PACKET LOSS CONCEALMENT BASED ON EXTRAPOLATION OF SPEECH WAVEFORM

Juin-Hwey Chen

Broadcom Corporation
Irvine, California, USA

ABSTRACT

A class of packet loss concealment algorithms for speech coding is presented. It generates the replacement waveform for the lost frame by direct extrapolation of the past speech waveform, with or without look-ahead. The ITU-T G.722 Appendix III standard is based on it. When a future frame is unavailable (without look-ahead), the PLC algorithm gives significantly better speech quality than G.711 Appendix I – by about 0.2 PESQ for high packet loss rates. When a future frame is available (with look-ahead), the PLC algorithm uses the decoded speech waveform in the future frame to guide the pitch contour of waveform extrapolation during the lost frame such that the extrapolated waveform is phase-aligned with the decoded waveform after the packet loss. This technique further improved PESQ by another 0.2 for high packet loss rates.

Index Terms— Packet Loss Concealment, PLC, Frame Erasure Concealment, G.722 Appendix III, phase-aligned.

1. INTRODUCTION

The basic ideas of Packet Loss Concealment (PLC) or Frame Erasure Concealment (FEC) date back to at least 1980s [1] – [3]. The early PLC in the 1980s mostly dealt with memoryless codecs such as Pulse Code Modulation (PCM). The PLC method in [3] extrapolates previously decoded speech waveform to fill the lost frame. It performs overlap-add between the extrapolated waveform and the decoded waveform to smooth out potential waveform discontinuity. This operation increases the signal delay, because speech samples near the end of each frame cannot be played back until they are overlap-added during the next frame. Based on [3], the PLC standard in the ITU-T G.711 Appendix I adds 30 samples (3.75 ms) of signal delay due to this overlap-add operation.

Since early 1990s, there has been a significant increase in PLC research activities, with most of them concentrated on predictive coding. This is probably because most modern speech coding standards are predictive codecs with built-in PLC. The PLC schemes for predictive codecs typically extrapolate the excitation signal and then filter the extrapolated excitation signal with an extrapolated synthesis filter to obtain the output speech. Any waveform discontinuity in the excitation signal due to the extrapolation is mostly smoothed out by synthesis filtering. Examples of such PLC methods can be found in [4] and [5].

An exception to this common technique of extrapolating the excitation signal is a PLC scheme developed in 2001 for predictive coding, which extrapolates the speech waveform directly [6]. The PLC without look-ahead [7], to be presented later in this paper, was developed in 2004 based on this technique in [6].

Later also in 2004, the PLC algorithm in [7] was further improved to take advantage of the situation when the frame after the current lost frame is received and available (with look-ahead). This condition can happen if additional buffering delay is added to the adaptive jitter buffer. Most of the distortion in the PLC output signal comes not from the lost frames, but from the frames after packet loss, often due to misalignment between the extrapolated waveform and the decoded waveform. When the speech waveform in the frame after the packet loss is available, this improved PLC algorithm with look-ahead [8] uses it to guide the waveform extrapolation during the lost frame in such a way that the waveform misalignment is largely eliminated, thus resulting in noticeable audio quality improvement.

Originally developed for memoryless or block-independent codecs, the PLC algorithms in [7] and [8] later found their use in the PLC of other types of codecs with memory, including an ITU-T standard PLC known as G.722 Appendix III [9].

The rest of this paper is organized as follows. Section 2 first presents the PLC algorithm without look-ahead [7]. Section 3 then describes the PLC algorithm with look-ahead [8]. Sections 4 and 5 discuss the complexity and performance of these two PLC algorithms. Section 6 gives the conclusion.

2. PLC WITHOUT LOOK-AHEAD

A high-level block diagram of the proposed PLC algorithm without look-ahead is shown in Fig. 1.

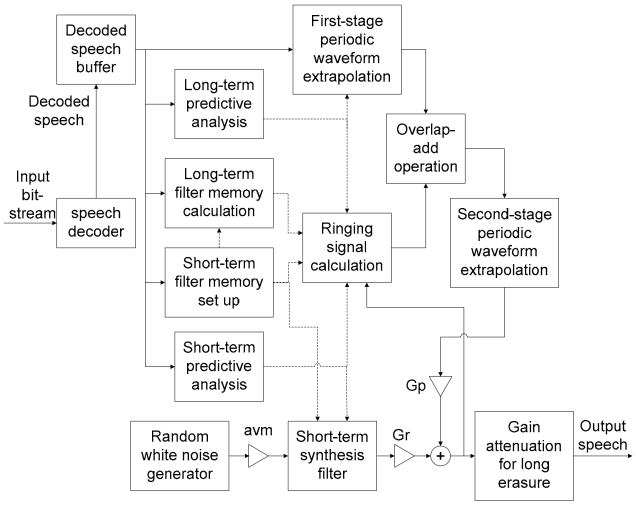
Decoded speech buffer
Decoded speech
Input bit-stream
speech decoder
Long-term predictive analysis
Long-term filter memory calculation
Short-term filter memory set up
Short-term predictive analysis
First-stage periodic waveform extrapolation
Overlap-add operation
Ringing signal calculation
Second-stage periodic waveform extrapolation
Gp
Random white noise generator
avm
Short-term synthesis filter
Gr
+
Gain attenuation for long erasure
Output speech

Fig. 1 The proposed PLC algorithm without look-ahead

The solid arrows in Fig. 1 indicate the flow of speech or related signals. The arrows with dashed lines indicate the control flow involving the updates of filter parameters and filter memory.

If the current frame is a "good" frame (received frame) but not the first good frame after a packet loss, the speech decoder decodes the input bit-stream normally. The decoded speech is stored in the decoded speech buffer and also played back as output speech. If the current frame is the first "bad" frame (lost frame) in a row, the PLC algorithm generates a filter ringing signal that naturally and smoothly extends the decoded speech waveform to the first several samples of the current frame. Overlap-adding such a filter ringing signal with the extrapolated signal will smooth out potential waveform discontinuity at the beginning of the first bad frame without adding any additional signal delay as in G.711 Appendix I and [3]. See [6] for more details about the filter ringing method.

Unlike the filter ringing method in [6] where the filter parameters and filter memory are readily available, when the PLC algorithm is not used with a predictive codec, it needs to calculate such information from the signal stored in the decoded speech buffer. This is achieved by first performing short-term predictive analysis on the previously decoded speech. The memory of the short-term filter for ringing signal calculation is then set up by taking the last few samples of the decoded speech in the last frame and reversing the order.

Long-term predictive analysis is next performed to extract the pitch period, the pitch predictor tap, and a scaling factor for extrapolation. The pitch extraction algorithm of the BroadVoice® codec [10] – [12] is used. To save the RAM memory, the PLC algorithm does not maintain a long buffer for the long-term filter memory. Instead, the long-term filter memory is calculated by using the short-term filter to inverse-filter the decoded speech only for the segment that is one pitch period earlier than the overlap-add region at the beginning of the first bad frame.

With the long-term and short-term filter coefficients and memory all set up, the ringing signal is calculated by feeding a zero signal through the long-term filter followed by the short-term filter. The resulting filtered signal is the ringing signal.

Next, the algorithm performs first-stage periodic waveform extrapolation (PWE) to extrapolate the speech signal up to the end of the overlap-add region at the beginning of the first bad frame, by periodically repeating the previous speech waveform using the extracted pitch period and the extrapolation scaling factor. The overlap-add operation is then performed to get a smooth transition from the ringing signal to the extrapolated speech signal. Then, second-stage PWE continues the extrapolation from the end of the overlap-add region of the current frame to the end of the overlap-add region of the next frame. The PWE is performed in two stages to avoid repeating potential waveform discontinuity at the beginning of the first bad frame if the pitch period is less than the frame size plus the overlap-add length.

Separately, a voicing measure is calculated to measure the degree of periodicity in previously decoded speech. This voicing measure controls *Gp* and *Gr*, the scaling factors for the periodic component and the random component, respectively. If the last good frame is essentially periodic, then *Gp* = 1 and *Gr* = 0. If the last good frame exhibits essentially no periodicity, then *Gp* = 0 and *Gr* = 1. If the last good frame is between these two extremes, then both *Gp* and *Gr* are non-zero, with *Gp* roughly proportional to the degree of periodicity in the last good frame, and *Gp* + *Gr* = 1.

The random component of the PLC output is obtained by filtering a scaled random white noise sequence with the short-term synthesis filter. The scaling factor *avm* is the average magnitude of the short-term prediction residual signal of the last good frame.

The filter ringing signal only needs to be calculated at the beginning of the first bad frame. Starting from the first bad frame, the waveform extrapolation is extended beyond the end of the current frame by the length of the overlap-add region, and the additional extrapolated samples is used in the same way as the filtering ringing signal for the next frame.

If the current frame is within 20 ms from the beginning of the first bad frame in a string of bad frames, the output of the adder in Fig. 1 is directly played back as the PLC output speech. If the current frame is more than 20 ms from the beginning of the first bad frame, then the output waveform of the adder is gradually attenuated toward zero until it reaches zero at 60 ms. This is to avoid unnatural tonal distortion due to prolonged packet loss.

If the current frame is the first good frame after a packet loss, the decoded speech is overlap-added with the extrapolated signal that goes beyond the end of the last bad frame to avoid the potential waveform discontinuity at the beginning of the frame.

Due to its good performance and reasonable complexity without introducing additional delay, this PLC algorithm without look-ahead was used in the ITU-T G.722 Appendix III standard [9] as the fundamental method to extrapolate the full-band speech waveform and handle waveform transition at frame boundaries. Of course, many other techniques were also added to G.722 Appendix III to handle other issues specific to G.722.

3. PLC WITH LOOK-AHEAD

If the current frame is a bad frame and the next frame is not available, the PLC algorithm with look-ahead defaults back to the algorithm described in the last section. If the next frame is a good frame that is received before the current bad frame is played back, the algorithm decodes the next frame first and then attempts to use the decoded speech to guide the waveform extrapolation of the current bad frame in such a way that the extrapolated waveform will be roughly phase-aligned with the decoded speech.

If the speech codec is a block-independent codec, i.e., a codec without frame-to-frame memory, the overlap-add region can be placed at the beginning of the next good frame since the decoded speech in the next good frame is not affected by the packet loss. If the speech codec has frame-to-frame memory but the decoded speech in the next good frame recovers back to normal in a relatively short time, then the overlap-add region can be delayed by an empirically determined amount of time until the decoded speech waveform has roughly recovered.

The PLC algorithm with look-ahead performs the "guided extrapolation" only if both the last frame and the next good frame are voiced frames, since this is the only condition when there may be a clearly defined "phase" of the pitch cycle waveform. When this condition is met, the system first performs a first-pass PWE using the pitch period of the last frame. The phase of the pitch cycle waveform for such a first-pass PWE is shown as the dashed saw-tooth wave in Fig. 2. The light shaded area to the left of the time index 0 is the last portion of the last frame. The light shaded area on the right is the first portion of the next good frame. In the particular example of Fig. 2, the overlap-add region, shown as the dark shaded area, is placed at the beginning of the next good frame. The time index *g*, at the center of this overlap-add region, is the target time instant for the second-pass PWE waveform to be fully aligned with the decoded waveform.

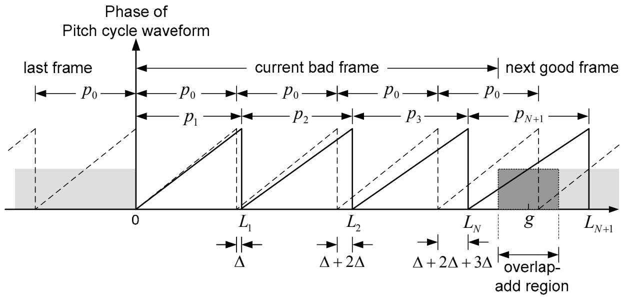

Fig. 2 Pitch cycle phase and timing diagram

Next, the relative time lag between the first-pass extrapolated waveform and the decoded waveform in the overlap-add region is determined for a time lag range around zero. If this time lag is zero, then the extrapolated waveform and the decoded waveform are in phase and no more adjustment needs to be made. The first-pass extrapolated waveform is overlap-added with the decoded waveform to produce the PLC output signal. If the time lag is not zero, the extrapolated waveform is out of phase with the decoded waveform. This indicates that the pitch period has changed during the bad frame. In this case, a new pitch period contour is calculated so that a second-pass PWE based on the new pitch period contour may bring the extrapolated waveform in phase.

Determining the new pitch period contour is relatively simple if the parameter *g* is less than the pitch period. However, in the general case when there are multiple pitch cycles during these *g* samples, it turns out to be a non-trivial mathematical problem.

To make this problem easier to solve, assume that for each new pitch cycle waveform, the pitch period is changed by Δ samples. Thus, if p_0 is the pitch period of the last frame, then the pitch period of the *k*-th new pitch cycle in the current bad frame is $p_k = p_0 + k\Delta$, as shown in the solid saw-tooth wave in Fig. 2. The location of the end of the *k*-th new pitch cycle is given by

$$L_k = \sum_{i=1}^{k} p_i = \sum_{i=1}^{k} (p_0 + i\Delta) = kp_0 + \frac{k(k+1)}{2}\Delta = kp_0 + O_k \quad ,$$

where $O_k = k(k+1)\Delta/2$ is the offset or the time lag between the first-pass extrapolated waveform and the second-pass extrapolated waveform at the end of the *k*-th new pitch cycle. This offset is also shown below the time axis in Fig. 2.

Suppose there are *N* full new pitch cycles plus a fractional cycle from time index 0 to time index *g*, as shown in Fig. 2. The number of samples for the fractional pitch cycle is $f = g - L_N$. The time lag *l* between the first-pass extrapolated waveform and the second-pass extrapolated waveform at time index *g* can be obtained by the linear interpolation between the two offsets O_N and O_{N+1}:

$$l = (1 - f / p_{N+1})O_N + (f / p_{N+1})O_{N+1}.$$

Replacing O_N and O_{N+1} with the formula for O_k above with *k* = *N* and *k* = *N* + 1, respectively, we have

$$2lp_{N+1} = (p_{N+1} - f)N(N+1)\Delta + f(N+1)(N+2)\Delta,$$

or

$$2lp_{N+1} = p_{N+1}N(N+1)\Delta + 2f(N+1)\Delta.$$

Plugging in

$$p_{N+1} = p_0 + (N+1)\Delta \text{ and}$$

$$f = g - L_N = g - Np_0 - N(N+1)\Delta/2 \;,$$

we have

$$2l[p_0 + (N+1)\Delta] = [p_0 + (N+1)\Delta]N(N+1)\Delta + [2g - 2Np_0 - N(N+1)\Delta](N+1)\Delta$$

or

$$2lp_0 = [2g - Np_0 - 2l](N+1)\Delta.$$

Thus, the pitch period change per pitch cycle is given by

$$\Delta = \frac{2lp_0}{[2g - Np_0 - 2l](N+1)}$$

The pitch period change per sample for the *k*-th new pitch cycle, $\delta_k = \Delta / p_k$, can then be used to calculate the new sample-adaptive pitch period at each sample, which is generally not an integer. The non-integer pitch period is rounded off to the nearest integer before it is used in PWE. To avoid the waveform discontinuity when the rounded pitch period changes, overlap-add is used to transition from the PWE output based on the old integer pitch period to the PWE output based on the new integer pitch period.

The scaling factor *c* for the second-pass PWE is obtained as follows. Start from the overlap-add region in the next good frame, trace back in time through the new pitch period until the region that is *m* pitch cycles earlier lands in the frame (or frames) before the current bad frame. Next, calculate *r*, the ratio of the average magnitude of the decoded speech in the overlap-add region in the next good frame over the average magnitude of the speech signal *m* pitch cycles earlier. The scaling factor *c* is then calculated as

$$c = \sqrt[m]{r} = r^{1/m} = 2^{\frac{1}{m}\log_2 r} \;.$$

We need to take the *m*-th root of *r* because the scaling factor *c* is applied *m* times through the *m* pitch cycles.

This two-pass guided PWE method eliminates a large number of audible glitches that were caused by misalignment between the extrapolated speech and the decoded speech.

The basic ideas behind this two-pass guided PWE method also inspired the development of the time-warping technique [13] in G.722 Appendix III. During the development of G.722 Appendix III, initially this two-pass guided PWE was applied in the first good frame after a packet loss (rather than in a bad frame when the next good frame is available). It was found that applying the technique this way required extending the PWE and delaying the overlap-add toward the end of the first good frame, which caused quality degradation that negated the benefits of this approach. To address this problem but still keep the spirit of the original guided PWE, which was to align the extrapolated speech with the decoded speech, the time-warping technique in [13] was thus invented. The decoded speech is time-warped toward the beginning of the first good frame to align with the extrapolated waveform. Since the overlap-add is not delayed, net quality improvement is achieved.

It is also possible to perform waveform interpolation between the two good frames before and after the packet loss. However, doing so normally requires the extraction of the pitch period of the speech segment after the packet loss, which in turn requires a long segment of decoded speech after the packet loss to be available. The segment length needs to be at least the pitch analysis window size plus the maximum possible pitch period, which is typically in the 25 to 35 ms range. For a small packet size of 10 ms, 3 to 4 consecutive packets must be received after a packet loss, and the PLC system must add 30 to 40 ms of delay. If only one future frame of 10 ms is available, interpolation cannot be applied since 10 ms is even smaller than the pitch period of some male voices.

In contrast, for the proposed two-pass guided PWE approach to work, it requires a look-ahead of no more than 10 ms (or even just 5 ms for block-independent codecs). Hence, this approach is particularly useful when the additional delay allowed for the PLC is too short for waveform interpolation but is at least 5 to 10 ms.

4. COMPLEXITY

The proposed PLC algorithm without look-ahead described in Section 2 has a reasonable complexity. The code size is on the order of 3 kwords, and the RAM and data table ROM each takes a few hundred words. On a typical fixed-point DSP, the computational complexity is roughly 3.8 MIPS worst case and 1.7 MIPS average for the narrowband (8 kHz) version and 5 MIPS worst-case and 2.3 MIPS average for the wideband (16 kHz) version. The proposed PLC algorithm with look-ahead described in Section 3 requires slightly higher complexity.

5. PERFORMANCE

Figure 3 shows the PESQ (ITU-T P.862) scores of the two proposed PLC algorithms (applied to G.711), G.711 Appendix I, and the zero-fill method that fills bad frames with zeros. All PESQ scores were measured with random packet loss and a packet size of 10 ms. For high packet loss rates, the proposed PLC without look-ahead is about 0.2 PESQ higher than G.711 Appendix I, which is quite significant. Listening comparison confirmed that it indeed sounded noticeably better than G.711 Appendix I. This was achieved without adding the 3.75 ms delay as in G.711 Appendix I. For high packet loss rates, the proposed PLC with look-ahead provides another 0.2 PESQ improvement and reaches a total improvement of 0.4 PESQ over G.711 Appendix I.

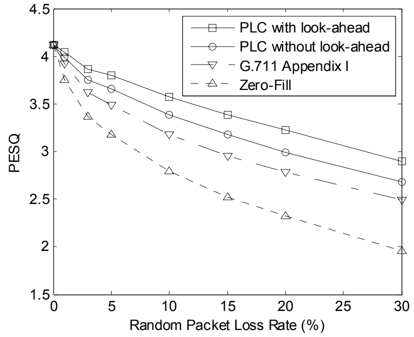

Fig. 3 PLC output speech quality as measured by PESQ

6. CONCLUSION

Novel PLC algorithms with or without look-ahead based on extrapolation of speech waveform were presented. Since they extrapolate the speech waveform directly, the algorithms can be applied to almost any speech codec. After the PLC output waveform is produced, updating the states of the speech decoder is the only remaining issue. Significant performance improvement over G.711 Appendix I is achieved.

7. ACKNOWLEDGMENT

The author would like to thank Jes Thyssen, Cheng-Chieh Lee, and Robert Zopf for their contributions to the C and assembly codes for the proposed PLC algorithm without look-ahead.

8. REFERENCES

[1] N. S. Jayant and S. W. Christensen, “Effects of packet losses in waveform coded speech and improvements due to an odd-even sample interpolation procedure,” *IEEE Trans. Communications*, vol. COM-29, pp. 101-109, February 1981.

[2] C. J. Weinstein and J. W. Forgie, “Experience with speech communication in packet networks,” *IEEE J. Selected Areas Communications*, vol. SAC-1, pp. 963-980, December 1983.

[3] D. J. Goodman, et al. “Waveform substitution techniques for recovering missing speech segments in packet voice communications”, *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-34, pp. 1440–1448, December 1986.

[4] C. R. Watkins and J.-H. Chen, “Improving 16 kb/s G.728 LD-CELP Speech Coder for Frame Erasure Channels,” *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, pp. 241-244, May 1995.

[5] R. Salami, et al., “Design and Description of CS-ACELP: a Toll Quality 8 kb/s Speech Coder,” *IEEE Trans. Speech and Audio Processing*, Vol. 6, No. 2, pp. 116 - 130, March 1998.

[6] J.-H. Chen, “Packet Loss Concealment for Predictive Speech Coding Based on Extrapolation of Speech Waveform,” *Conf. Rec. 41st Asilomar Conf. Signals, Systems, Computers*, pp. 2088 - 2092, November 2007.

[7] J.-H. Chen, U.S. Patent Application No. 20060265216, “Packet loss concealment for block-independent speech codecs,” filed September 26, 2005.

[8] J.-H. Chen, U.S. Patent Application No. 20080046235, “Packet loss concealment based on forced waveform alignment after packet loss,” provisional application filed August 15, 2006; utility application filed July 31, 2007.

[9] J. Thyssen, R. Zopf, J.-H. Chen, and N. Shetty, “A Candidate for the ITU-T G.722 packet loss concealment standard,” *Proc. IEEE Int’l Conf. Acoustics, Speech, and Signal Processing*, vol. 4, pp. IV-549 – IV-552, April 2007.

[10] J.-H. Chen and J. Thyssen, “The BroadVoice Speech Coding Algorithm,” *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, pp. IV-537 - IV-540, April 2007.

[11] “BV16 Speech Codec Specification for Voice over IP Applications in Cable Telephony,” American National Standard, ANSI/SCTE 24-21 2006.

[12] “BV32 Speech Codec Specification for Voice over IP Applications in Cable Telephony,” American National Standard, ANSI/SCTE 24-23 2007.

[13] R. Zopf, J. Thyssen, and J.-H. Chen, “Time-Warping and Re-Phasing in Packet Loss Concealment,” *Proc. Interspeech 2007 – Eurospeech*, pp.1677-1680, Antwerp, Belgium, August 2007.