

AN ESTIMATION ALGORITHM FOR AR MODELS WITH CLOSELY LOCATED LIGHTLY DAMPED LOW FREQUENCY POLES

*M Harteneck*¹

*R W Stewart*¹

*J G McWhirter*²

*I K Proudler*²

¹Signal Processing Division, Dept. of Electrical and Electronic Engineering, University of Strathclyde, Glasgow, G1 1XW, Scotland, U.K., email: moritz@spd.eee.strath.ac.uk, bob@spd.eee.strath.ac.uk

²DRA Malvern, St Andrews Road, Great Malvern, Worcestershire, WR14 3PS, U.K., email: jgm@signal.dra.hmg.gb, proudler@signal.dra.hmg.gb

ABSTRACT

In this paper we present a pole estimation algorithm which is based on an overdetermined adaptive IIR filter with an additional postprocessing stage to extract the pole locations from the adaptive weights. The adaptive filtering algorithm used, is a pseudo-linear regression algorithm which is solved by a time-recursive QR decomposition. Two pole classification schemes are presented to separate the true poles and the superfluous poles. The classification schemes are based on the occurrence of pole-zero cancelation and on the pole movement in the z -plane. Floating point simulations are presented to demonstrate the performance of the proposed algorithm.

1. INTRODUCTION

Autoregressive (AR) modeling [1] is an important issue in many signal processing and control applications. Some examples are: speech analysis, where the poles of the AR model determine the formant frequencies and bandwidths which can then be used further for coding and voice systems; sensor array processing, where the direction-of-arrival estimation using a linear array can be formulated as an adaptive polynomial rooting problem; and biomedical engineering where the trajectories of the roots of the AR model of EEG signals have been used to predict the onset of seizure.

Usually these pole locations are calculated in a two step algorithm where first the polynomial coefficients are estimated using adaptive IIR filtering techniques such as the full gradient, simplified gradient RPE algorithm [2] or even Feintuch's LMS algorithm [3] and then the roots are calculated using standard factorization schemes for polynomials such as Müller's Method [4].

One problem which causes these algorithms to fail is that if the poles to be identified are located too close together; then these poles are indistinguishable due to the inherent process noise of these gradient based algorithms and due to observation noise in the signals. Another problem which arises with the adaptive IIR filtering techniques is the potential danger of instability as the time-varying poles of the adaptive system might migrate outside of the stability region and cause the algorithm to diverge.

In this paper we review briefly in Sect. 2.1. the adaptive IIR filtering algorithm used, which is based on a pseudo-linear regression and a time-recursive QR decomposition. More details on the derivation and performance of this al-

gorithm can be found in [5, 6, 7]. This algorithm is then used in an overdetermined system identification setup to identify the poles of the unknown system. As the algorithm is overdetermined more poles and zeros are calculated than are actually present in the unknown system and therefore classification schemes have to be developed to separate the true poles from the superfluous poles. The information used for this separation is, in the noise-free case, that unused pole-zero pairs cancel and, in the noisy case, that true poles have a smaller variance than superfluous poles which move randomly in the z -plane. In Sect. 2.2., these two classification schemes are presented which are used to extract the poles from the information originating from the adaptive algorithm. In Sect. 2.3. considerations of the applicability of multirate schemes are presented which enhance the quality of estimation and the robustness of the proposed algorithm to observation noise. In Sect. 3. simulation results are presented to show the performance of the proposed algorithm.

2. DERIVATION

2.1. Adaptive IIR Filter

An adaptive IIR filter in output-error formulation tries to predict a signal $y(k)$ at time k by a linear combination of the input signal $x(k)$ and past samples of the estimated signal $\hat{y}(k)$, i.e.

$$\hat{y}(k) = \sum_{i=0}^{M-1} \hat{a}_i(k)x(k-i) + \sum_{i=1}^{L-1} \hat{b}_i(k-i)\hat{y}(k-i), \quad (1)$$

where $\{\hat{a}_i, \hat{b}_i\}$ are the adaptive weights and M and L are the number of the feedforward and feedback weights, respectively.

The error signal which the algorithm tries to minimize is defined as

$$e(k) = y(k) - \hat{y}(k). \quad (2)$$

In a least squares formulation, the performance criterion to minimize is an estimate of the power of the error signal or, for time varying systems, a windowed version thereof. The performance criterion $\xi(k)$ can be written as

$$\xi(k) = \sum_{i=0}^{k-1} \lambda^i e^2(k-i), \quad (3)$$

where λ is a forgetting factor which is usually chosen slightly smaller than 1 to make the algorithm "forget" system

changes of the unknown system and transients in the initialization phase.

This performance criterion $\xi(k)$ can be written, using appropriate vector definitions and approximations [6], in a matrix equation as

$$\xi(k) \approx \left\| \Lambda^{\frac{1}{2}}(k) \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(k) \end{bmatrix} - \underbrace{\begin{bmatrix} \mathbf{x}^T(1) & \hat{\mathbf{y}}^T(1) \\ \mathbf{x}^T(2) & \hat{\mathbf{y}}^T(2) \\ \vdots & \vdots \\ \mathbf{x}^T(k) & \hat{\mathbf{y}}^T(k) \end{bmatrix}}_{\mathbf{A}(k)} \underbrace{\begin{bmatrix} \hat{\mathbf{a}}(k) \\ \hat{\mathbf{b}}(k) \end{bmatrix}}_{\hat{\mathbf{w}}(k)} \right\|^2$$

$$\approx \left\| \Lambda^{\frac{1}{2}}(k) \mathbf{y}(k) - \Lambda^{\frac{1}{2}}(k) \mathbf{A}(k) \hat{\mathbf{w}}(k) \right\|^2, \quad (4)$$

where $\|\cdot\|^2$ denotes the Euclidean norm, $\Lambda(k)$ is a matrix to incorporate the forgetting factor, $\mathbf{x}(k)$ and $\hat{\mathbf{y}}(k)$ consist of the input signal $x(k)$ and the estimated output signal $\hat{y}(k)$, respectively, and $\hat{\mathbf{w}}(k)$ contains the adaptive parameters.

To minimize now the performance criterion (3), standard factorization schemes like the singular value decomposition or the QR decompositions can be used. To obtain an algorithm with a computational complexity of $O((M+L)^2)$, a time recursive QR decomposition as in [8] is used. The resulting parameters are then used in the classification schemes to extract the pole positions.

2.2. Classification Schemes

The adaptive filter produces at each time instant $M-1$ zeros and $L-1$ poles, which are, as the algorithm operates in an overdetermined setup, more than exist in the unknown system. Close examinations of the poles and zeros for each time step and over time have shown two main characteristics:

- In a low observation noise environment, superfluous poles and zeros cancel for each time step and move randomly in the z -plane (c.f. Figs. 2 and 3);
- In an environment with a higher observation noise, the superfluous poles and zeros do not cancel any more but still exhibit a large variance (c.f. Figs. 4 and 5).

Based on these two observations, two different classification schemes have been used to identify the poles. The first scheme is based on the fact that poles and zeros cancel in a low SNR environment. This scheme can be represented as

- Operate the adaptive algorithm for the system identification.
- Calculate the pole and zero locations for each time instant k using polynomial rooting techniques.
- Cancel poles and zeros which are in close vicinity for each time instant k .
- Eliminate outliers.
- Perform time averaging to obtain good estimates.

The second scheme is based on the fact that existing poles show a small variance and only involves the calculated pole locations:

- Operate the adaptive algorithm for the system identification.
- Calculate the pole locations for each time instant k using polynomial rooting techniques.
- Sort poles according to variance.
- Perform time averaging to obtain good estimates.

Simulation results show that the first scheme works better in environments with a low level of observation noise whereas the second scheme works better under conditions with a higher level. This is caused by the fact that with a low level of observation noise, some of the superfluous poles also exhibit the property of a low variance in their movements.

The second scheme performs well under observation noise until the observation noise level is high enough to cause closely located poles to "blend" into each other, i.e. become indistinguishable. One possible solution is to apply multi-rate techniques which are discussed in Section 2.3.

Another important fact is that the algorithm tries to evenly distribute the poles around the unit circle and therefore the closer together the poles of the unknown system are located, the higher is the order of the adaptive filter.

2.3. Multirate Considerations

To increase the resolution of the proposed technique and the robustness towards observation noise, these techniques can be used in a decimated setup. Therefore, first, the two input signals have to be filtered so that only the band of interest is contained in them and then they have to be decimated with a decimation factor D , whereby the level of aliasing has to be kept small to enable the adaptive filter to adapt correctly. This can be ensured by an appropriate choice of D as:

If $[f_l, f_u]$ is the occupied frequency range of a signal, where $0 \leq f_l < f_u \leq 1$ are the normalized lower and upper band edges, and D is the subsampling ratio applied to the signal, then, in order to keep the aliasing small, the subsampling ratio D has to be chosen in such a way that the numbers of the set $\left\{ \frac{1}{D}, \frac{2}{D}, \dots, \frac{D-1}{D} \right\}$ are not contained in $[f_l, f_u]$, i.e.

$$\left\{ \frac{1}{D}, \frac{2}{D}, \dots, \frac{D-1}{D} \right\} \notin [f_l, f_u]. \quad (5)$$

Under these conditions, the adaptive filter will identify the poles at the locations $z \mapsto z^D$. This mapping separates the poles further as the angles of the poles get multiplied by D and therefore allow a higher level of observation noise. The second effect is that, as the radii of the poles are smaller than 1, the poles move further away from the unit circle which facilitates convergence and estimation accuracy of the algorithm. Another advantage is that as the poles are distributed more evenly over the frequency range, the adaptive filter needs a lower estimation order than in a non-decimated setup.

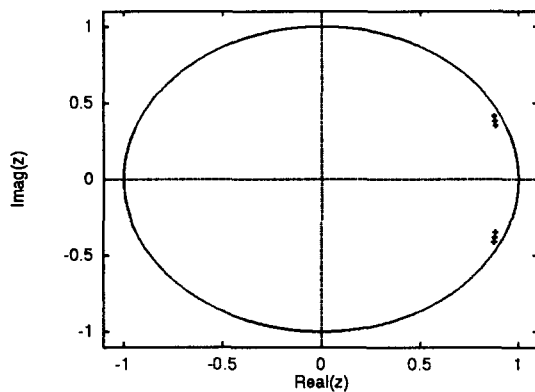


Figure 1. Unknown System, Pole-Zero Plot

i	r_i	$\Theta_i(p_i)$
0	0.97	± 0.14
1	0.96	± 0.13
2	0.95	± 0.12

Table 1. Unknown System, Exact Pole Locations

3. SIMULATIONS

To demonstrate the performance of the proposed algorithm, floating point simulations in a noisy and noise-free environment are presented for the non-decimated case in Section 3.1. and for the decimated case in Section 3.2.

The unknown system used for the simulations was always the 6th-order AR model whose pole-zero plot is shown in Fig. 1 and whose exact pole locations are shown in Tab. 1, where r_i is the radius of the pole i and Θ_i is the angle. It can be seen that this unknown system has poles located close to the unit circle, close to each other and in the low frequency region. The driving input noise and the observation noise were independent and white, with Gaussian distributions. The forgetting factor was set to $\lambda = 0.999$ which is equivalent to a effective window length of about 10000 samples.

3.1. Non-decimated Simulations

In Figs. 2 and 3 the calculated pole and zero locations, respectively, of an observation noise free simulation of the algorithm are shown for the last 1000 iterations of the sim-

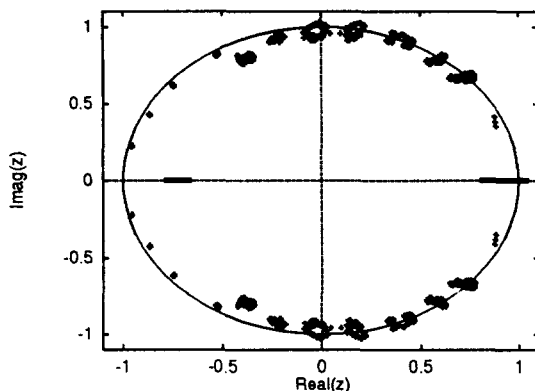


Figure 2. Noise-Free Simulation Results, Poles

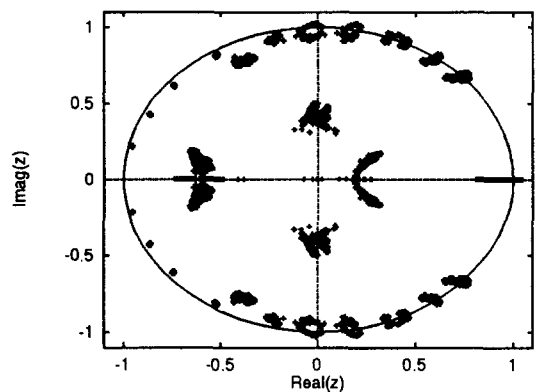


Figure 3. Noise-Free Simulation Results, Zeros

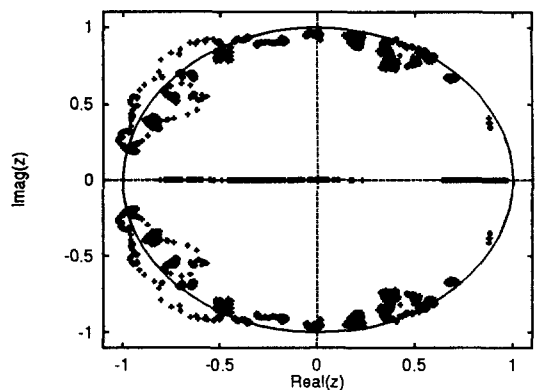


Figure 4. Noisy Simulation Results, Poles, SNR = 70 dB

ulations. The adaptive filter consists of $L = 30$ feedforward and $M = 30$ feedback weights. It can be seen that the superfluous poles and zeros cancel and the poles of interest show a small variance. In Tab. 2 it was recorded that these pole estimations yield a variance between -89 dB and -83 dB.

In Figs. 4 and 5 the calculated pole and zero location for the same setup but with signal to observation noise ratio of 70 dB are shown. Now it can be seen that the poles move away from the zeros and that the poles of interest show a small variance, i.e. a small movement in the z -plane. In Tab. 2 it can be seen that these poles have a variance between -60 dB and -52 dB.

3.2. Decimated Simulations

In this section, a comparison between the non-decimated setup and the decimated setup is performed. The algorithm was therefore applied to the same system but this time the input signals were first filtered to select the frequency band, then uniformly decimated by D and finally the algorithm was used to estimate the pole positions.

For the decimation by $D = 2$, the filter for the frequency selection was a 100 taps linear-phase FIR filter with a normalized cut-off frequency of 0.44 which yields a maximum aliasing level of -90 dB. The filter order of the adaptive filter has been chosen in this case to be $L = 20$ feedforward and $M = 20$ feedback weights.

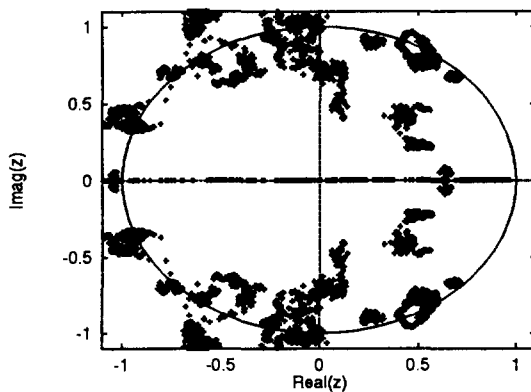


Figure 5. Noisy Simulation Results, Zeros, SNR = 70dB

SNR (dB)	Variance (dB) $D = 1$ $L = M = 30$	Variance (dB) $D = 2$ $L = M = 20$	Variance (dB) $D = 4$ $L = M = 10$
∞	[-89, -83]	[-103, -97]	[-101, -89]
70	[-60, -52]	[-83, -74]	[-80, -72]
64	no result	[-77, -68]	[-74, -67]
58	no result	[-71, -62]	[-67, -60]
52	no result	[-65, -56]	[-61, -56]
46	no result	[-58, -49]	[-57, -51]
40	no result	[-52, -43]	[-52, -44]
34	no result	no result	[-47, -39]
27	no result	no result	no result

Table 2. Estimation Variance for the decimation D and the feedforward and feedback weights L and M respectively

For the decimation by $D = 4$, the filter for the frequency selection was a 100 taps linear-phase FIR filter with a normalized cut-off frequency of 0.2 which yields a maximum aliasing level of -80 dB. The order of the adaptive filter has been chosen in this case to be $L = 10$ feedforward and $M = 10$ feedback weights.

The results for the two decimated setups ($D = 2$ and $D = 4$) and the non-decimated setup ($D = 1$) are shown in Tab. 2. Recall that the non-decimated setup fails when the signal to observation noise level exceeds 70 dB whereas the decimated setup of $D = 2$ and $D = 4$ give useful results up to a level of 40 dB and 34 dB, respectively. For the observation noise levels where both cases give results the decimated setup with $D = 4$ gives an improvement of 20 dB over the non-decimated setup as well as the setup with a decimation of $D = 2$.

4. CONCLUSIONS

In this paper we proposed an estimation algorithm for lightly damped low frequency poles. The algorithm was based on a least squares pseudo-linear regression IIR filtering algorithm and two classification schemes to extract the pole locations from the adaptive weights. Then, the use of the algorithm in a decimated environment was evaluated and it was shown that a reduction in estimation error and an increase in robustness towards observation noise can be achieved. Finally, floating point simulations are shown to

illustrate the performance of the proposed algorithm.

More generally the algorithm exhibits the ability to identify poles located very close together and close to the unit circle.

REFERENCES

- [1] A Nehorai and D Starer. Adaptive Pole Estimation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(5):825-838, May 1990.
- [2] J J Shynk. Adaptive IIR Filtering. *IEEE ASSP Magazine*, pages 4 - 21, November 1989.
- [3] P L Feintuch. An Adaptive Recursive LMS Filter. *Proceedings of the IEEE*, pages 1622 - 1624, November 1976.
- [4] S D Conte and C de Boor. *Elementary Numerical Analysis: An Algorithmic Approach*. McGraw-Hill, New York, 1990.
- [5] M Harteneck and R W Stewart. Adaptive IIR Filtering Using QR Matrix Decomposition. Submitted to *IEEE Transactions on Signal Processing*, April 1995.
- [6] J G McWhirter M Harteneck, R W Stewart and I K Proudler. An Effective Approach to Adaptive IIR Filtering. *Proceedings of ICASSP, Atlanta*, 3:1634-1637, 1996.
- [7] M Harteneck and Stewart R W. A Fast Converging Algorithm for Landau's Output-Error Method. *Third International Symposium on Methods and Models in Automation and Robotics, MMAR 96, Poland*, 1996.
- [8] S Haykin. *Adaptive Filter Theory*. Prentice Hall Information and Systems Sciences Series, 3rd edition, 1996.