METHOD AND APPARATUS FOR VECTOR QUANTIZING OF A SPECTRAL ENVELOPE REPRESENTATION

## RELATED APPLICATIONS

**[0001]** This application claims benefit of U.S. Provisional Pat. Appl. No. 60/667,901, entitled "CODING THE HIGH-FREQUENCY BAND OF WIDEBAND SPEECH," filed April 1, 2005. This application also claims benefit of U.S. Provisional Pat. Appl. No. 60/673,965, entitled "PARAMETER CODING IN A HIGH-BAND SPEECH CODER," filed April 22, 2005.

## FIELD OF THE INVENTION

**[0002]** This invention relates to signal processing.

## BACKGROUND

**[0003]** A speech encoder sends a characterization of the spectral envelope of a speech signal to a decoder in the form of a vector of line spectral frequencies (LSFs) or a similar representation. For efficient transmission, these LSFs are quantized.

## SUMMARY

**[0004]** A quantizer according to one embodiment is configured to quantize a smoothed value of an input value (such as a vector of line spectral frequencies or portion thereof) to produce a corresponding output value, where the smoothed value is based on a scale factor and a quantization error of a previous output value.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0005]** FIGURE 1a shows a block diagram of a speech encoder E100 according to an embodiment.

**[0006]** FIGURE 1b shows a block diagram of a speech decoder E200.

**[0007]** FIGURE 2 shows an example of a one-dimensional mapping typically performed by a scalar quantizer.

**[0008]** FIGURE 3 shows one simple example of a multidimensional mapping as performed by a vector quantizer.

**[0009]** FIGURE 4a shows one example of a one-dimensional signal, and FIGURE 4b shows an example of a version of this signal after quantization.

**[00010]** FIGURE 4c shows an example of the signal of FIGURE 4a as quantized by a quantizer 230a as shown in FIGURE 5.

**[00011]** FIGURE 4d shows an example of the signal of FIGURE 4a as quantized by a quantizer 230b as shown in FIGURE 6.

**[00012]** FIGURE 5 shows a block diagram of an implementation 230a of a quantizer 230 according to an embodiment.

**[00013]** FIGURE 6 shows a block diagram of an implementation 230b of a quantizer 230 according to an embodiment.

**[00014]** FIGURE 7a shows an example of a plot of frequency vs. log amplitude for a speech signal.

**[00015]** FIGURE 7b shows a block diagram of a basic linear prediction coding system.

**[00016]** FIGURE 8 shows a block diagram of an implementation A122 of narrowband encoder A120.

**[00017]** FIGURE 9 shows a block diagram of an implementation B112 of narrowband encoder B110.

**[00018]** FIGURE 10a is a block diagram of a wideband speech encoder A100.

**[00019]** FIGURE 10b is a block diagram of an implementation A102 of wideband speech encoder A100.

**[00020]** FIGURE 11a is a block diagram of a wideband speech decoder B100 corresponding to wideband speech encoder A100.

**[00021]** FIGURE 11b is an example of a wideband speech decoder B102 corresponding to wideband speech encoder A102.

## DETAILED DESCRIPTION

**[00022]** Due to quantization error, the spectral envelope reconstructed in the decoder may exhibit excessive fluctuations. These fluctuations may produce an objectionable "warbly" quality in the decoded signal. Embodiments include system, methods, and apparatus configured to perform high-quality wideband speech coding using temporal noise shaping quantization of spectral envelope parameters. Features include fixed or adaptive smoothing of coefficient representations such as highband LSFs. Particular applications described herein include a wideband speech coder that combines a narrowband signal with a highband signal.

**[00023]** Unless expressly limited by its context, the term "calculating" is used herein to indicate any of its ordinary meanings, such as computing, generating, and selecting from a list of values. Where the term "comprising" is used in the present description and claims, it does not exclude other elements or operations. The term "A is based on B" is used to indicate any of its ordinary meanings, including the cases (i) "A is equal to B" and (ii) "A is based on at least B." The term "Internet Protocol" includes version 4, as described in IETF (Internet Engineering Task Force) RFC (Request for Comments) 791, and subsequent versions such as version 6.

**[00024]** A speech encoder may be implemented according to a source-filter model that encodes the input speech signal as a set of parameters that describe a filter. For example, a spectral envelope of a speech signal is characterized by a number of peaks that represent resonances of the vocal tract and are called formants. FIGURE 7a shows one example of such a spectral envelope. Most speech coders encode at least this coarse spectral structure as a set of parameters such as filter coefficients.

**[00025]** FIGURE 1a shows a block diagram of a speech encoder E100 according to an embodiment. As shown in this example, the analysis module may be implemented as a linear prediction coding (LPC) analysis module 210 that encodes the spectral envelope of the speech signal S1 as a set of linear prediction (LP) coefficients (e.g., coefficients of an all-pole filter 1/A(z)). The analysis module typically processes the input signal as a series of nonoverlapping frames, with a new set of coefficients being calculated for each frame. The frame period is generally a period over which the signal may be

expected to be locally stationary; one common example is 20 milliseconds (equivalent to 160 samples at a sampling rate of 8 kHz). One example of a lowband LPC analysis module is configured to calculate a set of ten LP filter coefficients to characterize the formant structure of each 20-millisecond frame of lowband speech signal S20, and one example of a highband LPC analysis module is configured to calculate a set of six (alternatively, eight) LP filter coefficients to characterize the formant structure of each 20-millisecond frame of highband speech signal S30. It is also possible to implement the analysis module to process the input signal as a series of overlapping frames.

**[00026]** The analysis module may be configured to analyze the samples of each frame directly, or the samples may be weighted first according to a windowing function (for example, a Hamming window). The analysis may also be performed over a window that is larger than the frame, such as a 30-msec window. This window may be symmetric (e.g. 5-20-5, such that it includes the 5 milliseconds immediately before and after the 20-millisecond frame) or asymmetric (e.g. 10-20, such that it includes the last 10 milliseconds of the preceding frame). An LPC analysis module is typically configured to calculate the LP filter coefficients using a Levinson-Durbin recursion or the Leroux-Gueguen algorithm. In another implementation, the analysis module may be configured to calculate a set of cepstral coefficients for each frame instead of a set of LP filter coefficients.

**[00027]** The output rate of a speech encoder may be reduced significantly, with relatively little effect on reproduction quality, by quantizing the filter parameters. Linear prediction filter coefficients are difficult to quantize efficiently and are usually mapped by the speech encoder into another representation, such as line spectral pairs (LSPs) or line spectral frequencies (LSFs), for quantization and/or entropy encoding. Speech encoder E100 as shown in FIGURE 1a includes a LP filter coefficient-to-LSF transform 220 configured to transform the set of LP filter coefficients into a corresponding vector of LSFs. Other one-to-one representations of LP filter coefficients include parcor coefficients; log-area-ratio values; immittance spectral pairs (ISPs); and immittance spectral frequencies (ISFs), which are used in the GSM (Global System for Mobile Communications) AMR-WB (Adaptive Multirate-Wideband) codec. Typically a transform between a set of LP filter coefficients and a corresponding set of LSFs is

reversible, but embodiments also include implementations of a speech encoder in which the transform is not reversible without error.

**[00028]** A speech encoder typically includes a quantizer configured to quantize the set of narrowband LSFs (or other coefficient representation) and to output the result of this quantization as the filter parameters. Quantization is typically performed using a vector quantizer that encodes the input vector as an index to a corresponding vector entry in a table or codebook. Such a quantizer may also be configured to perform classified vector quantization. For example, such a quantizer may be configured to select one of a set of codebooks based on information that has already been coded within the same frame (e.g., in the lowband channel and/or in the highband channel). Such a technique typically provides increased coding efficiency at the expense of additional codebook storage.

**[00029]** FIGURE 1b shows a block diagram of a corresponding speech decoder E200 that includes an inverse quantizer 310 configured to dequantize the quantized LSFs S3, and a LSF-to-LP filter coefficient transform 320 configured to transform the dequantized LSF vector into a set of LP filter coefficients. A synthesis filter 330 configured according to the LP filter coefficients is typically driven by an excitation signal to produce a synthesized reproduction S5 of the input speech signal. The excitation signal may be based on a random noise signal and/or on a quantized representation of the residual as sent by the encoder. In some multiband coders such as wideband speech encoder A100 and decoder B100 (as described herein with reference to, e.g., FIGURES 10a,b and 11a,b), the excitation signal for one band is derived from the excitation signal for another band.

**[00030]** Quantization of the LSFs introduces a random error that is usually uncorrelated from one frame to the next. This error may cause the quantized LSFs to be less smooth than the unquantized LSFs and may reduce the perceptual quality of the decoded signal. Independent quantization of LSF vectors generally increases the amount of spectral fluctuation from frame to frame compared to the unquantized LSF vectors, and these spectral fluctuations may cause the decoded signal to sound unnatural.

**[00031]** One complicated solution was proposed by Knagenhjelm and Kleijn, in which a smoothing of the dequantized LSF parameters is performed in the decoder. This reduces the spectral fluctuations, but comes at the cost of additional delay. This application describes method that use temporal noise shaping on the encoder side, such that spectral fluctuations may be reduced without additional delay.

**[00032]** A quantizer is typically configured to map an input value to one of a set of discrete output values. A limited number of output values are available, such that a range of input values is mapped to a single output value. Quantization increases coding efficiency because an index that indicates the corresponding output value may be transmitted in fewer bits than the original input value. FIGURE 2 shows an example of a one-dimensional mapping typically performed by a scalar quantizer.

**[00033]** The quantizer could equally well be a vector quantizer, and LSFs are typically quantized using a vector quantizer. FIGURE 3 shows one simple example of a multidimensional mapping as performed by a vector quantizer. In this example, the input space is divided into a number of Voronoi regions (e.g., according to a nearest-neighbor criterion). The quantization maps each input value to a value that represents the corresponding Voronoi region (typically, the centroid), shown here as a point. In this example, the input space is divided into six regions, such that any input value may be represented by an index having only six different states.

**[00034]** If the input signal is very smooth, it can happen sometimes that the quantized output is much less smooth, according to a minimum step between values in the output space of the quantization. FIGURE 4a shows one example of a smooth one-dimensional signal that varies only within one quantization level (only one such level is shown here), and FIGURE 4b shows an example of this signal after quantization. Even though the input in FIGURE 4a varies over only a small range, the resulting output in FIGURE 4b contains more abrupt transitions and is much less smooth. Such an effect may lead to audible artifacts, and it may be desirable to reduce this effect for LSFs (or other representation of the spectral envelope to be quantized). For example, LSF quantization performance may be improved by incorporating temporal noise shaping.

**[00035]** In a method according to one embodiment, a vector of spectral envelope parameters is estimated once for every frame (or other block) of speech in the encoder.

The parameter vector is quantized for efficient transmission to the decoder. After quantization, the quantization error (defined as the difference between quantized and unquantized parameter vector) is stored. The quantization error of frame N–1 is reduced by a scale factor and added to the parameter vector of frame N, before quantizing the parameter vector of frame N. It may be desirable for the value of the scale factor to be smaller when the difference between current and previous estimated spectral envelopes is relatively large.

**[00036]** In a method according to one embodiment, the LSF quantization error vector is computed for each frame and multiplied by a scale factor $b$ having a value less than 1.0. Before quantization, the scaled quantization error for the previous frame is added to the LSF vector (input value V10). A quantization operation of such a method may be described by an expression such as the following:

$$y(n) = Q(s(n) + b[y(n-1) - s(n-1)]),$$

where $s(n)$ is the smoothed LSF vector pertaining to frame $n$, $y(n)$ is the quantized LSF vector pertaining to frame $n$, $Q(\cdot)$ is a nearest-neighbor quantization operation, and $b$ is the scale factor.

**[00037]** A quantizer 230 according to an embodiment is configured to produce a quantized output value V30 of a smoothed value V20 of an input value V10 (e.g., an LSF vector), where the smoothed value V20 is based on a scale factor $b$ V40 and a quantization error of a previous output value V30a. Such a quantizer may be applied to reduce spectral fluctuations without additional delay. FIGURE 5 shows a block diagram of one implementation 230a of quantizer 230, in which values that may be particular to this implementation are indicated by the index a. In this example, a quantization error is computed by subtracting the current value of smoothed value V20a from the current output value V30a as dequantized by inverse quantizer Q20. The error is stored to a delay element DE10. Smoothed value V20a itself is a sum of the current input value V10 and the quantization error of the previous frame as scaled (e.g. multiplied) by scale factor V40. Quantizer 230a may also be implemented such that the scale factor V40 is applied before storage of the quantization error to delay element DE10 instead.

**[00038]** FIGURE 4c shows an example of a (dequantized) sequence of output values V30a as produced by quantizer 230a in response to the input signal of FIGURE 4a. In this example, the value of $b$ is fixed at 0.5. It may be seen that the signal of FIGURE 4c is smoother than the fluctuating signal of FIGURE 4a.

**[00039]** It may be desirable to use a recursive function to calculate the feedback amount. For example, the quantization error may be calculated with respect to the current input value rather than with respect to the current smoothed value. Such a method may be described by an expression such as the following:

$$y(n) = Q[s(n)], \; s(n) = x(n) + b[y(n-1) - s(n-1)],$$

where $x(n)$ is the input LSF vector pertaining to frame $n$.

**[00040]** FIGURE 6 shows a block diagram of an implementation 230b of quantizer 230, in which values that may be particular to this implementation are indicated by the index b. In this example, a quantization error is computed by subtracting the current input value V10 from the current output value V30b as dequantized by inverse quantizer Q20. The error is stored to delay element DE10. Smoothed value V20b is a sum of the current input value V10 and the quantization error of the previous frame as scaled (e.g. multiplied) by scale factor V40. Quantizer 230b may also be implemented such that the scale factor V40 is applied before storage of the quantization error to delay element DE10 instead. It is also possible to use different values of scale factor V40 in implementation 230a as opposed to implementation 230b.

**[00041]** FIGURE 4d shows an example of a (dequantized) sequence of output values V30b as produced by quantizer 230b in response to the input signal of FIGURE 4a. In this example, the value of $b$ is fixed at 0.5. It may be seen that the signal of FIGURE 4d is smoother than the fluctuating signal of FIGURE 4a.

**[00042]** It is noted that embodiments as shown herein may be implemented by replacing or augmenting an existing quantizer Q10 according to an arrangement as shown in FIGURE 5 or 6. For example, quantizer Q10 may be implemented as a predictive vector quantizer, a multi-stage quantizer, a split vector quantizer, or according to any other scheme for LSF quantization.

**[00043]** In one example, the value of $b$ is fixed at a desired value between 0 and 1. Alternatively, it may be desired to adjust the value of the scale factor $b$ dynamically. For example, it may be desired to adjust the value of the scale factor $b$ depending on a degree of fluctuation already present in the unquantized LSF vectors. When the difference between the current and previous LSF vectors is large, the scale factor is close to zero and almost no noise shaping results. When the current LSF vector differs little from the previous one, the scale factor is close to 1.0. In such manner, transitions in the spectral envelope over time may be retained, minimizing spectral distortion when the speech signal is changing, while spectral fluctuations may be reduced when the speech signal is relatively constant from one frame to the next.

**[00044]** The value of $b$ may be made proportional to the distance between consecutive LSFs, and any of various distances between vectors may be used to determine the change between LSFs. The Euclidean norm is typically used, but others which may be used include Manhattan distance (1-norm), Chebyshev distance (infinity norm), Mahalanobis distance, Hamming distance.

**[00045]** It may be desired to use a weighted distance measure to determine a change between consecutive LSF vectors. For example, the distance $d$ may be calculated according to an expression such as the following:

$$d = \sum_{i=1}^{P} c_i (l_i - \hat{l}_i)^2 ,$$

where $l$ indicates the current LSF vector, $\hat{l}$ indicates the previous LSF vector, $P$ indicates the number of elements in each LSF vector, the index $i$ indicates the LSF vector element, and $c$ indicates a vector of weighting factors. The values of $c$ may be selected to emphasize lower frequency components that are more perceptually significant. In one example, $c_i$ has the value 1.0 for $i$ from 1 to 8, 0.8 for $i = 9$, and 0.4 for $i = 10$.

**[00046]** In another example, the distance $d$ between consecutive LSF vectors may be calculated according to an expression such as the following:

**[00047]** $$d = \sum_{i=1}^{P} c_i w_i (l_i - \hat{l}_i)^2 ,$$

**[00048]** where $w$ indicates a vector of variable weighting factors. In one such example, $w_i$ has the value $P(f_i)^r$ , where $P$ denotes the LPC power spectrum evaluated at the corresponding frequency $f$, and $r$ is a constant having a typical value of, e.g., 0.15 or 0.3. In another example, the values of $w$ are selected according to a corresponding weight function used in the ITU-T G.729 standard:

**[00049]** $$w_i = \begin{cases} 1.0 & if\,(2\pi(l_{i+1} - l_{i-1}) - 1) > 0 \\ 10(2\pi(l_{i+1} - l_{i-1}) - 1)^2 + 1 & otherwise \end{cases},$$

**[00050]** with boundary values close to 0 and 0.5 being selected in place of $l_{i-1}$ and $l_{i+1}$ for the lowest and highest elements of $w$, respectively. In such cases, $c_i$ may have values as indicated above. In another example, $c_i$ has the value 1.0, except for $c_4$ and $c_5$ which have the value 1.2.

**[00051]** It may be appreciated from FIGURES 4a–d that on a frame-by-frame basis, a temporal noise shaping method as described herein may increase the quantization error. Although the absolute squared error of the quantization operation may increase, however, a potential advantage is that the quantization error may be moved to a different part of the spectrum. For example, the quantization error may be moved to lower frequencies, thus becoming more smooth. As the input signal is also smooth, a smoother output signal may be obtained as a sum of the input signal and the smoothed quantization error.

**[00052]** FIGURE 7b shows an example of a basic source-filter arrangement as applied to coding of the spectral envelope of a narrowband signal S20. An analysis module calculates a set of parameters that characterize a filter corresponding to the speech sound over a period of time (typically 20 msec). A whitening filter (also called an analysis or prediction error filter) configured according to those filter parameters removes the spectral envelope to spectrally flatten the signal. The resulting whitened signal (also called a residual) has less energy and thus less variance and is easier to encode than the original speech signal. Errors resulting from coding of the residual signal may also be spread more evenly over the spectrum. The filter parameters and residual are typically quantized for efficient transmission over the channel. At the decoder, a synthesis filter configured according to the filter parameters is excited by a signal based on the residual to produce a synthesized version of the original speech

sound. The synthesis filter is typically configured to have a transfer function that is the inverse of the transfer function of the whitening filter. FIGURE 8 shows a block diagram of a basic implementation A122 of narrowband encoder A120.

**[00053]** As seen in FIGURE 8, narrowband encoder A122 also generates a residual signal by passing narrowband signal S20 through a whitening filter 260 (also called an analysis or prediction error filter) that is configured according to the set of filter coefficients. In this particular example, whitening filter 260 is implemented as a FIR filter, although IIR implementations may also be used. This residual signal will typically contain perceptually important information of the speech frame, such as long-term structure relating to pitch, that is not represented in narrowband filter parameters S40. Quantizer 270 is configured to calculate a quantized representation of this residual signal for output as encoded narrowband excitation signal S50. Such a quantizer typically includes a vector quantizer that encodes the input vector as an index to a corresponding vector entry in a table or codebook. Alternatively, such a quantizer may be configured to send one or more parameters from which the vector may be generated dynamically at the decoder, rather than retrieved from storage, as in a sparse codebook method. Such a method is used in coding schemes such as algebraic CELP (codebook excitation linear prediction) and codecs such as 3GPP2 (Third Generation Partnership 2) EVRC (Enhanced Variable Rate Codec).

**[00054]** It is desirable for narrowband encoder A120 to generate the encoded narrowband excitation signal according to the same filter parameter values that will be available to the corresponding narrowband decoder. In this manner, the resulting encoded narrowband excitation signal may already account to some extent for nonidealities in those parameter values, such as quantization error. Accordingly, it is desirable to configure the whitening filter using the same coefficient values that will be available at the decoder. In the basic example of encoder A122 as shown in FIGURE 8, inverse quantizer 240 dequantizes narrowband filter parameters S40, LSF-to-LP filter coefficient transform 250 maps the resulting values back to a corresponding set of LP filter coefficients, and this set of coefficients is used to configure whitening filter 260 to generate the residual signal that is quantized by quantizer 270.

**[00055]** Some implementations of narrowband encoder A120 are configured to calculate encoded narrowband excitation signal S50 by identifying one among a set of

codebook vectors that best matches the residual signal. It is noted, however, that narrowband encoder A120 may also be implemented to calculate a quantized representation of the residual signal without actually generating the residual signal. For example, narrowband encoder A120 may be configured to use a number of codebook vectors to generate corresponding synthesized signals (e.g., according to a current set of filter parameters), and to select the codebook vector associated with the generated signal that best matches the original narrowband signal S20 in a perceptually weighted domain.

**[00056]** FIGURE 9 shows a block diagram of an implementation B112 of narrowband decoder B110. Inverse quantizer 310 dequantizes narrowband filter parameters S40 (in this case, to a set of LSFs), and LSF-to-LP filter coefficient transform 320 transforms the LSFs into a set of filter coefficients (for example, as described above with reference to inverse quantizer 240 and transform 250 of narrowband encoder A122). Inverse quantizer 340 dequantizes narrowband residual signal S40 to produce a narrowband excitation signal S80. Based on the filter coefficients and narrowband excitation signal S80, narrowband synthesis filter 330 synthesizes narrowband signal S90. In other words, narrowband synthesis filter 330 is configured to spectrally shape narrowband excitation signal S80 according to the dequantized filter coefficients to produce narrowband signal S90. Narrowband decoder B112 also provides narrowband excitation signal S80 to highband encoder A200, which uses it to derive the highband excitation signal S120 as described herein. In some implementations as described below, narrowband decoder B110 may be configured to provide additional information to highband decoder B200 that relates to the narrowband signal, such as spectral tilt, pitch gain and lag, and speech mode. The system of narrowband encoder A122 and narrowband decoder B112 is a basic example of an analysis-by-synthesis speech codec.

**[00057]** Voice communications over the public switched telephone network (PSTN) have traditionally been limited in bandwidth to the frequency range of 300–3400 kHz. New networks for voice communications, such as cellular telephony and voice over IP (VoIP), may not have the same bandwidth limits, and it may be desirable to transmit and receive voice communications that include a wideband frequency range over such networks. For example, it may be desirable to support an audio frequency range that extends down to 50 Hz and/or up to 7 or 8 kHz. It may also be desirable to support

other applications, such as high-quality audio or audio/video conferencing, that may have audio speech content in ranges outside the traditional PSTN limits.

**[00058]** One approach to wideband speech coding involves scaling a narrowband speech coding technique (e.g., one configured to encode the range of 0–4 kHz) to cover the wideband spectrum. For example, a speech signal may be sampled at a higher rate to include components at high frequencies, and a narrowband coding technique may be reconfigured to use more filter coefficients to represent this wideband signal. Narrowband coding techniques such as CELP (codebook excited linear prediction) are computationally intensive, however, and a wideband CELP coder may consume too many processing cycles to be practical for many mobile and other embedded applications. Encoding the entire spectrum of a wideband signal to a desired quality using such a technique may also lead to an unacceptably large increase in bandwidth. Moreover, transcoding of such an encoded signal would be required before even its narrowband portion could be transmitted into and/or decoded by a system that only supports narrowband coding.

**[00059]** FIGURE 10a shows a block diagram of a wideband speech encoder A100 that includes separate narrowband and highband speech encoders A120 and A200, respectively. Either or both of narrowband and highband speech encoders A120 and A200 may be configured to perform quantization of LSFs (or another coefficient representation) using an implementation of quantizer 230 as disclosed herein. FIGURE 11a shows a block diagram of a corresponding wideband speech decoder B100. Filter banks A110 and B120 may be implemented to produce narrowband signal S20 and highband signal S30 from a wideband speech signal S10 according to the principles and implementations disclosed in the Patent Application "SYSTEMS, METHODS, AND APPARATUS FOR SPEECH SIGNAL FILTERING" filed herewith, Attorney Docket No. 050551, and this disclosure of such filter banks therein is hereby incorporated by reference.

**[00060]** It may be desirable to implement wideband speech coding such that at least the narrowband portion of the encoded signal may be sent through a narrowband channel (such as a PSTN channel) without transcoding or other significant modification. Efficiency of the wideband coding extension may also be desirable, for example, to

avoid a significant reduction in the number of users that may be serviced in applications such as wireless cellular telephony and broadcasting over wired and wireless channels.

**[00061]** One approach to wideband speech coding involves extrapolating the highband spectral envelope from the encoded narrowband spectral envelope. While such an approach may be implemented without any increase in bandwidth and without a need for transcoding, however, the coarse spectral envelope or formant structure of the highband portion of a speech signal generally cannot be predicted accurately from the spectral envelope of the narrowband portion.

**[00062]** One particular example of wideband speech encoder A100 is configured to encode wideband speech signal S10 at a rate of about 8.55 kbps (kilobits per second), with about 7.55 kbps being used for narrowband filter parameters S40 and encoded narrowband excitation signal S50, and about 1 kbps being used for highband coding parameters (e.g., filter parameters and/or gain parameters) S60.

**[00063]** It may be desired to combine the encoded lowband and highband signals into a single bitstream. For example, it may be desired to multiplex the encoded signals together for transmission (e.g., over a wired, optical, or wireless transmission channel), or for storage, as an encoded wideband speech signal. FIGURE 10b shows a block diagram of wideband speech encoder A102 that includes a multiplexer A130 configured to combine narrowband filter parameters S40, an encoded narrowband excitation signal S50, and highband coding parameters S60 into a multiplexed signal S70. FIGURE 11b shows a block diagram of a corresponding implementation B102 of wideband speech decoder B100.

**[00064]** It may be desirable for multiplexer A130 to be configured to embed the encoded lowband signal (including lowband filter parameters S40 and encoded lowband excitation signal S50) as a separable substream of multiplexed signal S70, such that the encoded lowband signal may be recovered and decoded independently of another portion of multiplexed signal S70 such as a highband and/or very-low-band signal. For example, multiplexed signal S70 may be arranged such that the encoded lowband signal may be recovered by stripping away the highband coding parameters S60. One potential advantage of such a feature is to avoid the need for transcoding the encoded

wideband signal before passing it to a system that supports decoding of the lowband signal but does not support decoding of the highband portion.

**[00065]** An apparatus including a noise-shaping quantizer and/or a lowband, highband, and/or wideband speech encoder as described herein may also include circuitry configured to transmit the encoded signal into a transmission channel such as a wired, optical, or wireless channel. Such an apparatus may also be configured to perform one or more channel encoding operations on the signal, such as error correction encoding (e.g., rate-compatible convolutional encoding) and/or error detection encoding (e.g., cyclic redundancy encoding), and/or one or more layers of network protocol encoding (e.g., Ethernet, TCP/IP, cdma2000).

**[00066]** It may be desirable to implement a lowband speech encoder A120 as an analysis-by-synthesis speech encoder. Codebook excitation linear prediction (CELP) coding is one popular family of analysis-by-synthesis coding, and implementations of such coders may perform waveform encoding of the residual, including such operations as selection of entries from fixed and adaptive codebooks, error minimization operations, and/or perceptual weighting operations. Other implementations of analysis-by-synthesis coding include mixed excitation linear prediction (MELP), algebraic CELP (ACELP), relaxation CELP (RCELP), regular pulse excitation (RPE), multi-pulse CELP (MPE), and vector-sum excited linear prediction (VSELP) coding. Related coding methods include multi-band excitation (MBE) and prototype waveform interpolation (PWI) coding. Examples of standardized analysis-by-synthesis speech codecs include the ETSI (European Telecommunications Standards Institute)-GSM full rate codec (GSM 06.10), which uses residual excited linear prediction (RELP); the GSM enhanced full rate codec (ETSI-GSM 06.60); the ITU (International Telecommunication Union) standard 11.8 kb/s G.729 Annex E coder; the IS (Interim Standard)-641 codecs for IS-136 (a time-division multiple access scheme); the GSM adaptive multirate (GSM-AMR) codecs; and the 4GV™ (Fourth-Generation Vocoder™) codec (QUALCOMM Incorporated, San Diego, CA). Existing implementations of RCELP coders include the Enhanced Variable Rate Codec (EVRC), as described in Telecommunications Industry Association (TIA) IS-127, and the Third Generation Partnership Project 2 (3GPP2) Selectable Mode Vocoder (SMV). The various lowband, highband, and wideband encoders described herein may be implemented according to any of these technologies,

or any other speech coding technology (whether known or to be developed) that represents a speech signal as (A) a set of parameters that describe a filter and (B) a quantized representation of a residual signal that provides at least part of an excitation used to drive the described filter to reproduce the speech signal.

**[00067]** As mentioned above, embodiments as described herein include implementations that may be used to perform embedded coding, supporting compatibility with narrowband systems and avoiding a need for transcoding. Support for highband coding may also serve to differentiate on a cost basis between chips, chipsets, devices, and/or networks having wideband support with backward compatibility, and those having narrowband support only. Support for highband coding as described herein may also be used in conjunction with a technique for supporting lowband coding, and a system, method, or apparatus according to such an embodiment may support coding of frequency components from, for example, about 50 or 100 Hz up to about 7 or 8 kHz.

**[00068]** As mentioned above, adding highband support to a speech coder may improve intelligibility, especially regarding differentiation of fricatives. Although such differentiation may usually be derived by a human listener from the particular context, highband support may serve as an enabling feature in speech recognition and other machine interpretation applications, such as systems for automated voice menu navigation and/or automatic call processing.

**[00069]** An apparatus according to an embodiment may be embedded into a portable device for wireless communications, such as a cellular telephone or personal digital assistant (PDA). Alternatively, such an apparatus may be included in another communications device such as a VoIP handset, a personal computer configured to support VoIP communications, or a network device configured to route telephonic or VoIP communications. For example, an apparatus according to an embodiment may be implemented in a chip or chipset for a communications device. Depending upon the particular application, such a device may also include such features as analog-to-digital and/or digital-to-analog conversion of a speech signal, circuitry for performing amplification and/or other signal processing operations on a speech signal, and/or radio-frequency circuitry for transmission and/or reception of the coded speech signal.

**[00070]** It is explicitly contemplated and disclosed that embodiments may include and/or be used with any one or more of the other features disclosed in the U.S. Provisional Pat. Appls. Nos. 60/667,901 and 60/673,965 of which this application claims benefit and/or the related applications filed herewith and listed above. Such features include shifting of highband signal S30 and/or highband excitation signal S120 according to a regularization or other shift of narrowband excitation signal S80 or narrowband residual signal S50. Such features include adaptive smoothing of LSFs, which may be performed prior to a quantization as described herein. Such features also include fixed or adaptive smoothing of a gain envelope, and adaptive attenuation of a gain envelope.

**[00071]** The foregoing presentation of the described embodiments is provided to enable any person skilled in the art to make or use the present invention. Various modifications to these embodiments are possible, and the generic principles presented herein may be applied to other embodiments as well. For example, an embodiment may be implemented in part or in whole as a hard-wired circuit, as a circuit configuration fabricated into an application-specific integrated circuit, or as a firmware program loaded into non-volatile storage or a software program loaded from or into a data storage medium as machine-readable code, such code being instructions executable by an array of logic elements such as a microprocessor or other digital signal processing unit. The data storage medium may be an array of storage elements such as semiconductor memory (which may include without limitation dynamic or static RAM (random-access memory), ROM (read-only memory), and/or flash RAM), or ferroelectric, magnetoresistive, ovonic, polymeric, or phase-change memory; or a disk medium such as a magnetic or optical disk. The term "software" should be understood to include source code, assembly language code, machine code, binary code, firmware, macrocode, microcode, any one or more sets or sequences of instructions executable by an array of logic elements, and any combination of such examples.

**[00072]** The various elements of implementations of a noise-shaping quantizer; highband speech encoder A200; wideband speech encoder A100 and A102; and arrangements including one or more such apparatus, may be implemented as electronic and/or optical devices residing, for example, on the same chip or among two or more chips in a chipset, although other arrangements without such limitation are also

contemplated. One or more elements of such an apparatus may be implemented in whole or in part as one or more sets of instructions arranged to execute on one or more fixed or programmable arrays of logic elements (e.g., transistors, gates) such as microprocessors, embedded processors, IP cores, digital signal processors, FPGAs (field-programmable gate arrays), ASSPs (application-specific standard products), and ASICs (application-specific integrated circuits). It is also possible for one or more such elements to have structure in common (e.g., a processor used to execute portions of code corresponding to different elements at different times, a set of instructions executed to perform tasks corresponding to different elements at different times, or an arrangement of electronic and/or optical devices performing operations for different elements at different times). Moreover, it is possible for one or more such elements to be used to perform tasks or execute other sets of instructions that are not directly related to an operation of the apparatus, such as a task relating to another operation of a device or system in which the apparatus is embedded.

**[00073]** Embodiments also include additional methods of speech processing, speech encoding, and highband burst suppression as are expressly disclosed herein, e.g., by descriptions of structural embodiments configured to perform such methods. Each of these methods may also be tangibly embodied (for example, in one or more data storage media as listed above) as one or more sets of instructions readable and/or executable by a machine including an array of logic elements (e.g., a processor, microprocessor, microcontroller, or other finite state machine). Thus, the present invention is not intended to be limited to the embodiments shown above but rather is to be accorded the widest scope consistent with the principles and novel features disclosed in any fashion herein.

**WHAT IS CLAIMED IS:**

1. A method for signal processing, said method comprising:

encoding a first frame and a second frame of a speech signal to produce corresponding first and second vectors, wherein the first vector represents a spectral envelope of the speech signal during the first frame and the second vector represents a spectral envelope of the speech signal during the second frame;

generating a first quantized vector, said generating including quantizing a third vector that is based on at least a portion of the first vector;

calculating a quantization error of the first quantized vector;

calculating a fourth vector, said calculating including adding a scaled version of the quantization error to at least a portion of the second vector; and

quantizing the fourth vector.

2. The method according to claim 1, wherein said calculating a quantization error includes calculating a difference between the first quantized vector and the third vector.

3. The method according to claim 1, wherein said calculating a quantization error includes calculating a difference between the first quantized vector and at least a portion of the first vector.

4. The method according to claim 1, said method including calculating the scaled quantization error, said calculating comprising multiplying the quantization error by a scale factor,

wherein the scale factor is based on a distance between at least a portion of the first vector and a corresponding portion of the second vector.

5. The method according to claim 4, wherein each among the first and second vectors includes a plurality of line spectral frequencies.

6. The method according to claim 1, wherein each among the first and second vectors includes a representation of a plurality of linear prediction filter coefficients.

7. The method according to claim 1, wherein each among the first and second vectors includes a plurality of line spectral frequencies.

8. A data storage medium having machine-executable instructions describing the method according to claim 1.

9. An apparatus comprising:

a speech encoder configured to encode a first frame of a speech signal into at least a first and to encode a second frame of a speech signal into at least a second vector, wherein the first vector represents a spectral envelope of the speech signal during the first frame and the second vector represents a spectral envelope of the speech signal during the second frame,

a quantizer configured to quantize a third vector that is based on at least a portion of the first vector to generate a first quantized vector;

a first adder configured to calculate a quantization error of the first quantized vector; and

a second adder configured to add a scaled version of the quantization error to at least a portion of the second vector to calculate a fourth vector;

wherein said quantizer is configured to quantize the fourth vector.

10. The apparatus according to claim 9, wherein said first adder is configured to calculate the quantization error based on a difference between the first quantized vector and the third vector.

11. The apparatus according to claim 9, wherein said first adder is configured to calculate the quantization error based on a difference between the first quantized vector and at least a portion of the first vector.

12. The apparatus according to claim 9, said apparatus including a multiplier configured to calculating the scaled quantization error based on a product of the quantization error and a scale factor,

wherein said apparatus includes logic configured to calculate the scale factor based on a distance between at least a portion of the first vector and a corresponding portion of the second vector.

13. The apparatus according to claim 12, wherein each among the first and second vectors includes a plurality of line spectral frequencies.

14. The apparatus according to claim 9, wherein each among the first and second vectors includes a representation of a plurality of linear prediction filter coefficients.

15. The apparatus according to claim 9, wherein each among the first and second vectors includes a plurality of line spectral frequencies.

16. The apparatus according to claim 9, said apparatus comprising a device for wireless communications.

17. The apparatus according to claim 9, said apparatus comprising a device configured to transmit a plurality of packets compliant with a version of the Internet Protocol, wherein the plurality of packets describes the first quantized vector.

18. An apparatus comprising:

means for encoding a first frame and a second frame of a speech signal to produce corresponding first and second vectors, wherein the first vector represents a spectral envelope of the speech signal during the first frame and the second vector represents a spectral envelope of the speech signal during the second frame;

means for generating a first quantized vector, said generating including quantizing a third vector that is based on at least a portion of the first vector;

means for calculating a quantization error of the first quantized vector; and

means for calculating a fourth vector, said calculating including adding a scaled version of the quantization error to at least a portion of the second vector,

wherein said means for generating a first quantized vector is configured to quantize the fourth vector.

19. The apparatus according to claim 18, wherein said means for calculating a quantization error is configured to calculate the quantization error based on a difference between the first quantized vector and the third vector.

20. The apparatus according to claim 18, wherein said means for calculating a quantization error is configured to calculate the quantization error based on a difference between the first quantized vector and at least a portion of the first vector.

21. The apparatus according to claim 18, said apparatus including means for calculating the scaled quantization error, said calculating comprising multiplying the quantization error by a scale factor,

wherein said apparatus comprises logic configured to calculate the scale factor based on a distance between at least a portion of the first vector and a corresponding portion of the second vector.

22. The apparatus according to claim 21, wherein each among the first and second vectors includes a plurality of line spectral frequencies.

23. The apparatus according to claim 18, said apparatus comprising a device for wireless communications.