(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
24 September 2015 (24.09.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/139958 A1

(51) International Patent Classification:
G10L 19/005 (2013.01) *G10L 19/06* (2013.01)

(21) International Application Number:
PCT/EP2015/054490

(22) International Filing Date:
4 March 2015 (04.03.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

14160774.7	19 March 2014 (19.03.2014)	EP
14167005.9	5 May 2014 (05.05.2014)	EP
14178769.7	28 July 2014 (28.07.2014)	EP

(71) Applicant: **FRAUNHOFER-GESELLSCHAFT ZUR FÖRDERUNG DER ANGEWANDTEN FORSCHUNG E.V.** [DE/DE]; Hansastraße 27c, 80686 München (DE).

(72) Inventors: **SCHNABEL, Michael**; Schneidbergweg 4a, 95179 Geroldsgrün (DE). **LECOMTE, Jérémie**; Turnstraße 7, 90763 Fürth (DE). **SPERSCHNEIDER, Ralph**; Debert 75, 91320 Ebermannstadt (DE). **JANDER, Manuel**; Liebigstr. 2, 91052 Erlangen (DE).

(74) Agents: **ZINKLER, Franz** et al.; Schoppe, Zimmermann, Stöckeler, Zinkler, Schenk & Partner mbB, Radlkoferstrasse 2, 81373 München (DE).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) Title: APPARATUS, METHOD AND CORRESPONDING COMPUTER PROGRAM FOR GENERATING AN ERROR CONCEALMENT SIGNAL USING POWER COMPENSATION

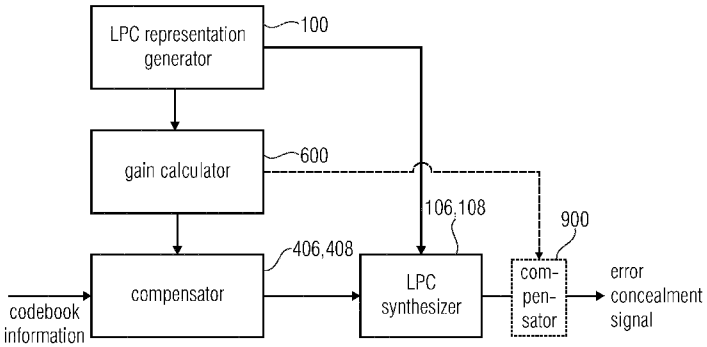


FIG 9

(57) Abstract: An apparatus for generating an error concealment signal, comprises: an LPC representation generator (100) for generating a replacement LPC representation; a gain calculator (600) for calculating a gain information from the LPC representations; a compensator (406, 408) for compensating a gain influence of the replacement LPC representation using the gain information; and an LPC synthesizer (106, 108) for filtering codebook information using the replacement LPC representation to obtain the error concealment signal, wherein the compensator (406, 408, 900) is configured for weighting the codebook information or an LPC synthesis output signal.

WO 2015/139958 A1

APPARATUS, METHOD AND CORRESPONDING COMPUTER PROGRAM FOR GENERATING AN ERROR CONCEALMENT SIGNAL USING POWER COMPENSATION

Specification

The present invention relates to audio coding and in particular to audio coding based on LPC-like processing in the context of codebooks.

Perceptual audio coders often utilize linear predictive coding (LPC) in order to model the human vocal tract and in order to reduce the amount of redundancy, which can be modeled by the LPC parameters. The LPC residual, which is obtained by filtering the input signal with the LPC filter, is further modeled and transmitted by representing it by one, two or more codebooks (examples are: adaptive codebook, glottal pulse codebook, innovative codebook, transition codebook, hybrid codebooks consisting of predictive and transform parts).

In case of a frame loss, a segment of speech/audio data (typically 10 ms or 20 ms) is lost. To make this loss as less audible as possible, various concealment techniques are applied. These techniques usually consist of extrapolation of the past, received data. This data may be: gains of codebooks, codebook vectors, parameters for modeling the codebooks and LPC coefficients. In all concealment technology known from state-of-the-art, the set of LPC coefficients, which is used for the signal synthesis, is either repeated (based on the last good set) or is extra-/interpolated.

ITU G.718 [1]: The LPC parameters (represented in the ISF domain) are extrapolated during concealment. The extrapolation consists of two steps. First, a long term target ISF vector is calculated. This long term target ISF vector is a weighted mean (with the fixed weighting factor$beta$) of

- an ISF vector representing the average of the last three known ISF vectors, and
- an offline trained ISF vector, which represents a long-term average spectral shape.

This long term target ISF vector is then interpolated with the last correctly received ISF vector once per frame using a time-varying factor $alpha$ to allow a cross-fade from the last

received ISF vector to the long term target ISF vector. The resulting ISF vector is subsequently converted back to the LPC domain, in order to generate intermediate steps (ISFs are transmitted every 20 ms, interpolation generates a set of LPCs every 5 ms). The LPCs are then used to synthesize the output signal by filtering the result of the sum of the adaptive and the fixed codebook, which are amplified with the corresponding codebook gains before addition. The fixed codebook contains noise during concealment. In case of consecutive frame loss, the adaptive codebook is fed back without adding the fixed codebook. Alternatively, the sum signal might be fed back, as done in AMR-WB [5].

In [2], a concealment scheme is described which utilizes two sets of LPC coefficients. One set of LPC coefficients is derived based on the last good received frame, the other set of LPC parameters is derived based on the first good received frame, but it is assumed that the signal evolves in reverse direction (towards the past). Then prediction is performed in two directions, one towards the future and one towards the past. Therefore, two representations of the missing frame are generated. Finally, both signals are weighted and averaged before being played out.

Fig. 8 shows an error concealment processing in accordance with the prior art. An adaptive codebook 800 provides an adaptive codebook information to an amplifier 808 which applies a codebook gain g_p to the information from the adaptive codebook 800. The output of the amplifier 808 is connected to an input of a combiner 810. Furthermore, a random noise generator 804 together with a fixed codebook 802 provides codebook information to a further amplifier g_c. The amplifier g_c indicated at 806 applies the gain factor g_c, which is the fixed codebook gain, to the information provided by the fixed codebook 802 together with the random noise generator 804. The output of the amplifier 806 is then additionally input into the combiner 810. The combiner 810 adds the result of both codebooks amplified by the corresponding codebook gains to obtain a combination signal which is then input into an LPC synthesis block 814. The LPC synthesis block 814 is controlled by replacement representation which is generated as discussed before.

This prior art procedure has certain drawbacks.

In order to cope with changing signal characteristics or in order to converge the LPC envelope towards background noise like-properties, the LPC is changed during concealment by extra/interpolation with some other LPC vectors. There is no possibility to precisely control the energy during concealment. While there is the chance to control the

codebook gains of the various codebooks, the LPC will implicitly influence the overall level or energy (even frequency dependent).

It might be envisioned to fade out to a distinct energy level (e.g. background noise level) during burst frame loss. This is not possible with state-of-the-art technology, even by controlling the codebook gains.

It is not possible to fade the noisy parts of the signal to background noise, while maintaining the possibility to synthesize tonal parts with the same spectral property as before the frame loss.

It is an object of the present invention to provide an improved concept for generating an error concealment signal.

This object is achieved by an apparatus for generating an error concealment signal of claim 1, a method of generating an error concealment signal of claim 14 or a computer program of claim 15.

In an aspect of the present invention, the apparatus for generating an error concealment signal comprises an LPC representation generator for generating a first replacement LPC representation and a different, second replacement LPC representation. Furthermore, an LPC synthesizer is provided for filtering a first codebook information using the first replacement LPC representation to obtain a first replacement signal and for filtering a second different codebook information using the second replacement LPC representation to obtain a second replacement signal. The outputs of the LPC synthesizer are combined by a replacement signal combiner combining the first replacement signal and the second replacement signal to obtain the error concealment signal.

The first codebook is preferably an adaptive codebook for providing the first codebook information and the second codebook as preferably a fixed codebook for providing the second codebook information. In other words, the first codebook represents the tonal part of the signal and the second or fixed codebook represents the noisy part of the signal and therefore can be considered to be a noise codebook.

The first codebook information for the adaptive codebook is generated using a mean value of last good LPC representations, the last good representation and a fading value.

Furthermore, the LPC representation for the second or fixed codebook is generated using the last good LPC representation fading value and a noise estimate. Depending on the implementation, the noise estimate can be a fixed value, an offline trained value or it can be adaptively derived from a signal preceding an error concealment situation.

Preferably, an LPC gain calculation for calculating an influence of a replacement LPC representation is performed and this information is then used in order to perform a compensation so that the power or loudness or, generally, an amplitude-related measure of the synthesis signal is similar to the corresponding synthesis signal before the error concealment operation.

In a further aspect, an apparatus for generating an error concealment signal comprises an LPC representation generator for generating one or more replacement LPC representations. Furthermore, the gain calculator is provided for calculating the gain information from the LPC representation and a compensator is then additionally provided for compensating a gain influence of the replacement LPC representation and this gain compensation operates using the gain operation provided by the gain calculator. An LPC synthesizer then filters a codebook information using the replacement LPC representation to obtain the error concealment signal, wherein the compensator is configured for weighting the codebook information before being synthesized by the LPC synthesizer or for weighting the LPC synthesis output signal. Thus, any gain or power or amplitude-related perceivable influence at the onset of an error concealment situation is reduced or eliminated.

This compensation is not only useful for individual LPC representations as outlined in the above aspect, but is also useful in the case of using only a single LPC replacement representation together with a single LPC synthesizer.

The gain values are determined by calculating impulse responses of the last good LPC representation and a replacement LPC representation and by particularly calculating an rms value over the impulse response of the corresponding LPC representation over a certain time which is between 3 and 8 ms and is preferably 5 ms.

In an implementation, the actual gain value is determined by dividing a new rms value, i.e. an rms value for a replacement LPC representation by an rms value of good LPC representation.

Preferably, the single or several replacement LPC representations is/are calculated using a background noise estimate which is preferably a background noise estimate derived from the currently decoded signals in contrast to an offline trained vector simply predetermined noise estimate.

In a further aspect, an apparatus for generating a signal comprises an LPC representation generator for generating one or more replacement LPC representations, and an LPC synthesizer for filtering a codebook information using the replacement LPC representation. Additionally, a noise estimator for estimating a noise estimate during a reception of good audio frames is provided, and this noise estimate depends on the good audio frames. The representation generator is configured to use the noise estimate estimated by the noise estimator in generating the replacement LPC representation.

Spectral representation of a past decoded signal is process to provide a noise spectral representation or target representation. The noise spectral representation is converted into a noise LPC representation and the noise LPC representation is preferably the same kind of LPC representation as the replacement LPC representation. ISF vectors are preferred for the specific LPC-related processing procedures.

Estimate is derived using a minimum statistics approach with optimal smoothing to a past decoded signal. This spectral noise estimate is then converted into a time domain representation. Then, a Levinson-Durbin recursion is performed using a first number of samples of the time domain representation, where the number of samples is equal to an LPC order. Then, the LPC coefficients are derived from the result of the Levinson-Durbin recursion and this result is finally transformed in a vector. The aspect of using individual LPC representations for individual codebooks, the aspect of using one or more LPC representations with a gain compensation and the aspect of using a noise estimate in generating one or more LPC representations, which estimate is not an offline-trained vector but is a noise estimate derived from the past decoded signal are individually useable for obtaining an improvement with respect to the prior art.

Additionally, these individual aspects can also be combined with each other so that, for example, the first aspect and the second aspect can be combined or the first aspect or the third aspect can be combined or the second aspect and the third aspect can be combined to each other to provide an even improved performance with respect to the prior art. Even

more preferably, all three aspects can be combined with each other to obtain improvements over the prior art. Thus, even though the aspects are described by separate figures all aspects can be applied in combination with each other, as can be seen by referring to the enclosed figures and description.

Preferred embodiments of the present invention are subsequently described with respect to the accompanying drawings, in which:

Fig. 1a illustrates an embodiment of the first aspect;

Fig. 1b illustrates a usage of an adaptive codebook;

Fig. 1c illustrates a usage of a fixed codebook in the case of a normal mode or a concealment mode;

Fig. 1d illustrates a flowchart for calculating the first LPC replacement representation;

Fig. 1e illustrates a flowchart for calculating the second LPC replacement representation;

Fig. 2 illustrates an overview over a decoder with error concealment controller and noise estimator;

Fig. 3 illustrates a detailed representation of the synthesis filters;

Fig. 4 illustrates a preferred embodiment combining the first aspect and the second aspect;

Fig. 5 illustrates a further embodiment combining the first and second aspects;

Fig. 6 illustrates the embodiment combining the first and second aspects;

Fig. 7a illustrates an embodiment for performing a gain compensation.

Fig. 7b illustrates a flowchart for performing a gain compensation;

Fig. 8 illustrates a prior art error concealment signal generator;

Fig. 9 illustrates an embodiment in accordance with the second aspect with gain compensation;

Fig. 10 illustrates a further implementation of the embodiment of Fig. 9;

Fig. 11 illustrates an embodiment of the third aspect using the noise estimator;

Fig. 12a illustrates a preferred implementation for calculating the noise estimate;

Fig. 12b illustrates a further preferred implementation for calculating the noise estimate; and

Fig. 13 illustrates the calculation of a single LPC replacement representation or individual LPC replacement representations for individual codebooks using a noise estimate and applying a fading operation.

Preferred embodiments of the present invention relate to controlling the level of the output signal by means of the codebook gains independently of any gain change caused by an extrapolated LPC and to control the LPC modeled spectral shape separately for each codebook. For this purpose, separate LPCs are applied for each codebook and compensation means are applied to compensate for any change of the LPC gain during concealment.

Embodiments of the present invention as defined in the different aspects or in combined aspects have the advantage of providing a high subjective quality of speech/audio in case of one or more data packets not being correctly or not being received at all at the decoder side.

Furthermore, the preferred embodiments compensate the gain differences between subsequent LPCs during concealment, which might result from the LPC coefficients being changed over time, and therefore unwanted level changes are avoided.

Furthermore, embodiments are advantageous in that during concealment two or more sets of LPC coefficients are used to independently influence the spectral behavior of voiced and unvoiced speech parts and also tonal and noise-like audio parts.

All aspects of the present invention provide an improved subjective audio quality.

According to one aspect of this invention, the energy is precisely controlled during the interpolation. Any gain that is introduced by changing the LPC is compensated.

According to another aspect of this invention, individual LPC coefficient sets are utilized for each of the codebook vectors. Each codebook vector is filtered by its corresponding LPC and the individual filtered signals are just afterwards summed up to obtain the synthesized output. In contrast, state-of-the-art technology first adds up all excitation vectors (being generated from different codebooks) and just then feeds the sum to a single LPC filter.

According to another aspect, a noise estimate is not used, for example as an offline-trained vector, but is actually derived from the past decoded frames so that, after a certain amount of erroneous or missing packets/frames, a fade-out to the actual background noise rather than any predetermined noise spectrum is obtained. This particularly results in a feeling of acceptance at a user side, but to the fact that even when an error situation occurs, the signal provided by the decoder after a certain number of frames is related to the preceding signal. However, the signal provided by a decoder in the case of a certain number of lost or erroneous frames is a signal completely unrelated to the signal provided by the decoder before an error situation.

Applying gain compensation for the time-varying gain of the LPC allows the following advantages:

It compensates any gain that is introduced by changing the LPC.

Hence, the level of the output signal can be controlled by the codebook gains of the various codebooks. This allows for a pre-determined fade-out by eliminating any unwanted influence by the interpolated LPC.

Using a separate set of LPC coefficients for each codebook used during concealment allows the following advantages:

It creates the possibility to influence the spectral shape of tonal and noise like parts of the signal separately.

It gives the chance to play out the voiced signal part almost unchanged (e.g. desired for vowels), while the noise part may quickly be converging to background noise.

It gives the chance to conceal voiced parts, and fade out the voiced part with arbitrary fading speed (e.g. fade out speed dependent from signal characteristics), while simultaneously maintaining the background noise during concealment. State-of-the-art codecs usually suffer from a very clean voiced concealment sound.

It provides means to fade to background noise during concealment smoothly, by fading out the tonal parts without changing the spectral properties, and fading the noise like parts to the background spectral envelope.

Fig. 1a illustrates an apparatus for generating an error concealment signal 111. The apparatus comprises an LPC representation generator 100 for generating a first replacement representation and additionally for generating a second replacement LPC representation. As outlined in Fig. 1a, the first replacement representation is input into an LPC synthesizer 106 for filtering a first codebook information output by a first codebook 102 such as an adaptive codebook 102 to obtain a first replacement signal at the output of block 106. Furthermore, the second replacement representation generated by the LPC representation generator 100 is input into the LPC synthesizer for filtering a second different codebook information provided by a second codebook 104 which is, for example, a fixed codebook, to obtain a second replacement signal at the output of block 108. Both replacement signals are then input into a replacement signal combiner 110 for combining the first replacement signal and the second replacement signal to obtain the error concealment signal 111. Both LPC synthesizers 106, 108 can be implemented in a single LPC synthesizer block or can be implemented as separate LPC synthesizer filters. In other implementations, both LPC synthesizer procedures can be implemented by two LPC filters actually being implemented and operating in parallel. However, the LPC synthesis can also be an LPC synthesis filter and a certain control so that the LPC synthesis filter provides an output signal for the first codebook information and the first replacement

representation and then, subsequent to this first operation, the control provides the second codebook information and the second replacement representation to the synthesis filter to obtain the second replacement signal in a serial way. Other implementations for the LPC synthesizer apart from a single or several synthesis blocks are clear for those skilled in the art.

Typically, the LPC synthesis output signals are time domain signals and the replacement signal combiner 110 performs a synthesis output signal combination by performing a synchronized sample-by-sample addition. However, other combinations, such as a weighted sample-by-sample addition or a frequency domain addition or any other signal combination can be performed by the replacement signal combiner 110 as well.

Furthermore, the first codebook 102 is indicated as comprising an adaptive codebook and the second codebook 104 is indicated as comprising a fixed codebook. However, the first codebook and the second codebook can be any codebooks such as a predictive codebook as the first codebook and a noise codebook as the second codebook. However, other codebooks can be glottal pulse codebooks, innovative codebooks, transition codebooks, hybrid codebooks consisting of predictive and transform parts, codebooks for individual voice generators such as males/females/children or codebooks for different sounds such as for animal sounds, etc.

Fig. 1b illustrates a representation of an adaptive codebook. The adaptive codebook is provided with a feedback loop 120 and receives, as an input, a pitch lag 118. The pitch lag can be a decoded pitch lag in the case of a good received frame/packet. However, if an error situation is detected indicating an erroneous or missing frame/packet, then an error concealment pitch lag 118 is provided by the decoder and input into the adaptive codebook. The adaptive codebook 102 can be implemented as a memory storing the fed back output values provided via the feedback line 120 and, depending on the applied pitch lag 118, a certain amount of sampling values is output by the adaptive codebook.

Furthermore, Fig. 1c illustrates a fixed codebook 104. In the case of the normal mode, the fixed codebook 104 receives a codebook index and, in response to the codebook index, a certain codebook entry 114 is provided by the fixed codebook as codebook information. However, if a concealment mode is determined, a codebook index is not available. Then, a noise generator 112 provided within the fixed codebook 104 is activated which provides a noise signal as the codebook information 116. Depending on the implementation, the

noise generator may provide a random codebook index. However, it is preferred that a noise generator actually provides a noise signal rather than a random codebook index. The noise generator 112 may be implemented as a certain hardware or software noise generator or can be implemented as noise tables or a certain "additional" entry in the fixed codebook which has a noise shape. Furthermore, combinations of the above procedures are possible, i.e. a noise codebook entry together with a certain post-processing.

Fig. 1d illustrates a preferred procedure for calculating a first replacement LPC representation in the case of an error. Step 130 illustrates the calculation of a mean value of LPC representations of two or more last good frames. Three last good frames are preferred. Thus, a mean value over the three last good frames is calculated in block 130 and provided to block 136. Furthermore, a stored last good frame LPC information is provided in step 132 and additionally provided to the block 136. Furthermore, a fading factor 134 is determined in block 134. Then, depending on the last good LPC information, depending on the mean value of the LPC information of the last good frame and depending on the fading factor of block 134, the first replacement representation 138 is calculated.

For the state-of-the-art just one LPC is applied. For the newly proposed method, each excitation vector, which is generated by either the adaptive or the fixed codebook, is filtered by its own set of LPC coefficients. The derivation of the individual ISF vectors is as follows:

Coefficient set A (for filtering the adaptive codebook) is determined by this formula:

$$isf' = \frac{isf^{-2}+isf^{-3}+isf^{-4}}{3} \quad \text{(block 136)}$$

$$isf_A{}^{-1} = alpha_A \cdot isf^{-2} + (1 - alpha) \cdot isf' \quad \text{(block 136)}$$

where $alpha_A$ is a time varying adaptive fading factor which may depend on signal stability, signal class, etc. isf^{-x} are the ISF coefficients, where x denotes the frame number, relative to the end of the current frame: x = -1 denotes the first lost ISF, x = -2 the last good, x=-3 second last good and so on.
This leads to fading the LPC which is used for filtering the tonal part, starting from the last correctly received frame towards the average LPC (averaged over three of the last good

20ms frames). The more frames get lost, the closer the ISF, which is used during concealment, will be to this short term average ISF vector (isf').

Fig. 1e illustrates a preferred procedure for calculating the second replacement representation. In block 140, a noise estimate is determined. Then, in block 142, a fading factor is determined. Additionally, in block 144, the last good frame is LPC information which has been stored before is provided. Then, in block 146, a second replacement representation is calculated. Preferably, a coefficient set B (for filtering the fixed codebook) is determined by this formula:

$$isf_B^{-1} = alpha_B \cdot isf^{-2} + (1 - beta) \cdot isf^{cng} \quad \text{(block 146)}$$

where isf^{cng} is the ISF coefficient set derived from a background noise estimate and $alpha_B$ is the time-varying fading speed factor which preferably is signal dependent. The target spectral shape is derived by tracing the past decoded signal in the FFT domain (power spectrum), using a minimum statistics approach with optimal smoothing, similar to [3]. This FFT estimate is converted to the LPC representation by calculating the auto-correlation by doing inverse FFT and then using Levinson-Durbin recursion to calculate LPC coefficients using the first N samples of the inverse FFT, where N is the LPC order. This LPC is then converted into the ISF domain to retrieve isf^{cng}. Alternatively – if such tracing of the background spectral shape is not available – the target spectral shape might also be derived based on any combination of an offline trained vector and the short-term spectral mean, as it is done in G.718 for the common target spectral shape.

Preferably, the fading factors A and α_B are determined depending on the decoded audio signal, i.e., depending on the decoded audio signal before the occurrence of an error. The fading factor may depend on signal stability, signal class, etc. Thus, is the signal is determined to be a quite noisy signal, then the fading factor is determined in such a way that the fading factor decreases, from time to time, more quickly than compared to a situation where a signal is quite tonal. In this situation, the fading factor decreases from one time frame to next time frame by a reduced amount. This makes sure that the fading out from the last good frame to the mean value of the last three good frames takes place more quickly in the case of noisy signals compared to non-noisy or tonal signals, where the fading out speed is reduced. Similar procedures can be performed for signal classes. For voiced signals, a fading out can be performed slower than for unvoiced signals or for

music signals a certain fading speed can be reduced compared to further signal characteristics and corresponding determinations of the fading factor can be applied.

As discussed in the context of Fig. 1e, a different fading factor α_B can be calculated for the second codebook information. Thus, the different codebook entries can be provided with a different fading speed. Thus, a fading out to the noise estimate as f^{cng} can be set differently from the fading speed from the last good frame ISF representation to the mean ISF representation as outlined in block 136 of Fig. 1d.

Fig. 2 illustrates an overview of a preferred implementation. An input line receives, for example, from a wireless input interface or a cable interface packets or frames of an audio signal. The data on the input line 202 is provided to a decoder 204 and at the same time to an error concealment controller 200. The error concealment controller determines whether received packet or frames are erroneous or missing. If this is determined, the error concealment controller inputs a control message to the decoder 204. In the Fig. 2 implementation, a "1" message on the control line CTRL signals that the decoder 204 is to operate in the concealment mode. However, if the error concealment controller does not find an error situation, then the control line CTRL carries a "0" message indicating a normal decoding mode as indicated in table 210 of Fig. 2. The decoder 204 is additionally connected to a noise estimator 206. During the normal decoding mode, the noise estimator 206 receives the decoded audio signal via a feedback line 208 and determines a noise estimate from the decoded signal. However, when the error concealment controller indicates a change from the normal decoding mode to the concealment mode, the noise estimator 206 provides the noise estimate to the decoder 204 so that the decoder 204 can perform an error concealment as discussed in the preceding and the next figures. Thus, the noise estimator 206 is additionally controlled by the control line CTRL from the error concealment controller to switch, from the normal noise estimation mode in the normal decoding mode to the noise estimate provision operation in the concealment mode.

Fig. 4 illustrates a preferred embodiment of the present invention in the context of a decoder, such as the decoder 204 of Fig. 2, having an adaptive codebook 102 and additionally having a fixed codebook 104. In the normal decoding mode indicated by a control line data "0" as discussed in the context of the table 210 in Fig. 2, the decoder operates as illustrated in Fig. 8, when item 804 is neglected. Thus, the correctly received packet comprises a fixed codebook index for controlling the fixed codebook 802, a fixed

codebook gain g_c for controlling amplifier 806 and an adaptive codebook g_p in order to control the amplifier 808. Furthermore, the adaptive codebook 800 is controlled by the transmitted pitch lag and the switch 812 is connected so that the adaptive codebook output is fed back into the input of the adaptive codebook. Furthermore, the coefficients for the LPC synthesis filter 804 are derived from the transmitted data.

However, if an error concealment situation is detected by the error concealment controller 202 of Fig. 2, the error concealment procedure is initiated in which, in contrast to the normal procedure, two synthesis filters 106, 108 are provided. Furthermore, the pitch lag for the adaptive codebook 102 is generated by an error concealment device. Additionally, the adaptive codebook gain g_p and the fixed codebook gain g_c are also synthesized by an error concealment procedure as known in the art in order to correctly control the amplifiers 402, 404.

Furthermore, depending on the signal class, a controller 409 controls the switch 405 in order to either feedback a combination of both codebook outputs (subsequent to the application of the corresponding codebook gain) or to only feedback the adaptive codebook output.

In accordance with an embodiment, the data for the LPC synthesis filter A 106 and the data for the LPC synthesis filter B 108 is generated by the LPC representation generator 100 of Fig. 1a and additionally a gain correction is performed by the amplifiers 406, 408. To this end, the gain compensation factors g_A and g_B are calculated in order to correctly drive the amplifiers 408, 406 so that any gain influence generated by the LPC representation is stopped. Finally, the output of the LPC synthesis filters A, B indicated by 106 and 108 are combined by the combiner 110, so that the error concealment signal is obtained.

Subsequently, the switching from the normal mode to the concealment mode on one hand and from the concealment mode back to the normal mode is discussed.

The transition from one common to several separate LPCs when switching from clean channel decoding to concealment does not cause any discontinuities, as the memory state of the last good LPC may be used to initialize each AR or MA memory of the separate LPCs. When doing so, a smooth transition from the last good to the first lost frame is ensured.

When switching from concealment to clean channel decoding (recovery phase), the approach of the separate LPCs introduces the challenge to correctly update the internal memory state of the single LPC filter during clean-channel decoding (usually AR (auto-regressive) models are used). Just using the AR memory of one LPC or an averaged AR memory would lead to discontinuities at the frame border between the last lost and the first good frame. In the following a method is described to overcome deal with this challenge:

A small portion of all excitation vectors (suggestion: 5ms) is added at the end of any concealed frame. This summed excitation vector may then be fed to the LPC which would be used for recovery. This is shown in Fig. 5. Depending on the implementation it is also possible to sum up the excitation vectors after the LPC gain compensation.

It is advisable to start at frame end minus 5ms, setting the LPC AR memory to zero, derive the LPC synthesis by using any of the individual LPC coefficient sets and save the memory state at the very end of the concealed frame. If the next frame is correctly received, this memory state may then be used for recovery (meaning: used for initializing the start-of-frame LPC memory), otherwise it is discarded. This memory has to be additionally introduced; it must be handled separately from any of the used LPC AR memories of the concealment used during concealment.

Another solution for recovery is to use the method LPC0, known from USAC [4].

Subsequently, Fig. 5 is discussed in more detail. Generally, the adaptive codebook 102 can be termed to be a predictive codebook as indicated in Fig. 5 or can be replaced by a predictive codebook. Furthermore, the fixed codebook 104 can be replaced or implemented as the noise codebook 104. The codebook gains g_p and g_c, in order to correctly drive the amplifiers 402, 404 are transmitted, in the normal mode, in the input data or can be synthesized by an error concealment procedure in the error concealment case. Furthermore, a third codebook 412, which can be any other codebook, is used which additionally has an associated codebook gain g_r as indicated by amplifier 414. In an embodiment, an additional LPC synthesis by a separate filter controlled by an LPC replacement representation for the other codebook is implemented in block 416. Furthermore, a gain correction g_c is performed in a similar way as discussed in the context of g_A and g_B, as outlined.

Furthermore, the additional recovery LPC synthesizer X indicated at 418 is shown which receives, as an input, a sum of at least a small portion of all excitation vectors such as 5 ms. This excitation vector is input into the LPC synthesizer X 418 memory states of the LPC synthesis filter X.

Then, when a switchback from the concealment mode to the normal mode occurs, the single LPC synthesis filter is controlled by copying the internal memory states of the LPC synthesis filter X into this single normal operating filter and additionally the coefficients of the filter are set by the correctly transmitted LPC representation.

Fig. 3 illustrates a further, more detailed implementation of the LPC synthesizer having two LPC synthesis filters 106, 108. Each filter is, for example, an FIR filter or an IIR filter having filter taps 304, 306 and filter-internal memories 304, 308. The filter taps 302, 306 are controlled by the corresponding LPC representation correctly transmitted or the corresponding replacement LPC representation generated by the LPC representation generator such as 100 of Fig. 1a. Furthermore, a memory initializer 320 is provided. The memory initializer 320 receives the last good LPC representation and, when switch over to the error concealment mode is performed, the memory initializer 320 provides the memory states of the single LPC synthesis filter to the filter-internal memories 304, 308. In particular, the memory initializer receives, instead of the last good LPC representation or in addition to the last good LPC representation, the last good memory states, i.e. the internal memory states of the single LPC filter in the processing, and particularly after the processing of the last good frame/packet.

Additionally, as already discussed in the context of Fig. 5, the memory initializer 320 can also be configured to perform the memory initialization procedure for a recovery from an error concealment situation to the normal non-erroneous operating mode. To this end, the memory initializer 320 or a separate future LPC memory initializer is configured for initializing a single LPC filter in the case of a recovery from an erroneous or lost frame to a good frame. The LPC memory initializer is configured for feeding at least a portion of a combined first codebook information and second codebook information or at least a portion of a combined weighted first codebook information or a weighted second codebook information into a separate LPC filter such as LPC filter 418 of Fig. 5. Additionally, the LPC memory initializer is configured for saving memory states obtained by processing the fed in values. Then, when a subsequent frame or packet is a good

frame or packet, the single LPC filter 814 of Fig. 8 for the normal mode is initialized using the saved memory states, i.e. the states from filter 418. Furthermore, as outlined in Fig. 5, the filter coefficients for the filter can be either the coefficient for LPC synthesis filter 106 or LPC synthesis filter 108 or LPC synthesis filter 416 or a weighted or unweighted combination of those coefficients.

Fig. 6 illustrates a further implementation with gain compensation. To this end, the apparatus for generating an error concealment signal comprises a gain calculator 600 and a compensator 406, 408, which has already been discussed in the context of Fig. 4 (406, 408) and Fig. 5 (406, 408, 409). In particular, the LPC representation calculator 100 outputs the first replacement LPC representation and the second replacement LPC representation to a gain calculator 600. The gain calculator then calculates a first gain information for the first replacement LPC representation and the second gain information for the second LPC replacement representation and provides this data to the compensator 406, 408, which receives, in addition to the first and second codebook information, as outlined in Fig. 4 or Fig. 5, the LPC of the last good frame/packet/block. Then, the compensator outputs the compensated signal. The input into the compensator can either be an output of amplifiers 402, 404, an output of the codebooks 102, 104 or an output of the synthesis blocks 106, 108 in the embodiment of Fig. 4.

Compensator 406, 408 partly or fully compensates a gain influence of the first replacement LPC in the first gain information and compensates a gain influence of the second replacement LPC representation using the second gain information.

In an embodiment, the calculator 600 is configured to calculate a last good power information related to a last good LPC representation before a start of the error concealment. Furthermore, the gain calculator 600 calculates a first power information for the first replacement LPC representation, a second power information for the second LPC representation, the first gain value using the last good power information and the first power information, and a second gain value using the last good power information and the second power information. Then, the compensation is performed in the compensator 406, 408 using the first gain value and using the second gain value. Depending on the information, however, the calculation of the last good power information can also be performed, as illustrated in the Fig. 6 embodiment, by the compensator directly. However, due to the fact that the calculation of the last good power information is basically performed in the same way as the first gain value for the first replacement representation

and the second gain value for the second replacement LPC representation, it is preferred to perform the calculation of all gain values in the gain calculator 600 as illustrated by the input 601.

In particular, the gain calculator 600 is configured to calculate from the last good LPC representation or the first and second LPC replacement representations an impulse response and to then calculate an rms (root mean square) value from the impulse response to obtain the correspondent power information in the gain compensation, each excitation vector is - after being gained by the corresponding codebook gain - again amplified by the gains: g_A or g_B. These gains are determined by calculating the impulse response of the currently used LPC and then calculating the rms:

$$rms_{new} = \sqrt{\sum_{t=0ms}^{5ms} imp_resp^2(t)}$$

The result is then compared to the rms of the last correctly received LPC and the quotient is used as gain factor in order to compensate for energy increase/loss of LPC interpolation:

$$g = \frac{rms_{old}}{rms_{new}}$$

This procedure can be seen as a kind of normalization. It compensates the gain, which is caused by LPC interpolation.

Subsequently, Figs. 7a and 7b are discussed in more detail to illustrate the apparatus for generating an error concealment signal or the gain calculator 600 or the compensator 406, 408 calculates the last good power information as indicated at 700 in Fig. 7a. Furthermore, the gain calculator 600 calculates the first and second power information for the first and second LPC replacement representation as indicated at 702. Then, as illustrated by 704, the first and the second gain values are calculated preferably by the gain calculator 600. Then, the codebook information or the weighted codebook information or the LPC synthesis output is compensated using these gain values as illustrated at 706. This compensation is preferably done by the amplifiers 406, 408.

To this end, several steps are performed in an preferred embodiment as illustrated in Fig. 7b. In step 710, an LPC representation, such as the first or second replacement LPC representation or the last good LPC representation is provided. In step 712 the codebook gains are applied to the codebook information/output as indicated by block 402, 404. Furthermore, in step 716, impulse responses are calculated from the corresponding LPC representations. Then, in step 718, an rms value is calculated for each impulse response and in block 720 the corresponding gain is calculated using an old rms value and a new rms value and this calculation is preferably done by dividing the old rms value by the new rms value. Finally, the result of block 720 is used to compensate the result of step 712 in order to finally obtained the compensated results as indicated at step 714.

Subsequently, a further aspect is discussed, i.e. an implementation for an apparatus for generating an error concealment signal which ha the LPC representation generator 100 generating only a single replacement LPC representation, such as for the situation illustrated in Fig. 8. In contrast to Fig. 8, however, the embodiment illustrating a further aspect in Fig. 9 comprises the gain calculator 600 and the compensator 406, 408. Thus, any gain influence by the replacement LPC representation generated by the LPC representation generator is compensated for. In particular, this gain compensation can be performed on the input side of the LPC synthesizer as illustrated in Fig. 9 by compensator 406, 408n or can be alternatively performed to the output of the LPC synthesizer as illustrated by the compensator 900 in order to finally obtain the error concealment signal. Thus, the compensator 406, 408, 900 is configured for weighting the codebook information or an LPC synthesis output signal provided by the LPC synthesizer 106, 108.

The other procedures for the LPC representation generator, the gain calculator, the compensator and the LPC synthesizer can be performed in the same way as discussed in the context of Figs. 1a to 8.

As has been outlined in the context of Fig. 4, the amplifier 402 and the amplifier 406 perform two weighting operations in series to each other, particularly in the case where not the sum of the multiplier output 402, 404 is fed back into the adaptive codebook, but where only the adaptive codebook output is fed back, i.e. when the switch 405 is in the illustrated position or the amplifier 404 and the amplifier 408 perform two weighting operations in series. In an embodiment, illustrated in Fig. 10, these two weighting operations can be performed in a single operation. To this end, the gain calculator 600 provides its output g_p or g_c to a single value calculator 1002. Furthermore, a codebook

gain generator 1000 is implemented in order to generate a concealment codebook gain as known in the art. The single value calculator 1002 then preferably calculators a product between g_p and g_A in order to obtain the single value. Furthermore, for the second branch, the single value calculator 1002 calculates a product between g_A or g_B in order to provide the single value for the lower branch in Fig. 4. A further procedure can be performed for the third branch having amplifiers 414, 409 of Fig. 5.

Then a manipulator 1004 is provided which together performs the operations of for example amplifiers 402, 406 to the codebook information of a single codebook or to the codebook information of two or more codebooks in order to finally obtain a manipulated signal such as a codebook signal or a concealment signal, depending on whether the manipulator 1004 is located before the LPC synthesizer in Fig. 9 or subsequent to the LPC synthesizer of Fig. 9. Fig. 11 illustrates a third aspect, in which the LPC representation generator 100, the LPC synthesizer 106, 108 and the additional noise estimator 206, which has already been discussed in the context of Fig. 2, are provided. The LPC synthesizer 106, 108 receives codebook information and a replacement LPC representation. The LPC representation is generated by the LPC representation generator using the noise estimate from the noise estimator 206, and the noise estimator 206 operates by determining the noise estimate from the last good frames. Thus, the noise estimate depends on the last good audio frames and the noise estimate is estimated during a reception of good audio frames, i.e. in the normal decoding mode indicated by "0" on the control line of Fig. 2 and this noise estimate generated during the normal decoding mode is then applied in the concealment mode as illustrated by the connection of blocks 206 and 204 in Fig. 2.

The noise estimator is configured to process a spectral representation of a past decoded signal to provide a noise spectral representation and to convert the noise spectral representation into a noise LPC representation, where the noise LPC representation is the same kind of an LPC representation as the replacement LPC representation. Thus, when the replacement LPC representation is in the ISF-domain representation or an ISF vector, then the noise LPC representation additionally is an ISF vector or ISF representation.

Furthermore, the noise estimator 206 is configured to apply a minimum statistics approach with optimal smoothing to a past decoded signal to derive the noise estimate. For this procedure, it is preferred to perform the procedure illustrated in [3]. However, other noise estimation procedures relying on, for example, suppression of tonal parts compared to

non-tonal parts in a spectrum in order to filter out the background noise or noise in an audio signal can be applied as well for obtaining the target spectral shape or noise spectral estimate.

Thus, in one embodiment, a spectral noise estimate is derived from a past decoded signal and the spectral noise estimate is then converted into an LPC representation and then into an ISF domain to obtain the final noise estimate or target spectral shape.

Fig. 12a illustrates a preferred embodiment. In step 1200, the past decoded signal is obtained, as for example illustrated in Fig. 2 by the feedback loop 208. In step 1202, a spectral representation, such as a Fast Fourier transform (FFT) representation is calculated. Then, in step 1204 a target spectral shape is derived such as by the minimum statistics approach with optimal smoothing or by any other noise estimator processing. Then, the target spectral shape is converted into an LPC representation as indicated by block 1206 and finally the LPC representation is converted to an ISF factor as outlined by block 1208 in order to finally obtain the target spectral shape in the ISF domain which can then be directly used by the LPC representation generator for generating a replacement LPC representation. In the equations of this application, the target spectral shape in the ISF domain is indicated as "ISF^{cng}".

In a preferred embodiment illustrated in Fig. 12b, the target spectral shape is derived for example by a minimum statistics approach and optimal smoothing. Then, in step 1212, a time domain representation is calculated by applying an inverse FFT, for example, to the target spectral shape. Then, LPC coefficients are calculated by using Levinson-Durbin recursion. However, the LPC coefficients calculation of block 1214 can also be performed by any other procedure apart from the mentioned Levinson-Durbin recursion. Then, in step 1216, the final ISF factor is calculated to obtain the noise estimate ISF^{cng} to be used by the LPC representation generator 100.

Subsequently, Fig. 13 is discussed for illustrating the usage of the noise estimate in the context of the calculation of a single LPC replacement representation 1308 for the procedure, for example, illustrated in Fig. 8 or for calculating individual LPC representations for individual codebooks as indicated by block 1310 for the embodiment illustrated in Fig. 1.

In step 1300, a mean value of two or three last good frames is calculated. In step 1302, the last good frame LPC representation is provided. Furthermore, in step 1304, a fading factor is provided which can be controlled, for example, by a separate signal analyzer which can be, for example, included in the error concealment controller 200 of Fig. 2. Then, in step 1306, a noise estimate is calculated and the procedure in step 1306 can be performed by any of the procedures illustrated in Figs. 12a, 12b.

In the context of calculating a single LPC replacement representation, the outputs of blocks 1300, 1304, 1306 are provided to the calculator 1308. Then, a single replacement LPC representation is calculated in such a way that subsequent to a certain number of lost or missing or erroneous frames/packets, the fading over to the noise estimate LPC representation is obtained.

However, individual LPC representations for an individual codebook, such as for the adaptive codebook and the fixed codebook, are calculated as indicated at block 1310, then the procedure as discussed before for calculating ISF_A^{-1} (LPC A) on the hand and the calculation of ISF_B^{-1} (LPC B) is performed.

Although the present invention has been described in the context of block diagrams where the blocks represent actual or logical hardware components, the present invention can also be implemented by a computer-implemented method. In the latter case, the blocks represent corresponding method steps where these steps stand for the functionalities performed by corresponding logical or physical hardware blocks.

Although some aspects have been described in the context of an apparatus, it is clear that these aspects also represent a description of the corresponding method, where a block or device corresponds to a method step or a feature of a method step. Analogously, aspects described in the context of a method step also represent a description of a corresponding block or item or feature of a corresponding apparatus. Some or all of the method steps may be executed by (or using) a hardware apparatus, like for example, a microprocessor, a programmable computer or an electronic circuit. In some embodiments, some one or more of the most important method steps may be executed by such an apparatus.

Depending on certain implementation requirements, embodiments of the invention can be implemented in hardware or in software. The implementation can be performed using a digital storage medium, for example a floppy disc, a DVD, a Blu-Ray, a CD, a ROM, a PROM, and EPROM, an EEPROM or a FLASH memory, having electronically readable

control signals stored thereon, which cooperate (or are capable of cooperating) with a programmable computer system such that the respective method is performed. Therefore, the digital storage medium may be computer readable.

Some embodiments according to the invention comprise a data carrier having electronically readable control signals, which are capable of cooperating with a programmable computer system, such that one of the methods described herein is performed.

Generally, embodiments of the present invention can be implemented as a computer program product with a program code, the program code being operative for performing one of the methods when the computer program product runs on a computer. The program code may, for example, be stored on a machine readable carrier.

Other embodiments comprise the computer program for performing one of the methods described herein, stored on a machine readable carrier.

In other words, an embodiment of the inventive method is, therefore, a computer program having a program code for performing one of the methods described herein, when the computer program runs on a computer.

A further embodiment of the inventive method is, therefore, a data carrier (or a non-transitory storage medium such as a digital storage medium, or a computer-readable medium) comprising, recorded thereon, the computer program for performing one of the methods described herein. The data carrier, the digital storage medium or the recorded medium are typically tangible and/or non-transitory.

A further embodiment of the invention method is, therefore, a data stream or a sequence of signals representing the computer program for performing one of the methods described herein. The data stream or the sequence of signals may, for example, be configured to be transferred via a data communication connection, for example, via the internet.

A further embodiment comprises a processing means, for example, a computer or a programmable logic device, configured to, or adapted to, perform one of the methods described herein.

A further embodiment comprises a computer having installed thereon the computer program for performing one of the methods described herein.

A further embodiment according to the invention comprises an apparatus or a system configured to transfer (for example, electronically or optically) a computer program for performing one of the methods described herein to a receiver. The receiver may, for example, be a computer, a mobile device, a memory device or the like. The apparatus or system may, for example, comprise a file server for transferring the computer program to the receiver .

In some embodiments, a programmable logic device (for example, a field programmable gate array) may be used to perform some or all of the functionalities of the methods described herein. In some embodiments, a field programmable gate array may cooperate with a microprocessor in order to perform one of the methods described herein. Generally, the methods are preferably performed by any hardware apparatus.

The above described embodiments are merely illustrative for the principles of the present invention. It is understood that modifications and variations of the arrangements and the details described herein will be apparent to others skilled in the art. It is the intent, therefore, to be limited only by the scope of the impending patent claims and not by the specific details presented by way of description and explanation of the embodiments herein.

References

[1] ITU-T G.718 Recommendation, 2006

[2] Kazuhiro Kondo, Kiyoshi Nakagawa, „A Packet Loss Concealment Method Using Recursive Linear Prediction" Department of Electrical Engineering, Yamagata University, Japan.

[3] R. Martin, Noise Power Spectral Density Estimation Based on Optimal Smoothing and Minimum Statistics, IEEE Transactions on speech and audio processing, vol. 9, no. 5, July 2001

[4] Ralf Geiger et. al., Patent application US20110173011 A1, Audio Encoder and Decoder for Encoding and Decoding Frames of a Sampled Audio Signal

[5] 3GPP TS 26.190; Transcoding functions; - 3GPP technical specification

Claims

1. Apparatus for generating an error concealment signal, comprising:

an LPC (linear prediction coding) representation generator (100) for generating a replacement LPC representation;

a gain calculator (600) for calculating a gain information from the LPC representations;

a compensator (406, 408) for compensating a gain influence of the replacement LPC representation using the gain information; and

an LPC synthesizer (106, 108) for filtering codebook information using the replacement LPC representation to obtain the error concealment signal,

wherein the compensator (406, 408, 900) is configured for weighting the codebook information or an LPC synthesis output signal.

2. Apparatus of claim 1,

wherein the gain calculator (600) is configured to calculate:

a last good frame (700) power information related to a last good LPC representation before a start of the error concealment;

a power information from the replacement LPC information (702);

a gain value using the last good power information (704),

wherein the compensator (406, 408, 900) is configured for compensating using the gain value.

3. Apparatus of claim 2,

wherein the gain calculator (600) is configured to calculate an impulse response (716) of the LPC replacement representation and to calculate an rms value (718) from the impulse response to obtain the power information.

4. Apparatus of any one of the preceding claims,

wherein the gain calculator (600) is configured to calculate the gain based on the following equation:

$$rms_{new} = \sqrt{\sum_{t=0ms}^{T} imp_resp^2(t)}$$

$$g = \frac{rms_{old}}{rms_{new}}$$

wherein rms_{new} is an rms value of the LPC replacement representation, wherein t is a time variable, wherein T is a predetermined time value between 3 and 8 ms or lower than a frame size, wherein imp_resp is an impulse response derived from representation, and wherein rms_{old} is an rms value derived from the last good frame.

5. Apparatus of one of the preceding claims, further comprising:

an adaptive codebook (102) for providing an adaptive codebook information;

a fixed codebook (104) for providing a fixed codebook information;

an adaptive codebook weighter (402) for weighting the adaptive codebook information, a fixed codebook weighter (404) for weighting the fixed codebook information,

wherein the compensator (406, 408) is configured to process an output of the adaptive codebook weighter (402) or the fixed codebook weighter (404) or a sum of outputs of the adaptive codebook weighter and the fixed codebook weighter.

6. Apparatus of claim 5,

wherein the adaptive codebook weighter (402) and the compensator (406) or the fixed codebook weighter (404) and the compensator (408) are implemented by a manipulator (1004) for manipulating a signal using a single manipulation information, the single manipulation information being derived from a codebook weighter information and a compensator information.

7. Apparatus of claim 5 or 6,

wherein the codebook weighters are configured to apply corresponding replacement codebook gains derived from corresponding last good received codebook gains.

8. Apparatus of one of the preceding claims,

wherein the LPC representation generator is configured for generating a further replacement LPC representation; and

wherein the LPC synthesizer is configured for filtering a further codebook information using the further replacement LPC representation, and wherein the apparatus further comprises a replacement signal combiner (110) for replacing LPC synthesizer outputs.

9. Apparatus of claim 8, further comprising:

an adaptive codebook (102) for providing the first codebook information; and

a fixed codebook (104) for providing the second codebook information.

10. Apparatus of claim 9,

wherein the fixed codebook (104) is configured to provide a noise signal (112) for the error concealment, and

wherein the adaptive codebook (102) is configured for providing an adaptive codebook content or an adaptive codebook content combined with an earlier fixed codebook content.

11. Apparatus of claim 10,

wherein the LPC representation generator (100) is configured to generate the first replacement LPC representation using one or at least two non-erroneous preceding LPC representations, and

to generate the second replacement LPC representation using a noise estimate and at least one non-erroneous preceding LPC representation.

12. Apparatus of claim 11,

wherein the LPC representation generator (100) is configured to generate the first replacement LPC representation using a mean value of at least two last good frames (130) and a weighted summation of the mean value and the last good frame (136), wherein a first weighting factor of the weighted summation changes over successive erroneous or lost frames,

wherein the LPC coefficient generator is configured to generate the second replacement LPC representation only using a weighted summation (146) of a last good frame (114) and the noise estimate (140), wherein a second weighting factor of the weighted summation changes over successive erroneous or lost frames.

13. Apparatus of claim 11 or 12,

a noise estimator (206) for estimating the noise estimate from one or more preceding good frames (208).

14. Method of generating an error concealment signal, comprising:

generating (100) a replacement LPC representation;

calculating (600) a gain information from the LPC representations;

compensating (406, 408) a gain influence of the replacement LPC representation using the gain information; and

filtering (106, 108) codebook information using the replacement LPC representation to obtain the error concealment signal,

wherein the compensating (406, 408, 900) is configured for weighting the codebook information or an LPC synthesis output signal.

15. Computer program for performing, when running on a computer or a processor, the method of claim 14.

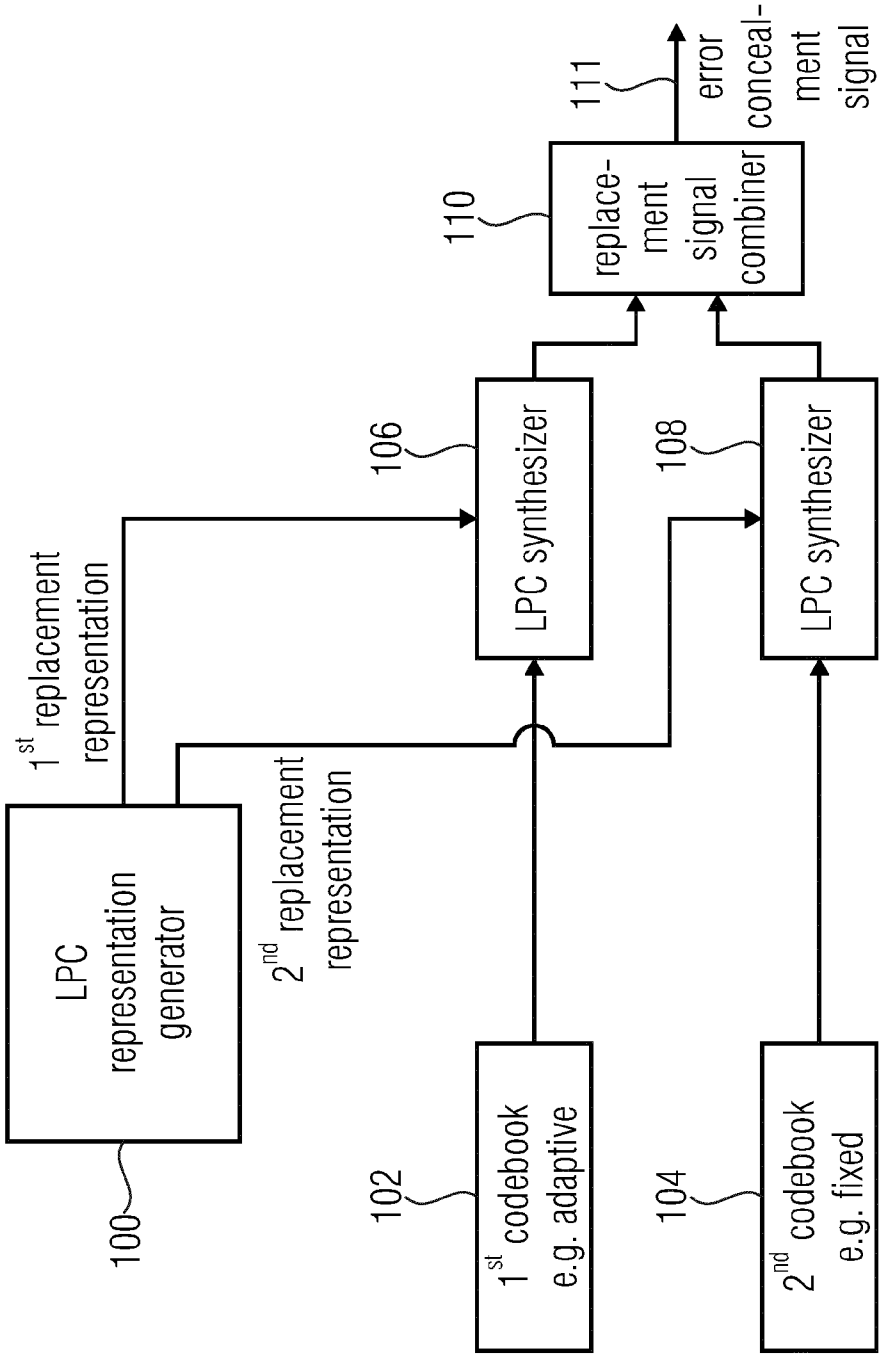

LPC representation generator
100
1st replacement representation
2nd replacement representation
1st codebook e.g. adaptive
102
2nd codebook e.g. fixed
104
LPC synthesizer
106
LPC synthesizer
108
replace-ment signal combiner
110
111
error conceal-ment signal

FIG 1A

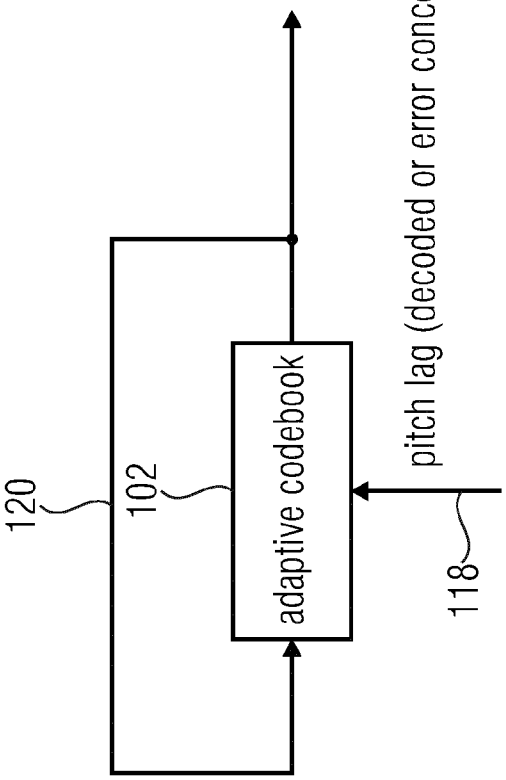

120
102
adaptive codebook
pitch lag (decoded or error concealment value)
118

FIG 1B

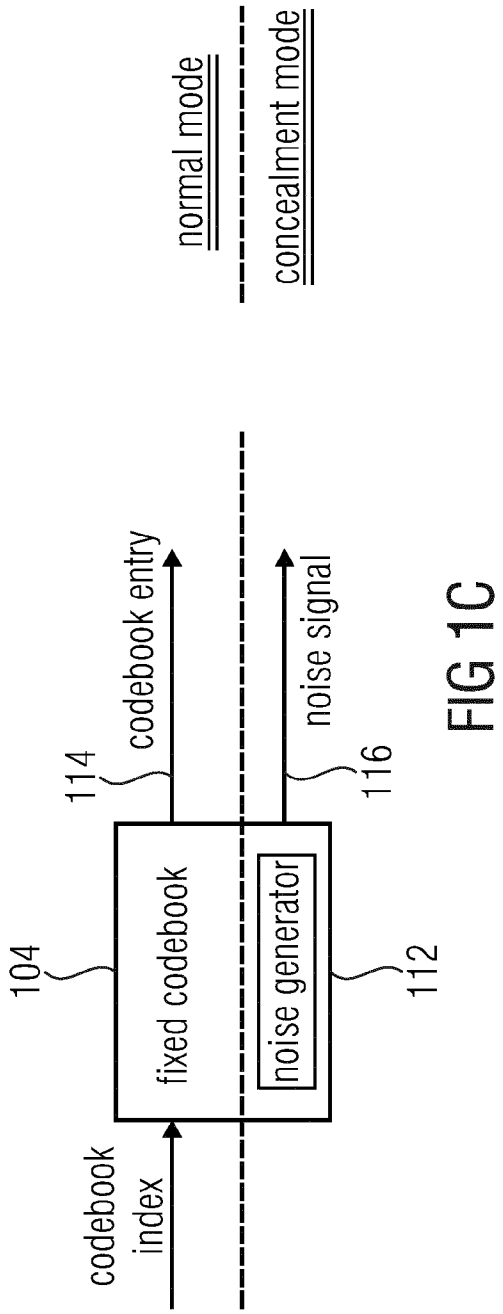

normal mode
concealment mode
114
codebook entry
116
noise signal
104
fixed codebook
noise generator
112
codebook index

FIG 1C

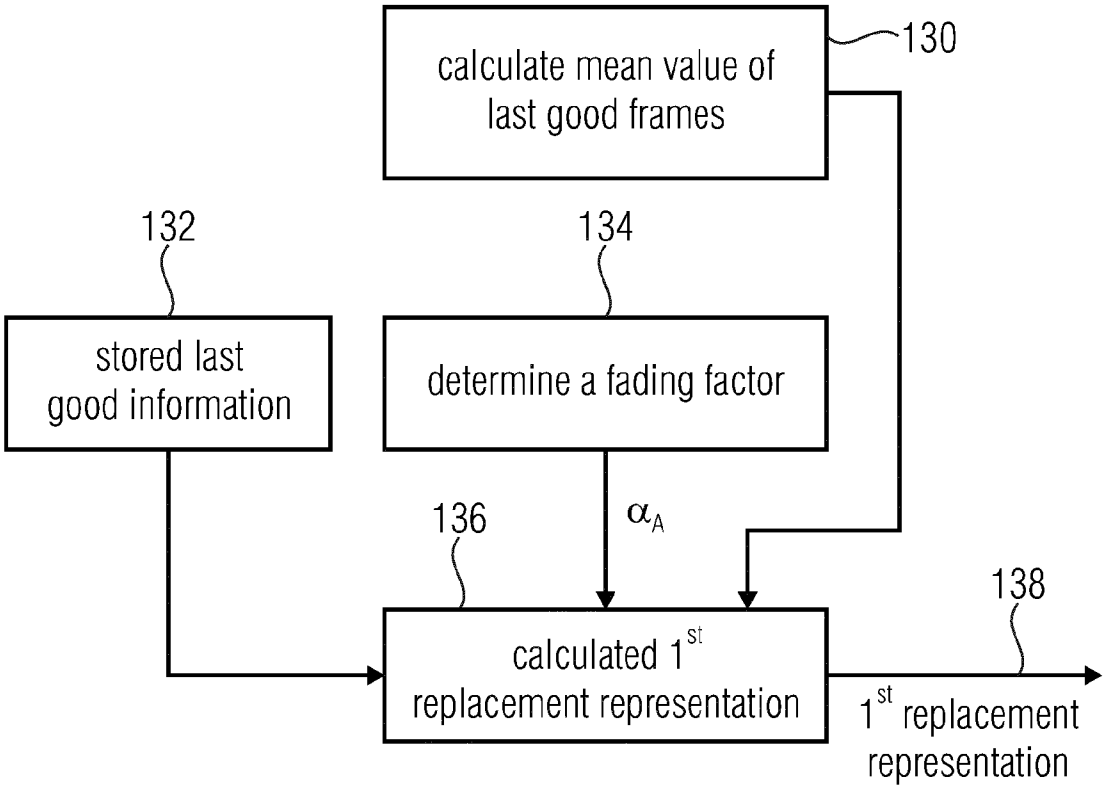

calculate mean value of last good frames
130
132
stored last good information
134
determine a fading factor
136
α_A
138
calculated 1st replacement representation
1st replacement representation

FIG 1D

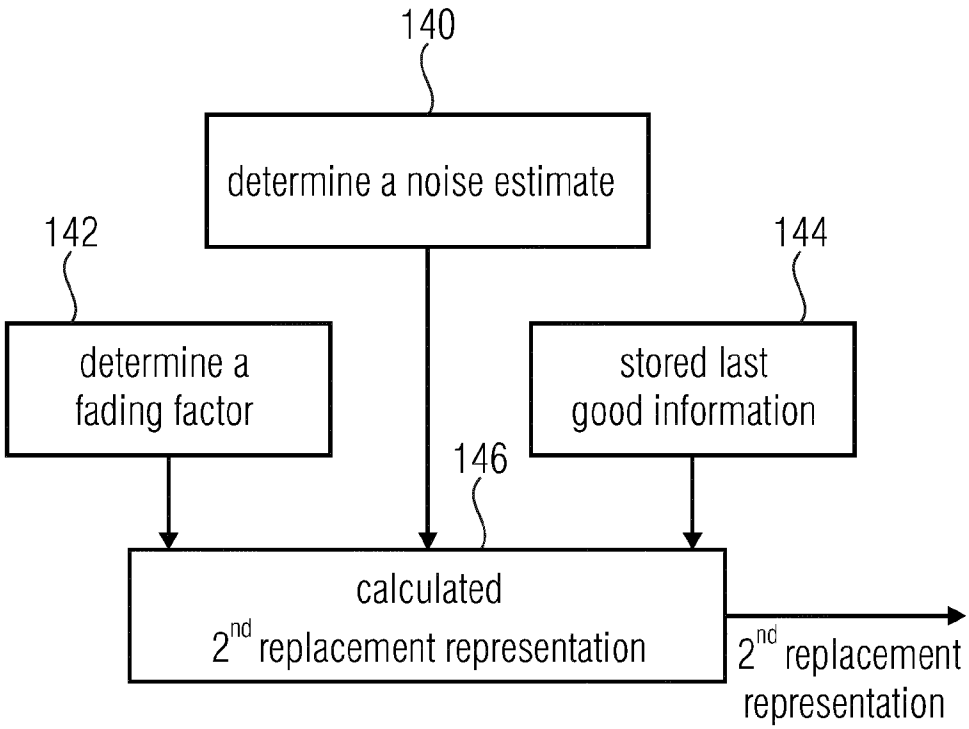

140
determine a noise estimate
142
determine a fading factor
144
stored last good information
146
calculated 2nd replacement representation
2nd replacement representation

FIG 1E

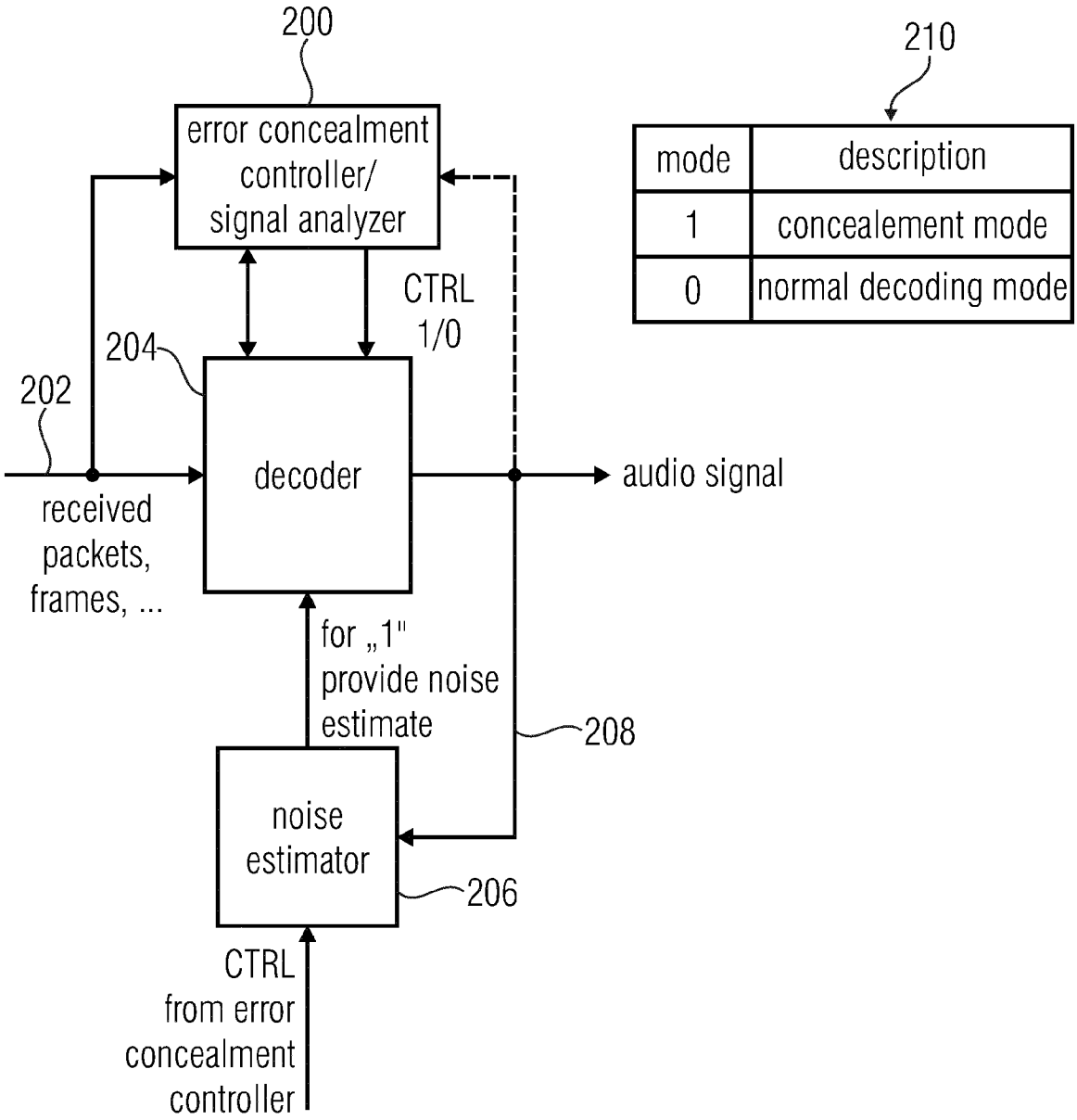

200
error concealment controller/ signal analyzer
CTRL 1/0
204
202
decoder
audio signal
received packets, frames, ...
for „1" provide noise estimate
208
noise estimator
206
CTRL from error concealment controller
210
mode
description
1
concealement mode
0
normal decoding mode

FIG 2

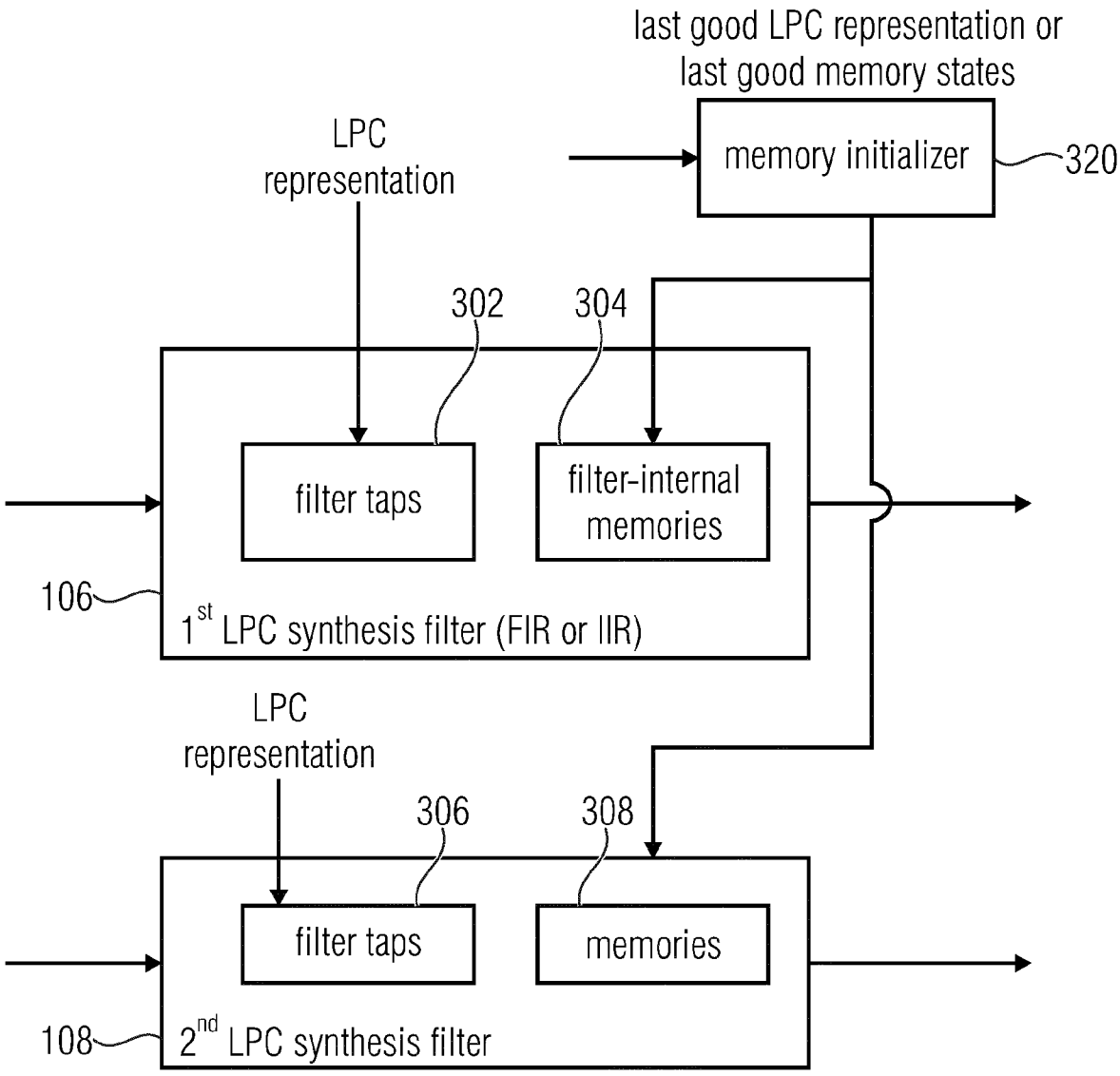

last good LPC representation or
last good memory states
LPC
representation
memory initializer
320
302
304
filter taps
filter-internal
memories
106
1st LPC synthesis filter (FIR or IIR)
LPC
representation
306
308
filter taps
memories
108
2nd LPC synthesis filter

FIG 3

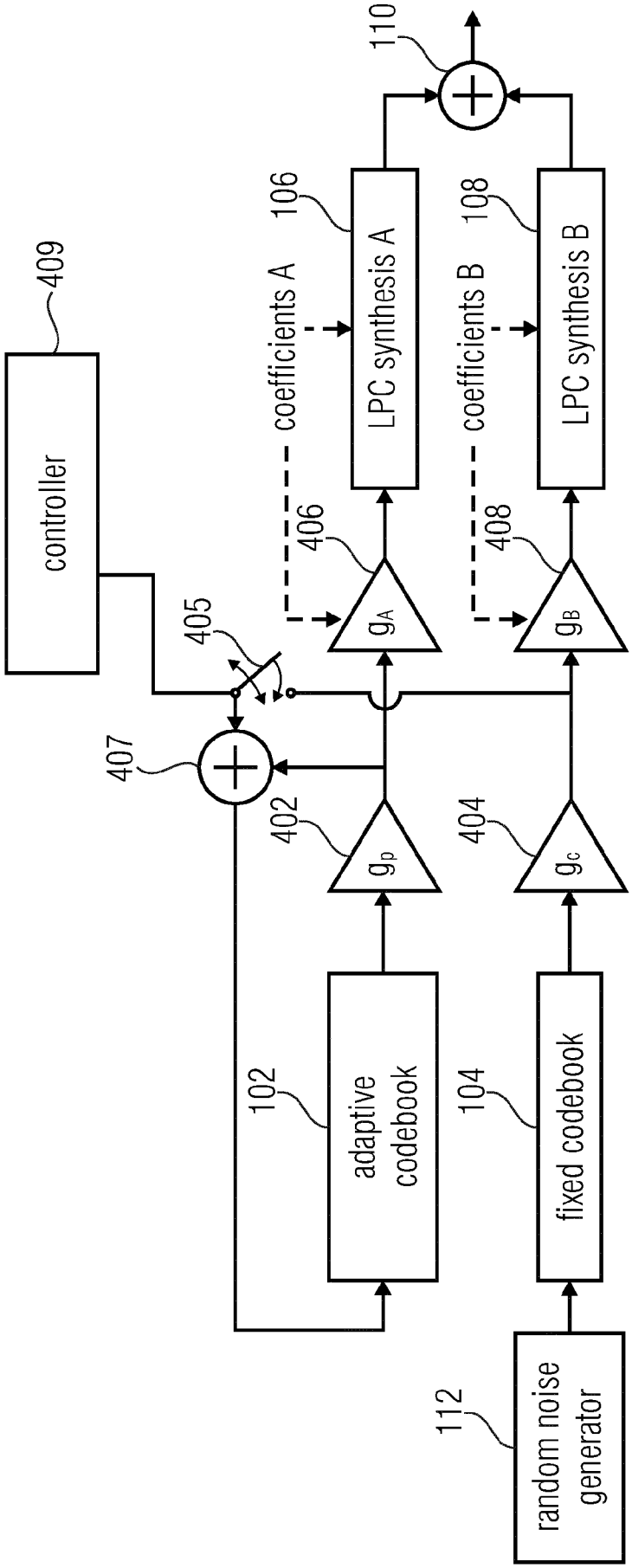

110
106
108
409
coefficients A
coefficients B
LPC synthesis A
LPC synthesis B
controller
406
408
405
g_A
g_B
407
402
404
g_p
g_c
102
104
adaptive codebook
fixed codebook
112
random noise generator

FIG 4

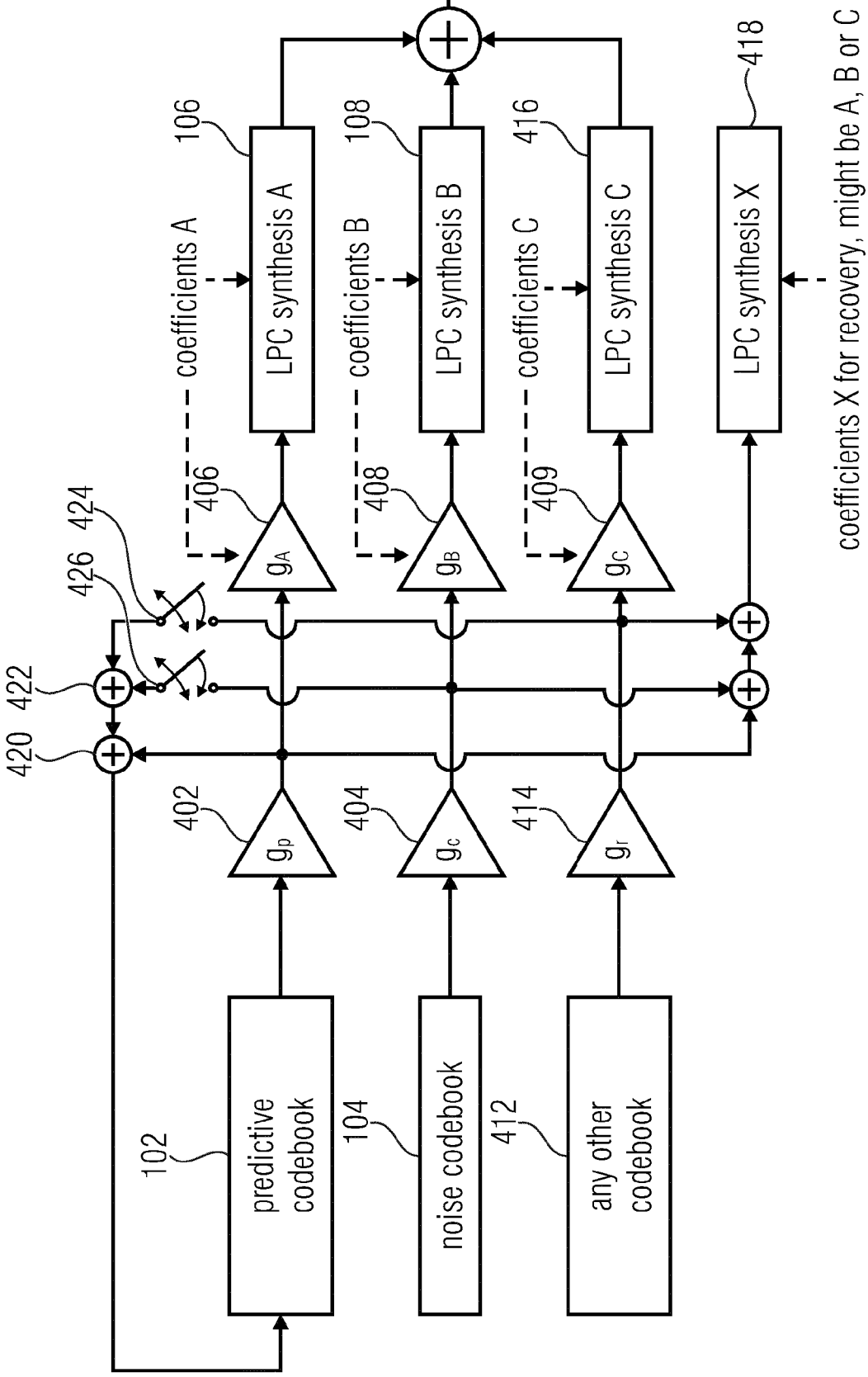

LPC synthesis A
LPC synthesis B
LPC synthesis C
LPC synthesis X
coefficients A
coefficients B
coefficients C
coefficients X for recovery, might be A, B or C
predictive codebook
noise codebook
any other codebook
102
104
106
108
402
404
406
408
409
412
414
416
418
420
422
424
426

FIG 5

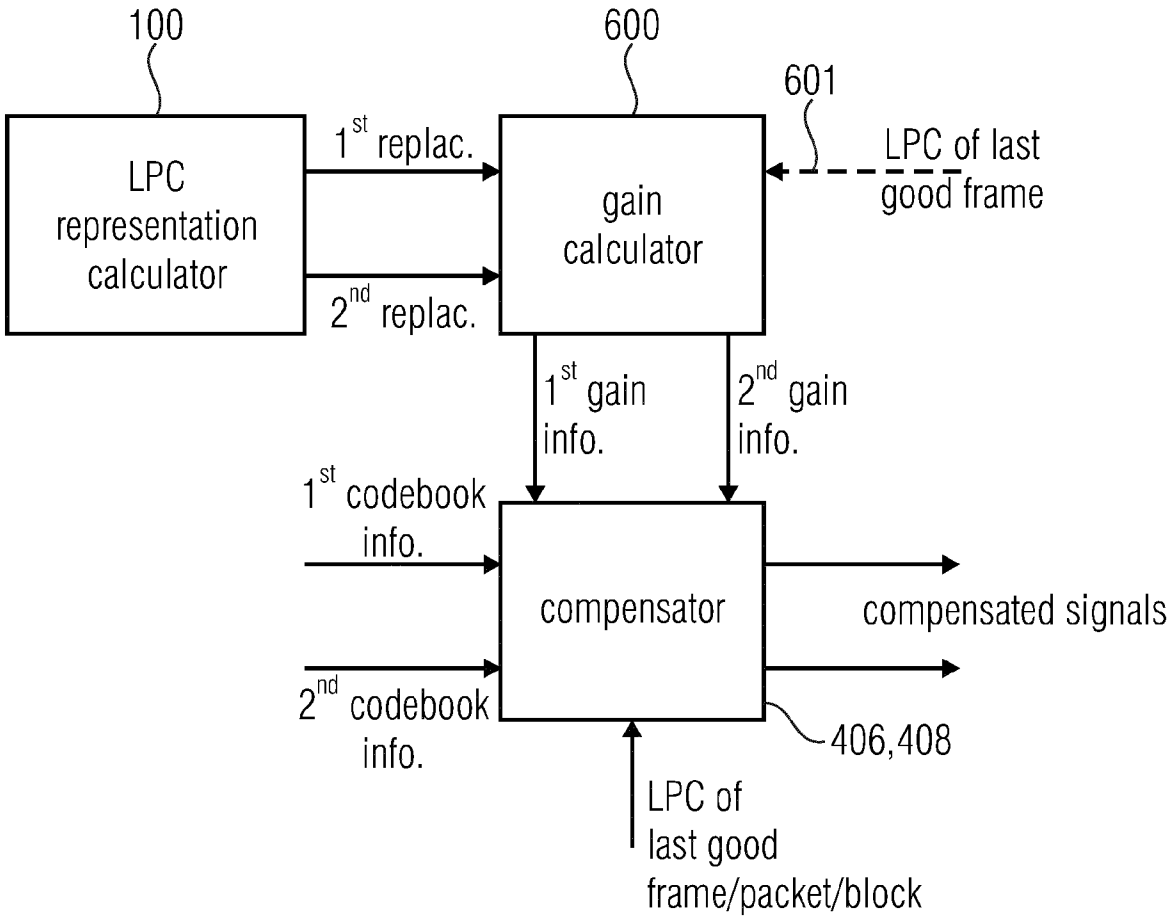

100
600
601
LPC representation calculator
1st replac.
2nd replac.
gain calculator
LPC of last good frame
1st gain info.
2nd gain info.
1st codebook info.
2nd codebook info.
compensator
compensated signals
406,408
LPC of last good frame/packet/block

FIG 6

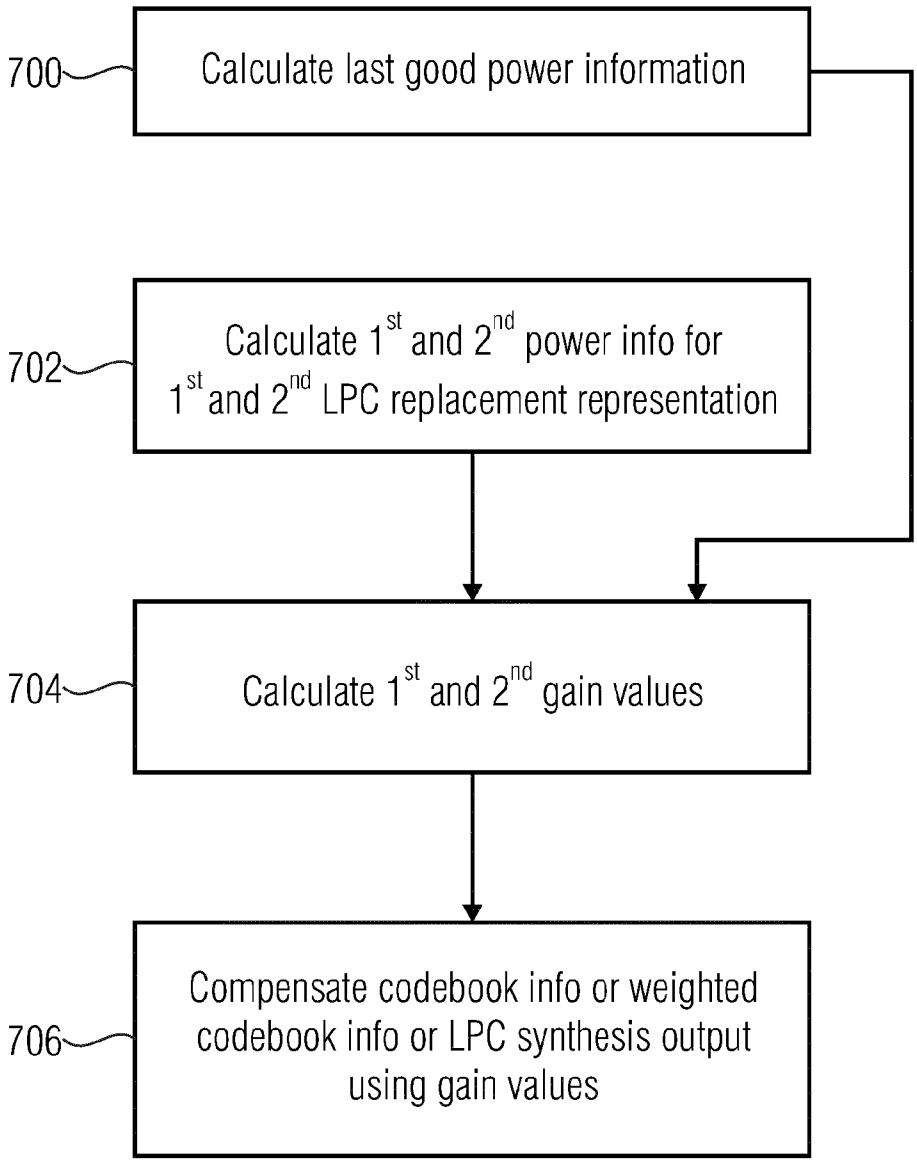

700
Calculate last good power information
702
Calculate 1st and 2nd power info for 1st and 2nd LPC replacement representation
704
Calculate 1st and 2nd gain values
706
Compensate codebook info or weighted codebook info or LPC synthesis output using gain values

FIG 7A

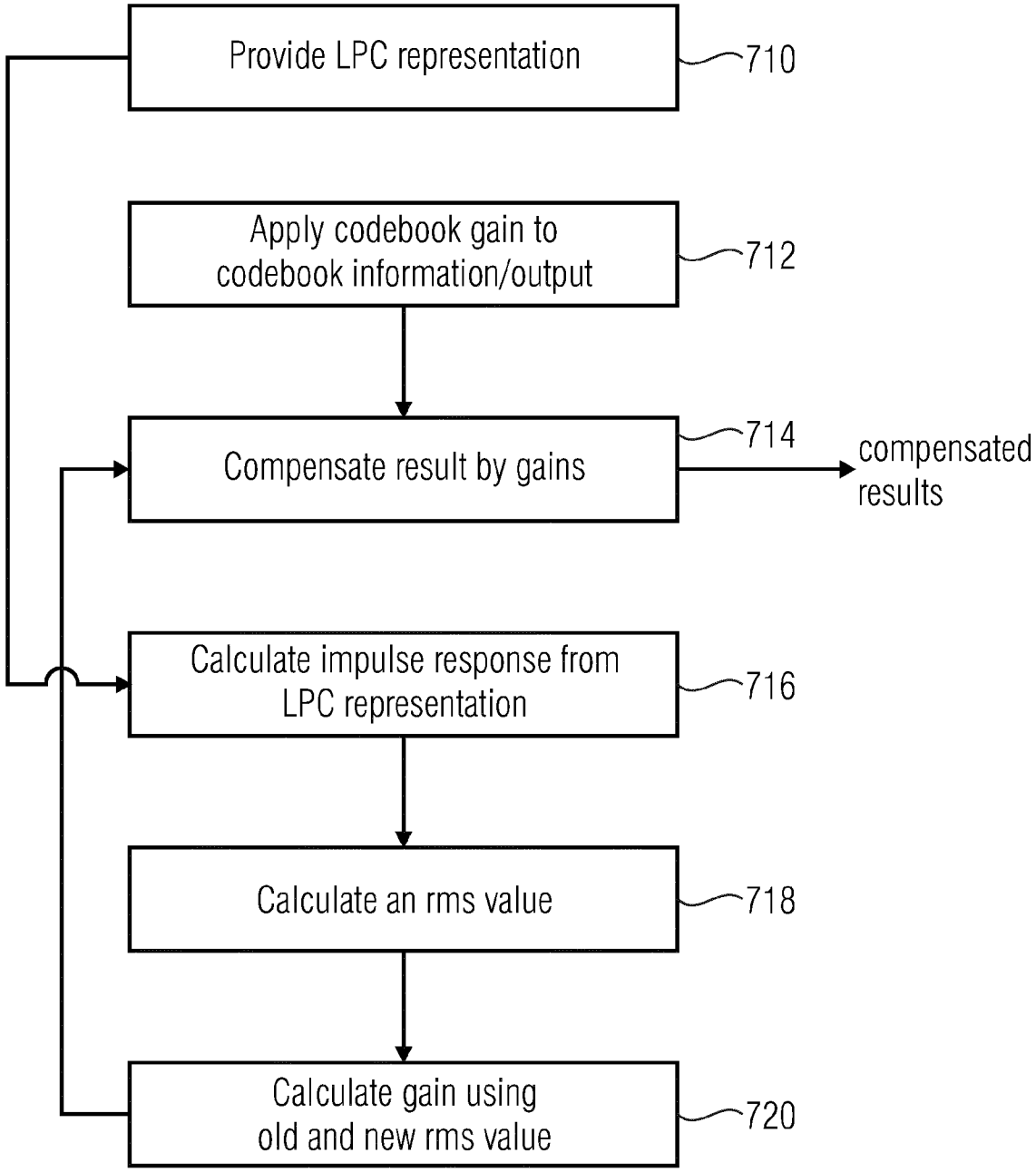

Provide LPC representation
710
Apply codebook gain to codebook information/output
712
Compensate result by gains
714
compensated results
Calculate impulse response from LPC representation
716
Calculate an rms value
718
Calculate gain using old and new rms value
720

FIG 7B

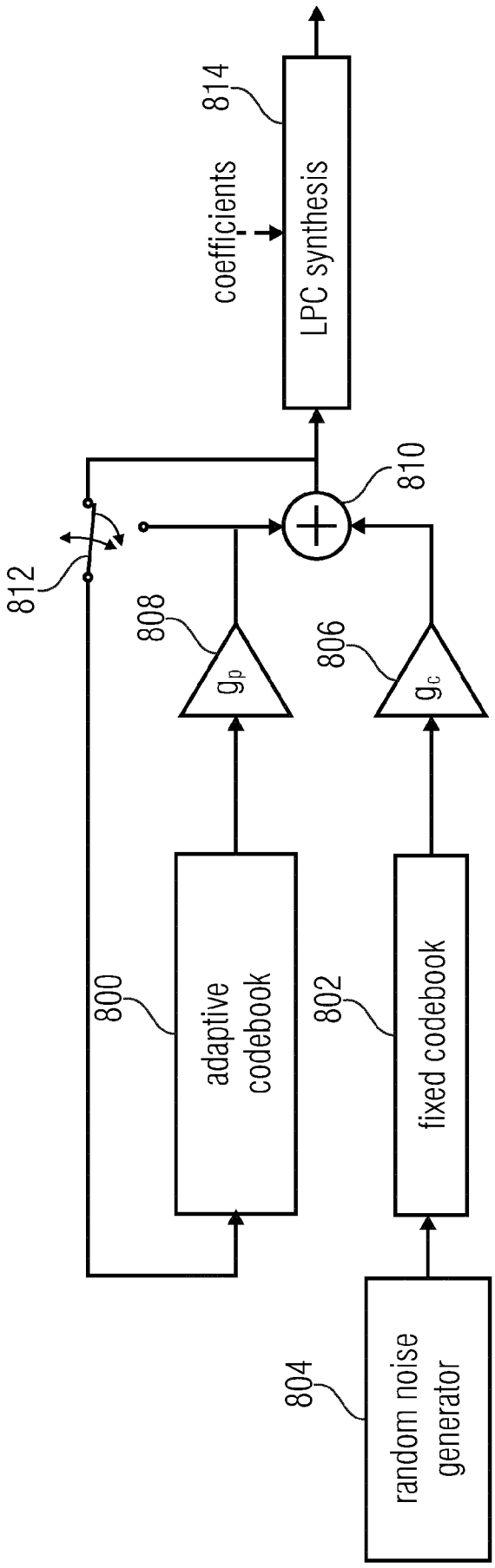


FIG 8
(PRIOR ART)

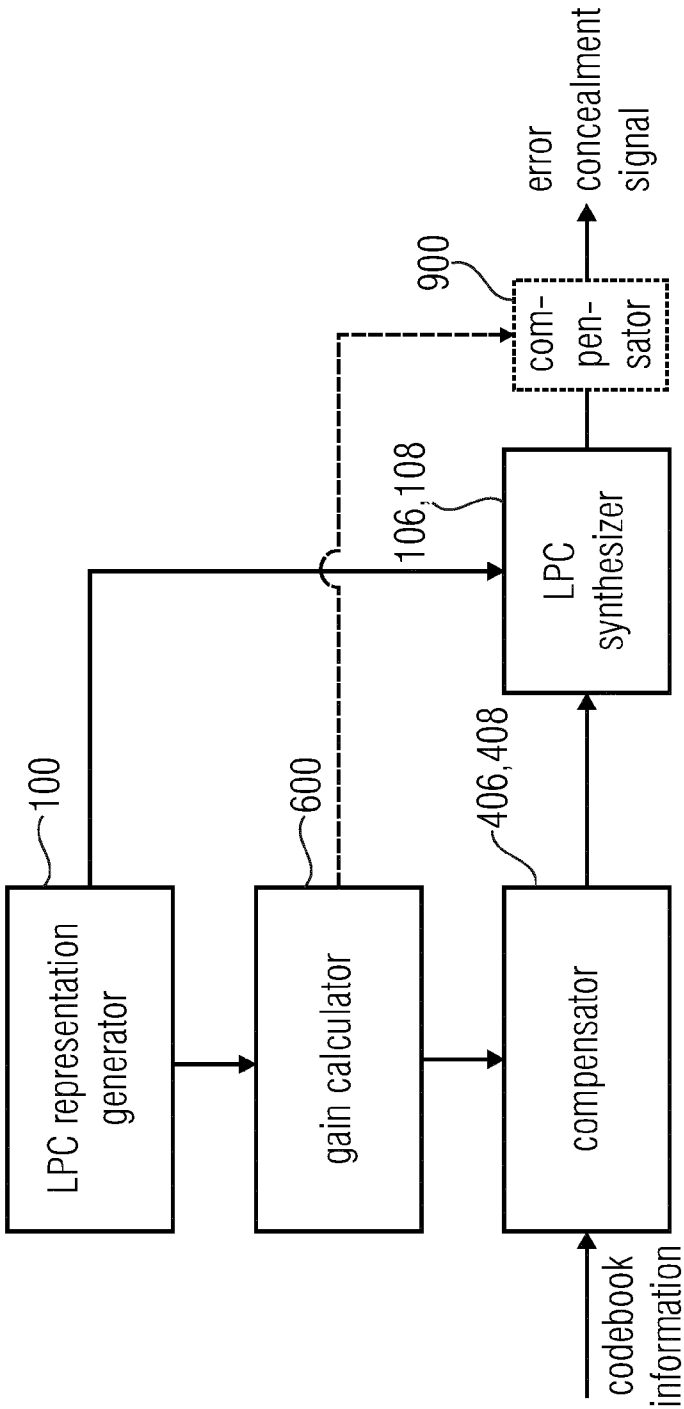

LPC representation generator
100
gain calculator
600
compensator
406,408
LPC synthesizer
106,108
com-pen-sator
900
error concealment signal
codebook information

FIG 9

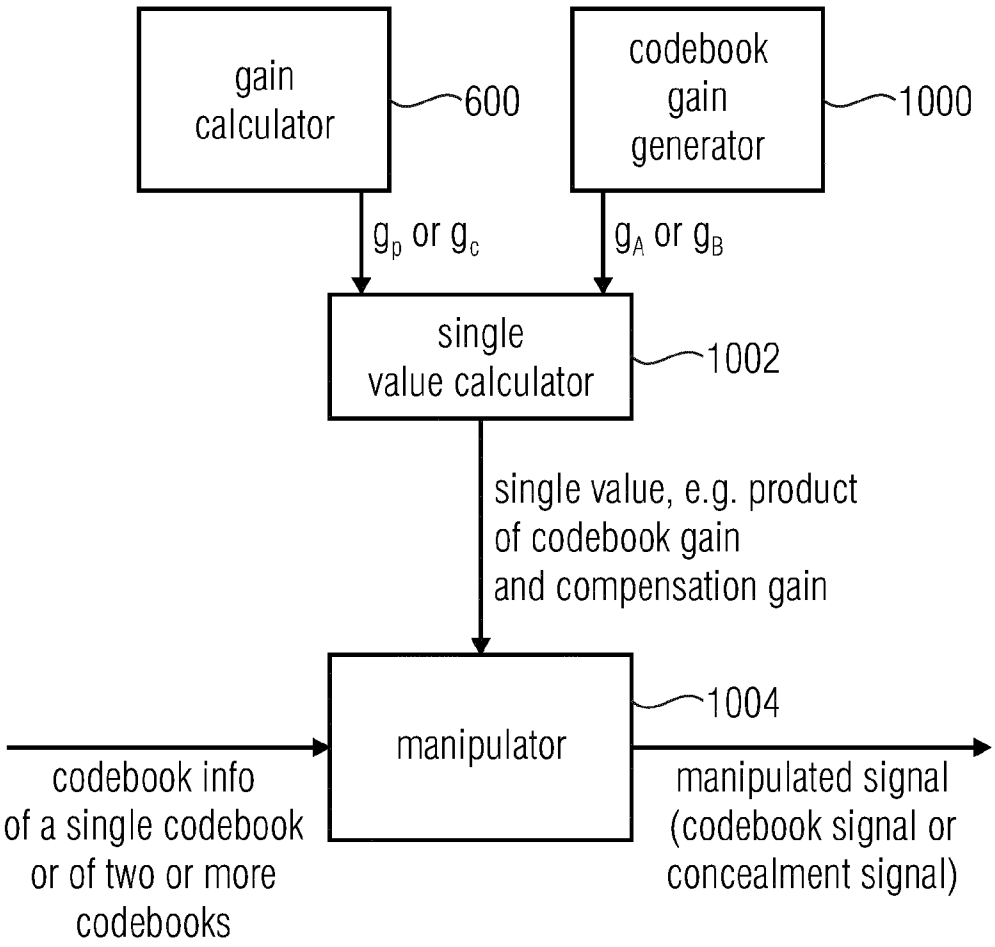

gain calculator
600
codebook gain generator
1000
g_p or g_c
g_A or g_B
single value calculator
1002
single value, e.g. product of codebook gain and compensation gain
manipulator
1004
codebook info of a single codebook or of two or more codebooks
manipulated signal (codebook signal or concealment signal)

FIG 10

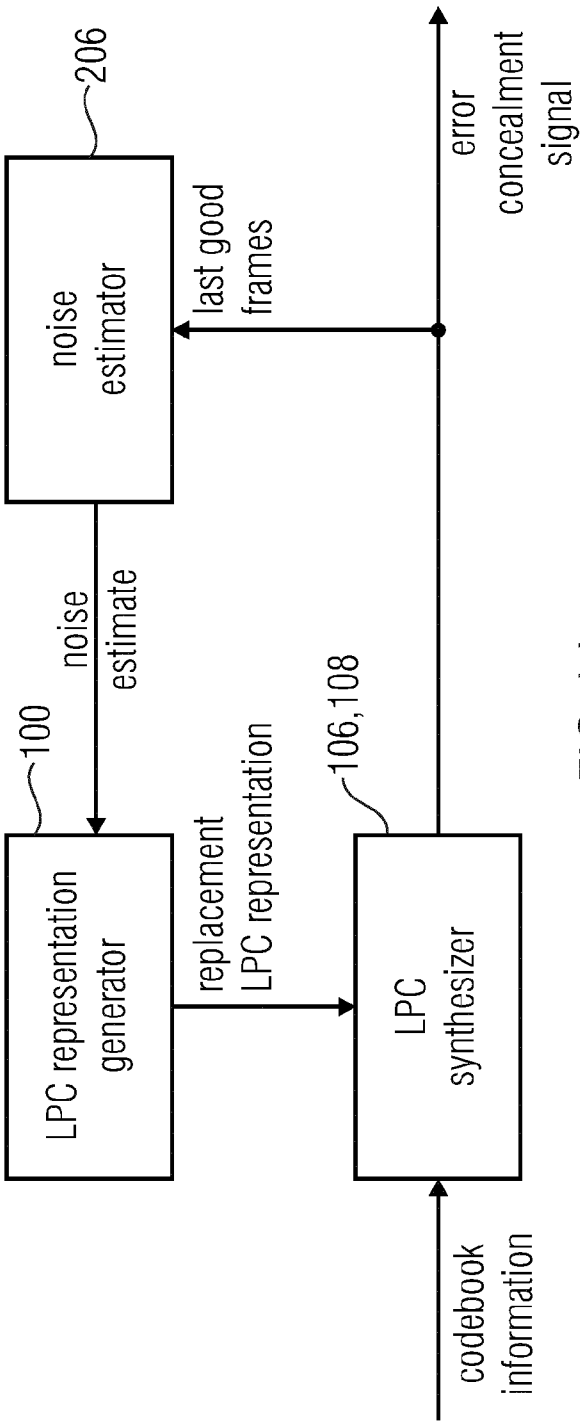

206
noise estimator
last good frames
error concealment signal
noise estimate
100
LPC representation generator
replacement LPC representation
106,108
LPC synthesizer
codebook information

FIG 11

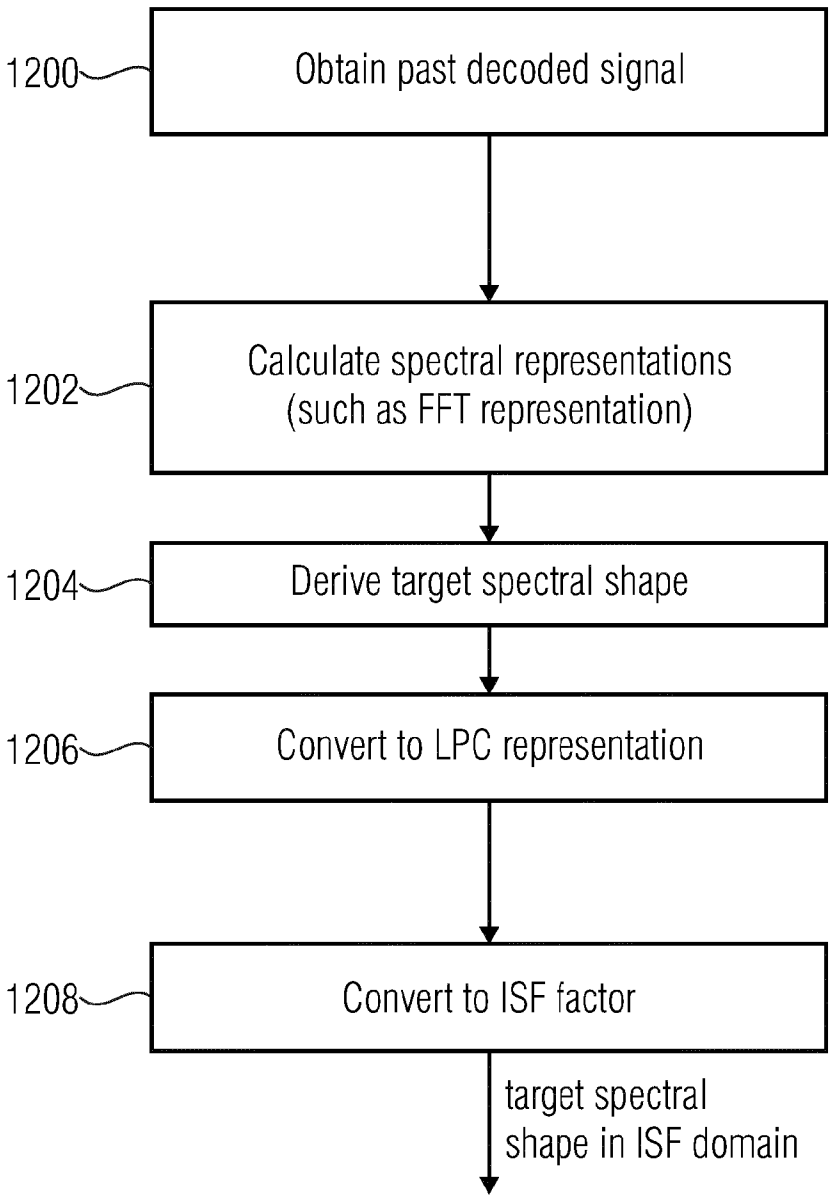

1200
Obtain past decoded signal
1202
Calculate spectral representations (such as FFT representation)
1204
Derive target spectral shape
1206
Convert to LPC representation
1208
Convert to ISF factor
target spectral shape in ISF domain

FIG 12A

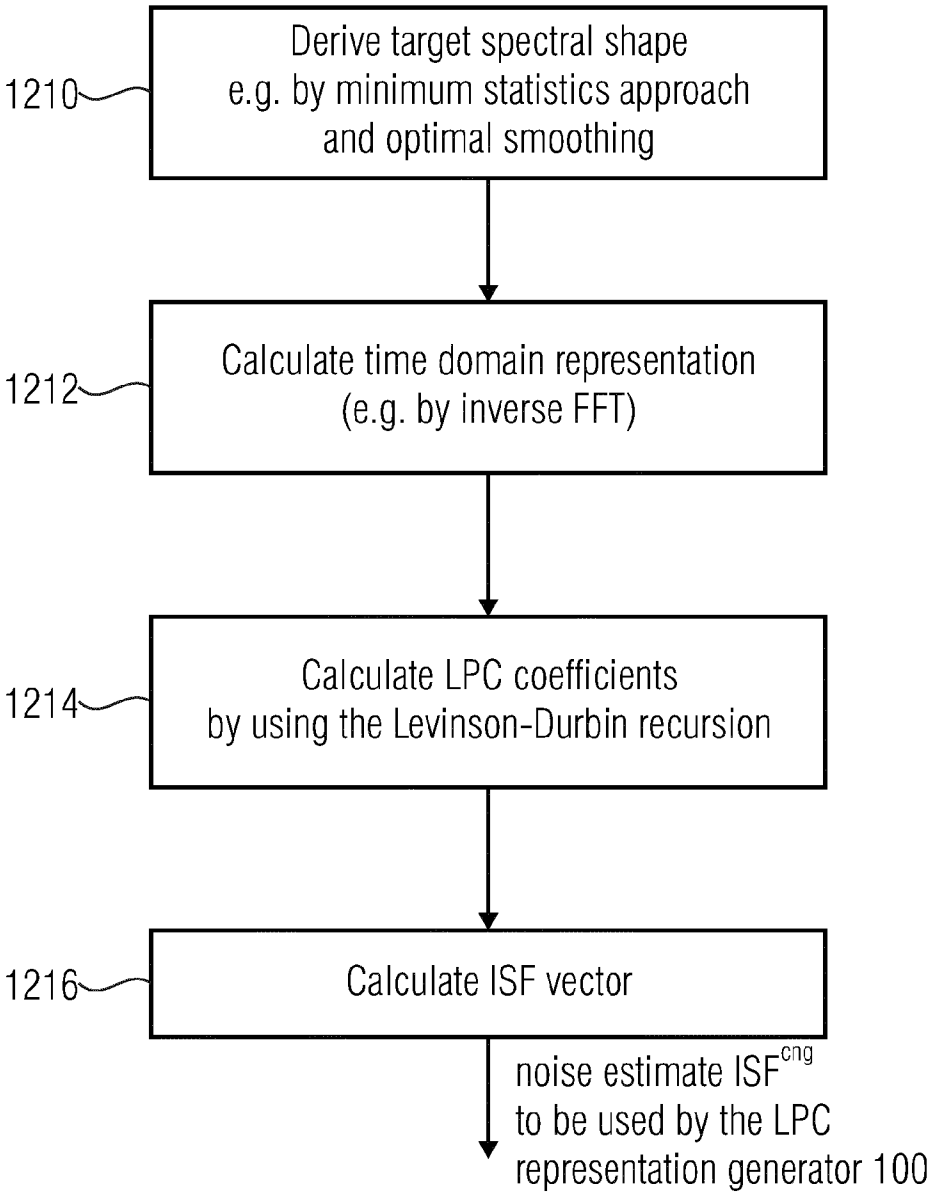

1210
Derive target spectral shape
e.g. by minimum statistics approach
and optimal smoothing
1212
Calculate time domain representation
(e.g. by inverse FFT)
1214
Calculate LPC coefficients
by using the Levinson-Durbin recursion
1216
Calculate ISF vector
noise estimate ISF^cng
to be used by the LPC
representation generator 100

FIG 12B

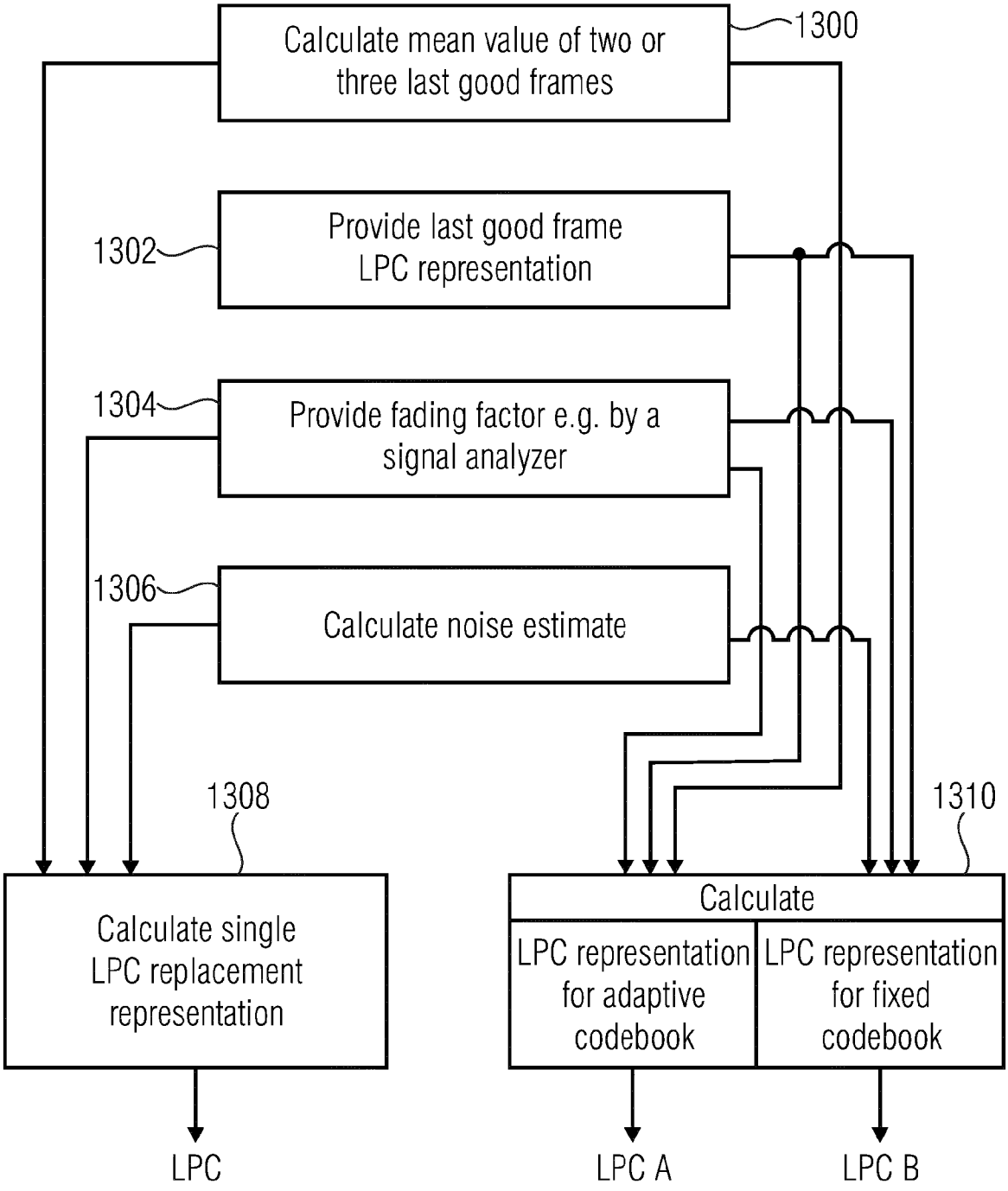

Calculate mean value of two or three last good frames
1300
Provide last good frame LPC representation
1302
Provide fading factor e.g. by a signal analyzer
1304
Calculate noise estimate
1306
1308
1310
Calculate single LPC replacement representation
Calculate
LPC representation for adaptive codebook
LPC representation for fixed codebook
LPC
LPC A
LPC B

FIG 13

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2015/054490

A. CLASSIFICATION OF SUBJECT MATTER
INV. G10L19/005 G10L19/06
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JON GIBBS MOTOROLA UK LTD UNITED KINGDOM: "Draft new ITU-T Recommendation G.VBR-EV "Frame error robust narrowband and wideband embedded variable bit-rate coding of speech and audio from 8-32 kbit/s" (for Consent);TD 534 (PLEN/16)", ITU-T DRAFT ; STUDY PERIOD 2005-2008, INTERNATIONAL TELECOMMUNICATION UNION, GENEVA ; CH, vol. 9/16, 22 April 2008 (2008-04-22), pages 1-243, XP017541194, [retrieved on 2008-09-26] sections 5.11.1.2, 5.12.2 sections 6.3.1, 6.3.2, 6.6 ----- -/--	1-15

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
7 May 2015	19/05/2015

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Tilp, Jan

5

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Juin-Hwey Chen ET AL: "Adaptive postfiltering for quality enhancement of coded speech", IEEE Transactions on Speech and Audio Processing, 1 January 1995 (1995-01-01), pages 59-71, XP055104008, DOI: 10.1109/89.365380 Retrieved from the Internet: URL:http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=365380 sections IV.B, V.A -----	1-4,14, 15