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

(11) EP 0 899 720 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
15.12.2004 Bulletin 2004/51

(51) Int Cl.[7]: **G10L 19/06**

(21) Application number: **98306906.3**

(22) Date of filing: **27.08.1998**

(54) **Quantization of linear prediction coefficients**

Quantisierung der linearen Prädiktionskoeffizienten

Quantisation des coefficients de prédiction linéaire

(84) Designated Contracting States:
DE FR GB IT NL

(30) Priority: **28.08.1997 US 57114 P**

(43) Date of publication of application:
03.03.1999 Bulletin 1999/09

(73) Proprietor: **TEXAS INSTRUMENTS INC.**
Dallas, Texas 75243 (US)

(72) Inventor: **McCree, Alan V.**
Dallas Texas 75248 (US)

(74) Representative: **Potter, Julian Mark et al**
D Young & Co
120 Holborn
London EC1N 2DY (GB)

(56) References cited:
EP-A- 0 905 680

- **MCCREE A ET AL: "A 1.7 kb/s MELP coder with improved analysis and quantization" PROCEEDINGS OF THE 1998 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, ICASSP '98 (CAT. NO.98CH36181), PROCEEDINGS OF THE 1998 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, SEATTLE, WA, USA, 12-1, pages 593-596 vol.2, XP002109401 1998, New York, NY, USA, IEEE, USAISBN: 0-7803-4428-6**
- **GARDNER W R ET AL: "THEORETICAL ANALYSIS OF THE HIGH-RATE VECTOR QUANTIZATION OF PLC PARAMETERS" IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, vol. 3, no. 5, 1 September 1995 (1995-09-01), pages 367-381, XP000641961 ISSN: 1063-6676**
- **MCCREE A ET AL: "A 2.4 kbit/s MELP coder candidate for the new U.S. Federal Standard" 1996 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING CONFERENCE PROCEEDINGS (CAT. NO.96CH35903), 1996 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING CONFERENCE PROCEEDINGS, ATLANTA, GA, USA, 7-10 M, pages 200-203 vol. 1, XP002109402 1996, New York, NY, USA, IEEE, USAISBN: 0-7803-3192-3**
- **COHN R P ET AL: "Incorporating perception into LSF quantization some experiments" 1997 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING (CAT. NO.97CB36052), 1997 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, MUNICH, GERMANY, 21-24 APRIL 1997, pages 1347-1350 vol.2, XP002109403 1997, Los Alamitos, CA, USA, IEEE Comput. Soc. Press, USAISBN: 0-8186-7919-0**
- **PALIWAL K K ET AL: "EFFICIENT VECTOR QUANTIZATION OF LPC PARAMETERS AT 24 BITS/FRAME" SPEECH PROCESSING 1, TORONTO, MAY 14 - 17, 1991, vol. 1, no. CONF. 16, 14 May 1991 (1991-05-14), pages 661-664, XP000245315 INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERSISBN: 0-7803-0003-3**

EP 0 899 720 B1

Description

TECHNICAL FIELD OF THE INVENTION

[0001] This invention relates to switched-predictive vector quantization and more particularly to quantization of LPC coefficients transformed to line spectral frequencies.

BACKGROUND OF THE INVENTION

[0002] Many speech coders, such as the new 2.4 kb/s Federal Standard Mixed Excitation Linear Prediction (MELP) coder (McCree, et al., entitled, "A 2.4 kbits/s MELP Coder Candidate for the New U. S. Federal Standard," Proc. ICASSP-96, pp. 200-203, May 1996.) use some form of Linear Predictive Coding (LPC) to represent the spectrum of the speech signal. A MELP coder is described in the patent document US-A-6 463 406 entitled "Mixed Excitation Linear Prediction with Fractional Pitch," filed 05/20/96. Fig. 1 illustrates such a MELP coder. The MELP coder is based on the traditional LPC vocoder with either a periodic impulse train or white noise exciting a 10th order on all-pole LPC filter. In the enhanced version, the synthesizer has the added capabilities of mixed pulse and noise excitation periodic or aperiodic pulses, adaptive spectral enhancement and pulse dispersion filter as shown in Fig. 1. Efficient quantization of the LPC coefficients is an important problem in these coders, since maintaining accuracy of the LPC has a significant effect on processed speech quality, but the bit rate of the LPC quantizer must be low in order to keep the overall bit rate of the speech coder small. The MELP coder for the new Federal Standard uses a 25-bit multi-stage vector quantizer (MSVQ) for line spectral frequencies (LSF) . There is a 1 to 1 transformation between the LPC coefficients and LSF coefficients.

[0003] Quantization is the process of converting input values into discrete values in accordance with some fidelity criterion. A typical example of quantization is the conversion of a continuous amplitude signal into discrete amplitude values. The signal is first sampled, then quantized.

[0004] For quantization, a range of expected values of the input signal is divided into a series of subranges. Each subrange has an associated quantization level. A sample value of the input signal that is within a certain subrange is converted to the associated quantizing level. For example, for 8-bit quantization, a sample of the input signal would be converted to one of 256 levels, each level represented by an 8-bit value.

[0005] Vector quantization is a method of quantization, which is based on the linear and non-linear correlation between samples and the shape of the probability distribution. Essentially, vector quantization is a lookup process, where the lookup table is referred to as a "codebook". The codebook lists each quantization level, and each level has an associated "code-vector". The vector quantization process compares an input vector to the code-vectors and determines the best code-vector in terms of minimum distortion. Where x is the input vector, the comparison of distortion values may be expressed as:

$$d(x, y^{(J)}) < d(x, y^{(k)}),$$

for all j not equal to k. The codebook is represented by $y^{(j)}$, where $y^{(j)}$ is the jth code-vector, $0 < j < L$, and L is the number of levels in the codebook.

[0006] Multi-stage vector quantization (MSVQ) is a type of vector quantization. This process obtains a central quantized vector (the output vector) by adding a number of quantized vectors. The output vector is sometimes referred to as a "reconstructed" vector. Each vector used in the reconstruction is from a different codebook, each codebook corresponding to a "stage" of the quantization process. Each codebook is designed especially for a stage of the search. An input vector is quantized with the first codebook, and the resulting error vector is quantized with the second codebook, etc. The set of vectors used in the reconstruction may be expressed as:

$$y^{(J0,J1-JS-1)} = y_0^{(J0)} + y_1^{(J1)} + y_{S-1}^{(JS-1)}$$

, where S is the number of stages and y_s is the codebook for the sth stage. For example, for a three-dimensional input vector, such as x = (2,3,4), the reconstruction vectors for a two-stage search might be y_0 = (1,2,3) and y_1 = (1,1,1) (a perfect quantization and not always the case).

[0007] During multi-stage vector quantization, the codebooks may be searched using a sub-optimal tree search algorithm, also known as an M-algorithm. At each stage, M-best number of "best" code-vectors are passed from one stage to the next. The "best" code-vectors are selected in terms of minimum distortion. The search continues until the final stage, when only one best code-vector is determined.

[0008] In predictive quantization a target vector for quantization in the current frame is the mean-removed input vector minus a predictive value. The predicted value is the previous quantized vector multiplied by a known prediction matrix. In switched prediction, there is more than one possible prediction matrix and the best prediction matrix is selected for each frame. See S. Wang, et al., "Product Code Vector Quantization of LPC Parameters," in Speech and Audio Coding for Wireless and Network Applications," Ch. 31, pp. 251-258, Kluwer Academic Publishers, 1993.

[0009] It is highly desirable to provide an improved weighted distance measure that better correlates with subjective speech quality.

SUMMARY OF THE INVENTION

[0010] In accordance with a preferred embodiment the present invention as claimed in the appended claims provides an improved method of vector quantization of LSF transformation of LPC coefficients by a new weighted distance measure that better correlates with subjective speech quality. This weighting includes running samples from the LPC filter from an impulse and applying these samples to a perceptual weighting filter.

DESCRIPTION OF THE DRAWINGS

[0011] Embodiments of the present invention will now be further described, by way of example, with reference to the accompanying drawings in which:

Fig. 1 is a block diagram of Mixed Excitation Linear Prediction Coder;
Fig. 2 is a block diagram of switch-predictive vector quantization encoder according to the present invention;
Fig. 3 is a block diagram of a decoder according to the present invention;
Fig. 4 is a flow chart for determining a weighted distance measure in accordance with an embodiment of the present invention; and
Fig. 5 is a block diagram of an encoder according to an embodiment of the present invention.

DESCRIPTION OF PREFERRED EMBODIMENTS OF THE PRESENT INVENTION

[0012] The new quantization method, like the one used in the 2.4 kb/s Federal Standard MELP coder, uses multi-stage vector quantization (MSVQ) of the Line Spectral Frequency (LSF) transformation of the LPC coefficients (LeBlanc, et al., entitled "Efficient Search and Design Procedures for Robust Multi-Stage VQ or LPC Parameters for 4kb/s Speech Coding," IEEE Transactions on Speech and Audio Processing, Vol. 1, No. 4, October 1993, pp. 373-385.) An efficient codebook search for multi-stage VQ is disclosed in US Patent Application Serial No. 09/003,172 cited above. However, the method, described herein, improves on the previous one in two ways: the use of switched prediction to take advantage of time redundancy and the use of a new weighted distance measure that better correlates with subjective speech quality.

[0013] In the Federal Standard MELP coder, the input LSF vector is quantized directly using MSVQ. However, there is a significant redundancy between LSF vectors of neighboring frames, and quantization accuracy can be improved by exploiting this redundancy. As discussed previously in predictive quantization, the target vector for quantization in the current frame is the mean-removed input vector minus a predicted value, where the predicted value is the previous quantized vector multiplied by a known prediction matrix. In switched prediction, there is more than one possible prediction matrix, and the best predictor or prediction matrix is selected for each frame. In accordance with an embodiment of the present invention, both the predictor matrix and the MSVQ codebooks are switched. For each input frame, we search every possible predictor/codebooks set combination for the predictor/codebooks set which minimizes the squared error. An index corresponding to this pair and the MSVQ codebook indices are then encoded for transmission. This differs from previous techniques in that the codebooks are switched as well as the predictors. Traditional methods share a single codebook set in order to reduce codebook storage, but we have found that the MSVQ codebooks used in switched predictive quantization can be considerably smaller than non-predictive codebooks, and that multiple smaller codebooks do not require any more storage space than one larger codebook. From our experiments, the use of separate predictor/codebooks pairs results in a significant performance improvement over a single shared codebook, with no increase in bit rate.

[0014] Referring to the LSF encoder with switched predictive quantizer 20 of Fig. 2, the 10 LPC coefficients are transformed by transformer 23 to 10 LSF coefficients of the Line Spectral Frequency (LSF) vectors. The LSF has 10 dimensional elements or coefficients (for 10 order all-pole filter). The LSF input vector is subtracted in adder 22 by a selected mean vector and the mean-removed input vector is subtracted in adder 25 by a predicted value. The resulting target vector for quantization vector e in the current frame is applied to multi-stage vector quantizer (MSVQ) 27. The predicted value is the previous quantized vector multiplied by a known prediction matrix at multiplier 26. The predicted

value in switched prediction has more than one possible prediction matrix. The best predictor (prediction matrix and mean vector) is selected for each frame. In accordance with an embodiment of the present invention, both the predictor (the prediction matrix and mean vector) and the MSVQ codebook set are switched. A control 29 first switches in via switch 28 prediction matrix 1 and mean vector 1 and first set of codebooks 1 in quantizer 27. The index corresponding to this first prediction matrix and the MSVQ codebook indices for the first set of codebooks are then provided out of the quantizer to gate 37. The predicted value is added to the quantized output $\hat{e}$ for the target vector e at adder 31 to produce a quantized mean-removed vector. The mean-removed vector is added at Adder 70 to the selected mean vector to get quantized vector $\hat{X}$. The squared error for each dimension is determined at squarer 35. The weighted squared error between the input vector X_i and the delayed quantized vector $\hat{X}_i$ is stored at control 29. The control 29 applies control signals to switch in via switch 28 prediction matrix 2 and mean vector 2 and codebook 2 set to likewise measure the weighted squared error for this set at squarer 35. The measured error from the first pair of prediction matrix 1 (with mean vector 1) and codebooks set 1 is compared with prediction matrix 2 (with mean vector 2) and codebook set 2. The set of indices for the codebooks with the minimum error is gated at gate 37 out of the encoder as encoded transmission of indices and a bit is sent out at terminal 38 from control 29 indicating from which pair of prediction matrix and codebooks set the indices was sent (codebook set 1 with mean vector 1 and predictor matrix 1 or codebook set 2 and prediction matrix 2 with mean vector 2). The mean-removed quantized vector from adder 31 associated with the minimum error is gated at gate 33a to frame delay 33 so as to provide the previous mean-removed quantized vector to multiplier 26.

[0015] Fig. 3 illustrates a decoder 40 for use with LSF encoder 20. At the decoder 40, the indices for the codebooks from the encoding are received at the quantizer 44 with two sets of codebooks corresponding to codebook set 1 and 2 in the encoder. The bit from terminal 38 selects the appropriate codebook set used in the encoder. The LSF quantized input is added to the predicted value at adder 41 where the predicted value is the previous mean-removed quantized value (from delay 43) multiplied at multiplier 45 by the prediction matrix at 42 that matches the best one selected at the encoder to get mean-removed quantized vector. Both prediction matrix 1 and mean value 1 and prediction matrix 2 and mean value 2 are stored at storage 42 of the decoder. The 1 bit from terminal 38 of the encoder selects the prediction matrix and the mean value at storage 42 that matches the encoder prediction matrix and mean value. The quantized mean-removed vector is added to the selected mean value at adder 48 to get the quantized LSF vector. The quantized LSF vector is transformed to LPC coefficients by transformer 46.

[0016] As discussed previously, LSF vector coefficients correspond to the LPC coefficients. The LSF vector coefficients have better quantization properties than LPC coefficients. There is a 1 to 1 transformation between these two vector coefficients. A weighting function is applied for a particular set of LSFs for a particular set of LPC coefficients that correspond.

[0017] The Federal Standard MELP coder uses a weighted Euclidean distance for LSF quantization due to its computational simplicity. However, this distance in the LSF domain does not necessarily correspond well with the ideal measure of quantization accuracy: perceived quality of the processed speech signal. The applicant has previously shown in the paper on the new 2.4 kb/s Federal Standard that a perceptually-weighted form of log spectral distortion has close correlation with subjective speech quality. The applicant teaches herein in accordance with an embodiment a weighted LSF distance which corresponds closely to this spectral distortion. This weighting function requires looking into the details of this transformation for a particular set of LSFs for a particular input vector x which is a set of LSFs for a particular set of LPC coefficients that correspond to that set. The coder computes the LPC coefficients and as discussed above, for purposes of quantization, this is converted to LSF vectors which are better behaved. As shown in Fig. 1, the actual synthesizer will take the quantized vector $\hat{X}$ and perform an inverse transformation to get an LPC filter for use in the actual speech synthesis. The optimal LSF weights for un-weighted spectral distortion are computed using the formula presented in paper of Gardner, et al., entitled, "Theoretical Analysis of the High-Rate Vector Quantization of the LPC Parameters," IEEE Transactions on Speech and Audio Processing, Vol. 3, No. 5, September 1995, pp. 367-381.

$$W_i = R_A(0)R_i(0) + 2\sum_{m=1}^{p-1} R_A(m)R_i(m)$$

where $R_A(m)$ is the autocorrelation of the impulse response of the LPC synthesis filter at lag m, and $R_i(m)$ is the correlation of the elements in the ith column of the Jacobian matrix of the transformation from LSF's to LPC coefficients. Therefore for a particular input vector x we compute the weight W_i.

[0018] The difference in the present solution is that perceptual weighting is applied to the synthesis filter impulse response prior to computation of the autocorrelation function $R_A(m)$, so as to reflect a perceptually-weighted form of spectral distortion.

[0019] In accordance with the weighting function as applies to the embodiment of Fig. 2, the weighting W_i is applied to the squared error at 35. The weighted output from error detector 35 is:

$$\Sigma W_i(X_i-\hat{X}_i)^2 .$$

[0020] Each entry in a 10 dimensional vector has a weight value. The error sums the weight value for each element. In applying the weight, for example, one of the elements has a weight value of three and the others are one then the element with three is given an emphasis by a factor of three times that of the other elements in determining error.

[0021] As stated previously, the weighting function requires looking into the details of the LPC to LSF conversion. The weight values are determined by applying an impulse to the LPC synthesis filter 21 and providing the resultant sampled output of the LPC synthesis filter 21 to a perceptual weighting filter 47. A computer 39 is programmed with a code based on a pseudo code that follows and is illustrated in the flow chart of Fig. 4. An impulse is gated to the LPC filter 21 and N samples of LPC synthesis filter response (step 51) are taken and applied to a perceptual weighting filter 47 (step 52). In accordance with one embodiment of the invention low frequencies are weighted more than high frequencies and use the well known Bark scale which matches how the human ear responds to sounds. The equation for Bark weighting $W_B(f)$ is

$$W_B(f) = \frac{1}{25+75(1+1.4(\frac{t}{1000})^2)^{0\,69}}$$

The coefficients of a filter with this response are determined in advance and stored and time domain coefficients are stored. An 8 order all-pole fit to this spectrum is determined and these 8 coefficients are used as the perceptual weighting filter. The following steps follow the equation for un-weighted spectral distortion from Gardner, et al. paper found on page 375 are expressed as

$$W_t = R_A(0)R_t(0) + 2\sum_{m=1}^{p-1} R_A(m)R_t(m)$$

where $R_A(m)$ is the autocorrelation of the impulse response of the LPC synthesis filter at lag m, where

$$R_A(k) = \sum_{n=0}^{\alpha} h(n)h(n+k)$$

h(n) is an impulse response, R_1 (m) is

$$R_{j_t}(m) = \sum_{n=1}^{v-m} (J_\omega(\omega))_{n,j} (J_\omega(\omega))_{m+n,j}$$
$$= \sum_{n=1}^{v-m} j_t(n)\, j_t(m+n)$$

is the correlation function of the elements in the ith column of the Jacobian matrix J_ω (ω) of the transformation from LSFs to LPC coefficients. Each column of J_ω (ω) can be found by

$$= \begin{cases} \sin(\omega_i) e^{-j\omega} \prod_{j=0,\, j \neq (i+1)/2}^{v/2} \tilde{p}_j(\omega); i \text{ odd} \\ \sin(\omega_i) e^{-j\omega} \prod_{j=0,\, j \neq i/2}^{v/2} \tilde{q}_j(\omega); i \text{ even} \end{cases}$$

since

$$\prod_{j=0,\, j \neq i}^{v/2} \tilde{p}_j(\omega) = P(\omega) / \tilde{p}_i(\omega)$$

The values of $j_i(n)$ can be found by simple polynomial division of the coefficients of $P(\omega)$ by the coefficients of $\tilde{p}_i(\omega)$. Since the first coefficient of $\tilde{p}_i(\omega) = 1$, no actual divisions are necessary in this procedure. Also, $j_i(n) = j_i(v + 1 - n)$: i odd; $0 < n < v$, so only half the values must be computed. Similar conditions with an anti-symmetry property exist for the even columns.

[0022] The autocorrelation function of the weighted impulse response is calculated (step 53 in Fig. 4). From that the Jacobian matrix for LSFs is computed (step 54). The correlation of rows of Jacobian matrix is then computed (step 55). The LSF weights are then calculated by multiplying correlation matrices (step 56). The computed weight value from computer 39, in Fig. 2, is applied to the error detector 35. The indices from the prediction matrix/codebook set with the least error is then gated from the quantizer 27. The system may be implemented using a microprocessor encapsulating computer 39 and control 29 utilizing the following pseudo code. The pseudo code for computing the weighting vector from the current LPC and LSF follows:

```
/* Compute weighting vector from current LPC and LSF's */
Compute N samples of LPC synthesis filter impulse response
Filter impulse response with perceptual weighting filter
Calculate the autocorrelation function of the weighted
impulse response
Compute Jacobian matrix for LSF's
Compute correlation of rows of Jacobian matrix
Calculate LSF weights by multiplying correlation matrices
```

[0023] The code for the above is provided in Appendix A.
The pseudo code for the encode input vector follows:

```
/* Encode input vector */
For all predictor, codebook pairs
  Remove mean from input LSF vector
  Subtract predicted value to get target vector
  Search MSVQ codebooks for best match to target vector using
weighted distance
  If Error <Emin
     Emin = Error
     best predictor index = current predictor
  Endif
End
Endcode best predictor index and codebook indices for
transmission
```

[0024] The pseudo code for regenerate quantized vector follows:

```
/* Regenerate quantized vector */
Sum MSVQ codevectors to produce quantized target
Add predicted value
Update memory of past quantized values (mean-removed)
Add mean to produce quantized LSF vector
```

[0025] We have implemented a 20-bit LSF quantizer based on this new approach which produces equivalent performance to the 25-bit quantizer used in the Federal Standard MELP coder, at a lower bit rate. There are two predictor/codebook pairs, with each consisting of a diagonal first-order prediction matrix and a four stage MSVQ with codebook of size 64, 32, 16, and 16 vectors each. Both the codebook storage and computational complexity of this new quantizer are less than in the previous version.

[0026] Although the present invention and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations can be made herein without departing from the scope of the invention as defined by the appended claims.

[0027] For example it is anticipated that the system and method be used without switched prediction for each frame as illustrated in Fig. 5 wherein the weighted error for each frame would be determined at error detector and codebook indices with the least error would be gated out by control 29 and gate 37. For each frame, the LPC filtered samples of the impulse at filter 21 should be filtered by perception weighting filter 47 and processed by computer 39 using code such as described in the pseudo code to provide the weight vales. Also the perception weighting filter may use other perceptual weighting besides the bark scale that is perceptually motivated such as weighting low frequencies more than high frequencies, or the perceptual weighting filter as is presently used in CELP coders.

APPENDIX A

[0028]

```
/*  Function vq_lspw: computs LSF weights

  Inputs:
    *p_lsp - LSF array
    *pc - LPC coefficients
    p - LPC modsl ordar

  Output:
    *w - array of weights

  Copyright 1997, Texas Instruments

*/

Float *vq_lspw(Float *w,Float *p_lsp,Float *pc,Int p)
{
    Int i,j,k,m;
    Float d, tmp, *tp, *ir, *R, *pz, *qz, *rem, *t, **J, **RJ;
    static Float bark_wt[8] = {
        -0.84602182,
        0.27673657,
        -0.10480262,
        0.05609138,
        -0.03315923,
        0.02132074,
        -0.01359822,
        0.00398910,
    };

    /* Allocate local array memory */
    MEM_ALLOC(MALLOC,ir,IRLENGTH+p,Float);
    ir+=&ir[p];
    MEM_ALLOC(MALLOC,R,p,Float);
    MEM_ALLOC(MALLOC,pz,p+2,Float);
    MEM_ALLOC(MALLOC,qz,p+2,Float);
    MEM_ALLOC(MALLOC,rem,p+2,Float);
    MEM_ALLOC(MALLOC,t,3,Float);
    MEM_2ALLOC(MALLOC,J,p+1,p+1,Float);
    MEM_2ALLOC(MALLOC,RJ,p+1,p,Float);

    /* calculate IRLENGTH samples of the synthesis filter impulse response*/
    for (i=-p; i<IRLENGTH; i++)
        ir[i] = 0.0;
    ir[0] = 1.0;
    for (i=0; i<IRLENGTH; i++)
    {
        for (j=1; j<=p; j++)
            ir[i] -= pc[j] * ir[i-j];
    }

    /* use all-pole model for frequency weighting */
    for (i=0; i<IRLENGTH; i++)
    {
        for (j=1; j<=8; j++)
            ir[i] -= bark_wt[j-1] * ir[i-j];
    }

    /* calculate the autocorrelation function of the impulse response */
    for (m=0; m<p; m++)                          /* for lags of 0 to p-1 */
    {
        R[m] = 0.0f;
        for (i=0; i<IRLENGTH-m; i++)
            R[m] += ir[i] * ir[i+m];
    }

    /* calculate P(z) and Q(z) */
    for (i=1; i<=p; i++)
    {
        pz[i] = pc[i] + pc[p+1-i];
        qz[i] = pc[i] - pc[p+1-i];
    }
```

APPENDIX A

```
pz[0] = qz[0] = pz[p+1] = 1.0f;
qz[p+1] = -1.0f;

/* calculate the J matrix */
/* use the rows of J to store the polynomials */
/* (rather than the columns, as in Gardner) */
t[0] = t[2] = 1.0f;
for (i=1; i<=p; i++)                                    /* for all the rows of J */
{
    t[1] = -2.0f * cos(PI * p_lsp[i]);
    tmp = sin(PI * p_lsp[i]);
    if (i != 2 * (i/2)) tp = pz;                        /* i is odd;  use p(z) */
    else                tp = qz;                        /* i is even; use q(z) */
    /* divide polynomial tp by polynomial t and put the result into */
    /* row J[i] */
    for (j=0; j<=p+1; j++)
        rem[j] = tp[j];
    for (k=p; k>=1; k--)
    {
        J[i][k] = rem[k+1];
        for (j=k; j>=k-1; j--)
            rem[j] -= J[i][k] * t[j-k+1];
    }
    /* multiply the ith row by the sin() term */
    for (j=1; j<=p; j++)
        J[i][j] *= tmp;
}

/* determine the 'correlation' function of the rows of J */
for (i=1; i<=p; i++)                                    /* for each row */
{
    for (m=0; m<p; m++)                                 /* for each lag */
    {
        RJ[i][m] = 0.0f;
        /* for each element in the row */
        for (j=1; j<=p-m; j++)
            RJ[i][m] += J[i][j] * J[i][j+m];
    }
}
/* finish the weight calculation */
for (i=1; i<=p; i++)
{
    tmp = 0.0f;
    for (m=1; m<p; m++)
        tmp += R[m] * RJ[i][m];
    w[i-1] = R[0] * RJ[i][0] + 2.0f * tmp;

}

/* Free local memory */
ir=&ir[-p];
MEM_FREE(FREE,ir);
MEM_FREE(FREE,R);
MEM_FREE(FREE,pz);
MEM_FREE(FREE,qz);
MEM_FREE(FREE,rem);
MEM_FREE(FREE,t);
MEM_2FREE(FREE,J);
MEM_2FREE(FREE,RJ);

return(w);
```

Claims

1. A method of vector quantization of LPC coefficients comprising the steps of:

translating LPC coefficients to LSF coefficients;
providing a plurality of predictor matrices and a quantizer with a plurality of codebooks for quantizing LSF target vectors using switched predictive quantization;
searching a predictor/codebooks set combination for determining LSF target vectors that result in quantized output that best match LPC coefficients by switching both the predictor matrices and the codebooks, which minimizes a squared error;
applying said target vectors to said codebooks to get quantized vectors;
said searching step comprising a step of determining the squared error multiplied by a weighting value for each dimension between the LSF coefficients and the quantized output wherein said weighting value is a function of perceptual weighting;
and said determining calculating step including the steps of:

calculating an autocorrelation function of a weighted impulse response; and
calculating LSF weights (56) from perceptual-weighted input response.

2. The method of Claim 1, comprising computing a Jacobian matrix (54) for said LSF vectors;
computing the correlation (55) of rows of the Jacobian matrix; and
wherein the LSF weights are calculated by multiplying correlation matrices.

3. The method of Claim 1or Claim 2, wherein said determining step comprises the further steps for finding said weighting value of:

applying an impulse to said LPC filter and running N samples (51) of the LPC synthesis response; and
filtering (52) the samples with a perceptual filter;
calculating the autocorrelation function (53) of the weighted impulse response;
computing the Jacobian matrix (54) for said LSF vectors;
computing the correlation (55) of rows of Jacobian matrix; and
calculating LSF weights (56) by multiplying correlation matrices.

4. The method of Claim 3 wherein the step (52) of filtering the samples with said perceptual filter comprises weighting low frequencies more than high frequencies.

5. The method of Claim 4 wherein the step (52) of filtering the samples with said perceptual filter comprises following the bark scale.

6. The method of any preceding Claim wherein said step of providing said quantizer comprises providing a multi-stage vector quantizer.

7. The method of any preceding Claim wherein said step of providing said quantizer comprises providing a quantizer having one or more sets of codebooks.

8. A quantizer (20) for a coder including an LPC filter (21) and a translator (23) for translating LPC coefficients to LSF coefficients comprising:

means for providing a plurality of predictor matrices and a plurality of codebooks (27) for quantizing LSF target vectors using switched predictive quantisation;
means (39) for searching a predictor/codebooks set combination for determining LSF target vectors that result in quantized output that best match LPC coefficients by switching both the predictor matrices and the codebooks, which minimizes a squared error;
means (29) for applying said LSF target vectors to said codebooks to provide a quantized output;
said searching means including means for applying an impulse to said LPC filter (21);
means for running samples of said LPC response;
a perceptual filter (47) for filtering said samples; and
means (35) for calculating an autocorrelation function by weighted response to provide LSF weights computed

from perceptual-weighted input response to the LPC filter (21).

9. The quantizer (20) of Claim 8, wherein the means (35) for calculating an autocorrelation function is operable to calculate a Jacobian matrix for the LSF vectors, a correlation of rows of the Jacobian matrix, and to calculate the LSF weights by multiplying correlation matrices.

10. The quantizer (20) of Claim 8 or Claim 9, further operable to:

run N samples of LPC synthesis response upon application of the weighted impulse response; and
filter the samples using a perceptual filter (47).

11. The quantizer (20) of Claim 10, wherein the perceptual filter (47) is used to weight low frequencies more than high frequencies.

12. The quantizer (20) of Claim 11, wherein the perceptual filter (47) follows the bark scale.

13. The quantizer (20) of any one of Claims 8 to 12, further operable to provide multi-stage vector quantization.

14. The quantizer (20) of any one of Claims 8 to 13, comprising one or more sets of codebooks (27).

15. A coder incorporating the quantizer (20) according to any one of Claims 8 to 14.

Patentansprüche

1. Verfahren zur Vektorquantisierung von LPC-Koeffizienten, das die folgenden Schritte umfasst:

Übersetzen von LPC-Koeffizienten in LSF-Koeffizienten;

Vorsehen mehrerer Prädiktormatrizen und eines Quantisierers mit mehreren Codebüchern zum Quantisieren von LSF-Zielvektoren unter Verwendung einer geschalteten prädiktiven Quantisierung;

Durchsuchen einer Prädiktor-/Codebuchmengen-Kombination, um LSF-Zielvektoren zu bestimmen, die einen quantisierten Ausgang zur Folge haben, mit dem die LPC-Koeffizienten am besten übereinstimmen, indem sowohl die Prädiktormatrizen als auch die Codebücher umgeschaltet werden, um einen quadratischen Fehler minimal zu machen;

Anwenden der Zielvektoren auf die Codebücher, um quantisierte Vektoren zu erhalten;
wobei der Durchsuchungsschritt einen Schritt des Bestimmens des mit einem Gewichtungswert für jede Dimension multiplizierten quadratischen Fehlers zwischen den LSF-Koeffizienten und dem quantisierten Ausgang umfasst, wobei der Gewichtungswert eine Funktion der perzeptuellen Gewichtung ist;
und der Bestimmungs- und Berechnungsschritt die folgenden Schritte umfasst:

Berechnen einer Autokorrelationsfunktion einer gewichteten Impulsantwort; und

Berechnen von LSF-Gewichten (56) aus der perzeptuell gewichteten Eingangsantwort.

2. Verfahren nach Anspruch 1, das umfasst: Berechnen einer Jacobi-Matrix (54) für die LSF-Vektoren; und Berechnen der Korrelation (55) von Zeilen der Jacobi-Matrix;
wobei die LSF-Gewichte durch Multiplizieren von Korrelationsmatrizen berechnet werden.

3. Verfahren nach Anspruch 1 oder Anspruch 2, bei dem der Bestimmungsschritt zum Auffinden des Gewichtungswertes die folgenden weiteren Schritte umfasst:

Eingeben eines Impulses in das LPC-Filter und Durchlaufenlassen von N Abtastwerten (51) der LPC-Syntheseantwort; und

Filtern (52) der Abtastwerte mit einem perzeptuellen Filter;

Berechnen der Autokorrelationsfunktion (53) der gewichteten Impulsantwort;

Berechnen der Jacobi-Matrix (54) für die LSF-Vektoren;

Berechnen der Korrelation (55) von Zeilen der Jacobi-Matrix; und

Berechnen von LSF-Gewichten (56) durch Multiplizieren von Korrelationsmatrizen.

4. Verfahren nach Anspruch 3, bei dem der Schritt (52) des Filterns der Abtastwerte mit dem perzeptuellen Filter das Gewichten niedriger Frequenzen in stärkerem Maß als hohe Frequenzen umfasst.

5. Verfahren nach Anspruch 4, bei dem der Schritt (52) des Filtems der Abtastwerte mit dem perzeptuellen Filter das Verfolgen der Barkskala umfasst.

6. Verfahren nach einem vorhergehenden Anspruch, bei dem der Schritt des Vorsehens des Quantisierers das Vorsehen eines mehrstufigen Vektorquantisierers umfasst.

7. Verfahren nach einem vorhergehenden Anspruch, bei dem der Schritt des Vorsehens des Quantisierers das Vorsehen eines Quantisierers, der eine oder mehrere Mengen von Codebüchern besitzt, umfasst.

8. Quantisierer (20) für einen Codierer, der ein LPC-Filter (21) und einen Übersetzer (23) zum Übersetzen von LPC-Koeffizienten in LSF-Koeffizienten enthält, mit:

Mitteln zum Vorsehen mehrerer Prädiktormatrizen und mehrerer Codebücher (27) zum Quantisieren von LSF-Zielvektoren unter Verwendung einer geschalteten prädiktiven Quantisierung;

Mitteln (39) zum Durchsuchen einer Prädiktor-/Codebuchmengenkombination, um LSF-Zielvektoren zu bestimmen, die einen quantisierten Ausgang zur Folge haben, mit dem die LPC-Koeffizienten am besten übereinstimmen, indem sowohl die Prädiktormatrizen als auch die Codebücher umgeschaltet werden, um einen quadratischen Fehler minimal zu machen;

Mitteln (29) zum Eingeben der LSF-Zielvektoren in die Codebücher, um einen quantisierten Ausgang zu schaffen;

wobei die Durchsuchungsmittel Mittel enthalten, um in das LPC-Filter (21) einen Impuls einzugeben;
Mitteln zum Durchlaufenlassen von Abtastwerten der LPC-Antwort;
einem perzeptuellen Filter (47) zum Filtern der Abtastwerte; und
Mitteln (35) zum Berechnen einer Autokorrelationsfunktion durch die gewichtete Antwort, um LSF-Gewichte zu schaffen, die aus der perzeptuell gewichteten Eingangsantwort in das LPC-Filter (21) berechnet werden.

9. Quantisierer (20) nach Anspruch 8, bei dem die Mittel (35) zum Berechnen einer Autokorrelationsfunktion so betreibbar sind, dass sie eine Jacobi-Matrix für die LSF-Vektoren, eine Korrelation von Zeilen der Jacobi-Matrix und die LSF-Gewichte durch Multiplizieren von Korrelationsmatrizen berechnen.

10. Quantisierer (20) nach Anspruch 8 oder Anspruch 9, der ferner so betreibbar ist, dass er:

N Abtastwerte der LPC-Syntheseantwort durchlaufen lässt, wenn die gewichtete Impulsantwort eingegeben wird; und

die Abtastwerte unter Verwendung eines perzeptuellen Filters (47) filtert.

11. Quantisierer (20) nach Anspruch 10, bei dem das perzeptuelle Filter (47) dazu verwendet wird, niedrige Frequenzen stärker als hohe Frequenzen zu gewichten.

12. Quantisierer (20) nach Anspruch 11, bei dem das perzeptuelle Filter (47) der Barkskala folgt.

13. Quantisierer (20) nach einem der Ansprüche 8 bis 12, der ferner so betreibbar ist, dass er eine mehrstufige Vektorquantisierung bereitstellt.

14. Quantisierer (20) nach einem der Ansprüche 8 bis 13, der eine oder mehrere Mengen von Codebüchern (27) umfasst.

15. Codierer, der den Quantisierer (20) nach einem der Ansprüche 8 bis 14 enthält.

Revendications

1. Procédé de quantification vectorielle de coefficients de codage par prédiction linéaire (LPC) comprenant les étapes consistant à :

convertir des coefficients LPC en coefficients de fréquences spectrales de lignes (LSF) ;
procurer une pluralité de matrices de prédiction et un quantificateur avec une pluralité de livres de codes pour quantifier des vecteurs cibles LSF en utilisant une quantification prédictive commutée ;
explorer une combinaison définie d'un prédicteur livres de codes pour déterminer des vecteurs cibles LSF qui produisent en résultat une sortie quantifiée qui corresponde au mieux à des coefficients LPC en commutant à la fois les matrices de prédiction et les livres de codes, ce qui minimise une erreur quadratique ;
appliquer lesdits vecteurs ciblés aux dits livres de codes pour obtenir des vecteurs quantifiés ;
ladite étape d'exploration comprenant une étape consistant à déterminer l'erreur quadratique multipliée par une valeur de pondération pour chaque dimension entre les coefficients de fréquences spectrales de lignes (LSF) et la sortie quantifiée dans laquelle ladite valeur de pondération est une fonction de pondération perceptuelle ;
et ladite étape de calcul de la détermination comprenant les étapes consistant à :

calculer une fonction d'autocorrélation d'une réponse d'impulsion pondérée ; et calculer des poids LSF (56) à partir d'une réponse pondérée selon un mode perceptuel appliquée en entrée.

2. Procédé selon la revendication 1 comprenant le calcul d'une matrice Jacobienne (54) pour lesdits vecteurs cibles LSF ;
le calcul de la corrélation (55) des lignes de la matrice Jacobienne ; et
dans lequel les poids LSF sont calculés en multipliant les matrices de corrélation.

3. Procédé selon la revendication 1 ou la revendication 2, dans lequel ladite étape de détermination comprend les autres étapes pour trouver ladite valeur de pondération consistant à :

appliquer une impulsion au dit filtre LPC et passer en machine N échantillons (51) de la réponse de synthèse LPC ; et
filtrer (52) les échantillons avec un filtre perceptuel ;
calculer la fonction d'autocorrélation (53) de la réponse d'impulsion pondérée ;
calculer la matrice Jacobienne (54) pour lesdits vecteurs LSF ;
calculer la corrélation (55) des lignes de la matrice Jacobienne ; et
calculer les poids LSF (56) en multipliant les matrices de corrélation.

4. Procédé selon la revendication 3 dans lequel l'étape (52) de filtrage des échantillons avec ledit filtre perceptuel comprend une plus grande pondération des fréquences basses que des fréquences hautes.

5. Procédé selon la revendication 4 dans lequel l'étape (52) de filtrage des échantillons avec ledit filtre perceptuel comprend de suivre l'échelle de Bark.

6. Procédé selon l'une quelconque des revendications précédentes dans lequel ladite étape pour procurer ledit quantificateur comprend de procurer un quantificateur vectoriel à étages multiples.

7. Procédé selon l'une quelconque des revendications précédentes dans lequel ladite étape pour procurer ledit quantificateur comprend de procurer un quantificateur ayant un ou plusieurs ensembles de livres de codes.

8. Quantificateur (20) pour un codeur comprenant un filtre LPC (21) et un traducteur (23) pour convertir des coefficients LPC en coefficients LSF comprenant :

un moyen pour procurer une pluralité de matrices de prédiction et une pluralité de livres de codes (27) pour quantifier des vecteurs cibles LSF en utilisant la quantification prédictive commutée ;
un moyen (39) pour explorer une combinaison définie d'un prédicteur/livres de codes pour déterminer des vecteurs cibles LSF qui produisent en résultat une sortie quantifiée qui corresponde au mieux à des coefficients LPC en commutant à la fois les matrices de prédiction et les livres de codes, ce qui minimise une erreur quadratique ;
un moyen (29) pour appliquer lesdits vecteurs cibles LSF aux dits livres de codes pour procurer une sortie quantifiée ;
ledit moyen pour explorer comprenant un moyen pour appliquer une impulsion au dit filtre LPC (21) ;
un moyen pour passer en machine des échantillons de ladite réponse LPC ;
un filtre perceptuel (47) pour filtrer lesdits échantillons ; et
un moyen (35) pour calculer une fonction d'autocorrélation avec une réponse pondérée pour procurer des poids LSF obtenues par calculs à partir de la réponse pondérée selon un mode perceptuel appliquée en entrée, au filtre LPC (21).

9. Quantificateur (20) selon la revendication 8, dans lequel le moyen (35) pour calculer une fonction d'autocorrélation peut être exploité pour calculer une matrice Jacobienne pour les vecteurs LSF, une corrélation des lignes de la matrice Jacobienne, et pour calculer les poids LSF en multipliant des matrices de corrélation.

10. Quantificateur (20) selon la revendication 8 ou la revendication 9, pouvant en outre être exploité pour :

passer en machine N échantillons de la réponse de synthèse LPC lors de l'application de la réponse d'impulsion pondérée ; et
filtrer les échantillons en utilisant un filtre perceptuel (47).

11. Quantificateur (20) selon la revendication 10, dans lequel le filtre perceptuel (47) est utilisé pour pondérer davantage des fréquences basses que des fréquences hautes.

12. Quantificateur (20) selon la revendication 11, dans lequel le filtre perceptuel (47) suit l'échelle de Bark.

13. Quantificateur (20) selon l'une quelconque des revendications 8 à 12, pouvant en outre être exploité pour procurer une quantification vectorielle à étages multiples.

14. Quantificateur (20) selon l'une quelconque des revendications 8 à 13, comprenant un ou plusieurs ensembles de livres de codes (27).

15. Codeur intégrant le quantificateur (20) selon l'une quelconque des revendications 8 à 14.

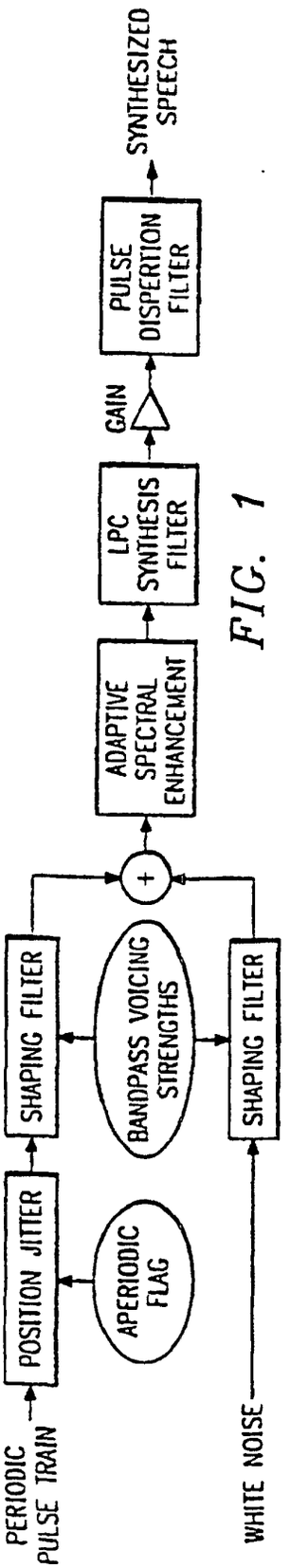
PERIODIC PULSE TRAIN
POSITION JITTER
APERIODIC FLAG
SHAPING FILTER
BANDPASS VOICING STRENGTHS
WHITE NOISE
SHAPING FILTER
ADAPTIVE SPECTRAL ENHANCEMENT
LPC SYNTHESIS FILTER
GAIN
PULSE DISPERTION FILTER
SYNTHESIZED SPEECH

FIG. 1

LPC COEFFICIENTS
IMPULSE
TRANSFORM
23
SYNTHESIS FILTER
21
WEIGHTING FILTER
47
COMPUTER
39
CONTROL
29
ERROR
ENCODER
SQUARER
35
20
LSF
22
INPUT
ADDER
PREDICTION MATRIX 1 MEAN 1
PREDICTION MATRIX 2 MEAN 2
24a
24b
SWITCH
28
MEAN VALUE
ADDER
25
MULTIPLIER
26
CODEBOOKS 1
CODEBOOKS 2
MSVQ
27
PREDICTIVE VALUE
DELAY
33
G
33a
ADDER
31
ADDER
70
G
37
INDICES
TERMINAL
38

FIG. 2

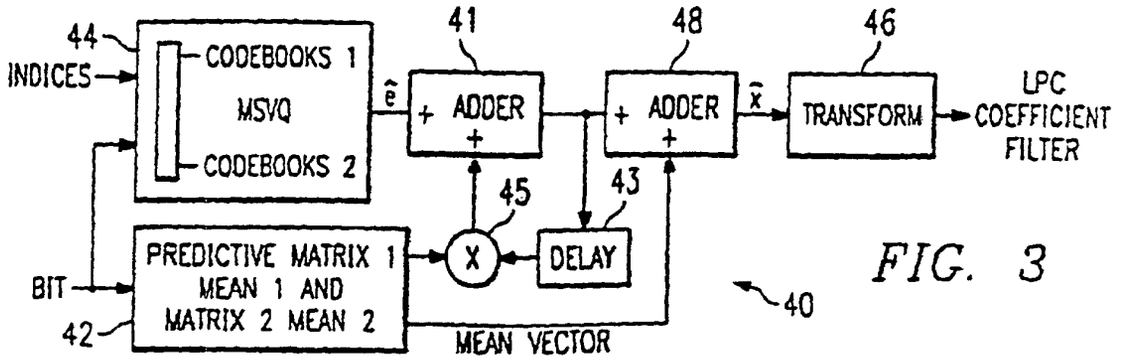

FIG. 3

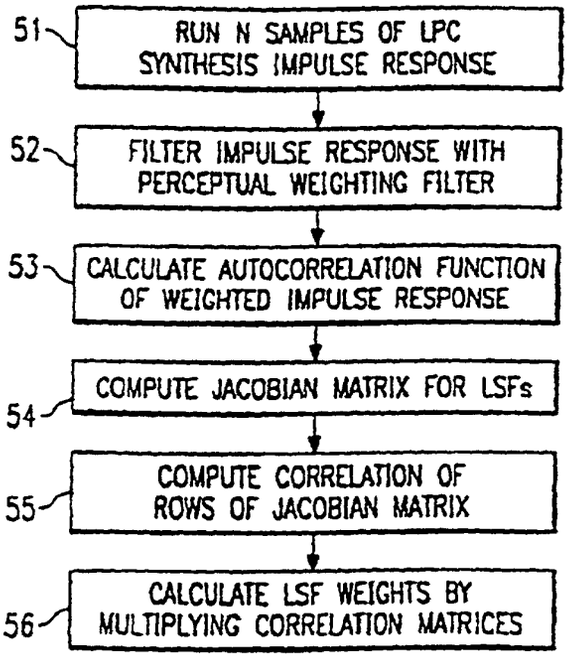

FIG. 4

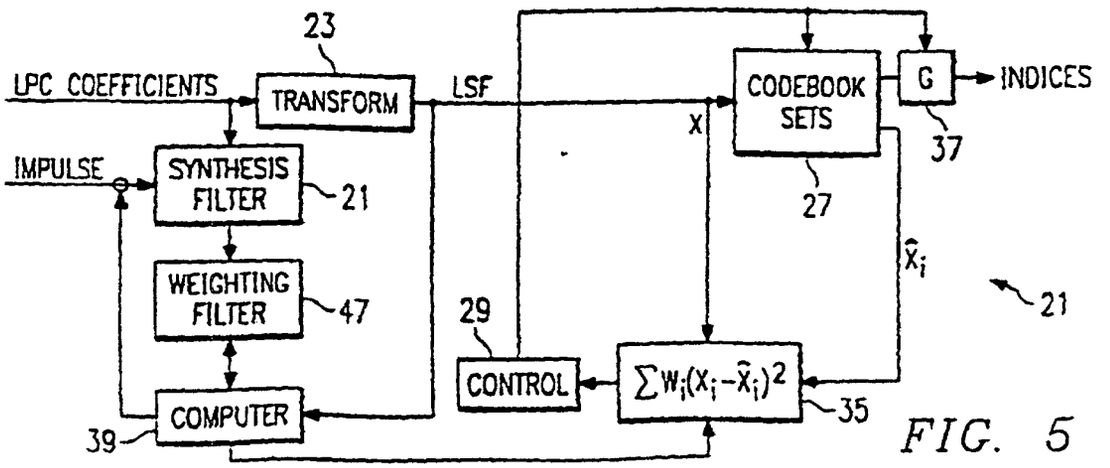

FIG. 5