Published in IET Signal Processing
Received on 1st August 2008
Revised on 27th January 2009
doi: 10.1049/iet-spr.2008.0141

ISSN 1751-9675

Gradient-adaptive algorithms for minimum-phase-all-pass decomposition of a finite impulse response system

M.F. Flanagan[1] *M. McLaughlin*[2] *A.D. Fagan*[1]

[1]*Department of Electronic and Electrical Engineering, University College Dublin, Belfield, Dublin 4, Ireland*
[2]*DecaWave Ltd., Digital Depot, Thomas St., Dublin 8, Ireland*
E-mail: mark.flanagan@ieee.org

Abstract: Adaptive algorithms which perform minimum-phase–all-pass (MP–AP) decomposition of a finite impulse response system are proposed. The first algorithm models the MP component of the system as a lattice filter cascaded with a gain stage. The algorithm has a low misconvergence probability, and is capable of detecting misconvergence during or after adaptation. Two further algorithms are proposed based on the theory of the bicepstrum. One adaptively solves a finite linear system of equations and the other an augmented nonlinear system. The first has an error sensitive to the proximity of the system zeros to the unit circle, whereas the second, although more computationally intensive, may approach the exact MP–AP decomposition given a sufficient number of iterations. These real-time MP–AP decomposition algorithms have applications in the stabilisation of compound precoding, which is a pre-equalisation technique included as an option in the V.92 high-speed modem standard.

1 Introduction

It is well known (see e.g. [1]) that any finite impulse response (FIR) system $G(z)$ of order p may be uniquely decomposed into the cascade of a minimum-phase (MP) component $W(z)$ (FIR, of order p) and an all-pass (AP) component. This decomposition may, of course, be effected by invoking standard rootfinding algorithms; in this paper however, adaptive algorithms are proposed that may effect this decomposition in a real-time digital signal processing (DSP) application. To the authors' knowledge such algorithms have not been reported previously in the literature. Our motivation is the fact that in compound precoding, a power-efficient method for combining decision-feedback equalisation (DFE) with trellis coding in a modem transmitter, a real-time MP–AP decomposition of the precoder feedforward filter is necessary for implementation stability [2, 3]. Compound precoding is an option in the V.92 modem standard [4].

The lattice filter is a linear filter structure commonly used in linear prediction of signals (see e.g. [5]), and the least-squares lattice predictor has been successfully applied to problems in speech coding [6] and blind equalisation [7]. It may be shown [5] that every transversal implementation of a minimum-phase FIR filter has an equivalent implementation in the form of a lattice filter followed by a gain stage. The bijective mapping from one filter implementation to the other may be achieved via the Levinson–Durbin and inverse Levinson–Durbin recursions. We derive and evaluate a gradient-adaptive algorithm to achieve this decomposition where the MP component is implemented as a lattice filter followed by a gain stage, and updated accordingly. All stochastic updates are derived by substituting sample correlation for true correlation in the steepest descent updates, as in the ordinary least-mean-square (LMS) algorithm. The lattice structure has the advantage that upon convergence we may easily test whether the filter $W(z)$ given by the algorithm is actually minimum-phase. A necessary and sufficient condition for the filter to be minimum-phase is that all the reflection coefficients of the lattice filter have magnitude less than unity; therefore failure of this condition, and hence misconvergence of the algorithm, may be detected at any stage during adaptation. In [8] a gradient-adaptive lattice filter was proposed for linear prediction; here the cost function was taken to be the sum of the forward and backward prediction energies, which is specific to the linear prediction problem. The matching [in the minimum mean-squared error (MMSE) sense] of the lattice filter output to a desired output signal, which is proposed in this paper, is of more general application.

The bicepstrum is a signal representation with applications in blind deconvolution [9, 10] as well as signal time delay estimation and system identification [11]. In these applications, the algorithms usually involve estimation of the higher-order statistics of a (possibly noisy) data sequence. We propose two gradient-adaptive algorithms for performing MP–AP decomposition of an FIR system based on the theory of the bicepstrum. Our present work differs fundamentally from the previous works described, in that we target decomposition of a *known system* in the absence of noise. Thus, our problem lies in how to handle an infinitely large but exact system of linear equations, as opposed to problems of moment or cumulant estimation and noise suppression. The theory of the bicepstrum yields a relationship between the infinite set of differential cepstrum parameters and a certain finite set of third-order cumulants, in the form of an infinite system of linear equations, these cumulants being generated in a straightforward manner from the coefficients of the filter $G(z)$. If this system could be solved exactly, the exact MP–AP decomposition of $G(z)$ could be computed using simple recursion formulas involving only a small subset of the differential cepstrum parameters. The first algorithm solves the truncated linear system via steepest descent, and involves an approximation error dependent on the proximity of the filter zeros to the unit circle. The second algorithm, based on steepest descent solution of a non-linear augmented system, is more computationally intensive, but simulations show that it may converge to the exact MP–AP decomposition.

The paper is organised as follows. Section 2 describes the proposed gradient adaptive lattice filtering algorithm for MP–AP decomposition, and Section 3 describes the bicepstrum-based algorithms. Section 4 presents simulation results for all algorithms and Section 5 concludes the work.

2 MP–AP decomposition based on adaptive lattice filtering

2.1 Adaptive lattice filtering for desired response

First we address the general problem of updating the reflection coefficients of a lattice filter in order to obtain MMSE between the upper output signal and a desired response. Consider the circuit shown in Fig. 1. The

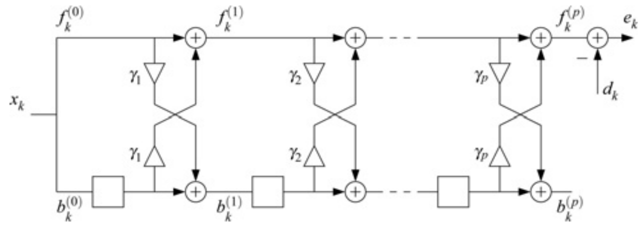

Figure 1 *Adaptive lattice filtering for desired response*

The reflection coefficients of the lattice filter are adapted to minimise the mean-squared error between the lattice filter output $f_k^{(p)}$ and the desired response d_k

equations governing the operation of the lattice filter are

$$\begin{aligned} f_k^{(m)} &= f_k^{(m-1)} + \gamma_m b_{k-1}^{(m-1)} \\ b_k^{(m)} &= \gamma_m f_k^{(m-1)} + b_{k-1}^{(m-1)} \end{aligned} \tag{1}$$

for $m = 1, 2, \ldots, p$. Given the input signal $\{x_k\}$ and the desired response $\{d_k\}$, the problem is to adapt the reflection coefficients γ_j, $j = 1, 2, \ldots, p$ in order to minimise the mean-squared error (MSE)

$$J(\{\gamma_j\}) = E\{e_k^2\}$$

where $e_k = f_k^{(p)} - d_k$. In order to derive a stochastic gradient algorithm, we need to estimate $\partial J/\partial\gamma_j$ for each γ_j. It is easy to see that for any $j \in \{1, 2, \ldots, p\}$, e_k can be written as $A + B\gamma_j$, where A, B are independent of γ_j. Therefore

$$\begin{aligned} \frac{\partial J}{\partial \gamma_j} &= \frac{\partial}{\partial \gamma_j} E\left\{\left(A + B\gamma_j\right)^2\right\} \\ &= 2E\{AB\} + 2\gamma_j E\{B^2\} \\ &= 2E\left\{\left(A + B\gamma_j\right)B\right\} \\ &= 2E\left\{e_k \frac{\partial e_k}{\partial \gamma_j}\right\} \\ &= 2E\left\{e_k \frac{\partial f_k^{(p)}}{\partial \gamma_j}\right\} \\ &= 2E\{e_k s_k^{(j)}\} \end{aligned}$$

where we define $s_k^{(j)} = \partial f_k^{(p)}/\partial\gamma_j$ for each $j \in \{1, 2, \ldots, p\}$.

Next, choose any $j \in \{1, 2, \ldots, p\}$. Observe that the sequences $f_k^{(j-1)}$ and $b_{k-1}^{(j-1)}$ are independent of γ_j. Therefore, putting $m = j$ in (1) and differentiating with respect to γ_j, we obtain

$$\frac{\partial f_k^{(j)}}{\partial \gamma_j} = b_{k-1}^{(j-1)}$$

doi: 10.1049/iet-spr.2008.0141

and

$$\frac{\partial b_k^{(j)}}{\partial \gamma_j} = f_k^{(j-1)}$$

Also, differentiating (1) with respect to γ_j for any $m \in \{j+1, j+2, \ldots, p\}$ yields

$$\frac{\partial f_k^{(m)}}{\partial \gamma_j} = \frac{\partial f_k^{(m-1)}}{\partial \gamma_j} + \gamma_m \frac{\partial b_{k-1}^{(m-1)}}{\partial \gamma_j}$$

$$\frac{\partial b_k^{(m)}}{\partial \gamma_j} = \gamma_m \frac{\partial f_k^{(m-1)}}{\partial \gamma_j} + \frac{\partial b_{k-1}^{(m-1)}}{\partial \gamma_j}$$

This means that each $s_k^{(j)}$ can be computed using a sublattice, as shown in Fig. 2. Each of these sublattices is attached to the main lattice of Fig. 1 at the appropriate point. The steepest descent update for the $\{\gamma_j\}$ is then simply

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} - \mu E\{e_k s_k^{(j)}\}$$

Replacing true correlation with sample correlation yields the adaptive algorithm

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} - \mu e_k s_k^{(j)}$$

Note that the sublattice of Fig. 2 which computes $s_k^{(j)}$ requires $p-j$ lattice stages. Therefore the entire lattice structure, including sublattices, requires $1+2+\ldots+p = p(p+1)/2$ lattice stages instead of the usual p stages.

2.2 Gradient-adaptive lattice-based MP–AP decomposition algorithm

The adaptive circuit for the MP–AP decomposition algorithm based on the gradient-adaptive lattice filter is shown in Fig. 3. The FIR filter $G(z)$ has order p and models the system to be decomposed. The MP component $W(z)$ is implemented as a lattice filter with coefficients $\{\gamma_j\}$ followed by a gain stage β. The IIR section composed of the filters $C(z)$ and $z^{-p}C(z^{-1}) - 1$ represents the all-pass component. All filters have order p in order to cater for the possibility of a maximum-phase $G(z)$. The coefficient c_p is set to unity and is not updated by the adaptive algorithm.

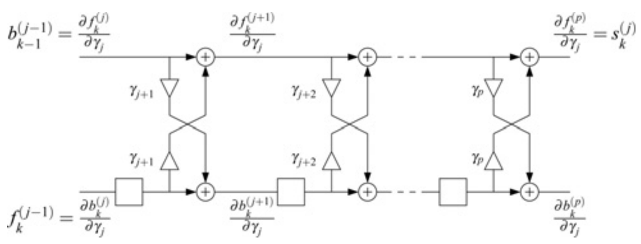

Figure 2 *Sublattice for computation of $s_k^{(j)}$, $j \in \{1, 2, \ldots, p\}$*

These sequences are used to update the reflection coefficients of the adaptive lattice filter of Fig. 1

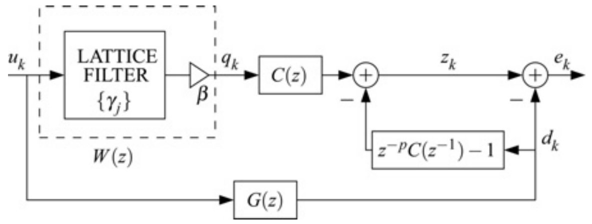

Figure 3 *Lattice-based adaptive circuit for MP–AP decomposition of an FIR system G(z)*

Note that the section comprising the feedforward and feedback filters with coefficients $\{c_j\}$ is guaranteed to be all-pass for any $\{c_j\}$ (and hence at any stage during adaptation) because of the reversal of the coefficients in the feedback section. The MSE is defined as $J(\{c_j\}, \{\gamma_j\}, \beta) = E\{e_k^2\}$. If the convolution of $\boldsymbol{w} = (w_0 \quad w_1 \quad \ldots \quad w_p)$ and $\boldsymbol{c} = (c_0 \quad c_1 \quad \ldots \quad c_p)$ is denoted $\boldsymbol{h} = (h_0 \quad h_1 \quad \ldots \quad h_{2p})$, then the MSE may be written as

$$J = \begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{c} \end{pmatrix}^{\mathrm{T}} \boldsymbol{Q} \begin{pmatrix} \boldsymbol{h} \\ \boldsymbol{c} \end{pmatrix}$$

where $\boldsymbol{Q}$ is a matrix having as entries elements of the autocorrelation sequence $R_u(i) = E\{u_k u_{k-i}\}$ and the cross-correlation sequence $R_{ud}(i) = E\{u_k d_{k-i}\}$. We assume that these two correlation sequences are stationary; hence so is the MSE. From Fig. 3, the estimation error at time step k may be written as

$$\begin{aligned} e_k &= z_k - d_k \\ &= \sum_{j=0}^{p} c_j q_{k-j} - \sum_{j=0}^{p-1} c_j d_{k-p+j} - d_k \\ &= \sum_{j=0}^{p-1} c_j (q_{k-j} - d_{k-p+j}) - \left(d_k - q_{k-p}\right) \end{aligned}$$

The sequences $\{q_k\}$ and $\{d_k\}$ are independent of $\{c_j\}$. Therefore the gradient of J with respect to $\{c_j\}$ is given by

$$\left(\frac{\partial J}{\partial c_j}\right)^{(k)} = 2E\{e_k(q_{k-j} - d_{k-p+j})\}$$

Replacing true correlation by sample correlation, we obtain the stochastic gradient update for $\{c_j\}$ as

$$c_j^{(k+1)} = c_j^{(k)} - \mu e_k (q_{k-j} - d_{k-p+j})$$

An equivalent circuit to that of Fig. 3 is shown in Fig. 4. Since $e_k = f_k^{(p)} - (a_k + d_k)$, the update for the reflection coefficients $\{\gamma_j\}$ is as presented in the discussion of Section 2.1 on adaptive lattice filtering for desired response, that is

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} - \mu e_k s_k^{(j)}, \quad j = 1, 2, \ldots, p$$

IET Signal Process., 2010, Vol. 4, Iss. 1, pp. 12–21
doi: 10.1049/iet-spr.2008.0141

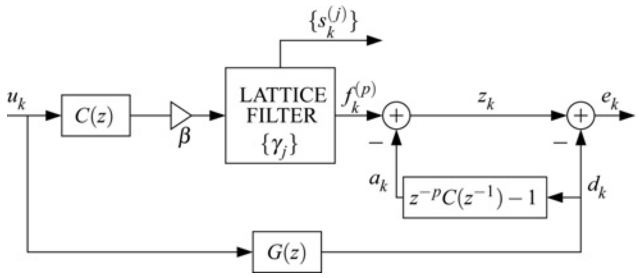

Figure 4 *Equivalent circuit to that of Fig. 3*

The lattice filter in this circuit is equipped with sublattices to compute $\{s_k^{(j)}\}$, $j = 1, 2, \ldots, p$

where $s_k^{(j)}$ is computed by the sublattices. To derive the update for the gain parameter β, note that if the order of the lattice filter and gain stage were switched in Fig. 4, the input to the gain stage would be $f_k^{(p)}/\beta$. Therefore the update for the gain parameter β may be obtained simply as the stochastic gradient update for a single-tap FIR filter, that is

$$\beta^{(k+1)} = \beta^{(k)} - \mu e_k \left(\frac{f_k^{(p)}}{\beta^{(k)}} \right)$$

Finally, note that if the circuit of Fig. 4 is implemented, the entire equivalent circuit of Fig. 3 need not be. The lattice filter (without sublattices) and gain stage, followed by a p-element delay line, are sufficient.

Note that the MSE (as a function of the coefficients to be updated) forms a multimodal surface. There are in general 2^p valid MMSE solutions, for each of which the lattice filter has the same magnitude response. This is in contrast to the usual application of lattice filtering in linear prediction, where the lattice filter must be minimum-phase. Only one of the 2^p MMSE solutions yields a minimum-phase $W(z)$; therefore convergence of the adaptive algorithm to the MP–AP decomposition is not guaranteed, but depends strongly on the initialisation of the algorithm. We advocate initialisation of the reflection coefficients to $\gamma_j^{(0)} = 0$ for $j = 1, 2, \ldots, p$; this filter is 'minimum-phase optimum' in the sense that its zeros are at maximum distance from the unit circle while maintaining minimum-phase.

3 MP–AP decomposition algorithms based on the bicepstrum

In this section, we describe two algorithms to perform MP–AP decomposition of the FIR system $G(z)$, based on the theory of the bicepstrum. The pertinent details of this theory are presented in Section 3.1. This theory yields a relationship between the infinite set of differential cepstrum parameters and a certain finite set of third-order cumulants, in the form of an infinite system of linear equations, these cumulants being generated in a straightforward manner from the coefficients of the filter $G(z)$. If this system could be solved exactly, the exact MP–AP decomposition of $G(z)$ could be computed using simple recursion formulas involving only a small subset of the differential cepstrum parameters. The first presented algorithm solves the truncated linear system via steepest descent, and the second performs steepest descent solution of this system augmented with non-linear equations. The proposed bicepstrum-based algorithms are generally more computationally intensive than the algorithm of Section 2.2 but do not suffer from the problem of misconvergence.

3.1 Theory of the bicepstrum

In this section we develop the relevant relationships between the minimum- and maximum-phase components of $G(z)$, the set of third-order cumulants associated with $G(z)$ and the differential cepstrum parameters. This development borrows from [10, 12].

For the sequence $\{g(n)\}$ with Z-transform $G(z)$, the complex cepstrum is defined as [1]

$$g_c(n) = Z^{-1}\{\log G(z)\}$$

and the differential cepstrum as

$$g_d(n) = Z^{-1}\left\{ \frac{\partial}{\partial z} \log G(z) \right\}$$

For the case at hand, the finite sequence of filter coefficients

$$g(0), g(1), \ldots, g(p)$$

has Z-transform

$$G(z) = \sum_{n=0}^{p} g(n) z^{-n}$$

This filter of order p has M zeros $\{a_i\}$ inside the unit circle and N zeros $\{c_i\}$ outside the unit circle, where $M + N = p$. $G(z)$ can thus be factored into minimum-phase and maximum-phase components as (here, to simplify the presentation, we adopt the notation $T^B(z) = z^{-t}T(z^{-1})$ where $T(z)$ is a filter of order t)

$$G(z) = AI(z)Q^B(z) \tag{2}$$

where both $I(z)$ and $Q(z)$ are causal, monic and minimum-phase, and have orders M and N, respectively. We use the notation

$$I(z) = \sum_{n=0}^{M} i(n) z^{-n}$$

and

$$Q(z) = \sum_{n=0}^{N} q(n) z^{-n}$$

$I(z)$ has zeros $\{a_i\}$ and $Q(z)$ has zeros $\{b_i\}$, and we define $c_i = 1/b_i$ for each i. We also define the

doi: 10.1049/iet-spr.2008.0141

anticausal filter $O(z) = Q^B(z)z^N$, so the factorisation of $G(z)$ becomes

$$G(z) = AI(z)O(z)z^{-N} \quad (3)$$

The quantities

$$A_k = \sum_{i=1}^{M} a_i^k \quad (4)$$

and

$$B_k = \sum_{i=1}^{N} b_i^k \quad (5)$$

for $k = 1, 2, \ldots$ are related to the differential cepstrum parameters as

$$i_d(n) = \begin{cases} A_{n-1}, & n \geq 2 \\ 0, & n \leq 1 \end{cases} \quad (6)$$

and

$$o_d(n) = \begin{cases} -B_{1-n}, & n \leq 0 \\ 0, & n \geq 1 \end{cases} \quad (7)$$

Therefore in what follows we shall also refer to $\{A_k\}$ and $\{B_k\}$ as the differential cepstrum parameters. Note that since both $I(z)$ and $Q(z)$ are minimum-phase, we have $|a_i| < 1$ and $|b_i| < 1$ for each i, and therefore the sequences $\{A_k\}$ and $\{B_k\}$ are both monotonically decreasing sequences that converge to zero as $k \to \infty$.

Next, if the system $G(z)$ is driven by a zero-mean input $x(n)$ satisfying

$$E\{x(i)x(j)x(k)\} = \begin{cases} 1, & i = j = k \\ 0, & \text{otherwise} \end{cases}$$

(The two-level i.i.d. sequence $\{x(n)|P(x(n) = 2^{2/3}) = 1/3, P(x(n) = -2^{-1/3}) = 2/3\}$, for example, satisfies this condition.)

Then the third-order cumulants of the output sequence $y(n)$ are

$$\begin{aligned} C(m, n) &= E\{y(i)y(i+n)y(i+m)\} \\ &= E\left\{\sum_{r\in\mathbb{Z}} g(r)x(i-r) \sum_{l\in\mathbb{Z}} g(l)x(i+n-l) \right. \\ &\quad \left. \times \sum_{s\in\mathbb{Z}} g(s)x(i+m-s)\right\} \\ &= \sum_{r\in\mathbb{Z}}\sum_{l\in\mathbb{Z}}\sum_{s\in\mathbb{Z}} g(r)g(l)g(s)E\{x(i-r) \\ &\quad \times x(i+n-l)x(i+m-s)\} \\ &= \sum_{r\in\mathbb{Z}} g(r)g(r+n)g(r+m) \end{aligned}$$

Note that if $G(z)$ is of order p then $C(m, n) = 0$ outside the hexagonal region

$$\{(m, n) \in \mathbb{Z}^2 : |m| \leq p, |n| \leq p, |m-n| \leq p\}$$

The bispectrum of $y(n)$ is defined as the two-dimensional Z-transform of the third-order cumulant sequence [12], that is

$$S(z_1, z_2) = \sum_{m\in\mathbb{Z}}\sum_{n\in\mathbb{Z}} C(m, n) z_1^{-m} z_2^{-n}$$

So

$$\begin{aligned} S(z_1, z_2) &= \sum_{m\in\mathbb{Z}}\sum_{n\in\mathbb{Z}}\left[\sum_{k\in\mathbb{Z}} g(k)g(k+n)g(k+m)\right] z_1^{-m} z_2^{-n} \\ &= \sum_{k\in\mathbb{Z}} g(k)\left[\sum_{m\in\mathbb{Z}} g(k+m)z_1^{-m}\right]\left[\sum_{n\in\mathbb{Z}} g(k+n)z_1^{-n}\right] \\ &= \sum_{k\in\mathbb{Z}} g(k)[G(z_1)z_1^k][G(z_2)z_2^k] \\ &= G(z_1)G(z_2)\sum_{k\in\mathbb{Z}} g(k)z_1^k z_2^k \\ &= G(z_1)G(z_2)G(z_1^{-1}z_2^{-1}) \end{aligned}$$

Note that the expression

$$\begin{aligned} &G(z_1)G(z_2)G(z_1^{-1}z_2^{-1}) \\ &= [AI(z_1)O(z_1)z_1^{-N}]\cdot[AI(z_2)O(z_2)z_2^{-N}] \\ &\quad \cdot\left[AI(z_1^{-1}z_2^{-1})O(z_1^{-1}z_2^{-1})(z_1^{-1}z_2^{-1})^{-N}\right] \\ &= A^3 I(z_1)O(z_1)I(z_2)O(z_2)I(z_1^{-1}z_2^{-1})O(z_1^{-1}z_2^{-1}) \end{aligned}$$

contains no term in N, the number of zeros of $G(z)$ outside the unit circle, even though $G(z)$ does. [This leads to the property that the algorithms of Sections 3.2 and 3.3 do not require knowledge of the value of N; this property is the reason for use of the bicepstrum rather than the complex cepstrum for this development.]

The bicepstrum of $y(n)$ is defined as

$$c(m, n) = Z^{-1}\{\log S(z_1, z_2)\}$$

and can be shown to equal [13]

$$c(m, n) = \begin{cases} \log A^3, & m = n = 0 \\ -A_n/n, & m = 0, n > 0 \\ B_{-n}/n, & m = 0, n < 0 \\ -A_m/m, & n = 0, m > 0 \\ B_{-m}/m, & n = 0, m < 0 \\ A_{-m}/m, & m = n < 0 \\ -B_m/m, & m = n > 0 \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

A relationship may be derived between the third-order

doi: 10.1049/iet-spr.2008.0141

cumulants $C(m, n)$ and the differential cepstrum parameters as follows. Defining

$$s(z_1, z_2) = \log S(z_1, z_2)$$

we have

$$\frac{\partial s(z_1, z_2)}{\partial z_1} = \frac{1}{S(z_1, z_2)} \frac{\partial S(z_1, z_2)}{\partial z_1}$$

$$\left[z_1 \frac{\partial s(z_1, z_2)}{\partial z_1}\right] S(z_1, z_2) = \left[z_1 \frac{\partial S(z_1, z_2)}{\partial z_1}\right]$$

Taking the two-dimensional inverse Z-transform

$$[-mc(m, n)] * C(m, n) = -mC(m, n) \quad (9)$$

Substituting (8) for the bicepstrum parameters $c(m, n)$ in the convolution sum yields

$$\begin{aligned} &\sum_{k=1}^{\infty} \{A_k[C(m-k, n) - C(m+k, n+k)] \\ &\quad + B_k[C(m-k, n-k) - C(m+k, n)]\} \\ &= -mC(m, n) \end{aligned} \quad (10)$$

This equation relates the third-order cumulants to the differential cepstrum parameters. The differential cepstrum parameters then relate to the minimum- and maximum-phase components of $G(z)$ via

$$\sum_{k=1}^{n} A_k i(n-k) = -ni(n) \quad (11)$$

and

$$\sum_{k=1}^{n} B_k q(n-k) = -nq(n) \quad (12)$$

for $n \geq 1$. We conclude that the theory of the bicepstrum yields a relationship (via (10)) between the set of third-order cumulants $\{C(m, n) : (m, n) \in \mathbb{Z}^2\}$ and the set of differential cepstrum parameters $\{A_k : k \geq 1\} \cup \{B_k : k \geq 1\}$, and also a relationship (via (11) and (12)) between a subset of the differential cepstrum parameters $\{A_k : 1 \leq k \leq n\} \cup \{B_k : 1 \leq k \leq n\}$ and the minimum- and maximum-phase components of $G(z)$, that is, $\{i(n) : n \geq 1\} \cup \{q(n) : n \geq 1\}$.

3.2 Bicepstrum-based gradient-adaptive algorithm 1: linear system solution

First note that (10) holds for each $(m, n) \in \mathbb{Z}^2$. We consider only equations corresponding to (m, n) pairs that lie in the union of three sets that are subsets of three lines in the plane

$$S_1 = \{(m, n) \in \mathbb{Z}^2 | \ m = 1, 0 \leq n \leq p+1\}$$

$$S_2 = \{(m, n) \in \mathbb{Z}^2 | \ n = 0, m \geq 2\}$$

$$S_3 = \{(m, n) \in \mathbb{Z}^2 | \ m = n \geq 2\}$$

[The choice of subsets S_1, S_2 and S_3 is motivated by the symmetry properties of the third-order cumulant sequence. The third-order cumulant sequence has a 6-fold symmetry in the (m, n) plane (cf. [14]), i.e. $C(m, n) = C(n, m) = C(-n, m-n) = C(-m, n-m) = C(m-n, -n) = C(n-m, -m)$. This symmetry implies that many of the equations in the system of (10) are linearly dependent. The choice of this particular subset of equations avoids linear dependence of equations due to these symmetries.]

Note that when $m > p$, the set S_2 yields difference equations in the $\{A_k\}$

$$\sum_{k=1}^{\infty} A_k C(m-k, 0) = 0, \quad m > p$$

and when $m > p$, the set S_3 yields difference equations in the $\{B_k\}$

$$\sum_{k=1}^{\infty} B_k C(m-k, m-k) = 0, \quad m > p$$

We have an infinite set of linear equations; therefore we consider only the equations for $m \leq E$, for some positive integer E. These correspond to the set $S = \{(m, n) \in S_1 \cup S_2 \cup S_3 | \ m \leq E\}$. This yields $2E + p$ equations in $2(E + p)$ unknowns $\{A_1, A_2, \ldots, A_{E+p}, B_1, B_2, \ldots, B_{E+p}\}$. This system is underdetermined by p equations; therefore we make the approximation $A_i = 0$ for $i > R = E + p - r$ and $B_i = 0$ for $i > S = E + p - s$, where $r = [p/2]$ and $r + s = p$. By (4) and (5), the accuracy of this approximation increases with increasing distance of the zeros of $G(z)$ from the unit circle; also, since $\{A_k\}$ and $\{B_k\}$ are monotonically decreasing sequences, the accuracy of the approximation increases with increasing E. The resulting reduced $(2E + p) \times (2E + p)$ system

$$\boldsymbol{Ja} + \boldsymbol{Qb} = \boldsymbol{u}$$

where $\boldsymbol{a} = (A_1 \ \ A_2 \ \ \ldots \ \ A_R)^{\mathrm{T}}$, $\boldsymbol{b} = (B_1 \ \ B_2 \ \ \ldots \ \ B_S)^{\mathrm{T}}$, $\boldsymbol{J}$ is an $(R + S) \times R$ matrix and $\boldsymbol{Q}$ is an $(R + S) \times S$ matrix is in general rank-deficient; its solution via singular value decomposition takes significant computational effort, especially for large values of E. Therefore a steepest descent algorithm is used in order to avoid the computational complexity associated with computation of the pseudoinverse. The error vector is defined as

$$\boldsymbol{e} = \boldsymbol{Ja} + \boldsymbol{Qb} - \boldsymbol{u} \quad (13)$$

The sum of error squares is then

$$I = \boldsymbol{e}^{\mathrm{T}}\boldsymbol{e} = \sum_{(m,n) \in S} e(m, n)^2$$

and the gradient of I with respect to the differential cepstrum

doi: 10.1049/iet-spr.2008.0141

parameters is given by

$$\left(\frac{\partial I}{\partial A_1} \quad \frac{\partial I}{\partial A_2} \quad \cdots \quad \frac{\partial I}{\partial A_R}\right)^{\mathrm{T}} = 2\boldsymbol{J}^{\mathrm{T}}\boldsymbol{e} \qquad (14)$$

and

$$\left(\frac{\partial I}{\partial B_1} \quad \frac{\partial I}{\partial B_2} \quad \cdots \quad \frac{\partial I}{\partial B_S}\right)^{\mathrm{T}} = 2\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{e} \qquad (15)$$

yielding a steepest descent update for the differential cepstrum parameters as

$$\boldsymbol{a}^{(k+1)} = \boldsymbol{a}^{(k)} - \mu\boldsymbol{J}^{\mathrm{T}}\boldsymbol{e}$$
$$\boldsymbol{b}^{(k+1)} = \boldsymbol{b}^{(k)} - \mu\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{e}$$

The minimum- and maximum-phase components of $G(z)$ are then solved for using the recursions (11) and (12). Finally, the gain parameter A is solved for as follows. First, the estimates of $i(n)$ and $q(n)$ yielded by (11) and (12) are combined to form $x(n) = i(n) * q^B(n)$, where $q^B(n)$ is the inverse Z-transform of $Q^B(z)$, and is equal to $q(N-n)$. Then, in light of (2), A is estimated as the mean value of $g(n)/x(n)$ over $n \in \{0, 1, \ldots, p\}$, that is

$$\hat{A} = \frac{1}{p+1}\sum_{n=0}^{p}\frac{g(n)}{x(n)}$$

3.3 Bicepstrum-based gradient-adaptive algorithm 2: non-linear system solution

A problem with the previous algorithm is that it can at best yield only an approximate solution for the differential cepstrum parameters. This is because a finite subset of the equations given by (10) does not contain enough information to solve for $\{A_k\}$ and $\{B_k\}$ exactly, and is also because certain of the $\{A_k\}$ and $\{B_k\}$ are approximated to zero.

The algorithm of this section solves iteratively for vectors $\boldsymbol{a} = (A_1\ A_2\ \ldots\ A_{E+p})^{\mathrm{T}}$ and $\boldsymbol{b} = (B_1\ B_2\ \ldots\ B_{E+p})^{\mathrm{T}}$. The linear system (10) with $(m, n) \in S$ yields $2E + p$ equations that must be satisfied; as in the algorithm of Section 3.2 this yields a cost function $I_1 = \boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}$, where the error vector $\boldsymbol{e}$ is given by (13). This time, however, none of the variables $\{A_k\}$ or $\{B_k\}$ are approximated to zero, that is, $r = s = 0$. Also, in light of (11), defining

$$\tau_n = \sum_{k=1}^{n} A_k i(n-k) + n i(n) \qquad (16)$$

we must have $\tau_n = 0$ for $n \geq 1$. We may obtain $\tau_n = 0$ for $n = 1, 2, \ldots, p$ by regarding the $\{i(n)|n = 1, 2, \ldots, p\}$ as functions of the $\{A_k|k = 1, 2, \ldots, p\}$ via (16). We can then form another cost function

$$I_2 = \boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{\tau}$$

where $\boldsymbol{\tau} = (\tau_{p+1}\ \tau_{p+2}\ \ldots\ \tau_{E+p})^{\mathrm{T}}$. To obtain an estimate of the gradient of I_2 with respect to the A_i's, we use the fact that from (16)

$$\frac{\partial \tau_n}{\partial A_j} = \begin{cases} i(n-j), & n \geq j \\ 0, & \text{otherwise} \end{cases}$$

so we have

$$\left(\frac{\partial I_2}{\partial A_1} \quad \frac{\partial I_2}{\partial A_2} \quad \cdots \quad \frac{\partial I_2}{\partial A_{E+p}}\right)^{\mathrm{T}} = 2\boldsymbol{M}_\tau^{\mathrm{T}}\boldsymbol{\tau} \qquad (17)$$

where $\boldsymbol{M}_\tau$ is the $E \times (E+p)$ Toeplitz matrix

$$\boldsymbol{M}_\tau = \begin{pmatrix} \frac{\partial \tau_{p+1}}{\partial A_1} & \frac{\partial \tau_{p+1}}{\partial A_2} & \cdots & \frac{\partial \tau_{p+1}}{\partial A_{p+E}} \\ \frac{\partial \tau_{p+2}}{\partial A_1} & \frac{\partial \tau_{p+2}}{\partial A_2} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \tau_{p+E}}{\partial A_1} & \cdots & \cdots & \frac{\partial \tau_{p+E}}{\partial A_{p+E}} \end{pmatrix}$$
$$= \begin{pmatrix} i(p) & i(p-1) & \cdots & 0 \\ 0 & i(p) & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & i(0) \end{pmatrix}$$

The gradient of I_2 with respect to the $\{B_i\}$ is zero.

A similar argument applies with

$$\sigma_n = \sum_{k=1}^{n} B_k q(n-k) + n q(n)$$

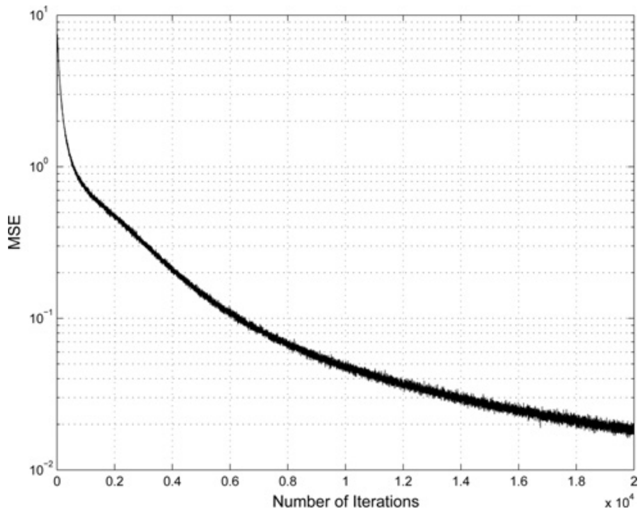

Figure 5 *MSE convergence, ensemble averaged over 5000 simulations of the adaptive lattice filtering algorithm of Section 2.2*

IET Signal Process., 2010, Vol. 4, Iss. 1, pp. 12–21
doi: 10.1049/iet-spr.2008.0141

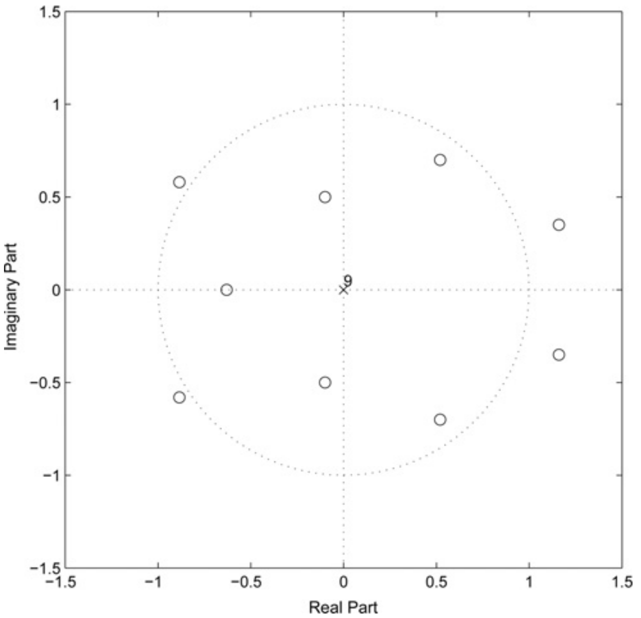

Figure 6 *Pole-zero diagram of the FIR system G(z) used to test the adaptive algorithm of Section 2.2*

The equations $\sigma_n = 0$ for $n = 1, 2, \ldots, p$ allow the recursive solution for the $q(n)$, $n = 1, 2, \ldots, p$. We form another cost function

$$I_3 = \boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\sigma}$$

where $\boldsymbol{\sigma} = (\sigma_{p+1} \quad \sigma_{p+2} \quad \ldots \quad \sigma_{E+p})^{\mathrm{T}}$, and the gradient of I_3 with respect to the $\{B_i\}$ is

$$\left(\frac{\partial I_3}{\partial B_1} \quad \frac{\partial I_3}{\partial B_2} \quad \ldots \quad \frac{\partial I_3}{\partial B_{E+p}}\right)^{\mathrm{T}} = 2\boldsymbol{M}_\sigma^{\mathrm{T}}\boldsymbol{\sigma} \qquad (18)$$

where M_σ is the $E \times (E+p)$ Toeplitz matrix

$$M_\sigma = \begin{pmatrix} q(p) & q(p-1) & \ldots & 0 \\ 0 & q(p) & \ldots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \ldots & q(0) \end{pmatrix}$$

The gradient of I_3 with respect to the $\{A_i\}$ is zero.

We seek the solution $(\boldsymbol{a}, \boldsymbol{b})$ that minimises the overall cost function

$$I = \sum_{k=1}^{3} I_k$$

It may be deduced from (14), (15), (17) and (18) that a gradient-based solution to this problem is achieved via the steepest descent updates

$$\boldsymbol{a}^{(k+1)} = \boldsymbol{a}^{(k)} - \mu\boldsymbol{J}^{\mathrm{T}}\boldsymbol{e} - \mu\boldsymbol{M}_\tau^{\mathrm{T}}\boldsymbol{\tau}$$

and

$$\boldsymbol{b}^{(k+1)} = \boldsymbol{b}^{(k)} - \mu\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{e} - \mu\boldsymbol{M}_\sigma^{\mathrm{T}}\boldsymbol{\sigma}$$

where at every iteration the final terms in the two equations are computed by using the $\boldsymbol{a}^{(k)}$ and $\boldsymbol{b}^{(k)}$ vectors to solve for $\{i(n)|n = 1, 2, \ldots, p\}$ and $\{q(n)|n = 1, 2, \ldots, p\}$ via (11) and (12).

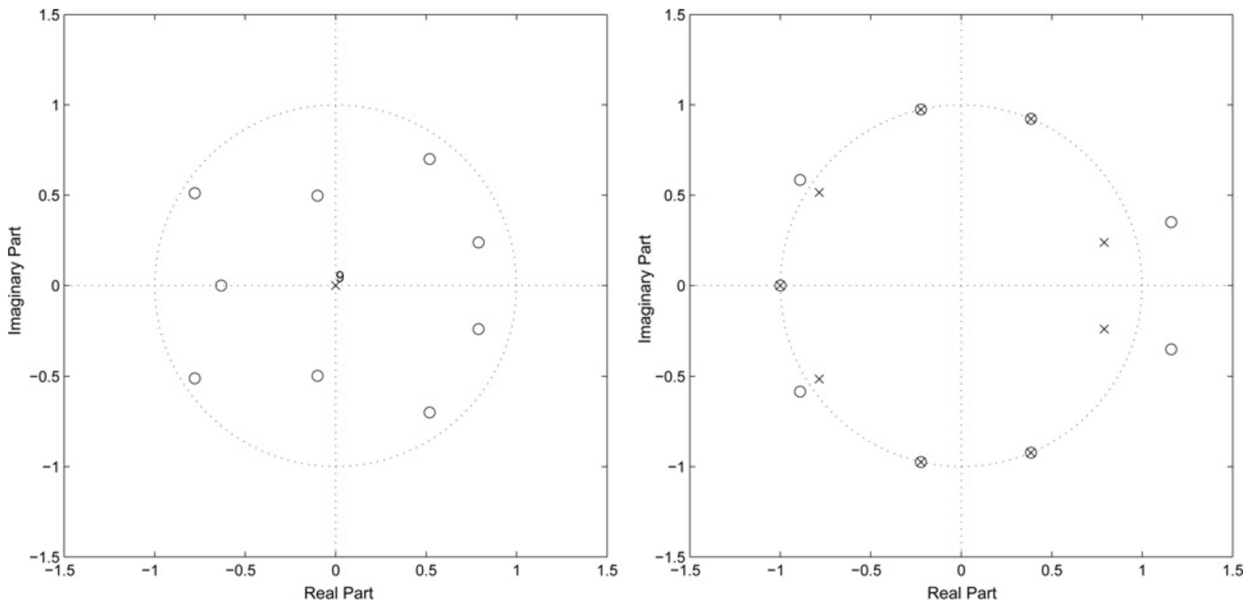

Figure 7 *Pole-zero diagram for the MP component (left) and the AP component (right) of G(z) given by the adaptive algorithm of Section 2.2*

doi: 10.1049/iet-spr.2008.0141

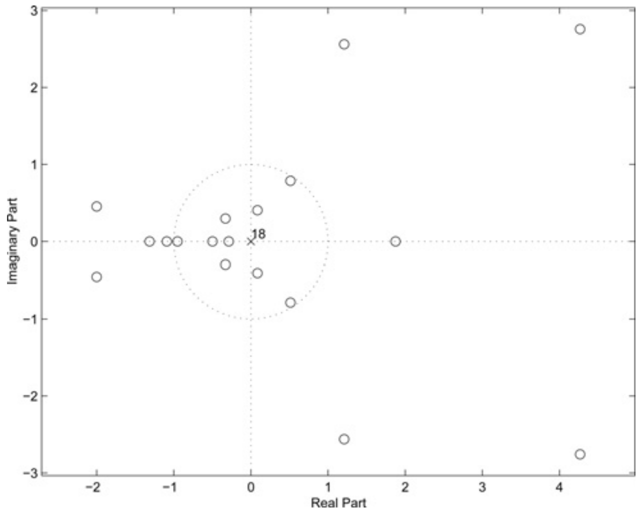

Figure 8 *Pole-zero diagram for the FIR filter G(z) used to test the performance of the bicepstrum-based algorithms*

4 Simulation results

The lattice-based adaptive algorithm of Section 2.2 was tested on a set of 5000 filters $G(z)$ of order $p = 9$, the coefficients of $G(z)$ in each case being taken from a uniform probability distribution on the interval $[-1, 1]$. Since the MSE function forms a multimodal surface, the choice of initial filter coefficients is important. The reflection coefficients were initialised to $\gamma_j^{(0)} = 0$ for $j = 1, 2, \ldots, p$ as described in Section 2.2. The gain parameter was initialised to $\beta^{(0)} = 10$. Small values of $\beta^{(0)}$ ($\sim$1) were found to result in more probable convergence to a non-minimum-phase $W(z)$, whereas large values of $\beta^{(0)}$ ($\sim$100) were found to result in numerical instability. $C(z)$ was initialised as a comb filter, that is, $c_0^{(0)} = c_p^{(0)} = 1$ and $c_j^{(0)} = 0$ for $j \notin \{0, p\}$, so that initially the all-pass section has cancelling pole-zero pairs evenly spaced on the unit circle. The excitation $\{u_k\}$ was chosen as a white sequence (with elements equiprobable in $\{-1, +1\}$) in order to identify the system $G(z)$ at all frequencies. The step size was chosen as $\mu = 10^{-3}$, small enough so that of the 5000 simulation runs, each of 20 000 iterations, none diverged.

As the filter $W(z)$ converges to the MP component of $G(z)$, the zeros of the all-pass filter move out from the unit circle and the poles of the all-pass filter (which are the reciprocals of its zeros) move inside the unit circle to provide the correct compensation. It was found that in 4628 of the 5000 cases, the filter given by the algorithm was minimum-phase (92.6% success rate), and the average final MSE was 1.9×10^{-2}. Fig. 5 shows the evolution of the ensemble averaged MSE with the number of iterations. Also, by way of illustration, z-plane results are shown in Figs. 6 and 7 for one of these filters $G(z)$. Fig. 6 shows the pole-zero diagram of the original filter to be decomposed. Fig. 7 shows the pole-zero diagrams of the MP and AP components given by the adaptive algorithm.

The adaptive algorithm of Section 3.2 was tested on a variety of systems $G(z)$ of different orders. The value of E required for adequate performance was found to increase with the filter order p, in general. Note, however, that for any p, the possibility of the zeros of $G(z)$ being very close to the unit circle means that good performance cannot be guaranteed no matter how large a value of E is chosen. As a general illustration of performance, we demonstrate the results where the algorithm is applied to the FIR filter of Fig. 8 with $p = 18$ and $M = N = 9$, which is particularly challenging in that it exhibits both a large value of p and a number of zeros close to the unit circle. The step size $\mu = 0.015$ was found to give good convergence, and the truncation depth was chosen as $E = 60$ (lower values of E caused noticeable performance degradation). Because of the rapid initial

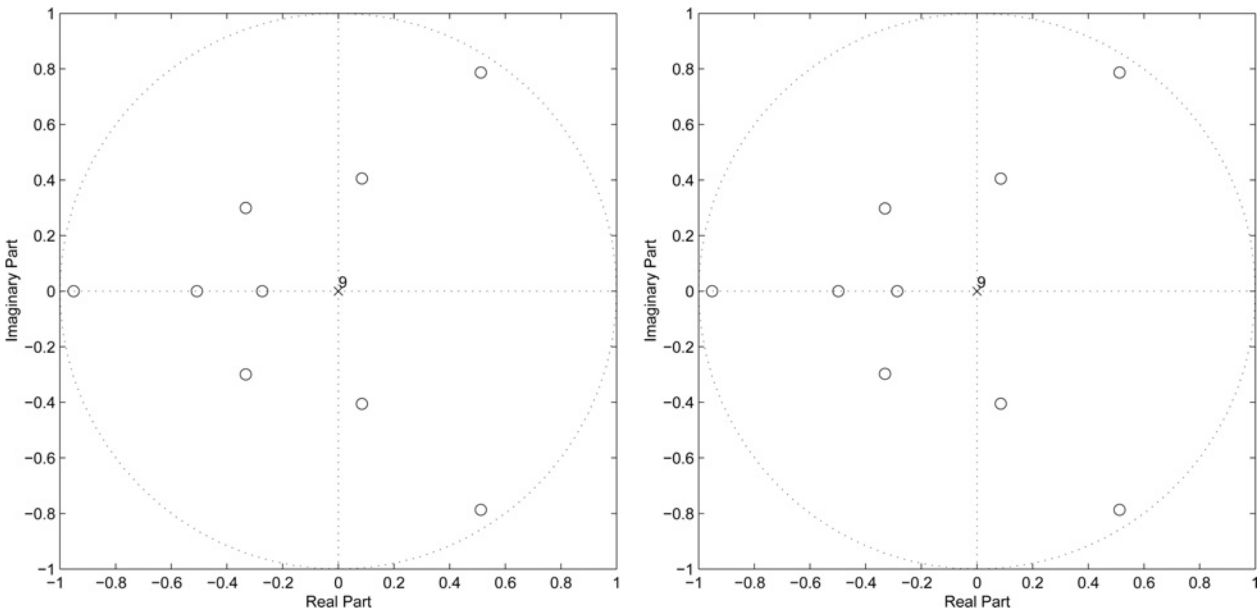

Figure 9 *Comparison of I(z) zeros given by the algorithm of Section III-B (left) with actual I(z) zeros (right) for the FIR system G(z)*

doi: 10.1049/iet-spr.2008.0141

convergence of the algorithm, only 1000 iterations were necessary to achieve a cost function value below 10^{-3}. The algorithm was terminated at this point, although (as in any other steepest descent algorithm) the cost function becomes arbitrarily small given a sufficient number of iterations. Fig. 9 compares the z-plane diagram of the estimated MP component with the actual z-plane diagram. It may be seen that the positions of the estimated zeros are indistinguishable from their actual positions in the complex plane, even though $G(z)$ has zeros close to the unit circle.

The adaptive algorithm of Section 3.3 was found to exhibit much slower initial convergence than the algorithm of Section 3.2, in general. As an illustration of performance, the algorithm was applied to the FIR system shown in Fig. 8. The simulation parameters were $E = 60$ and $\mu = 0.01$. After 8×10^4 iterations, all estimated coefficients of the polynomials $I(z)$ and $Q(z)$ were found to agree with their actual values to three decimal places (the corresponding z-plane diagrams are indistinguishable and are therefore omitted). In all cases tested, no flooring effect of the cost function was exhibited. It is left as an interesting open problem to prove that the proposed algorithm has the property of guaranteed convergence to the exact MP–AP system decomposition.

5 Conclusion

Three new adaptive algorithms that perform real-time MP–AP decomposition of an FIR system have been proposed and evaluated. The first algorithm, based on adaptive lattice filtering, models the minimum-phase component of the system as a lattice filter cascaded with a gain stage. With this adaptive filter structure, algorithm misconvergence is of low probability and is immediately detectable. The other two algorithms are based on the theory of the bicepstrum. The second algorithm solves a system of linear equations via steepest descent, and incurs an approximation error dependent on the proximity of the system zeros to the unit circle. The third algorithm solves a companion system of non-linear equations via recursion and has the distinct advantage of being capable of converging to the exact MP–AP decomposition. As well as having general application, these algorithms may be used to perform the MP–AP decomposition required for stability of compound precoding in the V.92 modem.

6 Acknowledgment

The authors would like to acknowledge the support of Enterprise Ireland and Lake Communications Ltd.

7 References

[1] OPPENHEIM A.V., SCHAFER R.W.: 'Discrete-time signal processing' (Prentice Hall, Englewood Cliffs, NJ, 1989)

[2] MCLAUGHLIN M.: 'Compound precoding: pre-equalization for channels with combined feedforward and feedback characteristics'. Irish Signals and Systems Conf., 2000, pp. 536–543

[3] FLANAGAN M.F., MCLAUGHLIN M., FAGAN A.D.: 'Compound precoding: a pre-equalization technique for the bandlimited Gaussian channel', *IET Commun.*, accepted for publication

[4] ITU-T Recommendation V.92, 'Enhancements to recommendation V.90', 2000

[5] HAYKIN S.: 'Adaptive filter theory', Information and System Science Series (Prentice Hall, 1996)

[6] MAKHOUL J.: 'Linear prediction: A tutorial review', *Proc. IEEE*, 1975, **63**, (4), pp. 561–580

[7] MANNERKOSKI J., TAYLOR D.P.: 'Blind equalization using least-squares lattice prediction', *IEEE Trans. Signal Process.*, 1999, **47**, (3), pp. 630–640

[8] MAKHOUL J., VISWANATHAN R.: 'Adaptive lattice methods for linear prediction'. IEEE Int. Conf. Acoustics, Speech and Signal Processing, ICASSP 1978, April 1978, pp. 83–86

[9] PETROPULU A.P., NIKIAS C.L.: 'Blind deconvolution based on signal reconstruction from partial information using higher-order spectra'. IEEE Int. Conf. Acoustics, Speech and Signal Processing, ICASSP 1991, April 1991, pp. 1757–1760

[10] HAYKIN S.: 'Blind deconvolution' (Prentice Hall, 1994)

[11] NIKIAS C.L., LIU F.: 'Bicepstrum computation based on second- and third-order statistics with applications'. IEEE Int. Conf. Acoustics, Speech and Signal Processing, ICASSP 1990, April 1990, pp. 2381–2385

[12] PROAKIS J.G., RADER C., LING F., NIKIAS C.L.: 'Advanced digital signal processing' (MacMillan, New York, 1992)

[13] PAN R., NIKIAS C.L.: 'The complex cepstrum of higher-order cumulants and nonminimum phase system identification', *IEEE Trans. Acoust., Speech Signal Process.*, 1988, **36**, (2), pp. 186–205

[14] HASHAD A.I., THERRIEN C.W.: 'Applying the symmetry properties of third order cumulants in the identification of non-Gaussian ARMA models'. IEEE Signal Process. Workshop on Higher-Order Statistics, June 1993, pp. 101–105