Consequently, we can replace (20) by

$$\frac{\partial y^*(n)}{\partial a_{Rk}(n)} = y^*(n-k) + \sum_{m=1}^{N-1} a_m(n) \frac{\partial y^*(n-m)}{\partial a_{Rk}(n-m)} \tag{22a}$$

$$\frac{\partial y^*(n)}{\partial a_{Ik}(n)} = jy^*(n-k) + \sum_{m=1}^{N-1} a_m(n) \frac{\partial y^*(n-m)}{\partial a_{Ik}(n-m)}. \tag{22b}$$

Notice that these equations are recursive in the partial derivatives since terms on the right-hand side correspond to delayed versions of the left-hand side. From (13), the component of η corresponding to a_k (which we denote η_{a_k}) is

$$\eta_{a_k}(n) = \frac{1}{2}\left(\frac{\partial y^*(n)}{\partial a_{Rk}(n)} + j\frac{\partial y^*(n)}{\partial a_{Ik}(n)}\right).$$

Substituting (22), we obtain the recursion

$$\begin{aligned}\eta_{a_k}(n) &= \sum_{m=1}^{N-1} a_m(n) \frac{1}{2}\left(\frac{\partial y^*(n-m)}{\partial a_{Rk}(n-m)} + j\frac{\partial y^*(n-m)}{\partial a_{Ik}(n-m)}\right) \\ &= \sum_{m=1}^{N-1} a_m(n)\, \eta_{a_k}(n-m).\end{aligned} \tag{23}$$

Recursive expressions for the other components of η and ψ, denoted by η_{b_k}, ψ_{a_k}, and ψ_{b_k}, are obtained in a similar way; they are

$$\begin{aligned}\eta_{b_k}(n) &= \frac{1}{2}\left(\frac{\partial y^*(n)}{\partial b_{Rk}(n)} + j\frac{\partial y^*(n)}{\partial b_{Ik}(n)}\right) \\ &= \sum_{m=1}^{N-1} a_m(n)\, \eta_{b_k}(n-m)\end{aligned} \tag{24}$$

$$\begin{aligned}\psi_{a_k}(n) &= \frac{1}{2}\left(\frac{\partial y(n)}{\partial a_{Rk}(n)} + j\frac{\partial y(n)}{\partial a_{Ik}(n)}\right) \\ &= y(n-k) + \sum_{m=1}^{N-1} a_m^*(n)\, \psi_{a_k}(n-m)\end{aligned} \tag{25}$$

$$\begin{aligned}\psi_{b_k}(n) &= \frac{1}{2}\left(\frac{\partial y(n)}{\partial b_{Rk}(n)} + j\frac{\partial y(n)}{\partial b_{Ik}(n)}\right) \\ &= x(n-k) + \sum_{m=1}^{N-1} a_m^*(n)\, \psi_{b_k}(n-m).\end{aligned} \tag{26}$$

Observe that (23) and (24) do not depend upon the input x or the output y. Consequently, they correspond to *unforced* difference equations. Since η_{a_k} and η_{b_k} are initially zero, they will remain zero for all n. (If they were initially nonzero, then they would decay to zero because we have assumed that (1) is stable.) We will therefore assume that η_{a_k} and η_{b_k} are precisely zero for all n. This leads to the result of (15).

From the forced difference equations of (25) and (26), we can compactly write ψ as

$$\psi(n) = \left(\frac{1}{1 - A^*(n, z^{-1})}\right)\phi(n), \tag{27}$$

where A^* and ϕ are given by (2) and (3), respectively. The complete Gauss–Newton (GN) algorithm is finally given by (18) and (19) where ψ is recursively computed by (27).

Observe, however, that computation of the components of ψ contributes a significant amount of complexity to the GN algorithm. Also, a large amount of storage is needed for past values of ψ_{a_k} and ψ_{b_k}. Fortunately, the approximation of (21) allows considerable simplification in the calculation of ψ [5], as we now show. Let us first define

$$y^f(n-1) = \psi_{a_1}(n) \tag{28a}$$

$$x^f(n) = \psi_{b_0}(n). \tag{28b}$$

Then (21) permits us to substitute

$$\psi_{a_k}(n) \approx y^f(n-k), \qquad k = 2, \cdots, N-1$$

$$\psi_{b_k}(n) \approx x^f(n-k), \qquad k = 1, \cdots, M-1.$$

We can therefore replace (27) with

$$\psi(n) = [y^f(n-1) \cdots y^f(n-N+1) \; x^f(n) \cdots x^f(n-M+1)]^T \tag{29}$$

where, from (25), (26), and (28), we have

$$y^f(n-1) = \left(\frac{1}{1 - A^*(n, z^{-1})}\right) y(n-1) \tag{30a}$$

$$x^f(n) = \left(\frac{1}{1 - A^*(n, z^{-1})}\right) x(n). \tag{30b}$$

The superscript f indicates that y^f and x^f correspond to filtered versions of y and x, respectively. The resulting simplified GN algorithm is thus (18) and (19) coupled with (29) and (30), which clearly requires less computation and storage than that of (27).

II. Conclusion

We have presented a Gauss–Newton (GN) algorithm for adaptive IIR filters with complex coefficients. Although the gradient estimate appears to have two separate components [see (13) and (14)], it was shown that one component is essentially zero. Consequently, the complex-coefficient GN algorithm is a straightforward generalization of the real-coefficient GN algorithm. The algorithm is stable provided the pole polynomial of the adaptive filter is kept minimum phase after each coefficient update. This requires the same stability-checking and pole-projection methods used by the real-coefficient GN algorithm [5].

References

[1] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification.* Cambridge, MA: MIT Press, 1983.

[2] J. J. Shynk and R. P. Gooch, "Frequency-domain adaptive pole-zero filtering," *Proc. IEEE*, vol. 73, pp. 1526–1528, Oct. 1985.

[3] B. Widrow, J. McCool, and M. Ball, "The complex LMS algorithm," *Proc. IEEE*, vol. 63, pp. 719–720, Apr. 1975.

[4] S. A. White, "An adaptive recursive digital filter," in *Proc. 9th Asilomar Conf. Circuits, Syst., Comput.*, Pacific Grove, CA, Nov. 1975, pp. 21–25.

[5] C. R. Johnson, Jr., "Adaptive IIR filtering: Current results and open issues," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 237–250, Mar. 1984.

The Poles of Symmetric Linear Prediction Models Lie on the Unit Circle

PETRE STOICA AND ARYE NEHORAI

***Abstract*—Symmetric linear prediction models have their poles on the unit circle. A simple proof of this result is presented.**

I. Introduction

Unconstrained linear prediction models have their poles located inside the unit circle (see, e.g., [1]–[3]). If the coefficients of the

Manuscript received November 15, 1985; revised February 14, 1986. The work of A. Nehorai was supported in part by the National Science Foundation under Grant DCI-8604351.

P. Stoica is with the Facultatea de Automatica, Inst. Politehn Bucuresti, Splaiul Independentei 313, Sector 7, R-77 206 Bucharest, Romania.

A. Nehorai is with the Department of Electrical Engineering, Yale University, New Haven, CT 06520.

IEEE Log Number 8609043.

model are constrained in some way, then this property no longer necessarily holds. An interesting class of constrained linear prediction models is that of symmetric models (see below). The poles of symmetric linear prediction models lie on the unit circle. In this correspondence, a simple proof is given of this result which was introduced (among other results) in [4] and independently in [5].

II. Main Result

Theorem: Let the symmetric polynomial[1]

$$A(z) = 1 + a_1 z + \cdots + a_{m-1} z^{m-1} + a_m z^m + a_{m-1} z^{m+1} + \cdots + a_1 z^{2m-1} + z^{2m}$$

satisfy

$$E[A(q^{-1})\, y(t)]^2 = \min \tag{1}$$

where $y(t)$ is a stationary process, q^{-1} denotes the unit delay operator, and E is the expectation operator. Then

$$A(z) \neq 0 \quad \text{for} \quad |z| \neq 1. \tag{2}$$

In other words, $A(z)$ has all its roots on the unit circle.

Proof: Let $\phi(\omega)$ denote the spectral density of $y(t)$. Since

$$E[A(q^{-1})\, y(t)]^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |A(e^{i\omega})|^2 \phi(\omega)\, d\omega = \min \tag{3}$$

where the minimum is over $\{a_i\}_{i=1}^{m}$, it follows that the inequality

$$|\tilde{A}(e^{i\omega})| < |A(e^{i\omega})|, \tag{4}$$

where $\tilde{A}(z)$ is any other symmetric monic polynomial of degree $2m$, cannot hold for all $\omega \in (-\pi, \pi]$. A general factor of $A(z)$ that may give roots which do not lie on the unit circle is the following:

$$1 + az + z^2$$

where a is either real or complex. If a is not real, then $1 + a^*z + z^2$, where a^* is the complex conjugate of a, also is a factor of $A(z)$. Since

$$\begin{aligned} |1 + ae^{i\omega} + e^{2i\omega}| &= |e^{-i\omega} + a + e^{i\omega}| \\ &= |e^{-i\omega} + a^* + e^{i\omega}| \end{aligned} \tag{5}$$

it follows that the inequality (4) holds if there exists some $\tilde{a}$ such that

$$\begin{aligned} |e^{-i\omega} + \tilde{a} + e^{i\omega}| &< |e^{-i\omega} + a + e^{i\omega}| \\ &= |a + 2\cos\omega| \quad \text{for} \quad \omega \in (-\pi, \pi]. \end{aligned} \tag{6}$$

Let us assume that a is not real. Then $\tilde{a} = \text{Real}\,\{a\}$ satisfies (6). Since (6) implies (4) which is a contradiction to (3), it follows that a must be real.

Next let us assume that $a > 2$, say, $a = 2 + \alpha$ where $\alpha > 0$. Then, we get

$$\begin{aligned} |a + 2\cos\omega|^2 &= |2 + \alpha + 2\cos\omega|^2 \\ &= \alpha^2 + 2\alpha(2 + 2\cos\omega) + (2 + 2\cos\omega)^2 \\ &> |2 + 2\cos\omega|^2 \\ &= |e^{-i\omega} + 2 + e^{i\omega}|^2, \qquad \omega \in (-\pi, \pi]. \end{aligned}$$

Thus, (6) is satisfied for $\tilde{a} = 2$ and, therefore, a cannot be larger than 2. Finally, note that for $a < -2$, say, $a = -2 - \alpha$ with $\alpha > 0$, we get

$$\begin{aligned} |a + 2\cos\omega|^2 &= |2 + \alpha - 2\cos\omega|^2 \\ &= \alpha^2 + 2\alpha(2 - 2\cos\omega) + (2 - 2\cos\omega)^2 \\ &> |-2 + 2\cos\omega|^2 \\ &= |e^{-i\omega} - 2 + e^{i\omega}|^2, \qquad \omega \in (-\pi, \pi] \end{aligned}$$

which implies that (6) holds for $\tilde{a} = -2$. Therefore, a must belong to the interval $[-2, 2]$ which means that $1 + az + z^2$ has its roots on the unit circle at $e^{\pm i\omega_o}$, where $\omega_o = \arccos(-a/2)$. ■

Next we consider the case of practical interest where only a finite-length sample $\{y(1), \cdots, y(N)\}$ of the observed process is available. For the "standard" least-squares method

$$\sum_{t=n+1}^{N} [A(q^{-1})\, y(t)]^2 = \min \qquad n = 2m, \tag{7}$$

it is not necessarily true that $A(z)$ has all the roots on the unit circle. For example, let $n = 2$ and $N = 3$. Then (7) becomes

$$[y(3) + ay(2) + y(1)]^2 = \min$$

which gives (assuming that $y(2) \neq 0$)

$$a = -[y(3) + y(1)]/y(2).$$

This value of a does not necessarily lie in the interval $[-2, 2]$.

Let us consider the following modified least-squares method:

$$\sum_{t=1}^{N+n} [A(q^{-1})\, y(t)]^2 = \min \tag{8}$$

where the extended sequence $\{y(t)\}_{t=1-n}^{N+n}$, which is needed in (8), is obtained by padding the initial sequence with zeros in the following way (observe that a similar padding was implicitly done also in [4]):

$$y(t) = \begin{cases} 0 & t = 1 - n, 0 \\ y(t) & t = 1, N \\ 0 & t = N + 1, N + n \end{cases} \tag{9}$$

For N much larger than n, it may be expected that the estimates of $\{a\}$ in (7) and, respectively, (8) are quite close to one another. The estimate of (8) has the virtue that the corresponding polynomial $A(z)$ has all its roots on the unit circle. To see this, note that from Parseval's theorem

$$\sum_{t=1}^{N+n} [A(q^{-1})\, y(t)]^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \psi_N(\omega)\, d\omega \tag{10}$$

where $\psi_N(\omega)$ is the periodogram of $A(q^{-1})\, y(t)$

$$\psi_N(\omega) = \left| \sum_{t=1}^{N+n} A(q^{-1})\, y(t)\, e^{-i\omega t} \right|^2. \tag{11}$$

Due to the special form of padding with zeros used in (9), it can be readily verified that

$$\psi_N(\omega) = |A(e^{i\omega})|^2 \phi_N(\omega) \tag{12}$$

where $\phi_N(\omega)$ is the periodogram of the observed sequence

$$\phi_N(\omega) = \left| \sum_{t=1}^{N} y(t)\, e^{-i\omega t} \right|^2. \tag{13}$$

Inserting (12) into (10) we get

$$\sum_{t=1}^{N+n} [A(q^{-1})\, y(t)]^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |A(e^{i\omega})|^2 \phi_N(\omega)\, d\omega = \min. \tag{14}$$

The proof of the theorem applies mutatis mutandis to (14) concluding that $A(z) \neq 0$ for $|z| \neq 1$ [compare (3) and (14)].

III. Conclusion

The above result is relevant to fitting linear prediction models to sinusoids-in-noise processes. At high signal-to-noise ratios (SNR's), such a process may be well approximated by a symmetric linear prediction model with all its poles lying on the unit circle. The angular positions of these poles are equal to the sinusoidal frequencies. As shown in [4] and [5], and in this correspondence, when fitting a symmetric linear prediction model to a sequence of data, the complicated constraint that the poles of the model be on

[1] For the sake of conciseness, we consider symmetric polynomials of even order which occur more frequently in applications. The case of odd-order symmetric polynomials can be treated similarly.

the unit circle is automatically satisfied under weak conditions. This property considerably simplifies the task of fitting linear prediction models with all the poles on the unit circle for a sinusoid-in-noise process. At medium or low SNR's, this approach to sinusoidal frequency estimation may, however, give severely biased estimates, and other approaches such as those of [6]–[8] should be used in such cases.

In [9] we discuss some computational aspects of the symmetric linear prediction models, including an elementary algebraic proof of the corresponding fast computation procedure designed in [4].

References

[1] U. Grenander and G. Szegö, *Toeplitz Forms and Their Applications.* Berkeley, CA: University of California Press, 1958.

[2] J. Makhoul, "Linear prediction—A tutorial review," *Proc. IEEE*, vol. 63, pp. 564–580, 1975.

[3] P. Stoica and A. Nehorai, "On stability and root location of linear prediction models," submitted for publication.

[4] D. M. Goodman and E. K. Miller, "A note on minimizing the prediction error when the zeros are restricted to the unit circle," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 502–505, June 1982.

[5] G. Cybenko, "Fast approximation of dominant harmonics," *SIAM J. Sci. Stat. Comput.*, vol. 5, pp. 317–331, June 1984.

[6] J. A. Cadzow, "High performance spectral estimation—A new ARMA method," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 524–529, Oct. 1980.

[7] D. W. Tufts and R. Kumaresan, "Estimation of frequencies of multiple sinusoids: Making linear prediction perform like maximum likelihood," *Proc. IEEE*, vol. 70, pp. 975–989, Sept. 1982.

[8] A Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 983–996, Aug. 1985.

[9] P. Stoica and A. Nehorai, "The poles of symmetric linear prediction models lie on the unit circle: A new proof and computational aspects," Center Syst. Sci., Dep. Elec. Eng., Yale Univ., New Haven, CT, Rep. 8606, July 1986.

On Residue Number System Decoding

RUDOLF E. THUN

Abstract—**The use of a residue number system (RNS) in digital systems and especially filter designs is facilitated by efficient algorithms for the conversion from RNS to binary numbers. The conversion is generally based on the Chinese remainder theorem or the mixed radix conversion. This correspondence describes another conversion algorithm which employs the direct pairwise solution of the Diophantine equations defining the number in the given moduli set. The algorithm provides a high degree of parallel computation.**

A residue number system (RNS) is based on an ordered n-tuple set of moduli $[m_1, m_2, \cdots, m_n]$ called the basic vector V_B of an RNS [1], [2]. The m_i's are integers and pairwise relative prime. Any integer $X \in |0, M_n)$ with $M_n = \Pi_{i=1}^{n} m_i$ has a unique residue presentation defined by

$$\{|X|m_1, |X|m_2, \cdots, |X|m_n\} \quad \text{where} \quad |X|m_1 = X \text{ modulo } m_i$$

can more simply be denoted as x_i. The x_i's satisfy the condition $0 \le x_i \le m_i$.

The conversion from RNS numbers to natural integers is commonly performed by use of the Chinese remainder theorem (CRT) or the mixed radix conversion (MRC) [3], [4]. The decoding according to the CRT is given by

$$X = \left| \sum_{i=1}^{n} |x_i \hat{m}_i^{-1}|_{m_i} \cdot \hat{m}_i \right|_{M_n} \tag{1}$$

where

$$\hat{m}_i = M_n m_i^{-1}; \quad \hat{m}_i^{-1} = \left|\frac{1}{\hat{m}_i}\right|_{m_i}; \quad \left| |\hat{m}_i|_{m_i} \cdot \left|\frac{1}{\hat{m}_i}\right|_{m_i} \right|_{m_i} = 1$$

and x_i is as before the residue modulo m_i.

The formula for the MRC can be expressed as

$$X = \sum_{i=0}^{n-1} a_i \prod_{k=0}^{i} m_k \tag{2}$$

where $0 \le a_i < m_{(i+1)}$ for all $i > 0$ and $m_o = 1$. The coefficients a_i are generally computed with a recursive algorithm employing intermediate variables S_{ij} with $x = S_{oo} \equiv a_o$; $x_l = S_{o(l-1)}$ for $l = 2, 3, \cdots, n$; and $S_{ii} \equiv a_i$ for $i = 0, 1, \cdots, (n-1)$. The recursive formula is then

$$S_{ij} = \left| (S_{(i-1)j} - S_{(i-1)(i-1)}) \cdot \left|\frac{1}{m_i}\right|_{m(j+1)} \right|_{m(j+1)} \tag{3}$$

where $j = i, (i+1), \cdots, (n-1)$ for each subscript i.

This correspondence describes an alternative method for the conversion of an RNS number X into natural integers which is based on the solution of the following Diophantine equations without invoking Fermat's theorem:

$$X = y_i m_i + x_i; \qquad i = 1, 2, \cdots, n \tag{4}$$

where all the y_i, m_i, and x_i are either zero or positive integers. Again, $X \in |0, M_n)$ with $M_n = \Pi_{i=1}^{n} m_i$. Equation (4) contains a set of $P = \Sigma_{b=1}^{n-1} b$ equations

$$y_j m_j + x_j = y_k m_k + x_k; \quad j \neq k \tag{5}$$

of which a subset of minimum size $q = (n-1)$ will be shown to solve for X.

The smallest positive integer solution of each of the equations (5) can be readily obtained by iteration or table lookup, since the integers of each equation can be kept small even for a large dynamic range (large M_n). The values of y_j and y_k for this smallest positive integer solution shall be designated $\tilde{y}_{jk}$ and $\tilde{y}_{kj}$.

Then

$$\tilde{y}_{jk} = \frac{m_k \tilde{y}_{kj} + x_k - x_j}{m_j}. \tag{6}$$

Lemma 1: All possible positive solutions of y_j as a function of y_k are obtained from $\tilde{y}_{jk}$ by adding to $\tilde{y}_{jk}$ multiples of m_k

$$y_j = f(y_k) = \tilde{y}_{jk} + c_{jk} m_k \tag{7}$$

where the coefficients $c_{jk} = 0, 1, \cdots, (L < M_n/m_j m_k)$.

Proof: since $X \in |0, M_n)$ and, according to (5), y_j is monotonously increasing with y_k, y_j, and y_k can be replaced in $(y_j + \Delta y_j)$ $m_j + x_j = (y_k + \Delta y_k) m_k + x_k$ with $\tilde{y}_{jk}$ and $\tilde{y}_{kj}$ without loss of generality. Furthermore, $\Delta y_{jk} = (\Delta y_{kj}/m_j)/m_k$. By definition $\Delta y_{kj}/m_j = c_{jk}$ = integer. The inequality $c_{jk} < M_n/m_j\, m_k$ follows from (4) and $X \in |0, M_n)$. Equations (7) have the same form as (4) and yield a minimum subset of $(n-2)$ equations

$$\tilde{c}_{jk} = \frac{m_l \tilde{c}_{jl} + \tilde{y}_{jl} - y_{jk}}{m_k} \tag{8}$$

and

$$c_{jk} = \tilde{c}_{jk} + c_{jkl} m_l; \qquad c_{jkl} = \text{positive integer}. \tag{9}$$

Each subsequent iteration of these equations reduces the number of unknowns by one, and a total of $(n-1)$ iterations is required to obtain all coefficients needed to solve for X by substitution. To combine (6) and (8) as well as (7) and (9), auxiliary variables t_{ij}

Manuscript received February 28, 1985; revised February 20, 1986.
The author is with the Raytheon Company, Lexington, MA 02173.
IEEE Log Number 8609041.

0096-3518/86/1000-1346$01.00