Matching Pursuits Sinusoidal Speech Coding

Çağrı Ö. Etemoğlu and Vladimir Cuperman, *Fellow, IEEE*

***Abstract*—This paper introduces a sinusoidal modeling technique for low bit rate speech coding wherein the parameters for each sinusoidal component are sequentially extracted by a closed-loop analysis. The sinusoidal modeling of the speech linear prediction (LP) residual is performed within the general framework of matching pursuits with a dictionary of sinusoids. The frequency space of sinusoids is restricted to sets of frequency intervals or *bins*, which in conjunction with the closed-loop analysis allow us to map the frequencies of the sinusoids into a frequency vector that is efficiently quantized. In voiced frames, two sets of frequency vectors are generated: one of them represents harmonically related and the other one nonharmonically related components of the voiced segment. This approach eliminates the need for voicing dependent cutoff frequency that is difficult to estimate correctly and to quantize at low bit rates. In transition frames, to efficiently extract and quantize the set of frequencies needed for the sinusoidal representation of the LP residual, we introduce frequency bin vector quantization (FBVQ). FBVQ selects a vector of nonuniformly spaced frequencies from a frequency codebook in order to represent the frequency domain information in transition regions. Our use of FBVQ with closed-loop searching contribute to an improvement of speech quality in transition frames. The effectiveness of the coding scheme is enhanced by exploiting the critical band concept of auditory perception in defining the frequency bins. To demonstrate the viability and the advantages of the new models studied, we designed a 4 kbps matching pursuits sinusoidal speech coder. Subjective results indicate that the proposed coder at 4 kbps has quality exceeding the 6.3 kbps G.723.1 coder.**

***Index Terms*—Matching pursuits, sinusoidal speech coding.**

I. Introduction

In this paper, a novel model for the speech waveform is derived. This model leads to an analysis/synthesis technique [1] whereby the waveform is characterized by amplitudes, frequencies, and phases of the component sine waves. The sinusoidal modeling is performed within the general framework of matching pursuits [2]–[4]. The parameters of each sinusoidal component are extracted sequentially using a closed-loop approach. To efficiently model speech, the phonetic character of individual frames should be considered. Hence, a multimode approach that uses a particular model for each different type (voiced, unvoiced, transition) of speech signal is adopted in this work.

The signal to be represented by the matching pursuits model can be the original speech signal or the linear prediction (LP) [5] residual signal. We believe that selecting the residual signal rather than the speech signal as the target signal for matching pursuits analysis is advantageous in following respects. First, compared to the magnitude spectrum of the original speech signal, the residual spectrum has enhanced harmonic structure especially at high frequencies. The enhancement of the peaks in the spectrum is mainly due to the reduced energy leakage from the strong harmonics into the weak ones, since harmonic peaks in the residual spectrum have less energy variation than the ones in the original speech spectrum. Second, the minimum phase characteristic of the LP synthesis filter adds naturalness to the synthetic phase model which is a polynomial approximation of the real phase. A third factor is the smoothing effect of LP filtering which alleviates the discontinuities due to the frame by frame analysis and synthesis. And finally, LP parameters through the use of line spectral frequency (LSF) representation, model efficiently the spectral envelope of the speech signal, thereby generating a residual spectrum easier to model than the original spectrum.

We will present the analysis/synthesis model in Section II and the matching pursuits analysis with a sinusoidal dictionary in Section III. Section IV introduces the *frequency bin* model which enables us to adapt the dictionary of the matching pursuits to different types of frames, to incorporate perceptual factors into analysis, and to reduce complexity. The remaining sections will describe in detail how matching pursuits analysis is applied to voiced, unvoiced and transition frames.

II. Analysis/Synthesis Model

In the analysis model, each frame of the linear prediction (LP) residual is represented as a sum of sinusoids which are weighted by a magnitude envelope $\sigma^k[n]$ [4]. Thus, for the kth frame, we have

$$\bar{s}^k[n] = \sigma^k[n] \sum_{i=1}^{M(k)} A_i^k \cos\left(\omega_i^k n + \theta_i^k\right). \tag{1}$$

The parameter set for each frame consists of an amplitude vector $\boldsymbol{A} = \{A_i\}$, a frequency vector $\boldsymbol{\omega} = \{\omega_i\}$, and a phase vector $\boldsymbol{\phi} = \{\phi_i\}$.

While the analysis model is the same for different modes, the synthesis has a unique phase model corresponding to each mode in order to exploit the specific type of redundancy particular to that type of frame. The synthesis as a function of the phase model is given by

$$\tilde{s}^k[n] = \sigma^k[n] \sum_{i=1}^{M(k)} A_i^k \cos\left(\theta_i^k(n)\right) \tag{2}$$

where the phase model $\theta_i^k(n)$ is derived from the frequency $\boldsymbol{\omega}$ and the phase $\boldsymbol{\phi}$ vectors extracted during the analysis. The syn-

Manuscript received November 8, 2001; revised April 28, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Peter Vary.

The authors are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106 USA (e-mail: cagrie@zte.com.tr; vladimir@dsp-consult.com).

Digital Object Identifier 10.1109/TSA.2003.815520

thesized frames are combined by using overlap-add to obtain the reconstructed LP residual, $\tilde{s}[n]$

$$\tilde{s}[n] = \sum_k W_s[n - kN_s]\tilde{s}^k[n] \tag{3}$$

where N_s is the synthesis frame size. The synthesis window obeys the constraint

$$\sum_k W_s[n - kN_s] = 1. \tag{4}$$

III. Analysis With Matching Pursuits

To effectively represent the LP residual as a sum of sinusoids, we adopt the general approach of matching pursuits [3], [4]. This is an iterative algorithm, which represents a given signal in terms of a linear combination of a set of M waveforms, selected sequentially from a redundant dictionary whose size is generally much larger than the number of terms needed for an adequate representation. In our case, the dictionary $\mathcal{D}$ is a set of cosine waveforms as described in Section II. The frequencies $\{\omega_j\}$ of the cosine waveforms forming the dictionary are defined by using a fine grid of L points ($L \gg M$) covering the spectral range of interest and given by $\omega_j = j\pi/(L-1)$ for $j = 0, 1, \ldots, L-1$. The frequencies, amplitudes, and phases for each term in the representation are parameters to be determined by the modeling algorithm. The sum of sinusoids is weighted by a magnitude envelope $\sigma[n]$, to track speech energy variations across the frame. Later we describe how this envelope is obtained and efficiently quantized.

In each iteration, a new sinusoidal term is added to the model, then the modeling error waveform (error residual) is formed. The parameters for each sinusoid are optimized to minimize a weighted measure of the error residual energy. Thus, the error residual after m iterations, $r_m[n]$, is given by

$$\begin{aligned} r_m[n] &= r_{m-1}[n] - \sigma[n]A_m \cos(\omega_m n + \phi_m) \\ &= s[n] - \sigma[n]\sum_{i=1}^{m} A_i \cos(\omega_i n + \phi_i) \end{aligned} \tag{5}$$

where for simplicity the frame index k is omitted.

At the mth iteration, the algorithm will search for the frequency point ω_m, which together with its optimal amplitude A_m and optimal phase ϕ_m minimizes the weighted energy E_m of the error residual given by

$$E_m = \sum_{n \in \mathcal{N}} w_a[n] \{r_{m-1}[n] - \sigma[n]A_m \cos(\omega_m n + \phi_m)\}^2 \tag{6}$$

where $\mathcal{N}$ denotes the time span of the current analysis frame. The analysis window $w_a[n]$ serves as a weighting in (6) and enhances the representation of the region in which $\tilde{s}_k[n]$ has the dominant contribution to $\tilde{s}[n]$.

While this algorithm is able to synthesize high quality speech, it has two major drawbacks. First, the computational complexity is very high, since at each iteration it eliminates only one frequency point and searches through essentially the entire grid of finely-spaced frequencies. Second, the resulting set of frequencies, representing the frame, are irregularly spaced and therefore are difficult to quantize at low bit rates.

These two problems motivated us to develop a novel *dynamic dictionary* matching pursuits algorithm based on a *frequency bin model* for structuring and reducing the allowed set of sinusoidal component frequencies in the dictionary. We refer to this set of frequencies as the *frequency space* of the dictionary.

IV. Analysis With Dynamic Dictionary Matching Pursuits Using a Frequency Bin Model

The dynamic dictionary matching pursuits is a modified matching pursuits algorithm, in which the dictionary is updated at each iteration by removing a group of dictionary elements. The complexity of this algorithm is substantially less than that of the conventional matching pursuits algorithm since the size of the dictionary gradually decreases with successive iterations.

The frequency bin structure represents the frequency space of allowed cosine waveforms as a set of nonoverlapping frequency intervals or *bins* where each bin consists of the set of frequency grid points contained in that interval. Since only one frequency within a given bin will be used in the decoder's synthesis procedure, the width of each bin is chosen as large as possible while satisfying the rule that the perceptual difference between the center frequency and any other frequency point in the bin should be insignificant when using the model in (2) for synthesis. With this requirement the widths of the bins must increase with increasing frequency, since the human auditory system's frequency resolution decreases as the frequency increases. This rule further guarantees that any frequency point in a bin can be quantized to that bin's center frequency without sacrificing perceptual information.

The frequency bin model will be combined with the dynamic dictionary matching pursuits as follows. At each iteration, the analysis procedure will choose the best matching frequency point from the frequency space (determined by the current set of bins); then the dictionary is updated by removing the entire bin corresponding to that frequency point. After all the bins are eliminated, the analysis will stop. Therefore, the number of iterations will be equal to the number of bins in the frequency space that forms the initial dictionary. This search process determines a set of sinusoids whose frequencies are still unquantized. For encoding, these frequency points are quantized to the center frequencies of their respective bins.

Specifically, for a given magnitude envelope $\sigma[n]$, at the mth iteration, given the current dictionary $\mathcal{D}_{m-1}$, and the current residual $r_{m-1}[n]$, we search the frequency space for the frequency point ω_m that minimizes (6). Then we update the residual by using (5) and finally update the dictionary, $\mathcal{D}_{m-1} \rightarrow \mathcal{D}_m$, by removing the bin in which ω_m resides from $\mathcal{D}_{m-1}$.

V. Voiced Analysis

Voiced speech, generated by the rhythmic oscillation of the vocal cords as air is forced out of the lungs, can be described as a *quasiperiodic* signal. Quasiperiodicity means that the signal has a periodic structure but the small variations in the glottal excitation, pitch frequency, and vocal tract, cause a change from one pitch period to the next. Since voiced speech is not exactly periodic, or equivalently, since the whole spectrum is not exactly

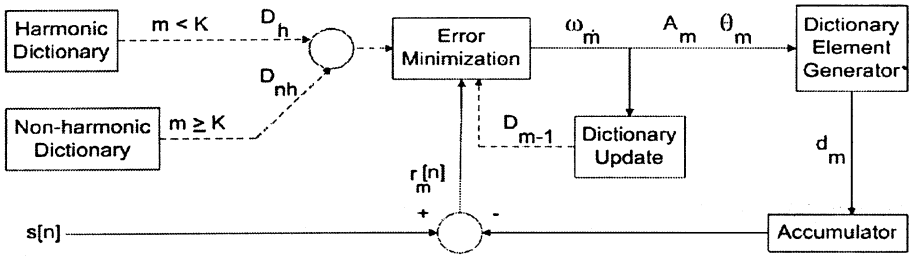


Fig. 1. Block diagram of voiced analysis.

harmonic, voiced segments should be represented with both harmonic and nonharmonic components.

For voiced frames, we use two frequency vectors to capture the frequency domain information. One of them is composed of harmonically related frequencies $\boldsymbol{\omega_h}$ and represents the periodic part; the other one is composed of nonharmonically related frequencies $\boldsymbol{\omega_{nh}}$ and represents the aperiodic part of the voiced segment. The elements of $\boldsymbol{\omega_h}$ are multiples of $\omega_o = 2\pi/p_o$, where p_o is the pitch period. Assuming there is not perceptually significant aperiodic energy in the lower frequency spectrum of voiced frames, the elements of $\boldsymbol{\omega_{nh}}$ are obtained by uniformly sampling the portion of the ERB-rate scale [6] above 1 kHz. Typical voiced frames do not have large energy variations across the frame, therefore the magnitude envelope is set to $\sigma[n] = 1$.

The voiced residual is modeled as a sum of the harmonic and the nonharmonic models, which have $\mathcal{D}_h$ and $\mathcal{D}_{nh}$, respectively, as their dictionaries. Fig. 1 shows a block diagram of voiced analysis. The initial dictionary $\mathcal{D}_0 = \mathcal{D}_h$ corresponds to harmonic analysis. In harmonic analysis, the frequency space of dictionary $\mathcal{D}_h$ consists of bins which are centered at the pitch harmonics and have bin widths given by $\omega_o/2$. Harmonic analysis generates the frequency $\boldsymbol{\omega_p}$, amplitude $\boldsymbol{A_p}$, and phase $\boldsymbol{\phi_p}$ vectors representing the periodic part. The frequency points generated during analysis (i.e., the components of the vector $\boldsymbol{\omega_p}$) do not have to be exact multiples of ω_o, enabling harmonic analysis to capture periodic components even in frames that have varying pitch period. Therefore the leakage from periodic to aperiodic part will be reduced, resulting in an error residual of negligible periodicity, which is suitable as an input for nonharmonic analysis. After K iterations, (K being the number of pitch harmonics, $K = \lfloor p_o/2 \rfloor$) the dictionary is set to $\mathcal{D}_K = \mathcal{D}_{nh}$ for nonharmonic analysis. The frequency space of dictionary $\mathcal{D}_{nh}$ consists of bins which have the elements of $\boldsymbol{\omega_{nh}}$ as their center frequencies. Non-harmonic analysis works on the error residual generated by the harmonic analysis. It generates the frequency $\boldsymbol{\omega_{ap}}$, amplitude $\boldsymbol{A_{ap}}$, and phase $\boldsymbol{\phi_{ap}}$ vectors representing the aperiodic part.

VI. Unvoiced Analysis

Speech segments that do not exhibit a harmonic structure are called unvoiced. The unvoiced speech sounds are generated when vocal cords do not oscillate but air flow is present. Fricatives such as "f" are examples of unvoiced sounds. Whispered speech is completely unvoiced. The magnitude spectrum of an unvoiced segment can be interpreted as the spectrum of a white noise signal, shaped by a spectral envelope. An analysis in

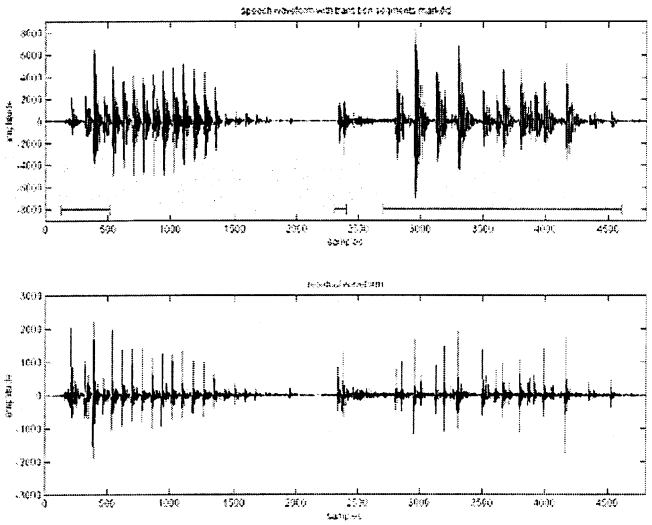


Fig. 2. Speech and residual plots with transition segments marked.

[7] using the Karhunen–Loeve expansion for noise-like signals shows that harmonically related frequencies can represent unvoiced speech provided the frequencies are dense enough. More specifically, perceptually high quality unvoiced speech can be synthesized provided that for a synthesis frame of 10 ms the fundamental frequency used is approximately 100 Hz and a random phase is used.

Unvoiced analysis is the same as harmonic analysis with bins located at multiples of 100 Hz, with the exception that a magnitude envelope is used as in the case of transition frames (Section VII-C).

VII. Transition Analysis

Transition segments of the speech include onsets, plosive sounds, and aperiodic glottal pulses. Fig. 2 shows speech and corresponding residual waveforms of a male speaker for 8 kHz sampling, where the transition segments are marked. The time domain waveform shows that the transition segments are characterized by local time events and are neither periodic nor noise like.

The analysis procedure described in Section IV assumes a given dictionary with an associated set of frequency bins. Transition frames vary a lot in terms frequency content and since the frequency information can not be parameterized by a single parameter (contrary to voiced frames), it is not possible to faithfully represent these frames by using a single dictionary. To overcome the difficulties in representing these frames we developed the frequency bin vector quantization (FBVQ) method

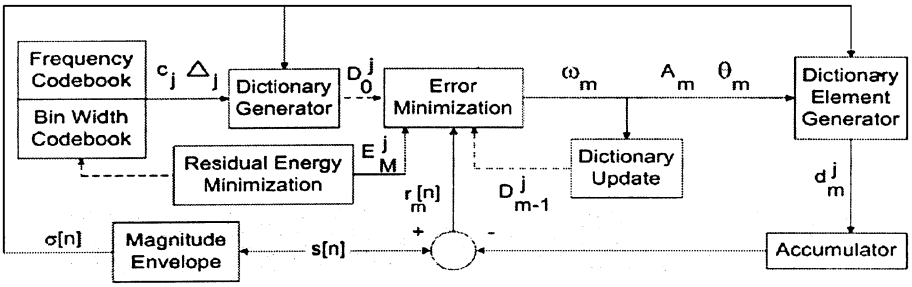


Fig. 3. Block diagram of FBVQ.

which generalizes the analysis to include the search of a family of dictionaries, represented by a vector quantization codebook.

A. Frequency Bin Vector Quantization

To efficiently quantize the set of frequencies needed for the sinusoidal representation of the LP residual in transition frames, we introduce FBVQ. FBVQ encoding is based on a pair of codebooks, a frequency codebook $\mathcal{C}_{\mathcal{F}}$ with elements $\boldsymbol{c_j}$ and a bin width codebook $\mathcal{C}_{\mathcal{B}}$ with elements $\boldsymbol{\Delta_j}$, where the vector $\boldsymbol{c_j}$ is an ordered set of M frequency values and $\boldsymbol{\Delta_j}$ is an ordered set of corresponding bin widths. From the jth pair of codevectors we can generate a dictionary of sinusoids $\mathcal{D}_0^j$, whose frequency space consists of all frequency grid points in the bins that are centered at the elements of $\boldsymbol{c_j}$ and have widths given by the bin widths vector $\boldsymbol{\Delta_j}$.

Fig. 3 shows a block diagram of the analysis procedure based on FBVQ. In the figure, M is the dimension of the frequency codevector. At the mth iteration corresponding to the jth frequency and bin width codevectors, the best matching dictionary element is denoted by d_m^j. Note that the frequency codevector dimension M is equal to the number of iterations, therefore E_M^j denotes the final residual energy corresponding to the dictionary generated by the jth pair of codevectors, $\boldsymbol{c_j}$ and $\boldsymbol{\Delta_j}$. The magnitude envelope $\sigma[n]$ reduces the effect of energy variations on the estimation of the sinusoidal parameters. The analysis procedure in conjunction with FBVQ finds the index of the codebook entry that best matches the signal perceptually, and calculates the corresponding amplitude and phase vectors. FBVQ will use the analysis to select the codebook index, whose corresponding dictionary yields the minimum residual energy E_M^j. In this context, the analysis will act as a method for computing the metric for Nearest Neighbor (NN) condition of FBVQ.

B. Codebook Design Issues

An effective design is needed for the frequency and bin width codebooks. A possible approach for selecting the frequency codevector $\boldsymbol{c_j}$ is to uniformly sample the conventional frequency scale, but this would not account for the nonuniform frequency resolution of the human auditory system. A better approach is to sample the equivalent rectangular bandwidth (ERB) rate scale [6] uniformly, since ERB-rate scale, like human auditory system, has a decreasing frequency resolution with increasing frequency.

The second design issue is the choice of the bin widths vector $\boldsymbol{\Delta_j}$. At low bit rates, we would like to model the input signal by using as few cosine waveforms as possible, since the number of parameters to encode is proportional to the number of cosine waveforms. Experimental evidence shows that we can synthesize good quality transition frames by 20–30 cosines.

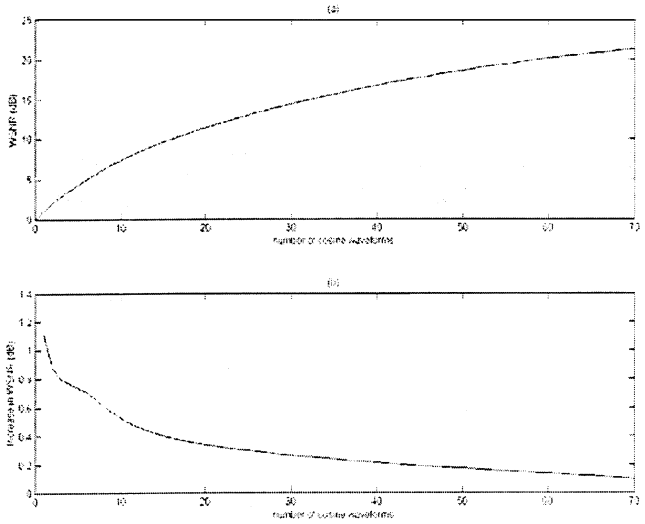


Fig. 4. WSNR as a function of the number of cosine waveforms used in modeling transition frames.

In order to develop objective measures for designing the codebook, we have analyzed more than 6000 transition frames using matching pursuits with a dictionary of sinusoids whose frequency space is unrestricted. A Hamming window of 100 samples was used as the analysis window $w_a[n]$ since its spectrum, $W_a(e^{j\omega})$, has a narrow mainlobe and sidelobes with relatively small magnitude. Based on this investigation, Fig. 4(a) shows the average weighted signal to noise ratio (WSNR) that can be achieved as a function of the number of cosine waveforms in transition frames. The amount of average increase in WSNR attained by adding a new cosine to the current approximation is shown in Fig. 4(b). According to ERB-rate scale we have to use nearly 30 cosine waveforms, for perceptually adequate coverage of 0–4 kHz. Informal listening tests show that 20–30 cosine waveforms are needed for faithful modeling of transition frames. We decided to use 32 cosine waveforms to model transition frames based on these observations. It can be seen from Fig. 4(a) that we can achieve 15 dB WSNR by using 32 cosine waveforms and the benefit of adding one more cosine to the model is less than 0.25 dB as displayed in Fig. 4(b).

Once the number of cosines is fixed, the number of bins will be the same, since the analysis generates a single cosine corresponding to each bin. To match the frequency resolution of the

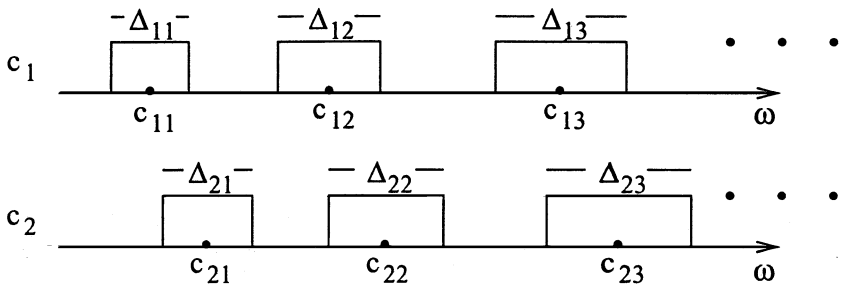


Fig. 5. Sample codevectors and their bin structure.

human auditory system, we derived a codebook of frequency codevectors $\{\boldsymbol{c_j}\}$ of dimension 32, by uniformly sampling the ERB-rate scale. Given the number of bins for a given type of frame, the size of the dictionary will be proportional to the bin widths. In this case $\boldsymbol{\Delta_j}$ determines the tradeoff between the modeling error caused by the reduction of the frequency space into bins and the quantization error caused by mapping the frequency point selected from each bin to that bin's center frequency. If we choose $\{\Delta_{jk}\}$'s too small, the quantization error will be negligible which means the reconstructed signal in the analyzer will be very similar to the one in the synthesizer, but the resulting small dictionary may not accurately model the input waveform. On the other hand, choosing $\{\Delta_{jk}\}$'s too large will lead to a large dictionary which can model the input waveform well, so the reconstructed signal in the analyzer will be of high quality, but the reconstructed signal in the synthesizer will suffer from large quantization errors. A good tradeoff can be obtained by increasing Δ_{jk} up to the point where the quantization error is still perceptually insignificant. In general Δ_{jk} will increase with frequency, since the human auditory system is more tolerable to quantization errors at higher frequencies. Based on our informal listening tests, we decided to increase the bin widths from 50 Hz to 120 Hz with increasing frequency.

C. Transition Model

In transition frames, FBVQ incorporating matching pursuits analysis extracts a frequency vector $\boldsymbol{\omega_t}$, and its corresponding amplitude $\boldsymbol{A_t}$ and phase $\boldsymbol{\phi_t}$ vectors. Then the frequency vector $\boldsymbol{\omega_t}$ is quantized as $\hat{\boldsymbol{\omega}}_{\boldsymbol{t}}$ which is a member of the frequency codebook $\mathcal{C}_{\mathcal{F}}$. Informal listening tests show that a codebook with eight codevectors is sufficient for high perceptual quality transition frames. Each codevector is obtained by uniformly sampling the ERB-rate scale in order to account for the nonuniform frequency resolution of the human auditory system as discussed in Section VII-B. Two codevectors and their corresponding frequency bins structure are illustrated in Fig. 5. In transition frames, a magnitude envelope $\sigma[n]$ is used to track speech energy variations in the frame. The envelope proved to be useful especially in transition frames corresponding to an onset or offset event where there is substantial energy variation in the frame. Since these frames are not stationary in nature, it is difficult to model them using only sine waves.

The magnitude envelope $\sigma[n]$ is formed by linearly interpolating the magnitude vector, $\boldsymbol{\sigma} = \{\sigma_i\}$

$$\sigma_i = \sqrt{\sum_j a_j s^2[l_o + iD - j]} \quad i = 0, 1, \ldots \tag{7}$$

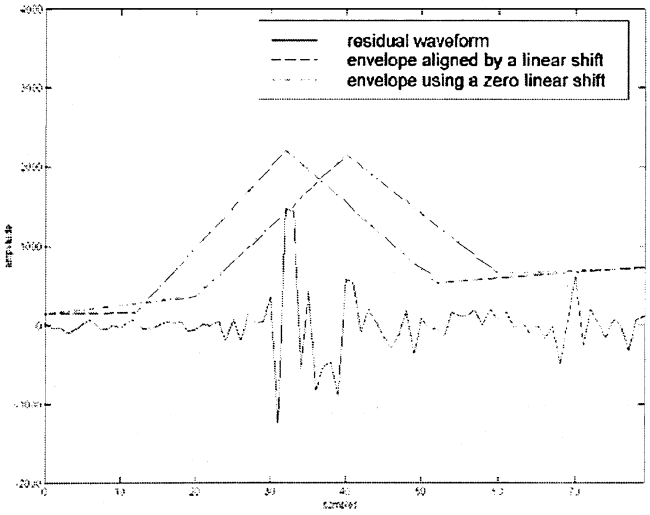


Fig. 6. Magnitude envelope alignment for accurate representation of energy variation in transition frames.

where D is the downsampling factor, and l_o is the time point corresponding to the maximum energy concentration in the transition frame. The energy concentration is measured at each sample time as the local energy around that sample. Before the energy calculation, the speech signal is upsampled 10 times for better resolution. In this upsampled domain, the local energy of each sample is computed as the average energy of the 30 points around that sample. The time point l_o is employed in (7) as a linear shift to align the downsampling operation with the largest energy concentration. This alignment alleviates the smearing of magnitude envelope caused by downsampling and linear interpolation.

Fig. 6 displays a residual waveform and two corresponding magnitude envelopes. The first magnitude envelope, drawn by a "-." (dash dot) line, is generated with $l_o = 0$ value. Observing this figure we can see that this envelope is misaligned and does not represent the energy variation of the residual waveform accurately. The second magnitude envelope, drawn by a "--" (dashed) line, uses l_o value (computed as described above) to align the downsampling instants with the energy concentration of the residual waveform. Clearly, this second envelope is aligned with the energy concentration of the waveform and has less smearing of the envelope due to downsampling and interpolation.

The coefficients $\{a_j\}$ in (7) correspond to a low-pass filter which is used to avoid aliasing in downsampling. The low-pass filter has a cut-off frequency of 200 Hz for the sampling period of 2.5 ms (i.e., $D = 20$ in samples). The quantized magni-

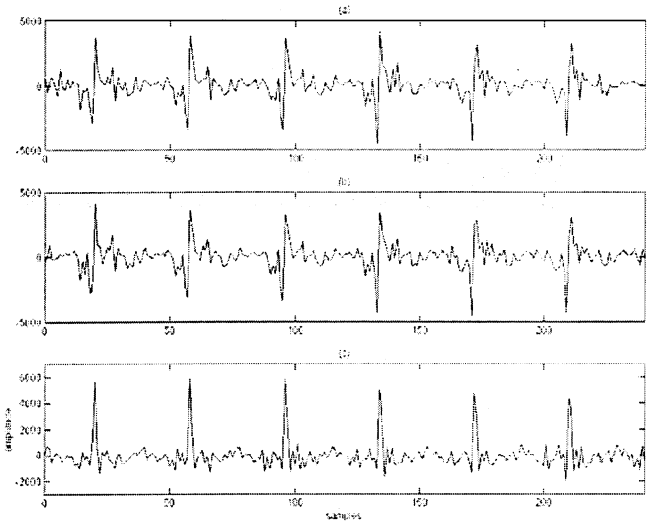


Fig. 7. Role of phase in harmonic synthesis. (a) Original residual, (b) residual reconstructed from 19 harmonics using magnitude and phase, and (c) residual reconstructed from 19 harmonics using magnitude only.

tude envelope, $\hat{\sigma}[n]$ is formed by linearly interpolating the quantized magnitude vector $\hat{\boldsymbol{\sigma}}$, whose elements have time locations aligned with l_0.

VIII. Voiced Synthesis

In voiced frames, two frequency vectors are employed to represent the frequency domain information as discussed in Section V. One of them represents the periodic part and the other one represents the aperiodic part of the voiced segment. Since these two parts are perceived differently and have different perceptual redundancies, each part has its particular synthesis model.

A. Harmonic Synthesis

The periodic part of a voiced segment is reconstructed using harmonic synthesis. The frequency domain information of the periodic part is represented by the vector $\boldsymbol{\omega_h}$ which is composed of harmonically related frequencies (i.e., the elements of $\boldsymbol{\omega_h}$ are multiples of $\omega_o = 2\pi/p_o$, where p_o is the pitch period). The amplitude and the phase vectors for the periodic part are denoted by $\boldsymbol{A_p}$ and $\boldsymbol{\phi_p}$, respectively. Hence, the jth harmonic has amplitude A_{p_j} and phase ϕ_{p_j}, and is given by

$$h_j[n] = A_{p_j}\cos(j\omega_o n + \phi_{p_j}) \\ A_{p_j}\cos(j\omega_o n - j\omega_o n_o + \psi_{p_j}). \quad (8)$$

The phase term $\phi_{p_j} = -j\omega_o n_o + \psi_{p_j}$ controls the relative time shift of the reconstructed signal through n_o which is the point in time where the local energy of pitch epoch is maximized. Moreover, the phase term defines the local pulse shape through ψ_{p_j} which is called the dispersion phase. Fig. 7 shows a voiced residual waveform reconstructed with and without ($\psi_{p_j} = 0$) dispersion phase. The reconstructed waveform with zero dispersion phase [Fig. 7(c)] has symmetrical pulse shape around the linear shift, n_o, as expected. The role of the phase term in governing both the location and the shape of the pitch pulse is evident from the waveform differences in Fig. 7(b) and Fig. 7(c).

While the linear shift parameter n_o is common to all sinusoids, the dispersion phase ψ_{p_j} has a distinct value for each sinusoid (i.e., for each j). Hence, the quantization of the dispersion phases poses a problem at low bit rates. Based on our informal listening tests, human auditory system can tolerate an inaccurate or absent dispersion phase. Based on these observations we decided to discard the dispersion phase in harmonic synthesis, and use zero dispersion phase for the reconstruction of the LP residual.

The function of the linear shift n_o, which is also called the onset time, is to bring the sinusoids into phase at times corresponding to the occurrence of a pitch pulse. Rather than attempting to estimate and use the onset time from the speech, it is possible to achieve the same perceptual effect simply by keeping track of successive onset times generated by the succession of pitch periods that are available at the decoder [7]. Assuming the pitch period is smoothly varying over the synthesis frame, and if n_o^{k-1} is the onset time for frame $k-1$ and p_o^{k-1} and p_o^k are the pitch period estimates for frames $k-1$ and k, respectively, then a succession of synthetic onset times can be generated by

$$n_o^{k-1}(i) = n_o^{k-1} + i\left(\frac{p_o^{k-1} + p_o^k}{2}\right) \quad i = 1, 2, \cdots, I. \quad (9)$$

The onset time for frame k, n_o^k, is chosen to be equal to $n_o^{k-1}(I)$ where $n_o^{k-1}(I)$ is the onset time closest to the center of frame k. An example of a typical sequence of onset times is shown in Fig. 8. Note that, given the linear shift in the first voiced frame (initial linear shift), (9) and the rule $n_o^k = n_o^{k-1}(I)$ can be applied iteratively, to obtain the linear shift values for a voiced segment. In our model, there is always a transition frame before the first voiced frame and the transition frame is used to obtain the initial linear shift. The exact method to estimate the initial linear shift is described in Section XI-A.

The decoder uses the quantized pitch periods in (9) to generate successive onset times. Therefore, no extra bits are required for the transmission of the linear phase. Since dispersion phase was discarded, linear phase constitutes the whole phase information in harmonic synthesis. The smooth evolution of this phase is achieved by using a cubic phase model [8]. Finally, the cosine waveforms with the cubic phase model are combined using a triangular overlap-and-add window to synthesize the periodic part of the voiced segment.

The speech waveform synthesized with the cubic phase model differs considerably from the original speech waveform. First, the synthesized speech and the original speech are not synchronized in time, since the exact onset time information has been replaced by the model derived from (9). Second, the shape of the pitch pulse (of the LP residual) is not preserved, since the individual phase dispersion terms have been discarded. We should note that even though the residual has zero dispersion phase, the reconstructed speech will have nonzero dispersion phase since the LP synthesis filter will supply the dispersion phase contribution of the vocal tract.

B. Nonharmonic Synthesis

The aperiodic part of a voiced segment basically has a noise like characteristic and represents the unvoiced components in a

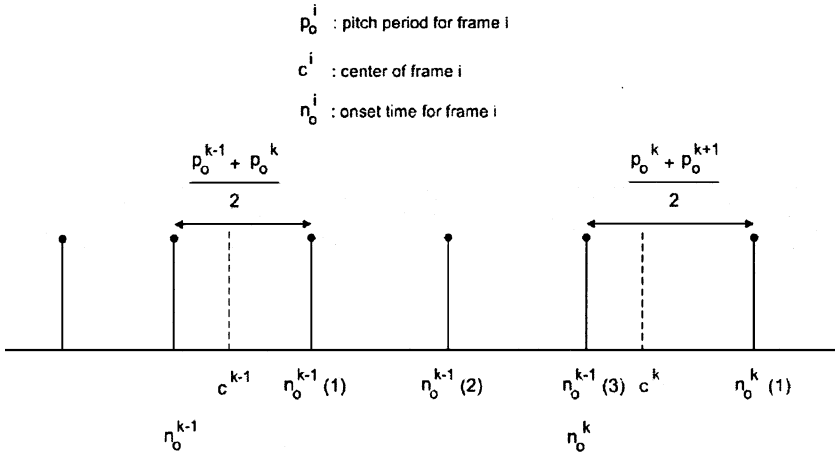


Fig. 8. Sequence of onset times.

voiced frame. Informal listening tests show that the phase components of the aperiodic part (i.e., the components of the vector $\boldsymbol{\phi_{ap}}$) can be replaced by random phase without introducing perceptual degradation. The perceptual insignificance of the phase information is mainly due to noisy characteristic of the aperiodic part. The nonharmonic synthesis represents the LP residual as

$$s_{nh}[n] = \sum_j A_{ap_j} \cos(\omega_{nh_j} n + \phi_{r_j}) \tag{10}$$

where ϕ_{r_j} is a random phase uniformly distributed in the interval $[-\pi, \pi)$.

IX. Unvoiced Synthesis

In unvoiced frames, the frequency domain information is captured by the spectral magnitudes at a set of densely spaced frequencies. The frequency components are separated by $f_u = 100$ Hz. The amplitude vector representing the unvoiced frame is denoted by $\boldsymbol{A_u}$.

The synthesis of the LP residual for unvoiced speech employs a magnitude envelope $\sigma[n]$ and is given by

$$s_u[n] = \sigma[n] \sum_j A_{u_j} \cos\left(2\pi j \frac{f_u}{F_s} n + \phi_{r_j}\right) \tag{11}$$

where $F_s = 8000$ Hz is the sampling frequency, and ϕ_{r_j} is a random phase uniformly distributed in the interval $[-\pi, \pi)$. One more time, the original phase values are replaced by random values in order to exploit the perceptual redundancy in the phase information in unvoiced components. Note that applying a random phase to the sinusoidal components can maintain a perception of randomness in the synthesized signal provided that long term periodicities inherent in sinusoidal components do not manifest themselves over the course of the synthesis frame. Therefore, in order to avoid introducing tonal character (periodicity) into the synthesized signal, synthesis frame length can not be too large. In our listening tests randomization of phases every 10 ms (i.e., a synthesis frame length of 10 ms) has proven to achieve desired randomness in unvoiced frames.

X. Transition Synthesis

Transition frames are synthesized by a frequency vector $\hat{\boldsymbol{\omega}}_t$ selected from a frequency codebook $\mathcal{C}_\mathcal{F}$ by using FBVQ, described in Section VII-A. The search procedure in FBVQ computes the amplitude $\boldsymbol{A_t}$ and phase $\boldsymbol{\phi_t}$ vectors corresponding to the frequency vector $\boldsymbol{\omega_t}$. The model for transition frames incorporates a magnitude envelope $\sigma[n]$, whereby the LP excitation signal is represented as

$$s_t[n] = \sigma[n] \sum_j A_{t_j} \cos(\omega_{t_j} n + \phi_{t_j}). \tag{12}$$

Furthermore, in transition frames the phase vector $\boldsymbol{\phi_t}$ is decomposed into a linear and a dispersion phase component as follows:

$$\phi_t = -\omega_t l_o + \psi_t \tag{13}$$

where $\boldsymbol{\psi_t}$ is the dispersion phase vector representing the structure of the transition segment. The parameter l_o, described in Section VII-C, gives the time location of the maximum energy concentration.

The model defined in (13) decomposes the spectral phase into two conceptually uncorrelated parameters: the time point where the energy concentration is maximized and the shape of the transition segment around this point. This decomposition enables quantization of the two phase components separately using different resolutions. The perceptually important dispersion phase $\boldsymbol{\psi_t}$ is quantized using a time domain distortion measure as in [9]. The maximum energy location l_o is also used as a linear shift in (7) for generating an effective, yet easy to quantize envelope as described in Section VII-C. Furthermore, l_o is employed to achieve a smooth switching between models of different types of frames as described in Section XI.

As we can observe from (12), the transition frames synthesized with this model are synchronized in time with the original speech. On the other hand, the original speech, and the synthesized voiced frames are not synchronized. During a switch from a transition frame to a voiced frame and vice-versa, since the successive frames are not synchronized audible distortion can

be perceived. This problem and the corresponding solutions are discussed in the next section.

XI. Model Switching

There are two cases of model switching in which synthesized speech quality can degrade significantly unless appropriate algorithm modifications are made. The first case is the switch from a transition frame to a voiced frame which generally takes place during the onset of a voiced segment. The second case, which typically happens during the offset of a voiced frame, is the switch from a voiced frame to a transition frame. The next sections describe each of these model switching problems and investigate possible solutions.

A. Switching From a Transition Segment to a Voiced Segment

The harmonic synthesis reconstructs the periodic portions of the voiced speech by employing a cubic phase model that provides phase continuity and signal smoothing between frames. The phase contour described by the cubic phase model is implemented frame by frame, where the final phase values of the previous frame serve as the initial phase values for the current frame. As described in Section VIII-A, since the dispersion phases are discarded, all the phase information corresponding to a voiced frame can be characterized by the onset time, n_o (in samples). However, the initial onset time must be chosen for the first frame at the beginning of a voiced segment. This initial onset time can be used to form the initial linear phase. The initial linear phase controls the relative time shift of the pitch pulses that characterize voiced speech.

During the onset of a voiced speech segment, the speech signal usually consists of a set of nonuniformly spaced pulses of varying structure which are usually represented by a transition model. The transition model is followed by a voiced model once the signal becomes voiced. While the auditory system is not sensitive to the absolute location of the pitch pulses, even a slight deviation in the relative location of the pitch pulses can be perceived as a strong artifact. Since the transition model synthesizes speech synchronized in time with the original speech while the voiced model synthesizes speech by employing a synthetic phase contour, the relative location of the pitch pulses is not preserved during a switch from transition frames to voiced frames. Fig. 9 presents an example that demonstrates the misalignment and phase synchronization problem during an onset. The speech signal corresponding to an onset segment is shown in Fig. 9(a). The classification of the segments are indicated by the markings "transition" and "voiced." Fig. 9(b) displays the corresponding synthesized speech in which the initial onset time of the voiced segment was set to $n_o = 0$. Clearly the synthesized speech in Fig. 9(b) is misaligned with the original speech and the beginning of the voiced segment is not synchronized with the end of the transition segment. More careful inspection of Fig. 9(b) shows that the relative pitch period is not preserved around time instant corresponding to 200th sample.

To overcome the phase synchronization problem, we use a synchronization method similar to [10] to estimate the proper initial onset time at the speech decoder. Assuming frames $k-1$ and k are transition and voiced frames, respectively, and l_o^{k-1} is the shift (described in Section VII-C) for frame $k-1$ and p_o^k is the pitch period estimate for frame k, then a set of synthetic onset times for frame k can be generated by

$$n_o^{k-1}(j) = l_o^{k-1} + jp_o^k \quad j = 1, 2, \cdots, J. \tag{14}$$

The initial onset time for frame k, n_o^k, is chosen to be equal to $n_o^{k-1}(J)$ where $n_o^{k-1}(J)$ is the onset time closest to the center of frame k. The (14) is similar to (9) used to generate successive onset times during a voiced segment. Fig. 9(c) shows the same synthesized speech, but this time the initial onset time is estimated using (14). As seen in Fig. 9(c), the transition and the voiced segments are aligned in synthesis, and the relative locations of the pitch pulses are preserved. No extra bits are transmitted for the onset synchronization, since no additional information is required at the decoder to perform the computation described by (14).

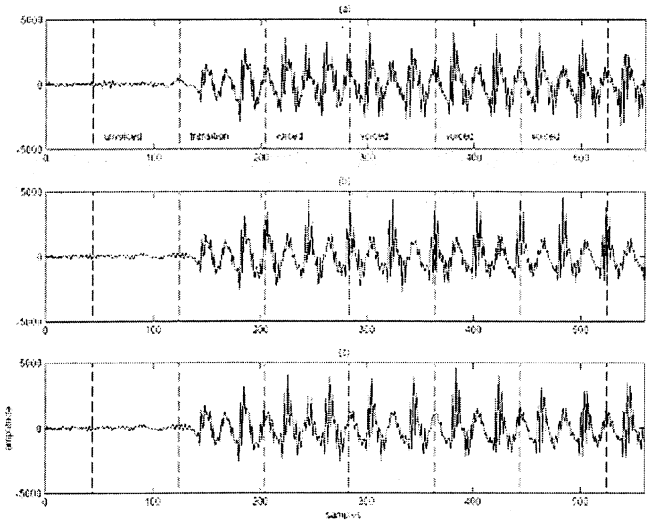


Fig. 9. Onset synchronization. (a) Original residual, (b) residual synthesized by a nonsynchronized ($n_o = 0$) voiced model, and (c) residual synthesized by a synchronized voiced model using an estimate of n_o.

B. Switching From a Voiced Segment to a Transition Segment

Switching from a voiced to a transition segment may occur during vowel offsets. In such cases a continuity problem similar to the continuity problem at the onset occurs, since the synchronization between the original and the synthesized speech during the voiced segment is not guaranteed. On the other hand, transition segments are always synchronized with the original, resulting in a possible misalignment and discontinuity during switching.

Fig. 10 presents a speech sample that illustrates the offset synchronization problem. The speech signal corresponding to an offset segment is shown in Fig. 10(a) where a voiced segment is followed by a transition segment. The classification of the segments are indicated by the markings "voiced" and "transition." The corresponding synthesized speech is displayed in Fig. 10(b). The misalignment is evident around the 200th sample where the relative locations of the pulses were not preserved.

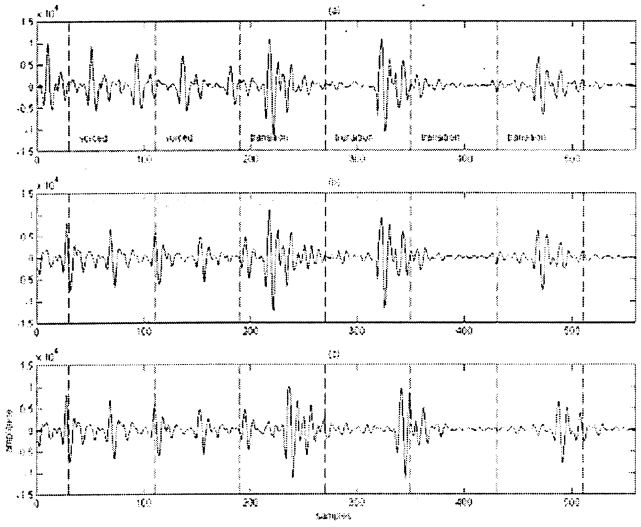


Fig. 10. Offset synchronization; (a) original residual, (b) residual synthesized by a nonsynchronized transition model, and (c) residual synthesized by a synchronized transition model using the drift information.

A synchronization module was employed to solve the phase synchronization problem when switching from a voiced segment to a transition segment. The synchronization is achieved by estimating the drift between the synthesized voiced excitation and the original residual. This drift is then applied to the analysis of the following transition segment, such that the encoder extracts a shifted transition segment for analysis. Since the decoder synthesizes the transition frames without the knowledge of the shift, it generates a transition waveform that is aligned with the previously synthesized voiced segment. Fig. 10(c) shows the same synthesized speech, but this time the alignment is performed by using the drift information in the encoder. As seen in Fig. 10(c), by using the drift information, the encoder shifts the transition segment in such a way that the relative locations of the pulses are preserved.

To estimate the drift during the voiced synthesis, two parameters namely, n_o and m_o are used. The parameter n_o is the linear shift corresponding to the last voiced frame and it is described in Section VIII-A. The point m_o corresponds to the time location where the energy concentration of the original residual is maximized and it is computed the same way as l_o in Section VII-C. If the difference between n_o and m_o is zero or a multiple of the pitch period, there is no drift between the original and the synthesized voiced excitation. Therefore the drift can be simply calculated as

$$d = (n_o - m_o) \bmod p_o \tag{15}$$

where d corresponds to the drift and p_o denotes the pitch period of the last voiced frame. Note that since the drift information is computed and applied in the encoder, no additional information is sent to the decoder.

XII. Matching Pursuits Sinusoidal Speech Coder

To demonstrate the viability and the advantages of the new models in low rate speech coding, this section describes a 4 kbps matching pursuits sinusoidal speech coder [11]. The general structure of the encoder and decoder corresponding to the matching pursuits sinusoidal speech coder is illustrated in Fig. 11. The coder operates on a frame size of 20 ms which corresponds to 160 speech samples for the sampling rate of 8 kHz. Each frame is divided into two 10 ms subframes. In addition to 20 ms frame size, the encoder requires an additional 20 ms of lookahead, resulting in a total algorithmic delay of 40 ms.

The input speech signal is high-pass filtered in the pre-processing block. The pre-processed signal is used in all subsequent coder operations. A tenth-order linear prediction (LP) analysis is performed once per 20 ms frame with a 25 ms asymmetric window to extract the LP coefficients. The LP analysis window consists of a half Hamming window and a quarter of a cosine function similar to the approach presented in [12] for the G.729 speech coding algorithm. The LP coefficients are transformed to line spectral frequencies (LSF) and quantized using predictive multistage vector quantization (MSVQ). The LSFs are interpolated each 5 ms and converted back to LP coefficients which are used by the LP analysis and synthesis filters.

The LP windowing procedure is illustrated in Fig. 12. A Hamming window centered at the subframe boundary is used as the matching pursuits analysis window. The matching pursuits analysis window is 180 samples for voiced frames, and 100 samples for unvoiced and transition frames. Since unvoiced and transition frames are nonstationary in nature a shorter analysis window achieves more accurate parameter estimation in these frames. The LP residual serves as the target signal for matching pursuits analysis and is obtained by filtering the pre-processed speech signal through the LP analysis filter. A dynamic programming based estimation technique [11], [13] is used to estimate the phonetic class and pitch period parameters each subframe. According to the classification information, the encoder is set to one of the three possible modes, namely voiced, unvoiced and transition. In voiced mode, dynamic dictionary matching pursuits analysis using both a harmonic and nonharmonic dictionary (Section V) is performed to obtain the sinusoidal parameters. In unvoiced mode, the matching pursuits analysis using frequency bins located at multiples of 100 Hz (Section VI) is used to extract model parameters. Finally, the FBVQ technique (Section VII-A) is employed in transition mode for the extraction of the parameters corresponding to sinusoidal components.

At the decoder (see Fig. 11), the model parameters corresponding to a 20 ms frame of speech are extracted from the received bit stream. The parameters include LSF coefficients, class, pitch, and excitation model parameters of the specific class. Based on the class information, the decoder is set to a particular decoding mode. Once the parameters are decoded, the LP excitation signal is synthesized using an excitation model defined for that mode. The LSF coefficients are interpolated each 5 ms and converted to LP synthesis filter coefficients. The speech is reconstructed by filtering the synthesized excitation signal through the LP synthesis filter. The reconstructed speech is passed through a post-processing block which is composed of short-term postfilter, tilt compensation filter and an adaptive gain control (AGC) unit for matching the energy of the post-processing output to the energy of the post-processing input.

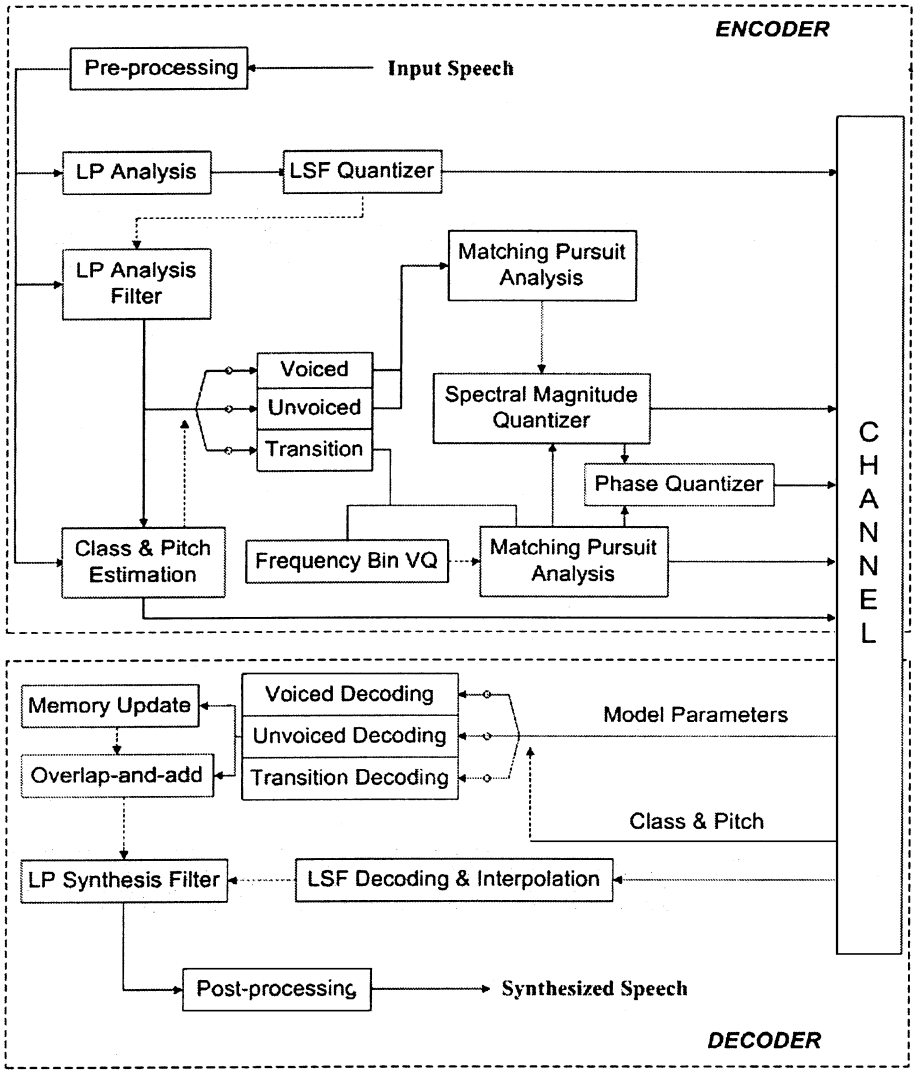


Fig. 11. Block diagram of the encoder and decoder.

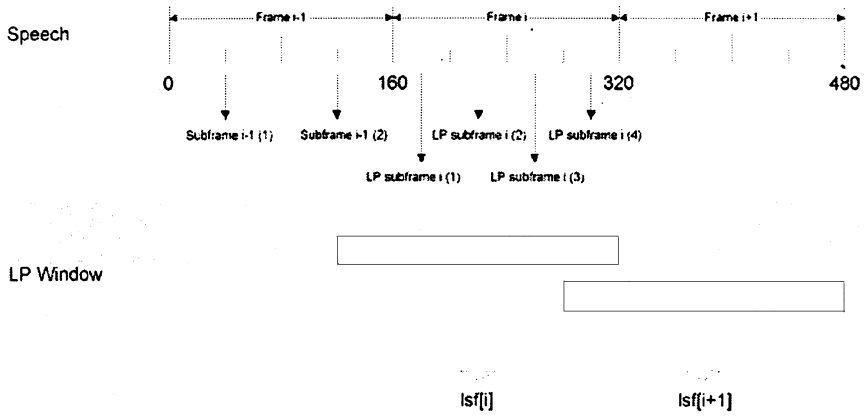


Fig. 12. Framing and LP windowing structure.

XIII. Summary of Bit Allocation

As mentioned, since there are three possible classes for each subframe (voiced, unvoiced, transition), we have a total of nine possible class combinations for each frame. In our coder, the combination corresponding to unvoiced first subframe and voiced second subframe is not allowed (see Section VIII-A). Therefore, we have eight possible class combinations for each 20 ms frame.

In Tables I and II, each class combination is denoted by combining the first letters of the corresponding classes. For example, "VT" denotes a frame in which the first subframe is voiced and

TABLE I
BIT ALLOCATION FOR VV, TT, AND UU FRAMES

Parameter	VV	TT	UU
LSF	24	18	24
Magnitude	40	27	34
Frequency	0	$2 \times 3 = 6$	0
Pitch	7+6=13	0	0
Linear Shift	0	4	0
Dispersion Phase	0	$2 \times 6 = 12$	0
Envelope	0	$2 \times 5 = 10$	$2 \times 9 = 18$
Classifier	3	3	3
Total	80	80	79
Bit-rate	4 kbps	4 kbps	3.95 kbps

TABLE II
BIT ALLOCATION FOR VT, TV, UT, TU, AND VU FRAMES

Parameter	VT and TV	UT and TU	VU
LSF	18	18	24
Magnitude	34	27	34
Frequency	3	3	0
Pitch	7	0	7
Linear Shift	4	4	0
Dispersion Phase	6	6	0
Envelope	5	5+9=14	9
Classifier	3	3	3
Total	80	75	77
Bit-rate	4 kbps	3.75 kbps	3.85 kbps

the second subframe is transition; and "TU" denotes a frame in which the first subframe is transition and the second subframe is unvoiced. Table I summarizes the bit allocation for VV, TT, and UU frames. The bit allocation for VT, TV, UT, TU, and VU frames are given in Table II.

XIV. SUBJECTIVE QUALITY TESTS

Two subjective quality tests were conducted to evaluate the quality of the 4 kbps matching pursuits sinusoidal speech coder. The first test is an A/B test, in which the proposed coder at 4 kbps is compared to G.723.1 at 6.3 kbps. The test data included sixteen modified intermediate reference system (MIRS) [14] filtered speech sentences spoken by eight female and eight male speakers. Twelve listeners compared the quality of 16 sentences. Each pair of sentences was played in random order and stereo headphones were used in which the same signal was sent to both channels. The scoring method was based on a three-way choice: prefer A, prefer B, or no preference. The results of the A/B comparison are shown in Table III. The results show that the subjective quality of the 4 kbps proposed coder is slightly better than that of 6.3 kbps G.723.1 coder.

The performance of the proposed coder was also evaluated using an informal mean opinion score (MOS) test. The test data

TABLE III
A/B TEST RESULTS

Speakers	Proposed coder (4 kbps)	G.723.1 (6.3 kbps)	No preference
Female	33.34%	28.12%	38.54%
Male	36.46%	20.83%	42.71%
Total	34.90%	24.48%	40.62%

TABLE IV
MOS TEST RESULTS

Coders	MOS female	MOS male	MOS overall
Proposed coder at 4 kbps	3.66	3.97	3.82
G.723.1 at 5.3 kbps	3.29	3.64	3.47
G.723.1 at 6.3 kbps	3.51	3.77	3.64
G.729 at 8 kbps	3.97	4.09	4.03

was generated by encoding the same 16 MIRS filtered sentences (eight female, eight male) using several speech codecs. The order of the codecs was randomized, and the same listening equipment as in the previous test was used.

Twelve listeners participated in the informal MOS test. While the number of listeners, due to limited resources, is less than typically employed in the formal MOS tests, we believe it is enough for obtaining informative results. The listeners were asked to rate the quality of each speech sentence using an absolute category rating (ACR) scale from 1 to 5, representing a subjective quality of bad, poor, fair, good and excellent. Coders included in the test are G.723.1 at 5.3 kbps, G.723.1 at 6.3 kbps, G.729 at 8 kbps, and the matching pursuits sinusoidal speech coder at 4 kbps. The test results are depicted in Table IV.

According to the results, the proposed coder at 4 kbps has quality better than G.723.1 at 5.3 kbps and 6.3 kbps, but worse than G.729 at 8 kbps. The proposed codec achieved results better than existing codecs at higher rates under clean speech conditions and comes close to toll quality at 4 kbps.

XV. CONCLUSION

In this paper, a sinusoidal speech model for low bit rate speech coding is described. The parameters of the model are extracted by a closed-loop analysis based on matching pursuits. To efficiently model speech, the phonetic character of individual frames is considered, hence a multimode approach that uses a particular model for each different type (voiced, unvoiced, transition) of speech signal is adopted. To exploit the redundancies in the speech waveform, the dictionary of the matching pursuits analysis is adapted to the different modes of the model and is composed of cosine waveforms. A frequency bin model is introduced to structure and reduce the allowed set of sinusoidal component frequencies in the dictionary. Furthermore by employing the frequency bin model the extracted frequency locations can be quantized without sacrificing perceptual quality. To overcome the modeling problems posed by frames that are nonstationary in nature such

as transition frames, we developed the Frequency Bin Vector Quantization (FBVQ) method which generalizes the analysis to include the search of a family of dictionaries, represented by a vector quantization codebook. Finally, to demonstrate the viability and the advantages of the new models studied, we designed a 4 kbps matching pursuits sinusoidal speech coder. Subjective results indicate that the proposed coder at 4 kbps has quality exceeding the 6.3 kbps G.723.1 coder.

REFERENCES

[1] C. O. Etemoglu, V. Cuperman, and A. Gersho, "Speech coding with an analysis-by-synthesis sinusoidal model," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, 2000, pp. 1371–1374.

[2] J. H. Friedman and W. Stuetzle, "Projection pursuit regression," *J. Amer. Statist. Assoc.*, vol. 76, pp. 817–823, 1981.

[3] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[4] E. B. George and M. J. T. Smith, "Speech analysis/synthesis and modification using an analysis-by-synthesis/overlap-add sinusoidal model," *IEEE Trans. Speech Audio Processing*, vol. 5, pp. 389–406, Sept. 1997.

[5] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. New York: Springer-Verlag, 1976.

[6] O. Ghitza, "Auditory models and human performance in tasks related to speech coding and speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 115–132, Jan. 1994.

[7] R. J. McAulay and T. F. Quatieri, "Sinusoidal coding," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds. Amsterdam, The Netherlands: Elsevier, 1995, ch. 4.

[8] ——, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 744–754, Aug. 1986.

[9] P. Lupini, "Harmonic Coding of Speech at Low Bit Rates," Ph.D. thesis, Simon Fraser Univ., Burnaby, BC, Canada, Sept. 1995.

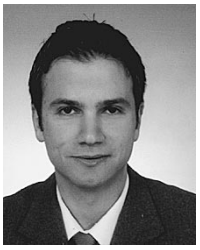
[10] E. Shlomot, V. Cuperman, and A. Gersho, "Combined harmonic and waveform coding of speech at low bit rates," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 1998, pp. 585–588.

[11] C. O. Etemoglu, "Matching pursuits sinusoidal speech coding," Ph.D. dissertation, Univ. of California, Santa Barbara, Dec. 2001.

[12] R. Salami, C. Laflamme, J. Adoul, A. Kataoka, S. Hayashi, T. Moriya, C. Lamblin, D. Massaloux, S. Proust, P. Kroon, and Y. Shoham, "Design and description of cs-acelp: a toll quality 8 kb/s speech coder," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 116–130, Mar. 1998.

[13] D. Talkin, "A robust algorithm for pitch tracking (rapt)," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds. Amsterdam, The Netherlands: Elsevier, 1995, ch. 14.

[14] Subjective Performance Assessment of Telephone-Band and Wideband Digital Codecs, ITU-T Recommend., p. 830, 1996.



Çağrı Ö. Etemoğlu was born in Ankara, Turkey, in 1975. He received the B.S. degree in electrical and electronics engineering from Bilkent University, Ankara, in 1996, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara, in 1997 and 2001, respectively. During his graduate studies, he investigated speech compression techniques and developed a low bit rate speech coder for applications with limited bandwidth.

Between 2001 and 2003, he was with Aware, Inc., Bedford, MA, where he was involved in the development of next-generation DSL modems. Since 2003, he has been with ZTE Corp., Ankara, working on data networks. His research interests include signal processing for communications, coding theory, and compression.

Vladimir Cuperman (F'96) received the B.Sc. degree from the Polytechnic of Bucharest, Bucharest, Romania, in 1960, the M.A. degree in applied mathematics from the University of Bucharest in 1972, and the M.Sc. and Ph.D. degrees from the University of California at Santa Barbara in 1981 and 1983, respectively.

From 1987 to 1996 he was a Professor in the School of Engineering Science at Simon Fraser University, Burnaby, BC, Canada, where his research interests included speech and digital signal processing. From 1996 to 2001, he was the CTO and co-founder of Signalcom, Inc. and held a research position at University of California, Santa Barbara. From 2000 to 2001, he was with Microsoft Corporation.

Dr. Cuperman was Editor for Speech Processing of the IEEE TRANSACTIONS ON COMMUNICATIONS from 1989–1992. He is a Professor Emeritus at Simon Fraser University.