Speech Communication 12 (1993) 41–56
North-Holland

Frequency domain adaptive postfiltering for enhancement of noisy speech*

Fang-Ming Wang
INRS-Telecommunications, Université du Québec, Verdun, Quebec, Canada H3E 1H6

Peter Kabal
Department of Electrical Engineering, McGill University, Montreal, Quebec, Canada H3A 2A7 and INRS-Telecommunications Université du Québec, Verdun, Quebec, Canada H3E 1H6

Ravi P. Ramachandran
Speech Research Department, AT&T Bell Laboratories, Murray Hill, NJ 07974, USA

Douglas O'Shaughnessy
INRS-Telecommunications, Université du Québec, Verdun, Quebec, Canada H3E 1H6

Received 30 September 1991
Revised 13 August 1992

Abstract. This paper presents a new frequency-domain approach to implement an adaptive postfilter for enhancement of noisy speech. The postfilter is described by a set of DFT coefficients which suppress noise in the spectral valleys and allow for more noise in formant regions which is masked by the speech signal. First, we perform an LPC analysis of the noisy speech and calculate the log magnitude spectrum of the input speech. After identifying the formants and valleys (by a new method), the log magnitude spectrum is modified to obtain the postfilter coefficients. The filtering operation is also done in the frequency domain through an FFT and an overlap-add strategy to get the postfiltered speech. Experimental results on 8-kHz-sampled speech show that this new frequency-domain approach results in enhanced speech of better perceptual quality than obtained by a time-domain method. This new method is especially efficient in eliminating high frequency noise and in preserving the weaker, high frequency formants in sonorant sounds.

Zusammenfassung. Dieser Beitrag beschreibt eine Frequenzbereichsmethode um Nachfilter zu entwickeln für die Qualitätsverbesserung von verrauschter Sprache. Der Nachfilter wird beschrieben durch eine Menge von DFT-Koeffizienten welche das Rauschen in den Tälern des Spektrums unterdrücken und mehr Rauschen gestatten in Formantenregionen wo das Rauschsignal maskiert wird durch das Sprachsignal. Zuerst führen wir eine LPC-Analyse des verrauschten Signals durch und wir errechnen das Log-Amplitudenspektrum des Eingangssignals. Nachdem die Formanten und Täler identifiziert sind (mit Hilfe einer neuen Methode), wird das Log-Amplitudenspektrum verändert um die Nachfilter-koeffizienten zu erhalten. Der Frequenzgang des Nachfilters hat lokale Minima in den Regionen die Tälern des Spektrums entsprechen und lokale Maxima von gleicher Grösse im Bereich der Formantenfrequenzen. Das Filtern wird ebenfalls im Spektralbereich durchgeführt mit Hilfe einer FFT und eine Überlappungs-Addierungs-Strategie um das nachgefilterte Sprachsignal zu erhalten. Experimentelle Resultate zeigen daß die neue Spektralbereichsmethode in Sprachsignalen resultiert welche eine bessere perzeptive Qualität aufweisen als solche welche mit einer Zeitbereichsmethode erzielt wurden. Das Sprachsignal wurde mit einer Frequenz vow 8 kHz abgetastet. Die neue Methode ist besonders wirksam in der Beseitigung von Hochfrequenzrauschen und in der Erhaltung der schwachen hochfrequenten Formanten der Sonoranten.

Résumé. L'article présente une nouvelle approche dans le domaine fréquentiel à l'implémentation de postfiltres adaptatifs pour l'amélioration de la parole bruitée. Le postfiltre est décrit par un ensemble de coefficients TFD qui atténuent le bruit das les vallées spectrales et qui tolèrent plus de bruit dans les régions formantiques où il est masqué par le signal de parole. D'abord,

* This work was supported by the Natural Sciences and Engineering Research Council of Canada.

nous effectuons une analyse LPC du signal bruité et nous calculons le spectre d'amplitude logarithmique de la parole à l'entrée. Après avoir identifié les formants et vallées (à l'aide d'une nouvelle méthode), le spectre d'amplitude logarithmique est modifié afin d'obtenir les coefficients du postfiltre. La réponse en fréquence du postfiltre a des minima locaux dans les régions qui correspondent aux vallées spectrales et des maxima locaux d'amplitude égale aux fréquences formantiques. Le filtrage est aussi effectué dans le domaine fréquentiel à l'aide d'une TFR et d'une stratégie chevauchement-addition pour obtenir le signal postfiltré. Les résultats expérimentaux obtenus sur de la parole échantillonnée à 8 kHz montrent que cette nouvelle méthode fréquentielle produit de la parole améliorée d'une qualité perceptive meilleure que celle obtenue par une méthode temporelle. La nouvelle méthode est particulièrement efficace pour éliminer du bruit à haute fréquence et pour préserver les faibles formants à fréquence élevée des sonantes.

Keywords. Speech enhancement; adaptive postfilter; noisy speech; formant; linear prediction; spectrum.

1. Introduction

The quality and intelligibility of speech is often degraded by background noise, by coding noise, by noise due to transmission over a channel, and by the presence of speakers other than the desired speaker (O'Shaughnessy, 1987). The aim of speech enhancement techniques is to process the degraded speech such that its quality and intelligibility are improved. One approach is to use an adaptive postfilter to enhance speech signals corrupted by noise.

Consider a typical spectrum of a speech signal that has both formant peaks and spectral valleys (our speech is sampled at 8000 times per second, and thus typically has four formants present in the range of 0–3.4 kHz). For speech degraded by additive white noise, it is known that the noise in the frequency regions corresponding to the valleys contributes the most to perceptual distortion. It is also known that more noise can be perceptually tolerated in the formant regions than in the valleys: i.e., noise in the formant regions is less perceptible than noise in the valleys. The role of a postfilter is to (1) accurately track the time-varying nature of speech and (2) suppress the noise residing in the spectral valleys. The frequency response of a postfilter corresponds to a modified version of the speech spectrum in which (1) there are local minima or dips in the regions corresponding to the spectral valleys and (2) local maxima or spectral peaks of equal magnitude at the formant frequencies. The dips in the regions corresponding to the spectral valleys wili suppress the noise, thereby accomplishing noise reduction. The spectral peaks of equal magnitude at the formant frequencies ensure that there is no additional lowpass tilt in the output signal (after postfiltering), and allow for more noise in the formant regions, which is masked by the speech signal. However, some speech distortion is introduced because the relative signal levels in the formant regions are altered due to the postfiltering. In implementing a postfilter, there is a tradeoff between noise reduction and speech distortion (Jayant and Ramamoorthy, 1986; Ramamoorthy et al., 1988). Note that the filter must be adaptive due to the time-varying nature of speech.

The approaches in (Jayant and Ramamoorthy, 1986; Ramamoorthy et al., 1988; Chen and Gersho, 1987) can be classified as time-domain methods in that the postfiltering is implemented temporally as a difference equation. Therefore, the postfilter can be described by a transfer function. The frequency response of the postfilter approximates a modified version of the spectrum of the noiseless input speech. The approximation is due to two main reasons: (1) the transfer function of the postfilter depends on the small number of LPC coefficients and (2) the LPC analysis is done on the noisy speech. In this paper, we develop a frequency-domain approach to accomplish postfiltering in which the postfilter is represented by a set of DFT coefficients. The motivation for using a frequency-domain approach is two-fold. First, in contrast to the time-domain approach, our method obtains independent control over different portions of the frequency spectrum, especially those corresponding to the formant locations and spectral valleys. Second, we are able to suppress the noise which is dominant in the temporal regions of a speech utterance corresponding to very low energy or silence, which cannot be accomplished by the time-domain approach. In fact, our experimental results will show the superiority of the new method to the approach in (Ramamoorthy et al., 1988).

This paper is organized as follows. In Section 2, the time-domain approaches are described. Our

frequency-domain approach is discussed in detail in Section 3. The experimental results (which include a comparison of the time- and frequency-domain methods) are given in Section 4.

2. Time-domain methods

The postfiltering methods in (Jayant and Ramamoorthy, 1986; Ramamoorthy et al., 1988: Chen and Gersho, 1987) have been realized with the purpose of enhancing speech degraded by coding noise. A general block diagram for this is shown in Figure 1. In this section, we discuss two methods of specifying a postfilter. One is based on a modified form of the inverse pole-zero prediction error filter in Adaptive Differential Pulse Code Modulation (ADPCM). The second method involves the use of LPC models.

2.1. Inverse prediction error filter

Consider a pole-zero prediction error filter $F(z)$ given by

$$F(z)=\frac{1-A(z)}{1+B(z)}. \tag{1}$$

Such a prediction error filter can remove near-sample redundancies in the speech prior to coding. The corresponding postfilter $H(z)$ is a modified form of the inverse prediction error filter as given by

$$H(z)=\frac{1+B'(z)}{1-A'(z)}, \tag{2}$$

where $A'(z)=A(z/\alpha)$, $B'(z)=B(z/\beta)$ and $0\leqslant\alpha,\beta\leqslant 1$. The frequency response of $H(z)$ can be chosen to correspond to the spectrum of the speech. The poles and zeros of the postfilter occur at the same frequencies as those of the input speech but are radially shifted. Therefore, the formant locations are preserved. The real parameters α and β influence the formant bandwidths; high values of α and β result in sharp resonances in the filtered speech (Ramamoorthy et al., 1988).

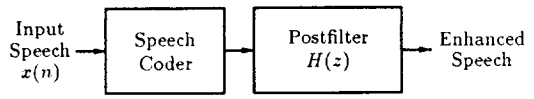


Fig. 1. Postfiltering decoded speech.

The postfilter $H(z)$ was implemented in (Jayant and Ramamoorthy, 1986; Ramamoorthy et al., 1988) as part of an ADPCM system in which $A(z)$ and $B(z)$ are second and sixth order polynomials, respectively. For the special case $B(z)=1$, we have a second order all-pole LPC model. In addition, $H(z)$ is adaptive in two ways. First, the coefficients of $A(z)$ and $B(z)$ are updated in each frame of speech to track the time-varying nature of the speech signal. With this adaptation and fixed values of α and β, adequate noise reduction results; however, this is accompanied by distortion in the output speech. A second type of adaptation results by allowing for a higher degree of postfiltering in the frames of speech that suffer from more noise. This is accomplished by varying α in each frame of speech; it is kept low for speech segments having a high signal to noise ratio (SNR). The main problem with this postfiltering scheme is that a lowpass tilt is still introduced, which contributes to speech distortion. We now describe an approach that mitigates this lowpass filtering effect.

2.2. LPC models

The transfer function of the postfilter based on an LPC model is

$$H(z)=\frac{A_M(z/\beta)}{A_N(z/\alpha)}, \tag{3}$$

where $A_P(z)$ is the inverse filter of a Pth order autoregressive process given by

$$A_P(z)=1-\sum_{i=1}^{P}a_i z^{-i}. \tag{4}$$

The denominator of $H(z)$ represents a modified version of the LPC model. The use of a model with a sufficiently high order (8 to 12, for our sampling rate of 8000 samples/s) achieves a good approximation of the input speech spectrum by providing information about formant bandwidths and locations. In addition, the factor α controls the formant bandwidths. The numerator term is a spectral tilt compensator that alleviates the lowpass filtering effect of the $1/A_N(z/\alpha)$ component.

The postfilter $H(z)$ accomplishes noise reduction by suppressing the noise around the spectral valleys but distorts the speech signal by sharpening the formant peaks. There is no apparent lowpass filtering effect, as occurs when using an inverse prediction error filter. The postfilter is adaptive in that the LPC coefficients a_i are updated in each frame; either forward or backward adaptation can be used. In (Ramamoorthy et al., 1988), fixed values of α and β are used. Moreover, for $0.5 \leqslant \alpha \leqslant 1$ and $\beta < \alpha$, the enhancement due to postfiltering is perceivable (Ramamoorthy et al., 1988). In (Chen and Gersho, 1987), a postfilter as above but with an additional highpass factor $1-\mu z^{-1}$ in the numerator is used to enhance speech at the output of an APC coder. This highpass factor further mitigates the lowpass tilt caused by the denominator term in $H(z)$.

3. Frequency-domain postfiltering

The time-domain approaches described above specify a transfer function for the postfilter and implement the filtering as a difference equation. Here, we propose a new frequency-domain method to accomplish postfiltering. The frequency-domain approach permits obtaining independent control over different portions of the frequency spectrum, especially those corresponding to the formant locations and spectral valleys. Specifically, consider Figure 2 which shows a block diagram of frequency-domain postfiltering.

In Figure 2, the postfilter is represented by its DFT coefficients $H(k)$. The coefficients $H(k)$ are multiplied by $P(k)$ which is a modified form of $X(k)$ (the DFT coefficients of the input noisy speech $x(n)$). The filtering of the input speech is performed in the frequency domain in that the filtered output is $Y(k) = P(k)H(k)$. The inverse DFT yields the postfiltered signal $y(n)$. Note that the calculation of the DFT is done by an FFT technique and that the length of the FFT is chosen to be sufficiently long so that $y(n)$ indeed represents a linear convolution of $p(n)$ and $h(n)$. The input speech, which is lowpass filtered to 3.4 kHz and sampled at 8000/s, is divided into frames of 128 samples (16 ms). The postfilter is adaptive in that $H(k)$ is updated every 128 samples.

3.1. Determination of postfilter coefficients

As in the time-domain approach, the DFT coefficients $H(k)$ of the postfilter are determined for the purposes of suppressing the noise around the spectral valleys and introducing no additional lowpass spectral tilt in the enhanced speech. As in (Ramamoorthy et al., 1988), consider Figure 3 which shows the spectrum envelopes of a voiced frame of noisy speech, of the postfilter and of the spectrum of the enhanced speech. In determining $H(k)$, the approach is to approximate the noisy speech spectrum by LPC analysis and to modify the spectrum based on formant detection such that no tilt is present (see Figure 3(b)). The enhanced speech will have deepened spectral valleys and sharpened formant peaks as compared to the original input spectrum. We will now describe in detail the various steps involved in finding the coefficients $H(k)$.

3.1.1. Calculation of log magnitude spectrum

An approximation of the speech spectrum is obtained by calculating the log magnitude spectrum of $1/A_P(z)$. The first step in calculating the log magnitude spectrum is to determine the LPC coefficients A_i and hence the filter $A_P(z)$. In each frame of speech, we use the autocorrelation method with a Hamming window of length 256 samples to perform a 16th order analysis. The autocorrelation method guarantees that $A_P(z)$ is minimum phase (Rabiner and Schafer, 1978). A window length of 256 samples includes two to three pitch periods in order to obtain accurate spectral estimates (O'Shaughnessy, 1987). Also, a Hamming window is preferred over a rectangular window, because it is a better lowpass filter, thereby leading to a better approximation of the input speech spectrum in which the formant peaks are more evident (Rabiner and Schafer, 1978; O'Shaughnessy, 1987). Since the objective is to determine formant locations and amplitudes, a 16th order analysis allows us to resolve 3–4 formants. Experiments have shown that a higher order analysis results in too many peaks in the spectrum, thereby making it difficult to identify the formant locations. Performing a lower order analysis (such as 10th order) does not result in enough peaks to resolve 4 formants.

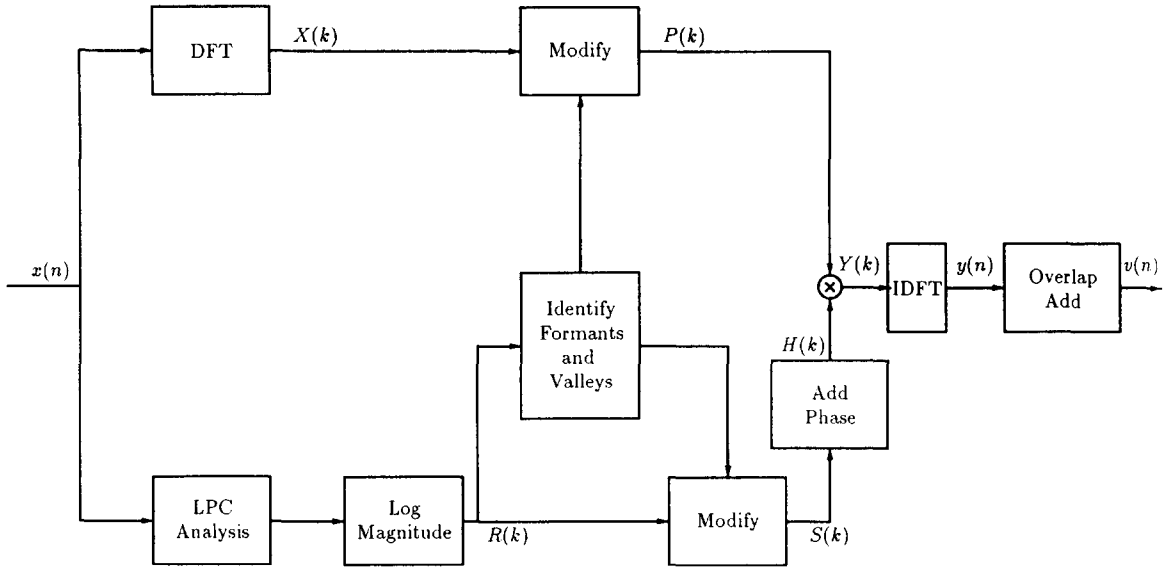


Fig. 2. Postfiltering in the frequency domain.

Given $A_P(z)$, the first step is to obtain $A_P(k)$, $k=0$ to 255, which is an FFT of $N_{PT}=256$ points of the sequence $\{1, -a_1, -a_2, \ldots, -a_p\}$. Note that a 256-point FFT is chosen since it is a power of 2, is of sufficient length to ensure that the filtered output $y(n)$ represents a linear convolution of $p(n)$ and $h(n)$, and provides for adequate frequency resolution. We then find the log magnitude spectrum $R(k)=20\log|1/A_P(k)|$. This is a discrete representation of an approximation to the speech spectrum and is used for identifying the formants, which is described next.

3.1.2. *Formant and valley identification*

The log magnitude LPC spectrum $R(k)$ (see Figure 2) is computed for a frame of speech corrupted by noise. Finding the amplitude and location of the formants is an important step in determining the postfilter coefficients $H(k)$. Although formant extraction has been performed for clean (noiseless) speech (Markel, 1972; McCandless, 1974), we face the more complicated situation of dealing with noisy speech. (Since these other formant trackers were not designed to handle noisy speech, we were unable to apply them to our speech and do a formal comparison.) Our formant tracker was designed to track the first three formants as well as possible when noise is present and when the signal level is low. For frames in which the speech signal energy is low (a frame having a low SNR, typically $\leqslant$10 dB), the amplitudes of the formants may be less than the peak value of the noise component, thereby making detection of such formants very difficult. Also, some speech segments have most of their energy at low frequencies. In such cases, the noise is dominant at high frequencies and any formants there may be undetected. This will affect the quality of the output speech in that the high-frequency components will not be restored satisfactorily.

We use a peak picking strategy to detect the formants. Given $R(k)$, we sequentially determine the local maxima and immediately decide whether or not to classify a particular peak as a formant. In each frame of speech, we find a maximum of four formants. The two major problems with a peak picking approach are that (McCandless, 1974) (1) some peaks may be spurious and (2) two formants may merge into one peak. The approach of deciding whether or not a peak corresponds to a formant is necessary to avoid classifying spurious peaks as formants. The second problem of merged peaks is not crucial for implementing a postfilter since a formant region containing merged peaks is sharpened anyway by the postfilter.

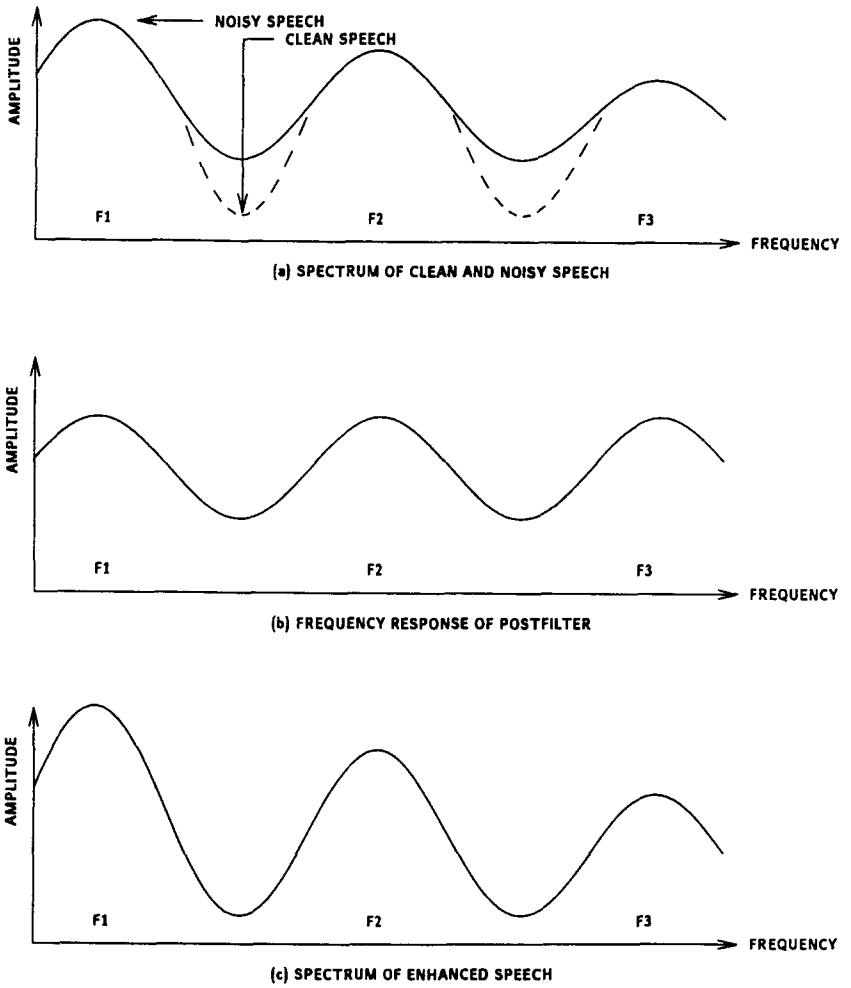


Fig. 3. Effect of postfiltering on the speech spectrum.

Before describing the algorithm for detecting the formants, there are two important quantities that must be determined. Note that only the first 129 points of $R(k)$ are significant and these points represent the range 0 to 4 kHz. First, the quantity $A_{MAX}=MAX(R(k))$, over $k=0$ to $N_{PT}/2=128$, is determined. The location (or the value of k) at which A_{MAX} occurs is denoted by L_{MAX}. Second, a quantity N_{AV} is calculated to approximate the average noise level that is present. In our database, the first ten frames of the signal correspond to a period of silence; in these frames, only noise is present. For each of these frames $m=1$ to 10, we calculate $A(m)$ as the sum of the amplitudes of the peaks of $R(k)$ divided by the number of peaks present in $R(k)$ Then, N_{AV} is the average of the ten values of $A(m)$.

We now proceed to analyze each frame of the noisy speech signal and perform three tasks. First, each frame is classified as either being unvoiced ($L_U=1$) or not unvoiced ($L_U=0$). The term, not unvoiced, includes voiced speech (either corresponding to weak or strong speech) and frames corresponding to pure noise (silence). Second, the formant amplitudes and locations are determined. Third, the amplitudes and locations of the spectral valleys are found. These three tasks are described below.

Detection of unvoiced segments

The method for identifying the unvoiced frames of speech is given below.

1. Set $L_U=1$.
2. If $A_{MAX}<2N_{AV}$, then $L_U=0$; stop.

3. If $L_{MAX} < N_{PT}/4 = 64$, then $L_U = 0$; stop.
4. Calculate A_{MX} (this quantity is discussed later).
5. If $A_{MX} < N_{AV}/2$, then $L_U = 0$; stop.

Note that the thresholds for comparison in Steps 2 and 5 were chosen empirically after examining many frames of speech.

The criterion in Step 2 indicates a frame of either pure noise or weak speech (a segment with low energy). For voiced speech (either strong or weak), the largest peak which is a formant occurs at a frequency below 2 kHz. It is for unvoiced speech (especially for the case where most of the energy is at high frequencies), that the largest peak occurs between 2.5 and 4 kHz (Markel, 1972). Therefore, satisfaction of the criterion in Step 3 is a positive indication of voiced speech. There are rare cases when segments of pure noise having much energy at high frequencies (like noise bursts) will not satisfy the criteria in Steps 2 and 3. We have to discriminate between this case and a truly unvoiced segment. The quantity A_{MX} is calculated by dividing the first 128 samples of $R(k)$ into four equal portions comprising 32 samples each. For each portion, we compute S_{R1}, S_{R2}, S_{R3} and S_{R4} as given by

$$S_{R1} = \sum_{k=0}^{31} R(k), \qquad S_{R2} = \sum_{k=32}^{63} R(k),$$

$$S_{R3} = \sum_{k=64}^{95} R(k), \qquad S_{R4} = \sum_{k=96}^{127} R(k). \tag{5}$$

Then, $A_{MX} = \mathrm{MAX}(S_{R1}, S_{R2}, S_{R3}, S_{R4})$. The major difference between unvoiced segments and segments of pure noise is that peaks occurring for unvoiced segments have a much wider bandwidth than those appearing for pure noise. In fact, it is the noise having a spectrum with a narrow peak at a high frequency that is undetected by the criteria in Steps 2 and 3. Therefore, the value of A_{MX} is higher for unvoiced segments. The criterion in Step 5 is a final test for the presence of a frame of unvoiced speech.

Determination of formant amplitudes and locations

Since our speech is bandlimited to 3.4 kHz, we only examine the points in $R(k)$ which correspond to the frequency range 0 to 3.4 kHz (i.e., the first 110 points of $R(k)$) for detecting the formants. The strategy is to sequentially examine each value of $R(k)$, locate a peak and immediately decide whether or not it is a formant. A local $R(k)$ peak is defined to exist when $R(k) \geq R(k-1)$ and $R(k) \geq R(k+1)$. The flowchart of the formant detection algorithm is given in Figure 4. In each frame of speech, the algorithm results in a total of N_F formants being found. The amplitudes of these formants are placed in the array $A_P(J)$. The index locations of $R(k)$ at which these formants occur are stored in the array $N_P(J)$.

Most of the steps in the formant detection algorithm of Figure 4 involve a comparison of A_{MAX} and N_{AV}. These comparisons are performed to differentiate between strong speech, middle amplitude speech, weak speech, unvoiced speech and pure noise before classifying a peak as a formant. Both A_{MAX} and N_{AV} depend on the SNR in that, as the SNR increases, A_{MAX} increases and/or N_{AV} decreases. The thresholds C_1, C_2 and C_3 that are used to compare A_{MAX} and N_{AV} are empirically chosen depending on the SNR. As the SNR increases, the values of C_1, C_2 and C_3 diminish. (We discovered no simple formula to obtain the values for these parameters; manual estimates were used, and the best results were obtained by allowing C_1, C_2 and C_3 to vary as a function of SNR.) The values of C_4 and C_5 are primarily chosen to detect the second formant and are sometimes used to establish the higher order formants. The typical ranges of the values of $C_{1\text{-}5}$ are $2.5 \leq C_1 \leq 4.0$, $3.5 \leq C_2 \leq 5.2$, $4.5 \leq C_3 \leq 6.5$, $1.6 \leq C_4 \leq 2.8$ and $8.0 \leq C_5 \leq 16.0$. If the SNR is 10 dB, for example, $(C_1, C_2, C_3, C_4, C_5) = (3.0, 4.5, 5.5, 2.0, 10.0)$.

In Figure 4, seven different testing blocks are numbered from 1 to 7. Consider the case when not more than one formant has been detected so far ($N_F \leq 1$). A positive test at block 3 indicates a frame of strong speech (peak energy A_{MAX} well above the noise level N_{AV}). Then, it is the test at block 6 that indicates the presence or absence of a formant (the formant candidate must be sufficiently high compared to the highest peak); this block is used primarily to detect the second formant (i.e., A_{MAX} refers to the first formant peak). A negative test at block 3 and a positive test at block 4 indicate neither strong nor weak speech (i.e., middle amplitude speech). Then, it is the test at block 7 that determines whether or not the peak corresponds to a formant; to accept a local peak as a formant, a

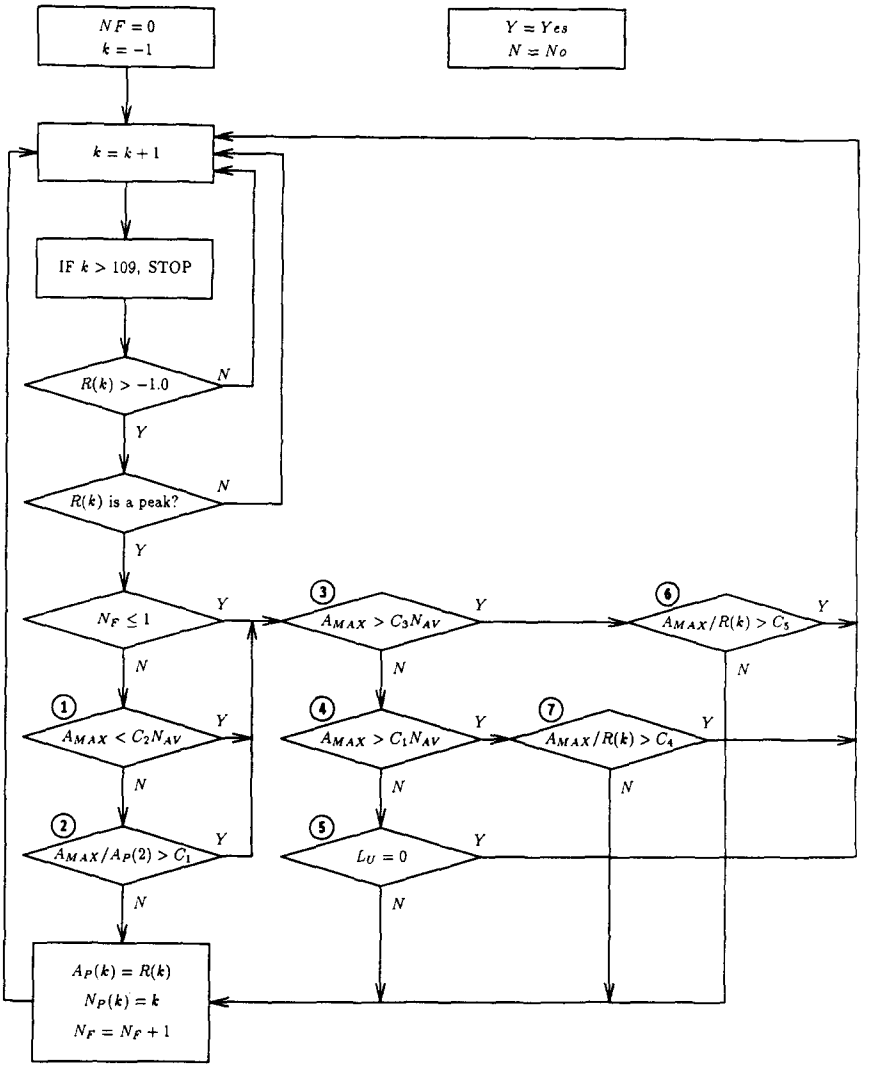


Fig. 4. Flowchart of the formant detection algorithm.

lower threshold is required in block 7 than in block 6, in the case of weaker speech (again, these tests are normally used to find the second formant). Negative tests at blocks 3 and 4 indicate weak speech, unvoiced speech or pure noise; in this case, the peak corresponds to a formant only if the frame was identified as unvoiced (block 5). Now, suppose instead that $N_F > 1$ (i.e., two or more formants already located). In a frame of voiced speech, we may encounter the problem in which the higher order formant peaks are hidden by the noise component. The tests at blocks 1 and 2 attempt to pick up these high frequency formants. We accept a candidate peak as a formant automatically if the peak energy is high enough compared to the noise (block 1), but not too high compared to the second formant peak (block 2) (if there is a big difference between the $F1$ and $F2$ peaks, higher-frequency formants are likely to be unreliably extracted in the noise background). If either test in blocks 1–2 succeeds (i.e., weak energy, or a big difference between $F1$ and $F2$ peaks), the candidate peak must pass through the normal tests (i.e., blocks 3–7).

Using the algorithm above, a total of N_F formants are found in each frame. We introduce an

additional modification that adjusts the number of formants in a particular frame to ensure a continuity in the formant trajectories from frame to frame. In our experiments, we encountered instances where the number of formants detected in a particular frame is suddenly less than the number in those previous frames. Then, in succeeding frames, the value of N_F returns to what was obtained in previous frames. For this isolated frame in which N_F is abruptly less than in previous frames, we have apparently missed one or more formant peaks at the higher frequencies due to a substantial noise component. This problem is dealt with as follows.

Suppose in a particular frame m, it is found that $N_F = 1$. If $N_F \geq 3$ in frame $m-1$ and $N_F \geq 2$ in frame $m-2$, there is an abrupt discontinuity in the formant trajectories. If the value of A_{MAX} in frame m is approximately equal to that in frame $m-1$, the value N_F is adjusted to be 2. The location of the second formant $N_P(2)$ in frame m is set to be equal to the location of the third formant peak in frame $m-1$. The corresponding amplitude $A_P(2) = R(N_P(2))$ for $R(k)$ calculated in frame m. A formant peak is reinserted at a relatively high frequency so that the postfiltering operation results in the recovery of the high frequency components of the speech signal. Similarly, an adjustment in the value of N_F from 2 to 3 in frame m is made if (1) $N_F \geq 4$ in frame $m-1$, (2) $N_F \geq 3$ in frame $m-2$ and (3) the values of A_{MAX} in frames m and $m-1$ are approximately equal. Now, $N_P(3)$ is set equal to the location of the fourth formant in the previous frame. Then, $A_P(3) = R(N_P(3))$. Lastly, the algorithm of Figure 4 can occasionally pick a total of five formant peaks. In this case, the peak with the lowest amplitude is discarded bringing N_F down to 4. Note that the method to adjust the number of formants in a particular frame is based on the number of formants detected in the previous two frames. Since there is no provision for looking ahead at succeeding frames, this approach is suitable for real-time applications.

Finding the valleys

Given the formant amplitudes $A_P(1), \ldots, A_P(N_F)$ and their corresponding locations $N_P(1), \ldots, N_P(N_F)$, we proceed to determine the amplitudes and locations of the valleys. This is needed to sharpen the formants and deepen the valleys with the postfilter. Between two formant locations $N_P(J)$ and $N_P(J+1)$, the local minimum in $R(k)$ corresponds to a valley. The amplitude of this valley is $A_V(J)$ and its location is $N_V(J)$. In this manner, $N_F - 1$ valleys are found.

The remaining issue is to examine $R(k)$ in the regions between $0 \leq k \leq N_P(1)$ and $N_P(N_F) < k \leq 109$. For $0 \leq k \leq N_P(1)$, either (1) $R(k)$ monotonically increases to $A_P(1)$ or (2) there is a local minimum of $R(k)$. In case (1), the location $k = 0$ is a valley and $N_V(0) = 0$. The corresponding amplitude is $A_V(0)$. In case (2), the local minimum in $R(k)$ is a valley with amplitude $A_V(0)$ and location $N_V(0)$. Then, $A_P(0) = R(0)$ and $N_P(0) = 0$, although this is not a true formant at zero frequency.

Similarly, the region $N_P(N_F) < k \leq 109$ is examined to establish one of two situations. If $R(k)$ monotically decreases in this region, $N_V(N_F) = 109$ and the corresponding amplitude is $A_V(N_F)$. However, this is not a valley. Otherwise, a valley is found in this region where the first local minimum of $R(k)$ occurs. In this case, $N_P(N_F + 1) = 109$ and the corresponding amplitude is $A_P(N_F + 1)$ (note again that this is not a formant).

3.1.3. Modification of the log magnitude spectrum

The log magnitude spectrum $R(k)$ is modified to become $S(k)$ such that in the postfiltered speech the formant peaks are sharpened, the spectral valleys are deepened and no additional lowpass tilt is present. The first step is to divide $R(k)$ into sections from $k = 0$ to $N_P(1)$, $N_P(1)$ to $N_P(2), \ldots, N_P(N_F)$ to 109, and finally 110 to 128. Each section is individually modified. This freedom of independently modifying different sections of $R(k)$ is exactly what the new frequency-domain approach provides for as opposed to the time-domain methods of Section 2.

We will first concentrate on the sections of $R(k)$ that correspond to actual formants, i.e., from $k = N_P(1)$ to $N_P(N_F)$. Figure 5 shows the general shapes of the envelopes of $R(k)$ and $S(k)$ in one of these sections (from $k = N_P(J)$ to $N_P(J+1)$). This section is further divided into two subsections from $k = N_P(J)$ to $N_V(J)$ and from $k = N_V(J)$ to $N_P(J+1)$ (marked as (A) and (B) in Figure 5). In subsection (A), the two endpoints of $S(k)$ are

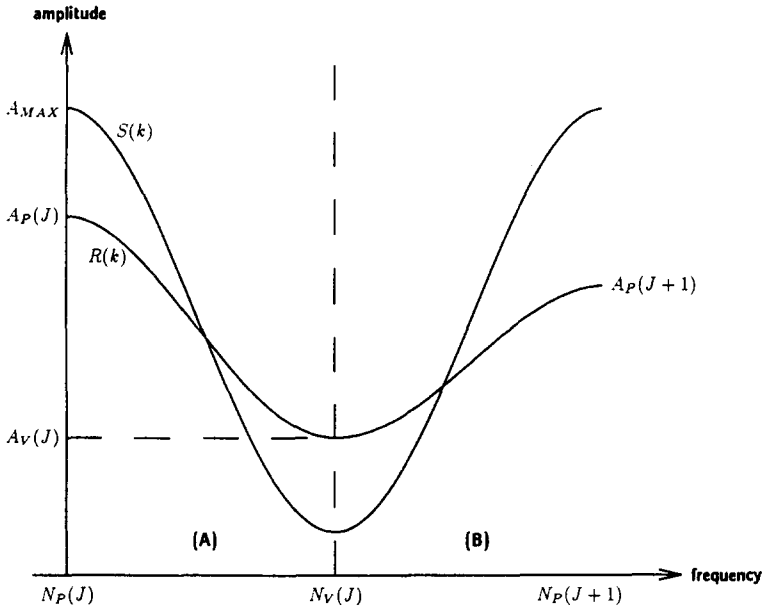


Fig. 5. The envelopes of a section of $R(k)$ and $S(k)$ between two formants.

assigned as follows:

$$S(N_P(J)) = A_{\mathrm{MAX}},$$
$$S(N_V(J)) = R(N_V(J)) + \tau A_{\mathrm{MAX}} = A_V(J) + \tau A_{\mathrm{MAX}}. \quad (6)$$

The frequency response of the postfilter will have peaks of equal amplitude A_{MAX} at the formant frequencies. The factor $\tau < 0$ reflects how much we want to deepen the valleys. In addition, τ depends on the SNR in that for a high SNR, a large value of τ is used. For an SNR of 10 dB, we use $\tau = -0.05$. Let $\Delta(N_P(J))$ be the change in $R(k)$ at $k = N_P(J)$. Then, $\Delta(N_P(J)) = S(N_P(J)) - R(N_P(J)) = A_{\mathrm{MAX}} - A_P(J)$. Similarly, $\Delta(N_V(J))$ is the change in $R(k)$ at $k = N_V(J)$. Then, $\Delta(N_V(J)) = S(N_V(J)) - R(N_V(J)) = \tau A_{\mathrm{MAX}}$. The changes in $R(k)$ at the intermediate points $N_P(J) < k < N_V(J)$ are calculated by linearly interpolating between the values $\Delta(N_P(J))$ and $\Delta(N_V(J))$. Specifically, $\Delta(k)$, which is the change in $R(k)$ for $N_P(J) < k < N_V(J)$, is given by

$$\Delta(k) = \frac{(\Delta(N_V(J)) - \Delta(N_P(J)))(k - N_P(J))}{N_V(J) - N_P(J)} + \Delta(N_P(J)). \quad (7)$$

Thus, $S(k) = R(k) + \Delta(k)$. A similar procedure of fixing the endpoints of $S(k)$ and linearly interpolating the changes $\Delta(k)$ between the two endpoints is adopted for subsection (B) (see Figure 5). In this way, all the sections of $R(k)$ corresponding to actual formants are modified to give $S(k)$.

We must now consider the regions from $0 \leqslant k \leqslant N_P(1)$, $N_P(N_F) \leqslant k \leqslant 109$ and $110 \leqslant k \leqslant 128$. First, consider the case $0 \leqslant k \leqslant N_P(1)$. If there is a local minimum of $R(k)$ at $N_V(0)$, we divide this region into two subsections, namely, (A) from $k = 0$ to $N_V(0)$ and (B) from $k = N_V(0)$ to $N_P(1)$. For subsection (A), $R(k)$ is modified such that the valley at $N_V(0)$ is deepened and such that there is no change at $k = 0$ ($S(0) = R(0) = A_P(0)$). This is because the peak at $k = 0$ is not a true formant. The intermediate points are modified by calculating $\Delta(k)$ as described above. The modification of $R(k)$ for subsection (B) follows the general method described above. Thus, for the first formant at $k = N_P(1)$, $S(N_P(1)) = A_{\mathrm{MAX}}$. If $R(k)$ monotonically increases from $k = 0$ to $N_P(1)$, then this entire section is modified by the general method outlined above such that the valley at $k = 0$ is deepened and the amplitude of the formant peak at $k = N_P(1)$ equals A_{MAX}.

Now, we examine the region $N_P(N_F) \leqslant k \leqslant 109$. If $R(k)$ monotonically decreases in this region, we follow the general method described above to get $S(k)$. Otherwise, two subsections are formed, (A) from $k = N_P(N_F)$ to $N_V(N_F)$ and (B) $k = N_V(N_F)$ to 109. For subsection (A), we follow the general method to get $S(k)$ such that the valley at $N_V(N_F)$ is deepened. Subsection (B) corresponds to relatively high frequencies. The magnitude spectrum at these frequencies should be deemphasized to mitigate the effects of noise at the high frequencies. Therefore, $S(k) = R(k) - 10$ for subsection (B). The range $110 \leqslant k \leqslant 128$ corresponds to the frequencies above 3.4 kHz in which there are no components of the speech signal (only noise is present). The magnitude spectrum must be severely depressed at these frequencies to virtually remove any noise components. Therefore, we set $S(k) = R(k) - 20$ for $110 \leqslant k \leqslant 128$.

The procedure for getting $S(k)$ from $R(k)$ has been given. However, there are two special cases in which we deviate from the general procedure. First, suppose we encounter a frame in which the speech signal is very strong compared to the noise level, with the first formant peak having the largest amplitude ($A_P(1) = A_{MAX}$). Then, some of the valleys may also have a high amplitude as compared to the noise level. In such a case, we do not have to deepen these valleys. If $A_V(0) >$ Formant peak of lowest amplitude, then $S(k) = R(k)$ for $k = 0$ to $N_P(1)$ (no change in the spectrum). Similarly, if $A_V(1) >$ Formant peak of lowest amplitude, then $S(k) = R(k)$ for $k = N_P(1)$ to $N_V(1)$. However, the section from $k = N_V(1)$ to $N_P(2)$ is modified to sharpen the second formant and retain the amplitude at $k = N_V(1)$. In this manner, the amplitudes of each valley are checked to decide whether these valleys have to be deepened at all. This flexibility of being able to preserve certain parts of the signal spectrum where the noise level is low is an advantage of the frequency-domain approach.

A second special case arises if a frame is unvoiced ($L_U = 1$) and $A_P(N_F) = A_{MAX}$. Having the largest peak at a high frequency is quite common for unvoiced frames. In particular, some unvoiced frames only have one broad formant resonance at a relatively high frequency. If the formant detection algorithm results in only one formant, we simply proceed to get $S(k)$ from $R(k)$. If $N_F > 1$, the amplitudes of the other formants must be tested before postfiltering. If the amplitudes of these formants are very low, there is much noise at the low frequencies which would be enhanced by the postfilter. To avoid this, we do the following. Suppose $N_F = 2$ and $A_P(1) < 2N_{AV}$. The first formant peak has a low amplitude and is discarded thereby bringing N_F down to 1. If $N_F > 2$, we test the first two peaks having amplitudes $A_P(1)$ and $A_P(2)$. If $A_P(1) < 2N_{AV}$, the peak at $N_P(1)$ is discarded. Similarly, if $A_P(2) < 2N_{AV}$, the peak at $N_P(2)$ is discarded.

3.1.4. Computation of $H(k)$

Now, the postfilter coefficients $H(k)$ must be determined from the modified log magnitude spectrum $S(k)$. Note that $S(k)$ is a representation of $|H(k)|$ in that $S(k) = 20 \log |H(k)|$. Therefore, $|H(k)| = 10^{S(k)/20}$. The phase of $H(k)$ is exactly the same as the phase of $1/A_P(k)$. Given that the phase of $1/A_P(k)$ is $\theta(k)$, $H(k) = |H(k)| e^{j\theta(k)}$. The postfilter coefficients are obtained by modifying only the magnitude of the LPC spectrum. The phase component remains unaltered.

3.2. Modification of $X(k)$ – smooth switching algorithm

In many speech utterances, there are transitions of very weak speech or silence to or from frames having a relatively stronger speech component. For speech degraded by noise, these transitions contain a substantial noise component. No formants ($N_F = 0$) are usually detected in these transition regions, and postfiltering is not useful for these frames because much noise will be present in the output signal. We must find a way to suppress the noise in these transition regions. The first step is to detect these transition regions (those for which $N_F = 0$). Second, $X(k)$ (the DFT of $x(n)$) must be modified to become $P(k)$ such that the noise in these regions is suppressed. One approach is to set $P(k) = 0$ in these frames thereby making the output signal zero. This introduces a very abrupt switching effect between frames in which there is a strong speech component and those having very weak speech and pure noise. Moreover, this abrupt switching effect is perceivable. We attempt to alleviate this problem by formulating a smooth switching algorithm to modify $X(k)$ in a region of very weak speech or

pure noise ($N_F=0$), that arises between frames having a stronger speech component ($N_F>0$).

Before finding $P(k)$, we must classify the different frames of the signal. The frames for which $N_F=0$ (very weak speech or pure noise) are classified into one of three possible states so that there is no abrupt modification of $X(k)$ to $P(k)=0$. Suppose no formants are found in the current frame m. If in addition $N_F>0$ in frame $m-1$, frame m is said to be in the state (0, 1), which indicates a transition from speech to pure noise. Furthermore, the next 7 frames are also assigned to state (0, 1) as long as no formants are detected in these frames. In any section of the utterance for which $N_F=0$, there is a maximum of 8 frames that are in state (0, 1). If there are less than 8 frames in state (0, 1), we have encountered a frame for which $N_F>0$.

After having 8 frames in state (0, 1), suppose we continue to encounter frames in which $N_F=0$. Then, these frames do not correspond to a transition from speech to pure noise, but are indeed segments of very weak or pure noise. These frames are assigned to state (0, 0). We further subdivide the frames in state (0, 0) to be in state (0, 0, 0) or (0, 0, 1). Frames in state (0, 0, 0) correspond to pure noise in which there is no indication of a transition to frames with a speech component. Frames in state (0, 0, 1) generally correspond to very weak speech and provide an indication of a transition to frames with a strong speech component. This distinction between the states (0, 0, 0) and (0. 0, 1) is necessary to avoid an abrupt switching effect. However, we face the problem of knowing in advance when the frames in which $N_F>0$ will be encountered. Before frames having a speech component appear, there are usually some frames of very weak speech at low frequencies in which no formants are detected. These frames of low frequency weak speech are in state (0, 0), but provide an indication that frames of stronger speech will appear, and can hence be classified as being in state (0, 0, 1). To distinguish between states (0, 0, 0) and (0, 0, 1), consider $R(k)$ for $k=5$ to 20 (which approximately corresponds to the low frequency range 150 to 500 Hz). If $R(k)>1.2N_{AV}$ for any k between 5 and 20, the corresponding frame is assigned to state (0, 0, 1). Otherwise, the frame is assigned state (0, 0, 0).

Finally, consider the frames in which formants are detected ($N_F>0$). These frames are either in state (1, 0) or (1, 1). Suppose $N_F>0$ in frame m and $N_F=0$ in frame $m-1$. Then, frame m is in state (1, 0), a state in which there is a transition from a segment of very weak speech or pure noise to a segment with a stronger speech component. All of the other frames for which $N_F>0$ are in state (1, 1).

The method for obtaining $P(k)$ from $X(k)$ depends on the state of the particular frame. Since a 256-point DFT is taken and all the signals are real, we will give the modification approach for only the first 129 points. First, consider the frames in state (0, 1). Given M consecutive frames (recall that the maximum value of M is 8) numbered $L=1, 2, \ldots, M$ in state (0, 1), we define a modification factor $D=0.8-0.1L$. Then,

$$P(k)=\begin{cases} DX(k), & k=5,\ldots,16, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

It is the low frequency components that are gradually deemphasized. For frames in state (0, 0, 0), $P(k)=0$ for all k, thereby eliminating the effect of postfiltering. For state (0, 0, 1), there is an indication that a strong speech component will appear soon. However, modifying $X(k)$ to allow for a smooth transition to frames with a strong speech component is difficult since there is no a priori knowledge as to when these frames will appear. Since state (0, 0, 1) usually corresponds to weak speech at low frequencies, we introduce the following scheme to preserve some low frequency components:

$$P(k)=\begin{cases} 0.3X(k), & k=5,\ldots,16, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

If a frame is either in state (1, 0) or (1, 1),

$$P(k)=\begin{cases} 0, & k=0,1, \\ 0, & k=110,\ldots,128, \\ X(k), & \text{otherwise.} \end{cases} \tag{10}$$

The noise at the high frequencies beyond 3400 Hz and at the very low frequencies is eliminated, because speech is bandlimited to 3400 Hz and is heavily corrupted by noise at the very low frequencies.

3.3. Generation of the postfiltered output signal

The postfiltering operation is performed only for the frames in which there is a relatively strong speech component (those in states (1, 0) and (1, 1)). In these cases, the DFT coefficients $Y(k)$ of the output signal $y(n)$ are determined as $Y(k) = H(k)P(k)$. Note that for frames in state (1, 0), the signal $y(n)$ is scaled by 0.2. This is done to further alleviate the abruptness in the transition from very weak speech or pure noise to a region of relatively stronger speech. Experiments have shown that this abruptness is perceptible if the scaling is not done. For the other frames in states (0, 0, 0), (0, 0, 1) and (0, 1), no postfiltering is done, i.e., $Y(k) = P(k)$.

Given $Y(k)$, a 256-point inverse DFT yields $y(n)$. For each of the 128-sample frames of the input $x(n)$, we have a 256-point output $y(n)$. The final postfiltered output $v(n)$ is obtained by introducing a 50 percent overlap between the 256-sample segments of $y(n)$ and adding the corresponding samples (an overlap-add strategy). Hence, for each 128-sample frame of the input $x(n)$, we get a 128-sample segment of the postfiltered signal $v(n)$.

4. Experimental results

In this section, we discuss the experimental results obtained after implementing both the time domain and our new frequency domain approaches. The input speech is degraded by additive white Gaussian noise corresponding to an SNR of 10 dB. We chose this level of noise because it renders the speech signal significantly less intelligible than in quiet conditions, while at the same time allowing reasonable quality improvement because perceptually-important spectral information in the noisy speech signal is not submerged in noise. At an SNR of 20 dB, speech is quite intelligible, and most enhancement methods at this noise level may actually degrade the quality of the speech signal, rather than enhancing it; at a level of 0 dB, speech is very noisy and difficult to enhance without losing intelligibility (because most of the important lower frequency formants have amplitude levels lower than that of the noise). Thus we felt that a 10 dB noise was most appropriate for testing.

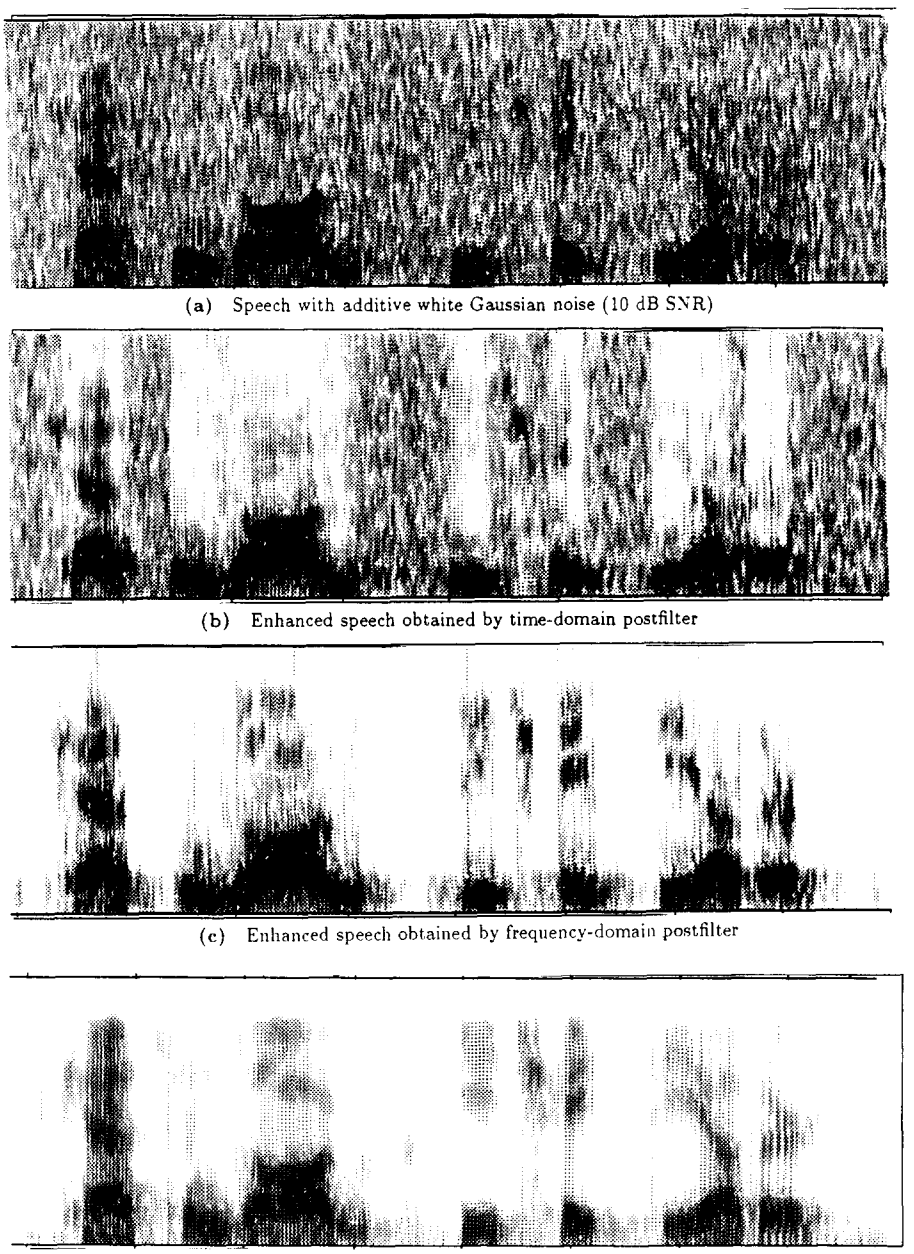
For the time domain approach, we use the postfilter based on an LPC model as given by

$$H(z) = \frac{A_M(z/\beta)}{A_N(z/\alpha)}. \tag{11}$$

For each 128-sample frame of speech, the LPC parameters are determined by an autocorrelation analysis using a Hamming window of 256 samples. The parameters describing the LPC model are given by $M = N = 16$, $\alpha = 1.0$ and $\beta = 0.2$. Note that postfiltering is applied to the entire signal in that there is no switching format between voiced and unvoiced segments. For the frequency domain method, the experimental conditions for an SNR of 10 dB have been given in Section 3.

Figure 6 shows the wideband spectrograms of clean, noisy, and enhanced speech for the sentence "Cats and dogs each hate the other" spoken by a male. Figure 6(a) shows the effect of added noise at 10 dB SNR. The strong first formant is visible throughout for the vowels, but the weaker second formant disappears in the word "each". Higher-frequency formants are only visible for strong vowels with a high first formant (e.g., $F3$ and $F4$ in "cats", "dogs"). Frication (e.g., the "ts" in "cats") is totally obscured by the noise. Figure 6(b) shows the results after enhancement in the time domain; where the speech is strong enough (i.e., during vowels), noise is suppressed in the non-formant regions, primarily at high frequencies. However, no effect is seen during the non-vowel portions of the speech, and the background noise is as strong as ever there. In Figure 6(c), we see the effect of our frequency-domain enhancement; the noise is significantly suppressed during the non-vowel portions of the signal. In addition, some frication is identified properly and retained in the output (e.g., the aspiration of /k/ in "cats", and the final frication in "each"). Furthermore, the higher formants are in general better modeled in the frequency-domain approach than in the time-domain method; e.g., $F3$ and $F4$ in "dogs", $F2$–$F4$ in "hate", and $F2$ in "the other". For comparison purposes, Figure 6(d) shows the original speech.

Figure 7 shows parallel results for a female speaker. The noise obscures almost all formant information above 2 kHz for the female speaker (Figure 7(a)). The comments above for the male



(a) Speech with additive white Gaussian noise (10 dB SNR)

(b) Enhanced speech obtained by time-domain postfilter

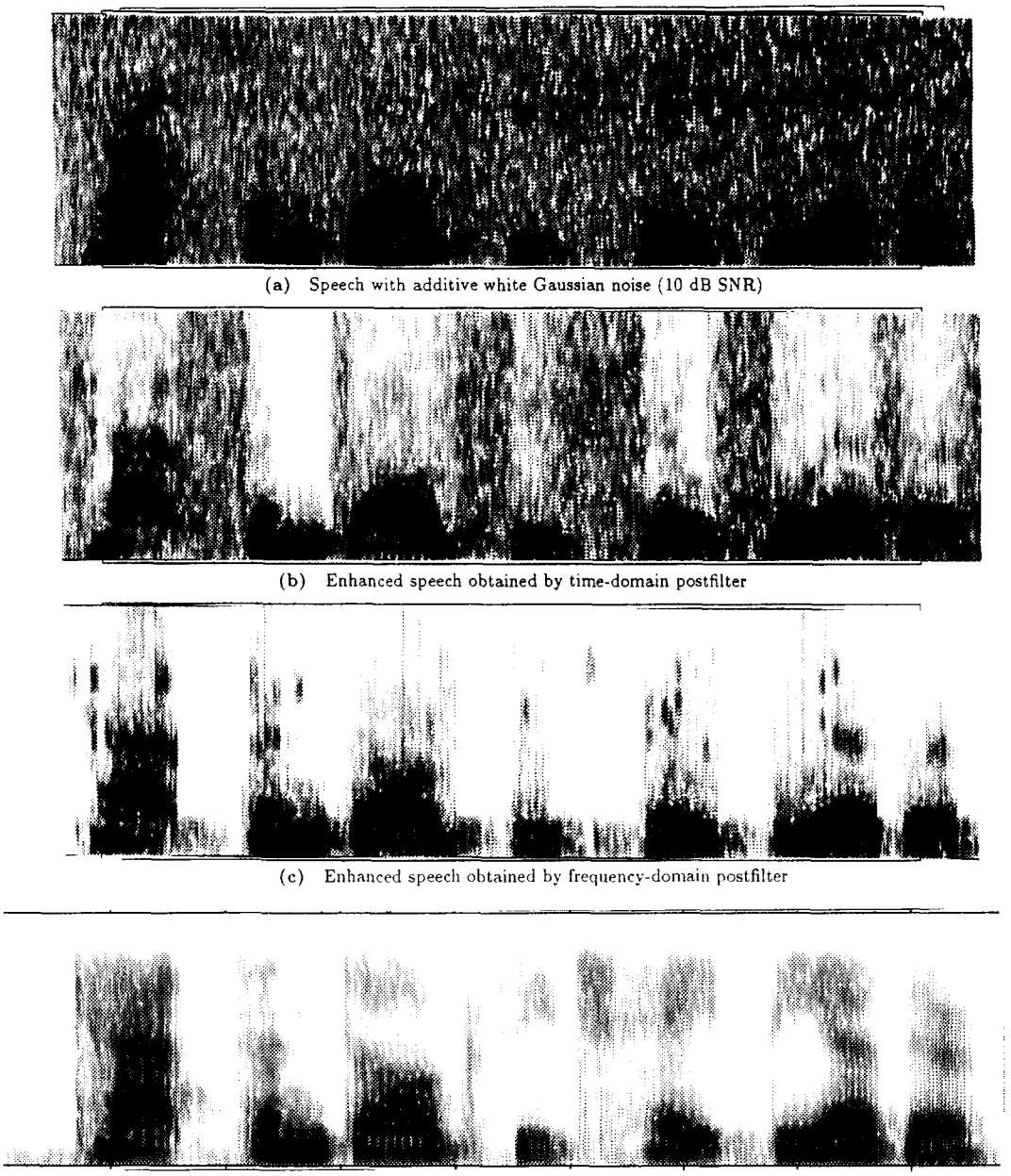
(c) Enhanced speech obtained by frequency-domain postfilter

(d) Original speech

Fig. 6. Wideband spectrograms of noisy, enhanced and clean speech for a male speaker.

speaker apply as well to the results for the female speaker in general.

Informal listening tests clearly indicate a preference for the frequency-domain method over the time-domain method (as described in Section 2.2 and further discussed in this section above). The much decreased noise level with the frequency-domain approach leads to much more pleasant speech, while retaining as much as possible of the phonetic information to keep intelligibility high. Our algorithm filters out energy from the frequency ranges where the noise dominates the speech



(a) Speech with additive white Gaussian noise (10 dB SNR)

(b) Enhanced speech obtained by time-domain postfilter

(c) Enhanced speech obtained by frequency-domain postfilter

(d) Original speech

Fig. 7. Wideband spectrograms of noisy, enhanced and clean speech for a female speaker.

information, but has minimal degrading effect on the remaining frequency ranges where the speech is strong enough. Thus, we are not "over-enhancing" the speech (as may occur in some enhancement methods) to the point where the noise is gone but the speech is heavily distorted or sounds synthetic. To be sure, distortions remain due to the loss of some formant information which could not be recovered from the noisy speech. However, our output speech is a perceptual improvement over

the input speech, and the remaining distortions do not cause the speech to sound synthetic; thus a natural quality remains intact.

5. Summary and conclusions

In this paper, we formulate a new frequency-domain approach for adaptive postfiltering of noisy speech. The technique is specific to the case of white noise. This method allows for independent control over different portions of the speech spectrum, especially those regions corresponding to the formants and spectral valleys. Based on an LPC analysis, we first compute the log magnitude spectrum which serves as a discrete representation of an approximation to the noisy speech spectrum.

From the log magnitude spectrum, a new peak picking strategy is used to detect the formants. Given the formants, the spectral valleys of the speech spectrum are determined. Then, the DFT coefficients of the postfilter are determined for the purposes of suppressing the noise around the spectral valleys, sharpening the formant peaks and introducing no additional lowpass spectral tilt in the enhanced speech. The actual postfiltering is also done by a DFT. In fact, this idea of performing the postfiltering in the frequency domain allows us to suppress the noise which is dominant in the temporal regions of a speech signal corresponding to low energy or silence. Moreover, there is no abrupt perceptual effect in the transition regions between low energy or silence and high energy voiced speech. Experimental results show that the perceptual speech quality is improved with the new method compared to the time-domain postfiltering method.

Acknowledgments

The authors would like to thank Dr. N.S. Jayant for discussing this work and providing constructive comments to improve the paper.

References

J.-H. Chen and A. Gersho, "Real-time vector APC speech coding at 4800 bps with adaptive postfiltering", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Dallas, TX, April 1987, pp. 2185–2188.

N.S. Jayant and V. Ramamoorthy (1986), "Adaptive postfiltering of 16 kb/s ADPCM speech", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Tokyo, Japan, April 1986, pp. 829–832.

J.D. Markel (1972), "Digital inverse filtering – A new tool for formant trajectory estimation", *IEEE Trans. Audio Electroacoust.*, Vol. AU-20, June 1972, pp. 129–137.

S.S. McCandless (1976), "An algorithm for automatic formant extraction using linear prediction spectra", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-22, April 1974, pp. 135–141.

D. O'Shaughnessy (1987), *Speech Communication Human and Machine* (Addison-Wesley, Reading, MA).

D. O'Shaugnessy (1989), "Enhancing speech degraded by additive noise or interfering speakers", *IEEE Comm. Mag.*, February 1989, pp. 46–52.

L.R. Rabiner and R.W. Schafer (1978), Digital Processing of Speech Signals (Prentice Hall, Englewood Cliffs, NJ).

V. Ramamoorthy, N.S. Jayant, R. V. Cox and M.M. Sondhi (1988), "Enhancement of ADPCM speech coding with backward-adaptive algorithms for postfiltering and noise feedback", *IEEE J. Sel. Areas Comm.*, Vol. 6, February 1988, pp. 364–382.