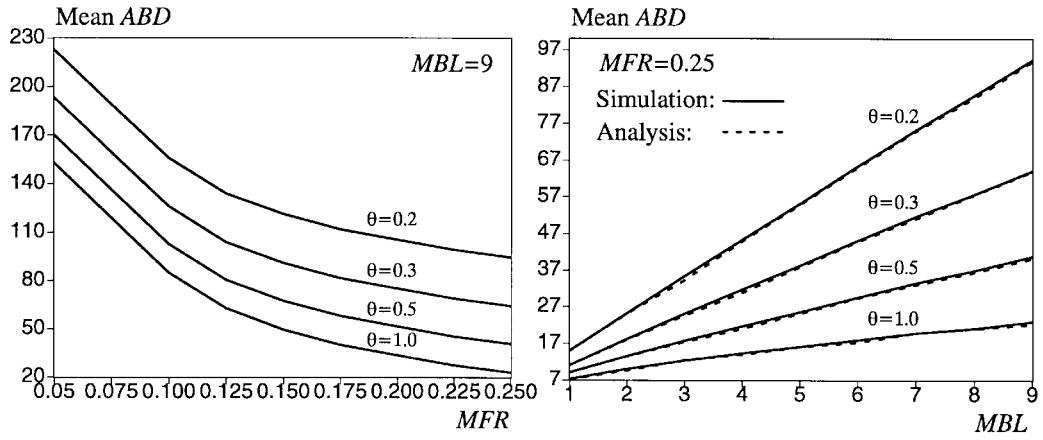


Fig. 5. Comparison of actual and predicted frame counts (F_i).



(a) (b)

Fig. 6. ABD under various loads and burstiness of arrivals. (a) Load effect on ABD. (b) Burstiness effect on ABD.

TABLE III
PERFORMANCE COMPARISONS BETWEEN OPTIMAL PLAYOUTS AND IVOS ($MBL = 2, \theta = 0.9$)

Performance Metrics \ Traffic Type		A_2	B_2	C_2	D_2	E_2	F_2	G_2	H_2	I_2
Mean *DOT*	Optimal playout (Pure CBR playout)	0	0	0	0	0	0	0	0	0
	IVoS	0.0545	0.0566	0.0544	0.048	0.0422	0.0336	0.0282	0.022	0.0125
Variance of *DOT*	Optimal playout (Pure CBR playout)	0	0	0	0	0	0	0	0	0
	IVoS	0.022	0.025	0.024	0.022	0.020	0.016	0.015	0.012	0.006
Mean *PD*	Optimal playout (Instant playout)	6.8	5.38	4.52	3.86	3.42	3.05	2.79	2.62	2.42
	IVoS (Deficiency)	6.85 (0.7%)	5.44 (1.1%)	4.59 (1.5%)	3.93 (1.8%)	3.48 (1.8%)	3.11 (2.0%)	2.86 (2.5%)	2.69 (2.7%)	2.49 (2.8%)

the back-propagation training operation by updating the WG based on the three traffic measurements over the talkspurt which has just passed.

In Fig. 5, we draw a comparison between the actual and predicted frame counts (F_i) in each talkspurt i, assuming an MMBP arrival of type A_6 listed in Table II. In this experiment, we employed a three-layer NN with five hidden nodes and a learning constant of 1.25. We have observed that the variation of this characteristic can mostly be captured by the predictor. The predictions of the other two characteristics, i.e., T_i and B_i also exhibit compatible results. For greater details, readers are referred to an early published paper [14].

B. CBR Enforcer

Based on the three predicted characteristics, the CBR enforcer mainly determines an ABD to be imposed on the first (marked) frame initiating the talkspurt. Let $\tilde{ABD}$ and ABD_i denote the random variable of ABD and ABD for talkspurt i, respectively. The enforcer aims at the achievement of quasi-CBR departures for all subsequent frames belonging to the same talkspurt. In principle, the buffering delay of a talkspurt should be large enough in compensation for the total number of time slots lacking the frame playout in the talkspurt.

Should the three traffic characteristics, T_i, F_i, B_i of talkspurt i be known, let us consider the best case, i.e., incurring the minimum ABD. In this case, the remaining frames (frames not belonging to the last burst) have arrived back to back at the beginning of the talkspurt. First, the entire playout duration, defined as the interval from the beginning of the talkspurt to the end of playout of the last frame, is clearly the sum of the length of talkspurt i (T_i) and the additional duration required to playback frames of the last burst $[B_i \times (\mathcal{F} - 1)]$. Moreover, the total elapsed time required for CBR playout, subject to a total number of F_i frames in talkspurt i is $F_i \times \mathcal{F}$. Therefore, the ABD_i can be given as the difference of the entire playout duration and the elapsed time for CBR playout, i.e., $[T_i + B_i \times (\mathcal{F} - 1)] - F_i \times \mathcal{F}$.

Now, consider the real case in which the remaining frames have arrived at different locations between the beginning and the last burst. Taking the localities of the remaining frames into account, we introduce a so-called locality parameter, denoted as θ $(0 < \theta \leq 1)$ Thus, we attain the theoretical buffering delay denoted as ABD_i^{TH} yielding CBR playout as

$$ABD_i^{TH} = \left[T_i + \left\lfloor \frac{1}{\theta} \times B_i \times (\mathcal{F} - 1) \right\rfloor - F_i \times \mathcal{F}\right]^+. \quad (11)$$

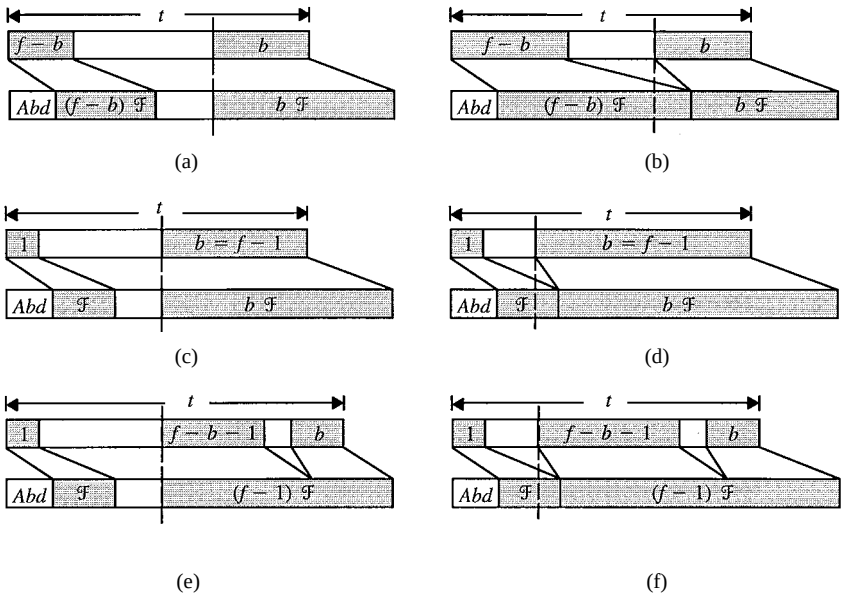


Fig. 7. Where t: a constant of $\bar{t}$ in (15); f: a constant of $\bar{f}$ in (15); b: a constant of $\bar{b}$ in (15). Abd: $\left[t + [\frac{1}{\theta} \times b \times (\mathcal{F} - 1] - f \times \mathcal{F}\right]^+$. Six cases for deriving upper and lower bounds of DOT and PD. (a) Best case 1: $t - b - Abd \geq (f - b)\mathcal{F}$. (b) Best case 2: $t - b - Abd < (f - b)\mathcal{F}$. (c) Worst case 1: $t - b - Abd \geq \mathcal{F}$. (d) Worst case 2: $t - b - Abd < \mathcal{F}$. (e) Worst case 3: $t - f - Abd \geq \mathcal{F}$. (f) Worst case 4: $t - f - Abd < \mathcal{F}$.

TABLE IV
DOT AND PD BOUNDS FOR THE DETERMINATION OF θ IN IVoS

QOS preference	Condition			Lower/Upper Bound
DOT	Best Case (1 & 2)			$DOT_{LB} = \left[\frac{t\text{–}Abd\text{–}b\text{–}(f\text{–}b)\mathfrak{F}}{f}\right]^+$
	Worst Case (1 & 2)	$b = f - 1$		$DOT_{UB} = \left[\frac{t\text{–}Abd\text{–}\mathfrak{F}\text{–}(f\text{–}1)}{f}\right]^+$
	Worst Case (3 & 4)	$b < f - 1$		$DOT_{UB} = \left[\frac{t\text{–}Abd\text{–}\mathfrak{F}\text{–}f}{f}\right]^+$
PD	Best Case (1)	$t\text{–}b\text{–}Abd \geq (f\text{–}b)\mathfrak{F}$		$PD_{LB} = (b - 1)(\mathfrak{F} - 1)$
	Best Case (2)	$t\text{–}b\text{–}Abd < (f\text{–}b)\mathfrak{F}$		$PD_{LB} = (b - 1)(\mathfrak{F} - 1) + [(f\text{–}b)\mathfrak{F}\text{–}(t\text{–}b\text{–}Abd)]$
	Worst Case (1)	$b = f - 1$	$t\text{–}b\text{–}Abd \geq \mathfrak{F}$	$PD_{UB} = (b - 1)(\mathfrak{F} - 1)$
	Worst Case (2)	$b = f - 1$	$t\text{–}b\text{–}Abd < \mathfrak{F}$	$PD_{UB} = (b - 1)(\mathfrak{F} - 1) + [\mathfrak{F}\text{–}(t\text{–}b\text{–}Abd)]$
	Worst Case (3)	$b < f - 1$	$t\text{–}f\text{–}Abd \geq \mathfrak{F}$	$PD_{UB} = (f - 1)\mathfrak{F} - f - (\mathfrak{F} - 1)$
	Worst Case (4)	$b < f - 1$	$t\text{–}f\text{–}Abd < \mathfrak{F}$	$PD_{UB} = [(f - 1)\mathfrak{F} - f - (\mathfrak{F} - 1)] + [\mathfrak{F}\text{–}(t\text{–}f\text{–}Abd)]$

It is worth noting that θ, being unity, corresponds to the best case given previously. The smaller the θ, the later and more widely the spread frames have arrived.

Furthermore, since T_i, F_i, and B_i are not known in advance, replacing them by $\hat{T}_i, \hat{F}_i$, and $\hat{B}_i$ as predicted by the NN predictor, we can formulate the actual buffering delay as

$$ABD_i = \left[\hat{T}_i + \left\lfloor \frac{1}{\theta} \times \hat{B}_i \times (\mathcal{F} - 1) \right\rfloor - \hat{F}_i \times \mathcal{F}\right]^+. \quad (12)$$

With such a delay imposed, the CBR enforcer then regulates the departure of frames in a rate of $1/\mathcal{F}$ frames/slot until the next marked frame initiating the subsequent talkspurt has been encountered. The process repeats until the end of the talk.

To formally examine the behavior of $A\tilde{B}D$ in relation to two traffic parameters, MFR and MBL, we formulate the pmf of $A\tilde{B}D$ in terms of $\tilde{t}, \tilde{f}$, and $\tilde{b}$ as

$$P\left[A\tilde{B}D = y\right] = \sum_{t=1}^{\infty} \sum_{f=1}^{t} P\left[A\tilde{B}D = y | \tilde{t} = t, \tilde{f} = f\right] P\left[\tilde{t} = t, \tilde{f} = f\right]$$

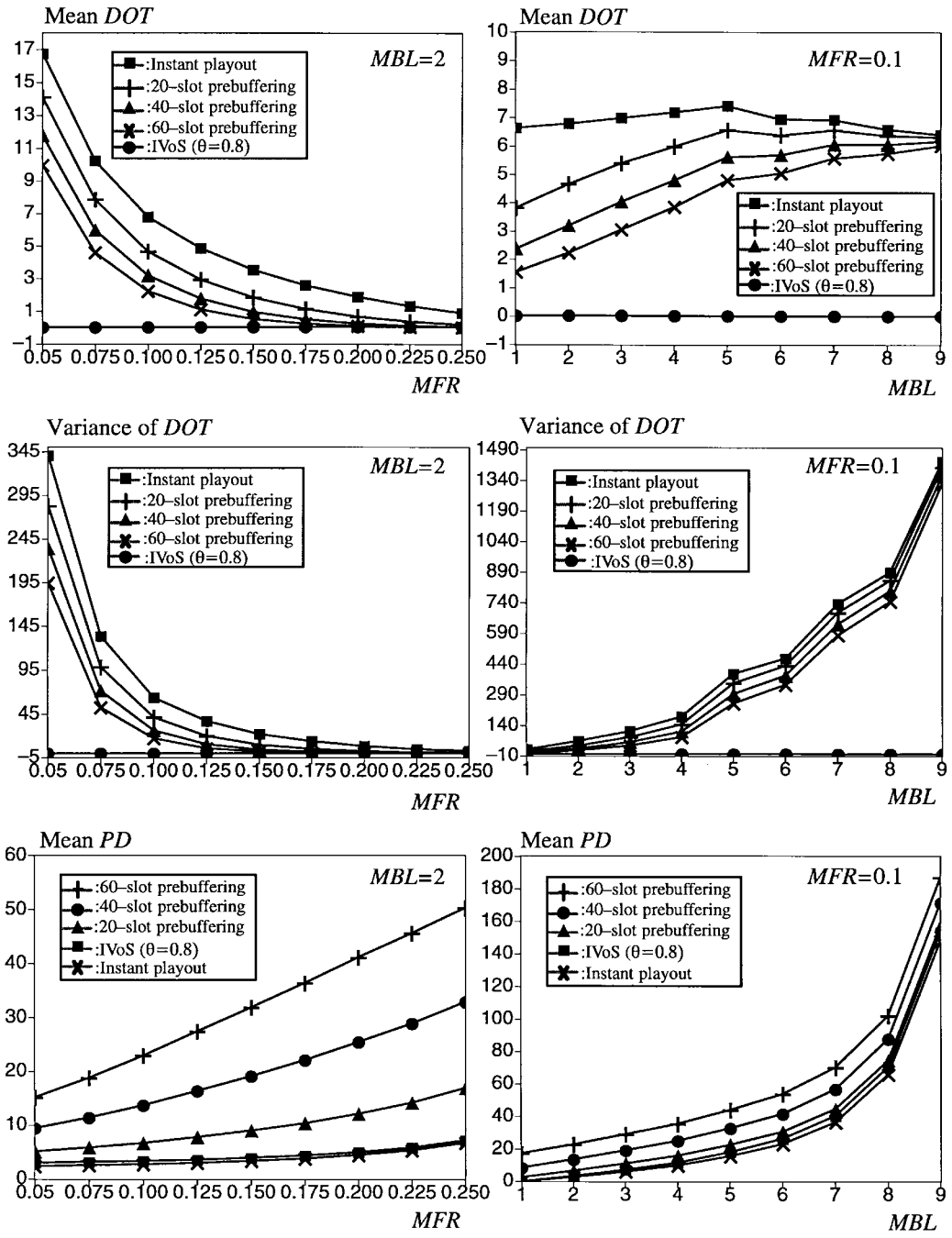


Fig. 8. Performance comparisons of playout approaches.

$$
\begin{aligned}
&= \sum_{t=1}^{\infty} \sum_{f=1}^{t} P\left[\tilde{t} + \left\lfloor \frac{1}{\theta}\,\tilde{b}(\mathcal{F}-1) \right\rfloor \right. \\
&\quad \left. - \tilde{f}\mathcal{F} = y | \tilde{t} = t, \tilde{f} = f\right] P[\tilde{t} = t, \tilde{f} = f] \\
&= \sum_{t=1}^{\infty} \sum_{f=1}^{t} P\left[\frac{\theta(y + \mathcal{F}f - t)}{\mathcal{F}-1} \le \tilde{b} \right. \\
&< \left. \frac{\theta(y + \mathcal{F}f - t + 1)}{\mathcal{F}-1} \middle| \tilde{t} = t, \tilde{f} = f \right] \\
&\quad \cdot P\left[\tilde{t} = t, \tilde{f} = f\right] \\
&= \sum_{t=1}^{\infty} \sum_{f=1}^{t} \sum_{b=\lceil \theta(y+\mathcal{F}f-t)/\mathcal{F}-1 \rceil}^{\lceil (\theta(y+\mathcal{F}f-t+1)/\mathcal{F}-1)-1 \rceil} \\
&\quad \cdot P\left[\tilde{t} = t, \tilde{f} = f, \tilde{b} = b\right], \qquad y > 0.
\end{aligned} \tag{13}
$$

Based on (8), (9), and (13), $E[A\tilde{B}D]$ can be directly computed.

We carried out analytical computation using Mathmetica and undertook event-based simulation in the C language. Both the analytical computation and simulation were performed in Pentium-Pro-200 PC's. The analytical computation terminated

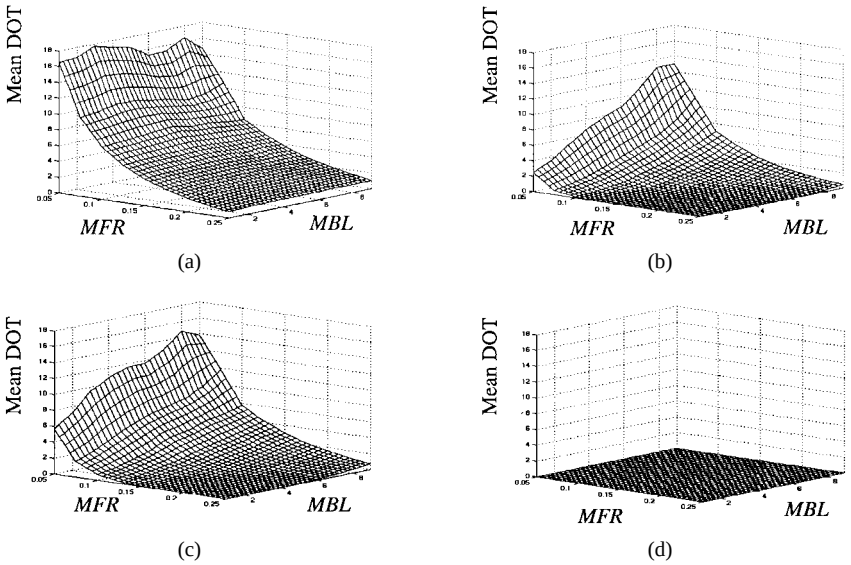


Fig. 9. Comparisons of mean DOT among three playout approaches. (a) Instant playout. (b) 100-slot prebuffering playout. (c) 200-slot prebuffering playout. (d) IVoS.

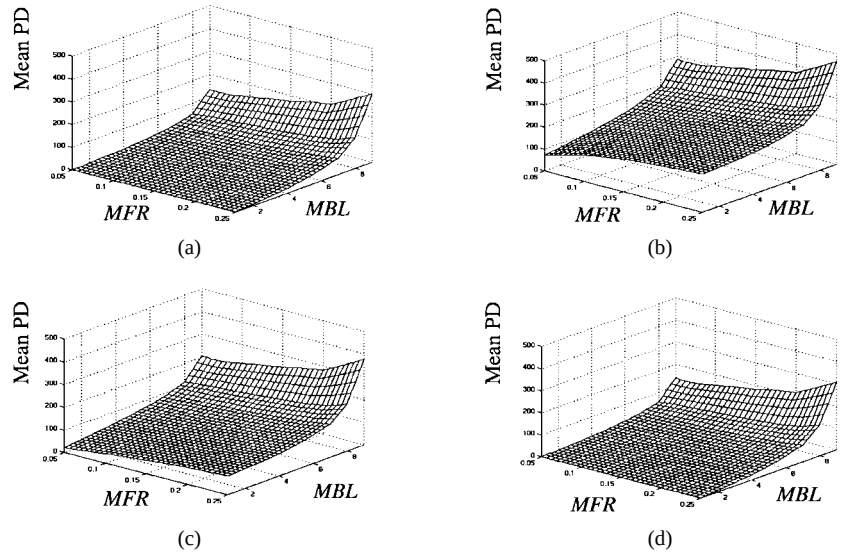


Fig. 10. Comparisons of mean PD among three playout approaches. (a) Instant playout. (b) 100-slot prebuffering playout. (c) 200-slot prebuffering playout. (d) IVoS.

when the cumulative distribution of $A\tilde{B}D$ reaches 99.9%. Simulation terminated when 10^6 slots had been executed (steady state was reached). In Fig. 6, we depict analytical and simulation results of mean ABD as a function of MBL under various θ's. In addition, due to the enormous amount of analytical computation time, results under various MFR's are gathered only via simulation. The figure demonstrates that analytical results are in profound agreement with simulation results. Moreover, mean ABD unanimously declines as MFR increases and increases with MBL. Finally, both figures display that a smaller θ results in greater mean ABD, as was expected.

In Table III, we display simulation results in order to compare the playout performances of IVoS and two other playout methods under an MBL of 2 and a θ of 0.9. The two playout methods are pure CBR playout and instant playout. The former achieves minimum mean and variance of DOT, and the latter yields minimum mean PD. Since pure CBR playout regulates the departure of all frames in a fully CBR fashion, it yields zero mean and variance of DOT. In comparison, IVoS offers negligible, minor distortion, as shown in Table III, under all nine traffic types. Moreover, with respect to mean PD, IVoS achieves satisfactory delay compared to the instant playout approach. As shown in Table III, the smaller the MFR, the greater the deficiency. It is also worth remembering that pure CBR unfortunately incurs larger PD. In contrast, instant playout undergoes poor mean and variance of DOT.

Up to this point, the problem left unsolved is the determination of θ. In principle, the selection of θ is based on the preference of QoS to DOT or to PD. A smaller (larger) θ is adopted if DOT (PD) takes precedence over PD (DOT).

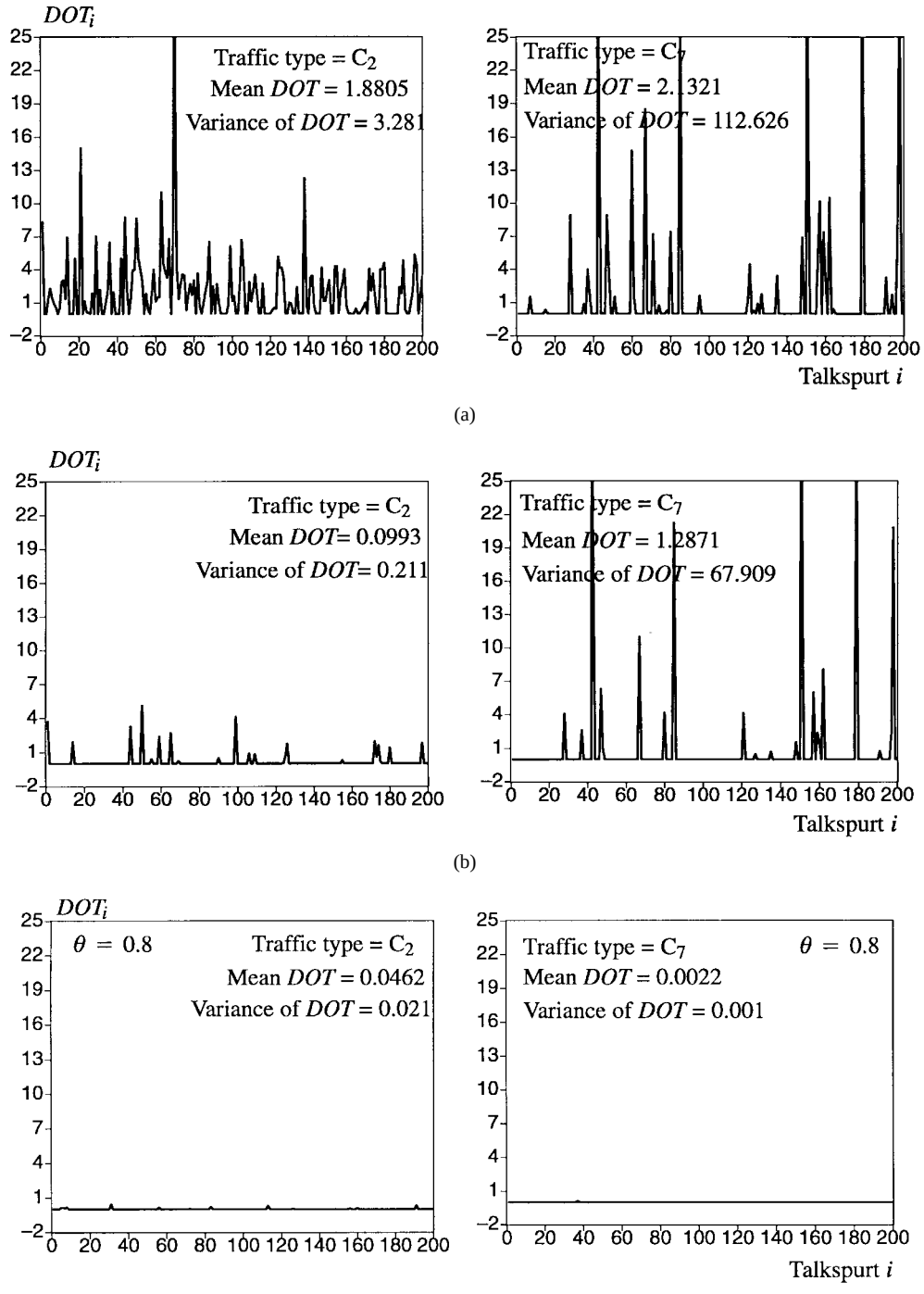


Fig. 11. Comparisons of DOT instances over talkspurts. (a) Instant playout approach. (b) Prebuffering playout (60-slot fixed buffering delay.) (c) IVoS. (d) Talkspurt i.

First, in the case of DOT, the problem can be formalized as the determination of θ's satisfying a given QoS requirement in terms of a ninety-ninth percentile of DOT. That is

$$\text{Find a range of } \theta\text{'s, satisfying } P\left[D\tilde{O}T \le \epsilon\right] \ge 0.99 \quad (14)$$

where ϵ is a given tolerable DOT value. To solve (14), we first discover two bounds of DOT; the upper bound and the lower bound, denoted as DOT_{UB} and DOT_{LB}, respectively. They can be derived under various conditions depicted in Fig. 7 and summarized in Table IV. With DOT_{UB} and DOT_{LB} introduced, the cumulative distribution function of $D\tilde{O}T$ can be formulated as

$$\begin{aligned} &P[D\tilde{O}T \le \epsilon] \\ &\quad = \sum_{t=1}^{\infty} \sum_{f=1}^{t} \sum_{b=1}^{f} P[D\tilde{O}T \le \epsilon | \tilde{t} = t, \tilde{f} = f, \tilde{b} = b] \\ &\qquad \cdot P[\tilde{t} = t, \tilde{f} = f, \tilde{b} = b] \end{aligned} \quad (15)$$

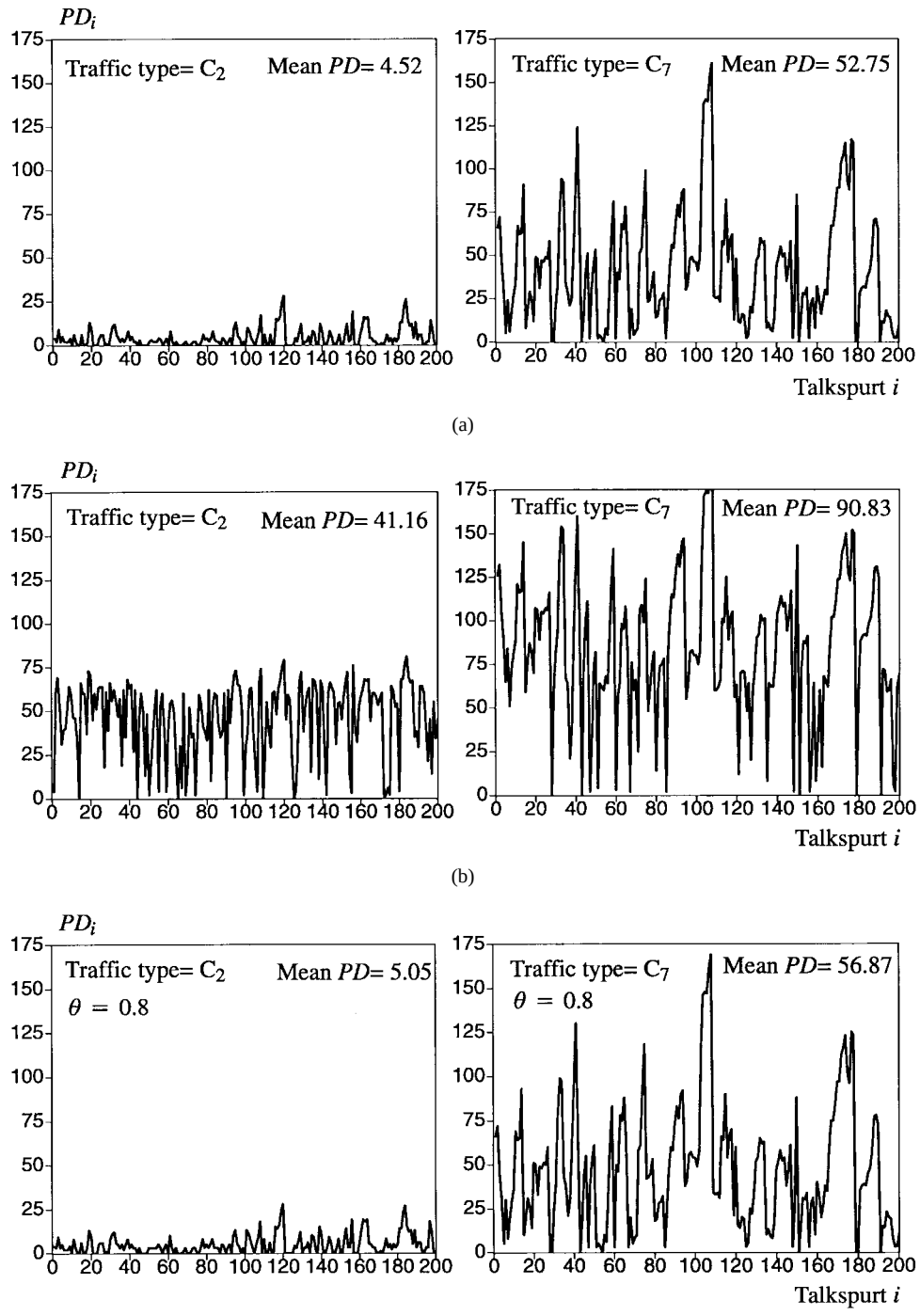


Fig. 12. Comparisons of PD instances over talkspurts. (a) Instant playout approach. (b) Prebuffering playout (60-slot fixed buffering delay.) (c) IVoS. (d) Talkspurt i.

where

$$P[\tilde{DOT} \le \epsilon | \tilde{t} = t, \tilde{f} = f, \tilde{b} = b] = \begin{cases} 0, & \text{if } \epsilon < DOT_{LB}, \\ 1, & \text{if } \epsilon \ge DOT_{UB}, \\ \dfrac{\epsilon - DOT_{LB}}{DOT_{UB} - DOT_{LB}}, & \text{if } DOT_{LB} \le \epsilon < DOT_{UB}, \\ & \text{and } DOT_{UB} \ne DOT_{LB} \end{cases} \quad (16)$$

assuming that $\tilde{DOT}$ is uniformly distributed between DOT_{LB} and DOT_{UB}.

On the basis of (14)–(16) and Table IV a viable range of θ's can be attained offline satisfying a given DOT requirement. Likewise, the same procedure can be similarly applied to the case of PD.

IV. Experimental Results

To demonstrate the viability of IVoS, using simulations, we compared IVoS and two other playout approaches in

terms of three performance metrics, the mean and variance of DOT, and mean PD. The two approaches are instant playout and prebuffering playout. In the instant playout approach, frames were queued and playbacked at a rate of $1/\mathcal{F}$ and below. It is worth noting that the instant playout approach differs from IVoS in the lack of buffering delay imposed on the first frame of each talkspurt. In the prebuffering approach, a predetermined fixed delay is imposed on the first frame of each talkspurt. For the purposes of showing performance contrast, in simulation we employed different variants of prebuffering playout using various fixed delays. All assumptions and parameters used for simulation are first summarized as follows:

- voice data rate during talkspurts: 64 kb/s;
- talk duration: 6.25 min (corresponding to 30 000 slots long);
- slot size: 100 bytes (corresponding to 12.5 ms/slot of voice);
- $\mathcal{F}$: 3 (corresponding to a mean load of 1/3 frame/slot during talkspurts);
- inbound traffic: nine traffic types, each with nine different burstiness listed in Table I;
- buffer size used for all approaches: 1200 slots (large enough to assure loss free);
- silence suppression rate: range from 25–85% as shown in Table I.

In Fig. 8, we draw comparisons between mean and variance of DOT and mean PD among three playout approaches under various MFR's and MBL's. In the prebuffering playout approach, we demonstrate three versions of prebuffering playout using 20, 40, and 60 slots of fixed buffering delays, respectively. Essentially, compared to other approaches, IVoS distinctively achieves superior playout yielding minimal mean and variance of DOT, irrespective of the increases in both MFR and MBL. On the contrary, the three other approaches undergo deteriorating mean and variance of DOT as MFR declines or as MBL increases. Moreover, they all suffer from poor variance of DOT, i.e., unstable speech intelligibility under medium to high burstiness traffic. As for mean PD, while all approaches yield longer delay under high MFR and MBL, IVoS retains compatibly short delay as compared to the optimal one, i.e., instant playout.

Figs. 9 and 10 plot two performance metrics under complete sets of MFR's and MBL's. It is particularly worth noting that, both versions of prebuffering (100 and 200 slots) playout achieve only bearable mean DOT, at the expense of a drastic increase in mean PD. In Figs. 11 and 12, we further plot instances of DOT and PD over talkspurt under two traffic types. The figures indicate that IVoS outperforms two other approaches in its invariably low DOT and acceptable PD over time. In addition, the figures clearly reveal a tradeoff problem between DOT and PD, using both the instant playout and prebuffering playout approaches. In contrast, IVoS is free from the tradeoff problem and achieves near-optimal performance. It is particularly worth noting that, in Fig. 11, under high burstiness traffic $(MBL = 7)$ the two playout approaches other than IVoS undergo high variance of DOT, resulting in unstable speech intelligibility throughout the entire talk.

V. Conclusions

In this paper, we have proposed IVoS, which is an NN-based intravoice synchronization mechanism. It is composed of three components: 1) the smoother buffer; 2) the NN traffic predictor; and 3) the CBR enforcer. The NN traffic predictor employs an online-trained BPNN to predict the talkspurt length, frame count, and the last burst length of every newly encountered talkspurt, based on the same set of characteristics of the past three talkspurt periods. According to the predicted characteristics, the CBR enforcer imposes an ABD, computed by means of a near-optimal simple closed-form formula, on the first frame of each talkspurt. It then regulates CBR-based departures for the rest of frames within the talkspurt, with the goal of assuring minimum mean and variance of DOT, and mean PD. The paper demonstrated simulation results which revealed that IVoS distinctively achieves superior playout yielding minimal mean and variance of DOT, irrespective of increases in MFR and MBL. In contrast, the two other playout approaches (instant playout and prebuffering playout) undergo deteriorating mean and variance of DOT as MFR declines. Moreover, they all suffer from poor variance of DOT, i.e., unstable speech intelligibility under medium and high burstiness traffic. As for mean PD, while existing approaches yield longer delay under high MFR and MBL, IVoS retains compatibly short delay as compared to the optimal one, i.e., instant playout.

References

[1] R. Steinmetz, "Human perception of jitter and media synchronization," *IEEE J. Selected Areas Commun.*, vol. 14, pp. 61–72, Jan. 1996.

[2] H. Eriksson, "MBone: The multicast backbone," *Commun. ACM*, vol. 37, pp. 54–66, Aug. 1994.

[3] D. E. Comer, *Internetworking with TCP/IP*. Englewood Cliffs, NJ: Prentice-Hall, 1994, vol. II.

[4] J. Bolot, "End-to-end frame delay and loss behavior in the Internet," in *Proc. ACM SIGCOMM*, Sept. 1993, pp. 289–298.

[5] H. Saito, *Teletraffic Technologies in ATM Networks*. Norwood, MA: Artech House, 1994.

[6] S. Nanda, D. Goodman, and U. Timor, "Performance of PRMA: A packet voice protocol for cellular systems," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 584–598, Aug. 1991.

[7] R. Onvural, *Asynchronous Transfer Mode Networks—Performance Issues*, 2nd ed. Norwood, MA: Artech House, 1995.

[8] J. Gruber and L. Strawczynski, "Subjective effects of variable delay and speech clipping in dynamically managed voice systems," *IEEE Trans. Commun.*, vol. COM-33, pp. 801–808, Aug. 1985.

[9] C. Nicolaou, "An architecture for real-time multimedia communication systems," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 391–400, Apr. 1990.

[10] W. Montgomery, "Techniques for frame voice synchronization," *IEEE J. Select. Areas Commun.*, vol. SAC-1, pp. 1022–1028, Dec. 1983.

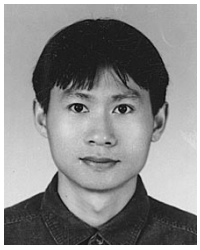
[11] R. Ramjee, J. Kurose, and D. Towsley, "Adaptive playout mechanisms for frameized audio applications in wide-area networks," in *Proc. IEEE INFOCOM*, 1994, pp. 680–688.

[12] S. Ramanathan and P. Rangan, "Feedback techniques for intra-media continuity and inter-media synchronization in distributed multimedia systems," *Comput. J.*, vol. 36, no. 1, pp. 19-31, 1993.

[13] T. Little and A. Ghafoor, "Multimedia synchronization protocols for broadband integrated services," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1368–1382, Dec. 1991.

[14] M. C. Yuang, P. L. Tien, and S. T. Liang, "Intelligent video smoother for multimedia communications," *IEEE J. Selected Areas Commun.*, vol. 15, pp. 136–146, Feb. 1997.

[15] M. C. Yuang, S. Liang, Y. Chen, and C. Shen, "Dynamic video playout smoothing method for multimedia applications," in *Proc. IEEE ICC*, 1996, pp. 1365–1369.
[16] Y. Xie, C. Liu, M. Lee, and T. Saadawi, "Adaptive multimedia synchronization in a teleconference system," in *Proc. IEEE ICC*, 1996, pp. 1355–1359.
[17] Y. Ishibashi, S. Tasaka, and A. Tsuji, "Measured performance of a live media synchronization mechanism in an ATM network," in *Proc. IEEE ICC*, 1996, pp. 1348–1354.
[18] H. Heffes and D. M. Lucantoni, "A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 856–868, June, 1986.
[19] L. Kleinrock, *Queueing Systems, Volume 1: Theory*. New York: Wiley, 1975.
[20] A. Tarraf and I. Habib, "A novel neural network traffic enforcement mechanism for ATM networks," *IEEE J. Select. Areas Commun.*, vol. 12, pp. 1088–1096, Aug. 1994.
[21] A. Tarraf, I. Habib, and T. Saadawi, "Intelligent traffic control for ATM broadband networks," *IEEE Commun. Mag.*, vol. 33, pp. 76–82, Oct. 1995.



Po L. Tien was born in Taiwan, R.O.C., in 1969. He received the B.S. degree in applied mathematics and the M.S. degree in computer and information science from the National Chiao Tung University, Taiwan, R.O.C., in 1992 and 1995, respectively. He is currently a Ph.D. candidate in the Department of Computer Science and Information Engineering at the National Chiao Tung University.

His current research interests include high-speed networking, multimedia communications, performance analysis, and applications of artificial neural networks.

Maria C. Yuang received the B.S. degree in applied mathematics from the National Chiao Tung University, Taiwan, R.O.C., in 1978, the M.S. degree in computer science from the University of Maryland, College Park, in 1981, and the Ph.D. degree in electrical engineering and computer science from the Polytechnic University, Brooklyn, NY, in 1989.

From 1981 to 1990 she was with AT&T Bell Laboratories and Bell Communications Research (Bellcore) where she was a Member of Technical Staff working on high-speed networking and protocol engineering. She was also an Adjunct Professor at the Department of Electrical Engineering at the Polytechnic University from 1989 to 1990. In 1990, she joined National Chiao Tung University, where she is currently a Professor of the Department of Computer Science and Information Engineering. Her current research interests include high-speed networking, multimedia communications, and performance analysis.