ADAPTIVE COMMON ROOT ESTIMATION AND THE COMMON ZEROS PROBLEM IN BLIND CHANNEL IDENTIFICATION

Nikolay D. Gaubitch[1], *Jacob Benesty*[2], *Patrick A. Naylor*[1]

[1]Department of Electrical and Electronic Engineering
Imperial College London
SW7 2AZ, UK

[2]Université du Québec, INRS-EMT
800 de la Gauchetière Ouest, Suite 6900
Montréal, Québec, H5A 1K6, Canada

E-mail: {ndg, p.naylor}@imperial.ac.uk, benesty@emt.inrs.ca

ABSTRACT

Many multichannel algorithms for blind channel identification and deconvolution rely on the identifiability condition that the channels are coprime, i.e. they do not have common zeros. This property has not received much attention in the literature, partly due to the difficulty of factoring the high order channel polynomials that arise in room acoustics. In this paper we propose a novel method for adaptive identification of the common roots of two polynomials. The algorithm is further used to gain some insight into the problem of common zeros in the context of adaptive blind channel identification. Simulation results are provided to demonstrate the identification and the detection of common zeros. We also consider approximately common zeros and show that they do not have to be exactly identical in order to degrade the adaptive channel identification performance.

1. INTRODUCTION

It was shown in [1] that multichannel blind system identification is dependent on the condition of two channel transfer functions being coprime, i.e. that they do not have any zeros in common. This prerequisite has been the basis for many blind channel identification algorithms e.g. [2, 3, 4] and also for multichannel inversion [5]. Despite its significance this property has not received much attention in the literature. One reason for not studying the common zero problem is the difficulty of factoring high degree polynomials such as those arising in acoustic signal processing.

In this paper, we propose a new method for adaptive and exact identification of the common roots between two polynomials without having to factor the polynomials. This is a two step approach where in the first step the distinct zero components of the two polynomials are identified blindly followed by the second step consisting of the (non-blind) estimation of the common zeros. This can also be used repeatedly to detect the number of common zeros. Furthermore, we apply our algorithm to investigate how close two roots have to be in order to be detected as common, which appears to not necessarily be only when they are identical. As a biproduct of this, it is demonstrated that the performance of the adaptive blind channel estimation algorithms is degraded severely when the zeros between two channels get close.

The idea of detecting and identifying the common roots of polynomials without having to factor them has been of interest for many years in mathematics and control systems theory. Mainly the interest has been in binary decisions of whether or not two polynomials are coprime and methods utilizing the Sylvester matrix of the polynomial coefficients are commonly applied [6]. An approximate common factor estimation and detection method was proposed in [7] and was recently extended further to the detection and identification of common roots in more than two polynomials [8].

The remainder of this paper is organized as follows. In Section 2 we formally introduce the problem of identification of the common roots in polynomials from a signal processing perspective. Section 3 presents the new two-step adaptive algorithm for common root detection and identification. In Section 4, simulation results are provided to demonstrate the proposed algorithm both for the cases of known and unknown number of common roots. Additionally, results are provided showing the effects of the distance between zeros on the adaptive blind channel estimation algorithm. Finally, in Section 5 conclusions are drawn from this investigation.

2. PROBLEM FORMULATION

For this study, a two-channel linear time-invariant system with a single common input is considered for reasons of clarity. The principles presented here can be straightforwardly extended to the general M-channel case. In the noiseless case, the relationship between the input, $x(n)$, the mth channel impulse response, $h_m(n)$, and the mth channel output, $y_m(n)$, is given by:

$$y_m(n) = h_m(n) * x(n), \qquad m = 1,2 \tag{1}$$

or equivalently

$$Y_m(z) = H_m(z)X(z), \qquad m = 1,2 \tag{2}$$

where $Y_m(z)$, $H_m(z)$ and $X(z)$ are the z-transforms of $y_m(n)$, $h_m(n)$ and $x(n)$ respectively and $*$ denotes convolution. Given the input and the output sequences, we would like to find the zeros that are common to both transfer functions, $H_1(z)$ and $H_2(z)$.

Let $H_C(z)$ denote the component with the roots common to both transfer function polynomials and let $H'_m(z)$ denote the characteristic zeros component of the mth channel, i.e. those zeros contained in $H_1(z)$ but not in $H_2(z)$. The transfer functions in (2) can now be written as:

$$H_m(z) = H_C(z)H'_m(z), \qquad m = 1,2 \tag{3}$$

where $\deg[H'_m(z)] \leqslant \deg[H_m(z)]$. Thus, the problem is to detect the number of common zeros, $\deg[H_C(z)]$ and also to identify the common roots component, $H_C(z)$.

3. ADAPTIVE COMMON ROOT ESTIMATION

In this Section, we derive the new method for identification of common roots. This is done in two steps. First, the components of the impulse response that do not contain common roots are blindly identified. Next using these estimates, the common roots part is found.

3.1 Step 1: Estimating the characteristic root components

Let us assume that the number of common roots is known, which is not true in practice and a method for detecting the number of common roots is provided at the end of this section. By substituting (3) into (2), the system outputs can be written:

$$\begin{aligned} Y_1(z) &= X(z)H_C(z)H_1'(z) = X_C(z)H_1'(z), \\ Y_2(z) &= X(z)H_C(z)H_2'(z) = X_C(z)H_2'(z), \end{aligned} \tag{4}$$

where $X_C(z) = X(z)H_C(z)$.

Since the two transfer functions $H_1'(z)$ and $H_2'(z)$ do not contain any common zeros, we can identify them blindly using, e.g. the adaptive multichannel LMS (MCLMS) algorithm [3, 4]. The error signal based on the cross-relation between the channels [1] can be formulated using the portions of the transfer functions with the distinctive zeros as [3]:

$$\tilde{e}(n) = \mathbf{y}_1^T(n)\hat{\mathbf{h}}_2'(n) - \mathbf{y}_2^T(n)\hat{\mathbf{h}}_1'(n), \tag{5}$$

where $\mathbf{y}_m(n) = [y(n)\ y(n-1)\ \ldots\ y(n-L'+1)]^T$ is the input vector, $\hat{\mathbf{h}}_m'(n) = [\hat{h}_{m,0}'(n)\ \hat{h}_{m,1}'(n)\ \ldots\ \hat{h}_{m,L'-1}'(n)]^T$ are the estimates of $\mathbf{h}_m'$ at time n. L' is the length of the channels $\mathbf{h}_m'$ and is the difference between the full channel length, L and the common zeros component, L_C. A priori knowledge of L is assumed, which is common practice in blind channel estimation. Consequently, both impulse responses can be estimated simultaneously by minimizing the squared error:

$$\hat{\mathbf{h}}' = \arg\min_{\hat{\mathbf{h}}'} \tilde{J}(n), \quad \text{subject to } \|\hat{\mathbf{h}}'\| = 1 \tag{6}$$

where $\tilde{J}(n) = \mathscr{E}\left\{\tilde{e}^2(n)\right\}$ is the cost function with $\mathscr{E}\{\cdot\}$ being the expectation operator. The constraint is imposed in order to avoid the trivial estimate of zero elements.

The channel estimate can be obtained adaptively using the following update equation [3]:

$$\hat{\mathbf{h}}'(n+1) = \frac{\hat{\mathbf{h}}'(n) - 2\mu_{hp}\tilde{e}(n)[\mathbf{y}(n) - \tilde{e}(n)\hat{\mathbf{h}}'(n)]}{\|\hat{\mathbf{h}}'(n) - 2\mu_{hp}\tilde{e}(n)[\mathbf{y}(n) - \tilde{e}(n)\hat{\mathbf{h}}'(n)]\|}, \tag{7}$$

where $\mathbf{y}(n) = [\mathbf{y}_1^T(n)\ \mathbf{y}_2^T(n)]^T$ are the concatenated input vectors, $\hat{\mathbf{h}}(n) = [\hat{\mathbf{h}}_2'^T(n)\ \ -\hat{\mathbf{h}}_1'^T(n)]^T$ are the impulse response estimates and μ_{hp} is a small positive adaptation step size. Note that the normalization results from the constraint in (6). It has been shown in [3] that this algorithm converges in the mean to the correct channel up to a common scaling factor when the transfer functions are coprime. Moreover, this adaptive minimization is equivalent to finding the eigenvector corresponding to the smallest eigenvalue in the data matrix using either singular value decomposition on the data matrix itself or eigenvalue decomposition on the input covariance matrix [9]. However, the adaptive algorithms are more computationally efficient.

3.2 Step 2: Estimating the common root components

We can now use the channel estimates obtained from (7) to generate the intermediate sequences $x_1'(n) = x(n) * \hat{h}_1'(n)$ and $x_2'(n) = x(n) * \hat{h}_2'(n)$. The common zero component, $h_C(n)$, can then be found by estimating $y_1(n)$ and $y_2(n)$. Based on this, the estimation error at the mth channel can be written:

$$e_m(n) = y_m(n) - \mathbf{x}_m'^T(n)\hat{\mathbf{h}}_C, \quad m = 1, 2 \tag{8}$$

where $\mathbf{x}_m'(n) = [x_m'(n)\ x_m'(n-1)\ \ldots\ x_m'(n-L_C+1)]^T$ is the input vector at time n, $\hat{\mathbf{h}}_C = [\hat{h}_{C,0}\ \hat{h}_{C,1}\ \ldots\ \hat{h}_{C,L_C-1})]^T$ is the vector of the estimated common zero component and $L_C = \deg[H_C(z)] + 1$ is the length of $h_C(n)$.

Using both estimation errors, a cost function can be formulated as follows:

$$J(n) = \mathscr{E}\left\{e_1^2(n) + e_2^2(n)\right\}. \tag{9}$$

The common zeros component of the two transfer functions can thus be found by minimizing the cost function in (9) with respect to $\hat{\mathbf{h}}_C$:

$$\hat{\mathbf{h}}_C = \arg\min_{\hat{\mathbf{h}}_C} J(n). \tag{10}$$

We next deploy a least mean square (LMS) adaptive algorithm [10] to solve the stated minimization problem and thus, to efficiently estimate the common zeros component. The iterative update of the estimated coefficients is written:

$$\hat{\mathbf{h}}_C(n+1) = \hat{\mathbf{h}}_C(n) - \frac{1}{2}\mu_{hc}\nabla J(n) \tag{11}$$

where μ_{hc} is the adaptation step size and ∇ is a gradient operator. In order to find the gradient we take the partial derivatives with respect to each component in $\hat{\mathbf{h}}_C$:

$$\begin{aligned} \nabla J(n) &= \frac{\partial J(n)}{\partial \hat{\mathbf{h}}_C} \\ &= \mathscr{E}\left\{\frac{\partial e_1^2(n)}{\partial \hat{\mathbf{h}}_C} + \frac{\partial e_2^2(n)}{\partial \hat{\mathbf{h}}_C}\right\} \\ &= \mathscr{E}\left\{2\frac{\partial e_1(n)}{\partial \hat{\mathbf{h}}_C}e_1(n) + 2\frac{\partial e_2(n)}{\partial \hat{\mathbf{h}}_C}e_2(n)\right\} \\ &= -2\mathscr{E}\left\{\mathbf{x}_1'(n)e_1(n) + \mathbf{x}_2'(n)e_2(n)\right\}. \end{aligned} \tag{12}$$

Inserting the estimation errors in (8) into (12) we arrive to the following expression for the gradient:

$$\nabla J(n) = -2(\mathbf{p}_1 + \mathbf{p}_2) + 2(\mathbf{R}_1 + \mathbf{R}_2)\hat{\mathbf{h}}_C, \tag{13}$$

where $\mathbf{R}_m = \mathscr{E}\{\mathbf{x}_m'(n)\mathbf{x}_m'^T(n)\}$ is the autocorrelation matrix of the input signal and $\mathbf{p}_m = \mathscr{E}\{\mathbf{x}_m'(n)y_m(n)\}$ is the cross-correlation between the input and the desired output.

For the LMS adaptive filter, the instantaneous estimates of the autocorrelation matrix, $\hat{\mathbf{R}}_m = \mathbf{x}_m'(n)\mathbf{x}_m'^T(n)$, and the cross-correlation vector, $\hat{\mathbf{p}}_m = \mathbf{x}_m'(n)y_m(n)$ are considered. Substituting these into (13) we arrive to the instantaneous gradient estimate:

$$\hat{\nabla} J(n) = -2\left(\mathbf{x}_1'(n)e_1(n) + \mathbf{x}_2'(n)e_2(n)\right). \tag{14}$$

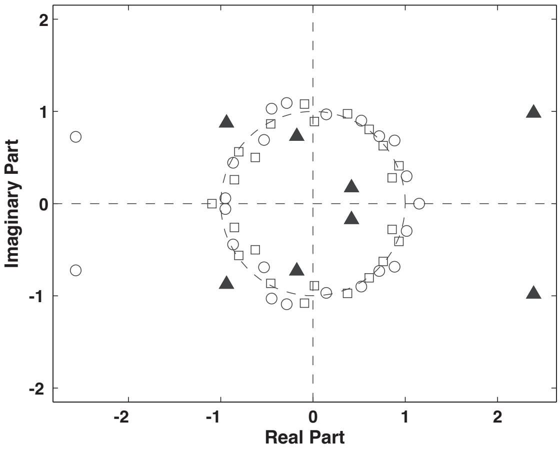

Figure 1: Channel zeros for channel 1 (circles), channel 2 (squares) and the common zeros (triangles).

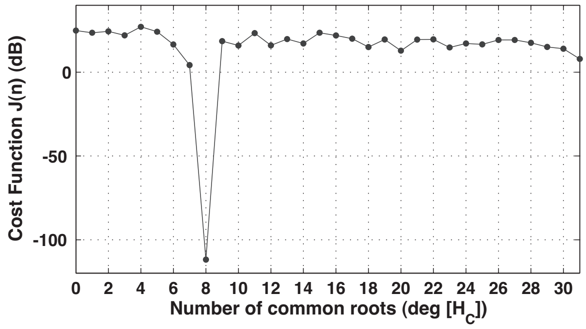

Figure 2: Cost function after $n = 200000$ iterations vs. estimated number of common zeros $\deg[H_C(z)]$.

Component Estimation	Convergence to $\mathrm{NPM}(n) = -60\mathrm{dB}$
Characteristic zeros	$n = 19644$
Common zeros	$n = 787$

Table 1: Number of iterations, n, required for the algorithm to converge to NPM $= -60$ dB for a channel of length $L = 32$ and with 8 common roots.

Finally, substituting (14) into (11), we obtain the coefficient update equation:

$$\hat{\mathbf{h}}_C(n+1) = \hat{\mathbf{h}}_C(n) + \mu_{hc}(\mathbf{x}'_1(n)e_1(n) + \mathbf{x}'_2(n)e_2(n)). \quad (15)$$

This far it has been assumed that the order of the common zeros component is known. Since this is not the case in practice, we propose that the identification can be performed repeatedly starting with the full channel length L then reducing this by 1 at each repetition and monitoring the mean squared error cost function, J. In this way, we start with an initial assumption of no common roots and then increment the number of common roots for each subsequent repetition. The true number of common roots is indicated when the mean square error is minimum as will be demonstrated by our simulation results.

4. SIMULATIONS

In this Section, simulations and results are provided to demonstrate the proposed algorithm. As a performance metric, the normalized projection misalignment (NPM) was used, which is defined as follows [11]:

$$\mathrm{NPM}(n) = 20\log_{10}\left(\frac{\|\varepsilon(n)\|}{\|\mathbf{h}\|}\right), \quad (16)$$

with

$$\varepsilon(n) = \mathbf{h} - \frac{\mathbf{h}^T\hat{\mathbf{h}}(n)}{\hat{\mathbf{h}}^T(n)\hat{\mathbf{h}}(n)}\hat{\mathbf{h}}(n),$$

where $\mathbf{h} = [\mathbf{h}_1^T \ \mathbf{h}_2^T]^T$ is the composite channel vector and $\hat{\mathbf{h}}(n) = [\hat{\mathbf{h}}_1^T(n) \ \hat{\mathbf{h}}_2^T(n)]^T$ is the composite vector of the channel estimates. Using this measure only the misalignment is accounted for, ignoring the effect of the arbitrary constant [11].

For the first experiment, we used a system comprising two random channels of length $L = 32$ with eight known common roots, i.e. $L_C = 9$. The channels zeros are shown in the z-plane plot in Fig. 1, where the characteristic roots for channel one are marked with circles, those for channel two with squares and the common roots with triangles. It was explicitly assured that the minimum inter-channel separation between roots satisfies $\Delta_i \geqslant 0.1$, $\forall i$, where $\Delta_i = \min\{|z_1(i) - \mathbf{z}_2|\}$, $i = 0, 1, \ldots, L-2$ is the distance between the ith zero in the first transfer function, $z_1(i)$, and any other zero of the second transfer function, $\mathbf{z}_2$, with $\mathbf{z}_m = [z_m(0) \ z_m(1) \ \ldots \ z_m(L-2)]$, $m = 1, 2$ being a vector of the zeros in channel m.

The algorithm was run using $\mu_{hp} = 10^{-5}$ for the blind identification and $\mu_{hc} = 0.2$ for the common zeros estimation. We used white Gaussian noise to excite the system under consideration. The results are summarized in Table 1 where it is shown that the the blind channel identification stage converged, in terms of NPM, to -60 dB in 19644 iterations while the second part, the common root estimation adaptive filter converged to -60 dB in 787 iterations. It is also important to note that the blind channel estimator would continue to converge, while the common roots identification has a convergence floor depending on the accuracy of Step 1.

Subsequently, we investigated the case where the order of the common zeros component is unknown and how the number of common roots can by detected by repetition. The correct number of common roots gives the minimum mean squared error in the common roots identification algorithm. Using the same channels in Fig. 1, we repeatedly ran the algorithm increasing the assumed number of common zeros each time. The algorithm was let to run for $n = 200000$ iterations at each repetition. The result is shown in Fig. 2 where it can be seen that the mean squared error after convergence is minimum for 8 common zeros, which is the correct answer.

Finally, we examined how close two zeros have to be in order to be detected as common by the algorithm. For clarity, different channels were used here with only two common zeros as shown in Fig. 3 where the characteristic roots

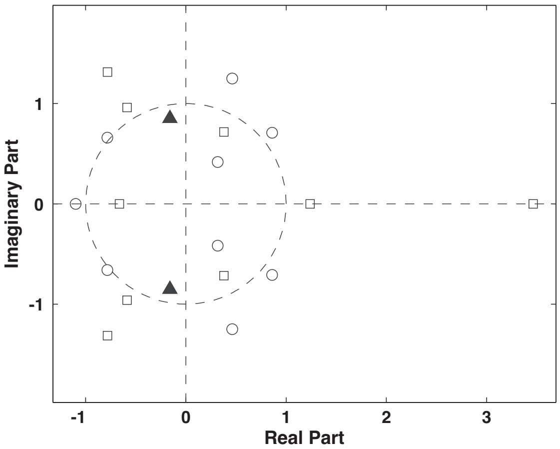

Figure 3: Channel zeros for channel 1 (circles), channel 2 (squares) and the common zeros (triangles).

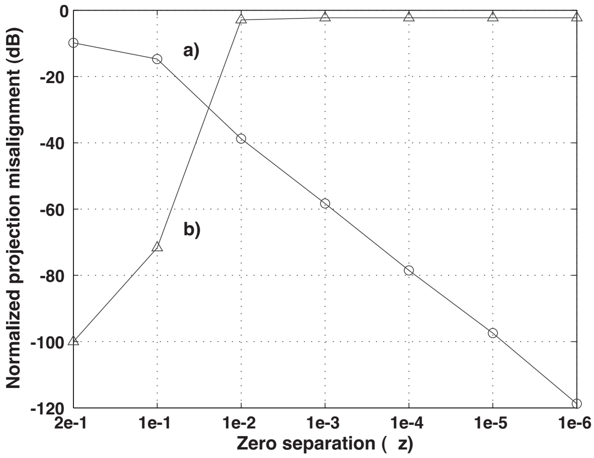

Figure 4: Misalignment after convergence vs. 'common zero' separation for a) the common zero estimation algorithm and b) blind estimation of the full channel.

for channel one are represented by circles, those for channel two with squares and the common roots with triangles. The inter-channel roots distance was set to $\Delta_i \geqslant 0.2$, $\forall i$. Keeping all the remaining zeros fixed, the common zeros were separated to a distance of $\Delta z = 0.2$ and then moved towards each other at the steps $\Delta z = 10^{-\gamma}$ for $\gamma = 1, 2, \ldots, 6$. $\Delta z = |z_{c1}(j) - z_{c2}(j)|$, $j = 0, 1, \ldots, L_C - 2$ is the distance between the 'common zeros' of the two channels. The result is shown in Fig. 4. It is interesting to note that zeros which are common to within $\Delta z = 0.01$ can be correctly identified as 'common'. This gives rise to two interesting points. First, our algorithm would be resistant to small perturbations of the common zeros of two polynomials. Second, the identification of the full length channels is severely degraded at that distance. The latter agrees with the result for an ill-conditioned channel in [3].

5. CONCLUSION

We have proposed and derived a novel approach for finding the common roots in polynomials using adaptive filters in a signal processing framework. The most attractive feature of this method lies in its ability to identify the components due to the common roots of two polynomials without having to factor these. It was demonstrated by simulations that this approach can accurately detect the number of common roots and the impulse response of these components. From our results two interesting observations were brought forward. First, it appears that the accuracy of blind channel algorithms is strongly affected by the distance of zeros and that two zeros do not have to be exactly equal to be considered common in adaptive blind channel identification. Secondly, this same feature makes the common root detection algorithm robust to perturbations that result in small separation of the roots that should otherwise be considered common. The proposed techniques can be applied to study the impact of common zeros on blind channel identification resulting in a better understanding and improvement of blind channel estimation algorithms.

REFERENCES

[1] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," vol. 43, no. 12, pp. 2982–2993, Dec. 1995.

[2] S. Subramaniam, A. P. Petropulu, and C. Wendt, "Cepstrum-based deconvolution for speech dereverberation," vol. 4, no. 5, pp. 392–396, Sept. 1996.

[3] Y. Huang and J. Benesty, "Adaptive multi-channel least mean square and newton algorithms for blind channel identification," *Signal Process.*, vol. 82, no. 8, pp. 1127–1138, Aug. 2002.

[4] ——, "A class of frequency-domain adaptive approaches to blind multichannel identification," vol. 51, no. 1, pp. 11–24, Jan. 2003.

[5] M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," vol. 36, no. 2, pp. 145–152, Feb. 1988.

[6] T. Kailath, *Linear Systems*. Englewood Cliffs, N.J.: Prentice Hall, 1980.

[7] P. Stoica and T. Söderström, "Common factor detection and estimation," *Automatica*, vol. 33, no. 5, pp. 1985–1989, May 1997.

[8] M. Agrawal, P. Stoica, and P. Ågren, "Common factor estimation and two applications in signal processing," *Signal Process.*, vol. 84, no. 2, pp. 421–429, Feb. 2004.

[9] S. Gannot and M. Moonen, "Subspace methods for multi-microphone speech dereverberation," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1074–1090, Oct. 2003.

[10] S. Haykin, *Adaptive Filter Theory*, 4th ed. Upper Saddle River, N.J.: Prentice Hall, 2001.

[11] D. R. Morgan, J. Benesty, and M. M. Sondhi, "On the evaluation of estimated impulse responses," vol. 5, no. 7, pp. 174–176, July 1998.