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

(11) EP 0 774 750 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
21.05.1997 Bulletin 1997/21

(51) Int Cl.6: G10L 9/14

(21) Application number: 96308081.7

(22) Date of filing: 07.11.1996

(84) Designated Contracting States:
DE FR GB SE

(30) Priority: 15.11.1995 US 6787
23.10.1996 US 735827

(71) Applicant: NOKIA MOBILE PHONES LTD.
24101 Salo (FI)

(72) Inventor: Ruoppila, Vesa T.
33720 Tampere (FI)

(74) Representative: Slingsby, Philip Roy et al
NOKIA MOBILE PHONES,
Patent Department,
St. Georges Court,
St. Georges Road,
9 High Street
Camberley, Surrey GU15 3QZ (GB)

(54) Determination of line spectrum frequencies for use in a radiotelephone

(57) An audio signal, such as a speech signal, is applied to a LPC filter. A method is disclosed for determining line spectrum frequencies of the LPC filter, expressed by symmetric and antisymmetric auxiliary polynomials, the zeros of which determine the line spectrum frequencies of the LPC filter. The method includes the steps of (a) expressing the auxiliary polynomials using explicit forms of Chebyshev polynomials; (b) iteratively solving a zero of a first of the polynomials using a zero of the other one of the polynomials; and (c) successively eliminating zeroes from the polynomials by polynomial deflation so as to determine the line spectrum frequencies. Also disclosed is a method for determining the immittance spectrum frequencies.

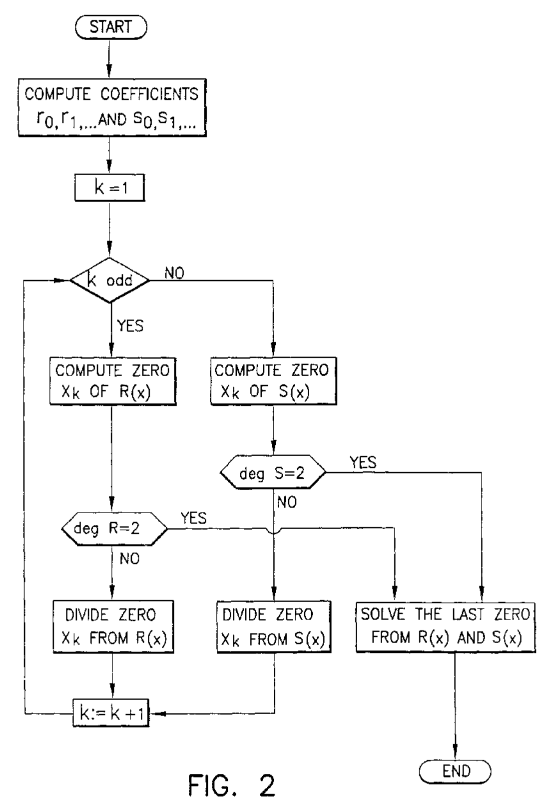

FIG. 2

EP 0 774 750 A2

Description

This invention relates generally to speech encoding methods and apparatus and, in particular, to linear predictive coding (LPC) speech and audio coding techniques that employ a line spectrum frequency representation of an LPC filter.

Linear predictive coding (LPC) is a known technique for analyzing a speech signal and for characterizing the signal in terms of coefficients which are encoded, broadcast, received and decoded to recover an approximation of the original signal. The parameters of a LPC filter are coded and sent as a part of the information stream. The use of line spectrum frequencies is an alternative to the use of, for example, polynomial coefficients or reflection coefficients for representing the LPC filter. The line spectrum frequencies have useful properties for quantization and interpolation which make them a more attractive representation than polynomial or reflection coefficients.

However, in known types of speech encoders that employ line spectrum frequencies, the procedure for deriving the line spectrum frequencies is computationally expensive. This disadvantage becomes especially apparent when implemented in real time or substantially real time in a speech encoder of, by example, a digital cellular telephone.

It is a first potential aim of this invention to provide a simplified and efficient technique for reducing the complexity of the line spectrum frequency computation.

It is a second potential aim of this invention to provide a radiotelephone having an LPC-based speech/audio encoder that employs line spectrum frequencies that are obtained in accordance with the method of this invention.

The foregoing and other problems may preferably be overcome and the potential aims of the invention realized by methods and apparatus in accordance with embodiments of this invention.

In this invention, a method for determining line spectrum frequencies of a LPC filter is disclosed. The predictor polynominal of the LPC filter is decomposed into symmetric and antisymmetric auxiliary polynomials, the zeros of which determine the line spectrum frequencies of the filter. In other words, the line spectrum frequency representation is determined by solving the zeros of the two auxiliary polynomials. Due to the symmetry of the auxiliary polynomials, their zeros are preferably solved from two cosine series. In speech codecs this is usually done by a bisection algorithm, and by employing the definition of Chebyshev polynomials to evaluate the cosine series.

A potential aim of this invention is to reduce the complexity of the line spectrum frequency computation. This may be obtained by rewriting the cosine series as polynomials using explicit forms of Chebyshev polynomials. This enables an evaluation of the series by nested multiplications. Moreover, the already-computed zeros are successively eliminated from the polynomial by polynomial deflation. This procedure and the properties of the auxiliary polynomials can enable the initial values to be chosen in the zero finding algorithm such that a zero is found by only a few iterations using the zero of the other polynomial. Thus, the invention may reduce considerably the arithmetic operations required to compute the line spectrum frequencies. The method of this invention thus has the potential to be implemented with relatively low complexity, and furthermore to be accomplished using fixed-point arithmetic.

The above set forth and other features of the invention are made more apparent in the ensuing detailed description of the invention when read in conjunction with the attached Drawings, wherein:

Fig. 1 is a block diagram showing a speech encoder employing a line spectrum frequency representation of the LPC filter.

Fig. 2 is a flow chart describing one method to implement the present invention. The procedure employed for computing the zeros of the auxiliary polynominals R and S (the block Compute zero) is presented in more detail in Figure 3.

Fig. 3 is a flow chart of one method to implement Newton's method. The polynomial G = R if i is odd. Otherwise G = S. G'(x) denotes the first derivative of the polynomial G at the point x. An additional refining procedure is often unnecessary if sufficient numerical accuracy is employed.

Figs. 4A-4D illustrate the progress of the technique (algorithm (29)) employing polynomial deflation.

A simplified block diagram of a speech encoder 10 employing the spectrum frequency representation of the LPC filter is presented in Fig. 1. The speech encoder 10 may form a portion of a radiotelephone, such as a digital cellular user terminal or a personal communicator device. An input audio signal, such as a speech signal obtained from a speech transducer or microphone 5, is converted into a digital form by an analog-to-digital (A/D) converter 12. The digital output of the A/D converter is preprocessed by separating the signal into frames of convenient length, typically of the order of tens of milliseconds. It should be noted that the A/D conversion is not necessary if the signal is already in digital form. After preprocessing, the signal is applied to an LPC analysis block 14. The LPC analysis produces coefficients for an LPC filter, also referred to herein as an LPC-analysis filter 16. The output of the LPC analysis block

14 is transformed into a line spectrum frequency (LSF) representation in block 18. The LSF coefficients may be quantized in block 20 and then interpolated in block 22 in order to construct a LPC analysis filter for each speech subframe. By example, and for a speech frame having a duration of 20 milliseconds, four 5 millesecond subframes may be used, wherein the analysis filter is constructed separately for each subframe. After the LPC filtering in block 16 the coefficients of a long-term prediction (LTP) filter are searched, and the residual is generated in the block labeled LTP analysis and filtering 24. The residual is encoded in the excitation encoding block 26, and the resulting encoded residual, i.e., an encoded excitation signal, is multiplexed (block 28) with the quantized LSF coefficients into a bit stream transmitted to a speech decoder (not shown) via a communication channel 30. By example, the communication channel 30 is a radio channel linking the mobile terminal to a base station (not shown) by a transmitter 32 and an antenna 34. The "side information" input to the multiplexer 28 determines, for example, the operational mode of the speech coder, particularly for variable rate codecs such as QCELP. For a speech coder operated in a fixed rate mode, this input may not be required.

The following detailed description of the invention pertains most particularly to the operation of the LPC to LSF block 18 of Fig. 1. The present invention does not require any modifications to other blocks presented in Fig. 1. It is also noted that the LPC is LSF transformation is not used in a decoder of present speech codecs. Therefore, the decoder of the speech codec is not considered in this description, although a decoder that employs a LPC to LSF transformation is also within the scope of the teaching of this invention.

1.1 Line Spectrum Frequency Representation

The nth degree predictor polynomial of the LPC filter

$$(1) \qquad A_n(z) = 1+a_1 z^{-1}+...+a_n z^{-n}$$

satisfies the recurrence formula

$$(2) \qquad A_{n+1}(z) = A_n(z)+k_{n+1} z^{-n-1} A_n(z^{-1}),$$

wherein $k_1, k_2,...,k_{n+1}$ are reflection coefficients. The recurrence formula (2) is called the Levinson-Durbin solution to the Yule-Walker equations. It expresses the relationship between the (n+1)th and the nth degree predictor polynomials. For the purposes of this description it is assumed that all roots of the predictor polynomial $A_n(z)$ are inside the unit circle, i.e., that the predictor polynomial is minimum phase.

By setting $k_{n+1} = 1$, the recurrence formula (2) gives the polynomial:

$$(3) \qquad P_{n+1}(z) \triangleq A_n(z) + Z^{-n-1} A_n(z^{-1}).$$

By construction, (3) is a symmetric polynomial in a sense that it satisfies the relation

$$P_{n+1}(z) = z^{-n-1} P_{n+1}(z^{-1}).$$

Similarly, by setting $k_{n+1} =-1$ in (2) one obtains an antisymmetric polynomial

$$(4) \qquad Q_{n+1}(z) \triangleq A_n(z) - z^{-n-1} A_n(z^{-1})$$

which has the property

$$Q_{n+1}(z) = -z^{-n-1} Q_{n+1}(z^{-1}).$$

From (3) and (4) it follows that the predictor polynomial (1) can be decomposed into a sum of symmetric and antisymmetric polynomials:

(5) $A_n(z) = (1/2)(P_{n+1}(z) + Q_{n+1}(z))$.

Actually, the roots of the polynomials $P_{n+1}(z)$ and $Q_{n+1}(z)$ determine the line spectrum frequencies of the predictor polynomial. Soong and Juang ("Line spectrum pair (LSP) and speech data compression", Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, San Diego, CA, pp.1.10.1-1.10.4, March 1984) have shown that if $A_n(z)$ is minimum phase, then the roots of $P_{n+1}(z)$ and $Q_{n+1}(z)$ are on the unit circle, and the roots are simple and separate from each other. Therefore, $P_{n+1}(z)$ and $Q_{n+1}(z)$ can be factored as follows:

$$(6) \quad P_{n+1}(z) = \begin{cases} (1+z^{-1}) \prod_{i=1,3,\ldots,n-1} (1-2z^{-1}\cos\omega_i + z^{-2}) & n \text{ even} \\ \prod_{i=1,3,\ldots,n} (1-2z^{-1}\cos\omega_i + z^{-2}) & n \text{ odd} \end{cases}$$

$$(7) \quad Q_{n+1}(z) = \begin{cases} (1-z^{-1}) \prod_{i=2,4,\ldots,n} (1-2z^{-1}\cos\omega_i + z^{-2}) & n \text{ even} \\ (1-z^{-2}) \prod_{i=2,4,\ldots,n-1} (1-2z^{-1}\cos\omega_i + z^{-2}) & n \text{ odd} \end{cases}$$

where $\omega_1, \omega_2, \ldots, \omega_n$ are the phase angles of the zeros of the polynomials

$$(8) \quad P(z) \triangleq \prod_{i=1,3\ldots} (1-2z^{-1}\cos\omega_i + z^{-2})$$

$$(9) \quad Q(z) \triangleq \prod_{i=2,4\ldots} (1-2z^{-1}\cos\omega_i + z^{-2})$$

such that

(10) $0 < \omega_1 < \omega_2 < \cdots < \omega_n < \pi$

$\omega_1, \omega_2, \ldots, \omega_n$ are the line spectrum frequencies of $A_n(z)$.

Note that both polynomials P(z) and Q(z) are symmetric, and the degree of P(z) is $2m_p$, where

$$m_p = \begin{cases} n/2 & n \text{ even} \\ (n+1)/2 & n \text{ odd} \end{cases}$$

The degree of the polynomial Q(z) is $2m_Q$, where

$$m_q = \begin{cases} n/2 & n \text{ even} \\ (n-1)/2 & n \text{ odd} \end{cases}$$

By explicitly using the symmetry of the polynomial, P(z) defined in (8) can be written in the form

$$(11)\qquad P(z) = 1 + p_i z^{-1} + p_2 z^{-2} + .. + p_2 z^{-2mp+2} + p_1 z^{-2mp+1} + z^{-2mp}$$
$$= z^{-mp} \{(z^{mp} + z^{-mp}) + p_1 (z^{mp-1} + z^{-mp+1}) + .. + p_{mp}\},$$

where p_1, p_2,..., p_{mp} are the coefficients of P(z). By substituting $z = e^{j\omega}$ and by employing the relation

$$z^k + z^{-k} = e^{j\omega k} + e^{-j\omega k} = 2\cos\omega k,$$

equation (11) gives

$$(12)\quad P(\omega) = e^{-j\omega mp}\left\{2\cos m_p\omega + 2p_1\cos(m_p - 1)\omega + \cdots + p_{mp}\right\}$$

The symmetric polynomial Q(z) can be rewritten similarly to the form

$$(13)\quad Q(\omega) = e^{-j\omega mQ}\left\{2\cos m_Q\omega + 2q_1\cos(m_Q - 1)\omega + \cdots + q_{mQ}\right\}$$

where q_1, q_2,..., q_{mQ} are the coefficients of the polynomial Q(z).

From equation (8) it follows that line spectrum frequencies $\omega_1, \omega_3, \ldots, \omega_{2mp}$ are the zeros of P(ω) in the interval [0, π]. Correspondingly, line spectrum frequencies ω_2, ω_4,..., ω_{2mQ} are the zeros of Q(ω) when $\omega \in [0,\pi]$. Hence, the line spectrum frequencies of $A_n(z)$ can be found by solving zeros of series

$$(14)\qquad R(\omega) \triangleq \cos m_p\omega + p_1\cos(m_p - 1)\omega + \ldots + (1/2)p_{mp},$$

$$(15)\qquad S(\omega) \triangleq \cos m_Q\omega + q_1\cos(m_Q - 1)\omega + \ldots + (1/2)q_{mQ},$$

where $\omega \in [0, \pi]$. In a QCELP speech coder, by example, line spectrum frequencies are solved directly from (14) and (15) (see, by example, TIA/EIA/IS-96-A, Speech Service Option Standard for Wideband Spread Spectrum Digital Cellular System (1994)). However, it is more desirable to rewrite the cosine series as polynomials in order to obviate the evaluation of trigonometric functions. This is discussed in more detail in the following section.

The relations between the coefficients of the predictor polynomial and the coefficients of the cosine series (14) and (15) can be derived from equations (3), (4) and (6)-(9).

1.2 Chebyshev Polynomials

The Chebyshev polynomials of the first kind are defined by the recurrence formula

$$(16)\qquad T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \qquad k = 1,2,\ldots$$

with initial conditions $T_0(x) = 1$ and $T_1(x) = x$. For x in the interval [-1,1], the Chebyshev polynomials have the closed-form expression

$$(17)\qquad T_k(x) = \cos\{k \arccos x\}, \qquad k = 0,1,\ldots$$

The explicit forms of the first few T_k are

$$T_2(x) = \cos\{2\arccos x\} = 2x^2 - 1,$$

$$T_3(x) = \cos\{3\arccos x\} = 4x^3 - 3x,$$

$$T_4(x) = \cos\{4\arccos x\} = 8x^4 - 8x^2 + 1,$$

$$T_5(x) = \cos\{5\arccos x\} = 16x^5 - 20x^3 + 5x.$$

By changing a variable,

$$(18) \qquad \omega = \arccos x,$$

and using (12), equations (14) and (15) give

$$(19) \qquad R(x) = T_{mp}(x) + p_1 T_{mp-1}(x) +..+ (1/2)P_{mp},$$

$$(20) \qquad S(x) = T_{mQ}(x) + q_1 T_{mQ-1}(x) +..+ (1/2)q_{mQ},$$

The line spectrum frequencies $\{\omega_i\}$ can be determined by solving the equations R(x) =0 and S(x) =0 for x in the interval [-1,1]. Once the roots $\{x_i\}$ are solved, the corresponding line spectrum frequencies are given by $\omega_i = \arccos x_i$.

Various methods to solve the zeros of (14) and (15) have been suggested in the literature. So as to provide a background for the teaching of this invention, a brief survey is made of a few algorithms presented in the literature. Also, the methods employed in some currently standardized speech codecs are discussed.

The procedure introduced by Soong and Juang evaluates (14) and (15) on a fine grid by discrete cosine transformation. Sign changes at adjacent grid points isolate intervals containing roots. After a sign change has been detected, the interval is bisected until a sufficiently accurate numerical estimate for the zero is obtained. A similar bisection based algorithm is used also in the QCELP speech codec (i.e., TIA/EIA/IS-96-A, Speech Service Option Standard for Wideband Spread Spectrum Digital Cellular System (1994)). However, in QCELP the equations (14) and (15) are evaluated by computing directly all the terms of the series without transformations. Since a large number of trigonometric functions have to be evaluated, the algorithm becomes inevitably complex, and the numerical accuracy may be poor in a fixed-point implementation.

Kang and Fransen ("Application of line spectrum pairs to low bit rate speech encoders", Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Tampa, FL, pp. 7.3.1-7.3.4, March 1984) have proposed autocorrelation based and ratio-filter based methods for finding line spectrum frequencies. However, these two methods also require an evaluation of a large number of trigonometric functions.

In order to obviate evaluation of trigonometric functions, Kabal and Ramachandran ("The computation of line spectrum frequencies using Chebyshev polynomials", IEEE Transactions on Speech and Audio Processing, vol. 34, no. 6, pp. 1419-1426, 1986) suggested that the cosine series (14) and (15) are transformed to the form (19) and (20). Then line spectrum frequencies $\{\omega_i\}$ can be determined by solving the zeros of (19) and (20) in the interval [-1,1]. Also Kabal and Ramachandran employed the bisection algorithm; Equations (19) and (20) are evaluated on a grid to locate sign changes at adjacent grid points. After a sign change has been detected, the zero is computed by successively bisecting the interval. Since this method is currently used in several speech codecs, e.g., in (TIA/EIA/IS-641 TDMA Cellular/PCS-Radio Interface-Enhanced Full-Rate Speech Codec (1996)), it is described in more detail in the next section.

Saoudi et al. ("A new efficient algorithm to compute the LSP parameters for speech coding", Signal Processing, vol. 28, pp. 201-212, 1992) reformulated the problem. That is, they introduced an algorithm which solves line spectrum frequencies from eigenvalues of tridiagonal symmetric matrices without computing the predictor polynomial. The eigenvalues are computed by the bisection method. Saoudi et al. also compared the complexity of several algorithms introduced for computing the line spectrum frequencies. Unfortunately, the results of the comparison are not comprehensive.

Also it is noted that Chan ("Computation of LSF parameters from reflection coefficients," Electronic Letters, vol. 27, no. 19, pp. 1773-1774, 1991, and "Efficient interconversion algorithm for PARCOR and LSP parameters," in Proceedings of International Symposium on Speech, Image Processing and Neural Networks, Hong Kong, pp. 603-606, 13-16 April 1994) has introduced a method for forming polynomials $P_{k=1}(z)$ and $Q_{k=1,2...,n}(z)$ for k without computing explicitly the coefficients of the predictor polynomial. However, this approach does not solve the zero finding problem.

As was mentioned above, Kabal and Ramachandran suggested that the cosine series (14) and (15) be rewritten to the form (19) and (20) by exploiting the Chebyshev polynomials. This allows an evaluation of the equations with a simple recursion, and without trigonometric functions. Kabal and Ramachandran have used the property (10), or equivalently

$$(21) \qquad 1 > x_1 > x_2 > ... > x_n > -1,$$

to reduce the complexity of their algorithm.

(23) Algorithm

(a) Initialization. Compute the coefficients of (19) and (20) when $A_n(z)$ is given. Partitate the upper half of the unit circle into N subintervals $[\omega_{Gk}, \omega_{G(k+1)}]$, k = 1,2, ..., N and such that $\omega_{G1} = 0$ and $\omega_{GN} = \pi$. Map the grid points $\{\omega_{Gk}\}$ into the real axis by $x_{Gk} = \cos \omega_{Gk}$. Set k = 1 and start the search from the polynomial R(x).

(b) Check if the polynomial under investigation has a sign change in the interval $[x_{Gk}, x_{G(k+1)}]$.

(c) If a sign change is detected, locate the zero x_i from the interval $[x_{Gk}, x_{G(k+1)}]$ by the bisection algorithm. After the zero has been found, continue the search from another polynomial.

(d) Set k: = k+1. Continue to (b) until all n zeros are found or all intervals are gone through.

Although the algorithm (22) is known to be used in speech codecs, it has several shortcomings. For example, if some zeros of the polynomials are sufficiently near each other, the algorithm has a tendency to miss zeros, since the sign change is not detected. This shortcoming can be circumvented by making the grid denser. In other words, the intervals are made sufficiently small so that two or more roots do not occur in the same interval. The definitive choice of the grid interval is always a compromise between a reliability and computational burden. That is, as the grid interval is made more dense, the overall processing burden increases as well.

In accordance with the teaching of this invention, a reduction in the complexity of the speech encoding, and more particularly, the computation of line spectrum frequencies is made possible. In accordance with the teaching of this invention, a significant improvement is achieved by evaluating (19) and (20) with a recursion requiring fewer arithmetic operations than the methods used in current speech encoders.

Further in accordance with the teaching of this invention, the method successively eliminates already-found zeros from the polynomials. The procedure is known as polynomial deflation, or as synthetic division in numerical analysis, see, e.g., Kincaid and Chaney. The use of polynomial deflation allows for the elimination of the sign-change detection procedure from the algorithm. Another advantage is that efficient algorithms (which have better convergence properties than the bisection method) can be exploited together with the property (21) to locate the zeros. For example, when zeros are computed by Newton's method, the algorithm is guaranteed to find all line spectrum frequencies if sufficient numerical accuracy is used.

The methods of this invention are presented in more detail in the following discussion.

Consider now the evaluation of R(x) and S(x) for a given value of x in the interval [-1, 1]. The Chebyshev series (19), and similarly (20), to be evaluated can be represented as

$$(23)\ R(x) = \sum_{k=1}^{mp} p_{mp-k} T_k(x) + (1/2) p_{mp} \triangleq \sum_{k=0}^{mp} a_k T_k(x)$$

with $p_0 = 1$, $a_0 = (1/2)\, p_{mp}$ and $a_k = p_{mp-k}$. The summation in (23) reduces to

$$(24)\ R(x) = \sum_{k=0}^{mp} a_k T_k(x) = b_0(x) - b_2(x) + a_0,$$

where $b_0(x)$ and b_2 (x) are given by the backward recurrence formula

$$b_k(x) = 2xb_{k+1}(x) - b_{k+2}(x) + a_k,$$

$$b_{mp}(x) = b_{mp+1}(x) = 0.$$

Kabal and Ramachandran have employed this technique in their algorithm.

Although (25) is a numerically robust method for evaluating the Chebyshev series, it requires (m_p + 1) multiplications and ($2m_p$ - 1) additions in this particular application. Also several data-move operations increase the workload in a DSP implementation.

In this invention, there is instead employed a more efficient method. To evaluate R(x) and S(x), instead convert (19) and (20) to the polynomials

$$(25) \qquad R(x) = r_0 x^{mp} + r_1 x^{mp-1} + .. + r_{mp},$$

$$(26) \qquad S(x) = s_0 x^{mQ} + s_1 x^{mQ-1} + .. + s_{mQ},$$

with the explicit forms of the Chebyshev polynomials. The polynomials (25) and (26) can then be evaluated effectively by the procedure of nested multiplications, known also as Horner's algorithm (see, for example, Kincaid and Cheney, Numerical Analysis: Mathematics of Scientific Computing, Brooks/Cole Publishing Company, 1991).

Horner's algorithm produces $R(x) = b_0(x)$, and similarly S(x), by the backward recurrence relation

$$(27) \qquad b_k(x) = xb_{k+1}(x) + r_k$$

with an initial value $b_{mp}(x) = r_{mp}$. The equation (27) requires m_p multiplications and m_p additions. Naturally, the recursion (27) can be used also in other mathematically equivalent forms.

Example

Consideration is made of the case when the degree of $A_n(z)$ is ten (n = 10). Hence $m_p = 5$. The conversion from (19) to (25) gives polynomial coefficients

$r_0 = 16$, $r_1 = 8p_1$,
$r_2 = 4p_2 - 20$, $r_3 = 2p_3 - 8p_1$,
$r_4 = -3p_2 + p_4 + 5$ $r_5 = p_1 - p_3 + p_5/2$,

when the polynomial

$R(x) = \{16x^5 - 20x^3 + 5x\} + p_1 \{8x^4 - 8x^2 + 1\}$
$+ p_2\{4x^3 - 3x\} + p_3\{2x^2 - 1\} + p_4 x + p_5/2$

is written in the open form. In this form the evaluation of R(x) requires 5 multiplications and 5 additions using Horner's algorithm, while the method (24) would require 6 multiplications and 9 additions. The difference is a total of five operations (plus data-move operations in a fixed-point implementation). As can be appreciated, in some speech coders the polynomials are evaluated hundreds of times in a single speech frame during the computation of line spectrum frequencies and, as a result, a large number of operations can be eliminated. The computation of $r_0, r_1, \ldots, r_5$ requires six additions and six multiplications. However, these coefficients are usually computed only once per frame.

A discussion is now made of the polynomial deflation aspects of this invention. It is first pointed out that it is possible to avoid the checking of sign-changes if already-found zeros are successively eliminated from the polynomials R(x) and S(x), or equivalently from R(ω) and S(ω). In numerical analysis this procedure is known as polynomial deflation or synthetic division. The polynomial deflation is based on the relation

$$(28) \qquad R(x) = r_a(x - x_1)(x - x_3) \ldots \underline{\Delta}\, r_a(x - x_1)R_Q(x).$$

By example, assume that n is even, and a zero $x = x_1$ of R(x) has been located. Then, the remaining zeros $x_3, \ldots, x_{n-1}$ are also zeros of $R_Q(x)$. To compute these zeros, the polynomial R(x) can be replaced by the quotient polynomial

$R_Q(x)$. This is referred to as the polynomial deflation. The procedure can be repeated; as soon as a zero is found, it can be factored out. In this way, the zero finding technique can operate with polynomials of lower and lower degree. The known zeros can be removed from a polynomial by the use of Horner's algorithm. Clearly, the use of polynomial deflation saves arithmetic operations.

Now, the algorithm employing polynomial deflation for computing the line spectrum frequencies, in accordance with the teaching of this invention, can be summarized as follows:
(29) Algorithm

(a) Initialization. Compute the coefficients of (25) and (26) when $A_n(z)$ is given. Start the search from the polynomial R(x) with an initial value $x_0 = 1$. Set k = 1.

(b) Compute the zero $x_k \in (-1, x_{k-1})$ by Newton's method. The previous zero x_{k-1} is a reasonable initial value. See Fig. 3.

(c) Refine the numerical estimate of the zero x_k by taking additional steps in Newton's method using the undeflated polynomial (Fig. 3). It is noted that this step of refining the numerical estimate is optional.

(d) If the degree of the polynomial is two, solve the last zero analytically (see the next section). Otherwise eliminate the zero x_k from the polynomial by the polynomial division technique. Continue the search from the other polynomial.

(e) Set k: = k + 1. Continue to (b) until all n zeros are found.

<u>Solution of the last zero of R(x) and S(x)</u>

When the deflated polynomials R(x) and S(x) are of degree 2, and their other zero is known, the last zero of the polynomials can be solved analytically. For example, consider the polynomial

$$R_Q(x) = r_{Q0}x^2 + r_{Q1}x + r_{Q2},$$

which is obtained by dividing already-found zeros from the polynomial R(x). Zeros of $R_Q(x)$ are denoted by x_{Q1} and x_{Q2}.

Assume now that the zero x_{Q1} has already been found, and the task is to solve x_{Q2}. Since the polynomial division $(r_{Q0}x^2 + r_{Q1}x + r_{Q2})/(x - x_{Q1})$ gives the quotient $R_{Q1}(x) = r_{Q0}x + r_{Q1} + r_{Q0}x_{Q1}$, the zero x_{Q2} is readily obtained from the equation $R_{Q1}(x) = 0$. The equation gives $x_{Q2} = -x_{Q1} - r_{Q1}/r_{Q0}$. By this way the computation of the square root resulting from the solution of $R_Q(x) = 0$ is obviated. Moreover, unnecessary Newton's iterations are not needed.

This procedure is applied in step (d) of algorithm (29), i.e., in the block Solve the last zero in Fig. 2.

Newton's method is described in Fig. 3 in more detail. Both the value of the polynomial, and its first derivative, are computed effectively by Horner's method.

Errors in the numerical estimate of x_k accumulate when k increases because the zeros of the successive quotient polynomials deviate more and more from the zeros of the undeflated polynomial. This is obviated by refining the zeros given by step (b) of algorithm (29). The zero can be refined by taking additional steps in Newton's method, but now using the undeflated polynomials (Fig. 3). However, if an adequate numerical accuracy is used, the refining step is not usually necessary.

The term 1/G(x) employed in the Newton's method can be approximated, for example, by tabulating its values in an appropriate range of G(x). If this approximation is denoted by $G_A(x)$, the Newton's step $x_k := x_k - G(x_k)/G(x_k)$ can be simplified to the form $x_k := x_k - G_A(x)G(x_k)$. Hence the division operation is not needed. However, it is obvious that the approximation weakens the convergence properties of the zero finder, although it also reduces the complexity of the fixed-point implementation.

In addition to the Newton's method, other algorithms such as the secant method, see Kincaid and Cheney, can be used for locating or refining the zeros.

Also the initial values can be chosen differently. For example, the algorithm can be started from $x_0 = -1$. In this case the estimates of zeros given by the procedure would be in ascending order $-1 < x_1 < x_2 < \ldots < x_n < 1$.

It should be noted that all equations required in the algorithm (29) can be derived straightforwardly in accordance with the teaching of this invention for the more general form of the polynomial $A_n(z)$, that is

$$A_n(z) = a_0 + a_1 z^{-1} + \ldots + a_n z^{-n}.$$

Consequently, the teaching of the present invention is not limited to the all-pole form of the LPC filter.

Figs. 4A-4D illustrate the progress of the technique (algorithm (29)) employing polynomial deflation. The zeros of the polynomials R(x) and S(x) corresponding to $A_{10}(z)$ are presented in Fig. 4A by crosses (x) and circles (o), respectively. The zero locating procedure is begun from $x_0 = 1$ (Fig. 4B). After the first zero x_1 has been found, $(x - x_1)$ is divided from the polynomial R(x). The polynomial R(x) after the first deflation is shown in Fig. 4C. Next, the search is continued from S(x) using x_1 as an initial value. When x_2 has been located, the procedure is again switched back to the deflated R(x) polynomial, and x_2 acts as a new initial value. The procedure is continued until all zeros have been found. Fig. 4D illustrates the polynomial R(x) after two deflations.

It can thus be appreciated that the is invention teaches in one aspect a method for determining line spectrum frequencies of a linear predictive coder (LPC) filter that is expressed as symmetric and antisymmetric polynomials, the zeros of which determine the line spectrum frequencies of the LPC filter. The method includes the steps of (a) expressing the polynomials using explicit forms of Chebyshev polynomials; (b) interatively solving a zero of a first of the polynomials using a zero of the other one of the polynomials; and (c) successively eliminating zeros from the polynomials by polynomial division so as to determine the line spectrum frequencies.

With regard to the step of iteratively solving, it should be noted that the first polynomial can be either the symmetric or the antisymmetric polynomial, in which case the other polynomial is then the antisymmetric or the symmetric polynomial, respectively. Reference with regard to this step can also be made to elements (b) and (d) of algorithm (29), the equation xk=xk-1 in Fig. 3, and the two separate branches (k odd/even) in Fig. 2. It should be further noted that any of the zeros of the first polynomial can be solved using any of the zeros of the other polynomial.

1.3 Immittance Spectrum Frequency Representation

Reference can be made to Y. Bistritz and S. Peller, "Immittance spectral pairs (ISP) for speech encoding," in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Minneapolis, MI, U.S. A., Vol. 2, pp. 9-12, 27-30 April 1993, for a more detailed understanding of this section.

The immittance spectrum frequency representation of the LPC filter is based on a similar polynomial decomposition of a predictor polynomial as the line spectrum frequency representation. The immittance spectrum frequency representation (see Y. Bistritz and Peller) is obtained when the polynomial $A_n(z)$ is decomposed as

$$(30) \qquad A_n(z) = (1/2)(P_n(z) + Q_n(z)),$$

where

$$(31) \qquad P_n(z) \triangleq A_n(z) + z^{-n} A_n(z^{-1}),$$

$$(32) \qquad Q_n(z) \triangleq A_n(z) - z^{-n} A_n(z^{-1}).$$

The symmetric polynomial $P_n(z)$ and the antisymmetric polynomial $Q_n(z)$ also have similar properties as polynomials (3) and (4). The roots of $P_n(z)$ and $Q_n(z)$ are on the unit circle, and they are simple and separate from each other. Hence, $P_n(z)$ and $Q_n(z)$ can be factored as follows:

$$(33) \qquad P_n(z) = \begin{cases} \prod_{i=1,3,\dots,n-1} (1 - 2z^{-1}\cos\omega_i + z^{-2}) & n \text{ even} \\ (1 + z^{-1}) \prod_{i=1,3,\dots,n-2} (1 - 2z^{-1}\cos\omega_i + z^{-2}) & n \text{ odd} \end{cases}$$

$$(34)\qquad Q_n(z)=\begin{cases} K(1-z^{-2})\prod\limits_{i=2,4,\ldots,n-2}(1-2z^{-1}\cos\omega_i+z^{-2}) & n\ even\\ K(1-z^{-1})\prod\limits_{i=2,4,\ldots,n-1}(1-2z^{-1}\cos\omega_i+z^{-2}) & n\ odd\end{cases}$$

where $K = -(k_n + 1)/(k_n - 1)$, and $\omega_1, \omega, \ldots, \omega_{n-1}$ are the phase angles of the zeros of the polynomials

$$(35)\qquad P(z)\underline{\Delta}\prod_{i=1,3\ldots}(1-2z^{-1}\cos\omega_i+z^{-2})$$

$$(36)\qquad Q(z)\underline{\Delta}\prod_{i=2,4\ldots}(1-2z^{-1}\cos\omega_i+z^{-2})$$

such that

$$0 < \omega_1 < \omega_2 < \ldots < \omega_{n-1} < \pi.$$

The phase angles $\omega_1, \omega_2, \ldots, \omega_{n-1}$ and the parameter k give an unique parametrization for the LPC filter. The properties of the immittance spectrum frequencies and their relation to line spectrum frequencies have been discussed in more detail by Bistritz and Peller.

It should be noted that the polynomials (35) and (36) are not identical to the polynomials defined in (3) and (4).

The degrees of the symmetric polynomials P(z) and Q(z) defined in (35) and (36) are $2m_p$ and $2m_Q$, respectively, where

$$M_p=\begin{cases} n/2 & n\ even\\ (n-1)/2 & n\ odd\end{cases}$$

and

$$M_Q=\begin{cases} (n-2)/2 & n\ even\\ (n-1)/2 & n\ odd\end{cases}$$

For <u>example</u>, if the degree of the predictor polynomial is ten (n = 10), then $m_p = 5$ and $m_Q = 4$. The polynomial $A_{10}(z)$ has only $n - 1 = m_p + m_Q = 9$ immittance spectrum frequencies. This is the most apparent difference to the line spectrum frequency representation. Note also, that immittance spectrum frequencies cannot generally be solved from line spectrum frequencies.

The relations between the coefficients of $A_n(z)$ and the coefficients of the polynomials P(z) and Q(z) are obtained straightforwardly from the equations

$$(38)\qquad P(z) = 1 + p_1z^{-1} + p_2z^{-2} + \ldots + p_2z^{-2mp+2} + p_1z^{-2mp+1} + z^{-2mp},$$

$$(39)\qquad Q(z) = 1 + q_1z^{-1} + q_2z^{-2} + \ldots + q_2z^{-2mp+2} + q_1z^{-2mp+1} + z^{-2mp},$$

and equations (31)-(34). Equations (38) and (39) can be written to the form (25) and (26) by proceeding as described above. Hence, the immittance spectrum frequencies ω_1, ω_2, ..., ω_{n-1} can be solved in accordance with this invention by the techniques summarized in the algorithm (29).

Although described above in the context of an audio encoder for use in a radiotelephone, it should be realized that the teachings of this invention are not limited for use in only this one important application. For example, and referring again to Fig. 1, the audio encoder can be used in a PC or workstation connected to a network. The communication channel 30 may then be a wired network (e.g., Internet).

Thus, the invention has been particularly shown and described with respect to preferred embodiments thereof, it will be understood by those skilled in the art that changes in form and details may be made therein without departing from the scope and spirit of the invention.

Claims

1. A method for determining line spectrum frequencies of a linear predictive coder (LPC) filter that is expressed as symmetric and antisymmetric polynomials, the zeros of which determine the line spectrum frequencies of the LPC filter, comprising the steps of:

   expressing the polynomials using explicit forms of Chebyshev polynomials;

   interatively solving a zero of a first of the polynomials using a zero of the other one of the polynomials; and

   successively eliminating zeros from the polynomials by polynomial division so as to determine the line spectrum frequencies.

2. A method as set forth in claim 1, and further comprising a step of transmitting a LPC coded signal to a communication channel.

3. A method as set forth in claim 2, wherein the step of transmitting transmits the LPC coded signal to a radio communication channel.

4. A method as set forth in any preceding claim, wherein the zero of the other one of the polynomials is used as an initial value in iteratively solving the zero of the first one of the polynomials.

5. A mobile station capable of wireless communications over a communication channel, said mobile station comprising a speech transducer for outputting a speech signal, and further comprising:

   a linear predictive coder (LPC) having an input coupled to the speech signal and an output coupled to the communication channel; said LPC comprising,

   a LPC filter having a first input coupled to the speech signal and an output;

   a LPC analysis block having an input coupled to the speech signal and an output for generating LPC coefficients for said LPC filter; and

   a transform block having an input coupled to said output of said LPC analysis block for transforming said LPC coefficients into a line spectrum frequency (LSF) representation thereof, said transform block having an output coupled to a second input of said LPC filter; wherein

   said LPC filter comprises symmetric and antisymmetric auxiliary polynomials, the zeros of which determine the line spectrum frequencies of the LPC filter, and wherein

   said transform block includes first means for expressing the auxiliary polynomials using explicit forms of Chebyshev polynomials;

   second means for iteratively solving a zero of a first of the polynomials using a zero of the other one of the polynomials; and

   said transform block further comprises third means for successively eliminating zeros from the polynomials by polynomial division so as to determine the line spectrum frequencies.

6. A mobile station as set forth in claim 5, and further comprising a quantizer block and an interpolator block coupled in series between said output of said transform block and said second input of said LPC analysis filter.

7. A mobile station as set forth in claim 5 or 6, wherein said third means is responsive to a degree of a given one of the polynomials being two, for solving the last zero analytically.

8. A method for determining the immittance spectrum frequencies of a Linear Predictive Coder (LPC) filter expressed as symmetric and antisymmetric polynomials, the zeros of which determine the immittance spectrum frequencies, comprising the steps of:

   expressing the polynomials using explicit forms of Chebyshev polynomials;

   iteratively solving a zero of a first one of the polynomials using a zero of the other one of the polynomials; and

   successively eliminating zeros from the polynomials by polynomial division so as to determine the immittance spectrum frequencies.

9. A method as set forth in claim 8, and further comprising a step of transmitting a LPC coded signal to a communication channel.

10. A method as set forth in claim 9, wherein the step of transmitting transmits the LPC coded signal to a radio communication channel.

11. A method as set forth in any of claims 8 to 10, wherein the zero of the other one of the polynomials is used as an initial value in iteratively solving the zero of the first one of the polynomials.

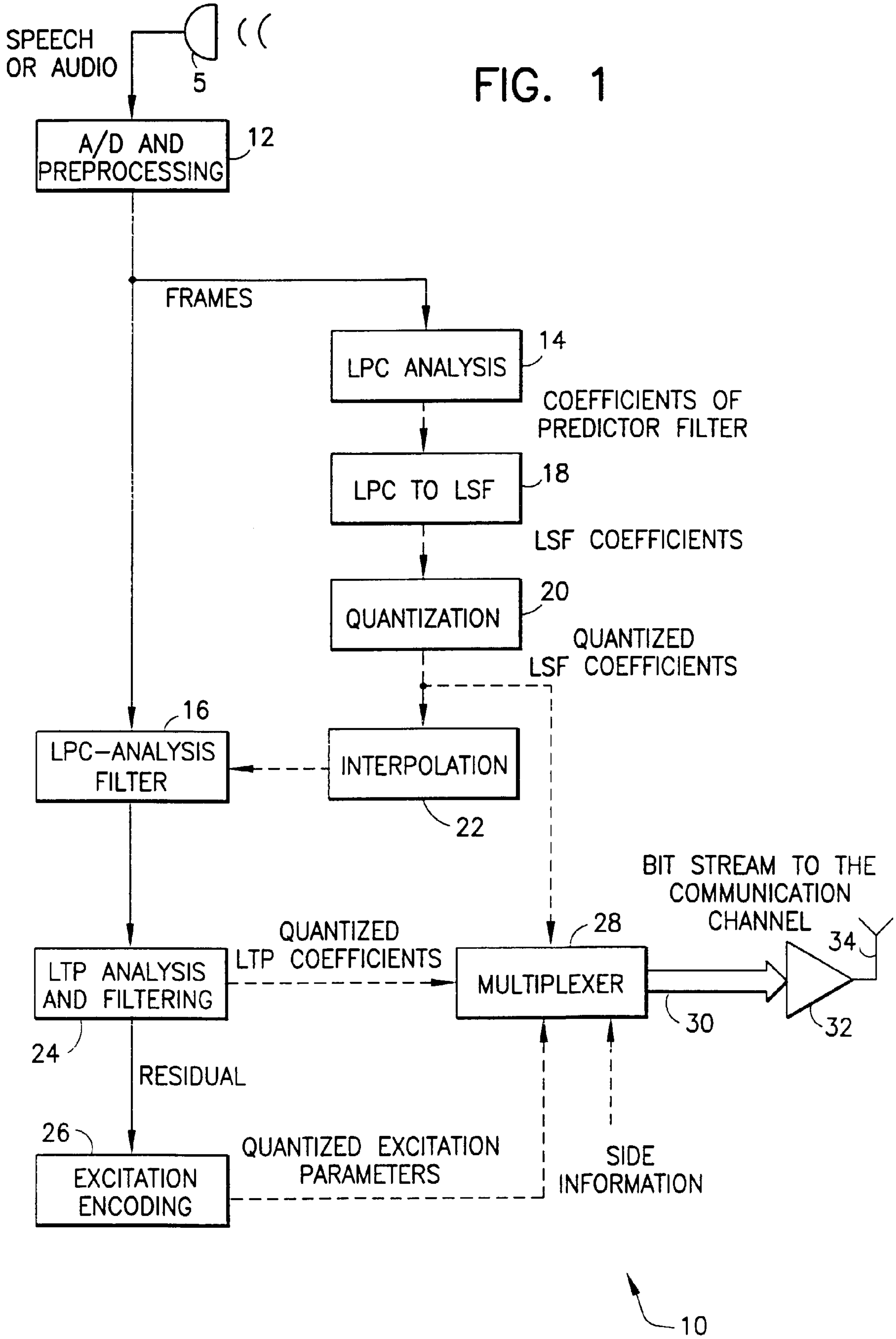
SPEECH
OR AUDIO
5
A/D AND
PREPROCESSING
12
FRAMES
LPC ANALYSIS
14
COEFFICIENTS OF
PREDICTOR FILTER
LPC TO LSF
18
LSF COEFFICIENTS
QUANTIZATION
20
QUANTIZED
LSF COEFFICIENTS
16
LPC–ANALYSIS
FILTER
INTERPOLATION
22
BIT STREAM TO THE
COMMUNICATION
CHANNEL
QUANTIZED
LTP COEFFICIENTS
28
34
LTP ANALYSIS
AND FILTERING
MULTIPLEXER
30
32
24
RESIDUAL
26
QUANTIZED EXCITATION
PARAMETERS
SIDE
INFORMATION
EXCITATION
ENCODING
10

FIG. 1

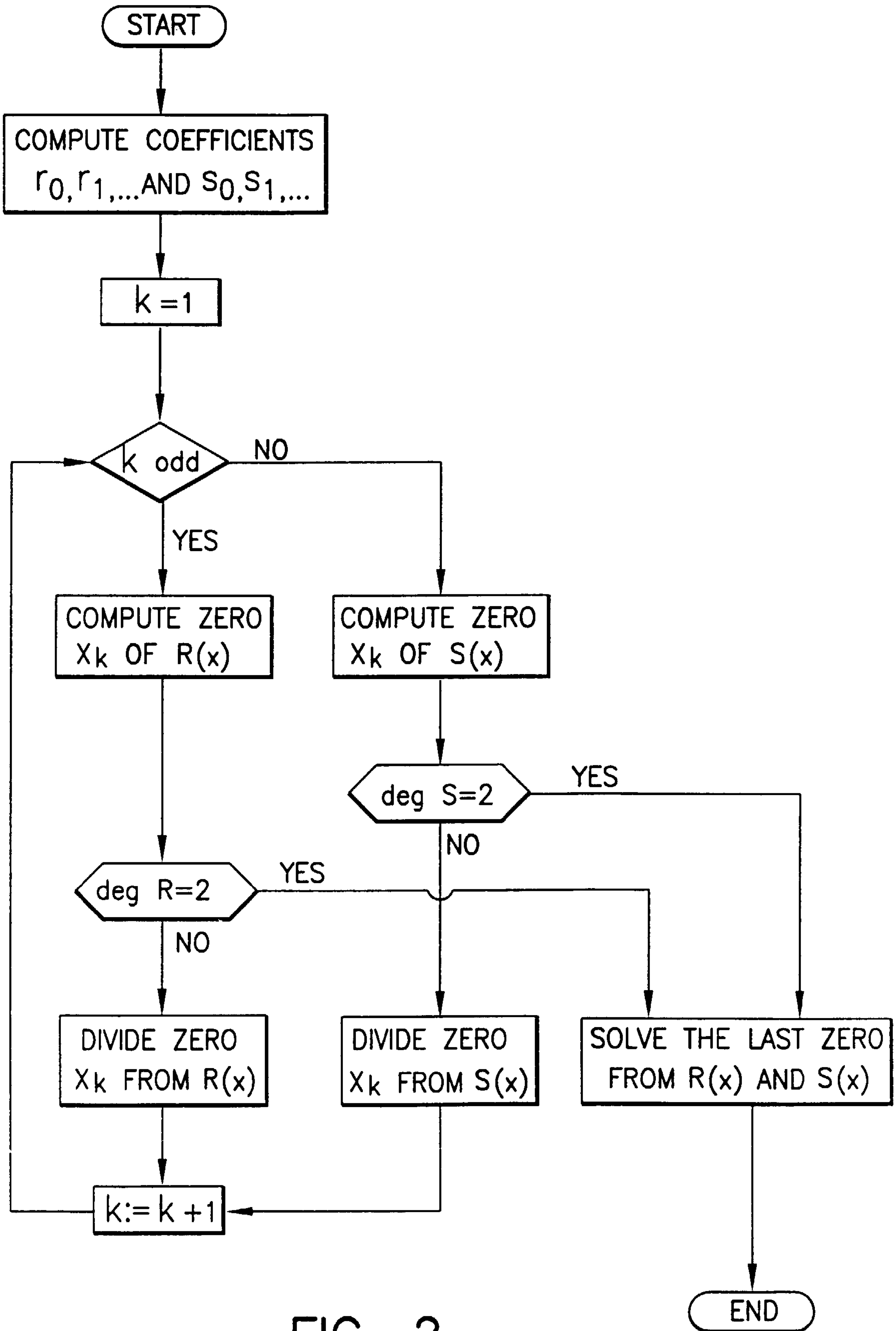
START
COMPUTE COEFFICIENTS
$r_0, r_1, \ldots$ AND $s_0, s_1, \ldots$
$k = 1$
k odd
NO
YES
COMPUTE ZERO
X_k OF R(x)
COMPUTE ZERO
X_k OF S(x)
deg S=2
YES
NO
deg R=2
YES
NO
DIVIDE ZERO
X_k FROM R(x)
DIVIDE ZERO
X_k FROM S(x)
SOLVE THE LAST ZERO
FROM R(x) AND S(x)
$k := k + 1$
END

FIG. 2

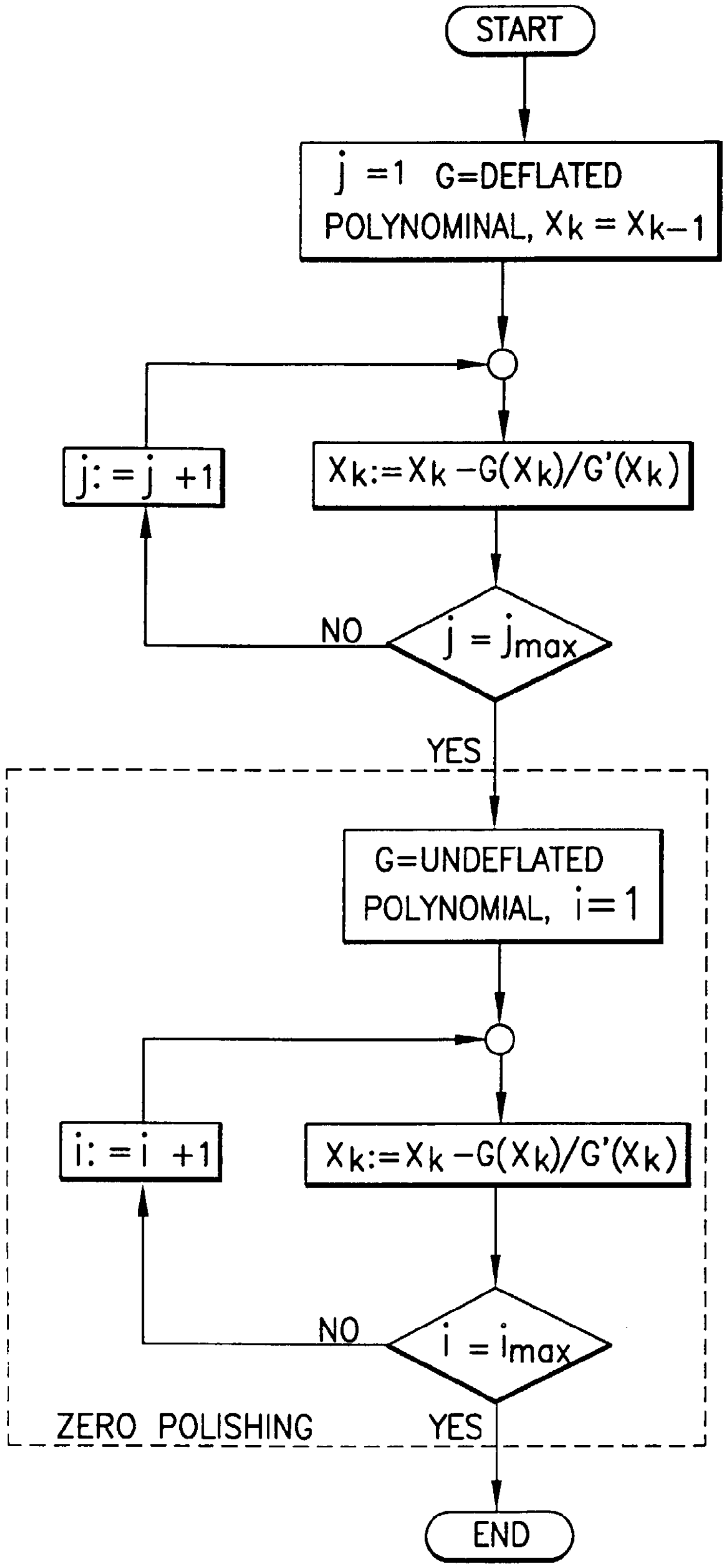
START
j =1 G=DEFLATED POLYNOMINAL, Xk = Xk−1
j: = j +1
Xk:=Xk −G(Xk)/G'(Xk)
NO
j = jmax
YES
G=UNDEFLATED POLYNOMIAL, i=1
i: = i +1
Xk:=Xk −G(Xk)/G'(Xk)
NO
i = imax
ZERO POLISHING
YES
END

FIG. 3

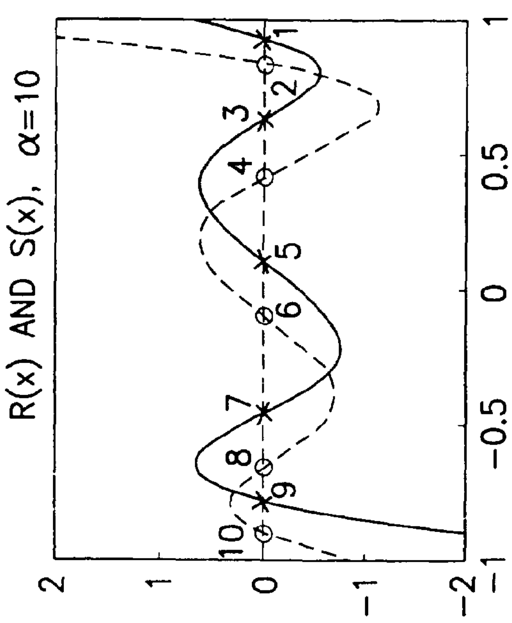

FIG. 4A

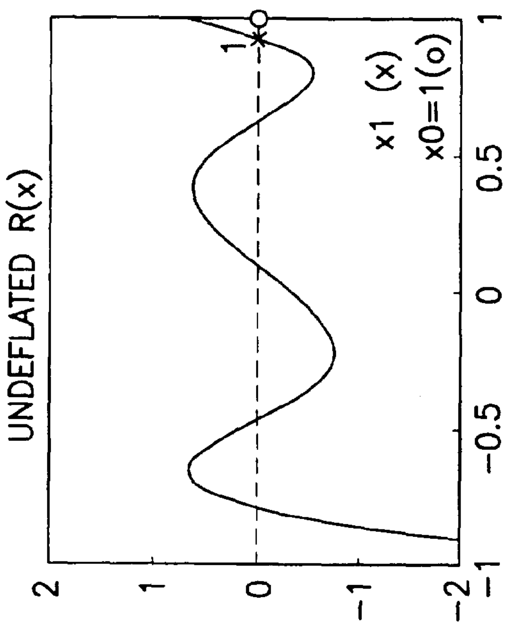

FIG. 4B

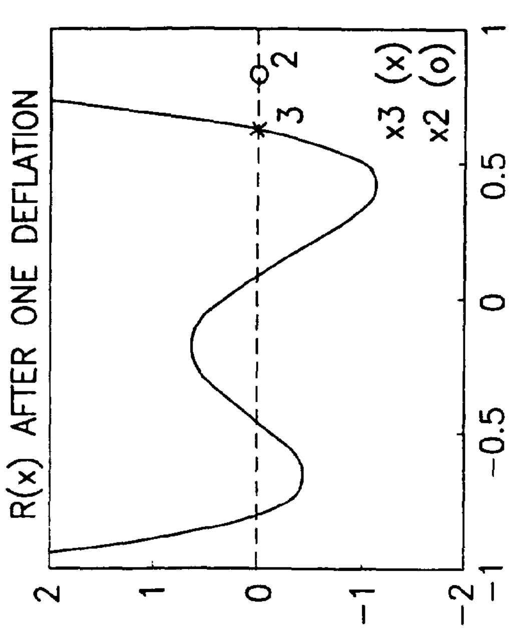

FIG. 4C

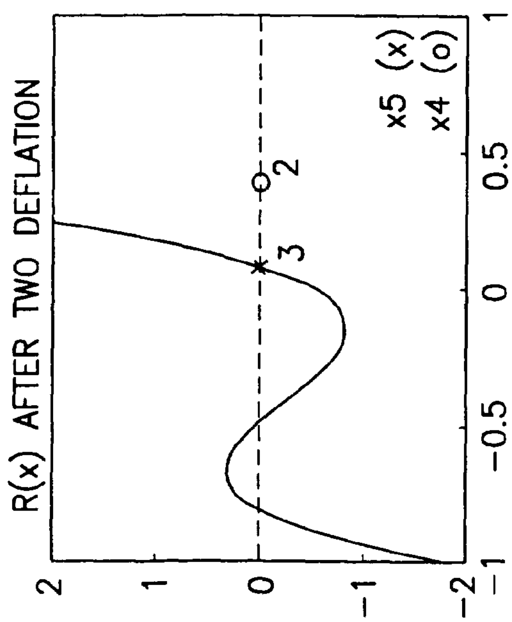

FIG. 4D