Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2729665 C 2016/11/22

(11)(21) 2 729 665

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(86) Date de dépôt PCT/PCT Filing Date: 2009/07/10
(87) Date publication PCT/PCT Publication Date: 2010/01/14
(45) Date de délivrance/Issue Date: 2016/11/22
(85) Entrée phase nationale/National Entry: 2010/12/30
(86) N° demande PCT/PCT Application No.: CA 2009/000980
(87) N° publication PCT/PCT Publication No.: 2010/003253
(30) Priorités/Priorities: 2008/07/10 (US61/129,669);
2009/01/27 (US61/202,075)

(51) Cl.Int./Int.Cl. *G10L 19/038* (2013.01),
G10L 19/087 (2013.01), *G10L 19/26* (2013.01),
G10L 19/12 (2013.01)

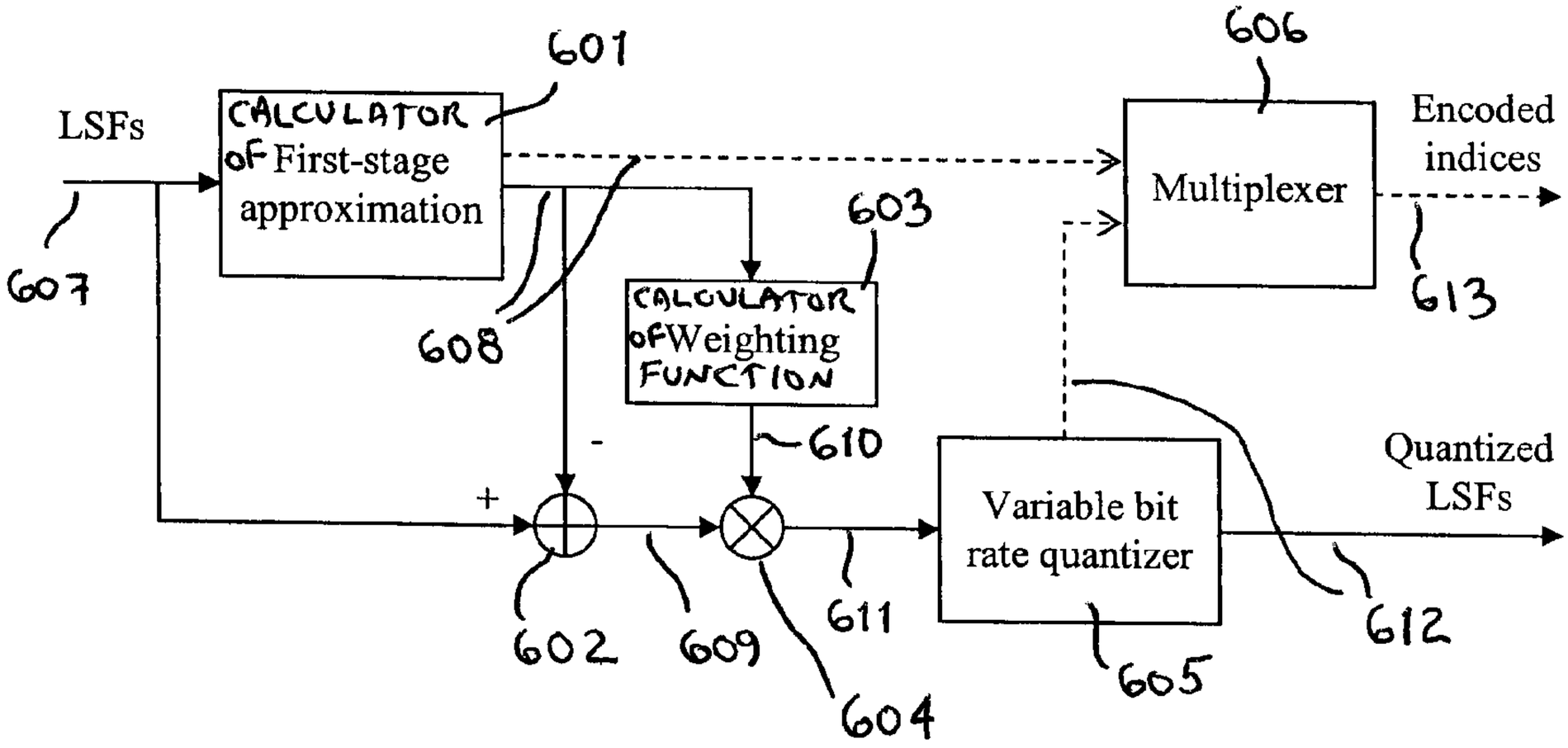
(72) Inventeurs/Inventors:
BESSETTE, BRUNO, CA;
GOURNAY, PHILIPPE, CA;
SALAMI, REDWAN, CA

(73) Propriétaire/Owner:
VOICEAGE CORPORATION, CA

(74) Agent: BCF LLP

(54) Titre : QUANTIFICATION DE FILTRE A CODAGE PREDICTIF LINEAIRE A DEBIT DE BITS VARIABLE ET DISPOSITIF ET PROCEDE DE QUANTIFICATION INVERSE

(54) Title: VARIABLE BIT RATE LPC FILTER QUANTIZING AND INVERSE QUANTIZING DEVICE AND METHOD



(57) Abrégé/Abstract:
A device and a method for quantizing a LPC filter in the form of an input vector in a quantization domain, comprises a calculator of a first-stage approximation of the input vector, a subtractor of the first-stage approximation from the input vector to produce a residual vector, a calculator of a weighting function from the first-stage approximation, a warper of the residual vector with the weighting function, and a quantizer of the weighted residual vector to supply a quantized weighted residual vector. A device and a method for inverse quantizing of a LPC filter, comprises means for receiving coded indices representative of a first-stage approximation of a vector representative of the LPC filter in a quantization domain and of a quantized weighted residual version of the vector, a calculator of an inverse weighting function from the first-stage approximation, an inverse quantizer of the quantized weighted residual version of the vector to produce a weighted residual vector, a multiplier of the weighted residual vector by the inverse weighting function to produce a residual vector, and an adder of the first-stage approximation with the residual vector to produce the vector representative of the LPC filter in the quantization domain.

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

CORRECTED VERSION

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 January 2010 (14.01.2010)

PCT

(10) International Publication Number
WO 2010/003253 A8

(51) International Patent Classification:
G10L 19/12 (2006.01)

(21) International Application Number:
PCT/CA2009/000980

(22) International Filing Date:
10 July 2009 (10.07.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/129,669 10 July 2008 (10.07.2008) US
61/202,075 27 January 2009 (27.01.2009) US

(71) Applicant *(for all designated States except US)*: **VOICEAGE CORPORATION** [CA/CA]; 750 Lucerne Road, Suite 250, Town of Mount Royal, Quebec H3R 2H6 (CA).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **BESSETTE, Bruno** [CA/CA]; 1600 Rue Murillo, Sherbrooke, Québec J1N 4G5 (CA). **GOURNAY, Philippe** [CA/CA]; 3012, rue du Sauvignon, Sherbrooke, Québec J1L 0A2 (CA). **SALAMI, Redwan** [CA/CA]; 4045, Place Albert-Dreux, Saint-Laurent, Québec H4R 2Y3 (CA).

(74) Agents: **PRINCE, Gaétan** et al.; BCF LLP, 1100 René-Lévesque Blvd. West, 25th Floor, Montréal, Québec, H3B 5C9 (CA).

(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, 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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (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, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

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

[Continued on next page]

(54) **Title**: VARIABLE BIT RATE LPC FILTER QUANTIZING AND INVERSE QUANTIZING DEVICE AND METHOD

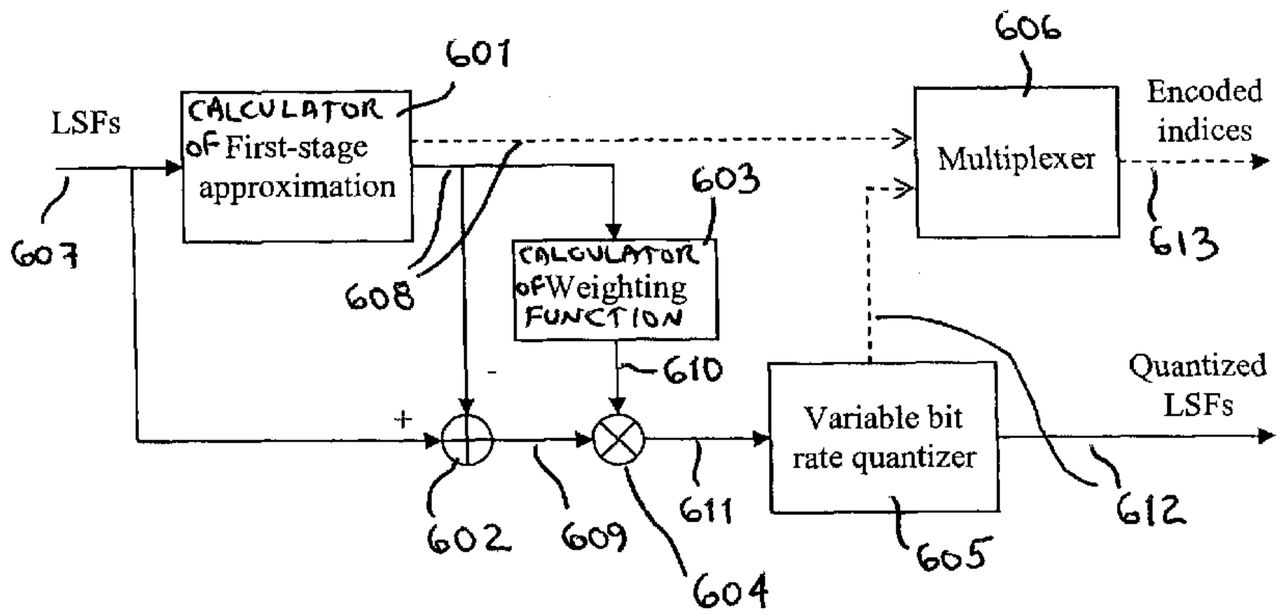


Figure 6

(57) **Abstract**: A device and a method for quantizing a LPC filter in the form of an input vector in a quantization domain, comprises a calculator of a first-stage approximation of the input vector, a subtractor of the first-stage approximation from the input vector to produce a residual vector, a calculator of a weighting function from the first-stage approximation, a warper of the residual vector with the weighting function, and a quantizer of the weighted residual vector to supply a quantized weighted residual vector. A device and a method for inverse quantizing of a LPC filter, comprises means for receiving coded indices representative of a first-stage approximation of a vector representative of the LPC filter in a quantization domain and of a quantized weighted residual version of the vector, a calculator of an inverse weighting function from the first-stage approximation, an inverse quantizer of the quantized weighted residual version of the vector to produce a weighted residual vector, a multiplier of the weighted residual vector by the inverse weighting function to produce a residual vector, and an adder of the first-stage approximation with the residual vector to produce the vector representative of the LPC filter in the quantization domain.

WO 2010/003253 A8

WO 2010/003253 A8

(48) Date of publication of this corrected version:
1 April 2010

(15) Information about Correction:
see Notice of 1 April 2010

VARIABLE BIT RATE LPC FILTER QUANTIZING AND INVERSE QUANTIZING DEVICE AND METHOD

FIELD OF THE INVENTION

[0001] The present invention relates to coding and decoding of a sound signal, for example an audio signal. More specifically, the present invention relates to variable bit rate LPC (Linear Prediction Coefficients) filter quantizing and inverse quantizing device and method.

BACKGROUND OF THE INVENTION

[0002] The demand for efficient digital speech and audio coding techniques with a good trade-off between subjective quality and bit rate is increasing in various application areas such as teleconferencing, multimedia, and wireless communication.

[0003] A speech coder converts a speech signal into a digital bit stream which is transmitted over a communication channel or stored in a storage medium. The speech signal to be coded is digitized, that is sampled and quantized using for example 16-bits per sample. A challenge of the speech coder is to represent the digital samples with a smaller number of bits while maintaining a good subjective speech quality. A speech decoder or synthesizer converts the transmitted or stored bit stream back to a sound signal.

[0004] *Code-Excited Linear Prediction* (CELP) coding is one of the best techniques for achieving a good compromise between subjective quality and bit rate. The CELP coding technique is a basis for several speech coding standards both in wireless and wireline applications. In CELP coding, the speech signal is sampled and processed in successive blocks of L samples usually called *frames*, where L is a

predetermined number of samples corresponding typically to 10-30 ms of speech. A linear prediction (LP) filter is computed and transmitted every frame; the LP filter is also known as LPC (Linear Prediction Coefficients) filter. The computation of the LPC filter typically uses a *lookahead*, for example a 5-15 ms speech segment from the subsequent frame. The *L*-sample frame is divided into smaller blocks called *subframes*. In each subframe, an excitation signal is usually obtained from two components, a past excitation and an innovative, fixed-codebook excitation. The past excitation is often referred to as the adaptive-codebook or pitch-codebook excitation. The parameters characterizing the excitation signal are coded and transmitted to the decoder, where the excitation signal is reconstructed and used as the input of the LPC filter.

[0005] In applications such as multimedia streaming and broadcast, it may be required to encode speech, music, and mixed content at low bit rate. For that purpose, encoding models have been developed which combine a CELP coding optimized for speech signals with transform coding optimized for audio signals. An example of such models is the AMR-WB+ [1], which switches between CELP and TCX (Transform Coded eXcitation). In order to improve the quality of music and mixed content, a long delay is used to allow for finer frequency resolution in the transform domain. In AMR-WB+, a so-called super-frame is used which consists of four CELP frames (typically 80 ms).

[0006] Although the CELP coding parameters are transmitted once every 4 frames in AMR-WB+, quantization of the LPC filter is performed separately in each frame. Also, the LPC filter is quantized with a fixed number of bits per frame in the case of CELP frames.

[0007] To overcome the above stated drawback, there is a need for a technique using variable bit rate quantization of the LPC filters to reduce the number of bits involved in the quantization of the LPC filters. Also, such a technique will be relevant to speech or audio coding models which use both a super-frame and a multi-mode

structure such as the AMR-WB+ [1] and USAC being developed in MPEG.

SUMMARY OF THE INVENTION

[0008] According to a first aspect of the present invention, there is provided a device for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising: means for computing a first-stage approximation of the input vector; means for subtracting the first-stage approximation from the input vector to produce a residual vector; means for calculating a weighting function from the first-stage approximation; means for applying the weighting function to the residual vector; and means for quantizing the weighted residual vector to supply a quantized weighted residual vector.

[0009] According to a second aspect of the present invention, there is provided a device for inverse quantizing of a LPC filter, comprising: means for receiving coded indices representative of a first-stage approximation of a vector representative of the LPC filter in a quantization domain, and a quantized weighted residual version of the vector; means for calculating an inverse weighting function from the first-stage approximation; means for inverse quantizing the quantized weighted residual version of the vector to produce a weighted residual vector; means for applying the inverse weighting function to the weighted residual vector to produce a residual vector; and means for adding the first-stage approximation with the residual vector to produce the vector representative of the LPC filter in the quantization domain.

[0010] According to a third aspect of the present invention, there is provided a method for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising: computing a first-stage approximation of the input vector; subtracting the first-stage approximation from the input vector to produce a residual vector; calculating a weighting function from the first-stage approximation; applying the weighting function to the residual vector; and quantizing the weighted residual vector to

supply a quantized weighted residual vector.

[0011] According to a fourth aspect of the present invention, there is provided a method for inverse quantizing of a LPC filter, comprising: receiving coded indices representative of a first-stage approximation of a vector representative of the LPC filter in a quantization domain, and a quantized weighted residual version of the vector; calculating an inverse weighting function from the first-stage approximation; inverse quantizing the quantized weighted residual version of the vector to produce a weighted residual vector; applying the inverse weighting function to the weighted residual vector to produce a residual vector; and adding the first-stage approximation with the residual vector to produce the vector representative of the LPC filter in the quantization domain.

[0012] The foregoing and other objects, advantages and features of the present invention will become more apparent upon reading of the following non restrictive description of illustrative embodiments thereof, given by way of example only with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWING

[0013] In the appended drawings:

[0014] Figure 1 is a block diagram illustrating an absolute and multi-reference differential LPC filter quantizer and quantizing method;

[0015] Figure 2 is a schematic diagram illustrating an open-loop quantization scheme;

[0016] Figure 3 is a flow chart illustrating a device and method for determining LPC filters to be transmitted in a configuration in which four (4) LPC filters are used and transmitted in a super-frame;

[0017] Figure 4a is a typical LPC analysis window and typical LPC analysis center position when one LPC filter is estimated per frame (or super-frame) in a LPC-based codec, wherein LPC0 corresponds to a last LPC filter calculated during the previous frame (or super-frame);

[0018] Figure 4b is a typical LPC analysis window when four (4) LPC filters are estimated per frame (or super-frame) in a LPC-based codec, wherein the LPC analysis window is centered at the end of the frame;

[0019] Figure 5 is a flow chart illustrating an example of an out-of-the-loop quantization scheme;

[0020] Figure 6 is a schematic block diagram of a weighted algebraic LPC quantizer and quantizing method;

[0021] Figure 7 is a schematic block diagram of a weighted algebraic LPC inverse quantizer and inverse quantizing method;

[0022] Figure 8 is a schematic block diagram of a quantizer and quantizing method; and

[0023] Figure 9 is a schematic block diagram of a decoder and decoding method.

DETAILED DESCRIPTION

Differential quantization with a choice of possible references

[0024] Differential quantization with a choice between several possible

references is used. More specifically, a LPC filter is differentially quantized relative to several possible references.

[0025] Consecutive LPC filters are known to exhibit a certain degree of correlation. To take advantage of this correlation, LPC quantizers generally make use of prediction. Instead of quantizing the vector of Linear Prediction Coefficients (LPC vector) representing the LPC filter directly, a differential (or predictive) quantizer first computes a predicted value of this LPC vector and, then, quantizes the difference (often called prediction residual) between the original LPC vector and the predicted LPC vector.

[0026] Prediction is normally based on previous values of the LPC filter. Two types of predictors are commonly used: moving average (MA) and auto-regressive (AR) predictors. Although AR predictors are often more efficient in reducing the L2-norm (mean square) of the data to be quantized than MA predictors, the latter are sometimes useful because they are less prone to error propagation in case of transmission errors [2].

[0027] Since the L2-norm of the prediction residual is on average lower than the L2-norm of the original LPC vector (the ratio between the two depending on the degree of predictability of the LPC filter), a differential (or predictive) quantizer can achieve the same degree of performance as an absolute quantizer but at a lower bit rate.

[0028] On average, prediction is indeed efficient at reducing the L2-norm of the data to be quantized. This behavior is not constant however; prediction is much more efficient during stable segments of signal than during transitional segments. Prediction can even lead to increased L2-norm values when the LPC filters change fast. Some performance improvement can be achieved by considering two different predictors, one for highly predictive segments the other for less predictive segments [3, 4]. As mentioned in the foregoing description, this technique uses only past values of the LPC

filter.

[0029] To overcome this problem, it is proposed to differentially quantize a LPC filter relative to a reference, for example a reference filter, chosen among a number of possible references. The possible reference filters are already quantized past or future LPC filters (hence available at the decoder as at the encoder), or the results of various extrapolation or interpolation operations applied to already quantized past or future LPC filters. The reference filter that provides the lower distortion at a given rate, or the lower bit rate for a given target distortion level, is selected.

[0030] Figure 1 is a block diagram illustrating a multi-reference LPC filter quantization device and method. A given LPC filter 101 represented by a vector of Linear Prediction Coefficients is inputted to the multi-reference LPC filter quantization device and method. The input LPC filter 101 is differentially quantized with respect to a reference chosen among a number of possible references 1, 2, ... , *n*. Possible references comprise:

[0031] - past or future quantized LPC filters;

[0032] - the result of extrapolation or interpolation operations applied to past or future quantized LPC filters; or

[0033] - any quantized value available both at the encoder and the decoder.

[0034] As a non-limitative example, the input LPC filter 101 can be differentially quantized with respect to the previous quantized LPC filter, the following quantized LPC filter, or a mean value of those two previous and following quantized LPC filters. A reference can also be a LPC filter quantized using an absolute quantizer, or the result of any kind of interpolation, extrapolation or prediction (AR or MA) applied to already quantized LPC filters.

[0035] Operations 102 and 103_1, 103_2, ... , 103_n: Still referring to Figure 1, the input LPC filter 101 is supplied to an absolute quantizer (Operation 102) and to differential quantizers (Operations 103_1, 103_2, ... , 103_n). The absolute quantizer (Operation 102) quantizes the absolute value (not a difference) of the input LPC filter 101. The differential quantizers (Operations 103_1, 103_2, ... , 103_n) are designed to differentially quantize the input LPC filter 101 with respect to respective references 1, 2, ... , n.

[0036] Operation 104: The multi-reference LPC filter quantization device and method of Figure 1 comprises a selector for selecting a reference amongst the references 1, 2, ... , n that provides the lowest distortion level at a given bit rate, or the lowest bit rate for a given target distortion level. More specifically, the selector (Operation 104) uses a selection criterion that minimizes the bit rate to achieve a certain target level of distortion, or that minimizes the level of distortion produced at a given bit rate.

[0037] In Operation 104, the selection of a reference amongst references 1, 2, ... , n to be actually used in the differential quantization process can be performed in closed-loop or in open-loop.

[0038] In closed-loop, all possible references are tried and the reference that optimizes a certain criterion of distortion or bit rate is chosen. For example, the closed-loop selection can be based on minimizing a weighted mean-squared error between the input LPC vector and the quantized LPC vector corresponding to each reference. Also, the spectral distortion between the input LPC vector and the quantized LPC vector can be used. Alternatively, the quantization using the possible references can be performed while maintaining a distortion under a certain threshold, and the reference that both meets with this criterion and uses the smaller number of bits is chosen. As will be explained in the following description, a variable bit rate algebraic vector quantizer can be used to quantize the scaled residual vector (difference between the input LPC vector and the reference) which uses a certain bit budget based on the energy of the scaled

residual vector. In this case, the reference which yields the smaller number of bits is chosen.

[0039] In open-loop, the selector of Operation 104 predetermines the reference based on the value of the Linear Prediction Coefficients of the input LPC filter to be quantized and of the Linear Prediction Coefficients of the available reference LPC filters. For example, the L2-norm of the residual vector is computed for all references and the reference that yields the smaller value is chosen.

[0040] Operation 105: Following the selection of one of the references 1, 2, … , *n* by the Operation 104, a transmitter (Operation 105) communicates or signals to the decoder (not shown) the quantized LPC filter (not shown) and an index indicative of the quantization mode (sub-operation 105_1), for example absolute or differential quantization. Also, when differential quantization is used, the transmitter (Operation 105) communicates or signals to the decoder indices representative of the selected reference and associated differential quantizer of Operations 103_1, 103_2, … , 103_n (sub-operation 105_2). Some specific bits are transmitted to the decoder for such signaling.

[0041] Using a number of different possible references makes differential quantization more efficient in terms of reduction of the L2-norm of prediction residual compared to restricting to past values only as in conventional prediction. Also, for a given target level of distortion, this technique is more efficient in terms of average bit rate.

Switched absolute or differential quantization

[0042] According to a second aspect, switched absolute / differential (or predictive) quantization is used. Figure 1 illustrates an example of an absolute/differential scheme which selects between one absolute quantizer (Operation 102) and *n* differential quantizers (Operations 103_1, 103_2, … , 103_n) using respective,

different references 1, 2, … , *n*. Again, the selection of a quantizer can be made by the selector of Operation 104 amongst the absolute and differential quantizers (Operations 102 and 103_1, 103_2, … , 103_n), wherein the selected quantizer will, according to the selection criterion, minimize the level of distortion produced at a given bit rate or minimize the bit rate to achieve a target level of distortion.

[0043] Some LPC filters can be coded using the absolute quantizer (Operation 102). The other LPC filters are coded differentially with respect to one or several reference LPC filters in the differential quantizers (Operations 103_1, 103_2, … , 103_n).

[0044] The absolute quantizer (Operation 102) can be used as a safety-net solution for the otherwise differentially quantized LPC filters, for example in the case of large LPC deviations or when the absolute quantizer (Operation 102) is more efficient than the differential quantizers (Operations 103_1, 103_2, … , 103_n) in terms of bit rate. The reference LPC filter(s) can be all within the same super-frame to avoid introducing dependencies between super-frames which usually pose problems in case of transmission errors (packet losses or frame erasures).

[0045] As explained in the foregoing description, the use of prediction in LPC quantization leads to a reduced L2-norm of the data to be quantized and consequently to a reduction in average bit rate for achieving a certain level of performance. Prediction is not always equally efficient however. In switched LPC [3, 4], a pre-classification of the LPC filter is performed and different predictors are used depending on the predictability of the LPC filter to be quantized. However this technique has been developed in the context of a fixed bit rate, the two differential quantizers requiring a same number of bits to code an LPC filter.

[0046] Also, there may be provided one or several absolute quantizers (Operation 102). Moreover, there may be provided one or several differential (predictive) quantizers (Operations 103_1, 103_2, … , 103_n). Several differential quantizers

(Operations 103_1, 103_2, … , 103_n) involves several possible references such as 1, 2, … , *n* and/or several differential quantizer sizes and/or structures.

[0047] As described in the foregoing description, when several differential quantizers (Operations 103_1, 103_2, … , 103_n) are used, selection of the actual differential quantizer to be used can be performed in an open-loop or in a closed-loop selecting process.

[0048] When differential quantization fails to achieve a target level of distortion, or when absolute quantization requires a smaller number of bits than differential quantization to achieve that level of distortion, absolute quantization is used as a safety-net solution. One or several bits, depending on the number of possible absolute and differential quantizers is (are) transmitted through the transmitter (Operation 105) to indicate to the decoder (not shown) the actual quantizer being used.

[0049] Absolute / differential quantization combines the advantages of predictive quantization (reduction in bit rate associated with the reduction of the L2-norm of the data to be quantized) with the generality of absolute quantization (which is used as a safety-net in case differential (or predictive) quantization does not achieve a target, for example unnoticeable, level of distortion).

[0050] When several differential quantizers (Operations 103_1, 103_2, … , 103_n) are included, these differential quantizers can make use of either a same predictor or different predictors. In particular, but not exclusively, these several differential quantizers can use the same prediction coefficients or different prediction coefficients.

[0051] The decoder comprises means for receiving and extracting from the bit stream, for example a demultiplexer, (a) the quantized LPC filter and (b) the index (indices) or information:

[0052] - about the quantization mode to determine if the LPC filter has been quantized using absolute quantization or differential quantization; and

[0053] - about the reference amongst the plurality of possible references that has been used for quantizing the LPC filter.

[0054] If the information about the quantization mode indicates that the LPC filter has been quantized using absolute quantization, an absolute inverse quantizer (not shown) is provided for inverse quantizing the quantized LPC filter. If the information about the quantization mode indicates that the LPC filter has been quantized using differential quantization, a differential inverse quantizer (not shown) then differentially inverse quantizes the multi-reference differentially quantized LPC filter using the reference corresponding to the extracted reference information.

Out-of-the-loop quantization scheme

[0055] The AMR-WB+ codec is a hybrid codec that switches between a time-domain coding model based on the ACELP coding scheme, and a transform-domain coding model called TCX. The AMR-WB+ proceeds as follows [1]:

[0056] - The input signal is segmented into super-frames of four (4) frames;

[0057] - Each super-frame is encoded using a combination of four (4) possible coding modes, each coding mode covering a different duration:

- ACELP (covering a duration of one (1) frame);

- TCX256 (covering a duration of one (1) frame);

- TCX512 (covering a duration of two (2) frames); and

- TCX1024 (covering a duration of four (4) frames).

[0058] There are therefore 26 possible mode combinations to code each super-frame.

[0059] For a given super-frame, the combination of modes which minimizes a total weighted error is determined by a "closed-loop" mode selection procedure. More specifically, instead of testing the 26 combinations, the selection of the mode is performed through eleven (11) different trials (tree search, see Table 1). In AMR-WB+ codec, the closed-loop selection is based on minimizing the mean-squared error between the input and codec signal in a weighted domain (or maximizing the signal to quantization noise ratio).

Trial	Frame 1	Frame 2	Frame 3	Frame 4
1	ACELP			
2	TCX256			
3		ACELP		
4		TCX256		
5	TCX512			
6			ACELP	
7			TCX256	
8				ACELP
9				TCX256
10			TCX512	
11	TCX1024			

Table 1 – The 11 trials for closed-loop mode selection in AMR-WB+

[0060] The LPC filters are one of the various parameters transmitted by the AMR-WB+ codec. Following are some key elements regarding the quantization and transmission of those LPC filters.

[0061] Although the different coding modes do not cover the same number of frames, the number of LPC filters transmitted to the decoder is the same for all coding modes and equal to 1. Only the LPC filter corresponding to the end of the covered segment is transmitted. More specifically, in the case of TCX1024, one (1) LPC filter is calculated and transmitted for a duration of four (4) frames. In the case of TCX512, one (1) LPC filter is calculated and transmitted for a duration of two (2) frames. In the case of TCX256 or ACELP, one (1) LPC filter is calculated and transmitted for the duration of one (1) frame.

[0062] The AMR-WB+ codec uses a (first order moving-average) predictive LPC quantizer. The operation of the latter quantizer depends on the previously transmitted LPC filter, and consequently on the previously selected coding mode. Therefore, because the exact combination of modes is not known until the entire super-frame is coded, some LPC filters are encoded several times before the final combination of modes is determined.

[0063] For example, the LPC filter located at the end of frame 3 is transmitted to the decoder only when the third frame is encoded as ACELP or TCX256. It is not transmitted when frames 3 and 4 are jointly encoded using TCX512. With regards to the LPC filter located at the end of frame 2, it is transmitted in all combinations of modes except in TCX1024. Therefore, the prediction performed when quantizing the last LPC filter of the super-frame depends on the combination of modes for the whole super-frame.

[0064] The principle of the disclosed technique is that the order in which the LPC filters are quantized is chosen so that, once the closed-loop decision is finalized,

the quantization information corresponding to the unnecessary LPC filters can be skipped from the transmission with no effect on the way the other filters will be transmitted and decoded at the decoder. For each LPC filter to be quantized using the differential quantization strategy described above, this imposes some constraints on the possible reference LPC filters.

[0065] The following example is given with reference to Figure 2.

[0066] Operation 1 of Figure 2: To avoid any inter super-frame dependencies, at least one LPC filter is quantized using an absolute LPC quantizer. Since filter LPC4 of frame 4 of the super-frame is always transmitted whatever the coding mode combination determined by the closed-loop selection procedure, it is convenient to quantize that filter LPC4 using an absolute quantizer.

[0067] Operation 2 of Figure 2: The next LPC filter to be quantized is filter LPC2 of frame 2 of the super-frame which is transmitted for all combinations of modes except for TCX1024. A differential quantizer can be used, for example to code the difference between filter LPC2 and the absolute quantized version of filter LPC4. The same absolute quantizer as used for coding filter LPC4 can also be used as a safety-net solution, for example in the case of large LPC deviations or when the absolute LPC quantizer is more efficient than the differential quantizer in terms of bit rate and/or level of distortion.

[0068] Operation 3 of Figure 2: The remaining two LPC filters (filter LPC1 of frame 1 of the super-frame and filter LPC3 of frame 3 of the super-frame) are also quantized using the same differential/absolute quantization strategy. Both LPC filters can be quantized relative to the quantized version of filter LPC2. Some alternative strategies are given herein below.

[0069] Figure 5 is a flow chart illustrating in more detail an example of an out-of-

the-loop quantization scheme.

[0070] Operation 501: An absolute quantizer quantizes the filter LPC4.

[0071] Operation 502: Operation 502 is optional and used in a first LPC-based coding frame after a non-LPC-based coding frame. An absolute quantizer quantizes the filter LPC0 or a differential quantizer diffentially quantizes the filter LPC0 relative to the quantized filter LPC4. The filter LPC0 is the last LPC filter (LPC4) from the previous super-frame and can be used as a possible reference for quantizing the filters LPC1 to LPC4.

[0072] Operation 503: An absolute quantizer quantizes the filter LPC2 or a differential quantizer diffentially quantizes the filter LPC2 relative to the quantized filter LPC4 used as reference.

[0073] Operation 504: An absolute quantizer quantizes the filter LPC1, a differential quantizer diffentially quantizes the filter LPC1 relative to the quantized filter LPC2 used as reference, or a differential quantizer differentially quantizes the filter LPC1 relative to (quantized filter LPC2+quantized filter LPC0)/2) used as reference.

[0074] Operation 505: An absolute quantizer quantizes the filter LPC3, a differential quantizer diffentially quantizes the filter LPC3 relative to the quantized filter LPC2 used as reference, a differential quantizer differentially quantizes the filter LPC3 relative to quantized filter LPC4 used as reference, or a differential quantizer differentially quantizes the filter LPC3 relative to (quantized filter LPC2+quantized filter LPC4)/2) used as reference.

[0075] Figure 3 is a flow chart illustrating determination of LPC filters to be transmitted in a configuration where four (4) LPC filters can be calculated and

transmitted in a super-frame.

[0076] First of all it should be kept in mind that quantized filter LPC1 is transmitted only when ACELP and/or TCX256 is selected for the first half of the super-frame. Similarly, filter LPC3 is transmitted only when ACELP and/or TCX256 is used for the second half of that super-frame.

[0077] Operation 301: Filter LPC1 of frame 1 of the super-frame, filter LPC2 of frame 2 of the super-frame, filter LPC3 of frame 3 of the super-frame, and filter LPC4 of frame 4 of the super-frame are quantized using for example the quantization stategy illustrated and described in relation to Figures 2 and 5. Of course, other quantization strategies are possible.

[0078] Operation 302: Closed-loop selection of the coding modes as described hereinabove is performed.

[0079] Operation 303: The quantized filter LPC4 is transmitted to the decoder for example through the transmitter 105 of Figure 1. The decoder comprises:

- means for receiving and extracting from the received bitstream, for example a demultiplexer, the quantized filter LPC4; and

- an absolute inverse quantizer supplied with the quantized filter LPC4 for inverse quantizing the quantized filter LPC4.

[0080] Operation 304: If the super-frame is coded using mode TCX1024, no further transmission is required.

[0081] Operation 305: If the first, second, third and fourth frames of the super-frame are not coded using mode TCX1024, the quantized filter LPC2 and an index

indicative of one of the absolute quantization mode and the differential quantization mode are transmitted to the decoder for example through the transmitter 105 of Figure 1. The decoder comprises:

- means for receiving and extracting from the received bitstream, for example a demultiplexer, the quantized filter LPC2 and the index indicative of one of the absolute quantization mode and the differential quantization mode; and

- an absolute inverse quantizer supplied with the quantized filter LPC2 and the index indicative of the absolute quantization mode for inverse quantizing the quantized filter LPC2, or a differential inverse quantizer supplied with the quantized filter LPC2 and the index indicative of the differential quantization mode for inverse quantizing the quantized filter LPC2.

[0082] Operation 306: If frames 1 and 2 of the super-frame are coded using mode TCX512, the quantized filter LPC1 is not transmitted to the decoder.

[0083] Operation 307: If frames 1 and 2 of the super-frame are not coded using mode TCX512, i.e. if frames 1 and 2 of the super-frame are coded using ACELP or TCX256, the quantized filter LPC1, and an index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC0)/2 used as reference are transmitted to the decoder for example through the transmitter 105 of Figure 1. The decoder comprises:

- means for receiving and extracting from the received bitstream, for example a demultiplexer, the quantized filter LPC1, and the index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC0)/2 used as reference; and

- an absolute inverse quantizer supplied with the quantized filter LPC1 and the index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC0)/2) used as reference for inverse quantizing the quantized filter LPC1.

[0084] Operation 308: If frames 3 and 4 of the super-frame are coded using mode TCX512, the quantized filter LPC3 is not transmitted to the decoder.

[0085] Operation 309: If frames 3 and 4 of the super-frame are not coded using mode TCX512, i.e. if frames 3 and 4 of the super-frame are coded using ACELP or TCX256, the quantized filter LPC3 and the index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, the differential quantization mode relative to quantized filter LPC4 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC4)/2 used as reference are transmitted to the decoder for example through the transmitter 105 of Figure 1. The decoder comprises:

- means for receiving and extracting from the received bitstream, for example a demultiplexer, the quantized filter LPC3, and the index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, the differential quantization mode relative to quantized filter LPC4 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC4)/2 used as reference; and

- an absolute inverse quantizer supplied with the quantized filter LPC3 and the index indicative of one of the absolute quantization mode, the differential quantization mode relative to quantized filter LPC2 used as reference, the differential quantization mode relative to quantized filter LPC4 used as reference, and the differential quantization mode relative to (quantized filter LPC2+quantized filter LPC4)/2 used as reference for

inverse quantizing the quantized filter LPC3.

[0086] Some benefits of the above described solution comprise:

- Quantizing the whole set of LPC filters prior to the closed-loop selection of the coding modes saves complexity;

- Using a differential quantizer in the global quantization scheme preserves some of the bit rate saving that was gained by, for example, the predictive quantizer in the original AMR-WB+ quantization scheme.

[0087] The following variants can be used to build the reference LPC filters that are used in the differential quantizers (Operations such as 103_1, 103_2, … , 103_n):

- If inter super-frame dependency is not an issue, the last LPC filter (LPC4) from the previous super-frame (LPC0) can be used as a possible reference for encoding the filters LPC1 to LPC4;

- When different reference LPC filters are available, for example filter LPC0 and LPC4 when coding filter LPC2, a specific bit pattern can be transmitted to the decoder to indicate which of the references is actually used. For example, selection of the reference can be performed as described hereinabove with reference to Figure 1, for example on the basis of a distance or a bit rate measurement.

- When different reference LPC filters are available, additional secondary reference LPC filters can be obtained by applying various extrapolations or interpolations schemes to the already available reference LPC filters. A specific bit pattern can be transmitted to indicate the actual interpolation or extrapolation strategy selected by the coder. For example, filter LPC3 can be quantized differentially with respect to the quantized versions of either filter LPC2 or LPC4, or even with respect to an

interpolated value (e.g. average) between these two quantized filters LPC2 and LPC4 (see Operation 505 of Figure 5).

[0088] The above described "out-of-the loop" quantization scheme can be extended to coding more than four (4) LPC filters: for example to quantize and transmit filter LPC0 together with the super-frame. In that case, filter LPC0 corresponding to the last LCP filter (LPC4) calculated during the previous super-frame could be, as a non-limitative example, be quantized relative to filter LPC4 since this filter LPC4 is always available as a reference. Quantized filter LPC0 is transmitted to the decoder along with an index indicative of one of the absolute quantization mode and the differential quantization mode. The decoder comprises comprises:

- means for receiving and extracting from the received bitstream, for example a demultiplexer, the quantized filter LPC0, and the index indicative of one of the absolute quantization mode and the differential quantization mode; and

- an absolute inverse quantizer supplied with the quantized filter LPC0 and the index indicative of the absolute quantization mode for inverse quantizing the quantized filter LPC0, or a differential inverse quantizer supplied with the quantized filter LPC0, and the index indicative of the differential quantization mode for inverse quantizing the quantized filter LPC0.

[0089] Transmitting filter LPC0 to the decoder is useful to initialize an LPC-based codec in the case of switching from a non-LPC-based coding mode to an LPC-based coding mode. Examples of non-LPC-based coding modes are: pulse code modulation (PCM), and transform coding used for example by MP3 and by the advanced audio codec AAC. Examples of LPC-based coding modes are: code excited linear prediction (CELP) and algebraic CELP (ACELP) used by the AMR-WB+ codec [1].

[0090] In LPC-based codecs, one or several LPC filters per frame (or per super-frame) are estimated and transmitted to the decoder. When one single LPC filter per frame is estimated and transmitted, this LPC filter is most often estimated using an LPC analysis window centered on the end of the frame as represented in Figure 4a. When several LPC filters are transmitted per frame (or per super-frame as in the AMR-WB+ codec), they are most often estimated at regularly spaced positions over the length of the frame as represented on Figure 4b. Filter LPC0 in Figures 4a and 4b is in fact the last LPC filter of the previous frame (or super-frame) which is quantized and transmitted to the decoder.

[0091] Typical LPC-based codecs generally use interpolated values for the LPC filters. In the example of Figure 4a, for example, the LPC-based codec would typically divide the frame into four (4) sub-frames and use a different interpolated LPC filter for each sub-frame, the LPC filter of the first sub-frame being closer to filter LPC0 and the LPC filter of the 4^{th} sub-frame being closer to filter LPC1.

[0092] In a codec which switches from a non-LPC-based coding mode to an LPC-based coding mode, the filter LPC0 used to operate the LPC-based codec is normally not available at the first frame following the switch from the non-LPC-based coding mode to the LPC-based coding mode.

[0093] In that context, it is proposed to provide a value for filter LPC0 which is available both at the coder and the decoder when coding and decoding the first frame after the switch from the non-LPC-based coding mode to the LPC-based coding mode. More specifically, the value of the filter LPC0 is obtained at the decoder from the parameters transmitted from the coder.

[0094] According to a first solution, the filter LPC0 is determined at the coder (using LPC analysis well-know to those of ordinary skill in the art), quantized and transmitted to the decoder after the switch from the non-LPC-based coding mode to the

LPC-based coding mode has been decided. The decoder uses the transmitted quantized value and the filter LPC0. To quantize the filter LPC0 efficiently, the out-of-the-loop quantization scheme as described above, extended to more than four (4) LPC filters can be used.

[0095] The following describes second and third solutions to estimate the filter LPC0 at the decoder from transmitted parameters:

- Estimation of the filter LPC0 from the other transmitted LPC filters using, for example, extrapolation; and

- Estimation of the filter LPC0 from the other transmitted parameters. For example the filter LPC0 can be estimated by applying the conventional LPC analysis procedure to the past decoded signal, more specifically the output of the switched decoder prior to the switch from the non-LPC-based coding mode to the LPC-based coding mode.

Quantization with a uniform algebraic vector quantizer

[0096] The principle of stochastic vector quantization is to search a codebook of vectors for the nearest neighbor (generally in terms of Euclidian distance or weighted Euclidian distance) of the vector to be quantized. When quantizing LPC filters in the LSF (Line Spectral Frequency) or ISF (Immitance Spectral Frequency) domains, a weighting Euclidian distance is generally used, each component of the vector being weighted differently depending on its value and the value of the other components [5]. The purpose of that weighting is to make the minimization of the Euclidian distance behave as closely as possible to a minimization of the spectral distortion. Unlike a stochastic quantizer, a uniform algebraic vector quantizer does not perform an exhaustive search of a codebook. It is therefore difficult to introduce a weighting function in the distance computation.

[0097] In the solution proposed herein, the LPC filters are quantized, as a non-limitative example, in the LSF domain. Appropriate means for converting the LPC filter in the LSF quantization domain to form the input LSF vector are therefore provided. More specifically, the LSF residual vector, i.e. the difference between the input LSF vector and a first-stage approximation of this input LSF vector, is warped using a weighting function computed from the first-stage approximation, wherein the first-stage approximation uses a stochastic absolute quantizer of the input LSF vector, a differential quantizer of the input LSF vector, an interpolator of the input LSF vector, or other element that gives an estimate of the input LSF vector to be quantized. Warping means that different weights are applied to the components of the LSF residual vector. Because the first-stage approximation is also available at the decoder, the inverse weights can also be computed at the decoder and the inverse warping can be applied to the quantized LSF residual vector. Warping the LSF residual vector according to a model that minimizes the spectral distortion is useful when the quantizer is uniform. The quantized LSFs received at the decoder are a combination of the first-stage approximation with a variable bit rate quantization, for example AVQ (Algebraic Vector Quantization), refinement which is inverse-warped at the decoder.

[0098] Some benefits of the proposed solution are the following:

- With a good weighting function, a uniform quantizer can provide a relatively uniform spectral distortion.

- The advantages of variable bit rate vector quantization, for example AVQ (Algebraic Vector Quantization), over SVQ (Stochastic Vector Quantization) are a smaller number of tables (memory), less complexity and higher bit-rate granularity.

- Another advantage in favor of variable bit rate vector quantization, for example AVQ (Algebraic Vector Quantization), is its unlimited codebook size; this guarantees the same spectral distortion for any type of signal.

[0099] The general principle for the quantization of a given LPC filter is given in Figure 6. In this non-limitative example, the LPC filter is quantized in the LSF domain.

[00100] Operation 601: A calculator computes a first-stage approximation 608 of the input LSF vector 607.

[00101] Operation 602: A subtractor subtracts the first-stage approximation 608 from Operation 601 from the input LSF vector 607 to produce a residual LSF vector 609.

[00102] Operation 603: A calculator derives a LSF weighting function 610 from the first-stage approximation 608 of Operation 601.

[00103] Operation 604: A multiplier, or warper, applies the LSF weighting function 610 from Operation 603 to the residual LSF vector 609 from Operation 602.

[00104] Operation 605: A variable bit rate quantizer, for example an algebraic vector quantizer (AVQ) quantizes the resulting weighted residual LSF vector 611 to supply a quantized weighted residual LSF vector 612.

[00105] Operation 606: A multiplexer is responsive to the first-stage approximation 608 from Operation 601 and the quantized weighted residual LSF vector 612 from Operation 605 to multiplex and transmit the corresponding coded indices 613.

[00106] The first-stage approximation (Operation 601) can be calculated in different ways. As a non-limitative example, the calculator of the first-stage approximation 608 can be an absolute stochastic vector quantizer of the input LSF vector 607 with a small number of bits, or a differential quantizer of the input LSF

vector 607 using a reference as explained above where the first-stage approximation is the reference itself. For example, when quantizing the vector LPC1 as in Figure 5, Operation 504, the calculator of the first-stage approximation 608 can be an absolute quantizer with 8 bits, or quantized filter LPC2 or (quantized filter LPC2+quantized filter LPC0)/2.

[00107] The calculation and purpose of the weighting function (Operation 603) is described herein below.

[00108] The corresponding inverse quantizer is illustrated in Figure 7.

[00109] Operation 701: The coded indices 707 from the coder are demultiplexed by a demultiplexer.

[00110] Operation 702: The demultiplexed coded indices include the first-stage approximation 708.

[00111] Operation 703: Since the first-stage approximation is available at the decoder as at the coder (Operation 702), a calculator can be used to calculate the inverse LSF weighting function 709.

[00112] Operation 704: Decoded indices 710 representative of the quantized weighted residual LSF vector are supplied to a variable bit rate inverse vector quantizer, for example an algebraic inverse vector quantizer (inverse AVQ) to recover the weighted residual LSF vector 711.

[00113] Operation 705: A multiplier multiplies the weighted residual LSF vector 711 from Operation 704 by the inverse LSF weighting function 709 from Operation 703 to recover the residual LSF vector 712.

[00114] Operation 706: An adder sums the first-stage approximation 708 from Operation 702 with the residual LSF vector 712 from Operation 705 to form the decoded LSF vector 713. The decoded LSF vector 713 is a combination of the first-stage approximation from Operation 702 with the variable bit rate inverse quantization refinement (Operation 704) which is inverse-weighted (Operation 705) at the decoder.

First-stage approximation

[00115] As explained above a given LPC filter can be quantized using several quantization modes including absolute quantization and differential quantization using several references. The first-stage approximation depends on the quantization mode. In the case of absolute quantization, the first-stage approximation can use a vector quantizer with a small number of bits (e.g. 8 bits). In the case of differential quantization, the first-stage approximation constitutes the reference itself. For example, when quantizing the vector LPC3 as illustrated in Figure 5 (Operation 505), the first-stage approximation can be one of the following:

- 8-bit VQ (absolute quantization);
- Quantized filter LPC2 (differential quantization using quantized filter LPC2 as reference);
- Quantized filter LPC4 (differential quantization using quantized filter LPC4 as reference); or
- Average of quantized filters LPC2 and LPC4 (differential quantization using (quantized filter LPC2+quantized filter LPC4)/2 as reference).

[00116] As a non-limitative example, in the case of a p-th order LPC filter expressed with LSF parameters, in the absolute quantization mode, the first-stage

approximation is calculated using a p-dimensional, 8-bit stochastic vector quantizer applied to the input LSF vector. A codebook search uses a weighted Euclidian distance in which each component of the squared difference between the input LSF vector and the codebook entry is multiplied by the weight $wt(i)$. For example, the weight $wt(i)$ can be given by the following expression:

$$wt(i)=\frac{1}{d_i}+\frac{1}{d_{i+1}}, \quad i=0,\ldots,p-1$$

with:

$$d_0 = f(0)$$
$$d_p = SF/2 - f(p-1)$$
$$d_i = f(i) - f(i-1), \quad i=1,\ldots,p-1$$

where $f(i)$, $i=0,...,p-1$ is the input LSF vector to be quantized, p is the order of LP analysis, and SF is the internal sampling frequency of the LPC-based codec (in Hz).

[00117] In the differential quantization modes, the first-stage approximation is based on already quantized LPC filters.

[00118] As explained with reference to Figure 5, the set of LPC filters is quantized in the following order: LPC4, LPC2, LPC1 and then LPC3. When required, the optional filter LPC0 is quantized after the filter LPC4. Therefore differential quantization of filter LPC2 can only be done with respect to LPC4, while differential quantization of filter LPC3 can be done with respect to LPC2, LPC4 or a combination of both LPC2 and LPC4; LPC1 is not considered a good choice because it is not adjacent to LPC3.

[00119] For each first-stage approximation $f_{1st}(i)$, the residual LSF vector is calculated as:

$$r(i) = f(i) - f_{1st}(i), \quad i = 0,\ldots, p-1$$

[00120] As shown in Figure 6, the residual LSF vector 609 from Operation 602 is weighted (Operation 604) with the weighting function 610 from Operation 603 computed based on the first-stage approximation $f_{1st}(i)$ to obtain a warped residual LSF vector 611 (Operation 604). The warped residual LSF vector 611 is then quantized using a variable bit rate quantizer, for example an algebraic vector quantizer (Operation 605).

[00121] For example, the weights applied to the components of the p-th residual LSF vector can be given by the following relation:

$$w(i) = \frac{1}{W} * \frac{400}{\sqrt{d_i . d_{i+1}}}, \quad i = 0,\ldots, p-1$$

with:

$$d_0 = f_{1st}(0)$$
$$d_p = SF/2 - f_{1st}(p-1)$$
$$d_i = f_{1st}(i) - f_{1st}(i-1), \quad i = 1,\ldots, p-1$$

where $f_{1st}(i)$ is the first-stage approximation, SF is the internal sampling frequency in Hz of the LPC-based codec, and W is a scaling factor which depends on the quantization mode. The value of W is chosen in order to obtain a certain target spectral distortion and/or a certain target average bit rate once the warped residual LSF vector is quantized with the variable bit rate quantizer. As a non-limitative example, the variable bit rate vector quantizer chooses the bit rate for a certain vector based on its average energy.

[00122] In an illustrative example, the four (4) LPC filters in a super-frame, as well as the optional LPC0 filter are quantized according to Figure 5. Table 2 shows the used scaling factor for each quantization mode, and the encoding of the mode index

used in this example. Note that the quantization mode specifies which of the absolute or differential quantization is used, and in case of differential quantization it specifies the used reference filter. As explained above the reference filter used in differential quantization is the actual first-stage approximation for variable bit rate quantizing.

Filter	**Quantization mode**	**First stage approximation**	**Encoded mode**	***W***
LPC4	Absolute	8-bit VQ	(none)	60
LPC0	Absolute	8-bit VQ	0	60
	Relative LPC4	Quantized LPC4	1	63
LPC2	Absolute	8-bit VQ	0	60
	Relative LPC4	Quantized LPC4	1	63
LPC1	Absolute	8-bit VQ	00	60
	Relative (LPC0+LPC2)/2 (Note 1)	Quantized (LPC0+LPC2)/2	01	65
	Relative LPC2	Quantized LPC2	10	64
LPC3	Absolute	8-bit VQ	10	60
	Relative (LPC2+LPC4)/2	Quantized (LPC2+LPC4)/2	0	65
	Relative LPC2	Quantized LPC2	110	64
	Relative LPC4	Quantized LPC4	111	64
Note 1: in this mode, there is no second-stage AVQ quantizer				

Table 2 – Possible absolute and relative quantization modes and corresponding bitstream signaling, and the scaling factor and the weighting function

[00123] Figure 8 is a schematic block diagram explaining the quantization procedure as described herein above.

[00124] Operations 801, 801_1, 801_2, ... , 801_n: The input LSF vector 800 is supplied to an absolute quantizer (Operation 801) for performing, for example, a 8-bit absolute vector quantization of the input LSF vector 800. The input LSF vector is also supplied to differential quantizers (Operations 801_1, 801_2, ... , 801_n) for performing differential quantization of the input LSF vector 800. The differential quantizers use respective, different references as explained in the foregoing description with reference to Figure 1. The 8-bit VQ in Operation 801 and references in Operations 801_1, 801_2, ... , 801_n represent the first-stage approximation.

[00125] In Operations 802, 802_1, 802_2, ... , 802_n, a calculator calculates a residual LSF vector from the first stage approximation vector from the Operations 801, 801_1, 801_2, ... , 801_n, respectively. The residual vector is calculated as the difference between the input vector and the first-stage approximation. This corresponds to Operations 601 and 602 of Figure 6.

[00126] In Operations 803, 803_1, 803_2, ... , 803_n, a calculator calculates a weighting function to warp the residual LSF vector from the Operations 802, 802_1, 802_2, ... , 802_n, respectively. This corresponds to Operations 601 and 603 of Figure 6.

[00127] In Operations 804, 804_1, 804_2, ... , 804_n, a warper multiplies the residual LSF vector from the Operations 802, 802_1, 802_2, ... , 802_n, respectively, by the weighting function from the Operations 803, 803_1, 803_2, ... , 803_n, respectively.

[00128] In Operations 805, 805_1, 805_2, ... , 805_n, a variable bit rate quantizer, for example an algebraic vector quantizer (AVQ) quantizes the resulting weighted residual LSF vector from the Operations 804, 804_1, 804_2, ... , 804_n, respectively, to supply a quantized weighted residual LSF vector.

[00129] In Operation 806, the selection of a quantization mode is performed by a selector amongst absolute quantization (Operation 801) and differential quantization

using one of the references 1, 2, … , n (Operations 801_1, 801_2, … , 801_n). For example, Operation 806 could select the quantization mode (Operations 801, 801_1, 801_2, … , 801_n) that yields a lower distortion for a given bit rate or the lower bit rate for a target level of distortion. Regarding the selection amongst 8-bit VQ and references 1, 2, … , n, the selection can be performed in closed-loop or in open-loop. In closed-loop, all possible references are tried and the reference that optimizes a certain criterion of distortion or bit rate is chosen, for example the lower distortion for a given bit rate or the lower bit rate for a target level of distortion. In open-loop, the Operation 806 predetermines the reference based on the value of the Linear Prediction Coefficients of the LPC filter to be quantized and of the Linear Prediction Coefficients of the available reference LPC filters.

[00130] Operation 807: Following the selection in Operation 806, a transmitter (Operation 807) communicates or signals to the decoder (not shown) an index indicative of:

[00131] - the quantization mode (sub-operation 807_1), for example absolute or differential quantization; and

[00132] – in the case of differential quantization, of the selected reference and associated differential quantizer of Operations 801_1, 801_2, … , 801_n (sub-operation 807_2).

[00133] Some specific bits are transmitted to the decoder for such signaling.

Algebraic vector quantizer

[00134] A possible algebraic vector quantizer (AVQ) used for example in Operation 605 of Figure 6 and Operations 805, 805_1, 805_2, … , 805_n of Figure 8 is based on the 8-dimensional RE8 lattice vector quantizer used to quantize the spectrum in TCX

modes of AMR-WB+ [1].

[00135] For a 16th order LPC, each weighted residual LSF vector is split into two 8-dimensional sub-vectors B_1 and B_2. Each of these two sub-vectors is quantized using the three-operation approach described below.

[00136] LSF vectors do not have all the same sensitivity to quantization error, whereby a certain quantization error applied to one LSF vector can have more impact on spectral distortion than the same quantization error applied to another LSF vector. The weighting operation gives the same relative sensitivity to all weighted LSF vectors. The AVQ has the particularity of introducing the same level of quantization error to the weighted LSF vectors (uniform quantization error). When performing the inverse quantization, the inverse weighting which is applied to inverse-quantized weighted LSF vectors is also obviously applied to the quantization error. Thus, the originally uniform quantization error is distributed among quantized LSF vectors, the more sensitive LSF vectors acquiring a smaller quantization error and the less sensitive LSF vectors acquiring a larger quantization error. As a consequence, the impact of quantization error on spectral distortion is minimized.

[00137] As explained in Reference [1], the RE8 quantizer uses a fixed and predetermined quantization. As a consequence, the bit rate required to encode a subvector increases with the amplitude of this subvector.

[00138] The scaling factor W controls the amplitude of the weighted LSF vectors. Therefore, the scaling factor W also controls both the bit rate needed to quantize the LSF vector and the average spectral distortion.

First operation: find nearest neighbor in lattice RE_8

[00139] In this first operation, an 8-dimensional sub-vector B_k is rounded as a

point in the lattice RE_8, to produce its quantized version, $\hat{B}_k$. Before looking at the quantization procedure, it is worthwhile to look at the properties of this lattice. The lattice RE_8 is defined as follows:

$$RE_8 = 2D_8 \cup \{2D_8 + (1,\cdots,1)\}$$

that is as the union of a lattice $2D_8$ and a version of the lattice $2D_8$ shifted by the vector (1,1,1,1,1,1,1,1). Therefore, searching for the nearest neighbour in the lattice RE_8 is equivalent to searching for the nearest neighbour in the lattice $2D_8$, then searching for the nearest neighbour in the lattice $2D_8$ + (1,1,1,1,1,1,1,1), and finally selecting the best of those two lattice points. The lattice $2D_8$ is the lattice D_8 scaled by a factor of 2, with the lattice D_8 defined as:

$$D_8 = \{(x_1,\cdots,x_8) \in Z^8 \mid x_1 + \cdots + x_8 \quad is \quad even\}$$

That is, the points in the lattice D_8 are all integers, with the constraint that the sum of all components is even. This also implies that the sum of the components of a point in lattice $2D_8$ is an integer multiple of 4.

[00140] From this definition of lattice RE_8, it is straightforward to develop a fast algorithm to search for the nearest neighbour of an 8-dimensional sub-vector B_k among all lattice points in lattice RE_8. This can be done by applying the following operations. The components of sub-vector B_k are floating point values, and the result of the quantization, $\hat{B}_k$, will be a vector of integers.

1. $z_k = 0.5 * B_k$
2. Round each component of z_k to the nearest integer, to generate $\bar{z}_k$
3. $y1_k = 2\,\bar{z}_k$
4. Calculate S as the sum of the components of $y1_k$

5. If S is not an integer multiple of 4 (negative values are possible), then modify one of its components as follows:
   - find the position I where abs($z_k(i)$- $y1_k(i)$) is the highest
   - if $z_k(I)$- $y1_k(I) < 0$, then $y1_k(I) = y1_k(I) - 2$
   - if $z_k(I)$- $y1_k(I) > 0$, then $y1_k(I) = y1_k(I) + 2$
6. $z_k = 0.5 * (B_k - \mathbf{1.0})$ where **1.0** denotes a vector in which all the components are 1's
7. Round each component of z_k to the nearest integer, to generate $\bar{z}_k$
8. $y2_k = 2\,\bar{z}_k$
9. Calculate S as the sum of the components of $y2_k$
10. If S is not an integer multiple of 4 (negative values are possible), then modify one of its components as follows:
    - find the position I where abs($z_k(I)$- $y2_k(I)$) is the highest
    - if $z_k(I)$- $y2_k(I) < 0$, then $y2_k(I) = y2_k(I) - 2$
    - if $z_k(I)$- $y2_k$(I) > 0, then $y2_k(I) = y2_k(I) + 2$
11. $y2_k = y2_k + 1.0$
12. Compute $e1_k = (B_k - y1_k)^2$ and $e2_k = (B_k - y2_k)^2$
13. If $e1_k > e2_k$, then the best lattice point (nearest neighbour in the lattice) is $y1_k$ otherwise the best lattice point is $y2_k$.

$\hat{B}_k = c_k$ where c_k is the best lattice point as selected above.

Second operation: computation of the indices

[00141] In the first operation, each 8-dimensional sub-vector B_k was rounded as a point in the lattice RE_8. The result is $\hat{B}_k = c_k$, the quantized version of B_k. In the present second operation an index is computed for each c_k for transmission to the decoder. The computation of this index is performed as follows.

[00142] The calculation of an index for a given point in the lattice RE_8 is based on two basic principles:

1. All points in the lattice RE_8 lie on concentric spheres of radius $\sqrt{8m}$ with $m = 0, 1, 2, 3$, etc., and each lattice point on a given sphere can be generated by permuting the coordinates of reference points called *leaders*. There are very few leaders on a sphere, compared to the total number of lattice points which lie on the sphere. Codebooks of different bit rates can be constructed by including only spheres up to a given number m. See Reference [6] for more details, where codebooks Q_0, Q_1, Q_2, Q_3, Q_4, and Q_5 are constructed with respectively 0, 4, 8, 12, 16 and 20 bits. Hence, codebook Q_n requires $4n$ bits to index any point in that codebook.

2. From a base codebook C (i.e. a codebook containing all lattice points from a given set of spheres up to a number m), an extended codebook can be generated by multiplying the elements of the base codebook C by a factor M, and adding a second-stage codebook called the *Voronoi extension*. This construction is given by $\boldsymbol{y} = M\boldsymbol{z} + \boldsymbol{v}$, where M is the scale factor, $\boldsymbol{z}$ is a point in the base codebook and $\boldsymbol{v}$ is the Voronoi extension. The extension is computed in such a way that any point $\boldsymbol{y} = M\boldsymbol{z} + \boldsymbol{v}$ is also a point in the lattice RE_8. The extended codebook includes lattice points that extend further out from the origin than the base codebook.

[00143] In the present case, the base codebook C in the LPC quantizer can be either codebook Q_0, Q_2, Q_3 or Q_4 from Reference [6]. When a given lattice point c_k is not included in these base codebooks, the Voronoi extension is applied, using this time only the codebook Q_3 or Q_4. Note that here, $Q_2 \subset Q_3$ but $Q_3 \not\subset Q_4$.

[00144] Then, the calculation of the index for each lattice point c_k, obtained in the first operation, is performed according to the following operations.

Verify if c_k is in the base codebook C. This implies verifying if c_k is an element of codebooks Q_0, Q_2, Q_3 or Q_4 from Reference [6].

- If c_k is an element of the base codebook C, the index used to encode c_k is

thus the codebook number n_k plus the index I_k of the codevector c_k in the codebook Q_{n_k}. The codebook number n_k is encoded as described in a third operation. The index I_k indicates the rank of the codevector c_k, i.e. the permutation to be applied to a specific leader to obtain c_k (see Reference [7]). If $n_k = 0$, then I_k uses no bits. Otherwise, the index I_k uses $4n_k$ bits.

- If c_k is not in the base codebook, then apply the Voronoi extension through the following sub-operations, using this time only the codebook Q_3 or Q_4 as the base codebook.

V0 Set the extension order $r = 1$ and the scale factor $M = 2^r = 2$.

V1 Compute the Voronoi index $\boldsymbol{k}$ of the lattice point c_k. The Voronoi index $\boldsymbol{k}$ depends on the extension order r and the scale factor M. The Voronoi index is computed via modulo operations such that $\boldsymbol{k}$ depends only on the relative position of c_k in a scaled and translated Voronoi region:

$$\boldsymbol{k} = \mathrm{mod}_M (c_k G^{-1}).$$

where G is the generator matrix and $\mathrm{mod}_M(\cdot)$ is the component-wise modulo-M operation. Hence, the Voronoi index $\boldsymbol{k}$ is a vector of integers with each component comprised in the interval 0 to M– 1.

V2 Compute the Voronoi codevector $\boldsymbol{v}$ from the Voronoi index $\boldsymbol{k}$. This can be implemented using an algorithm as described in reference [8].

V3 Compute the difference vector $\boldsymbol{w} = c_k - \boldsymbol{v}$. This difference vector $\boldsymbol{w}$ always belongs to the scaled lattice $m\Lambda$, where Λ is the lattice RE_8. Compute $\boldsymbol{z} = \boldsymbol{w}/M$, i.e., apply the inverse scaling to the difference vector $\boldsymbol{w}$. The codevector $\boldsymbol{z}$ belongs to the lattice Λ since $\boldsymbol{w}$ belongs to $M\Lambda$.

V4 Verify if $\boldsymbol{z}$ is in the base codebook C (i.e. in Q_3 or Q_4)

If $\boldsymbol{z}$ is not in the base codebook C, increment the extension order r by 1, multiply the scale factor M by 2, and go back to sub-operation V1.

Otherwise, if $\boldsymbol{z}$ is in the base codebook C, then an extension order r and a scaling factor $M = 2^r$ sufficiently large to encode the index of c_k has been found. The index is formed of three parts: 1) the

codebook number n_k as a unary code defined below; 2) the rank I_k of z in the corresponding base codebook (either Q_3 or Q_4); and 3) the 8 indices of the Voronoi index vector $\boldsymbol{k}$ calculated in sub-operation V1, where each index requires exactly r bits (r is the Voronoi extension order set in sub-operation V0). The codebook number n_k is encoded as described in the third operation.

[00145] The lattice point c_k is then described as:

$$c_k = M\,\boldsymbol{z} + \boldsymbol{v}.$$

Third operation: variable length coding of the codebook numbers

[00146] The codebook numbers n_k are encoded using a variable-length code which depends on the position of the LPC filter and on the quantization mode, as indicated in Table 3.

Filter	Quantization mode	n_k mode
LPC4	Absolute	0
LPC0	Absolute	0
	Relative LPC4	3
LPC2	Absolute	0
	Relative LPC4	3
LPC1	Absolute	0
	Relative (LPC0+LPC2)/2	1
	Relative LPC2	2
LPC3	Absolute	0
	Relative (LPC2+LPC4)/2	1
	Relative LPC2	2
	Relative LPC4	2

Table 3 - Coding modes for codebook numbers n_k

n_k modes 0 and 3:

[00147] The codebook number n_k is encoded as a variable length code, as follows:

$Q_2 \rightarrow$ the code for n_k is 00

$Q_3 \rightarrow$ the code for n_k is 01

$Q_4 \rightarrow$ the code for n_k is 10

Others: the code for n_k is 11 followed by:

$Q_5 \rightarrow 0$

$Q_6 \rightarrow 10$

$Q_0 \rightarrow 110$

$Q_7 \rightarrow 1110$

$Q_8 \rightarrow 11110$

etc.

n_k mode 1:

[00148] The codebook number n_k is encoded as a unary code, as follows:

$Q_0 \rightarrow$ unary code for n_k is 0

$Q_2 \rightarrow$ unary code for n_k is 10

$Q_3 \rightarrow$ unary code for n_k is 110

$Q_4 \rightarrow$ unary code for n_k is 1110

etc.

n_k mode 2:

[00149] The codebook number n_k is encoded as a variable length code, as follows:

$Q_2 \rightarrow$ the code for n_k is 00

$Q_3 \rightarrow$ the code for n_k is 01

$Q_4 \rightarrow$ the code for n_k is 10

Others: the code for n_k is 11 followed by:

$Q_0 \rightarrow 0$

$Q_5 \rightarrow 10$

$Q_6 \rightarrow 110$

etc.

Quantization mode decision

[00150] For each LSF vector, all possible absolute and differential quantization modes as described in Table 2 are each tried and, for example, the quantization mode which requires the minimum number of bits is selected. The encoded quantization mode and the corresponding set of quantization indices are transmitted to the decoder.

[00151] As mentioned in the foregoing description, the actual number of quantized LPC filters transmitted from the coder to the decoder is not fixed but rather depends on the ACELP/TCX decision taken at the coder. For example, long TCX (TCX1024) requires only the transmission of quantized filter LPC4 while any combination involving ACELP or short TCX (TCX256) requires the transmission of all four (4) quantized LPC filters LPC1 to LPC4. Only the quantized LPC filters that are required by the ACELP/TCX mode configuration are actually transmitted.

Decoding process of algebraic vector quantizer

[00152] As mentioned herein above, the actual number of quantized LPC filters coded within the bitstream depends on the ACELP/TCX mode combination of the super-frame. The ACELP/TCX mode combination is extracted from the bitstream and determines the coding modes, mod[k] for k=0 to 3, of each of the four (4) frames composing the super-frame. The mode value is 0 for ACELP, 1 for TCX256, 2 for TCX512, 3 for TCX1024.

[00153] In addition to the one (1) to four (4) quantized LPC filters of the super-frame, the above described, optional quantized filter LPC0 is transmitted for the first super-frame of each segment coded using the linear-prediction based codec.

[00154] The order in which the quantized LPC filters are normally found in the bitstream is: LPC4, the optional LPC0, LPC2, LPC1, and LPC3.

[00155] The condition for the presence of a given LPC filter within the bitstream

is summarized in Table 4.

LPC filter	*Present if*
LPC0	1st super-frame encoded using LP
LPC1	mod[0]<2
LPC2	mod[2]<3
LPC3	mod[2]<2
LPC4	Always

Table 4 – Condition for the presence of a given LPC filter in the bitstream

[00156] Figure 9 is a schematic block diagram summarizing the decoding process.

[00157] Operations 901 and 902: The decoder comprises means for receiving and extracting, for example a demultiplexer, from the received bit stream the quantization indices corresponding to each of the quantized LPC filters required by the ACELP/TCX mode combination. For a given quantized LPC filter, a determiner of quantization mode extracts from the bit stream received from the coder the index or information related to the quantization mode, and determines whether the quantization mode is the absolute or differential quantization mode as indicated in Table 2.

[00158] Operations 903 and 905: When Operations 901 and 902 determine that the quantization mode is the absolute quantization mode, an extractor extracts from the bit stream the index or indices corresponding to the stochastic VQ-quantized first-stage approximation (Operation 903). A calculator then computes the first-stage approximation through inverse-quantization (Operation 905).

[00159] Operations 904 and 905: When Operations 901 and 902 determine that

the quantization mode is the differential quantization mode (not the absolute quantization mode), an extractor extracts from the bit stream the indices or information representative of the reference amongst the plurality of possible references, for example the reference LPC vector (Operation 904). The calculator then computes from this information the first-stage approximation as described with reference to Table 2 (Operation 905).

[00160] In Operation 906, an extractor of VQ information extracts from the bit stream received from the coder variable bit rate VQ information, for example AVQ information. More specifically, as a non-limitative example, the AVQ information for the two residual LSF sub-vectors $\hat{B}_k$ are extracted from the bit stream. The AVQ information normally comprises two encoded codebook numbers and the corresponding AVQ indices. The only exception is when filter LPC1 is differentially quantized relative to (quantized filter LPC0+quantized filter LPC2)/2, since in this case there is no AVQ information present in the bit stream. In the case of the latter exception, the quantized LSF vector 909 is outputted as the first-stage approximation from Operation 905.

[00161] Operation 907: An inverse algebraic vector quantizer receives the extracted AVQ information from Operation 906 to inverse quantize, or inverse weight and recover the AVQ contribution.

Decoding of AVQ indices

[00162] Decoding the LPC filters involves decoding the extracted AVQ information, for example the AVQ parameters describing each quantized sub-vector $\hat{B}_k$ of the weighted residual LSF vector. In the foregoing example, each sub-vector B_k has a dimension 8. The AVQ parameters for each sub-vector B_k are described in the second operation of the above described algebraic vector quantization. For each quantized sub-vector $\hat{B}_k$, three sets of binary indices are sent by the coder to the decoder:

a) the codebook number n_k, transmitted using an entropy code as described in the third operation of the above described algebraic vector quantization;

b) the rank I_k of a selected lattice point $\boldsymbol{z}$ in a base codebook, which indicates what permutation has to be applied to a specific leader (see second operation of the above described algebraic vector quantization) to obtain a lattice point $\boldsymbol{z}$; and

c) if the quantized sub-vector $\hat{B}_k$ (a lattice point in lattice RE_8) was not in the base codebook, the 8 indices of the Voronoi extension index vector $\boldsymbol{k}$ calculated in sub-operation **V1** of the second operation of the above described algebraic vector quantization; from the Voronoi extension indices, an extension vector $\boldsymbol{v}$ can be computed as taught by Reference [8]. The number of bits in each component of index vector $\boldsymbol{k}$ is given by the extension order r, which can be obtained from the code value of index n_k. The scaling factor M of the Voronoi extension is given by $M = 2^r$.

[00163] Then, from the scaling factor M, the Voronoi extension vector $\boldsymbol{v}$ (a lattice point in lattice RE_8) and the lattice point $\boldsymbol{z}$ in the base codebook (also a lattice point in lattice RE_8), each quantized scaled sub-vector $\hat{B}_k$ can be computed using the following relation:

$$\hat{B}_k = M\boldsymbol{z} + \boldsymbol{v}.$$

[00164] When there is no Voronoi extension (i.e. $n_k < 5$, M=1 and $\boldsymbol{z}$=0), the base codebook is either codebook Q_0, Q_2, Q_3 or Q_4 from Reference [6]. No bits are then required to transmit vector $\boldsymbol{k}$. Otherwise, when Voronoi extension is used because $\hat{B}_k$ is large enough, then only Q_3 or Q_4 from Reference [6] is used as a base codebook. The selection of Q_3 or Q_4 is implicit in the codebook number value n_k, as described in the second operation of the above described algebraic vector quantization.

[00165] Operation 908: An adder sums the first-stage approximation from

Operation 905 to the inverse-weighed AVQ contribution from Operation 907 to reconstruct and recover the quantized LSF vector 909.

[00166] Although the present invention has been defined in the foregoing description by means of illustrative embodiments thereof, the scope of the claims should not be limited by those embodiments, but should be given the broadest interpretation consistent with the description as a whole.

References

[1] 3GPP Technical Specification TS 26.290, "Audio Codec Processing Functions; Extended Adaptive Multi-Rate - Wideband (AMR-WB+) Codec; Transcoding Functions," June 2005.

[2] J. Skoglund, J. Linden, "Predictive VQ for Noisy Channel Spectrum Coding: AR Or MA?," IEEE 1997 International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97), pp. 1351-1354, Munich, Germany, April 21-24, 1997.

[3] H. Zarrinkoub, P. Mermelstein, "Switched Prediction and Quantization of LSP Frequencies," IEEE 1996 International Conference on Acoustics, Speech, and Signal Processing (ICASSP'96), Vol. 2, pp.757-760, 7-10 May 1996.

[4] A. V. McCree, "Method for Switched-Predictive Quantization," US Patent No. 6,122,608.

[5] R. Laroia, N. Phamdo, and N. Farvardin, "Robust and Efficient Quantization of Speech LSP Parameters Using Structured Vector Quantizers," IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'1991), pp. 641-644, Washington, DC, April 14-17, 1991.

[6] M. Xie and J.-P. Adoul, "Embedded Algebraic Vector Quantization (EAVQ) with Application to Wideband Audio Coding," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Atlanta, GA, U.S.A, Vol. 1, pp. 240-243, 1996.

[7] P. Rault, C. Guillemot, "Indexing Algorithm for Zn, An, Dn and Dn++ Lattice Vector Quantizers, IEEE Transactions on Multimedia, Vol. 3, No. 4, December 2001.

[8] J.H. Conway and N.J.A. Sloane, "A Fast Encoding Method for Lattice Codes and Quantizers," IEEE Trans. Inform. Theory, Vol. IT-29, No. 6, pp. 820-824, Nov. 1983.

WHAT IS CLAIMED IS:

1. A device for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising:

a plurality of quantization modules using respective, distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously quantized LPC filters, and wherein each quantization module comprises:

a calculator of a first-stage approximation of the input vector, wherein the first stage approximation includes (a) the result of absolute quantization of the input vector in the case of the absolute quantization mode and (b) the reference in the case of the differential quantization modes;

a subtractor of the first-stage approximation from the input vector to produce a residual vector;

a calculator of weights of a weighting function using a mathematical relation including the first-stage approximation as a variable;

a warper of the residual vector by applying the weights of the weighting function to the residual vector; and

a quantizer of the weighted residual vector to supply a quantized weighted residual vector; and

a quantization mode selector configured to select one of the quantization modules and the corresponding quantization mode based on a level of distortion.

2. An LPC filter quantizing device according to claim 1, further comprising means for converting the LPC filter in the quantization domain to form the input vector.

3. An LPC filter quantizing device according to claim 1, further comprising a multiplexer of the first-stage approximation of the input vector and the quantized weighted residual vector.

4. An LPC filter quantizing device according to claim 1, wherein the calculator of the weights of the weighting function uses the first-stage approximation and a scaling factor which is dependent upon the quantization mode selected by the quantization mode selector to calculate the weights.

5. An LPC filter quantizing device according to claim 4, wherein the scaling factor has a value chosen to attain at least one of a certain average bit rate and a certain average distortion.

6. An LPC filter quantizing device according to claim 1, wherein the quantization domain is a line spectral frequency domain.

7. An LPC filter quantizing device according to claim 1, wherein the weighting function uses a scaling factor based on a type of the LPC filter and on a selected quantization mode.

8. An LPC filter quantizing device according to claim 1, wherein the quantization mode selector is configured to select the quantization mode providing a lower level of distortion for a given bit rate.

9. An LPC filter quantizing device according to claim 1, wherein the quantization mode selector is configured to select the quantization mode providing a lower bit rate for a target level of distortion.

10. A device for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising:

a plurality of quantization modules using respective, distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously quantized LPC filters, and wherein each quantization module comprises:

a calculator of a first-stage approximation of the input vector, wherein the first stage approximation includes (a) the result of absolute quantization of the input vector in the case of the absolute quantization mode and (b) the reference in the case of the differential quantization modes;

a subtractor of the first-stage approximation from the input vector to produce a residual vector;

a calculator of weights of a weighting function using a mathematical relation including the first-stage approximation as a variable;

a warper of the residual vector by applying the weights of the weighting function to the residual vector; and

a quantizer of the weighted residual vector to supply a quantized weighted residual vector; and

a quantization mode selector configured to select one of the quantization modules and the corresponding quantization mode based on a level of distortion;

wherein, in each of the quantization modules, the quantizer of the weighted residual vector comprises a variable bit rate quantizer.

11. An LPC filter quantizing device according to claim 10, wherein the variable bit rate quantizer comprises an algebraic vector quantizer.

12. A device for inverse quantizing a LPC filter quantized in an encoder, in the form of an input vector in a quantization domain, by a LPC filter quantizing device comprising (a)

a plurality of quantization modules using respective, distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously quantized LPC filters, and wherein each quantization module comprises (i) a calculator of a first-stage approximation of the input vector, wherein the first stage approximation includes the result of absolute quantization of the input vector in the case of the absolute quantization mode and the reference in the case of the differential quantization modes, (ii) a subtractor of the first-stage approximation from the input vector to produce a residual vector, (iii) a calculator of weights of a weighting function using a mathematical relation including the first-stage approximation as a variable, (iv) a warper of the residual vector by applying the weights of the weighting function to the residual vector, and (v) a quantizer of the weighted residual vector to supply a quantized weighted residual vector; and (b) a quantization mode selector configured to select one of the quantization modules and the corresponding quantization mode based on a level of distortion, the LPC filter inverse quantizing device comprising:

a demultiplexer for receiving, from the encoder, and for demultiplexing coded indices representative of (a) the first-stage approximation from the selected quantization module, (b) the selected quantization mode, and (c) the quantized weighted residual vector from the selected quantization module;

a determiner of the selected quantization mode using the demultiplexed coded indices;

a calculator of the first-stage approximation in response to the demultiplexed coded indices, and as a function of the determined quantization mode;

a calculator of an inverse weighting function from the first-stage approximation;

an inverse quantizer of the quantized weighted residual vector responsive to the demultiplexed coded indices to produce a weighted residual vector;

a multiplier of the weighted residual vector by the inverse weighting function to produce a residual vector; and

an adder of the first-stage approximation with the residual vector to produce a vector representative of the LPC filter in the quantization domain.

13. An LPC filter inverse quantizing device according to claim 12, wherein the inverse quantizer comprises a variable bit rate inverse algebraic vector quantizer.

14. An LPC filter inverse quantizing device according to claim 12, wherein the quantization domain is a line spectral frequency domain.

15. A method implemented in an encoder for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising:

providing a plurality of distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously quantized LPC filters;

performing, for each of the plurality of distinct quantization modes, the following operations:

computing a first-stage approximation of the input vector, wherein the first stage approximation includes (a) the result of absolute quantization of the input vector in the case of the absolute quantization mode and (b) the reference in the case of the differential quantization modes;

subtracting the first-stage approximation from the input vector to produce a residual vector;

calculating weights of a weighting function using a mathematical relation including the first-stage approximation as a variable;

applying the weights of the weighting function to the residual vector; and

quantizing the weighted residual vector to supply a quantized weighted residual vector;

selecting one of the quantization modes based on a level of distortion; and

transmitting from the encoder to a decoder the selected quantization mode, the first-stage approximation of the input vector obtained using the selected quantization mode, and the quantized weighted residual vector obtained using the selected quantization mode.

16. An LPC filter quantizing method according to claim 15, further comprising converting the LPC filter in the quantization domain to form the input vector.

17. An LPC filter quantizing method according to claim 15, further comprising multiplexing the first-stage approximation of the input vector and the quantized weighted residual vector obtained using the selected quantization mode.

18. An LPC filter quantizing method according to claim 15, wherein calculating the weights of the weighting function comprises using the first-stage approximation and a scaling factor which is dependent upon the selected quantization mode.

19. An LPC filter quantizing method according to claim 18, wherein the scaling factor has a value chosen to attain at least one of a certain average bit rate and a certain average distortion.

20. An LPC filter quantizing method according to claim 15, wherein the quantization domain is a line spectral frequency domain.

21. A method implemented in an encoder for quantizing a LPC filter in the form of an input vector in a quantization domain, comprising:

providing a plurality of distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously quantized LPC filters;

performing, for each of the plurality of distinct quantization modes, the following operations:

computing a first-stage approximation of the input vector, wherein the first stage approximation includes (a) the result of absolute quantization of the input vector in the case of the absolute quantization mode and (b) the reference in the case of the differential quantization modes;

subtracting the first-stage approximation from the input vector to produce a residual vector;

calculating weights of a weighting function using a mathematical relation including the first-stage approximation as a variable;

applying the weights of the weighting function to the residual vector; and

quantizing the weighted residual vector to supply a quantized weighted residual vector;

selecting one of the quantization modes based on a level of distortion; and

transmitting from the encoder to a decoder the selected quantization mode, the first-stage approximation of the input vector obtained using the selected quantization mode, and the quantized weighted residual vector obtained using the selected quantization mode;

wherein quantizing the weighted residual vector comprises using a variable bit rate quantizer.

22. An LPC filter quantizing method according to claim 21, wherein using the variable bit rate quantizer comprises using an algebraic vector quantizer.

23. A method implemented in a decoder for inverse quantizing a LPC filter quantized in an encoder, in the form of an input vector in a quantization domain, by a LPC filter quantizing method comprising (a) providing a plurality of distinct quantization modes, wherein the quantization modes include an absolute quantization mode and differential quantization modes, wherein the differential quantization modes use respective, different references each based on a previously quantized LPC filter or a combination of previously

quantized LPC filters, (b) performing, for each of the plurality of distinct quantization modes, the following operations: (i) computing a first-stage approximation of the input vector, wherein the first stage approximation includes the result of absolute quantization of the input vector in the case of the absolute quantization mode and the reference in the case of the differential quantization modes, (ii) subtracting the first-stage approximation from the input vector to produce a residual vector, (iii) calculating weights of a weighting function using a mathematical relation including the first-stage approximation as a variable, (iv) applying the weights of the weighting function to the residual vector, and (v) quantizing the weighted residual vector to supply a quantized weighted residual vector, (c) selecting one of the quantization modes based on a level of distortion, and (d) transmitting from the encoder to the decoder the selected quantization mode, the first-stage approximation of the input vector obtained using the selected quantization mode, and the quantized weighted residual vector obtained using the selected quantization mode, the LPC filter inverse quantizing method comprising:

receiving at the decoder, from the encoder, coded indices representative of (a) the first-stage approximation from the selected quantization mode, (b) the selected quantization mode, and (c) the quantized weighted residual vector from the selected quantization mode;

determining the selected quantization mode using the received coded indices;

calculating, in response to the received coded indices, the first-stage approximation as a function of the determined quantization mode;

calculating an inverse weighting function from the first-stage approximation;

inverse quantizing the quantized weighted residual vector in response to the received coded indices to produce a weighted residual vector;

applying the inverse weighting function to the weighted residual vector to produce a residual vector; and

adding the first-stage approximation with the residual vector to produce a vector representative of the LPC filter in the quantization domain.

24. An LPC filter inverse quantizing method according to claim 23, wherein receiving the coded indices comprises demultiplexing the coded indices.

25. An LPC filter inverse quantizing method according to claim 23, wherein inverse quantizing the quantized weighted residual vector comprises variable bit rate inverse algebraic vector quantizing the quantized weighted residual vector.

26. An LPC filter inverse quantizing method according to claim 23, wherein applying the inverse weighting function to the weighted residual vector comprises multiplying the weighted residual vector by the inverse weighting function.

27. An LPC filter inverse quantizing method according to claim 23, wherein the quantization domain is a line spectral frequency domain.

28. An LPC filter quantizing device according to claim 7, wherein a combination of the type of LPC filter, of the quantization mode, and of the first-stage approximation is selected from the group consisting of:

LPC4 filter, with absolute quantization, and with 8-bit vector quantization (VQ) approximation;

LPC0 filter, with absolute quantization, and with 8-bit VQ approximation;

LPC0 filter, with relative LPC4 quantization, and with quantized LPC4 approximation;

LPC2 filter, with absolute quantization, and with 8-bit VQ approximation;

LPC2 filter, with relative LPC4 quantization, and with quantized LPC4 approximation;

LPC1 filter, with absolute quantization, and with 8-bit VQ approximation;

LPC1 filter, with relative (LPC0+LPC2)/2 quantization, and with quantized (LPC0+ LPC2)/2 approximation;

LPC1 filter, with relative LPC2 quantization, and with quantized LPC2 approximation;

LPC3 filter, with absolute quantization, and with 8-bit VQ approximation;

LPC3 filter, with relative (LPC2+LPC4)/2 quantization, and with quantized (LPC2+ LPC4)/2 approximation;

LPC3 filter, with relative LPC2 quantization, and with quantized LPC2 approximation; and

LPC3 filter, with relative LPC4 quantization, and with quantized LPC4 approximation;

wherein the LPC0 filter is the LPC filter of a last frame of a previous super-frame, the LPC1 filter is the LPC filter of a first frame of a current super-frame, the LPC2 filter is the LPC filter of a second frame of the current super-frame, the LPC3 filter is the LPC filter of a third frame of the current super-frame and the LPC4 filter is the LPC filter of a fourth frame of the current super-frame.

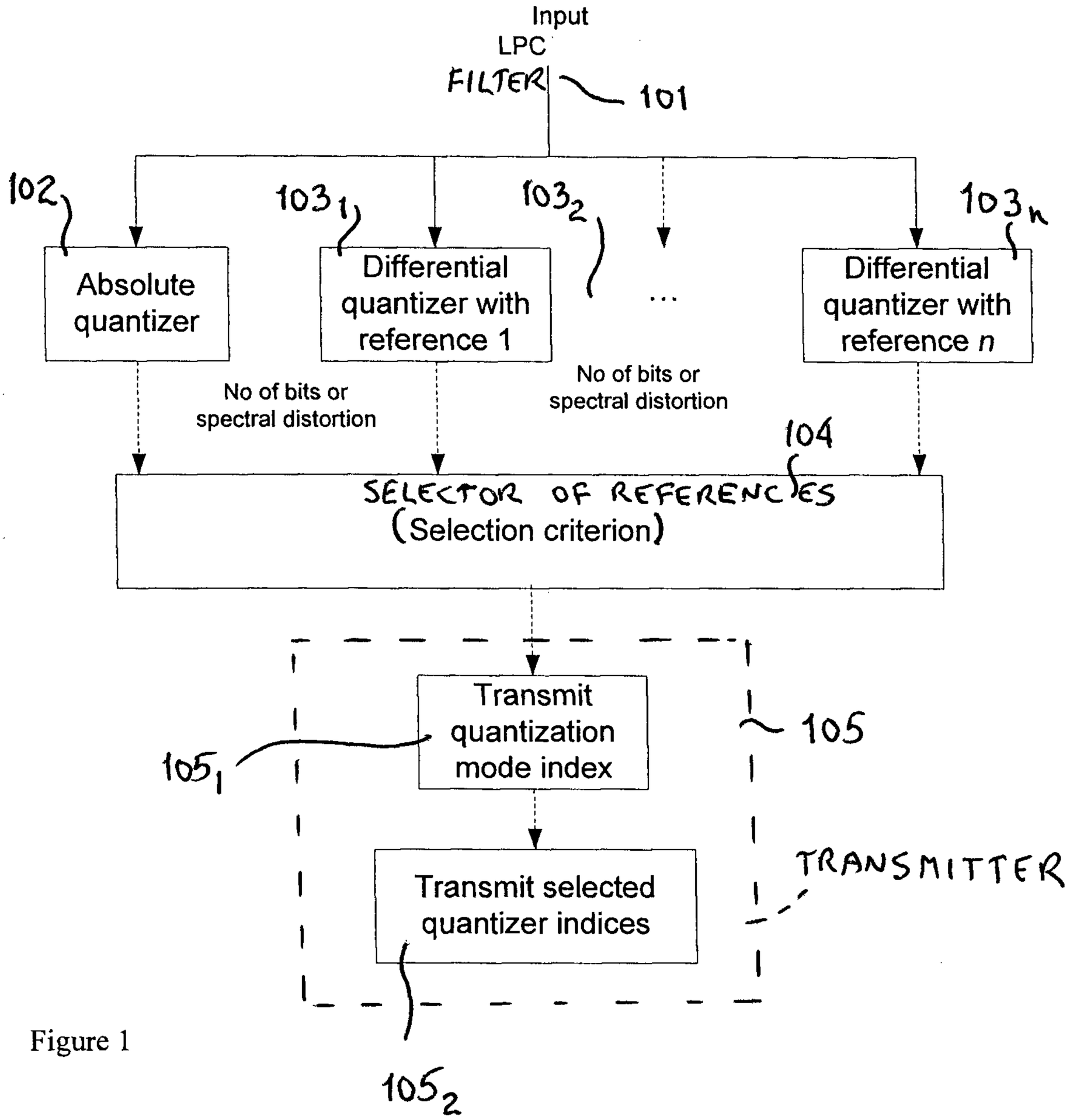

Input
LPC
FILTER
101
102
103 1
103 2
103 n
Absolute quantizer
Differential quantizer with reference 1
Differential quantizer with reference n
No of bits or spectral distortion
No of bits or spectral distortion
104
SELECTOR OF REFERENCES
(Selection criterion)
Transmit quantization mode index
105
105 1
Transmit selected quantizer indices
TRANSMITTER
105 2

Figure 1

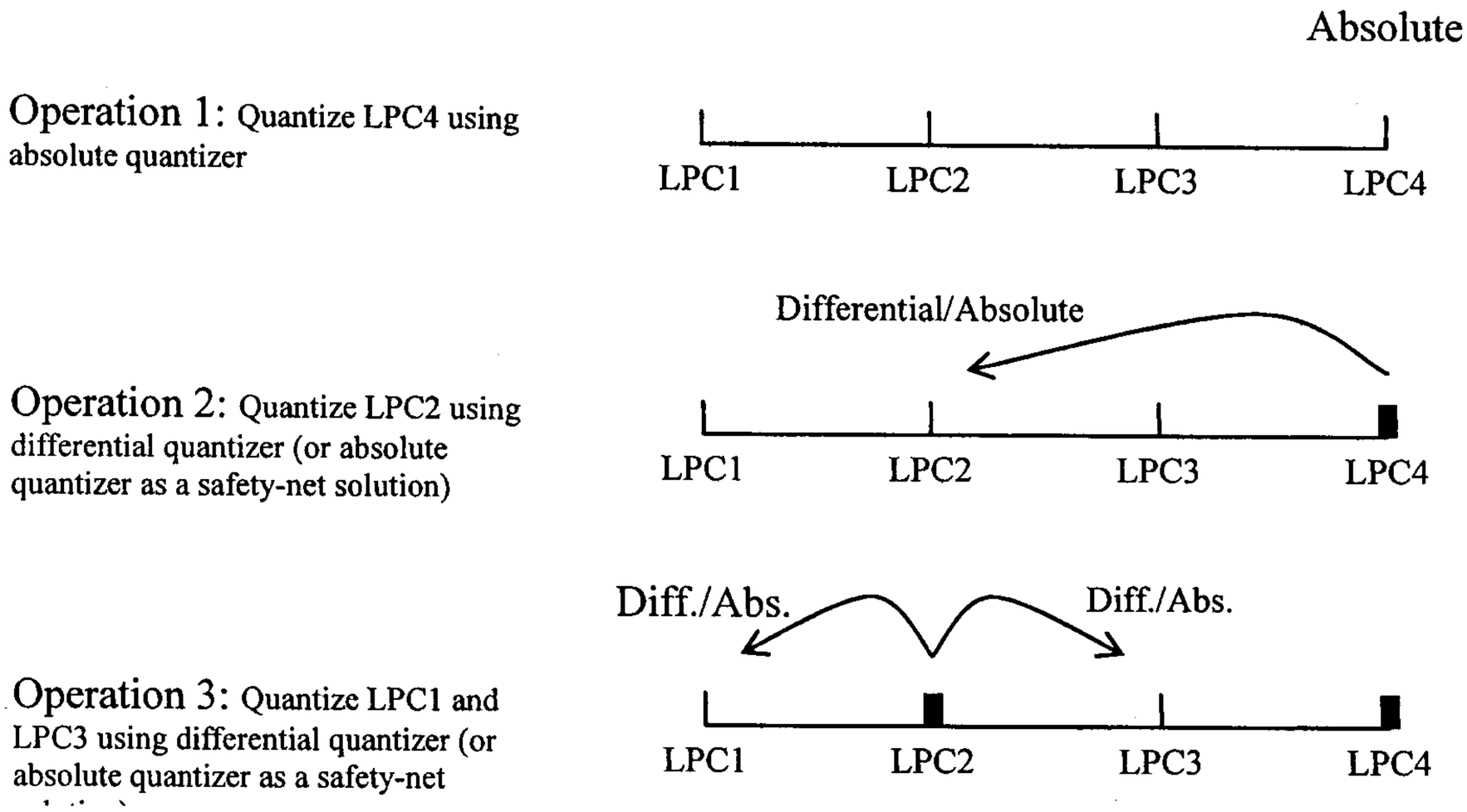

Absolute
Operation 1: Quantize LPC4 using absolute quantizer
LPC1
LPC2
LPC3
LPC4
Differential/Absolute
Operation 2: Quantize LPC2 using differential quantizer (or absolute quantizer as a safety-net solution)
LPC1
LPC2
LPC3
LPC4
Diff./Abs.
Diff./Abs.
Operation 3: Quantize LPC1 and LPC3 using differential quantizer (or absolute quantizer as a safety-net
LPC1
LPC2
LPC3
LPC4

Figure 2

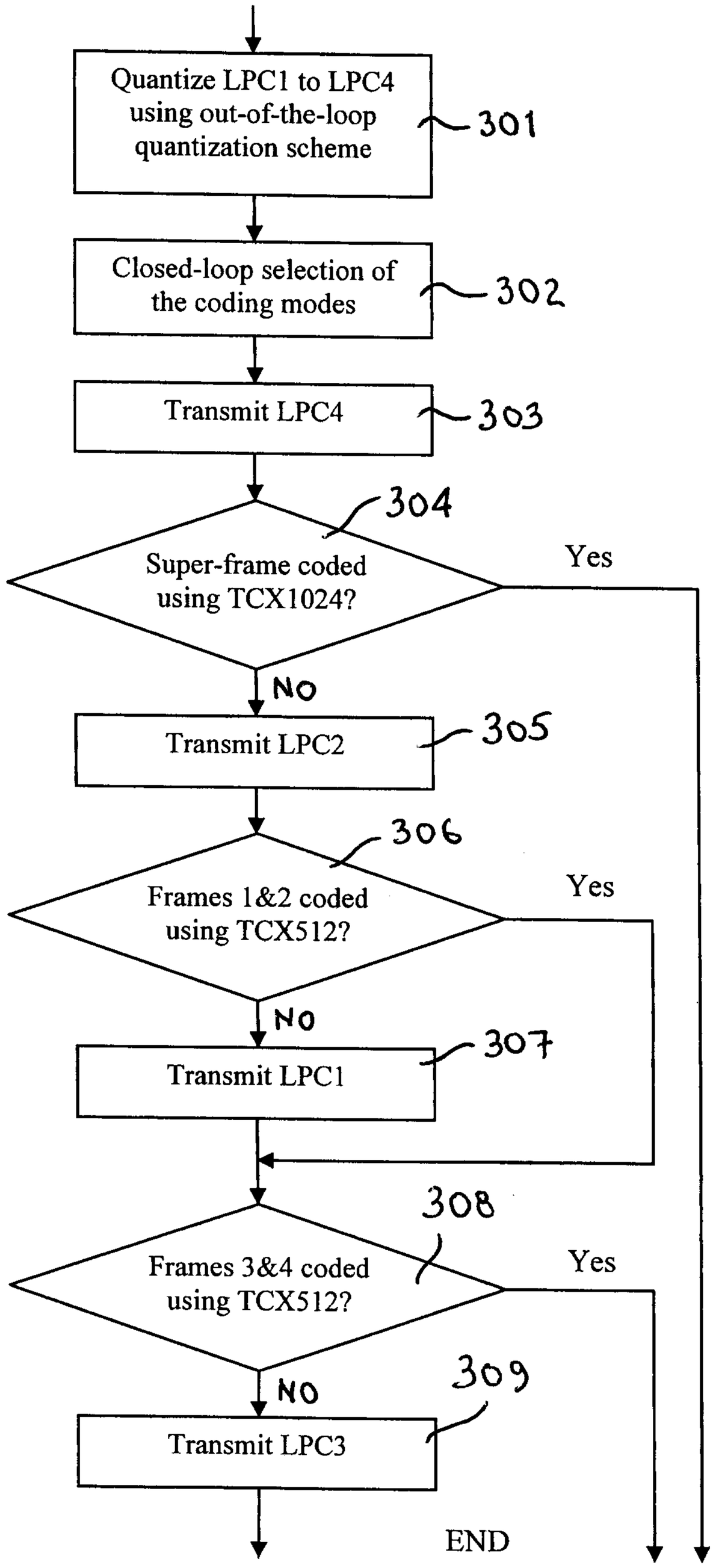

Quantize LPC1 to LPC4 using out-of-the-loop quantization scheme
301
Closed-loop selection of the coding modes
302
Transmit LPC4
303
304
Super-frame coded using TCX1024?
Yes
NO
Transmit LPC2
305
306
Frames 1&2 coded using TCX512?
Yes
NO
Transmit LPC1
307
308
Frames 3&4 coded using TCX512?
Yes
NO
309
Transmit LPC3
END

Figure 3

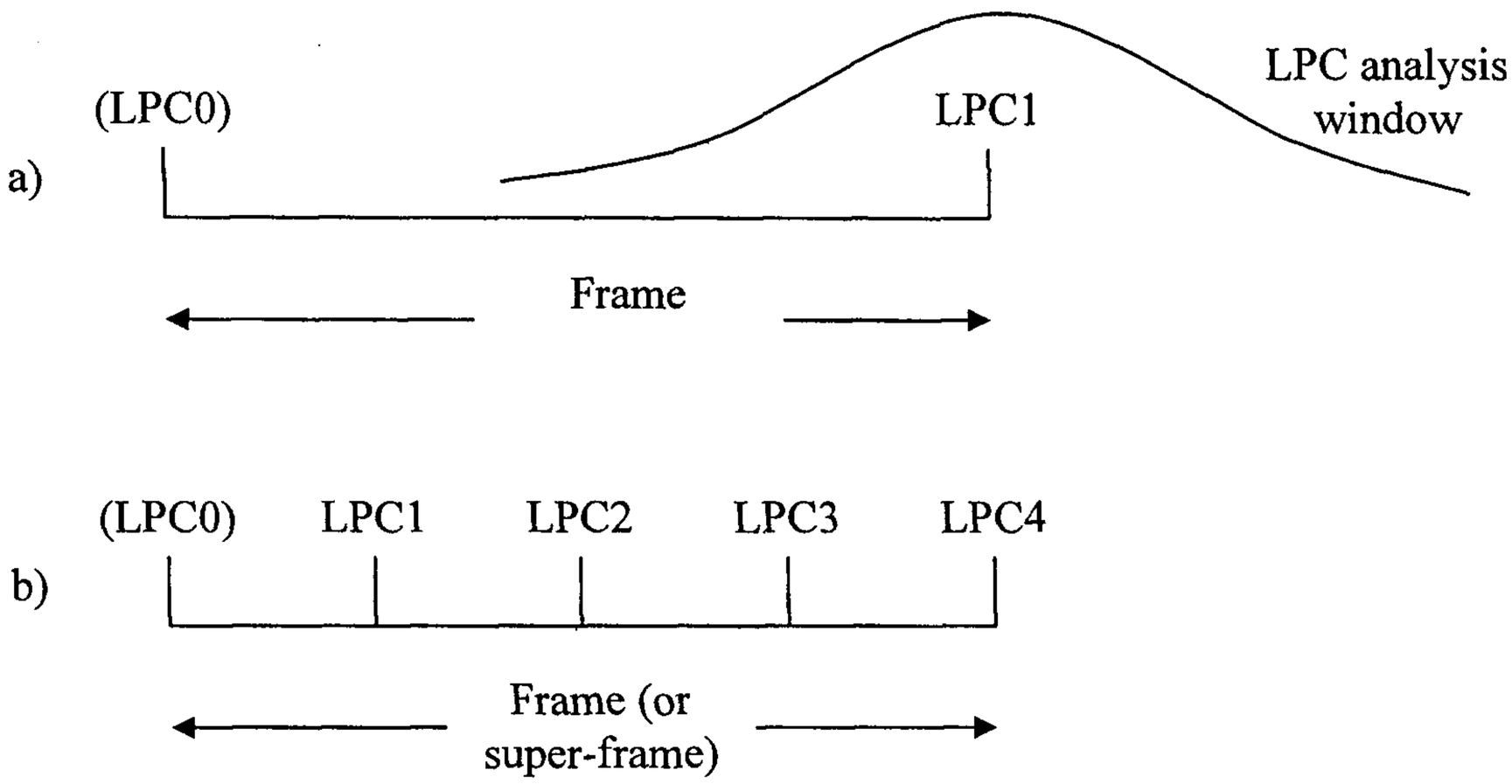

a)
(LPC0)
LPC1
LPC analysis window
Frame
b)
(LPC0)
LPC1
LPC2
LPC3
LPC4
Frame (or super-frame)

Figure 4

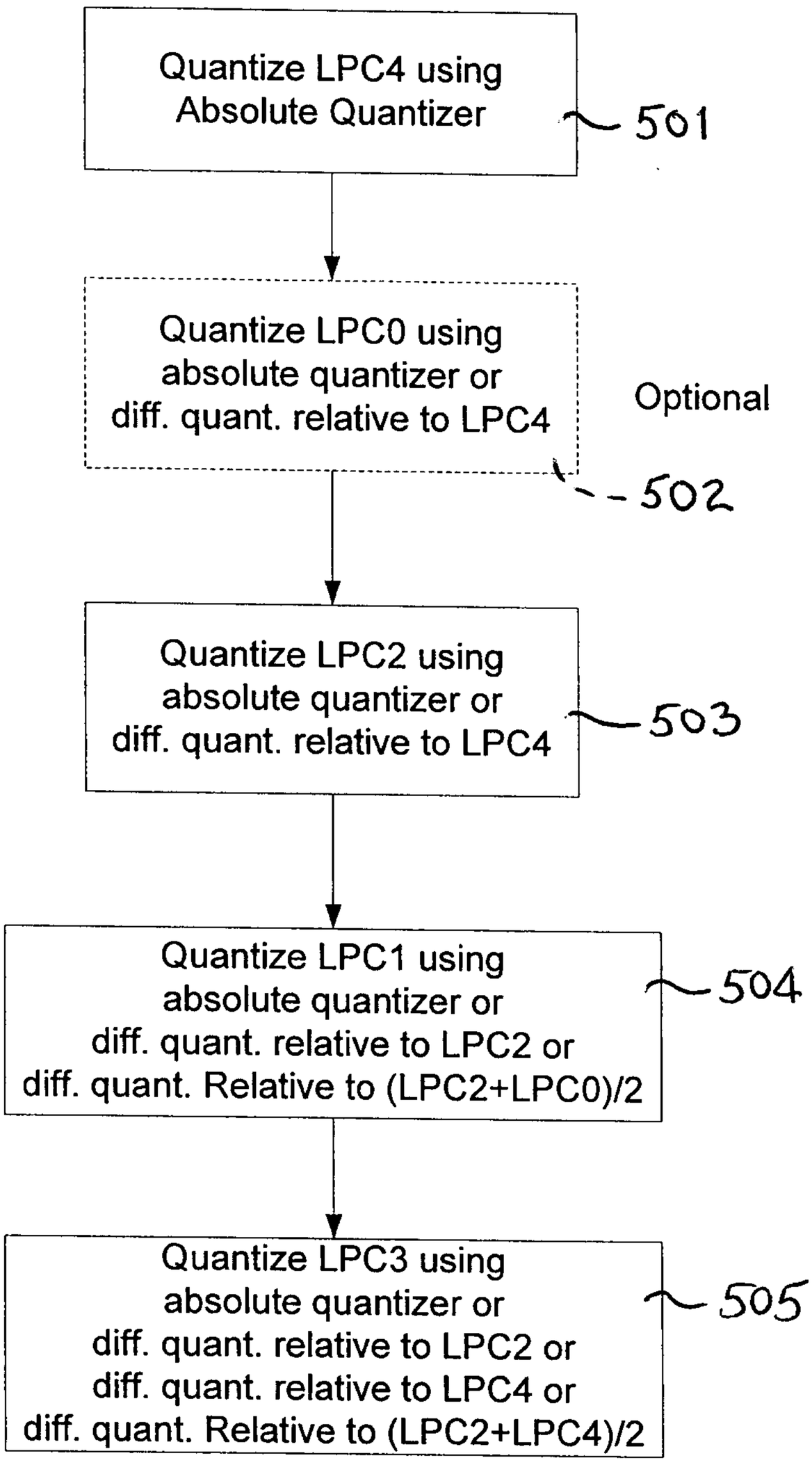

Quantize LPC4 using Absolute Quantizer
501
Quantize LPC0 using absolute quantizer or diff. quant. relative to LPC4
Optional
502
Quantize LPC2 using absolute quantizer or diff. quant. relative to LPC4
503
Quantize LPC1 using absolute quantizer or diff. quant. relative to LPC2 or diff. quant. Relative to (LPC2+LPC0)/2
504
Quantize LPC3 using absolute quantizer or diff. quant. relative to LPC2 or diff. quant. relative to LPC4 or diff. quant. Relative to (LPC2+LPC4)/2
505

Figure 5

Figure 6

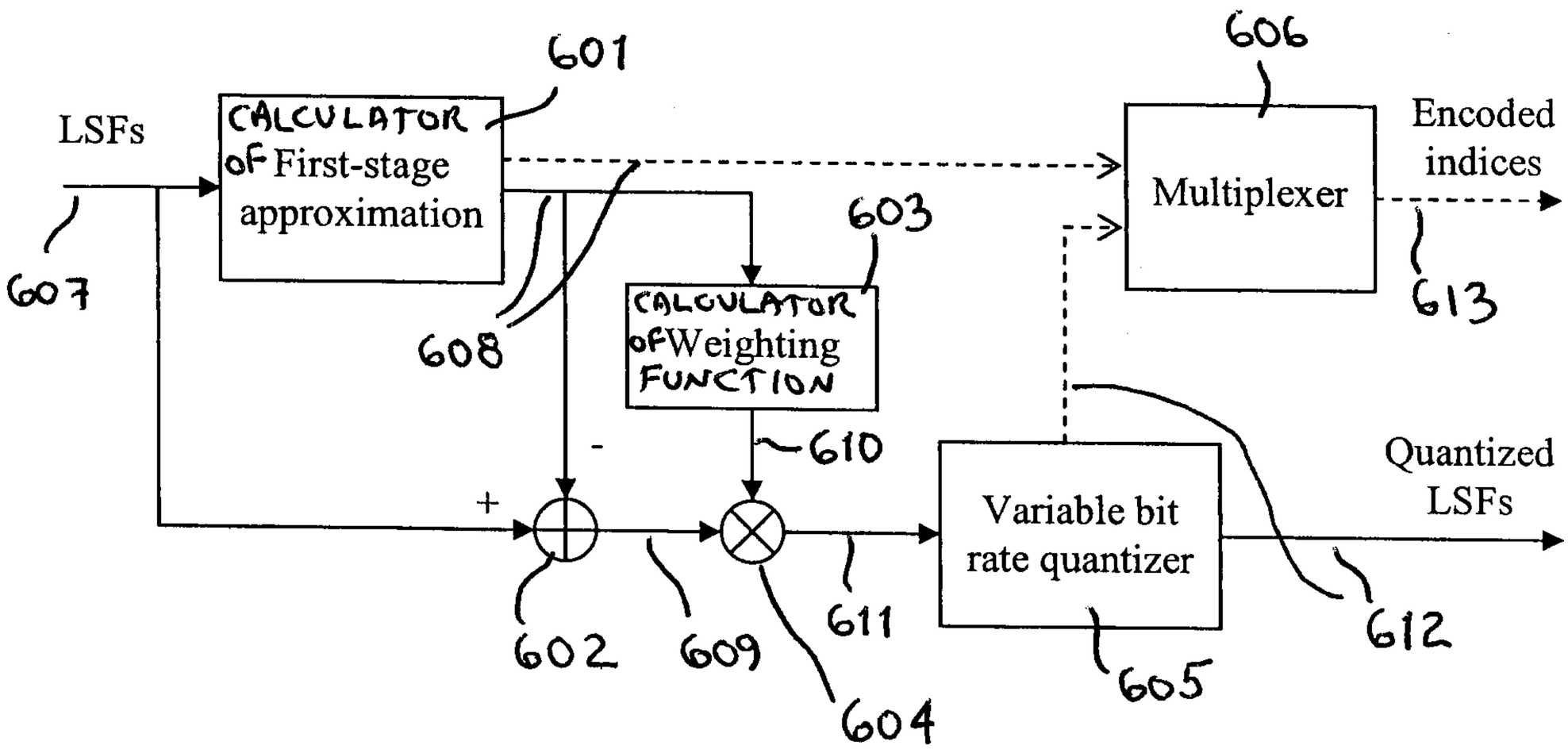
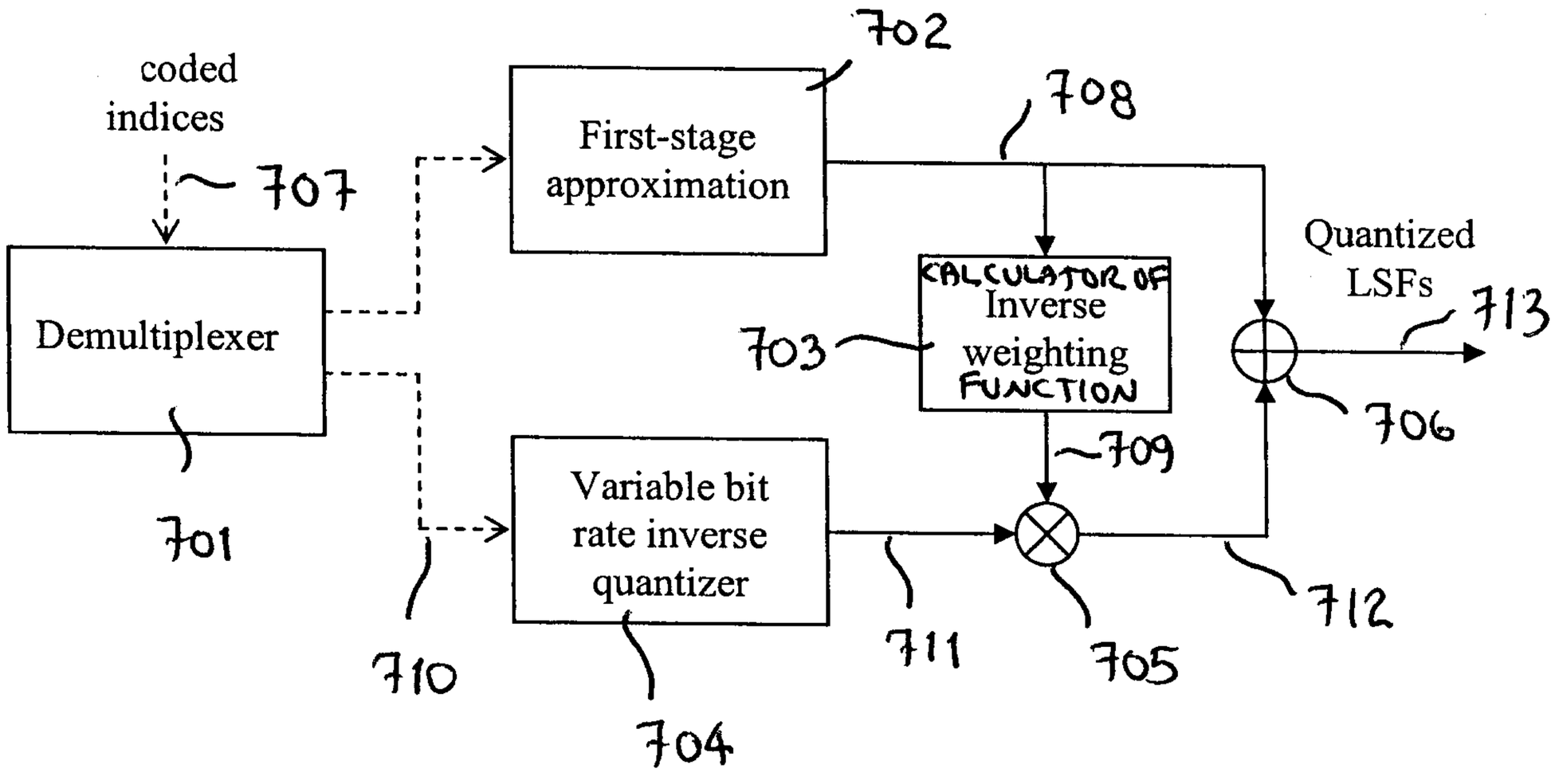


Figure 7



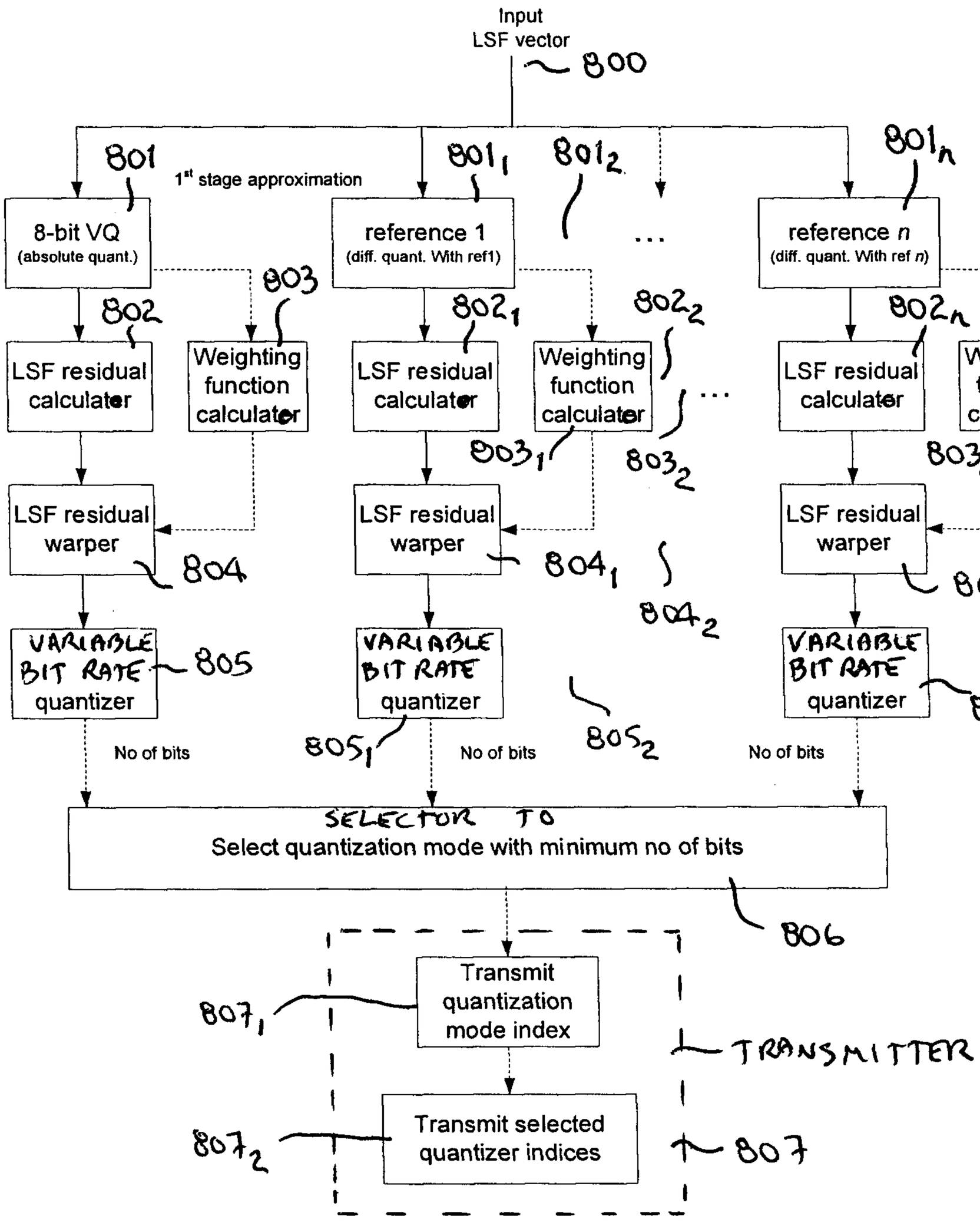

Input
LSF vector
800
801
1st stage approximation
801₁
801₂
...
801ₙ
8-bit VQ
(absolute quant.)
reference 1
(diff. quant. With ref1)
reference n
(diff. quant. With ref n)
802
803
802₁
802₂
802ₙ
LSF residual calculater
Weighting function calculater
803₁
803₂
803ₙ
LSF residual warper
804
804₁
804₂
804ₙ
VARIABLE BIT RATE quantizer
805
805₁
805₂
805ₙ
No of bits
SELECTOR TO
Select quantization mode with minimum no of bits
806
Transmit quantization mode index
807₁
Transmit selected quantizer indices
807₂
TRANSMITTER
807

Figure 8

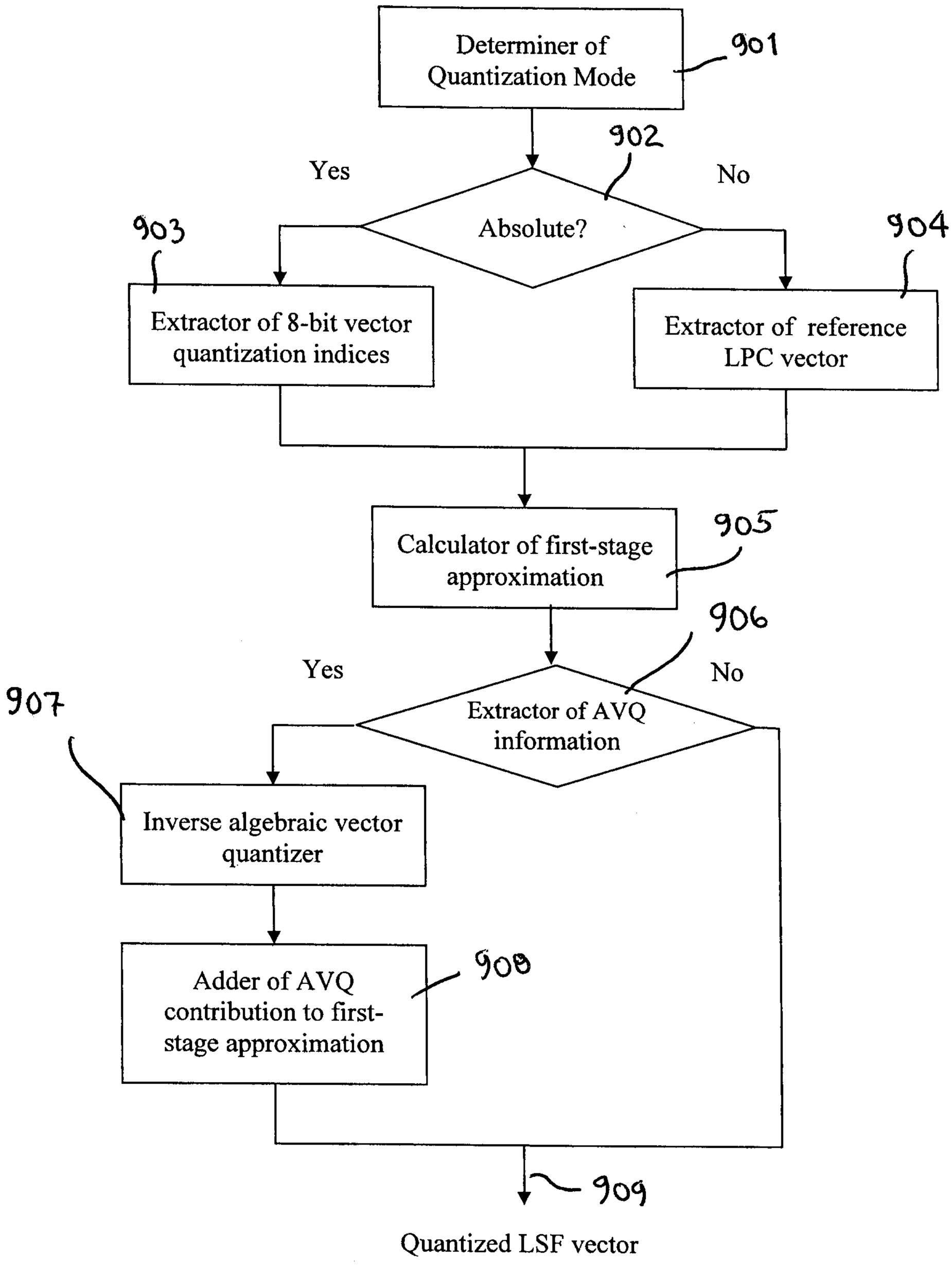
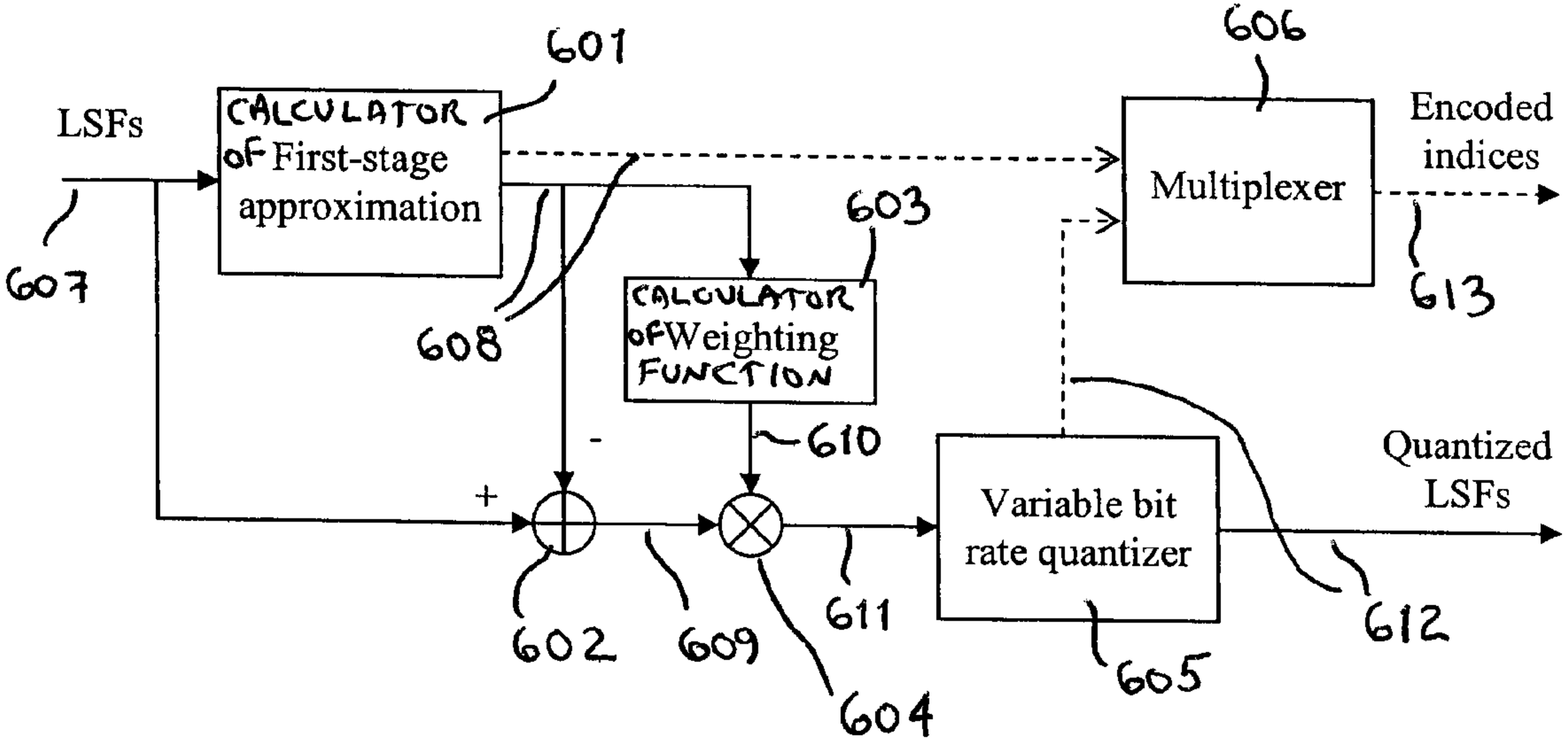

Determiner of Quantization Mode
901
902
Yes
No
Absolute?
903
Extractor of 8-bit vector quantization indices
904
Extractor of reference LPC vector
Calculator of first-stage approximation
905
906
Yes
No
Extractor of AVQ information
907
Inverse algebraic vector quantizer
Adder of AVQ contribution to first-stage approximation
908
909
Quantized LSF vector

Figure 9


LSFs
607
CALCULATOR OF First-stage approximation
601
608
CALCULATOR OF Weighting FUNCTION
603
610
606
Multiplexer
Encoded indices
613
+
-
602
609
611
Variable bit rate quantizer
605
612
Quantized LSFs
604