Adaptive Postfiltering for Quality Enhancement of Coded Speech

Juin-Hwey Chen, *Senior Member, IEEE,* and Allen Gersho, *Fellow, IEEE*

Abstract— **An adaptive postfiltering algorithm for enhancing the perceptual quality of coded speech is presented. The postfilter consists of a long-term postfilter section in cascade with a short-term postfilter section and includes spectral tilt compensation and automatic gain control. The long-term section emphasizes pitch harmonics and attenuates the spectral valleys between pitch harmonics. The short-term section, on the other hand, emphasizes speech formants and attenuates the spectral valleys between formants. Both filter sections have poles and zeros. Unlike earlier postfilters that often introduced a substantial amount of muffling to the output speech, our postfilter significantly reduces this effect by minimizing the spectral tilt in its frequency response. As a result, this postfilter achieves noticeable noise reduction while introducing only minimal distortion in speech. The complexity of the postfilter is quite low. Variations of this postfilter are now being used in several national and international speech coding standards. This paper presents for the first time a complete description of our original postfiltering algorithm and the underlying ideas that motivated its development.**

I. Introduction

EARLY speech coders operating at high bit-rates were usually designed to minimize the energy of quantization noise, or equivalently, to maximize the signal-to-noise ratio (SNR). In these traditional coders, the coding noise is roughly white, i.e., the noise spectrum is roughly flat. As the encoding rate goes down to 16 kb/s and below, the SNR also drops and the noise floor of this white coding noise is elevated to such an extent that it is very difficult, if not impossible, to keep it below the threshold of audibility.

Two perceptually motivated approaches were proposed to deal with this problem. The first one uses *noise spectral shaping* at the speech encoder. This method was first proposed in the late 1970s by Atal, Schroeder, and Hall [2], [3] and by Makhoul and Berouti [4]. It has been used successfully in adaptive predictive coding (APC) [2], [4], [5], multipulse linear predictive coding (MPLPC) [6], and code-excited linear prediction (CELP) coders [7]. The basic idea is to shape the spectrum of the coding noise so that it follows the speech spectrum to some extent. Roughly speaking, the ratio of signal-to-noise power densities at each frequency should exceed some minimum value that depends on frequency and the local character of the speech signal. Coding noise spectrally shaped in this way is less audible to human ears due to the noise-masking effect of the human auditory system [3], [8], [9]. However, as will be discussed later, at low encoding rates, noise spectral shaping alone is not sufficient to make the coding noise inaudible.

The second perceptually-based approach uses an *adaptive postfilter* at the speech decoder output. The use of an adaptive rather than fixed filter is based on the need to change the filtering operation according to the local character of the speech spectrum. The idea of filtering speech with a "formant-equalized" frequency response, or even the idea of enhancing noisy speech with a filter having a speech-like frequency response, at least dates back to a U.S. patent by Schroeder in 1965 [10]. In 1981, Sondhi *et al.* used Schroeder's idea of a "formant-equalized" frequency response in a speech enhancement system [11]. In 1982, Malah and Cox reported the use of pitch-adaptive comb filtering as a speech enhancement technique [12].

To the best of our knowledge, adaptive postfiltering as a postprocessing technique for speech coding was first proposed in 1981 by Smith and Allen for enhancing the output of an adaptive delta modulation (ADM) coder [13]. The postfilter they used was an adaptive low-pass filter implemented by a short-time Fourier analysis/synthesis method. The cutoff frequency of the low-pass filter was adaptive and was chosen so that all spectral components above this frequency constituted only 1% of the total energy of the input signal. This adaptive cutoff frequency needed to be transmitted as side information. By eliminating the "out-of-band" high frequency noise, this postfiltering technique improved the speech quality of a 16 kb/s ADM coder to the extent that it was comparable to a 24 kb/s ADM coder without postfiltering [13]. Jayant extended this postfiltering idea to the adaptive differential pulse code modulation (ADPCM) coder. [14]. Instead of the frequency-domain approach, he switched between a bank of four fixed-bandwidth low-pass finite impulse response (FIR) filters and achieved a similar perceptual improvement in ADPCM-coded speech.

The use of postfiltering for speech coding did not become popular until 1984 when Ramamoorthy and Jayant proposed a new postfiltering technique described in [15] and in a U.S. patent [16]. A specific postfilter for 24 kb/s ADPCM was shown in [15], which also describes using the technique to

Manuscript received February 24, 1994; approved May 9, 1994. This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the National Aeronautics and Space Administration. The associate editor coordinating the review of this paper and approving it for publication was Dr. Spiros Dimolitsas.

J.-H. Chen was with the Department of Electrical and Computer Engineering, University of California, Santa Barbara. He is now with the Speech Coding Research Department, AT&T Bell Laboratories, Murray Hill, NJ 07974 USA.

A. Gersho is with the Center for Information Processing Research, Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA.

IEEE Log Number 9406780.

enhance speech degraded by additive white Gaussian noise. The ADPCM postfilter proposed in [15] moves the poles and zeros of the synthesis filter radially toward the origin by suitably chosen factors. Such a postfilter reduces the perceived level of coding noise [15]. However, if the coding noise of ADPCM is high, sufficient noise reduction requires a "strong" postfilter, which makes speech sound muffled [17] (similar to a low-pass filtering effect when the filter cutoff frequency is lower than the effective signal bandwidth). This postfiltering technique has been used in many applications, including the enhancement of 12 kb/s subband-coded speech as well as 24 and 16 kb/s ADPCM-coded speech [18].

In 1986, Yatsuzuka, Iizuka, and Yamazaki were the first to combine adaptive postfiltering and noise spectral shaping in a speech coder—in this case a 4.8 to 16 kb/s variable-rate APC coder [19]. Yatsuzuka *et al.* were also the first to propose explicitly an additional long-term postfilter section based on the pitch periodicity in speech. Their short-term and long-term postfilter sections were respectively obtained by moving the poles of short-term and long-term synthesis filters of APC toward the origin. This all-pole postfilter had the same muffling (or low-pass) effect mentioned above. An all-pole short-term postfilter for a sequentially-adaptive APC coder was also reported by Zarkadis and Evans in 1987 [20].

In his 1987 thesis [1], Chen described a postfilter which significantly reduced the low-pass effect. This postfilter was described in a U.S. patent [21]. The postfilter proposed in [1] contained elaborate long-term and short-term postfilter sections which achieved significant noise reduction without making the speech sound muffled. The short-term postfilter section of this postfilter was reported by Chen and Gersho [22] in 1987. At the same time, Kroon and Atal proposed the use of postfiltering in a CELP coder [23]. The postfilter they used was essentially the same as the postfilter proposed by Yatsuzuka *et al.* [19] for APC. Also at the same time, Veeneman and Mazor described an improved version of Malah and Cox's pitch-adaptive comb filter [12], with both coefficients and pitch period adapted for enhancing block-coded speech [24].

Since 1987, the use of our postfiltering algorithm [1], [22] in CELP-like coders has become very popular. Recently, different variations of our postfiltering algorithm have been incorporated into several national and international speech coding standards. These include the U.S. Federal Standard 4.8 kb/s CELP (FS1016) [25], the North American digital cellular radio standard 8 kb/s VSELP (IS-54) [26], [27], the Japanese digital cellular radio standard 6.7 kb/s VSELP (JDC), and the recently adopted CCITT standard 16 kb/s low-delay CELP (Recommendation G. 728) [28], [29]. Recently, a frequency domain method for adaptive postfiltering to suppress noise in spectral valleys was reported by Wang *et al.* [30].

In this paper, we present for the first time a complete description of our original postfiltering algorithm in [1] and the underlying ideas that motivated its development. We start with an explanation of the principle and philosophy of our postfilter design in Section II. This is followed by a description of the short-term postfilter and the long-term postfilter in Sections III and IV, respectively. Next, we describe in Section V the structure and operation of the combined postfilter (with both short-term and long-term sections). In Section VI, we comment on the performance of the postfilter. We then discuss in Section VII the variations of our postfiltering algorithm that are currently used in the speech coding standards mentioned above. Our conclusions are given in Section VIII.

II. Noise Masking and Postfiltering

The classical Wiener theory of optimal filtering tells how to optimally filter a noise contaminated signal to minimize the noise power at the filter output. The theory shows that for a signal with power spectral density $S(\omega)$ contaminated by independent additive noise with spectral density $N(\omega)$, the optimal filter transfer function for minimizing mean squared error (MSE) between the filter output and the original signal is given by $H(\omega) = S(\omega)/[S(\omega) + N(\omega)]$. See, for example, [31]. Thus, in frequency bands where the signal-to-noise power density ratio (SNR) is large, the filter gain is approximately unity and in bands where the SNR is small the filter gain is very small. For postfiltering of coded speech, this theory suggests that we seek a filter whose transfer function has a magnitude that depends on the SNR at each frequency and that, at least qualitatively, follows the above behavior. Such a filter would necessarily be adaptive in order to track the time-varying spectral character of the speech signal. Of course, the performance objective should really be perceived quality rather than MSE. Therefore, even if the ideal Wiener filter could be computed, it would not be optimal for speech enhancement. Nevertheless, the theory provides a conceptual starting point for the search for an effective postfiltering technique. Perceptual considerations are needed to find an effective trade-off between noise reduction and signal distortion resulting from a postfiltering operation.

In [15], Ramamoorthy and Jayant explained from an intuitive perspective (rather than from a Wiener filtering perspective) why adaptive postfiltering could reduce perceived noise. In this section, we give another explanation that takes into account auditory masking of noise, based on established properties of the human hearing system. We also describe the general philosophy of our postfilter design.

Given a pure tone with a certain frequency and intensity, for a normal listener there is a masking threshold function associated with this tone such that if noise is added to the tone and the power spectrum of the noise is strictly below the masking threshold at all frequencies, that noise will be inaudible, i.e., it will be completely masked by the tone [9]. In general, the masking threshold has a peak at the frequency of the tone, and monotonically decreases on both sides of the peak. This means the noise components near the tone frequency are allowed to have higher intensities than other noise components that are farther away from that frequency while remaining inaudible.

Some limited studies have also been performed on suprathreshold masking that reduces the loudness of the noise rather than making the noise completely inaudible [3]. In this case, a pulsating narrow-band noise burst is above the masking threshold and is partially masked by a masker tone (i.e., has a reduced noise loudness). From the experimental data in [3], it

can be seen that for a given loudness of the partially masked noise, the intensity of the noise varies as a function of the difference between the center frequency of the narrow-band noise and the frequency of the masker tone. Such a function generally has a shape similar to that of the masking threshold function. Consequently, even for low-bit-rate speech coding when suprathreshold masking is present and it is difficult to make the noise inaudible, the masking threshold function still provides a useful guideline for reducing noise loudness.

A short segment of a speech signal can be considered as a superposition of many sine waves. If each of these sine waves were presented alone to a normal listener, there would be an associated masking threshold function with a peak at the frequency of that sine wave. When all such sine waves are superimposed, their associated threshold functions must also superimpose. Exactly how these functions interact with each other is unknown. However, no matter how complicated the interaction might be, there must exist an overall masking threshold function for the given segment of speech signal such that an added noise will be inaudible if its power spectrum is below the threshold at all frequencies. The overall masking threshold function follows to some extent the spectral peaks and valleys of the speech spectrum. (The suprathreshold masking curves for limiting noise to a given level of subjective loudness will be similar in shape.) This characteristic behavior of the masking threshold function is more commonly associated with the spectral envelope of speech. However, by spectral peaks, we are referring not only to the formant peaks, but also to the pitch harmonic peaks for voiced speech. In other words, we believe that at least at the low frequency end, the masking threshold function also follows the pitch harmonic peaks and valleys to some extent.

There is little psychophysical evidence to justify that superimposing tone masking curves will give the same qualitative behavior for pitch harmonic peaks. However, at least at the lower frequencies some justification can be given. Specifically, for the first few critical bands at the low frequency region of the spectrum, the bandwidths are only 100 Hz or slightly higher[1]. Hence, except for very low pitch male voices, it is not likely that two or more pitch harmonics will fall within a single critical band at the low frequency end, and therefore our ears should have enough frequency resolution there to resolve two adjacent pitch harmonic peaks. The effectiveness of our long-term postfilter appears to validate the exploitation of masking for pitch harmonics.

If a speech coder can push the coding noise below the masking threshold function at all frequencies and maintain this over time as the speech spectrum changes, then the coded speech will be noise-free as far as our auditory perception is concerned. In practice, however, such an ideal coder is quite difficult to develop, especially at low bit-rates. Noise spectral shaping may help to obtain the desired shape of the noise spectrum. However, in most cases, lowering noise components at certain frequencies can only be achieved at the price of increased noise components at other frequencies [2]. Therefore, at very low encoding rates when the average level of coding noise is quite high, it is very difficult, if not impossible, to force noise below the threshold at all frequencies. The situation is similar to stepping on a balloon: when we use noise spectral shaping to reduce the noise components in the spectral valley regions, the noise components near formants will exceed the threshold; on the other hand, if we reduce the noise near formants, the noise in valley regions will exceed the threshold. Hence, at low encoding rates, noise spectral shaping alone is not adequate to make the coding noise inaudible.

In speech perception, the formants of speech are perceptually much more important than spectral valley regions. Since we cannot push the noise below the threshold in both formant and valley regions at low encoding rates, a good strategy is to sacrifice valley regions and preserve the formants. This can be done in analysis-by-synthesis coders by tuning the perceptual weighting filter so that it keeps the noise below the masking threshold in formant regions. Of course, in doing so, the noise components in some of the valley regions may exceed the threshold. However, these noise components can later be made inaudible by attenuating them with a postfilter. In performing such attenuation, the speech components in valley regions will also be attenuated. Fortunately, the *just noticeable difference* (JND) for the intensity of spectral valleys can be as large as 10 dB [32]. In other words, the intensity of spectral valleys can be altered by as much as 10 dB before our ears can detect the difference. Therefore, by attenuating the components in spectral valleys, the postfilter only introduces minimal distortion in the speech signal, but it could achieve a substantial noise reduction.

As an example, in the upper plot of Fig. 1, we show the spectrum of a segment of speech sampled at 8 kHz. Suppose during speech encoding we have used noise spectral shaping in such a way that the noise components around spectral peaks are below the masking threshold while the noise components in valley regions are not. Then, most of the perceived coding noise comes from spectral valleys, including the valleys between pitch harmonic peaks. In this case, a useful postfilter may have a frequency response shown in the lower plot of Fig. 1. This postfilter attenuates the frequency components between pitch harmonics as well as the components between formants. An important feature of this frequency response is that the three spectral envelope peaks corresponding to the three formants have roughly the same height. This feature ensures that the relative intensity of the three formants will remain roughly unchanged after postfiltering. This is essential to avoid the undesirable low-pass effect normally associated with previous postfiltering schemes.

The frequency responses of previously developed postfilters often have an overall slope, or spectral tilt, which tends to follow the tilt of the speech spectrum. For voiced speech, the spectral envelope has a low-pass spectral tilt with roughly 6 dB per octave spectral fall-off. This results from the net effect of the glottal source low-pass character and the lip radiation high frequency boost. Since speech quality is predominated by voiced sounds, many previous postfilters had low-pass spectral tilt most of the time. (The postfilter proposed in [13] is an

freq. res. của bộ lọc hình 1

voiced sound : âm hữu thanh

[1] Here we are referring to the critical bandwidths listed in Table I of Scharf's chapter in [8]. Scharf gives the following definition of the critical band: "As a purely empirical phenomenon, the critical band is that bandwidth at which subjective responses rather abruptly change."

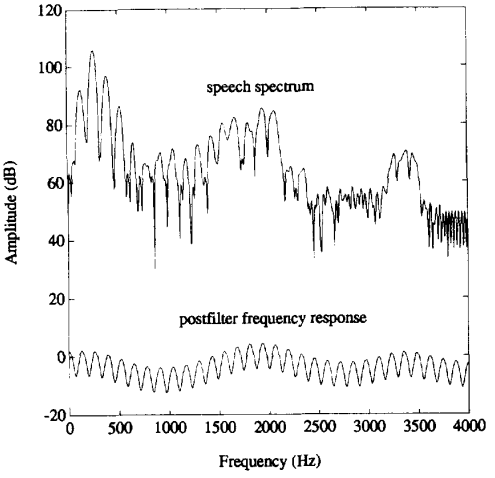

Fig. 1. An example of the speech spectrum and the corresponding postfilter frequency response.

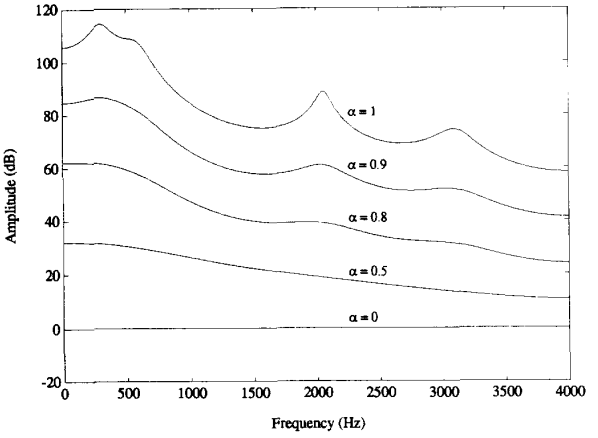

Fig. 2. An example of the frequency response of the modified LPC synthesis filter $1/[1 - P(z/\alpha)]$ for different values of α. Adjacent plots are separated by a 20 dB offset to enhance visibility.

exception.) Thus, speech filtered by these postfilters often sounds muffled.

Our goal was to develop a postfilter with attenuation between formant peaks and pitch harmonic peaks but without a spectral tilt (as in the example of Fig. 1). To accomplish this goal, we use two stages of postfiltering: a long-term postfilter and a short-term postfilter with spectral tilt compensation. The short-term postfilter has a frequency response similar to the spectral envelope of the frequency response in Fig. 1. The long-term postfilter adds the fine structure (closely spaced peaks) to the overall frequency response. These two filter stages and the combined postfilter are described in the next three sections.

III. Short-Term Postfilter

chức năng của bộ short-term

The frequency response of an ideal short-term postfilter should follow the peaks and valleys of the spectral envelope of speech without giving an overall spectral tilt. In a predictive speech coder employing linear prediction, the synthesis filter (often called the *LPC filter*) has a frequency response which closely follows the spectral envelope of the input speech. Therefore, it is natural to derive the short-term postfilter from the LPC predictor.

Let the transfer function of the LPC predictor be $P(z) = \sum_{i=1}^{M} a_i z^{-i}$, where a_i is the ith LPC predictor coefficient and M is the LPC predictor order, which is typically chosen as 10. The corresponding LPC synthesis filter has a transfer function of $1/[1 - P(z)]$, and its frequency response is often referred to as the *LPC spectrum*. The plot at the top of Fig. 2 shows an example of such an LPC spectrum for a voiced sound.

If we scale down the radii of the poles of the LPC synthesis filter by a factor of α where $0 < \alpha < 1$ (that is, moving the poles radially toward the origin of the z-plane), then the corresponding modified filter has a transfer function of $1/[1 - P(z/\alpha)]$, where $P(z/\alpha) = \sum_{i=1}^{M} a_i \alpha^i z^{-i}$. The poles of $1/[1 - P(z)]$ are inside the unit circle since the LPC synthesis filters used in practical coders are stable filters. Therefore, the poles of $1/[1 - P(z/\alpha)]$ are not only inside but also farther away from the unit circle. Consequently, the frequency response of $1/[1 - P(z/\alpha)]$ has lower peaks with wider bandwidth than that of $1/[1 - P(z)]$. In Fig. 2, we show the frequency responses of the filter $1/[1 - P(z/\alpha)]$ for $\alpha =$ 1, 0.9, 0.8, 0.5, and 0. As can be seen in Fig. 2, the frequency response becomes smoother and flatter as α decreases toward zero.

chọn alpha như nào?

As discussed above, the postfilter proposed in [15] for ADPCM moves the poles and zeros of the 2-pole, 6-zero synthesis filter toward the origin. If this idea is used in an LPC synthesis filter, the short-term postfilter will have the form $1/[1 - P(z/\alpha)]$. This form of short-term postfilter was indeed used by Yatsuzuka *et al.* [19] and Kroon and Atal [23]. Such a postfilter does reduce the perceived noise level. However, when coding noise is high, sufficient noise reduction is accompanied by muffled speech. This is due to the fact that the frequency response of this postfilter generally has a low-pass spectral tilt for voiced speech, as can be seen in Fig. 2.

thêm các điểm cực lên tử số (tăng số điểm 0) để giảm độ dốc

To reduce the spectral tilt of the all-pole postfilter $1/[1 - P(z/\alpha)]$, we added M zeros with the same phase angles as the M poles. The transfer function of the resulting pole-zero postfilter has the form

$$H_s(z) = \frac{1 - P(z/\beta)}{1 - P(z/\alpha)}, \quad 0 < \beta < \alpha < 1. \tag{1}$$

The frequency response of $H_s(z)$ can be expressed as

$$20 \log |H_s(e^{j\omega})| = 20 \log \frac{1}{|1 - P(e^{j\omega}/\alpha)|} - 20 \log \frac{1}{|1 - P(e^{j\omega}/\beta)|}. \tag{2}$$

Therefore, in the logarithmic scale, the frequency response of $H_s(z)$ is simply the difference between the frequency responses of two modified LPC synthesis filters $1/[1-P(z/\alpha)]$ and $1/[1 - P(z/\beta)]$.

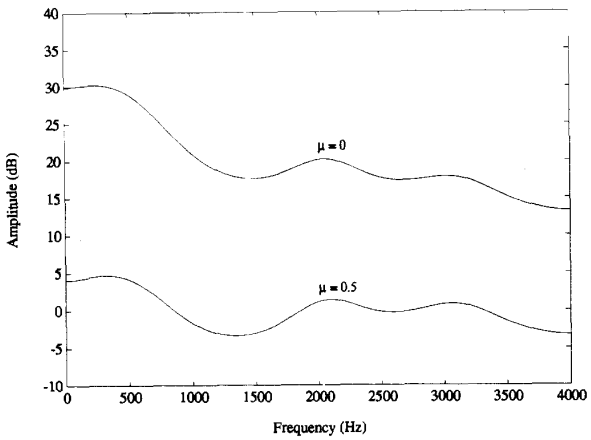

Fig. 3. Frequency response of the short-term postfilter $[1 - \mu z^{-1}]$ $[1 - P(z/\beta)]/[(1 - P(z/\alpha)]$ corresponding to the LPC spectrum in Fig. 2. The two plots were obtained with $\alpha = 0.8$, $\beta = 0.5$, and $\mu = 0$ or 0.5. The two plots are separated by a 20 dB offset to enhance visibility.

alpha, beta phụ thuộc vào bộ coder được sử dụng Hệ số tối ưu cần được đo đạc qua thực nghiệm

The optimal values of α and β depend on the bit-rate and the type of speech coder used, and they generally need to be determined empirically based on subjective listening tests. Our postfilter was originally developed for the 4.8 kb/s vector APC (VAPC) coder [22]. For that coder, we chose the parameters α and β to be 0.8 and 0.5, respectively. From Fig. 2, we see that the response of $1/[1 - P(z/\alpha)]$ for $\alpha = 0.8$ has both spectral tilt and formant peaks (although greatly smoothed), while the response for $\alpha = 0.5$ has spectral tilt only. Thus, with $\alpha = 0.8$ and $\beta = 0.5$ in (2), we can remove the spectral tilt to a large extent by subtracting the response for $\alpha = 0.5$ from the response for $\alpha = 0.8$. The upper curve in Fig. 3 shows the resulting postfilter frequency response. (Note that the vertical scale in Fig. 3 has been amplified relative to Fig. 2.)

In informal listening tests, we found that the low-pass effect was significantly reduced after the numerator term $[1 - P(z/\beta)]$ was included in the transfer function $H_s(z)$. However, the filtered speech was still slightly muffled. To further reduce the low-pass effect, we added a first-order filter with a transfer function of $[1 - \mu z^{-1}]$ in cascade with $H_s(z)$. The parameter μ was chosen to be 0.5 for the 4.8 kb/s VAPC coder. Such a filter provided a slightly high-pass spectral tilt and thus helped to reduce the low-pass effect. The lower curve in Fig. 3 shows the overall frequency response of the cascaded filter, which has the spectral tilt further reduced.

The first-order filter $[1 - \mu z^{-1}]$ can be made adaptive to better track the spectral tilt of $H_s(z)$. In computer simulations, however, we found that a fixed filter with $\mu = 0.5$ gave quite satisfactory results. Therefore, for the VAPC postfilter, we used a fixed value of μ for simplicity.

IV. Long-Term Postfilter

chức năng của bộ long-term

The function of a long-term postfilter is to attenuate frequency components between pitch harmonic peaks. Again, no overall spectral tilt should be introduced. Such a long-term postfilter can be derived from the pitch predictor typically used in predictive coders like APC or CELP, because the pitch predictor contains the information about the pitch period and the degree of periodicity.

In APC, VAPC, or CELP coders, a three-tap pitch predictor is frequently used [2], [22], [7]. The pitch synthesis filter corresponding to such a three-tap pitch predictor is not guaranteed to be stable. Since the poles of such a synthesis filter may be outside the unit circle, moving the poles toward the origin may not have the same effect as in a stable LPC synthesis filter (in terms of spectral peak reduction and bandwidth broadening). Even if the three-tap pitch synthesis filter is stabilized, as was done in VAPC, its frequency response may have an undesirable spectral tilt. In contrast, a long-term postfilter derived from a one-tap pitch predictor does not have these problems.

Consider a one-tap pitch predictor with a transfer function of $[1 - gz^{-p}]$, where g is the predictor coefficient and p is the pitch period (in terms of number of samples). The corresponding pitch synthesis filter is given by $1/[1 - gz^{-p}]$, which has p poles with the same radius $g^{1/p}$ and uniformly spaced phase angles. Assume that g is positive (which is normally the case), then the poles are located at phase angles $0, 2\pi/p, 4\pi/p, \ldots, (p-1)2\pi/p$, which correspond to the frequencies of pitch harmonics. Therefore, to achieve the desired spectral peaks at pitch harmonic frequencies, we initially choose the long-term postfilter as $1/[1-\lambda z^{-p}]$, where $\lambda < 1$ is a suitably chosen coefficient. The upper curve of Fig. 4 shows a typical frequency response of such a postfilter with $\lambda = 0.5$ and $p = 30$.

phải ở dưới mẫu để điểm cực trùng với điểm peaks

Zeros can also be used in the long-term postfilter to provide more flexibility and more control of the frequency response. To keep the spectral peaks at the correct frequencies, the zeros should be placed at phase angles corresponding to the valleys between pitch harmonics, namely, $\pi/p, 3\pi/p, \ldots (2p-1)\pi/p$. For positive γ, the polynomial $[1 + \gamma z^{-p}]$ has roots located at such phase angles. As an example, the middle curve of Fig. 4 shows the frequency response of the filter $[1 + 0.5z^{-30}]$.

zeros ở trên tử

With both poles and zeros, the long-term postfilter can be represented by the transfer function

$$H_l(z) = G_l \frac{1 + \gamma z^{-p}}{1 - \lambda z^{-p}} \tag{3}$$

where G_l is an adaptive scaling factor. The bottom curve of Fig. 4 shows the frequency response for the example of $\gamma = \lambda = 0.25$, $p = 30$, and $G_l = 1$. Note that the pitch period p in (3) should be the "true" pitch rather than the double pitch or triple pitch sometimes produced by some pitch detectors. In Fig. 4, if p were double or triple the true pitch of 30, then the frequency response of $H_l(z)$ would have extraneous peaks between pitch harmonics. In this case, we would not get sufficient attenuation between pitch harmonics, which defeats the purpose of the long-term postfilter.

True Pitch

We now describe how γ, λ, and G_l are determined. The discussion above is based on the assumption that a voiced speech frame is encountered. In practice, however, there are also unvoiced frames as well as transition frames. Appropriate values of γ, λ, and G_l should be chosen according to the "voicing"information, or degree of periodicity in speech.

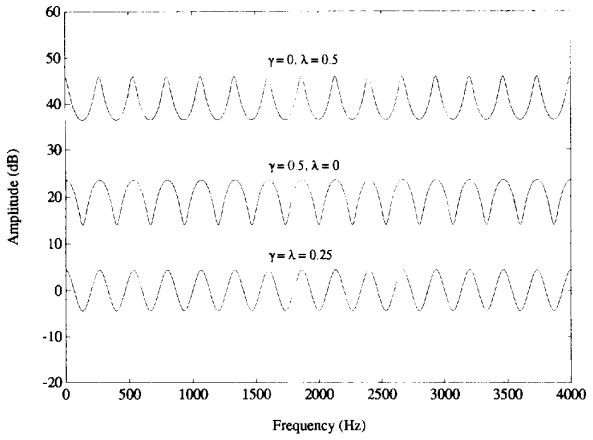

Fig. 4. Frequency response of the long-term postfilter $G_l[1 + \gamma z^{-p}]/[1 - \lambda z^{-p}]$. In all three plots, $p = 30$ and $G_l = 1$. Adjacent plots are separated by a 20 dB offset to enhance visibility.

A. Coefficient Determination

We found that the tap weight g of the one-tap pitch predictor is a good indicator of voicing: It tends to be close to unity during steady-state voiced speech, and it approaches zero during unvoiced speech. In addition, at the beginning of some voiced speech segments where the amplitude gradually builds up, g tends to be greater than unity. On the other hand, in the trailing edge of some voiced segments where the amplitude gradually decreases, g tends to be less than unity. In both cases, g is a rough approximation of the ratio between the waveform amplitude of a pitch period and that of its preceding pitch period. Therefore, the tap weight g provides useful information about voicing and the change in the speech waveform envelope. For a three-tap pitch predictor $P_l(z) = b_1 z^{-p+1} + b_2 z^{-p} + b_3 z^{-p-1}$, we found that $b = (b_1 + b_2 + b_3)$, the sum of the three tap weights, plays a similar role. That is, it tends to be unity for voiced speech, zero for unvoiced speech, and so on.

In the postfilter of VAPC, we used b as the voicing indicator since a three-tap pitch predictor is used in this coder and its parameters are available at the decoder. On the other hand, for coders with a single-tap pitch predictor, it is natural to use the tap weight g as the voicing indicator since it is readily available. For speech coders without a pitch predictor at all, and for speech coders that do not transmit the true pitch (such as those using an adaptive codebook, e.g., [33]), a long-term postfilter can still be used, but a separate pitch analysis on coded speech is needed as will be discussed later. In these cases, the single tap weight g can be used as the voicing indicator, since it is easier to compute and probably provides more accurate voicing information than $b = (b_1 + b_2 + b_3)$.

We define a parameter x as the *voicing indicator* for the postfilter and we take x to be either g or b. Then, the factors γ and λ are determined as follows:

$$\gamma = C_z f(x), \quad \lambda = C_p f(x), \quad 0 < C_z, C_p < 1 \qquad (4)$$

with

$$f(x) = \begin{cases} 0 & \text{if } x < U_{\text{th}} \quad \text{unvoiced} \\ x & \text{if } U_{\text{th}} \le x \le 1 \quad \text{voiced} \\ 1 & \text{if } x > 1 \quad \text{voiced} \end{cases} \qquad (5)$$

where U_{th} is a threshold value for turning off the long-term postfilter during unvoiced speech.

The values of C_z and C_p are fixed and they control the radii of zeros and poles, respectively. As they get closer to unity, the zeros and poles get closer to the unit circle, and the magnitude difference between spectral peaks and valleys also gets larger. Thus, C_z and C_p control the relative attenuation of spectral valleys between pitch harmonics. The term $f(x)$ allows the filter to dynamically adjust the amount of such attenuation according to the degree of speech periodicity indicated by x. When x is greater than unity, we have to clip $f(x)$ to unity to prevent the poles and zeros from being on or outside the unit circle. On the other hand, when x is less than the threshold U_{th}, which is typically chosen as 0.6, we treat the corresponding frame as unvoiced and set $f(x)$ to zero. This in effect omits the long-term postfiltering operation. The reason is that we do not want to introduce any artificial periodicity into unvoiced speech. For x between 1 and U_{th}, we consider the current frame of speech to be voiced, and the long-term postfilter coefficients γ and λ are proportional to x. This in effect adjusts the degree of long-term postfiltering according to the degree of periodicity in the speech waveform.

B. Gain Control for Long-Term Postfilter

Appropriate gain control is very important for the long-term postfilter to function effectively. For those speech frames corresponding to unvoiced speech or most consonants, the voicing indicator x is typically less than U_{th}; thus, γ and λ in (3) are zero, and the long-term postfilter is effectively turned off. In this case, if G_l is held fixed at unity, then the power of the speech signal remains the same after long-term postfiltering. On the other hand, for steady-state voiced frames (corresponding to vowels), if $G_l = 1$, the signal power is generally amplified by the long-term postfilter, because the waveform in each pitch period is reinforced by the waveform in the last pitch period according to (3). This difference in the postfilter power gain for consonants and vowels leads to a reduced volume of consonants relative to vowels. (Keeping consonants the same while amplifying vowels has the same perceptual effect as attenuating the consonants while keeping vowels the same.) When this happens, the speech quality suffers because the consonants are overwhelmed by the vowels, giving a perception that the speech is not spoken clearly. The speech intelligibility is also reduced as a result. Such degradation in speech quality and intelligibility can be avoided if we can accurately estimate the power gain of the filter during voiced frames and adjust G_l accordingly to bring the power gain back to near unity.

A common method for estimating the power gain of a filter is to compute the energy of the impulse response of the filter. However, this method is valid only for impulse or white noise inputs. For other inputs, the power gain of a filter actually depends on the filter input signal. For voiced speech

cách tính g

voice indicator là do bộ codec quyết định.

Sử dụng g hay b tùy vào codec

gán x = g hoặc b

U_th

G_l = 1 at unvoiced speech / consonant

giữ công suất các phụ âm trong khi khuếch đại các nguyên âm cũng tương đương với việc giảm phụ âm và tăng nguyên âm

frames, the postfilter input is nearly periodic. In this case, the power gain estimated by using the impulse response energy is inaccurate. In the following, we derive a simple formula which gives a much more accurate estimate of the power gain of voiced frames.

Suppose at time n that the input and the output of the long-term postfilter are $s(n)$ and $y(n)$, respectively. Then, from (3), we have

$$y(n) - \lambda y(n-p) = G_l[s(n) + \gamma s(n-p)]. \quad (6)$$

For voiced speech frames, we can make the assumptions

$$s(n) \approx \xi s(n-p) \quad \text{and} \quad y(n) \approx \xi y(n-p) \quad (7)$$

which mean that the waveform samples in each pitch period are roughly ξ times their corresponding samples in the preceding pitch period. As discussed earlier, g or $(b_1 + b_2 + b_3)$ can be used as a rough approximation of ξ. Denoting g or $(b_1 + b_2 + b_3)$ by x as before, we can then assume

$$s(n) \approx x s(n-p) \quad \text{and} \quad y(n) \approx x y(n-p). \quad (8)$$

From (6) and (8), we obtain

$$y(n) - \frac{\lambda}{x} y(n) \approx G_l\left[s(n) + \frac{\gamma}{x} s(n)\right] \quad (9)$$

or

$$y(n) \approx G_l\left[\frac{1+\gamma/x}{1-\lambda/x}\right] s(n). \quad (10)$$

Thus, if $G_l = 1$, the amplitude of $\{y(n)\}$ is larger than that of the $\{s(n)\}$ by a factor of approximately $(1+\gamma/x)/(1-\lambda/x)$, which is greater than 1 for reasonable values of γ, λ, and x. By examining speech waveforms before and after such long-term postfiltering, we found that this estimate of amplification factor was reasonably accurate. To cancel out this amplification effect, we can choose the scaling factor G_l as

$$G_l = \frac{1-\lambda/x}{1+\lambda/x}. \quad (11)$$

The scaling factor G_l given in (11) is simple to compute and yet very effective. In simulations, we found that after this scaling factor was used to replace a unity G_l, the power gain of voiced frames was reduced from large values to near unity. Since all frames now have roughly the same power gain, the consonants no longer have reduced relative volume, and the filtered speech sounds clearer.

C. Memory Effects

The preceding development makes use of the fact that the long-term postfilter parameters are fixed within a speech frame. Also, we did not consider interframe effects. In practice, the postfilter is time-varying with its parameters updated frame-by-frame and the internal state (memory) of the filter at the start of a frame is determined by the prior history. Thus, the filtering operation of a postfilter in a given frame may have some undesirable effects on the filtering in the following frames, especially when the pitch period is large or is changing from frame to frame. It is desirable to reduce such effects as much as possible. The all-pole portion (the denominator term) of the transfer function in (3) corresponds to a recursive infinite impulse response (IIR) filtering operation whose effect propagates into the future. On the other hand, the all-zero portion (the numerator term) corresponds to nonrecursive FIR filtering, so its filtering effect is largely confined to the current frame. We found that when used alone and for the same values of γ and λ, the all-zero portion of the long-term postfilter gave slightly clearer speech than the all-pole portion of the filter.

Originally, we started with the all-pole long-term postfilter of Yatsuzuka *et al.* [19]. Later we added the all-zero portion in order to make the long-term postfilter more general and also to have more flexibility in fine tuning the shape of each spectral peak in the frequency response of the long-term postfilter. (Such flexibility can be seen from the three different shapes of the spectral peaks in Fig. 4.) Subsequently, as mentioned above, we found that the all-pole long-term postfilter tended to make the filtered speech slightly unclear. Hence, in practical applications, we chose the factor λ to be small or even zero. On the other hand, the value of γ was typically around 0.5.

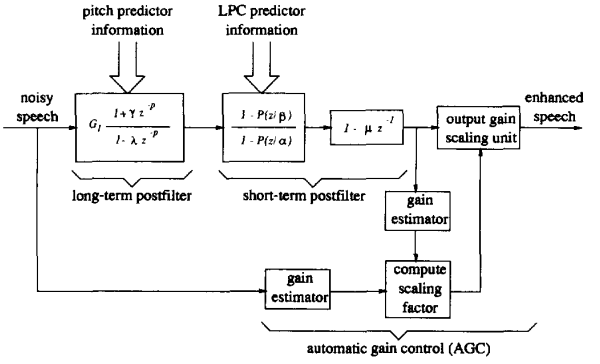

Fig. 5. Structure of the combined postfilter.

V. Combined Postfilter

The short-term and the long-term postfilters described above can be cascaded to achieve greater noise reduction. The combined postfilter is shown in Fig. 5. The noisy speech first goes through the long-term postfilter and then through the short-term postfilter. The combined transfer function for the overall postfilter is given by

$$H(z) = GG_l \frac{1+\gamma z^{-p}}{1-\lambda z^{-p}} \frac{1-P(z/\beta)}{1-P(z/\alpha)}(1-\mu z^{-1}) \quad (12)$$

which includes both long-term and short-term factors, the first-order spectral tilt filter, and gain control factors. As described earlier, $P(z)$ is the short-term linear prediction filter for the current adaptation frame, β, α, γ, λ, and μ are nonnegative design parameters, p is the pitch period, and G_l is the adaptive gain scaling factor for the long-term postfilter, given by (11). Finally, G is an adaptive scaling factor for the combined postfilter, and it is controlled by an automatic gain control (AGC) mechanism to be discussed below.

vì p là chu kỳ pitch nên epsilon có thể coi gần 1

* chu kỳ pitch nên 2 tín hiệu cách nhau 1 chu kỳ gần bằng nhau

A. Automatic Gain Control

The long-term postfilter has its own scaling factor G_l which, as mentioned above, is quite effective. However, the short-term postfilter does not have a similar scaling factor, since a simple formula like (11) is not available for the short-term postfilter. If we did not add a suitable gain control for the short-term postfilter, then different parts of the speech signal would be amplified by different amounts. (For a waveform example of this effect, see Fig. 17(a) of [18].) In general, the power gain of the short-term postfilter would be high for those speech frames where the prediction gain of the LPC predictor is high, and vice-versa. Therefore, without proper gain control, the short-term postfilter output signal will have irregular gain variations that are strongly dependent on the input signal. This gives what some people have called the "amplitude modulation effect," and it makes the output speech sound unnatural. To avoid such undesirable gain variations, we added *automatic gain control* (AGC) at the output of the short-term postfilter.

The purpose of AGC is to assure that the postfiltered speech has roughly the same power as the unfiltered noisy speech. Yatsuzuka *et al.* [19] proposed the use of an adaptive gain factor but did not indicate how it would be computed. Our postfilter AGC scheme is described below. We estimate the power of the unfiltered and filtered speech separately, then use the ratio of the two values to determine a suitable scaling factor. The speech power is estimated by an exponential-average estimator. Let $\sigma_1^2(n)$ and $\sigma_2^2(n)$ be the estimated power at the time index n for unfiltered speech $s(n)$ and filtered speech $r(n)$, respectively. Then, the two estimated power values are given by

cách ước lượng hệ số công suất của tín hiệu chưa lọc và sau khi lọc.

Từ đó tính ra hệ số cân bằng

$$\sigma_1^2(n) = \zeta\sigma_1^2(n-1) + (1-\zeta)s^2(n) \quad (13)$$

$$\sigma_2^2(n) = \zeta\sigma_2^2(n-1) + (1-\zeta)r^2(n). \quad (14)$$

A suitable value of ζ is 0.99, which corresponds to *syllabic* adaptation [34] with a time constant of 25 ms. We could use a smaller value of ζ, say 0.9, to obtain *instantaneous* adaptation [34]. However, we found that for such a small ζ, the AGC introduces a clicking noise in the scaled speech.

The power estimates $\sigma_1^2(n)$ and $\sigma_2^2(n)$ are updated sample-by-sample. For each sample, we compute the ratio $\sigma_1^2(n)/\sigma_2^2(n)$ and then the square root. The resulting value $\sigma_1(n)/\sigma_2(n)$ is the desired factor used to scale the filtered speech $\{r(n)\}$. With this AGC operation, the difference between the gains of the postfilter input and output is reduced significantly.

One possible simplification of this AGC operation is to replace the power estimation by magnitude estimation. With such a simplification, (13) and (14) are replaced by

$$\sigma_1(n) = \zeta\sigma_1(n-1) + (1-\zeta)|s(n)| \quad (15)$$

$$\sigma_2(n) = \zeta\sigma_2(n-1) + (1-\zeta)|r(n)|. \quad (16)$$

Then, the scaling factor can be directly computed as $G = \sigma_1(n)/\sigma_2(n)$, the ratio of the outputs of the two magnitude estimators. This scheme eliminates the need for the square root operation.

It should be noted that even though the combined postfilter has an overall gain control unit as described above, it is still advantageous to have separate gain control for the long-term postfilter. The reason is that with $\zeta = 0.99$, the syllabic adaptation is relatively slow and cannot always follow rapid changes in signal amplitudes. (If we reduce ζ to have instantaneous adaptation, then it introduces a clicking noise, as noted earlier.) For voiced speech, usually both the long-term postfilter and the short-term postfilter tend to amplify the speech signal. The use of separate gain control for the long-term postfilter eliminates most of the amplification due to the long-term postfilter. This will leave the overall AGC unit with a smaller gain fluctuation to correct, and thus make the AGC easier and more effective.

The adaptive postfilter we show in Fig. 5 is meant to be general; hence, how we obtain LPC predictor information and pitch predictor information is left unspecified. Depending on applications, such predictor information can be obtained by directly analyzing the coded speech or by decoding the predictor information received at the speech decoder. In fact, the lower plot in Fig. 1 is the actual frequency response of the combined postfilter in Fig. 5 (less AGC) where the LPC and pitch predictor information were obtained by directly analyzing the coded speech. The particular speech frame used to generate Fig. 1 was in the middle of a male utterance of the word "*you*." The postfilter parameters were $\alpha = 0.8$, $\beta = 0.5$, $\mu = 0.5$, $C = 0.4$, $C_p = 0.1$, $p = 62$, and $x = g = 0.97$.

B. Coder-Independent Enhancement of Noisy Speech

For most predictive coders such as VAPC and CELP, the transmitted data for each speech frame provides the receiver with the short-term and long-term predictor parameters needed for the postfilter. However, it is also possible to operate the postfilter without use of the transmitted LPC or pitch parameters by directly acquiring the needed parameter values from analysis operations on the decoded speech. In particular, as indicated earlier, for coders that do not transmit long-term prediction parameters, the only way to find the long-term postfilter parameters is by performing the pitch analysis in the decoder. Even when the encoder transmits the relevant parameters, it is a design option whether to use these parameters or to recompute them from the decoded speech. Usually it is simpler and more reliable to use the received parameters, however, this choice may depend on the specific coding method. It is important to recognize that, in general, the postfilter can be implemented as a separate postprocessing stage at the decoder that is independent of the particular coding method.

In addition to the quality enhancement of coded speech, our postfilter can also be used for general speech enhancement. When speech is corrupted by noise, the speech quality can be enhanced by postfiltering provided that the formant and pitch-harmonic information can be reliably extracted from the noisy speech by some suitably robust LPC and pitch analysis methods. In fact, even with simple correlation-type pitch detection and regular LPC analysis, we have observed significant quality improvement when this postfilter is applied to enhance noisy input speech.

C. Postfiltering With Analysis-by-Synthesis Coders

Suppose a postfilter is included as part of the decoder structure in a speech coding system that uses analysis-by-synthesis coding, such as CELP. Then, it is possible for the encoder to be modified so that the postfilter is included in the synthesis process for the excitation search. In other words, the postfilter can be included inside the closed loop of the excitation codebook search. In such coding schemes, the encoder searches for an excitation that will lead to a segment (subframe or vector) of synthesized (and postfiltered) speech as close as possible to the corresponding subframe. Since the postfilter is part of the actual synthesis structure used in the decoder, this might seem like the logical way to implement the search process. However, we have chosen not to do this in the VAPC coder. The reason is that the perceptually weighted mean squared error criterion, generally used in the search process, is inadequate for discriminating between the relative qualities of speech with and without postfiltering. In particular, we have found that adaptive postfiltering often reduces the SNR of coded speech while at the same time enhancing perceived quality. The perceptually weighted MSE, while better than a standard unweighted MSE, tends to be monotonically correlated with unweighted MSE and is not sufficiently sensitive to small differences in perceived quality. Thus, including postfiltering in the loop is not likely to help the search process, unless a much more effective objective distortion measure for closed loop search becomes available.

Complexity is another reason for omitting the postfilter from the encoder codebook search loop. Current CELP coders use highly efficient codebook search procedures to keep the search complexity low enough for practical use. The inclusion of a postfilter in the closed loop codebook search is likely to increase the search complexity.

Finally, we note that including the postfilter in the closed loop search would in some sense contradict the basic idea and purpose of postfiltering. Philosophically, postfiltering is a heuristically designed supplemental operation that attempts to improve the decoded speech signal, correcting for some inadequacies in the overall operation of the coder/decoder scheme. In the process of correcting for some perceptual degradation, it also inevitably introduces some other distortion. If the postfilter is included in the closed loop search, its tendency to introduce postfiltering distortion may interfere with the effort of the codebook search to minimize the weighted MSE. Or, looking at it from a different viewpoint, the codebook search may interfere with the postfilter's effort to attenuate spectral valleys, since the codebook search may attempt to find a codevector that boosts the spectral valleys to reduce the spectral mismatch (and weighted MSE) there. Including the postfilter in the closed loop search is equivalent to eliminating the use of a postfilter but instead incorporating the postfiltering method into a modified overall synthesis filter in a complicated manner.

In fact, the use of a *residual-shaping filter* similar to our short-term postfilter has been included in the closed-loop search of a CELP coder as reported by Lee and Un [35]. Also, the use of an adaptive comb filter similar to our long-term postfilter in the closed-loop search of a CELP coder was reported in [36]. In both cases, the results indicated some perceptual enhancement resulting from the modified synthesis filters. Nevertheless, until we are able to design a closed-loop coding scheme that truly optimizes perceptual quality, it is likely that postfiltering will remain a useful adjunct to speech coders.

D. Postfilter Complexity

The complexity of our proposed postfilter is quite low compared with the complexity of the current generation of speech coders, especially if the LPC predictor and pitch predictor information is readily available in the speech decoder. Even in cases where such information is extracted from noisy speech by additional predictive analyses, the overall complexity is usually only a small fraction of the total complexity of a speech coder. For example, the combined postfilter described in this section has been implemented on an AT&T DSP32 floating-point DSP chip as part of a real-time 4.8 kb/s VAPC coder [22]. It required only 0.42 million instructions per second (MIPS) on the DSP32. This is less than 10% of the total coder complexity.

VI. Performance

Since the postfilter attempts to reduce the *perceived* level of noise, it is difficult to gauge the effectiveness of postfiltering quantitatively by objective measures. Similarly, it is difficult to characterize quantitatively the perceptual distortion of speech introduced by a postfilter. The *mean opinion score* (MOS) [34] obtained from a formal subjective listening test is often regarded as the best measure of perceptual quality of speech, since it is a direct measure of the "customer satisfaction" among the potential end users of the speech coder tested. However, MOS tests are so involved and costly that they are generally not available in a university research environment. Therefore, when we developed our postfilter, the best we could do was to evaluate the noise-reduction capability of the postfilter by informal listening tests. Here we describe the results qualitatively.

First, to study the possible distortion of speech introduced by our postfilter, we performed an experiment in which the postfilter was used to filter unquantized clean speech. The pitch predictor and the LPC predictor were obtained by directly analyzing the clean speech, and the parameters of these predictors were not quantized. We found that the unfiltered original speech and its filtered version sounded essentially the same. Therefore, in this best possible case when no quantization was involved, the distortion introduced by this postfilter was essentially negligible. In contrast, when the postfilter was used to filter noisy speech produced by low bit-rate VAPC and CELP coders, the distortion produced by the postfilter became slightly more noticeable.

In general, the lower the coder's bit-rate, the more noticeable such postfiltering distortion becomes. There are two reasons for this. First, as the bit-rate decreases, the level of coding noise increases. Thus, to achieve sufficient reduction of perceived level of noise, the postfilter needs to be tuned to provide more postfiltering effect. This inevitably introduces

more postfiltering distortion to speech. The second reason is that as the bit-rate goes down, the VAPC and CELP coders generally allocate fewer bits for the LPC and pitch predictor information. This generally leads to less accurate LPC and pitch parameters, which in turn leads to less accurate postfilter parameters and more postfiltering distortion.

The postfiltering distortion strongly depends on the tuning of the postfilter parameters. For the long-term postfilter, the frequency response does not have any spectral tilt. As C_z and C_p approach unity, more noise reduction is achieved, but the filtered speech tends to lose some "crispness" at the high frequency end. As C_z and C_p approach zero, the long-term postfilter is disabled. In between these two extremes there is a continuum, and selecting $C_z + C_p = 0.5$ with C_p close to zero seems to provide useful noise reduction without excessive loss of "crispness."

For the short-term postfilter, although we tried hard to eliminate the spectral tilt in its frequency response, some residual tilt (that varies with time) is difficult to avoid. Perceptually, the effect is slight muffling when there is a residual low-pass spectral tilt, and a high-frequency boost when there is a high-pass tilt. For the pole-zero filter $H_s(z)$ in (1), if $\alpha = \beta$, then $H_s(z) = 1$and there is no net filtering effect. On the other hand, if α approaches 0.9 or 1 and β is significantly smaller than α, the filter achieves greatest noise reduction, but the speech sounds synthetic, with an LPC vocoder type of sound character, and the muffling effect is very noticeable. Again, in between these two extremes there is a balance somewhere. For the 4.8 kb/s VAPC coder, the choice of $\alpha = 0.8$, $\beta = 0.5$, and $\mu = 0.5$ gives significant noise reduction without much synthetic quality or muffling. For 8, 9.6, and 16 kb/s VAPC coders, the coded speech is less noisy, and hence less postfiltering is needed. In such a case, we used $\alpha = 0.7$, $\beta = 0.4$, and $\mu = 0.4$.

In the paragraphs above, we attempted to describe the kinds of speech distortion caused by the postfilter. However, it should be emphasized that when the postfilter is properly tuned for a given speech coder, it can keep the speech distortion at a barely noticeable level while still achieving significant reduction of coding noise. The perceptual impact of the noise reduction is much greater than that of the speech distortion introduced by the postfilter. As a result, when the unfiltered noisy speech and its postfiltered version were both presented to a group of listeners, nearly all listeners preferred the postfiltered speech.

As to the noise reduction capability, our informal listening also showed that, when used alone, either the long-term section or the short-term section of our postfilter could effectively reduce the perceived level of noise, although the short-term postfilter achieved slightly more noise reduction. When the two postfilter sections were cascaded as described in Section V, the noise reduction of the combined postfilter was greater than either of the two used alone. However, the incremental improvement of speech quality due to the addition of a second postfilter section was not as significant as the quality improvement of that section when used alone.

When the combined postfilter was used in the VAPC decoders, the subjects who participated in the informal listening tests unanimously agreed that the noise reduction provided by the postfilter was perceptually very noticeable. For 9.6 kb/s VAPC, the postfilter reduced the coding noise to an almost inaudible level. For 4.8 kb/s VAPC, although the coding noise was still audible after postfiltering, the noise reduction provided by the postfilter made the coded speech much more pleasant to listen to.

One interesting and noteworthy observation is that when the long-term postfilter was used alone, it clearly offered more noise reduction for female speech than for male speech. Our explanation is the following. Female speech generally has higher pitch frequencies, so the pitch harmonics of female speech are spaced farther apart along the frequency axis. Thus, the noise components between pitch harmonics are also farther away from the nearby pitch harmonics (the maskers). As a result, these noise components are not masked well by the adjacent pitch harmonics (at least not as well as in the case of male speech). In other words, the contribution of the noise components between pitch harmonics to the total perceived noise is greater for female speech than for male speech. Consequently, when such noise components are attenuated by the long-term postfilter, more reduction of perceived noise can be achieved for female speech.

In December 1992, we had a chance to test a slightly modified version of our postfilter described above in a formal subjective listening test performed at AT&T Bell Laboratories. The test used 33 listeners. The source material included six talkers' voices with intermediate reference system (IRS) frequency weighting (CCITT Recommendation P.48) and six talkers' voices without IRS weighting. There were a total of 48 stimuli, with the total length of speech exceeding five minutes. The test conditions included a simplified version of the 8 kb/s LD-CELP in [37] with and without a postfilter, as well as the CCITT G.728 16 kb/s LD-CELP coder with and without a postfilter. All four conditions were for a single encoding (without tandems). The postfilter used in G.728 is described in the next section. The postfilter used in the simplified 8 kb/s LD-CELP was very similar to the G.728 postfilter. The only differences are a longer coefficient update period and the following parameter changes optimized for a single encoding: $C_z = 0.5$, $U_{\text{th}} = 0.5$, $\alpha = 0.7$, $\beta = 0.4$, and $\mu = -0.4k_1$, where k_1 is the first reflection coefficient, whose significance in this context will be discussed in the next section. The MOS results showed that the G.728 postfilter improved the MOS by 0.15 for unweighted speech and by 0.20 for IRS-weighted speech. The postfilter for the simplified 8 kb/s LD-CELP gave even more significant improvements—it improved the MOS by 0.27 for unweighted speech and by 0.41 for IRS-weighted speech. (The G.728 postfilter achieved less MOS improvements because it was tuned for three tandems as described in the next section.) In such an MOS test, an MOS difference of 0.1 or more is generally considered statistically significant. These MOS results provide a direct proof that our postfilter indeed improves the perceptual quality of speech.

VII. Variations

After the publication of our short-term postfiltering algorithm [22], several variations of it were proposed for use with

other speech coders. The pole-zero short-term postfilter in the form of (1) was first proposed by us in 1987 [1], [22]. A similar but slightly generalized form of such a short-term postfilter was later proposed in 1988 by Ramamoorthy *et al.* for ADPCM postfiltering [18].

Kleijn *et al.* used our short-term postfilter in their 4.8 and 8 kb/s CELP coders and proposed an enhanced version of our first-order spectral tilt compensation filter [33]. While our original postfilter used a fixed all-zero filter $[1 - \mu z^{-1}]$ with $\mu = 0.5$, they proposed an adaptive version with $\mu = -0.5k_1$, where k_1 is the first reflection coefficient computed from the quantized LPC parameters. For highly correlated voiced speech, k_1 is close to -1, and μ approaches 0.5. This provides a desired high-pass filtering effect to compensate for the low-pass spectral tilt. On the other hand, for unvoiced speech, the speech spectrum tends to have a high-pass spectral tilt, and k_1 tends to be positive. Since μ becomes negative in this case, the filter $[1 - \mu z^{-1}]$ automatically changes to a low-pass filter to compensate for the high-pass spectral tilt. Kleijn *et al.* also proposed an alternative all-pole spectral tilt compensation filter $1/[1 - 0.5k_1 z^{-1}]$. They reported that this all-pole filter gave slightly higher subjective quality, although the objective performance deteriorated. Later, the enhanced version of our short-term postfilter as proposed by Kleijn *et al.* was recommended for use with the U.S. Federal standard 4.8 kb/s CELP (FS1016) [25].

Another variation of our short-term postfilter is used in the 8 kb/s VSELP coder [26] (North American digital cellular radio standard IS-54) and the 6.7 kb/s VSELP coder (Japanese digital cellular radio standard). In the postfilters of these VSELP coders, Gerson and Jasiuk [26] proposed a modified numerator polynomial in (1) and a simplified AGC scheme. To obtain their modified numerator polynomial in (1), they first calculated the denominator polynomial coefficients with $\alpha = 0.8$. The resulting coefficients were converted to the autocorrelation domain. They then used a spectral smoothing technique [38] to smooth the resulting LPC spectrum, where the equivalent bandwidth of the smoothing was 1200 Hz. This was done by applying a binomial window [38] to the autocorrelation coefficients and then converting to the numerator polynomial coefficients by using the Levinson-Durbin recursion. They claimed that the numerator polynomial obtained this way tracked the spectral tilt of the denominator polynomial more closely. However, they still retained our fixed first-order spectral tilt compensation filter $[1 - \mu z^{-1}]$ with $\mu = 0.4$.

In their AGC scheme, Gerson and Jasiuk did not perform the division and square root operations sample-by-sample as described in Section V above. Instead, they performed such operations only once a subframe (a subframe is 40 samples for 8 kb/s VSELP). Then, the resulting value was passed through a first-order low-pass filter to obtain a smoothed sample-by-sample update of the gain scaling factor. This modification of our original AGC scheme resulted in a lower computational complexity. They also used a "pitch prefilter" to filter the excitation signal before it was passed through the LPC synthesis filter. The pitch prefilter has the same form as the long-term postfilter proposed by Yatsuzuka *et al.* [19].

Finally, our postfilter was also used in an international speech coding standard—the CCITT Recommendation G.728 (16 kb/s LD-CELP) [28], [29]. Here, our original postfiltering algorithm was modified to include the adaptive spectral tilt compensation filter of Kleijn *et al.* and the simplified AGC scheme of Gerson and Jasiuk. However, unlike all previous postfilters which were optimized for a single stage of coding, this postfilter was optimized for three asynchronous tandems of G.728. Previously, it was commonly believed that postfiltering has a detrimental effect in tandem coding. However, in [39] it was demonstrated that when properly tuned for tandeming, the postfilter could actually improve the speech quality after three tandems by a very significant amount (by as much as 0.81 MOS in one French subjective test).

The postfilter for G.728 uses a simplified version of the long-term postfilter in Section IV. The simplification involves setting $\lambda = 0$ to disable the all-pole long-term postfilter in (3). The scaling factor G_l is also simplified to $G_l = 1/(1 + \gamma)$. The parameter C_z is reduced from 0.5 to 0.15 in order to provide the right amount of long-term postfiltering after three tandems. The threshold for unvoiced speech is kept at $U_{\text{th}} = 0.6$. The short-term postfilter in Section III is also used in G.728, except that the parameters have been changed to $\alpha = 0.75$, $\beta = 0.65$, and $\mu = -0.15k_1$, again to obtain the right amount of short- term postfiltering after three tandems. The AGC scheme in the G.728 postfilter is based on the magnitude estimators defined by (14) and (15), so no square root operation is required. In addition, the ratio $\sigma_1(n)/\sigma_2(n)$ is calculated only once every five samples (corresponding to one speech vector in LD-CELP). The ratio is passed through a first-order low-pass filter to obtain the sample-by-sample update of the AGC scaling factor.

An earlier version of the 16 kb/s LD-CELP coder without a postfilter met all of CCITT's performance requirements except for three asynchronous tandem encodings of speech [39]. The CCITT indicated that if the coder's tandeming performance could not be improved substantially to meet their tandeming requirement, then the coder would be recommended only for point-to-point applications and not recommended for network applications. As mentioned above, the postfilter of G.728 gave a very large improvement to the speech quality after three tandems. It enabled the 16 kb/s LD-CELP coder not only to meet but actually to exceed the tandeming requirement and achieve equivalent or better speech quality than the 32 kb/s ADPCM (G.721) for 1, 2, and 4 asynchronous encodings [29]. Hence, this postfilter of G.728 played a significant part in the acceptance of the 16 kb/s LD-CELP by the CCITT as a universal standard for network applications.

VIII. Conclusion

We have presented an adaptive postfiltering algorithm which achieves significant noise reduction without introducing significant distortion in speech. Our postfilter consists of a pole-zero long-term postfilter in cascade with a pole-zero short-term postfilter. The long-term postfilter is derived from the pitch predictor. It attenuates the spectral valleys between pitch harmonics. The short-term postfilter is derived from the LPC

distortion là méo (sự biến đổi ko mong muốn của bản thân tín hiệu)

noise là nhiễu do tác nhân bên ngoài tác động lên

predictor (including a first-order spectral tilt compensation filter) and attenuates the spectral valleys between formants. The long-term postfilter has an effective gain control of its own, while the output of the short-term postfilter has to be scaled by an AGC unit to ensure a correct power level. The complexity of the combined postfilter is typically only a small fraction of the total complexity of the current generation of speech coders. A formal subjective listening test showed that this postfilter indeed produced statistically significant improvements in the mean opinion scores. Since its initial publication, this postfilter has been tested by various researchers and has proved to be very effective in quality enhancement of coded speech. Hence, its variations have been successfully used in several national and international speech coding standards.

ACKNOWLEDGMENT

The authors acknowledge the help of P. Kroon and M. E. Perkins in coordinating and conducting the December 1992 MOS test mentioned in Section VI.

REFERENCES

[1] J.-H. Chen, "Low-bit-rate predictive coding of speech waveforms based on vector quantization," Ph.D. dissertation, Univ. Calif., Santa Barbara, Mar. 1987.
[2] B. S. Atal and M. R. Schroeder "Predictive coding of speech and subjective error criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 247–254, June 1979.
[3] M. R. Schroeder, B. S. Atal, and J. L. Hall, "Optimizing digital speech coders by exploiting masking properties of the human ear," *J. Acoust. Soc. Amer.*, vol. 66, pp. 1647–1652, Dec. 1979.
[4] J. Makhoul and M. Berouti, "Adaptive noise spectral shaping and entropy coding in predictive coding of speech," *IEEE Trans. Acoust., Speech, Signal Processing* vol. ASSP-27, pp. 247–254, Feb. 1979.
[5] B. S. Atal, "Predictive coding of speech at low bit rates," *IEEE Trans. Commun.*, vol. COM-30, no. 4, pp. 600–614, Apr. 1982.
[6] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," in *Proc. IEEE ICASSP*, Apr. 1982, pp. 614–617.
[7] M. R. Schroeder and B. S. Atal, "Code-excited linear prediction (CELP): High quality speech at very low bit rates," in *Proc. IEEE ICASSP*, Mar. 1985, pp. 937–940.
[8] J. V. Tobias, Ed., *Foundations of Modern Auditory Theory*. New York and London: Academic, 1970.
[9] D. O'Shaughnessy, *Speech Communication: Human and Machine*. Reading, MA: Addison-Wesley, 1987.
[10] M. R. Schroeder, U.S. Patent No. 3 180 936, April 1965.
[11] M. M. Sondhi, C. E. Schmidt, and L. R. Rabiner, "Improving the quality of a noisy speech signal," *Bell Syst. Tech. J.*, vol. 60, no. 8, pp. 1847–1859, Oct. 1981.
[12] D. Malah and R. Cox, "A generalized comb filter technique for speech enhancement," in *Proc. IEEE ICASSP*, Apr. 1982, pp. 160–163.
[13] J. O. Smith and J. B. Allen, "Variable bandwidth adaptive delta modulation," *Bell Syst. Tech. J.*, pp. 719–737, May–June 1981.
[14] N. S. Jayant, "Adaptive post-filtering of ADPCM speech," *Bell Syst. Tech. J.*, pp. 707–717, May–June 1981.
[15] V. Ramamoorthy and N. S. Jayant, "Enhancement of ADPCM speech by adaptive postfiltering," *Bell Syst. Tech. J.*, vol. 63, no. 8, pp 1465–1475, Oct. 1984.
[16] N. S. Jayant and V. Ramamoorthy, "Predictive communication system filtering arrangement," U.S. Patent No. 4 617 676, Oct. 14, 1986.
[17] ——, "Adaptive postfiltering of 16 kb/s-ADPCM speech," in *Proc. IEEE ICASSP*, Apr. 1986, pp. 829–832.
[18] V. Ramamoorthy *et al.*, "Enhancement of ADPCM speech coding with backward-adaptive algorithms for postfiltering and noise feedback," *IEEE J. Selected Areas Commun.*, vol. 6, pp. 364–382, Feb. 1988.
[19] Y. Yatsuzuka, S. Iizuka, and T. Yamazaki, "A variable rate coding by APC with maximum likelihood quantization from 4.8 kbit/s to 16 kbit/s," in *Proc. IEEE ICASSP*, Apr. 1986, pp. 3071–3074.
[20] D. J. Zarkadis and B. G. Evans, "16 kbit/s adaptive predictive coding of speech with adaptive postfiltering," *Electron. Letts.*, vol. 23, no. 7, pp. 358–360, 26 Mar. 1987.
[21] J.-H. Chen and A. Gersho, "Vector adaptive predictive coder for speech and audio," U.S. Patent No. 4 969 192, Nov. 6, 1990.
[22] ——, "Real-time vector APC speech coding at 4800 bps with adaptive postfiltering," in *Proc. IEEE ICASSP*, Apr. 1987, pp. 2185–2188.
[23] P. Kroon and B. S. Atal, "Quantization procedures for 4.8 kbps CELP coders," in *Proc. IEEE ICASSP*, Apr. 1987, pp. 1650–1654.
[24] D. E. Veeneman and B. Mazor, "Enhancement of block-coded speech," in *Proc. IEEE ICASSP*, Apr. 1987, pp. 193–196.
[25] J. P. Campbell, V. C. Welch, and T. E. Tremain, "An expandable error-protected 4800 bps CELP coder (U.S. Federal Standard 4800 bps voice coder)," in *Proc. IEEE ICASSP*, May 1989, pp. 735–738.
[26] I. Gerson and M. A. Jasiuk, "Vector sum excited linear prediction (VSELP) speech coding at 8 kbps," in *Proc. IEEE ICASSP*, Apr. 1990, pp. 461–464.
[27] "Dual-mode mobile station-base station compatibility standard (IS-54)," *EIA/TIA Project Number 2215*, Electronic Industries Assn., Eng. Dept., Dec. 1989.
[28] J.-H. Chen, "A robust low-delay CELP speech coder at 16 kbit/s," in *Proc. IEEE Global Commun. Conf.*, Nov. 1989, pp. 1237–1241.
[29] J.-H. Chen *et al.*, "A low-delay CELP coder for the CCITT 16 kb/s speech coding standard," *IEEE J. Selected Areas Commun.*, pp. 830–849, June 1992.
[30] F.-M. Wang *et al.*, "Frequency domain adaptive postfiltering of enhancement of noisy speech," *Speech Commun.*, vol. 12, pp. 41–56, 1993.
[31] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, 1991.
[32] O. Ghitza and J. L. Goldstein, "Scalar LPC quantization based on formant JNDs," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 697–708, Aug. 1986.
[33] W. B. Kleijn, D. J. Krasinski, and R. H. Ketchum, "Improved speech quality and efficient vector quantization in SELP," in *Proc. IEEE ICASSP*, Apr. 1988, pp. 155–158.
[34] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
[35] J. I. Lee and C. K. Un, "Improving speech quality of CELP coder," *Electron. Letts.*, vol. 25, no. 19, pp. 1275–1277, 14 Sept. 1989.
[36] S. Wang and A. Gersho, "Improving the excitation for phonetically-segmented VXC speech coding below 4 KBPS," in *Proc. IEEE Global Commun. Conf.*, 1990, pp. 945–950.
[37] J.-H. Chen and M. S. Rauchwerk "An 8 kb/s low-delay CELP speech coder," in *Proc. IEEE Global Comm. Conf.*, Dec. 1991, pp. 1894–1898.
[38] Y. Tohkura, F. Itakura, and S. Hashimoto, "Spectral smoothing technique in PARCOR speech analysis-synthesis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, pp. 587–596, Dec. 1978.
[39] J.-H. Chen, N. S. Jayant, and R. V. Cox, "Improving the performance of the 16 kb/s LD-CELP speech coder," in *Proc. IEEE ICASSP*, Mar. 1992, pp. I-69–I-72.

Juin-Hwey Chen (S'84–M'87–SM'92) received the B.S.E.E. degree from National Taiwan University, Taipei, Republic of China, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Santa Barbara (UCSB), in 1983 and 1987, respectively.

At UCSB he was a Teaching Assistant from 1983 to 1984 and a Research Assistant from 1984 to 1987. He was a Senior Engineer at Codex Corporation, Mansfield, MA, from 1987 to 1988, where he worked on speech coding and packetized voice. Since 1988, he has been with AT&T Bell Laboratories, Murray Hill, NJ, first in the Signal Processing Research Department and then in the Speech Coding Research Department. While working at Bell Labs. he created several speech coding algorithms for use in various AT&T products and services. His research there has also led to the ITU-T (formerly CCITT) G.728 speech coding standard (16 kb/s low-delay CELP). Besides speech processing, he is also interested in image processing and digital communications.

Allen Gersho (S'58–M'64–SM'78–F'82) received the B.S. degree from the Massachusetts Institute of Technology in 1960 and the Ph.D. degree from Cornell University in 1983.

He was at Bell Laboratories from 1963 to 1980, and is now Professor of Electrical and Computer Engineering at the University of California, Santa Barbara. His current research activities are in signal compression methodologies and algorithm development for speech, audio, image, and video coding. He holds patents on speech coding, quantization, adaptive equalization, digital filtering, and modulation and coding for voiceband data modems. He is co-author with R. M. Gray of the book *Vector Quantization and Signal Compression*, published in 1992 by Kluwer Academic Publishers, and is co-editor of two books on speech coding.

Dr. Gersho served as a member of the Board of Governors of the IEEE Communications Society from 1982 to 1985, and is a member of various IEEE technical, award, and conference management committees. He has served as Editor of *IEEE Communications Magazine* and as Associate Editor of IEEE TRANSACTIONS ON COMMUNICATIONS. He received NASA "Tech Brief" awards for technical innovation in 1987, 1988, and 1992. In 1980, he was co-recipient of the Guillemin-Cauer Prize Paper Award from the Circuits and Systems Society. He received the Donald McClennan Meritorious Service Award from the IEEE Communications Society in 1983, and in 1984 he was awarded an IEEE Centennial Medal. In 1992, he was co-recipient of the Video Technology Best Paper Award from the IEEE Circuits and Systems Society.