LSP-BASED MULTIPLE-DESCRIPTION CODING FOR REAL-TIME LOW BIT-RATE VOICE TRANSMISSIONS

Dong Lin and *Benjamin W. Wah*

Department of Electrical and Computer Engineering
and the Coordinated Science Laboratory
University of Illinois, Urbana
Urbana, IL 61801, USA
E-mail: {dlin, wah}@manip.crhc.uiuc.edu

ABSTRACT

A fundamental issue in real-time interactive voice transmissions over unreliable IP networks is the loss or late arrival of packets for playback. Such losses cannot be recovered by retransmissions due to tight time constraints in interactive applications. This problem is especially serious in transmitting low bit-rate coded speech when pervasive dependencies are introduced in a bit stream, leading to the loss of subsequent dependent frames when a single packet is lost or arrives late. In this paper, we propose a novel LSP-based multiple-description coding method that adapts its number of descriptions to network loss conditions in order to conceal packet losses in transmitting low-bit-rate coded speech over lossy packet networks. Based on high correlations observed in linear predictor parameters, in the form of Line Spectral Pairs (LSPs), of adjacent frames, we generate multiple descriptions in a sender by interleaving LSPs, and reconstruct lost LSPs in a receiver by linear interpolations. Without increasing the transmission bandwidth, our scheme represents a trade-off between the quality of received packets and the ability to reconstruct lost packets. Our experimental results on FS CELP show good performance.

1. INTRODUCTION

Error concealment is important when transmitting real-time packets that may be dropped or arrive late. The design of such schemes for low/very low bit-rate speech coding standards, such as Federal Standard 1016 CELP, is difficult due to dependencies introduced during coding. Most of these standards are based on the principle of linear prediction (LP) [1], whose linear prediction coefficients are commonly represented as *Line Spectral Pairs* (LSP). Since their coding algorithms assume an error-free channel, they remove as much temporal redundancies as possible in order to maximize their coding gain. As a result, the loss of one or more packets may result in subsequent frames not decode-able and severe quality degradation.

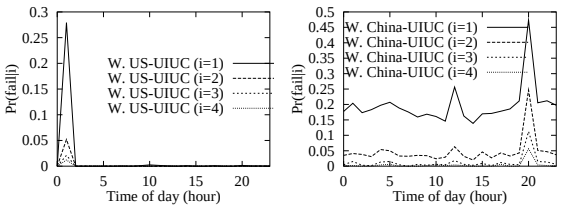
Existing error-concealment schemes for low bit-rate speech are either those with redundancies and those without.

Packet-level redundancies is not the best for fault tolerance in the Internet because they require considerable increases in bandwidth over non-redundant schemes. Typical methods include adding copies of previous frames, using parity or forward error-correction (FEC) codes to protect every n packets by a redundant packet, using FEC to protect only sensitive information in LP-coders, and piggy-backing in a packet a redundant version of some previous packets obtained by a lower bit-rate coder. In addition, redundant information can be sent to protect part of each packet. Receivers then use waveform substitution to replace lost packets by finding a best match on the redundant segments received. The drawback of these approaches is that good quality can only be achieved by sending considerable amount of redundant information.

In contrast, schemes with *zero-redundancy control* exploit implicit redundancies in voice streams and the property that voice transmissions can tolerate some loss without a lot of perceivable differences. Simple schemes typically perform loss-concealment actions at receivers alone. For instance, lost packets can be recreated by a) padding silence or white noise, b) repeating the last received packet, c) pattern matching using small segments of samples immediately before or after the lost packets, d) pitch-period replication by estimating pitch periods using speech segments immediately before the lost packets, e) performing waveform substitution based on previously received frames on each sub-band of linear prediction residues, f) copying coder parameters from the most recent error-free packet to both reconstruct the lost packet and update coder states, and g) repeating the parameters of the previous frame with simple modifications. These strategies work well when losses are infrequent and when packet sizes are small [2], but fail in networks with a high probability of loss.

Dissimilar to the single description-coding (SDC) schemes above, multi-description coding (MDC) is a zero-redundancy scheme that divides a data stream into equally important streams in such a way that the decoding quality with any subset is acceptable, and that better quality is obtained by more descriptions. It is assumed that losses to different descriptions are uncorrelated, and that the probability of losing all the descriptions is small. A straightforward way to implement MDC is *interleaving* (also called *sample-based MDC*) in which adjacent samples are distributed to different packets, thereby converting bursty losses to random losses that are much easier to recover. Receivers may reconstruct lost samples by odd-even sample interpolation, pattern-matching sample interpolation, and Kalman-based sample interpolation. We explain in Section 2 why this method is not suitable for concealing errors in low-bit-rate coded speech. In this paper, we focus on LP coder-specific MDC error-concealment methods.

PROC. IEEE INT'L CONF. ON MULTIMEDIA & EXPO, 2002.



a) UIUC-Berkeley b) UIUC-China

Figure 1: $Pr(fail|i)$, probabilities of bursty losses that cannot be recovered under interleaving factor i, in round-trip paths between UIUC and two remote locations.

2. LOSS CHARACTERISTICS IN THE INTERNET

This section presents results on loss characteristics of real-time transmissions for domestic and international connections and concludes that MDC is suitable for concealing packet losses.

During the experiments carried out in the first week of November 2001, a computer at UIUC periodically sent 2000 probe packets, at a rate of 30 packets per second and 500 bytes per packet, at the beginning of each hour over a 24-hour period to the echo port of a remote computer, and monitored the packets bounced back. Statistics, such as the sending and arrival times of each packet, was collected. To account for "delayed losses," each packet received had a scheduled "playback" time calculated from the arrival time of the first received packet and the difference of their sequence numbers. A packet was considered lost if it had been delayed by more than 200 msec of its scheduled playback time.

Interleaving is a good method to ease reconstruction because burst lengths are usually small. Define an *interleaving set* to be a collection of related information that is interleaved to different descriptions. When the burst length is less than the interleaving factor, or when a bursty loss involves information from different interleaving sets and some information in each interleaving set is received correctly, the information received can be used to recover the lost parts. For instance, with sample-based interleaving and an interleaving factor of two, a bursty loss of length one and a bursty loss of length two with samples belonging to different interleaving sets can be recovered by interpolations. With an interleaving factor of four, a bursty loss of length less than or equal to three and a bursty loss of length four, five, or six, with lost packets belonging to different interleaving sets, can be recovered. In general, with an interleaving factor of i, it is possible to recover a bursty loss of length less than or equal to $i-1$ and some bursty losses of length in the range $[i,\ (2i-2)]$.

The graphs in Figure 1 plot $Pr(fail \mid i)$, the probability that a packet cannot be recovered for interleaving factor i, and show that $Pr(fail \mid i)$ drops quickly when i increases. For all times and the two connections, $Pr(fail \mid i)$ is negligible when $i \geq 4$. Moreover, $i = 2$ works well for the connection to Berkeley, achieving $Pr(fail \mid i)$ well below 5%. For the connection to China (Figure 1b), an interleaving factor of two is not always enough because about 20% of the total losses will not be recoverable. The above experimental results suggest that a small number of descriptions (between two to four) is adequate. In most cases, two-way MDC leads to good recovery.

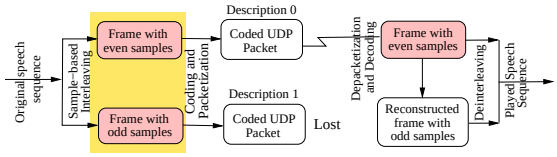


Figure 2: Sample-based MDC and reconstruction of a lost description at a receiver (shown with two descriptions).

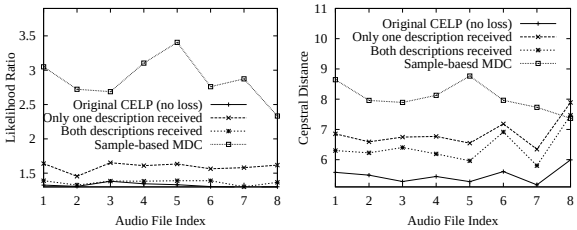


Figure 3: Quality comparison in terms of LR and CD among SDC, sample-based MDC with both streams received, and two-way LSP-based MDC under two scenarios in synthetic experiments.

3. LSP-BASED MDC

In this section, we propose a novel LSP-based MDC scheme to help packet-loss recovery for low-bit-rate LP coders.

First, we illustrate that the traditional sample-based MDC is not suitable for linear predictive coders. Figure 2 shows the sample-based MDC scheme in which a speech stream is interleaved into two streams, one containing the even samples and the other containing the odd ones, before coding each using an LP coder. Under the best condition, both coded streams will be received, decoded separately, and de-interleaved to rebuild the original stream. Even in this case, the playback quality is very poor, as illustrated in Figure 3 for FS CELP. Here, performance is measured by the *Itakura-Saito Likelihood Ratio* (LR) and the *Cepstral Distance* (CD).

Based on the eight sample voice streams, Figure 3 shows that both LR and CD of sample-based MDC increase dramatically for all the files tested under no loss, when compared to the decoding quality of SDC. Subjective hearing tests also indicate that sample-based MDC performs poorly. (Results on two-way LSP-based MDC are described later.)

The quality degradations of sample-based MDC are due to two major factors: aliasing introduced when the original stream is down-sampled, and the doubling of the time span of a coded frame in each interleaved stream. To avoid these drawbacks, we investigate the possibility of interleaving parameters after coding in LP coders, as shown in Figure 4.

We first study the properties of coder parameters. As mentioned in Section 1, the common part in modeling a vocal tract in most low-bit-rate speech coders is the linear predictor. From the physical point, since a vocal tract changes slowly and smoothly when one speaks, we study the properties of linear predictors, often represented by LSPs. There are three important properties of LSPs that allow them to be used for error concealment. First, the difference of adjacent LSPs is closely related to the formant bandwidths of speech [3], which suffice to specify the entire spectral envelope for vowels. This close relationship means that linear interpolations of LSPs, which is equivalent to the interpolation of the difference of adjacent LSPs, is closely related to the generation of smooth formant information. Second, we have found experimentally that LSPs

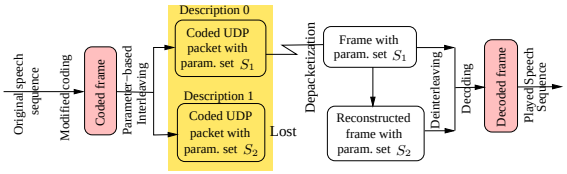


Figure 4: Proposed two-way MDC for LP speech coders.

Table 1: Inter-frame correlations of LSPs for the eight test streams (8000-Hz sampling rate, 30-msec frames, 45-msec Hamming window, 10^{th} analysis order, and 8061 frames).

Frame Distance	LSP									
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
1	0.84	0.81	0.82	0.79	0.78	0.77	0.77	0.80	0.74	0.64
2	0.62	0.60	0.63	0.59	0.56	0.56	0.57	0.61	0.53	0.44
3	0.43	0.45	0.50	0.45	0.41	0.42	0.43	0.48	0.39	0.34

change slowly from one frame to the next and have high inter-frame correlations. Such correlations are illustrated in the experimental results in Table 1, which shows high correlations for all the ten indices under a typical frame period of 30 msec. The correlations found are general because the tested streams involve significant variations in speaker characteristics. These results mean that the LSPs in a lost frame can be reconstructed from those received in adjacent frames. Last, the vector of LSP indices in a frame are monotonically increasing. This means that they are stable linear predictors [4], and that a vector of interpolated LSPs will also be stable linear predictors when using linear interpolations to reconstruct lost LSPs.

In short, linear interpolation of LSPs can be used to approximate the smooth changing of vocal tracts. Based on these observations, we propose an LSP-based MDC for LP coders.

We illustrate our LSP-based MDC scheme on the FS CELP coder. Figure 5a shows, in the original SDC coder, the generation of a 34-bit LSP vector for each 240-sample speech frame and four groups of adaptive and stochastic codewords (with 110 bits total), one for each 60-sample subframe. The 144-bit coded frame is then encapsulated in UDP before sent.

The single-description decoding process is the reverse process in Figure 5a. When one or more consecutive packets are lost, the receiver will not be able to recover the parameters in the corresponding coded frames and will play silence in the decoded frames. The performance of SDC is very sensitive to losses and burst lengths because decoding is state dependent. When valid packets are received again, there will be several frame delays before the proper states of the decoder are restored and satisfactory quality is achieved.

In our two-way LSP-based MDC design, the sender groups each pair of 240-sample frames in the original speech sequence into an interleaved set, performs linear prediction analysis, once for each frame, in order to generate a 34-bit LSP vector, and distributes the two LSP vectors to two frames in the two descriptions (see Figure 5b). However, instead of generating codewords for four 60-sample subframes (110 bits total), it extends the subframe size to 120 samples, generates codewords for four 120-sample subframes (110 bits total), and replicates all the codewords to the two frames of both descriptions. We replicate the codewords because they are not strongly correlated and cannot be reconstructed from codewords in adjacent subframes. With replicated codewords, we need to extend the subframe size in coding in order to keep the frame size in each description to be 144 bits, the same size as a coded frame in SDC. After coding and parameter interleaving, Description i, $i = 0, 1$, has frames that contain the LSPs of frame $2n + i$ and the codewords from all speech frames. Finally, the sender encapsulates a frame in each description in a UDP packet and alternates between Descriptions 0 and 1 in sending packets to the destination. Note that we have maintained the same frame size of 240 in linear prediction analysis and have overcome the aliasing problem, without decomposing the original speech samples into odd-even ones.

At the receiver side, if all the frames in both descriptions are received, the receiver carries out the reverse process in Figure 5b. It first deinterleaves the information received into a single coded stream by extracting the LSPs from frames in both descriptions and the codewords from frames in either description, before decoding the coded stream. Obviously, the quality of the decoded stream is equivalent to a coder with a frame size of 240 and a subframe size of 120. As said already, since we have preserved the precision of linear prediction analysis and have eliminated aliasing, the decoded stream can be guaranteed to have better quality than sample-based MDC. However, the decoded stream has worse quality than that of SDC because of its increased subframe size.

When some frames in one description are lost, the receiver only needs to reconstruct the lost LSPs, using the LSPs in those frames received in the other description. It does not reconstruct the codewords because they are replicated in both descriptions. For example, if a frame in Description 1 of Figure 5b is lost, then the receiver reconstructs the LSPs in the lost frame by averaging the LSPs of the immediately preceding and following frames in Description 0. It is easy to see that such reconstructions result in stable linear predictors. Moreover, since the receiver reconstructs the coding parameters of lost frames before decoding, it does not need to estimate the decoding states of lost frames as done in SDC.

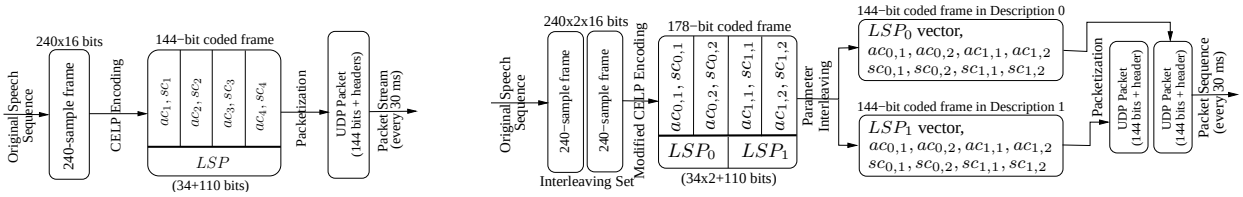
The above idea can be extended to four-way LSP-based MDC by extending the subframe size to 240 samples. Its quality is expected to be worse than that of two-way MDC due to its longer subframe size. Its details are not discussed here due to space limitations. We do not study MDC beyond four ways because we have shown in Figure 1 that four-way interleaving will be enough to conceal errors in most, if not all, of the cases. Further, an interleaving degree larger than four will result in even larger subframe sizes that will degrade quality further at the receiver, even when there are no losses.

The computational complexity of the MDC scheme is low. For example, for two-way MDC, one MDC excitation vector corresponds to two SDC excitation vectors. Hence, although the time to generate an MDC excitation vector is doubled, overall, the computation time is about the same as that of SDC.

4. EXPERIMENT RESULTS

In this section, we test our proposed two-way MDC algorithms on FS CELP using eight test streams. In order to compare various algorithms under the same operating condition, we further used trace-driven simulations that fed packet traces to our prototype and evaluated the statistics offline.

Figure 3 compares the decoding quality of two-way LSP-based MDC in synthetic experiments between the cases when both descriptions are received and when only one description is received. When both descriptions are received, LSP-based



a) SDC at sender b) LSP-based two-way MDC at sender

Figure 5: FS CELP under single description and the decomposition of an interleaved set of frames into multiple descriptions in LSP-based MDC (ac: adaptive codeword; sc: stochastic codeword).

MDC for FS CELP has almost no degradation in terms of LR and about 10-20% degradation in terms of CD when compared to SDC, and significant improvements when compared to sample-based MDC. When only one description is received, LSP-based MDC still gives good quality and performs better than sample-based MDC with no loss.

We have built a prototype in Linux that codes input speech in real-time from a microphone using SDC or LSP-based MDC, transmits the coded packets to the echo port of a remote computer, reconstructs any lost information from the packets received, and plays the reconstructed stream.

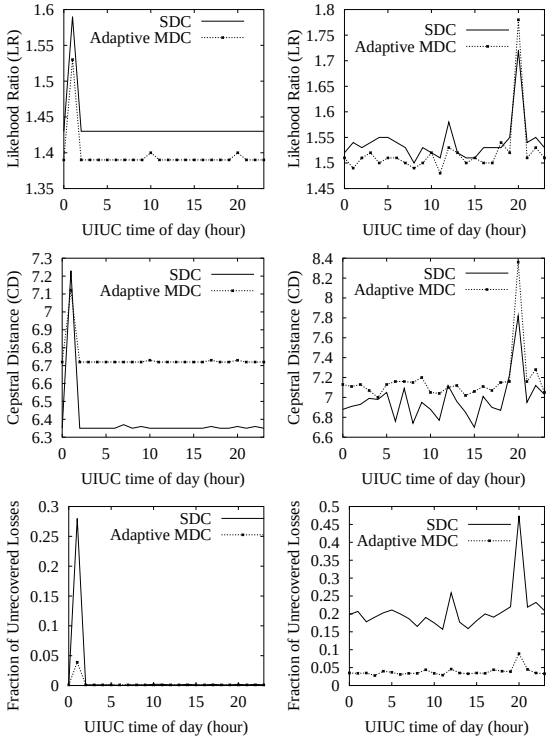
In order to adapt the number of descriptions to various loss conditions, the receiver currently collects loss statistics every second and sends to the sender a one-bit message in UDP, indicating whether two-way or four-way MDC should be used. Since the feedbacks sent by the receiver are subject to loss as well, the receiver will send a feedback packet every second, regardless of whether the degree of interleaving is changed. Our strategy is designed to avoid operating in four-way MDC as much as possible unless there are many long bursty losses, as two-way MDC performs better under low-loss conditions.

Figure 6 shows trace-driven results for FS CELP to Berkeley and China averaged over all received and reconstructed frames. One must be careful in comparing the results because LR and CD are not computed for unrecovered frames (between 1-28% for the UIUC-Berkeley connection and between 20-45% for the UIUC-China connection for SDC). To illustrate this difference, we also plot the fraction of frames that were lost or unrecovered at the receiver for both schemes.

In general, adaptive MDC always have less distortions than SDC in terms of LR, but may have more distortions than SDC in terms of CD. Some distortions in adaptive MDC are introduced because excitations are extracted on larger subframes that are reflected in terms of CD. Other distortions in terms of both LR and CD are introduced when some frames are lost and unrecoverable, leading to incorrect decoding states for subsequent frames received. Such distortions happen in both cases, but affect the quality of SDC more severely due to its large fraction of unrecoverable frames. Based on the combined effects, adaptive MDC almost always performs better than SDC in terms of LR, but may perform better or worse than SDC on received and reconstructed frames, depending on the fraction of unrecoverable losses.

From the perspective of end users, SDC will give discontinuous playback in high-loss connections due to its difficulty in restoring proper decoding states after one or more packets are lost. In contract, adaptive MDC will give much smoother playback, despite slightly lower quality on all the frames received or reconstructed due to its increased subframe size.

Due to space limitations, we do not show results on other close-looped (ITU G.723.1 ACELP and ITU G.723.1 MP-MLQ) and open-looped (FS MELP) linear-prediction coders.



a) UIUC-Berkeley b) UIUC-Western China

Figure 6: Comparison of reconstruction quality between SDC and adaptive MDC for FS CELP on received and reconstructed frames over a 24-hour period for the round-trip connections between UIUC and two destinations.

5. REFERENCES

[1] R. V. Cox and P. Kroon, "Low bit-rate speech coders for multimedia communication," *IEEE Communication Magazine*, vol. 34, no. 12, pp. 34–41, Dec. 1996.

[2] V. Hardman, M. A. Sasse, M. Handley, and A. Watson, "Reliable audio for use over the Internet," in *Int'l Networking Conf.*, June 1995, pp. 171–178.

[3] R. Goldberg and L. Riek, *A Practical Handbook of Speech Coders*, Boca Raton, Florida: CRC Press, 2000.

[4] F. K. Soong and B. H. Juang, "Line spectrum pair (LSP) and speech data compression," in *IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, 1984, pp. 1.10.1–1.10.4.