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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 August 2012 (23.08.2012)

WIPO | PCT

(10) International Publication Number
WO 2012/110476 A1

(51) International Patent Classification:
G10L 19/06 (2006.01) *G10L 19/14* (2006.01)

(21) International Application Number:
PCT/EP2012/052455

(22) International Filing Date:
14 February 2012 (14.02.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/442,632 14 February 2011 (14.02.2011) US

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

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **MARKOVIC, Goran** [RS/DE]; Aachener Straße 19, 90425 Nürnberg (DE). **FUCHS, Guillaume** [FR/DE]; Fürther Straße 17, 91058 Erlangen (DE). **RETTELBACH, Nikolaus** [DE/DE]; Spessartstr. 38, 90427 Nürnberg (DE). **HELMRICH, Christian** [DE/DE]; Hauptstraße 68, 91054 Erlangen (DE). **SCHUBERT, Benjamin** [DE/DE]; Zickstrasse 6, 90429 Nürnberg (DE).

(74) Agents: SCHENK, Markus et al.; P. O. Box 246, 82043 Pullach (DE).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, 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, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH,

[Continued on next page]

(54) Title: LINEAR PREDICTION BASED CODING SCHEME USING SPECTRAL DOMAIN NOISE SHAPING

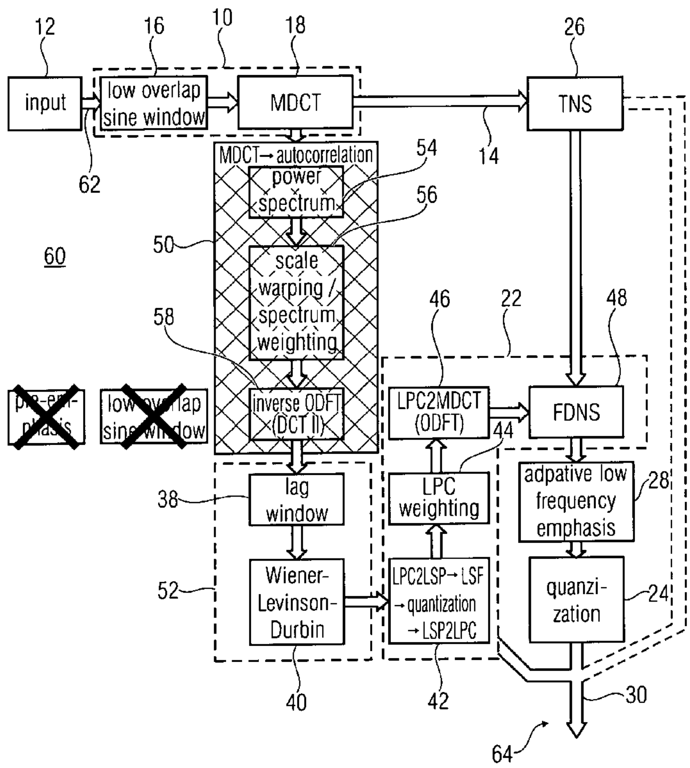


FIG 2

(57) Abstract: An encoding concept which is linear prediction based and uses spectral domain noise shaping is rendered less complex at a comparable coding efficiency in terms of, for example, rate/distortion ratio, by using the spectral decomposition of the audio input signal into a spectrogram comprising a sequence of spectra for both linear prediction coefficient computation as well as spectral domain shaping based on the linear prediction coefficients. The coding efficiency may remain even if such a lapped transform is used for the spectral decomposition which causes aliasing and necessitates time aliasing cancellation such as critically sampled lapped transforms such as an MDCT.

WO 2012/110476 A1

GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

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

Linear Prediction Based Coding Scheme Using Spectral Domain Noise Shaping

Description

The present invention is concerned with a linear prediction based audio codec using frequency domain noise shaping such as the TCX mode known from USAC.

As a relatively new audio codec, USAC has recently been finalized. USAC is a codec which supports switching between several coding modes such as an AAC like coding mode, a time-domain coding mode using linear prediction coding, namely ACELP, and transform coded excitation coding forming an intermediate coding mode according to which spectral domain shaping is controlled using the linear prediction coefficients transmitted via the data stream. In WO 2011147950, a proposal has been made to render the USAC coding scheme more suitable for low delay applications by excluding the AAC like coding mode from availability and restricting the coding modes to ACELP and TCX only. Further, it has been proposed to reduce the frame length.

However, it would be favorable to have a possibility at hand to reduce the complexity of a linear prediction based coding scheme using spectral domain shaping while achieving similar coding efficiency in terms of, for example, rate/distortion ratio sense.

Thus, it is an object of the present invention to provide such a linear prediction based coding scheme using spectral domain shaping allowing for a complexity reduction at a comparable or even increased coding efficiency.

This object is achieved by the subject matter of the pending independent claims.

It is a basic idea underlying the present invention that an encoding concept which is linear prediction based and uses spectral domain noise shaping may be rendered less complex at a comparable coding efficiency in terms of, for example, rate/distortion ratio, if the spectral decomposition of the audio input signal into a spectrogram comprising a sequence of spectra is used for both linear prediction coefficient computation as well as the input for a spectral domain shaping based on the linear prediction coefficients.

In this regard, it has been found out that the coding efficiency remains even if such a lapped transform is used for the spectral decomposition which causes aliasing and necessitates time aliasing cancellation such as critically sampled lapped transforms such as an MDCT.

Advantageous implementations of aspects of the present invention are subject of the dependent claims.

In particular, preferred embodiments of the present application are described with respect to the figures, among which

Fig. 1 shows a block diagram of an audio encoder in accordance with a comparison or embodiment;

Fig. 2 shows an audio encoder in accordance with an embodiment of the present application;

Fig. 3 shows a block diagram of a possible audio decoder fitting to the audio encoder of Fig. 2; and

Fig. 4 shows a block diagram of an alternative audio encoder in accordance with an embodiment of the present application.

In order to ease the understanding of the main aspects and advantages of the embodiments of the present invention further described below, reference is preliminarily made to Fig. 1 which shows a linear prediction based audio encoder using spectral domain noise shaping.

In particular, the audio encoder of Fig. 1 comprises a spectral decomposer 10 for spectrally decomposing an input audio signal 12 into a spectrogram consisting of a sequence of spectra, which is indicated at 14 in Fig. 1. As is shown in Fig. 1, the spectral decomposer 10 may use an MDCT in order to transfer the input audio signal 10 from time domain to spectral domain. In particular, a windower 16 precedes the MDCT module 18 of the spectral decomposer 10 so as to window mutually overlapping portions of the input audio signal 12 which windowed portions are individually subject to the respective transform in the MDCT module 18 so as to obtain the spectra of the sequence of spectra of spectrogram 14. However, spectral decomposer 10 may, alternatively, use any other lapped transform causing aliasing such as any other critically sampled lapped transform.

Further, the audio encoder of Fig. 1 comprises a linear prediction analyzer 20 for analyzing the input audio signal 12 so as to derive linear prediction coefficients therefrom. A spectral domain shaper 22 of audio encoder of Fig. 1 is configured to spectrally shape a current spectrum of the sequence of spectra of spectrogram 14 based on the linear prediction coefficients provided by linear prediction analyzer 20. In particular, the spectral domain shaper 22 is configured to spectrally shape a current spectrum entering the spectral domain shaper 22 in accordance with a transfer function which corresponds to a linear prediction analysis filter transfer function by converting the linear prediction coefficients from analyzer 20 into spectral weighting values and applying the latter weighting values as divisors so as to spectrally form or shape the current spectrum. The shaped spectrum is subject to quantization in a quantizer 24 of audio encoder of Fig. 1. Due to the shaping in the spectral domain shaper 22, the quantization noise which results upon de-shaping the quantized spectrum at the decoder side, is shifted so as to be hidden, i.e. the coding is as perceptually transparent as possible.

For sake of completeness only, it is noted that a temporal noise shaping module 26 may optionally subject the spectra forwarded from spectral decomposer 10 to spectral domain shaper 22 to a temporal noise shaping, and a low frequency emphasis module 28 may adaptively filter each shaped spectrum output by spectral domain shaper 22 prior to quantization 24.

The quantized and spectrally shaped spectrum is inserted into the data stream 30 along with information on the linear prediction coefficients used in spectral shaping so that, at the decoding side, the de-shaping and de-quantization may be performed.

The most parts of the audio codec, one exception being the TNS module 26, shown in Fig. 1 are, for example, embodied and described in the new audio codec USAC and in particular, within the TCX mode thereof. Accordingly, for further details, reference is made, exemplarily, to the USAC standard, for example [1].

Nevertheless, more emphasis is provided in the following with regard to the linear prediction analyzer 20. As is shown in Fig. 1, the linear prediction analyzer 20 directly operates on the input audio signal 12. A pre-emphasis module 32 pre-filters the input audio signal 12 such as, for example, by FIR filtering, and thereinafter, an autocorrelation is continuously derived by a concatenation of a windower 34, autocorrelator 36 and lag windower 38. Windower 34 forms windowed portions out of the pre-filtered input audio signal which windowed portions may mutually overlap in time. Autocorrelator 36 computes an autocorrelation per windowed portion output by windower 34 and lag windower 38 is optionally provided to apply a lag window function onto the autocorrelations so as to render the autocorrelations more suitable for the following linear prediction parameter estimate algorithm. In particular, a linear prediction parameter estimator 40 receives the lag window output and performs, for example, a Wiener-Levinson-Durbin or other suitable algorithm onto the windowed autocorrelations so as to derive linear prediction coefficients per autocorrelation. Within the spectral domain shaper 22, the resulting linear prediction coefficients are passed through a chain of modules 42, 44, 46 and 48. The module 42 is responsible for transferring information on the linear prediction coefficients within the data stream 30 to the decoding side. As shown in Fig. 1, the linear prediction coefficient data stream inserter 42 may be configured to perform a quantization of the linear prediction coefficients determined by linear prediction analyzer 20 in a line spectral pair or line spectral frequency domain with coding the quantized coefficients into data stream 30 and re-converting the quantized prediction values into LPC coefficients again. Optionally, some interpolation may be used in order to reduce an update rate at which information onto the linear prediction coefficients is conveyed within data stream 30. Accordingly, the subsequent module 44 which is responsible for subjecting the linear prediction coefficients concerning the current spectrum entering the spectral domain shaper 22 to some weighting process, has access to linear prediction coefficients as they are also available at the decoding side, i.e. access to the quantized linear prediction coefficients. A subsequent module 46, converts the weighted linear prediction coefficients to spectral weightings which are then applied by the frequency domain noise shaper module 48 so as to spectrally shape the inbound current spectrum.

As became clear from the above discussion, the linear prediction analysis performed by analyzer 20 causes overhead which completely adds-up to the spectral decomposition and the spectral domain shaping performed in blocks 10 and 22 and accordingly, the computational overhead is considerable.

Fig. 2 shows an audio encoder according to an embodiment of the present application which offers comparable coding efficiency, but has reduced coding complexity.

Briefly spoken, in the audio encoder of Fig. 2 which represents an embodiment of the present application, the linear prediction analyzer of Fig. 1 is replaced by a concatenation of an autocorrelation computer 50 and a linear prediction coefficient computer 52 serially connected between spectral decomposer 10 and spectral domain shaper 22. The motivation for the modification from Fig. 1 to Fig. 2 and the mathematical explanation which reveals the detailed functionality of modules 50 and 52 will be provided in the following. However, it is obvious that the computational overhead of the audio encoder of Fig. 2 is reduced compared to the audio encoder of Fig. 1 considering that the autocorrelation computer 50 involves less complex computations when compared to a sequence of computations involved with the autocorrelation and the windowing prior to the autocorrelation.

Before describing the detailed and mathematical framework of the embodiment of Fig. 2, the structure of the audio encoder of Fig. 2 is briefly described. In particular, the audio encoder of Fig. 2 which is generally indicated using reference sign 60 comprises an input 62 for receiving the input audio signal 12 and an output 64 for outputting the data stream 30 into which the audio encoder encodes the input audio signal 12. Spectral decomposer 10, temporal noise shaper 26, spectral domain shaper 22, low frequency emphasizer 28 and quantizer 24 are connected in series in the order of their mentioning between input 62 and output 64. Temporal noise shaper 26 and low frequency emphasizer 28 are optional modules and may, in accordance with an alternative embodiment, be left away. If present, the temporal noise shaper 26 may be configured to be activatable adaptively, i.e. the temporal noise shaping by temporal noise shaper 26 may be activated or deactivated depending on the input audio signal's characteristic, for example, with a result of the decision being, for example, transferred to the decoding side via data stream 30 as will be explained in more detail below.

As shown in Fig. 1, the spectral domain shaper 22 of Fig. 2 is internally constructed as it has been described with respect to Fig. 1. However, the internal structure of Fig. 2 is not to be interpreted as a critical issue and the internal structure of the spectral domain shaper 22 may also be different when compared to the exact structure shown in Fig. 2.

The linear prediction coefficient computer 52 of Fig. 2 comprises the lag windower 38 and the linear prediction coefficient estimator 40 which are serially connected between the autocorrelation computer 50 on the one hand and the spectral domain shaper 22 on the other hand. It should be noted that the lag windower, for example, is also an optional feature. If present, the window applied by lag windower 38 on the individual autocorrelations provided by autocorrelation computer 50 could be a Gaussian or binomial shaped window. With regard to the linear prediction coefficient estimator 40, it is noted that same not necessarily uses the Wiener-Levinson-Durbin algorithm. Rather, a different algorithm could be used in order to compute the linear prediction coefficients.

Internally, the autocorrelation computer 50 comprises a sequence of a power spectrum computer 54 followed by a scale warper/spectrum weighter 56 which in turn is followed by an inverse transformer 58. The details and significance of the sequence of modules 54 to 58 will be described in more detail below.

In order to understand as to why it is possible to co-use the spectral decomposition of decomposer 10 for both, spectral domain noise shaping within shaper 22 as well as linear prediction coefficient computation, one should consider the Wiener-Khinichin Theorem which shows that an autocorrelation can be calculated using a DFT:

$$R_m = \frac{1}{N}\sum_{k=0}^{N-1} S_k e^{\frac{2\pi i}{N}km} \qquad m = 0,\ldots,N-1$$

where

$$S_k = X_k X_k^*$$
$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn}$$
$$R_m = E(x_n x_{n-m}^*)$$
$$k = 0,\ldots,N-1 \qquad m = 0,\ldots,N-1$$

Thus, R_m are the autocorrelation coefficients of the autocorrelation of the signal's portion x_n of which the DFT is X_k.

Accordingly, if spectral decomposer 10 would use a DFT in order to implement the lapped transform and generate the sequence of spectra of the input audio signal 12, then autocorrelation calculator 50 would be able to perform a faster calculation of an autocorrelation at its output, merely by obeying the just outlined Wiener-Khinichin Theorem.

If the values for all lags m of the autocorrelation are required, the DFT of the spectral decomposer 10 could be performed using an FFT and an inverse FFT could be used within the autocorrelation computer 50 so as to derive the autocorrelation therefrom using the just mentioned formula. When, however, only M<<N lags are needed, it would be faster to use an FFT for the spectral decomposition and directly apply an inverse DFT so as to obtain the relevant autocorrelation coefficients.

The same holds true when the DFT mentioned above is replaced with an ODFT, i.e. odd frequency DFT, where a generalized DFT of a time sequence x is defined as:

$$X_k^{odft} = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N(k+b)(n+a)}} \qquad k = 0,\ldots,N-1$$

and

$$a = 0 \qquad b = \frac{1}{2}$$

is set for ODFT (Odd Frequency DFT).

If, however, an MDCT is used in the embodiment of Fig. 2, rather than a DFT or FFT, things differ. The MDCT involves a discrete cosine transform of type IV and only reveals a real-valued spectrum. That is, phase information gets lost by this transformation. The MDCT can be written as:

$$X_k = \sum_{n=0}^{2N-1} x_n \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2} + \frac{N}{2}\right)\left(k + \frac{1}{2}\right)\right] \qquad k = 0, \ldots, N-1$$

where x_n with n = 0 … 2N-1 defines a current windowed portion of the input audio signal 12 as output by windower 16 and X_k is, accordingly, the k-th spectral coefficient of the resulting spectrum for this windowed portion.

The power spectrum computer 54 calculates from the output of the MDCT the power spectrum by squaring each transform coefficient X_k according to:

$$S_k = |X_k|^2 \qquad k = 0, \ldots, N-1$$

The relation between an MDCT spectrum as defined by X_k and an ODFT spectrum X_k^{ODFT} can be written as:

$$X_k = Re\left(X_k^{odft}\right)\cos(\theta_k) + Im\left(X_k^{odft}\right)\sin(\theta_k) \qquad k = 0, \ldots, N-1$$

$$\theta_k = \frac{\pi}{N}\left(\frac{1}{2} + \frac{N}{2}\right)\left(k + \frac{1}{2}\right)$$

$$|X_k| = \left|X_k^{odft}\right|\left|\cos\left[arg\left(X_k^{odft}\right) - \theta_k\right]\right|$$

This means that using the MDCT instead of an ODFT as input for the autocorrelation computer 50 performing the MDCT to autocorrelation procedure, is equivalent to the autocorrelation obtained from the ODFT with a spectrum weighting of

$$f_k^{mdct} = \left|\cos\left[\arg\left(X_k^{odft}\right) - \theta_k\right]\right|$$

This distortion of the autocorrelation determined is, however, transparent for the decoding side as the spectral domain shaping within shaper 22 takes place in exactly the same spectral domain as the one of the spectral decomposer 10, namely the MDCT. In other words, since the frequency domain noise shaping by frequency domain noise shaper 48 of Fig. 2 is applied in the MDCT domain, this effectively means that the spectrum weighting f_k^{mdct} cancels out the modulation of the MDCT and produces similar results as a conventional LPC as shown in Fig. 1 would produce when the MDCT would be replaced with an ODFT.

Accordingly, in the autocorrelation computer 50, the inverse transformer 58 performs an inverse ODFT and an inverse ODFT of a symmetrical real input is equal to a DCT type II:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right]$$

Thus, this allows a fast computation of the MDCT based LPC in the autocorrelation computer 50 of Fig. 2, as the autocorrelation as determined by the inverse ODFT at the output of inverse transformer 58 comes at a relatively low computational cost as merely minor computational steps are necessary such as the just outlined squaring and the power spectrum computer 54 and the inverse ODFT in the inverse transformer 58.

Details regarding the scale warper/spectrum weighter 56 have not yet been described. In particular, this module is optional and may be left away or replaced by a frequency domain decimator. Details regarding possible measures performed by module 56 are described in the following. Before that, however, some details regarding some of the other elements shown in Fig. 2 are outlined. Regarding the lag windower 38, for example, it is noted that same may perform a white noise compensation in order to improve the conditioning of the linear prediction coefficient estimation performed by estimator 40. The LPC weighting performed in module 44 is optional, but if present, it may be performed so as to achieve an actual bandwidth expansion. That is, poles of the LPCs are moved toward the origin by a constant factor according to, for example,

$$A'(z) = A\left(\frac{z}{\gamma}\right)$$

Thus, the LPC weighting thus performed approximates the simultaneous masking. A constant of $\gamma = 0.92$ or somewhere between 0.85 and 0.95, both inclusively, produces good results.

Regarding module 42 it is noted that variable bitrate coding or some other entropy coding scheme may be used in order to encode the information concerning the linear prediction coefficients into the data stream 30. As already mentioned above, the quantization could be performed in the LSP/LSF domain, but the ISP/ISF domain is also feasible.

Regarding the LPC-to-MDCT module 46 which converts the LPC into spectral weighting values which are called, in case of MDCT domain, MDCT gains in the following, reference is made, for example, to the USAC codec where this transform is explained in detail. Briefly spoken, the LPC coefficients may be subject to an ODFT so as to obtain MDCT gains, the inverse of which may then be used as weightings for shaping the spectrum in module 48 by applying the resulting weightings onto respective bands of the spectrum. For example, 16 LPC coefficients are converted into MDCT gains. Naturally, instead of weighting using the inverse, weighting using the MDCT gains in non-inverted form is used at the decoder side in order to obtain a transfer function resembling an LPC synthesis filter so as to form the quantization noise as already mentioned above. Thus, summarizing, in module 46, the gains used by the FDNS 48 are obtained from the linear prediction coefficients using an ODFT and are called MDCT gains in case of using MDCT.

For sake of completeness, Fig. 3 shows a possible implementation for an audio decoder which could be used in order to reconstruct the audio signal from the data stream 30 again. The decoder of Fig. 3 comprises a low frequency de-emphasizer 80, which is optional, a spectral domain deshaper 82, a temporal noise deshaper 84, which is also optional, and a spectral-to-time domain converter 86, which are serially connected between a data stream input 88 of the audio decoder at which the data stream 30 enters, and an output 90 of the audio decoder where the reconstructed audio signal is output. The low frequency de-emphasizer receives from the data stream 30 the quantized and spectrally shaped spectrum and performs a filtering thereon, which is inverse to the low frequency emphasizer's transfer function of Fig. 2. As already mentioned, de-emphasizer 80 is, however, optional.

The spectral domain deshaper 82 has a structure which is very similar to that of the spectral domain shaper 22 of Fig. 2. In particular, internally same comprises a concatenation of LPC extractor 92, LPC weighter 94, which is equal to LPC weighter 44, an LPC to MDCT converter 96, which is also equal to module 46 of Fig. 2, and a frequency domain noise shaper 98 which applies the MDCT gains onto the inbound (de-emphasized) spectrum inversely to FDNS 48 of Fig. 2, i.e. by multiplication rather than division in order to obtain a transfer function which corresponds to a linear prediction synthesis filter of the linear prediction coefficients extracted from the data stream 30 by LPC extractor 92. The LPC extractor 92 may perform the above mentioned retransform from a corresponding quantization domain such as LSP/LSF or ISP/ISF to obtain the linear prediction coefficients for the individual spectrums coded into data stream 30 for the consecutive mutually overlapping portions of the audio signal to be reconstructed.

The time domain noise shaper 84 reverses the filtering of module 26 of Fig. 2, and possible implementations for these modules are described in more detail below. In any case, however, TNS module 84 of Fig. 3 is optional and may be left away as has also been mentioned with regard to TNS module 26 of Fig. 2.

The spectral composer 86 comprises, internally, an inverse transformer 100 performing, for example, an IMDCT individually onto the inbound de-shaped spectra, followed by an aliasing canceller such as an overlap-add adder 102 configured to correctly temporally register the reconstructed windowed versions output by retransformer 100 so as to perform time aliasing cancellation between same and to output the reconstructed audio signal at output 90.

As already mentioned above, due to the spectral domain shaping 22 in accordance with a transfer function corresponding to an LPC analysis filter defined by the LPC coefficients conveyed within data stream 30, the quantization in quantizer 24, which has, for example, a spectrally flat noise, is shaped by the spectral domain deshaper 82 at a decoding side in a manner so as to be hidden below the masking threshold.

Different possibilities exist for implementing the TNS module 26 and the inverse thereof in the decoder, namely module 84. Temporal noise shaping is for shaping the noise in the temporal sense within the time portions which the individual spectra spectrally formed by the spectral domain shaper referred to. Temporal noise shaping is especially useful in case of transients being present within the respective time portion the current spectrum refers to. In accordance with a specific embodiment, the temporal noise shaper 26 is configured as a spectrum predictor configured to predictively filter the current spectrum or the sequence of spectra output by the spectral decomposer 10 along a spectral dimension. That is, spectrum predictor 26 may also determine prediction filter coefficients which may be inserted into the data stream 30. This is illustrated by a dashed line in Fig. 2. As a consequence, the temporal noise filtered spectra are flattened along the spectral dimension and owing to the relationship between spectral domain and time domain, the inverse filtering within the time domain noise deshaper 84 in accordance with the transmitted time domain noise shaping prediction filters within data stream 30, the deshaping leads to a hiding or compressing of the noise within the times or time at which the attack or transients occur. So called pre-echoes are thereby avoided.

In other words, by predictively filtering the current spectrum in time domain noise shaper 26, the time domain noise shaper 26 obtains as spectrum reminder, i.e. the predictively filtered spectrum which is forwarded to the spectral domain shaper 22, wherein the corresponding prediction coefficients are inserted into the data stream 30. The time domain noise deshaper 84, in turn, receives from the spectral domain deshaper 82 the de-shaped spectrum and reverses the time domain filtering along the spectral domain by inversely filtering this spectrum in accordance with the prediction filters received from data stream, or extracted from data stream 30. In other words, time domain noise shaper 26 uses an analysis prediction filter such as a linear prediction filter, whereas the time domain noise deshaper 84 uses a corresponding synthesis filter based on the same prediction coefficients.

As already mentioned, the audio encoder may be configured to decide to enable or disable the temporal-noise shaping depending on the filter prediction gain or a tonality or transiency of the audio input signal 12 at the respective time portion corresponding to the current spectrum. Again, the respective information on the decision is inserted into the data stream 30.

In the following, the possibility is discussed according to which the autocorrelation computer 50 is configured to compute the autocorrelation from the predictively filtered, i.e. TNS-filtered, version of the spectrum rather than the unfiltered spectrum as shown in Fig. 2. Two possibilities exist: the TNS-filtered spectrums may be used whenever TNS is applied, or in a manner chosen by the audio encoder based on, for example, characteristics of the input audio signal 12 to be encoded. Accordingly, the audio encoder of Fig. 4 differs from the audio encoder of Fig. 2 in that the input of the autocorrelation computer 50 is connected to both the output of the spectral decomposer 10 as well as the output of the TNS module 26.

As just mentioned, the TNS-filtered MDCT spectrum as output by spectral decomposer 10 can be used as an input or basis for the autocorrelation computation within computer 50. As just mentioned, the TNS-filtered spectrum could be used whenever TNS is applied, or the audio encoder could decide for spectra to which TNS was applied between using the unfiltered spectrum or the TNS-filtered spectrum. The decision could be made, as mentioned above, depending on the audio input signal's characteristics. The decision could be, however, transparent for the decoder, which merely applies the LPC coefficient information for the frequency domain deshaping. Another possibility would be that the audio encoder switches between the TNS-filtered spectrum and the non-filtered spectrum for spectrums to which TNS was applied, i.e. to make the decision between these two options for these spectrums, depending on a chosen transform length of the spectral decomposer 10.

To be more precise, the decomposer 10 in Fig. 4 may be configured to switch between different transform lengths in spectrally decomposing the audio input signal so that the spectra output by the spectral decomposer 10 would be of different spectral resolution. That is, spectral decomposer 10 would, for example, use a lapped transform such as the MDCT, in order to transform mutually overlapping time portions of different length onto transforms or spectrums of also varying length, with the transform length of the spectra corresponding to the length of the corresponding overlapping time portions. In that case, the autocorrelation computer 50 could be configured to compute the autocorrelation from the predictively filtered or TNS-filtered current spectrum in case of a spectral resolution of the current spectrum fulfilling a predetermined criterion, or from the not predictively filtered, i.e. unfiltered, current spectrum in case of the spectral resolution of the current spectrum not fulfilling the predetermined criterion. The predetermined criterion could be, for example, that the current spectrum's spectral resolution exceeds some threshold. For example, using the TNS-filtered spectrum as output by TNS module 26 for the autocorrelation computation is beneficial for longer frames (time portions) such as frames longer than 15 ms, but may be disadvantageous for short frames (temporal portions) being shorter than, for example, 15 ms, and accordingly, the input into the autocorrelation computer 50 for longer frames may be the TNS-filtered MDCT spectrum, whereas for shorter frames the MDCT spectrum as output by decomposer 10 may be used directly.

Until now it has not yet been described which perceptual relevant modifications could be performed onto the power spectrum within module 56. Now, various measures are explained, and they could be applied individually or in combination onto all embodiments and variants described so far. In particular, a spectrum weighting could be applied by module 56 onto the power spectrum output by power spectrum computer 54. The spectrum weighting could be:

$$S'_k = f_k^2 S_k \qquad k = 0, \ldots, N-1$$

wherein S_k are the coefficients of the power spectrum as already mentioned above.

Spectral weighting can be used as a mechanism for distributing the quantization noise in accordance with psychoacoustical aspects. Spectrum weighting corresponding to a pre-emphasis in the sense of Fig. 1 could be defined by:

$$f_k^{emph} = \sqrt{1 + \mu^2 - 2\mu\cos\left(\frac{k\pi}{N}\right)}.$$

Moreover, scale warping could be used within module 56. The full spectrum could be divided, for example, into M bands for spectrums corresponding to frames or time portions of a sample length of l_1 and 2M bands for spectrums corresponding to time portions of frames having a sample length of l_2, wherein l_2 may be two times l_1, wherein l_1 may be 64, 128 or 256. In particular, the division could obey:

$$E_m = \sum_{k=I_m}^{I_{m+1}-1} S_k \qquad m = 0, \ldots, M-1 .$$

The band division could include frequency warping to an approximation of the Bark scale according to:

$$I_m \approx \frac{NF_s}{2Bark2Freq\left[m\frac{Freq2Bark\left(\frac{F_s}{2}\right)}{M}\right]},$$

alternatively the bands could be equally distributed to form a linear scale according to:

$$I_m = m\frac{N}{M}.$$

For the spectrums of frames of length l_1, for example, a number of bands could be between 20 and 40, and between 48 and 72 for spectrums belonging to frames of length l_2, wherein 32 bands for spectrums of frames of length l_1 and 64 bands for spectrums of frames of length l_2 are preferred.

Spectral weighting and frequency warping as optionally performed by optional module 56 could be regarded as a means of bit allocation (quantization noise shaping). Spectrum weighting in a linear scale corresponding to the pre-emphasis could be performed using a constant $\mu = 0.9$ or a constant lying somewhere between 0.8 and 0.95, so that the corresponding pre-emphasis would approximately correspond to Bark scale warping.

Modification of the power spectrum within module 56 may include spreading of the power spectrum, modeling the simultaneous masking, and thus replace the LPC Weighting modules 44 and 94.

If a linear scale is used and the spectrum weighting corresponding to the pre-emphasis is applied, then the results of the audio encoder of Fig. 4 as obtained at the decoding side, i.e. at the output of the audio decoder of Fig. 3, are perceptually very similar to the conventional reconstruction result as obtained in accordance with the embodiment of Fig. 1.

Some listening test results have been performed using the embodiments identified above. From the tests, it turned out that the conventional LPC analysis as shown in Fig. 1 and the linear scale MDCT based LPC analysis produced perceptually equivalent results when

- The spectrum weighting in the MDCT based LPC analysis corresponds to the pre-emphasis in the conventional LPC analysis,
- The same windowing is used within the spectral decomposition, such as a low overlap sine window, and
- The linear scale is used in the MDCT based LPC analysis.

The negligible difference between the conventional LPC analysis and the linear scale MDCT based LPC analysis probably comes from the fact that the LPC is used for the quantization noise shaping and that there are enough bits at 48 kbit/s to code MDCT coefficients precisely enough.

Further, it turned out that using the Bark scale or non-linear scale by applying scale warping within module 56 results in coding efficiency or listening test results according to which the Bark scale outperforms the linear scale for the test audio pieces Applause, Fatboy, RockYou, Waiting, bohemian, fuguepremikres, kraftwerk, lesvoleurs, teardrop.

The Bark scale fails miserably for hockey and linchpin. Another item that has problems in the Bark scale is bibilolo, but it wasn't included in the test as it presents an experimental music with specific spectrum structure. Some listeners also expressed strong dislike of the bibilolo item.

However, it is possible for the audio encoder of Figs. 2 and 4 to switch between different scales. That is, module 56 could apply different scaling for different spectrums in dependency on the audio signal's characteristics such as the transiency or tonality or use different frequency scales to produce multiple quantized signals and a measure to determine which of the quantized signals is perceptually the best. It turned out that scale switching results in improvements in the presence of transients such as the transients in RockYou and linchpin when compared to both non-switched versions (Bark and linear scale).

It should be mentioned that the above outlined embodiments could be used as the TCX mode in a multi-mode audio codec such as a codec supporting ACELP and the above outlined embodiment as a TCX-like mode. As a framing, frames of a constant length such as 20 ms could be used. In this way, a kind of low delay version of the USAC codec could be obtained which is very efficient. As the TNS, the TNS from AAC-ELD could be used. To reduce the number of bits used for side information, the number of filters could be fixed to two, one operating from 600 Hz to 4500 Hz and a second from 4500 Hz to the end of the core coder spectrum. The filters could be independently switched on and off. The filters could be applied and transmitted as a lattice using parcor coefficients. The maximum order of a filter could be set to be eight and four bits could be used per filter coefficient. Huffman coding could be used to reduce the number of bits used for the order of a filter and for its coefficients.

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

Depending on certain implementation requirements, embodiments of the invention can be implemented in hardware or in software. The implementation can be performed using a digital storage medium, for example a floppy disk, a DVD, a Blu-Ray, a CD, a ROM, a PROM, an EPROM, an EEPROM or a FLASH memory, having electronically readable control signals stored thereon, which cooperate (or are capable of cooperating) with a programmable computer system such that the respective method is performed. Therefore, the digital storage medium may be computer readable.

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

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

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

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

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

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

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

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

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

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

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

Literature:

[1]: USAC codec (Unified Speech and Audio Codec), ISO/IEC CD 23003-3 dated September 24, 2010

Claims

1. Audio encoder comprising

a spectral decomposer (10) for spectrally decomposing an audio input signal (12) into a spectrogram (14) of a sequence of spectrums;

an autocorrelation computer (50) configured to compute an autocorrelation from a current spectrum of the sequence of spectrums;

a linear prediction coefficient computer (52) configured to compute linear prediction coefficients based on the autocorrelation;

a spectral domain shaper (22) configured to spectrally shape the current spectrum based on the linear prediction coefficients; and

a quantization stage (24) configured to quantize the spectrally shaped spectrum;

wherein the audio encoder is configured to insert information on the quantized spectrally shaped spectrum and information on the linear prediction coefficients into a data stream.

2. Audio encoder according to claim 1, further comprising

a spectrum predictor (26) configured to predictively filter the current spectrum along a spectral dimension, wherein the spectral domain shaper is configured to spectrally shape the predictively filtered current spectrum, and the audio encoder is configured to insert information on how to reverse the predictive filtering into the data stream.

3. Audio encoder according to claim 2, wherein the spectrum predictor is configured to perform linear prediction filtering on the current spectrum along the spectral dimension, wherein the data stream former is configured such that the information on how to reverse the predictive filtering comprises information on further linear prediction coefficients underlying the linear prediction filtering on the current spectrum along the spectral dimension.

4. Audio encoder according to claim 2 or 3, wherein the audio encoder is configured to decide to enable or disable the spectrum predictor depending on a tonality or transiency of the audio input signal or a filter prediction gain, wherein the audio encoder is configured to insert information on the decision.

5. Audio encoder according to any of claims 2 to 4, wherein the autocorrelation computer is configured to compute the autocorrelation from the predictively filtered current spectrum.

6. Audio encoder according to any of claims 2 to 5, wherein the spectral decomposer (10) is configured to switch between different transform lengths in spectrally decomposing the audio input signal (12) so that the spectrums are of different spectral resolution, wherein the autocorrelation computer (50) is configured to compute the autocorrelation from the predictively filtered current spectrum in case of a spectral resolution of the current spectrum fulfilling a predetermined criterion, or from the not predictively filtered current spectrum in case of the spectral resolution of the current spectrum not fulfilling the predetermined criterion.

7. Audio encoder according to claim 6, wherein the autocorrelation computer is configured such that the predetermined criterion is fulfilled if the spectral resolution of the current spectrum is higher than a spectral resolution threshold.

8. Audio encoder according to any of claims 1 to 7, wherein the autocorrelation computer is configured to, in computing the autocorrelation from the current spectrum, compute the power spectrum from the current spectrum, perceptually weight the power spectrum and subject the perceptually weighted power spectrum to an inverse transform.

9. Audio encoder according to claim 8, wherein the autocorrelation computer is configured to change a frequency scale of the current spectrum and to perform the perceptual weighting of the power spectrum in the changed frequency scale.

10. Audio encoder according to any of claims 1 to 9, wherein the audio encoder is configured to insert the information on the linear prediction coefficients into the data stream in a quantized form, wherein the spectral domain shaper is configured

to spectrally shape the current spectrum based on the quantized linear prediction coefficients.

11. Audio encoder according to claim 10, wherein the audio encoder is configured to insert the information on the linear prediction coefficients into the data stream in a form according to which quantization of the linear prediction coefficients takes place in the LSF or LSP domain.

12. Audio encoding method comprising

spectrally decomposing an audio input signal (12) into a spectrogram (14) of a sequence of spectrums;

computing an autocorrelation from a current spectrum of the sequence of spectrums;

computing linear prediction coefficients based on the audio correlation;

spectrally shaping the current spectrum based on the linear prediction coefficients;

quantizing the spectrally shaped spectrum; and

inserting information on the quantized spectrally shaped spectrum and information on the linear prediction coefficients into a data stream.

13. Computer program having a program code for performing, when running on a computer, a method according to claim 12.

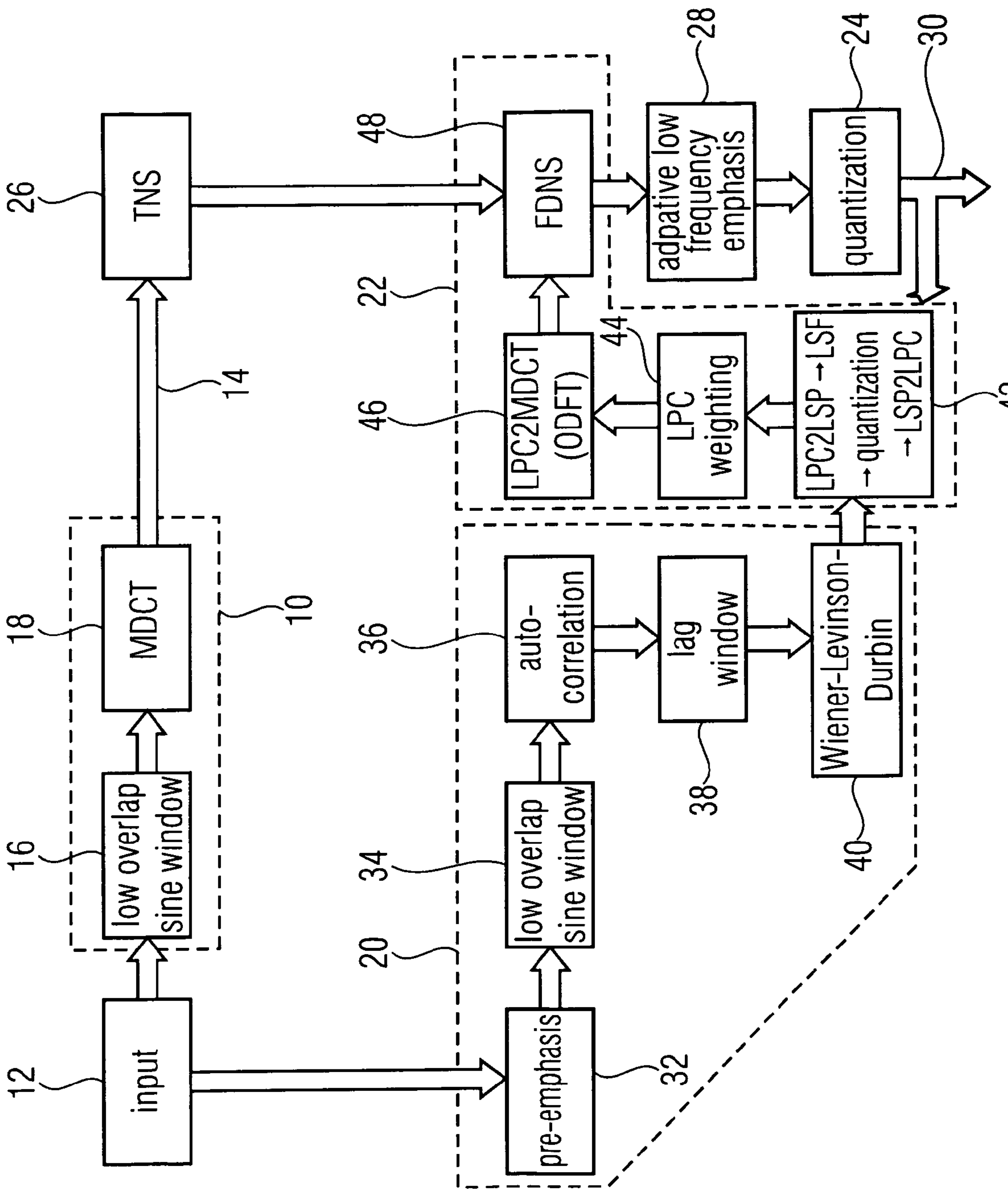
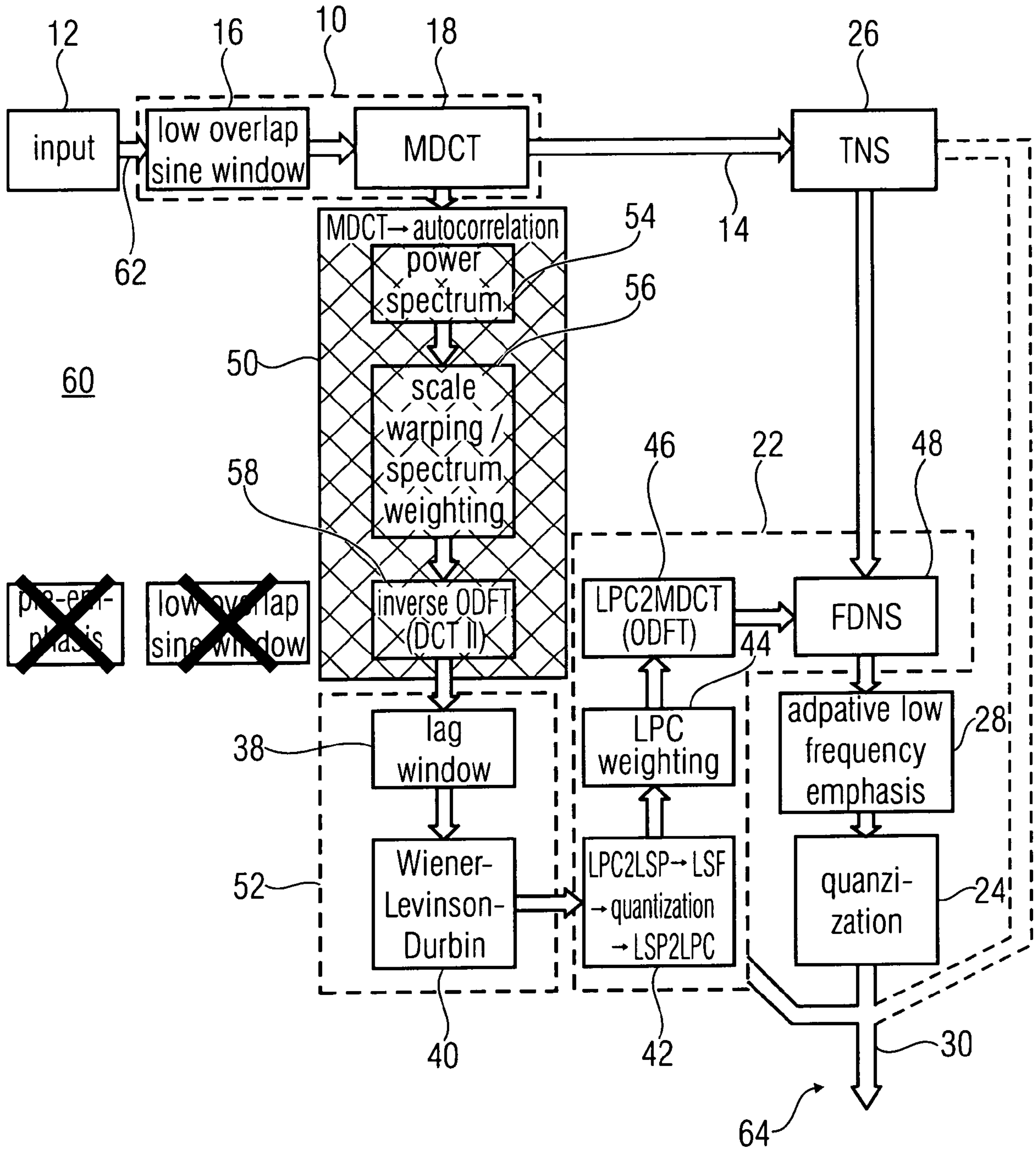

input
12
low overlap sine window
16
MDCT
18
10
14
TNS
26
pre-emphasis
32
low overlap sine window
34
20
auto-correlation
36
lag window
38
Wiener-Levinson-Durbin
40
LPC2LSP→LSF →quantization →LSP2LPC
42
LPC weighting
44
LPC2MDCT (ODFT)
46
FDNS
48
22
adpative low frequency emphasis
28
quantization
24
30

FIG 1



12
16
10
18
26
input
low overlap sine window
MDCT
TNS
62
14
MDCT→autocorrelation
54
power spectrum
56
50
60
scale warping / spectrum weighting
46
22
48
58
inverse ODFT (DCT II)
LPC2MDCT (ODFT)
FDNS
44
38
lag window
LPC weighting
adpative low frequency emphasis
28
52
Wiener-Levinson-Durbin
LPC2LSP→LSF →quantization →LSP2LPC
quanzi-zation
24
40
42
30
64

FIG 2

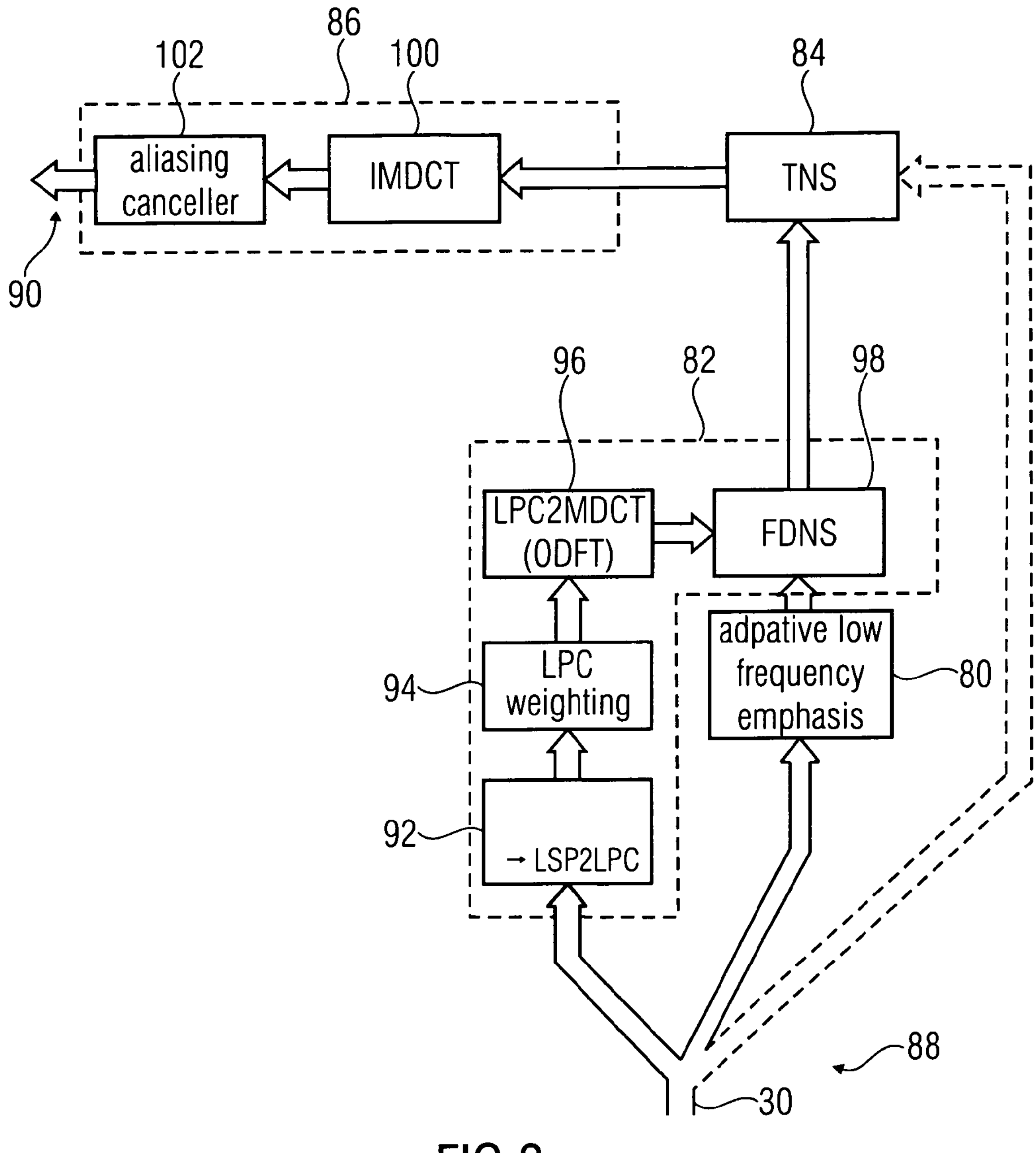

86
102
100
84
aliasing canceller
IMDCT
TNS
90
96
82
98
LPC2MDCT (ODFT)
FDNS
adpative low frequency emphasis
80
94
LPC weighting
92
→ LSP2LPC
88
30

FIG 3

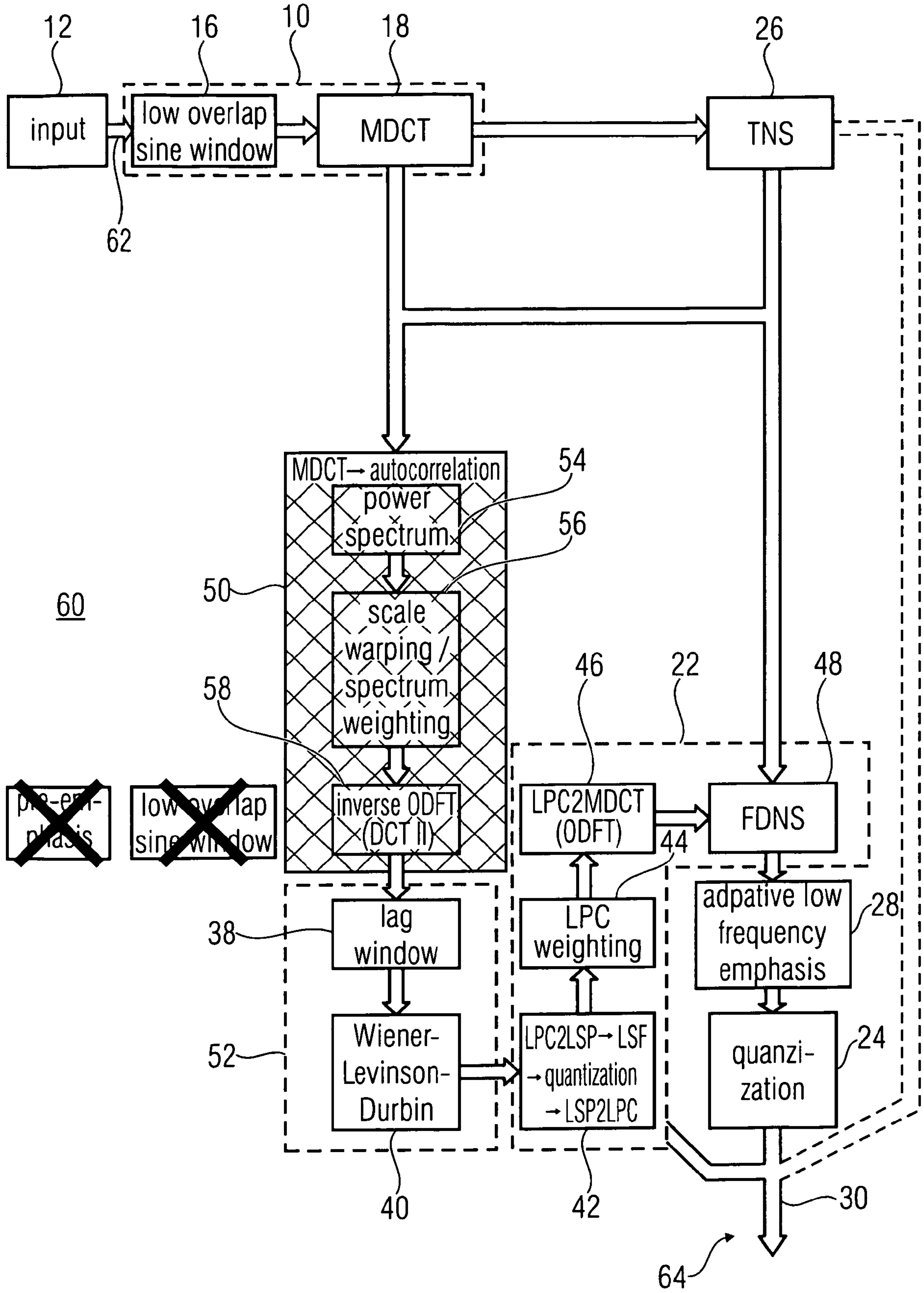

12
16
10
18
26
input
low overlap sine window
MDCT
TNS
62
MDCT→ autocorrelation
54
power spectrum
56
50
60
scale warping / spectrum weighting
46
22
48
58
inverse ODFT (DCT II)
LPC2MDCT (ODFT)
FDNS
44
38
lag window
LPC weighting
adpative low frequency emphasis
28
52
Wiener-Levinson-Durbin
LPC2LSP→ LSF →quantization →LSP2LPC
quanzi-zation
24
30
40
42
64

FIG 4

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2012/052455

A. CLASSIFICATION OF SUBJECT MATTER
INV. G10L19/06 G10L19/14
ADD.

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

B. FIELDS SEARCHED

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

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

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

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MOTLICEK ET AL: "Audio Coding Based on Long Temporal Contexts", INTERNET CITATION, April 2006 (2006-04), XP002423396, Retrieved from the Internet: URL:http://www.idiap.ch/publications/motlicek-idiap-rr-06-30.bib.abs.html [retrieved on 2007-03-06] pages 2-3; figure 1	1-13
X	EP 1 852 851 A1 (BEIJING MEDIA WORKS CO LTD [CN]; BEIJING E WORLD TECHNOLOGY CO [CN]; C) 7 November 2007 (2007-11-07) paragraph [0007] figure 5 paragraphs [0029] - [0036] page 13, lines 6-57	1-13

-/--

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
25 April 2012	10/05/2012
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Ramos Sánchez, U

3

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2012/052455

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	MAX NEUENDORF ET AL: "Completion of Core Experiment on unification of USAC Windowing and Frame Transitions", 91. MPEG MEETING; 18-1-2010 - 22-1-2010; KYOTO; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. M17167, 16 January 2010 (2010-01-16), XP030045757, pages 2-3 -----	1-13

3

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2012/052455

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 1852851 A1	07-11-2007	EP 1852851 A1	07-11-2007
		WO 2005096274 A1	13-10-2005