(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 3 132 443 B1

(12) EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention of the grant of the patent:
26.12.2018 Bulletin 2018/52

(21) Application number: **14889618.6**

(22) Date of filing: **25.07.2014**

(51) Int Cl.:
G10L 19/12 *(2013.01)*
G10L 19/06 *(2013.01)*
G10L 19/26 *(2013.01)*
G10L 19/07 (2013.01)
G10L 21/038 (2013.01)

(86) International application number:
PCT/CA2014/050706

(87) International publication number:
WO 2015/157843 (22.10.2015 Gazette 2015/42)

(54) **METHODS, ENCODER AND DECODER FOR LINEAR PREDICTIVE ENCODING AND DECODING OF SOUND SIGNALS UPON TRANSITION BETWEEN FRAMES HAVING DIFFERENT SAMPLING RATES**

VERFAHREN, CODIERER UND DECODIERER ZUR LINEAREN PRÄDIKTIVEN CODIERUNG UND DECODIERUNG VON TONSIGNALEN BEIM ÜBERGANG ZWISCHEN RAHMEN MIT UNTERSCHIEDLICHEN ABTASTRATEN

PROCÉDÉS, CODEUR ET DÉCODEUR POUR LE CODAGE ET LE DÉCODAGE PRÉDICTIFS LINÉAIRES DE SIGNAUX SONORES LORS DE LA TRANSITION ENTRE DES TRAMES POSSÉDANT DES TAUX D'ÉCHANTILLONNAGE DIFFÉRENTS

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR

(30) Priority: **17.04.2014 US 201461980865 P**

(43) Date of publication of application:
22.02.2017 Bulletin 2017/08

(73) Proprietor: **VoiceAge Corporation**
Town of Mount Royal, Quebec H3R 2H6 (CA)

(72) Inventors:
- **SALAMI, Redwan**
  Saint-Laurent, Québec H4N 4A2 (CA)
- **EKSLER, Vaclav**
  Sherbrooke, Québec J1K 2C3 (CA)

(74) Representative: **Ipside**
6, Impasse Michel Labrousse
31100 Toulouse (FR)

(56) References cited:
US-A1- 2008 077 401
US-A1- 2013 332 153
US-B2- 8 315 863
US-B2- 8 401 843
US-B2- 8 589 151

- **BESSETTE ET AL.: 'The Adaptive Multirate Wideband Speech Codec (AMR-WB' IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING vol. 10, no. 8, November 2002, ISSN 1063-6676 pages 620 - 636, XP055231143**

Printed by Jouve, 75001 PARIS (FR)

Description

TECHNICAL FIELD

[0001] The present disclosure relates to the field of sound coding. More specifically, the present disclosure relates to methods, an encoder and a decoder for linear predictive encoding and decoding of sound signals upon transition between frames having different sampling rates.

BACKGROUND

[0002] The demand for efficient digital wideband speech/audio encoding techniques with a good subjective quality/bit rate trade-off is increasing for numerous applications such as audio/video teleconferencing, multimedia, and wireless applications, as well as Internet and packet network applications. Until recently, telephone bandwidths in the range of 200-3400 Hz were mainly used in speech coding applications. However, there is an increasing demand for wideband speech applications in order to increase the intelligibility and naturalness of the speech signals. A bandwidth in the range 50-7000 Hz was found sufficient for delivering a face-to-face speech quality. For audio signals, this range gives an acceptable audio quality, but is still lower than the CD (Compact Disk) quality which operates in the range 20-20000 Hz.

[0003] A speech encoder converts a speech signal into a digital bit stream that is transmitted over a communication channel (or stored in a storage medium). The speech signal is digitized (sampled and quantized with usually 16-bits per sample) and the speech encoder has the role of representing these digital samples with a smaller number of bits while maintaining a good subjective speech quality. The speech decoder or synthesizer operates on the transmitted or stored bit stream and converts it back to a sound signal.

[0004] One of the best available techniques capable of achieving a good quality/bit rate trade-off is the so-called CELP (Code Excited Linear Prediction) technique. According to this technique, the sampled speech signal is processed in successive blocks of L samples usually called *frames* where L is some predetermined number (corresponding to 10-30 ms of speech). In CELP, an LP (Linear Prediction) synthesis filter is computed and transmitted every frame. The L-sample frame is further divided into smaller blocks called *subframes* of N samples, where $L=kN$ and k is the number of subframes in a frame (N usually corresponds to 4-10 ms of speech). An excitation signal is determined in each subframe, which usually comprises two components: one from the past excitation (also called pitch contribution or adaptive codebook) and the other from an innovative codebook (also called fixed codebook). This excitation signal is transmitted and used at the decoder as the input of the LP synthesis filter in order to obtain the synthesized speech.

[0005] To synthesize speech according to the CELP technique, each block of N samples is synthesized by filtering an appropriate codevector from the innovative codebook through time-varying filters modeling the spectral characteristics of the speech signal. These filters comprise a pitch synthesis filter (usually implemented as an adaptive codebook containing the past excitation signal) and an LP synthesis filter. At the encoder end, the synthesis output is computed for all, or a subset, of the codevectors from the innovative codebook (codebook search). The retained innovative codevector is the one producing the synthesis output closest to the original speech signal according to a perceptually weighted distortion measure. This perceptual weighting is performed using a so-called perceptual weighting filter, which is usually derived from the LP synthesis filter.

[0006] In LP-based coders such as CELP, an LP filter is computed then quantized and transmitted once per frame. However, in order to insure smooth evolution of the LP synthesis filter, the filter parameters are interpolated in each subframe, based on the LP parameters from the past frame. The LP filter parameters are not suitable for quantization due to filter stability issues. Another LP representation more efficient for quantization and interpolation is usually used. A commonly used LP parameter representation is the line spectral frequency (LSF) domain.

[0007] In wideband coding the sound signal is sampled at 16000 samples per second and the encoded bandwidth extended up to 7 kHz. However, at low bit rate wideband coding (below 16 kbit/s) it is usually more efficient to down-sample the input signal to a slightly lower rate, and apply the CELP model to a lower bandwidth, then use bandwidth extension at the decoder to generate the signal up to 7 kHz. This is due to the fact that CELP models lower frequencies with high energy better than higher frequency. So it is more efficient to focus the model on the lower bandwidth at low bit rates. AMR-WB standard (Reference [1]) is such a coding example, where the input signal is down-sampled to 12800 samples per second, and the CELP encodes the signal up to 6.4 kHz. At the decoder bandwidth extension is used to generate a signal from 6.4 to 7 kHz. However, at bit rates higher than 16 kbit/s it is more efficient to use CELP to encode the signal up to 7 kHz, since there are enough bits to represent the entire bandwidth.

[0008] Most recent coders are multi-rate coders covering a wide range of bit rates to enable flexibility in different application scenarios. Again AMR-WB is such an example, where the encoder operates at bit rates from 6.6 to 23.85 kbit/s. In multi-rate coders the codec should be able to switch between different bit rates on a frame basis without introducing switching artefacts. In AMR-WB this is easily achieved since all the rates use CELP at 12.8 kHz internal sampling rate. However, in a recent coder using 12.8 kHz sampling at bit rates below 16 kbit/s and 16 kHz sampling at bit rates higher than 16 kbits/s, the issues related to switching the bit rate between frames using different sampling rates need to be addressed. The main issues are in the LP filter transition, and in the memory of the synthesis filter and adaptive codebook. The patent application US2008/0077401 A1 discloses a method for transcoding a CELP based compressed voice bitstream from source codec to destination codec involving a generic method for converting between LSP coefficients via a linear transform.

[0009] Therefore there remains a need for efficient methods for switching LP-based codecs between two bit rates with different internal sampling rates.

SUMMARY

[0010] According to the present disclosure, there is provided a method implemented in a sound signal encoder or a sound decoder, in accordance with claim 1, for converting linear predictive (LP) filter parameters from a sound signal sampling rate S1 to a sound signal sampling rate S2. A power spectrum of a LP synthesis filter is computed, at the sampling rate S1, using the LP filter parameters. The power spectrum of the LP synthesis filter is modified to convert it from the sampling rate S1 to the sampling rate S2. The modified power spectrum of the LP synthesis filter is inverse transformed to determine autocorrelations of the LP synthesis filter at the sampling rate S2. The autocorrelations are used to compute the LP filter parameters at the sampling rate S2.

[0011] According to the present disclosure, there is also provided a device for use in a sound signal encoder or a sound signal decoder, in accordance with claim 10, for converting linear predictive (LP) filter parameters from a sound signal sampling rate S1 to a sound signal sampling rate S2. The device comprises a processor configured to:

- compute, at the sampling rate S1, a power spectrum of a LP synthesis filter using the received LP filter parameters,
- modify the power spectrum of the LP synthesis filter to convert it from the sampling rate S1 to the sampling rate S2,
- inverse transform the modified power spectrum of the LP synthesis filter to determine autocorrelations of the LP synthesis filter at the sampling rate S2, and
- use the autocorrelations to compute the LP filter parameters at the sampling rate S2.

[0012] A computer-readable non-transitory memory storing code instructions is provided in accordance with claim 17. The foregoing and other objects, advantages and features of the present disclosure will become more apparent upon reading of the following non-restrictive description of an illustrative embodiment thereof, given by way of example only with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] In the appended drawings:

Figure 1 is a schematic block diagram of a sound communication system depicting an example of use of sound encoding and decoding;

Figure 2 is a schematic block diagram illustrating the structure of a CELP-based encoder and decoder, part of the sound communication system of Figure 1 ;

Figure 3 illustrates an example of framing and interpolation of LP parameters;

Figure 4 is a block diagram illustrating an embodiment for converting the LP filter parameters between two different sampling rates; and

Figure 5 is a simplified block diagram of an example configuration of hardware components forming the encoder and/or decoder of Figures 1 and 2.

DETAILED DESCRIPTION

[0014] The non-restrictive illustrative embodiment of the present disclosure is concerned with a method and a device for efficient switching, in an LP-based codec, between frames using different internal sampling rates. The switching method and device can be used with any sound signals, including speech and audio signals. The switching between 16 kHz and 12.8 kHz internal sampling rates is given by way of example, however, the switching method and device can also be applied to other sampling rates.

[0015] Figure 1 is a schematic block diagram of a sound communication system depicting an example of use of sound encoding and decoding. A sound communication system 100 supports transmission and reproduction of a sound signal across a communication channel 101. The communication channel 101 may comprise, for example, a wire, optical or fibre link. Alternatively, the communication channel 101 may comprise at least in part a radio frequency link. The radio frequency link often supports multiple, simultaneous speech communications requiring shared bandwidth resources such as may be found with cellular telephony. Although not shown, the communication channel 101 may be replaced by a storage device in a single device embodiment of the communication system 101 that records and stores the encoded sound signal for later playback.

[0016] Still referring to Figure 1, for example a microphone 102 produces an original analog sound signal 103 that is supplied to an analog-to-digital (A/D) converter 104 for converting it into an original digital sound signal 105. The original digital sound signal 105 may also be recorded and supplied from a storage device (not shown). A sound encoder 106 encodes the original digital sound signal 105 thereby producing a set of encoding parameters 107 that are coded into a binary form and delivered to an optional channel encoder 108. The optional channel encoder 108, when present, adds redundancy to the binary representation of the coding parameters before transmitting them over the communication channel 101. On the receiver side, an optional channel decoder 109 utilizes the above mentioned redundant information in a digital bit stream 111 to detect and correct channel errors that may have occurred during the transmission over the communication channel 101, producing received encoding parameters 112. A sound decoder 110 converts the received encoding parameters 112 for creating a synthesized digital sound signal 113. The synthesized digital sound signal 113 reconstructed in the sound decoder 110 is converted to a synthesized analog sound signal 114 in a digital-to-analog (D/A) converter 115 and played back in a loudspeaker unit 116. Alternatively, the synthesized digital sound signal 113 may also be supplied to and recorded in a storage device (not shown).

[0017] Figure 2 is a schematic block diagram illustrating the structure of a CELP-based encoder and decoder, part of the sound communication system of Figure 1. As illustrated in Figure 2, a sound codec comprises two basic parts: the sound encoder 106 and the sound decoder 110 both introduced in the foregoing description of Figure 1. The encoder 106 is supplied with the original digital sound signal 105, determines the encoding parameters 107, described herein below, representing the original analog sound signal 103. These parameters 107 are encoded into the digital bit stream 111 that is transmitted using a communication channel, for example the communication channel 101 of Figure 1, to the decoder 110. The sound decoder 110 reconstructs the synthesized digital sound signal 113 to be as similar as possible to the original digital sound signal 105.

[0018] Presently, the most widespread speech coding techniques are based on Linear Prediction (LP), in particular CELP. In LP-based coding, the synthesized digital sound signal 113 is produced by filtering an excitation 214 through a LP synthesis filter 216 having a transfer function $1/A(z)$. In CELP, the excitation 214 is typically composed of two parts: a first-stage, adaptive-codebook contribution 222 selected from an adaptive codebook 218 and amplified by an adaptive-codebook gain g_p 226 and a second-stage, fixed-codebook contribution 224 selected from a fixed codebook 220 and amplified by a fixed-codebook gain g_c 228. Generally speaking, the adaptive codebook contribution 222 models the periodic part of the excitation and the fixed codebook contribution 214 is added to model the evolution of the sound signal.

[0019] The sound signal is processed by frames of typically 20 ms and the LP filter parameters are transmitted once per frame. In CELP, the frame is further divided in several subframes to encode the excitation. The subframe length is typically 5 ms.

[0020] CELP uses a principle called Analysis-by-Synthesis where possible decoder outputs are tried (synthesized) already during the coding process at the encoder 106 and then compared to the original digital sound signal 105. The encoder 106 thus includes elements similar to those of the decoder 110. These elements includes an adaptive codebook contribution 250 selected from an adaptive codebook 242 that supplies a past excitation signal $v(n)$ convolved with the impulse response of a weighted synthesis filter $H(z)$ (see 238) (cascade of the LP synthesis filter $1/A(z)$ and the perceptual weighting filter $W(z)$), the result $y_1(n)$ of which is amplified by an adaptive-codebook gain g_p 240. Also included is a fixed codebook contribution 252 selected from a fixed codebook 244 that supplies an innovative codevector $c_k(n)$ convolved with the impulse response of the weighted synthesis filter $H(z)$ (see 246), the result $y_2(n)$ of which is amplified by a fixed codebook gain g_c 248.

[0021] The encoder 106 also comprises a perceptual weighting filter $W(z)$ 233 and a provider 234 of a zero-input response of the cascade ($H(z)$) of the LP synthesis filter $1/A(z)$ and the perceptual weighting filter $W(z)$. Subtractors 236, 254 and 256 respectively subtract the zero-input response, the adaptive codebook contribution 250 and the fixed code-book contribution 252 from the original digital sound signal 105 filtered by the perceptual weighting filter 233 to provide a mean-squared error 232 between the original digital sound signal 105 and the synthesized digital sound signal 113.

[0022] The codebook search minimizes the mean-squared error 232 between the original digital sound signal 105 and the synthesized digital sound signal 113 in a perceptually weighted domain, where discrete time index $n = 0, 1, ..., N-1$, and N is the length of the subframe. The perceptual weighting filter $W(z)$ exploits the frequency masking effect and typically is derived from a LP filter $A(z)$.

[0023] An example of the perceptual weighting filter $W(z)$ for WB (wideband, bandwidth of 50 - 7000 Hz) signals can be found in Reference [1].

[0024] Since the memory of the LP synthesis filter $1/A(z)$ and the weighting filter $W(z)$ is independent from the searched codevectors, this memory can be subtracted from the original digital sound signal 105 prior to the fixed codebook search. Filtering of the candidate codevectors can then be done by means of a convolution with the impulse response of the cascade of the filters $1/A(z)$ and $W(z)$, represented by $H(z)$ in Figure 2.

[0025] The digital bit stream 111 transmitted from the encoder 106 to the decoder 110 contains typically the following parameters 107: quantized parameters of the LP filter $A(z)$, indices of the adaptive codebook 242 and of the fixed codebook 244, and the gains g_p 240 and g_c 248 of the adaptive codebook 242 and of the fixed codebook 244.

Converting LP filter parameters when switching at frame boundaries with different sampling rates

[0026] In LP-based coding the LP filter $A(z)$ is determined once per frame, and then interpolated for each subframe. Figure 3 illustrates an example of framing and interpolation of LP parameters. In this example, a present frame is divided into four subframes SF1, SF2, SF3 and SF4, and the LP analysis window is centered at the last subframe SF4. Thus the LP parameters resulting from LP analysis in the present frame, F1, are used as is in the last subframe, that is SF4 = F1. For the first three subframes SF1, SF2 and SF3, the LP parameters are obtained by interpolating the parameters in the present frame, F1, and a previous frame, F0. That is:

$$SF1 = 0.75\ F0 + 0.25\ F1;$$

$$SF2 = 0.5\ F0 + 0.5\ F1;$$

$$SF3 = 0.25\ F0 + 0.75\ F1$$

$$SF4 = F1.$$

[0027] Other interpolation examples may alternatively be used depending on the LP analysis window shape, length and position. In another embodiment, the coder switches between 12.8 kHz and 16 kHz internal sampling rates, where 4 subframes per frame are used at 12.8 kHz and 5 subframes per frame are used at 16 kHz, and where the LP parameters are also quantized in the middle of the present frame (Fm). In this other embodiment, LP parameter interpolation for a 12.8 kHz frame is given by:

$$SF1 = 0.5\ F0 + 0.5\ Fm;$$

$$SF2 = Fm;$$

$$SF3 = 0.5\ Fm + 0.5\ F1;$$

$$SF4 = F1.$$

[0028] For a 16 kHz sampling, the interpolation is given by:

$$SF1 = 0.55\ F0 + 0.45\ Fm;$$

$$SF2 = 0.15\ F0 + 0.85\ Fm;$$

$$SF3 = 0.75\ Fm + 0.25\ F1;$$

$$SF4 = 0.35\ Fm + 0.65\ F1;$$

$$SF5 = F1.$$

[0029] LP analysis results in computing the parameters of the LP synthesis filter using:

$$\frac{1}{A(z)} = \frac{1}{1+\sum_{i=1}^{M} a_i z^{-i}} = \frac{1}{1+a_1 z^{-1} + a_2 z^{-2} + \ldots + a_M z^{-M}} \qquad (1)$$

where a_i, $i = 1,...,M$, are LP filter parameters and M is the filter order.

[0030] The LP filter parameters are transformed to another domain for quantization and interpolation purposes. Other LP parameter representations commonly used are reflection coefficients, log-area ratios, immitance spectrum pairs (used in AMR-WB; Reference [1]), and line spectrum pairs, which are also called line spectrum frequencies (LSF). In this illustrative embodiment, the line spectrum frequency representation is used. An example of a method that can be used to convert the LP parameters to LSF parameters and vice versa can be found in Reference [2]. The interpolation example in the previous paragraph is applied to the LSF parameters, which can be in the frequency domain in the range between 0 and Fs/2 (where Fs is the sampling frequency), or in the scaled frequency domain between 0 and π, or in the cosine domain (cosine of scaled frequency).

[0031] As described above, different internal sampling rates may be used at different bit rates to improve quality in multi-rate LP-based coding. In this illustrative embodiment, a multi-rate CELP wideband coder is used where an internal sampling rate of 12.8 kHz is used at lower bit rates and an internal sampling rate of 16 kHz at higher bit rates. At a 12.8 kHz sampling rate, the LSFs cover the bandwidth from 0 to 6.4 kHz, while at a 16 kHz sampling rate they cover the range from 0 to 8 kHz. When switching the bit rate between two frames where the internal sampling rate is different, some issues are addressed to insure seamless switching. These issues include the interpolation of LP filter parameters and the memories of the synthesis filter and the adaptive codebook, which are at different sampling rates.

[0032] The present disclosure introduces a method for efficient interpolation of LP parameters between two frames at different internal sampling rates. By way of example, the switching between 12.8 kHz and 16 kHz sampling rates is considered. The disclosed techniques are however not limited to these particular sampling rates and may apply to other internal sampling rates.

[0033] Let's assume that the encoder is switching from a frame F1 with internal sampling rate S1 to a frame F2 with internal sampling rate S2. The LP parameters in the first frame are denoted $LSF1_{S1}$ and the LP parameters at the second frame are denoted $LSF2_{S2}$. In order to update the LP parameters in each subframe of frame F2, the LP parameters LSF1 and LSF2 are interpolated. In order to perform the interpolation, the filters have to be set at the same sampling rate. This requires performing LP analysis of frame F1 at sampling rate S2. To avoid transmitting the LP filter twice at the two sampling rates in frame F1, the LP analysis at sampling rate S2 can be performed on the past synthesis signal which is available at both encoder and decoder. This approach involves re-sampling the past synthesis signal from rate S1 to rate S2, and performing complete LP analysis, this operation being repeated at the decoder, which is usually computationally demanding.

[0034] Alternative method and devices are disclosed herein for converting LP synthesis filter parameters LSF1 from sampling rate S1 to sampling rate S2 without the need to re-sample the past synthesis and perform complete LP analysis. The method, used at encoding and/or at decoding, comprises computing the power spectrum of the LP synthesis filter at rate S1; modifying the power spectrum to convert it from rate S1 to rate S2; converting the modified power spectrum back to the time domain to obtain the filter autocorrelation at rate S2; and finally use the autocorrelation to compute LP filter parameters at rate S2.

[0035] In at least some embodiments, modifying the power spectrum to convert it from rate S1 to rate S2 comprises the following operations:

[0036] If S1 is larger than S2, modifying the power spectrum comprises truncating the K-sample power spectrum down to K(S2/S1) samples, that is, removing K(S1-S2)/S1 samples.

[0037] On the other hand, if S1 is smaller than S2, then modifying the power spectrum comprises extending the K-sample power spectrum up to K(S2/S1) samples, that is, adding K(S2-S1)/S1 samples.

[0038] Computing the LP filter at rate S2 from the autocorrelations can be done using the Levinson-Durbin algorithm (see Reference [1]). Once the LP filter is converted to rate S2, the LP filter parameters are transformed to the interpolation domain, which is an LSF domain in this illustrative embodiment.

[0039] The procedure described above is summarized in Figure 4, which is a block diagram illustrating an embodiment for converting the LP filter parameters between two different sampling rates.

[0040] Sequence 300 of operations shows that a simple method for the computation of the power spectrum of the LP synthesis filter 1/A(z) is to evaluate the frequency response of the filter at K frequencies from 0 to 2π.

[0041] The frequency response of the synthesis filter is given by

$$\frac{1}{A(\omega)} = \frac{1}{1+\sum_{i=1}^{M} a_i e^{-j\omega i}} = \frac{1}{1+\sum_{i=1}^{M} a_i \cos(\omega i) + j\sum_{i=1}^{M} a_i \sin(\omega i)} \qquad (2)$$

and the power spectrum of the synthesis filter is calculated as an energy of the frequency response of the synthesis filter, given by

$$P(\omega) = \frac{1}{\left|A(\omega)\right|^2} = \frac{1}{\left(1+\sum_{i=1}^{M} a_i \cos(\omega i)\right)^2 + \left(\sum_{i=1}^{M} a_i \sin(\omega i)\right)^2} \qquad (3)$$

[0042] Initially, the LP filter is at a rate equal to S1 (operation 310). A K-sample (i.e. discrete) power spectrum of the LP synthesis filter is computed (operation 320) by sampling the frequency range from 0 to 2π. That is

$$P(k) = \frac{1}{\left(1+\sum_{i=1}^{M} a_i \cos(\frac{2\pi ik}{K})\right)^2 + \left(\sum_{i=1}^{M} a_i \sin(\frac{2\pi ik}{K})\right)^2}, \qquad k = 0,\ldots,K-1 \qquad (4)$$

[0043] Note that it is possible to reduce operational complexity by computing $P(k)$ only for $k = 0,\ldots,K/2$ since the power spectrum from π to 2π is a mirror of that from 0 to π.

[0044] A test (operation 330) determines which of the following cases apply. In a first case, the sampling rate S1 is larger than the sampling rate S2, and the power spectrum for frame F1 is truncated (operation 340) such that the new number of samples is $K(S2/S1)$.

[0045] In more details, when S1 is larger than S2, the length of the truncated power spectrum is $K_2 = K(S2/S1)$ samples. Since the power spectrum is truncated, it is computed from $k = 0,\ldots,K_2/2$. Since the power spectrum is symmetric around $K_2/2$, then it is assumed that

$$P(K_2/2+k) = P(K_2/2-k), \text{ from } k = 1,\ldots,K_2/2-1$$

[0046] The Fourier Transform of the autocorrelations of a signal gives the power spectrum of that signal. Thus, applying inverse Fourier Transform to the truncated power spectrum results in the autocorrelations of the impulse response of the synthesis filter at sampling rate S2.

[0047] The Inverse Discrete Fourier Transform (IDFT) of the truncated power spectrum is given by

$$R(i) = \frac{1}{K_2}\sum_{k=0}^{K_2-1} P(k) e^{j2\pi ik/K_2} \qquad (5)$$

[0048] Since the filter order is M, then the IDFT may be computed only for $i = 0,\ldots,M$. Further, since the power spectrum is real and symmetric, then the IDFT of the power spectrum is also real and symmetric. Given the symmetry of the power spectrum, and that only M+1 correlations are needed, the inverse transform of the power spectrum can be given as

$$R(i)=\frac{1}{K_2}\left(P(0)+(-1)^i P(K_2/2)+2(-1)^i\sum_{k=1}^{K_2/2-1}P(K_2/2-k)\cos(2\pi ik/K_2)\right) \quad (6)$$

[0049] That is

$$R(0)=\frac{1}{K_2}\left(P(0)+P(K_2/2)+2\sum_{k=1}^{K_2/2-1}P(k)\right) \quad (7)$$

$$R(i)=\frac{1}{K_2}\left(P(0)-P(K_2/2)-2\sum_{k=1}^{K_2/2-1}P(K_2/2-k)\cos(2\pi ik/K_2)\right) \text{ for } i=1,3,...,M-1$$

$$R(i)=\frac{1}{K_2}\left(P(0)+P(K_2/2)+2\sum_{k=1}^{K_2/2-1}P(K_2/2-k)\cos(2\pi ik/K_2)\right) \text{ for } i=2,4,...,M$$

[0050] After the autocorrelations are computed at sampling rate S2, Levinson-Durbin algorithm (see Reference [1]) can be used to compute the parameters of the LP filter at sampling rate S2. Then, the LP filter parameters are transformed to the LSF domain for interpolation with the LSFs of frame F2 in order to obtain LP parameters at each subframe.

[0051] In the illustrative example where the coder encodes a wideband signal and is switching from a frame with an internal sampling rate S1=16 kHz to a frame with internal sampling rate S2=12.8 kHz, assuming that K = 100, the length of the truncated power spectrum is K_2 =100(12800/16000) = 80 samples. The power spectrum is computed for 41 samples using Equation (4), and then the autocorrelations are computed using Equation (7) with K_2 = 80.

[0052] In a second case, when the test (operation 330) determines that S1 is smaller than S2, the length of the extended power spectrum is $K_2 = K(S2/S1)$ samples (operation 350). After computing the power spectrum from k = 0,...,K/2, the power spectrum is extended to $K_2/2$. Since there is no original spectral content between $K/2$ and $K_2/2$, extending the power spectrum can be done by inserting a number of samples up to $K_2/2$ using very low sample values. A simple approach is to repeat the sample at $K/2$ up to $K_2/2$. Since the power spectrum is symmetric around $K_2/2$ then it is assumed that

$$P(K_2/2+k)=P(K_2/2-k), \text{ from } k=1,...,K_2/2-1$$

[0053] In either cases, the inverse DFT is then computed as in Equation (6) to obtain the autocorrelations at sampling rate S2 (operation 360) and the Levinson-Durbin algorithm (see Reference [1]) is used to compute the LP filter parameters at sampling rate S2 (operation 370). Then filter parameters are transformed to the LSF domain for interpolation with the LSFs of frame F2 in order to obtain LP parameters at each subframe.

[0054] Again, let's take the illustrative example where the coder is switching from a frame with an internal sampling rate S1=12.8 kHz to a frame with internal sampling rate S2=16 kHz, and let's assume that K = 80. The length of the extended power spectrum is K_2 = 80(16000/12800) = 100 samples. The power spectrum is computed for 51 samples using Equation (4), and then the autocorrelations are computed using Equation (7) with K_2 =100.

[0055] Note that other methods can be used to compute the power spectrum of the LP synthesis filter or the inverse DFT of the power spectrum without departing from the spirit of the present disclosure.

[0056] Note that in this illustrative embodiment converting the LP filter parameters between different internal sampling rates is applied to the quantized LP parameters, in order to determine the interpolated synthesis filter parameters in each subframe, and this is repeated at the decoder. It is noted that the weighting filter uses unquantized LP filter parameters, but it was found sufficient to interpolate between the unquantized filter parameters in new frame F2 and sampling-converted quantized LP parameters from past frame F1 in order to determine the parameters of the weighting filter in each subframe. This avoids the need to apply LP filter sampling conversion on the unquantized LP filter parameters as well.

Other considerations when switching at frame boundaries with different sampling rates

[0057] Another issue to be considered when switching between frames with different internal sampling rates is the content of the adaptive codebook, which usually contains the past excitation signal. If the new frame has an internal sampling rate S2 and the previous frame has an internal sampling rate S1, then the content of the adaptive codebook is re-sampled from rate S1 to rate S2, and this is performed at both the encoder and the decoder.

[0058] In order to reduce the complexity, in this disclosure, the new frame F2 is forced to use a transient encoding mode which is independent of the past excitation history and thus does not use the history of the adaptive codebook. An example of transient mode encoding can be found in PCT patent application WO 2008/049221 A1 "Method and device for coding transition frames in speech signals", the disclosure of which is incorporated by reference herein.

[0059] Another consideration when switching at frame boundaries with different sampling rates is the memory of the predictive quantizers. As an example, LP-parameter quantizers usually use predictive quantization, which may not work properly when the parameters are at different sampling rates. In order to reduce switching artefacts, the LP-parameter quantizer may be forced into a non-predictive coding mode when switching between different sampling rates.

[0060] A further consideration is the memory of the synthesis filter, which may be resampled when switching between frames with different sampling rates.

[0061] Finally, the additional complexity that arises from converting LP filter parameters when switching between frames with different internal sampling rates may be compensated by modifying parts of the encoding or decoding processing. For example, in order not to increase the encoder complexity, the fixed codebook search may be modified by lowering the number of iterations in the first subframe of the frame (see Reference [1] for an example of fixed codebook search).

[0062] Additionally, in order not to increase the decoder complexity, certain post-processing can be skipped. For example, in this illustrative embodiment, a post-processing technique as described in US patent 7,529,660 "Method and device for frequency-selective pitch enhancement of synthesized speech", the disclosure of which is incorporated by reference herein, may be used. This post-filtering is skipped in the first frame after switching to a different internal sampling rate (skipping this post-filtering also overcomes the need of past synthesis utilized in the post-filter).

[0063] Further, other parameters that depend on the sampling rate may be scaled accordingly. For example, the past pitch delay used for decoder classifier and frame erasure concealment may be scaled by the factor S2/S1.

[0064] Figure 5 is a simplified block diagram of an example configuration of hardware components forming the encoder and/or decoder of Figures 1 and 2. A device 400 may be implemented as a part of a mobile terminal, as a part of a portable media player, a base station, Internet equipment or in any similar device, and may incorporate the encoder 106, the decoder 110, or both the encoder 106 and the decoder 110. The device 400 includes a processor 406 and a memory 408. The processor 406 may comprise one or more distinct processors for executing code instructions to perform the operations of Figure 4. The processor 406 may embody various elements of the encoder 106 and of the decoder 110 of Figures 1 and 2. The processor 406 may further execute tasks of a mobile terminal, of a portable media player, base station, Internet equipement and the like. The memory 408 is operatively connected to the processor 406. The memory 408, which may be a non-transitory memory, stores the code instructions executable by the processor 406.

[0065] An audio input 402 is present in the device 400 when used as an encoder 106. The audio input 402 may include for example a microphone or an interface connectable to a microphone. The audio input 402 may include the microphone 102 and the A/D converter 104 and produce the original analog sound signal 103 and/or the original digital sound signal 105. Alternatively, the audio input 402 may receive the original digital sound signal 105. Likewise, an encoded output 404 is present when the device 400 is used as an encoder 106 and is configured to forward the encoding parameters 107 or the digital bit stream 111 containing the parameters 107, including the LP filter parameters, to a remote decoder via a communication link, for example via the communication channel 101, or toward a further memory (not shown) for storage. Non-limiting implementation examples of the encoded output 404 comprise a radio interface of a mobile terminal, a physical interface such as for example a universal serial bus (USB) port of a portable media player, and the like.

[0066] An encoded input 403 and an audio output 405 are both present in the device 400 when used as a decoder 110. The encoded input 403 may be constructed to receive the encoding parameters 107 or the digital bit stream 111 containing the parameters 107, including the LP filter parameters from an encoded output 404 of an encoder 106. When the device 400 includes both the encoder 106 and the decoder 110, the encoded output 404 and the encoded input 403 may form a common communication module. The audio output 405 may comprise the D/A converter 115 and the loudspeaker unit 116. Alternatively, the audio output 405 may comprise an interface connectable to an audio player, to a loudspeaker, to a recording device, and the like.

[0067] The audio input 402 or the encoded input 403 may also receive signals from a storage device (not shown). In the same manner, the encoded output 404 and the audio output 405 may supply the output signal to a storage device (not shown) for recording.

[0068] The audio input 402, the encoded input 403, the encoded output 404 and the audio output 405 are all operatively connected to the processor 406.

[0069] Those of ordinary skill in the art will realize that the description of the methods, encoder and decoder for linear predictive encoding and decoding of sound signals are illustrative only and are not intended to be in any way limiting. Other embodiments will readily suggest themselves to such persons with ordinary skill in the art having the benefit of the present disclosure. Furthermore, the disclosed methods, encoder and decoder may be customized to offer valuable solutions to existing needs and problems of switching linear prediction based codecs between two bit rates with different sampling rates.

[0070] In the interest of clarity, not all of the routine features of the implementations of methods, encoder and decoder are shown and described. It will, of course, be appreciated that in the development of any such actual implementation of the methods, encoder and decoder, numerous implementation-specific decisions may need to be made in order to achieve the developer's specific goals, such as compliance with application-, system-, network- and business-related constraints, and that these specific goals will vary from one implementation to another and from one developer to another. Moreover, it will be appreciated that a development effort might be complex and time-consuming, but would nevertheless be a routine undertaking of engineering for those of ordinary skill in the field of sound coding having the benefit of the present disclosure.

[0071] In accordance with the present disclosure, the components, process operations, and/or data structures described herein may be implemented using various types of operating systems, computing platforms, network devices, computer programs, and/or general purpose machines. In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable gate arrays (FPGAs), application specific integrated circuits (ASICs), or the like, may also be used. Where a method comprising a series of operations is implemented by a computer or a machine and those operations may be stored as a series of instructions readable by the machine, they may be stored on a tangible medium.

[0072] Systems and modules described herein may comprise software, firmware, hardware, or any combination(s) of software, firmware, or hardware suitable for the purposes described herein.

[0073] Although the present disclosure has been described hereinabove by way of non-restrictive, illustrative embodiments thereof, these embodiments may be modified at will within the scope of the appended claims.

REFERENCES

[0074] The following references are incorporated by reference herein.

[1] 3GPP Technical Specification 26.190, "Adaptive Multi-Rate-Wideband (AMR-WB) speech codec; Transcoding functions," July 2005; http://www.3gpp.org.

[2] ITU-T Recommendation G.729 "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)", 01/2007.

Claims

1. A method implemented in a sound signal encoder or a sound signal decoder for converting linear predictive (LP) filter parameters from a sound signal sampling rate S1 to a sound signal sampling rate S2, the method being **characterised by**:

computing, at the sampling rate S1, a power spectrum of a LP synthesis filter using the LP filter parameters;
modifying the power spectrum of the LP synthesis filter to convert it from the sampling rate S1 to the sampling rate S2;
inverse transforming the modified power spectrum of the LP synthesis filter to determine autocorrelations of the LP synthesis filter at the sampling rate S2; and
using the autocorrelations to compute the LP filter parameters at the sampling rate S2.

2. A method as recited in claim 1, wherein modifying the power spectrum of the LP synthesis filter to convert it from the sampling rate S1 to the sampling rate S2 comprises:

if S1 is less than S2, extending the power spectrum of the LP synthesis filter based on a ratio between S1 and S2;
if S1 is larger than S2, truncating the power spectrum of the LP synthesis filter based on the ratio between S1 and S2.

3. A method as recited in any one of claims 1 and 2, wherein the conversion of the LP filter parameters occurs when an encoder switches from a frame with the sampling rate S1 to a frame with the sampling rate S2.

4. A method as recited in claim 3, comprising, when implemented in a sound signal encoder, computing LP filter parameters in each subframe of a current frame by interpolating LP filter parameters of the current frame at the sampling rate S2 with LP filter parameters of a past frame converted from the sampling rate S1 to the sampling rate S2.

5. A method as recited in claim 4, comprising, when implemented in a sound signal encoder, forcing the current frame to an encoding mode that does not use a history of an adaptive codebook.

6. A method as recited in claim 4 or 5, comprising, when implemented in a sound signal encoder, forcing a LP parameter quantizer to use a non-predictive quantization method in the current frame.

7. A method as recited in any one of claims 1 to 6, comprising:

   computing the power spectrum of the LP synthesis filter at K samples;
   extending the power spectrum of the LP synthesis filter to K(S2/S1) samples when the sampling rate S1 is less than the sampling rate S2; and
   truncating the power spectrum of the LP synthesis filter to K(S2/S1) samples when the sampling rate S1 is greater than the sampling rate S2.

8. A method as recited in any one of claims 1 to 7, comprising computing the power spectrum of the LP synthesis filter as an energy of a frequency response of the LP synthesis filter.

9. A method as recited in claim 3, comprising, when implemented in a sound signal decoder, computing LP filter parameters in each subframe of a new frame by interpolating LP filter parameters of a current frame at the sampling rate S2 with LP filter parameters of a past frame converted from the sampling rate S1 to the sampling rate S2.

10. A device for use in a sound signal encoder or a sound signal decoder for converting linear predictive (LP) filter parameters from a sound signal sampling rate S1 to a sound signal sampling rate S2, the device being **characterised in that** it comprises:

   a processor configured to:

      compute, at the sampling rate S1, a power spectrum of a LP synthesis filter using the LP filter parameters,
      modify the power spectrum of the LP synthesis filter to convert it from the sampling rate S1 to the sampling rate S2,
      inverse transform the modified power spectrum of the LP synthesis filter to determine autocorrelations of the LP synthesis filter at the sampling rate S2, and
      use the autocorrelations to compute the LP filter parameters at the sampling rate S2.

11. A device as recited in claim 10, wherein the processor is configured to:

   extend the power spectrum of the LP synthesis filter based on a ratio between S1 and S2 if S1 is less than S2; and
   truncate the power spectrum of the LP synthesis filter based on the ratio between S1 and S2 if S1 is larger than S2.

12. A device as recited in any one of claims 10 and 11, wherein the processor is configured to compute LP filter parameters in each subframe of a current frame by interpolating LP filter parameters of the current frame at the sampling rate S2 with LP filter parameters of a past frame converted from the sampling rate S1 to the sampling rate S2.

13. A device as recited in any one of claims 10 to 12, wherein the processor is configured to:

   computing the power spectrum of the LP synthesis filter at K samples;
   extend the power spectrum of the LP synthesis filter to K(S2/S1) samples when the sampling rate S1 is less than the sampling rate S2; and
   truncate the power spectrum of the LP synthesis filter to K(S2/S1) samples when the sampling rate S1 is greater than the sampling rate S2.

14. A device as recited in any one of claims 10 to 13, wherein the processor is configured to compute the power spectrum of the LP synthesis filter as an energy of a frequency response of the LP synthesis filter.

15. A device as recited in any one of claims 10 to 14, wherein the processor is configured to inverse transform the modified power spectrum of the LP synthesis filter by using an inverse discrete Fourier Transform.

16. A device as recited in any one of claims 10 to 15, further comprising a non-transitory memory storing code instructions

executable by the processor to perform the computing, modifying, inverse transforming, and using operations.

17. A computer-readable non-transitory memory storing code instructions for performing, when running on the processor of any one of claims 10 to 16, a method as recited in any one of claims 1 to 9.

Patentansprüche

1. Verfahren, das in einem Schallsignal-Codierer oder einem Schallsignal-Decodierer implementiert ist, zum Umwandeln von linear-prädiktiven (LP) Filterparametern von einer Schallsignal-Abtastrate S1 in eine Schallsignal-Abtastrate S2, wobei das Verfahren **gekennzeichnet ist durch**:

   Berechnen, bei der Abtastrate S1, eines Leistungsspektrums eines LP-Synthesefilters unter Verwendung der LP-Filterparameter;
   Modifizieren des Leistungsspektrums des LP-Synthesefilters, um dasselbe von der Abtastrate S1 in die Abtastrate S2 umzuwandeln;
   inverses Transformieren des modifizierten Leistungsspektrums des LP-Synthesefilters, um Autokorrelationen des LP-Synthesefilters bei der Abtastrate S2 zu bestimmen; und
   Verwenden der Autokorrelationen, um die LP-Filterparameter bei der Abtastrate S2 zu berechnen.

2. Verfahren nach Anspruch 1, wobei das Modifizieren des Leistungsspektrums des LP-Synthesefilters, um dasselbe von der Abtastrate S1 in die Abtastrate S2 umzuwandeln, umfasst:

   wenn S1 kleiner ist als S2, Erweitern des Leistungsspektrums des LP-Synthesefilters basierend auf einem Verhältnis zwischen S1 und S2;
   wenn S1 größer ist als S2, Trunkieren des Leistungsspektrums des LP-Synthesefilters basierend auf dem Verhältnis zwischen S1 und S2.

3. Verfahren nach einem der Ansprüche 1 und 2, wobei die Umwandlung der LP-Filterparameter erfolgt, wenn ein Codierer von einem Frame mit der Abtastrate S1 auf einen Frame mit der Abtastrate S2 umschaltet.

4. Verfahren nach Anspruch 3, das, wenn es in einem Schallsignal-Codierer implementiert ist, das Berechnen von LP-Filterparametern in jedem Subframe eines aktuellen Frames durch Interpolieren von LP-Filterparametern des aktuellen Frames bei der Abtastrate S2 mit LP-Filterparametern eines früheren Frames umfasst, der von der Abtastrate S1 in die Abtastrate S2 umgewandelt wurde.

5. Verfahren nach Anspruch 4, das, wenn es in einem Schallsignal-Codierer implementiert ist, das Zwingen des aktuellen Frames zu einem Codiermodus umfasst, der keine Historie eines adaptiven Codebuchs verwendet.

6. Verfahren nach Anspruch 4 oder 5, das, wenn es in einem Schallsignal-Codierer implementiert ist, das Zwingen eines LP-Parameterquantisierers dazu umfasst, ein nicht prädiktives Quantisierungsverfahren im aktuellen Frame zu verwenden.

7. Verfahren nach einem der Ansprüche 1 bis 6, umfassend:

   Berechnen des Leistungsspektrums des LP-Synthesefilters bei K Abtastungen;
   Erweitern des Leistungsspektrums des LP-Synthesefilters auf K(S2/S1) Abtastungen, wenn die Abtastrate S1 kleiner ist als die Abtastrate S2; und
   Trunkieren des Leistungsspektrums des LP-Synthesefilters auf K(S2/S1) Abtastungen, wenn die Abtastrate S1 größer ist als die Abtastrate S2.

8. Verfahren nach einem der Ansprüche 1 bis 7, das das Berechnen des Leistungsspektrums des LP-Synthesefilters als eine Energie eines Frequenzgangs des LP-Synthesefilters umfasst.

9. Verfahren nach Anspruch 3, das, wenn es in einem Schallsignal-Decodierer implementiert ist, das Berechnen von LP-Filterparametern in jedem Subframe eines neuen Frames durch Interpolieren von LP-Filterparametern eines aktuellen Frames bei der Abtastrate S2 mit LP-Filterparametern eines früheren Frames umfasst, der von der Abtastrate S1 in die Abtastrate S2 umgewandelt wurde.

10. Vorrichtung zur Verwendung in einem Schallsignal-Codierer oder einem Schallsignal-Decodierer zum Umwandeln von linear-prädiktiven (LP) Filterparametern von einer Schallsignal-Abtastrate S1 in eine Schallsignal-Abtastrate S2, wobei die Vorrichtung **dadurch gekennzeichnet ist, dass** sie umfasst:

einen Prozessor, der dazu konfiguriert ist:

bei der Abtastrate S1 ein Leistungsspektrum eines LP-Synthesefilters unter Verwendung der LP-Filterparameter zu berechnen,
das Leistungsspektrum des LP-Synthesefilters zu modifizieren, um dasselbe von der Abtastrate S1 in die Abtastrate S2 umzuwandeln,
das modifizierte Leistungsspektrum des LP-Synthesefilters invers zu transformieren, um Autokorrelationen des LP-Synthesefilters bei der Abtastrate S2 zu bestimmen, und
die Autokorrelationen zu verwenden, um die LP-Filterparameter bei der Abtastrate S2 zu berechnen.

11. Vorrichtung nach Anspruch 10, wobei der Prozessor dazu konfiguriert ist:

das Leistungsspektrum des LP-Synthesefilters basierend auf einem Verhältnis zwischen S1 und S2 zu erweitern, wenn S1 kleiner ist als S2; und
das Leistungsspektrum des LP-Synthesefilters basierend auf dem Verhältnis zwischen S1 und S2 zu trunkieren, wenn S1 größer ist als S2.

12. Vorrichtung nach einem der Ansprüche 10 und 11, wobei der Prozessor dazu konfiguriert ist, LP-Filterparameter in jedem Subframe eines aktuellen Frames durch Interpolieren von LP-Filterparametern des aktuellen Frames bei der Abtastrate S2 mit LP-Filterparametern eines früheren Frames zu berechnen, der von der Abtastrate S1 in die Abtastrate S2 umgewandelt wurde.

13. Vorrichtung nach einem der Ansprüche 10 bis 12, wobei der Prozessor dazu konfiguriert ist:

das Leistungsspektrum des LP-Synthesefilters bei K Abtastungen zu berechnen;
das Leistungsspektrum des LP-Synthesefilters auf K(S2/S1) Abtastungen zu erweitern, wenn die Abtastrate S1 kleiner ist als die Abtastrate S2; und
das Leistungsspektrum des LP-Synthesefilters auf K(S2/S1) Abtastungen zu trunkieren, wenn die Abtastrate S1 größer ist als die Abtastrate S2.

14. Vorrichtung nach einem der Ansprüche 10 bis 13, wobei der Prozessor dazu konfiguriert ist, das Leistungsspektrum des LP-Synthesefilters als eine Energie eines Frequenzgangs des LP-Synthesefilters zu berechnen.

15. Vorrichtung nach einem der Ansprüche 10 bis 14, wobei der Prozessor dazu konfiguriert ist, das modifizierte Leistungsspektrums des LP-Synthesefilters durch Verwenden einer inversen diskreten Fourier-Transformation invers zu transformieren.

16. Vorrichtung nach einem der Ansprüche 10 bis 15, weiter einen nicht transitorischen Speicher umfassend, welcher Codeanweisungen speichert, die vom Prozessor ausführbar sind, um die Vorgänge des Berechnens, Modifizierens, inversen Transformierens und Verwendens durchzuführen.

17. Computerlesbarer nicht transitorischer Speicher, der Codeanweisungen speichert zum Durchführen eines Verfahrens nach einem der Ansprüche 1 bis 9, wenn dieselben auf dem Prozessor nach einem der Ansprüche 10 bis 16 ausgeführt werden.

Revendications

1. Procédé mis en oeuvre dans un codeur de signal sonore ou dans un décodeur de signal sonore pour convertir des paramètres de filtre prédictif linéaire (LP) d'un taux d'échantillonnage de signal sonore S1 à un taux d'échantillonnage de signal sonore S2, le procédé étant **caractérisé par** :

le calcul, au taux d'échantillonnage S1, d'un spectre de puissance d'un filtre de synthèse LP en utilisant les paramètres de filtre LP ;

la modification du spectre de puissance du filtre de synthèse LP pour le convertir du taux d'échantillonnage S1 au taux d'échantillonnage S2 ;
la transformation inverse du spectre de puissance modifié du filtre de synthèse LP pour déterminer des autocorrélations du filtre de synthèse LP au taux d'échantillonnage S2 ; et
l'utilisation des autocorrélations pour calculer les paramètres de filtre LP au taux d'échantillonnage S2.

2. Procédé selon la revendication 1, dans lequel la modification du spectre de puissance du filtre de synthèse LP pour le convertir du taux d'échantillonnage S1 au taux d'échantillonnage S2 comprend :

si S1 est inférieur à S2, l'extension du spectre de puissance du filtre de synthèse LP sur la base d'un rapport entre S1 et S2 ;
si S1 est supérieur à S2, le fait de tronquer le spectre de puissance du filtre de synthèse LP sur la base du rapport entre S1 et S2.

3. Procédé selon l'une quelconque des revendications 1 et 2, dans lequel la conversion des paramètres de filtre LP se déroule lorsqu'un codeur commute d'une trame avec le taux d'échantillonnage S1 à une trame avec le taux d'échantillonnage S2.

4. Procédé selon la revendication 3, comprenant, lorsqu'il est mis en oeuvre dans un codeur de signal sonore, le calcul de paramètres de filtre LP dans chaque sous-trame d'une trame actuelle par l'interpolation de paramètres de filtre LP de la trame actuelle au taux d'échantillonnage S2 avec des paramètres de filtre LP d'une trame passée convertie du taux d'échantillonnage S1 au taux d'échantillonnage S2.

5. Procédé selon la revendication 4, comprenant, lorsqu'il est mis en oeuvre dans un codeur de signal sonore, le forçage de la trame actuelle dans un mode de codage qui n'utilise pas un historique d'un livre de code adaptatif.

6. Procédé selon la revendication 4 ou 5, comprenant, lorsqu'il est mis en oeuvre dans un codeur de signal sonore, le forçage d'un quantificateur de paramètres LP pour utiliser un procédé de quantification prédictive dans la trame actuelle.

7. Procédé selon l'une quelconque des revendications 1 à 6, comprenant :

le calcul du spectre de puissance du filtre de synthèse LP à K échantillons ;
l'extension du spectre de puissance du filtre de synthèse LP à K(S2/S1) échantillons lorsque le taux d'échantillonnage S1 est inférieur au taux d'échantillonnage S2 ; et
le fait de tronquer le spectre de puissance du filtre de synthèse LP à K(S2/S1) échantillons lorsque le taux d'échantillonnage S1 est supérieur au taux d'échantillonnage S2.

8. Procédé selon l'une quelconque des revendications 1 à 7, comprenant le calcul du spectre de puissance du filtre de synthèse LP en tant qu'une énergie d'une réponse fréquentielle du filtre de synthèse LP.

9. Procédé selon la revendication 3, comprenant, lorsqu'il est mis en oeuvre dans un décodeur de signal sonore, le calcul de paramètres de filtre LP dans chaque sous-trame d'une nouvelle trame par l'interpolation de paramètres de filtre LP d'une trame actuelle au taux d'échantillonnage S2 avec des paramètres de filtre LP d'une trame passée convertie du taux d'échantillonnage S1 au taux d'échantillonnage S2.

10. Dispositif destiné à être utilisé dans un codeur de signal sonore ou dans un décodeur de signal sonore pour convertir des paramètres de filtre prédictif linéaire (LP) d'un taux d'échantillonnage de signal sonore S1 à un taux d'échantillonnage de signal sonore S2, le dispositif étant **caractérisé en ce qu'**il comprend :

un processeur configuré pour :

calculer, au taux d'échantillonnage S1, un spectre de puissance d'un filtre de synthèse LP en utilisant les paramètres de filtre LP,
modifier le spectre de puissance du filtre de synthèse LP pour le convertir du taux d'échantillonnage S1 au taux d'échantillonnage S2,
effectuer une transformation inverse du spectre de puissance modifié du filtre de synthèse LP pour déterminer des autocorrélations du filtre de synthèse LP au taux d'échantillonnage S2, et

utiliser les autocorrélations pour calculer les paramètres de filtre LP au taux d'échantillonnage S2.

11. Dispositif selon la revendication 10, dans lequel le processeur est configuré pour :

étendre le spectre de puissance du filtre de synthèse LP sur la base d'un rapport entre S1 et S2 si S1 est inférieur à S2 ; et
tronquer le spectre de puissance du filtre de synthèse LP sur la base du rapport entre S1 et S2 si S1 est supérieur à S2.

12. Dispositif selon l'une quelconque des revendications 10 et 11, dans lequel le processeur est configuré pour calculer des paramètres de filtre LP dans chaque sous-trame d'une trame actuelle par l'interpolation de paramètres de filtre LP de la trame actuelle au taux d'échantillonnage S2 avec des paramètres de filtre LP d'une trame passée convertie du taux d'échantillonnage S1 au taux d'échantillonnage S2.

13. Dispositif selon l'une quelconque des revendications 10 à 12, dans lequel le processeur est configuré pour :

calculer le spectre de puissance du filtre de synthèse LP à K échantillons ;
étendre le spectre de puissance du filtre de synthèse LP à K(S2/S1) échantillons lorsque le taux d'échantillonnage S1 est inférieur au taux d'échantillonnage S2 ; et
tronquer le spectre de puissance du filtre de synthèse LP à K(S2/S1) échantillons lorsque le taux d'échantillonnage S1 est supérieur au taux d'échantillonnage S2.

14. Dispositif selon l'une quelconque des revendications 10 à 13, dans lequel le processeur est configuré pour calculer le spectre de puissance du filtre de synthèse LP en tant qu'une énergie d'une réponse fréquentielle du filtre de synthèse LP.

15. Dispositif selon l'une quelconque des revendications 10 à 14, dans lequel le processeur est configuré pour effectuer la transformation inverse du spectre de puissance modifié du filtre de synthèse LP par l'utilisation d'une transformée discrète inverse de Fourier.

16. Dispositif selon l'une quelconque des revendications 10 à 15, comprenant en outre une mémoire non transitoire mémorisant des instructions de code exécutables par le processeur pour effectuer les opérations de calcul, de modification, de transformation inverse et d'utilisation.

17. Mémoire non transitoire lisible par ordinateur mémorisant des instructions de code pour effectuer, lorsqu'elles sont exécutées sur le processeur selon l'une quelconque des revendications 10 à 16, un procédé selon l'une quelconque des revendications 1 à 9.

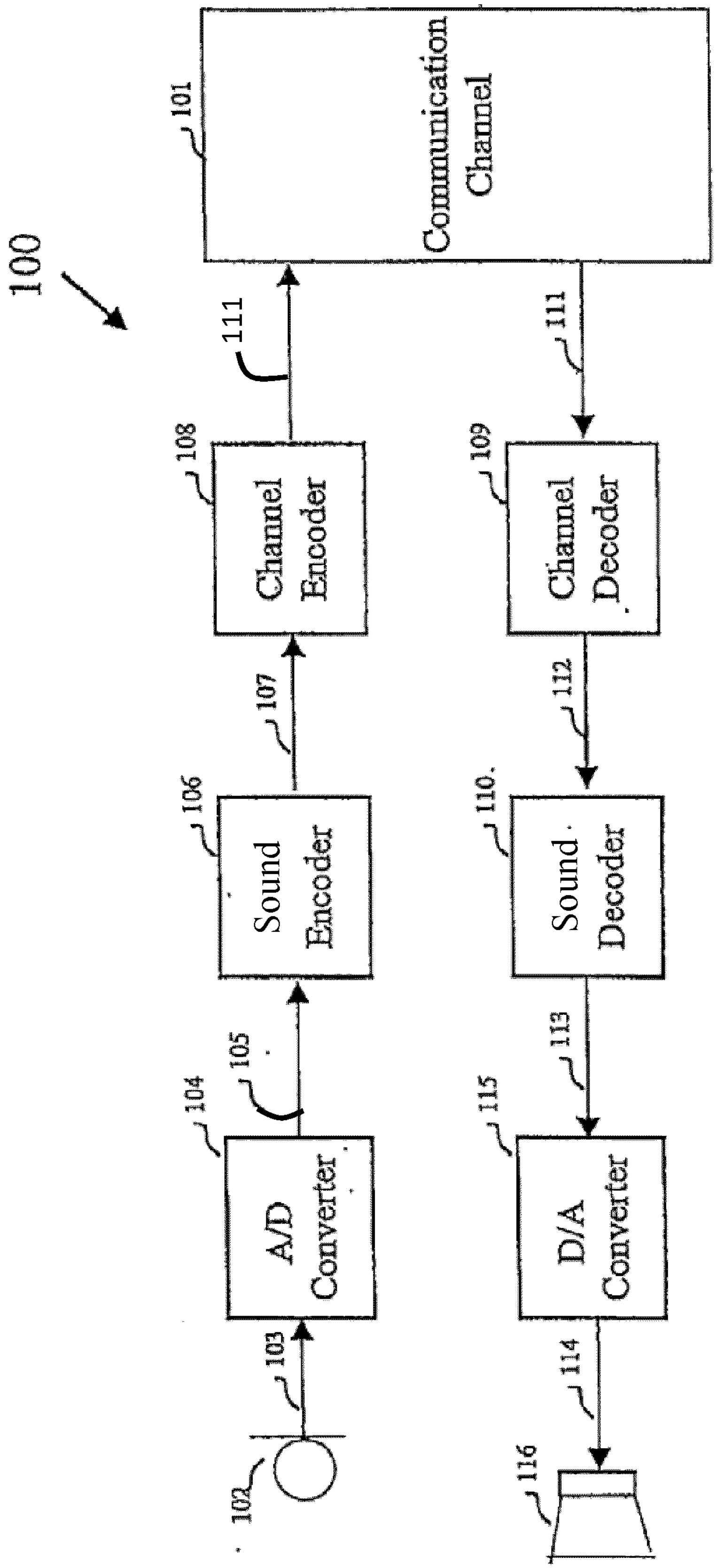
100
101
Communication Channel
102
103
104
A/D Converter
105
106
Sound Encoder
107
108
Channel Encoder
111
111
109
Channel Decoder
112
110
Sound Decoder
113
115
D/A Converter
114
116

Figure 1

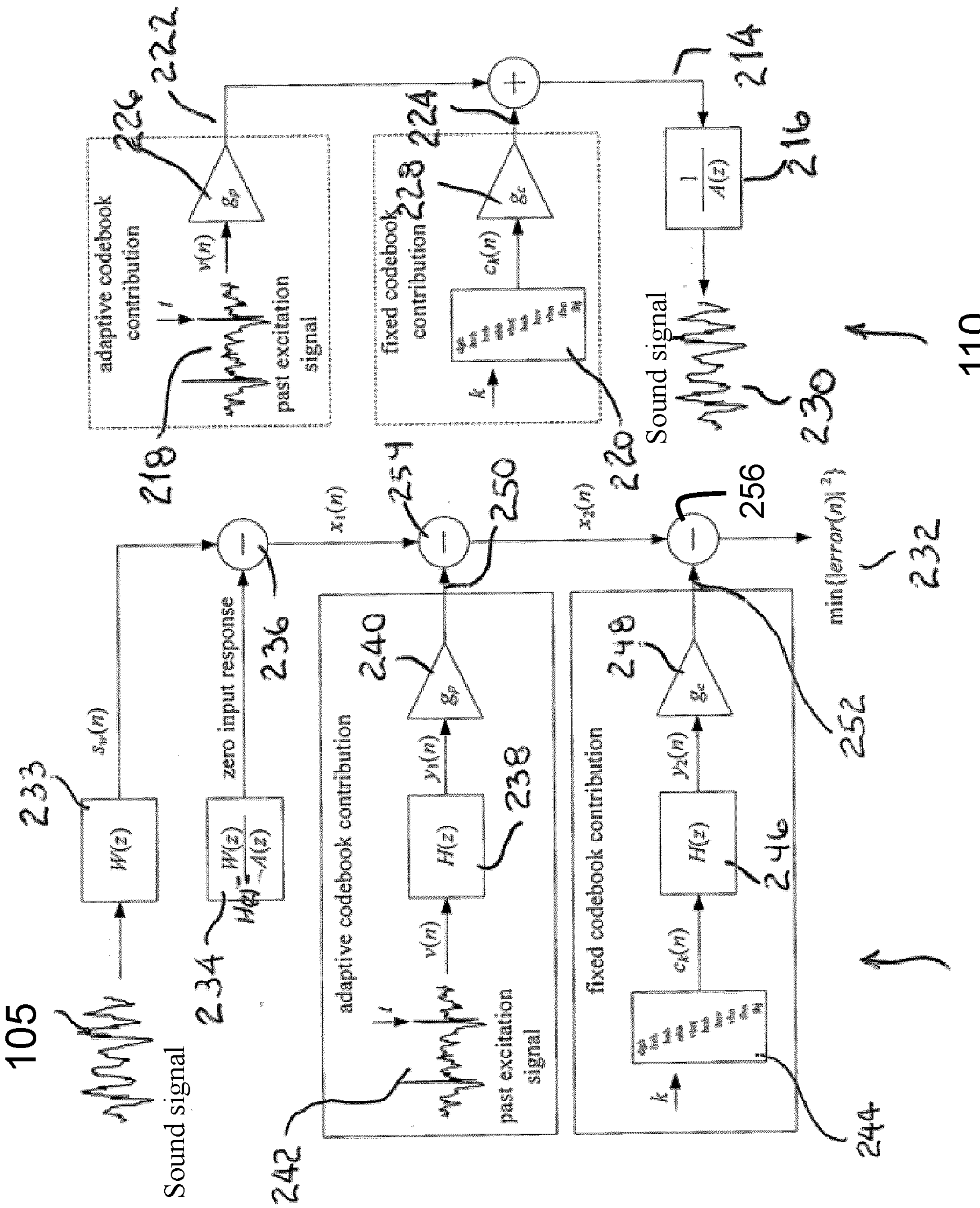
105
Sound signal
233
W(z)
s_w(n)
234
H(z) = W(z)/A(z)
zero input response
236
x_1(n)
254
250
x_2(n)
256
min{|error(n)|^2}
232
adaptive codebook contribution
242
past excitation signal
v(n)
H(z)
238
y_1(n)
g_p
240
fixed codebook contribution
k
244
c_k(n)
H(z)
246
y_2(n)
g_c
248
252
106
218
adaptive codebook contribution
past excitation signal
v(n)
g_p
226
222
fixed codebook contribution
220
c_k(n)
g_c
228
224
214
1/A(z)
216
Sound signal
230
110

Figure 2

SF1 = 0.75 F0 + 0.25 F1

SF2 = 0.5 F0 + 0.5 F1

SF3 = 0.25 F1 + 0.75 F1

SF4 = F1

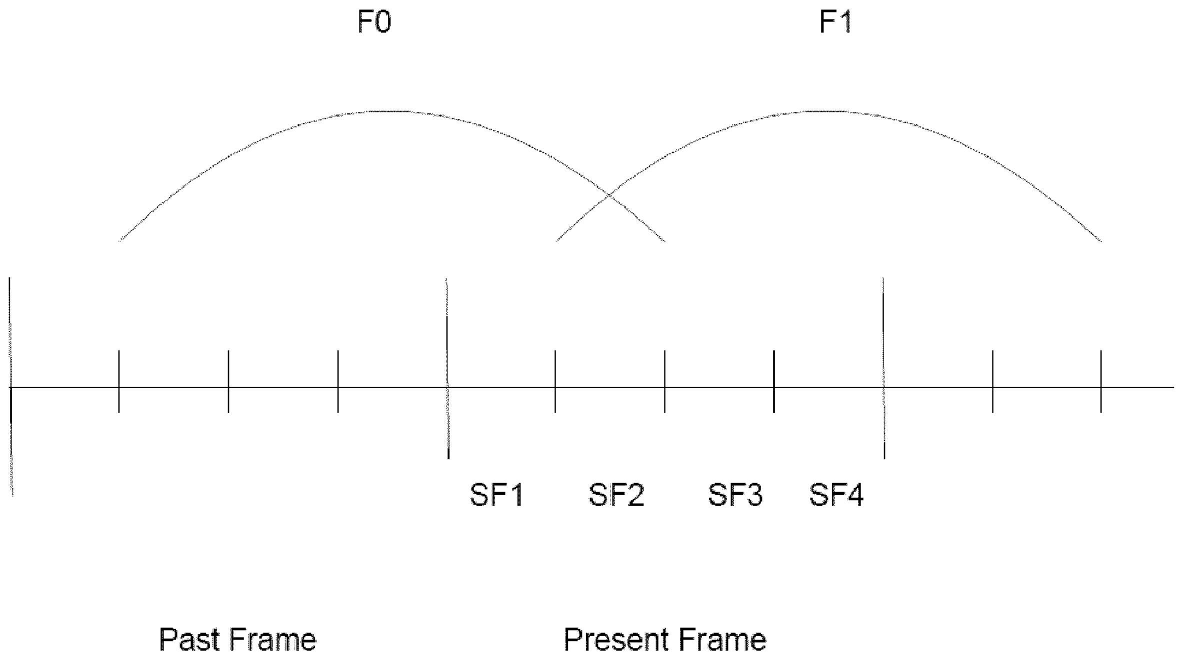

Figure 3

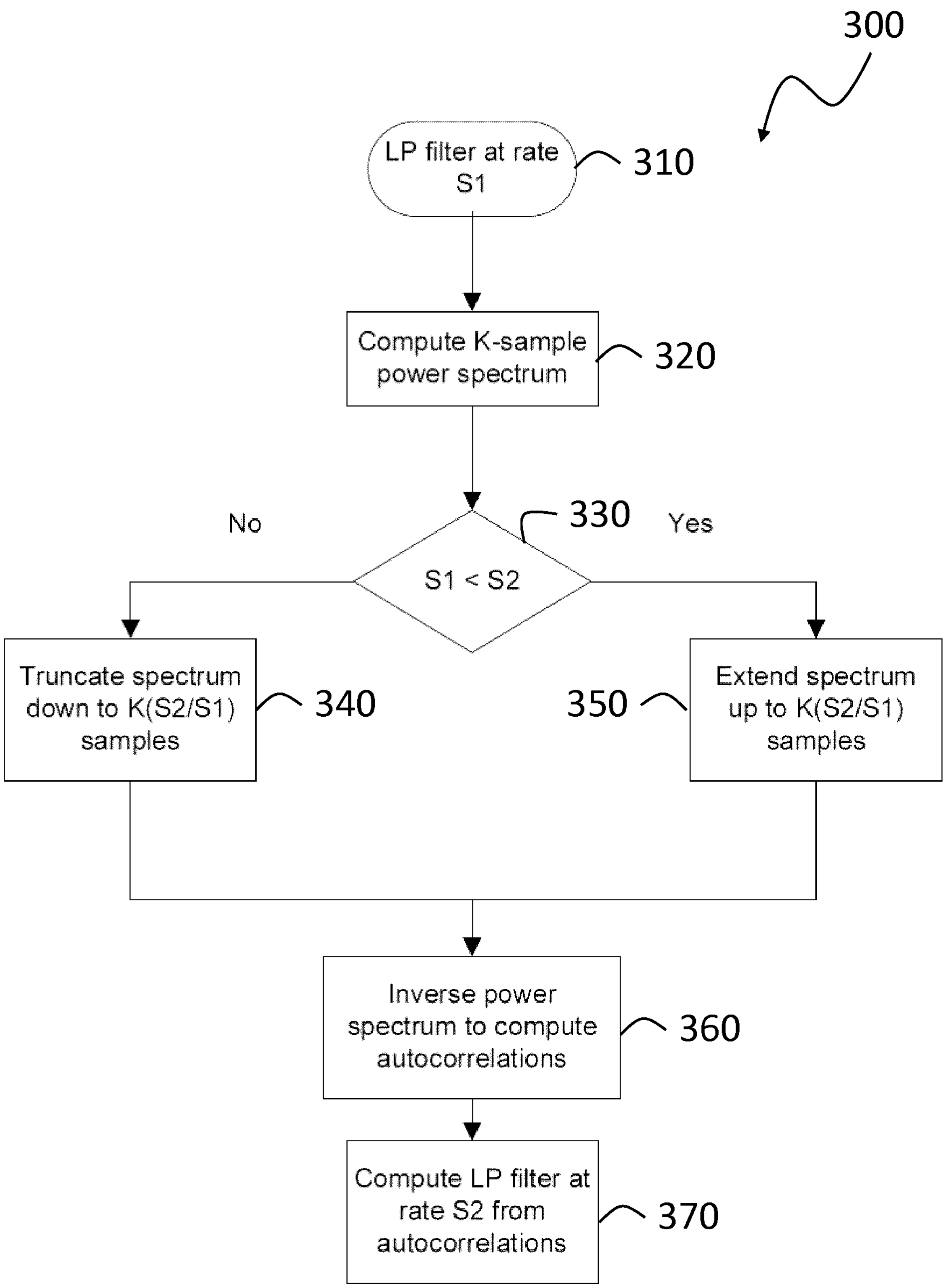
300
LP filter at rate S1
310
Compute K-sample power spectrum
320
No
330
Yes
S1 < S2
Truncate spectrum down to K(S2/S1) samples
340
350
Extend spectrum up to K(S2/S1) samples
Inverse power spectrum to compute autocorrelations
360
Compute LP filter at rate S2 from autocorrelations
370

Figure 4

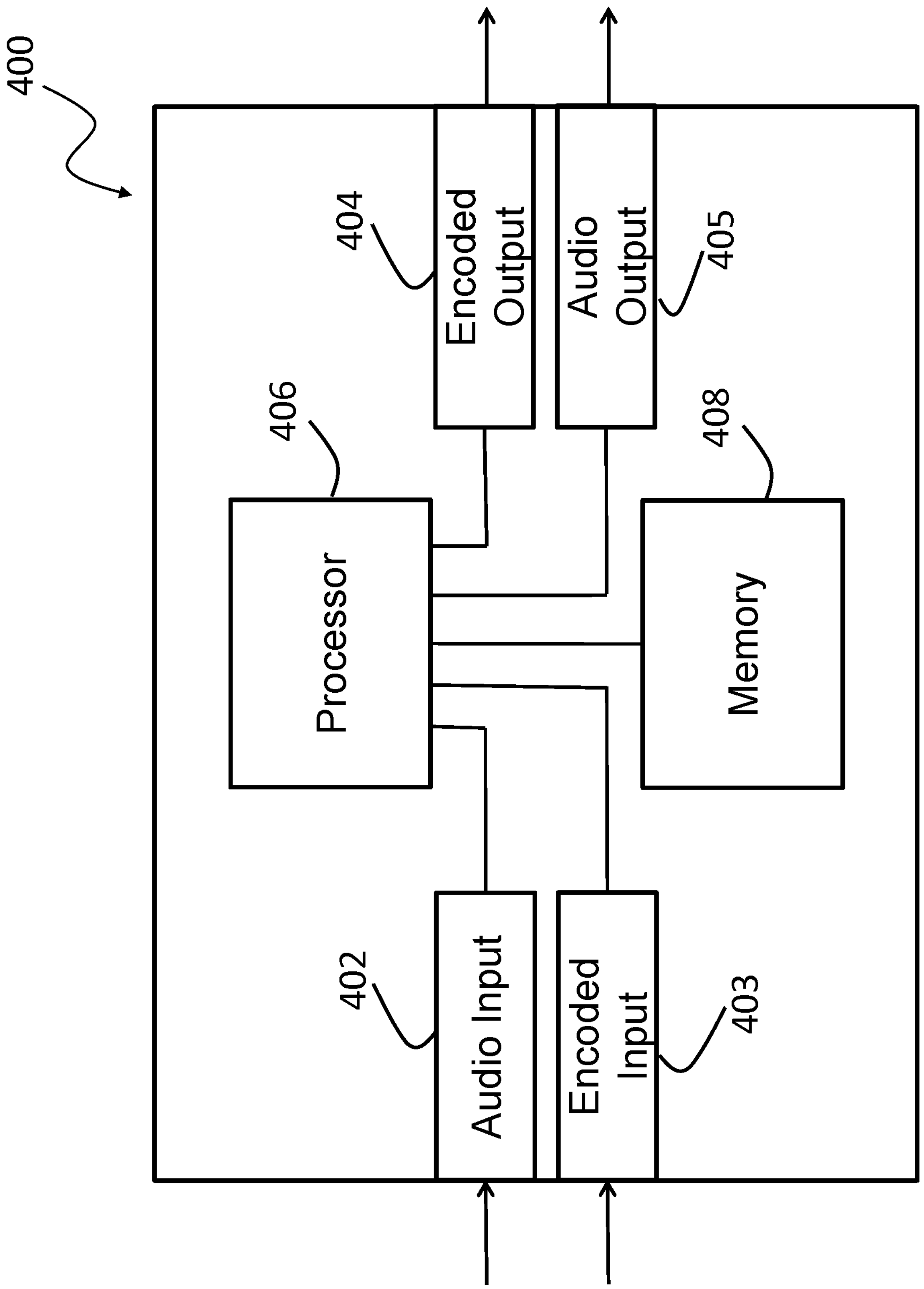
400
404
Encoded Output
Audio Output
405
406
408
Processor
Memory
402
Audio Input
Encoded Input
403

Figure 5

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 20080077401 A1 **[0008]**
- WO 2008049221 A1 **[0058]**
- US 7529660 B **[0062]**

Non-patent literature cited in the description

- Adaptive Multi-Rate-Wideband (AMR-WB) speech codec; Transcoding functions. *3GPP Technical Specification 26.190,* July 2005, http://www.3gpp.org **[0074]**
- Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP). *ITU-T Recommendation G.729,* January 2007 **[0074]**