AD-A241 038

(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
ELECTE
OCT 0 3 1991
S D
D

THESIS

NATURAL RESONANCE EXTRACTION
AND ANNIHILATION FILTERING METHODS
FOR RADAR TARGET IDENTIFICATION

by

Timothy James Murphy

September, 1990

Thesis Advisor: Michael A. Morgan

Approved for public release; distribution is unlimited.

91-12188

Unclassified

security classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution/Availability of Report Approved for public release; distribution is unlimited.
2b Declassification/Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School / 6b Office Symbol (if applicable) 32	7a Name of Monitoring Organization Naval Postgraduate School
6c Address (city, state, and ZIP code) Monterey, CA 93943-5000	7b Address (city, state, and ZIP code) Monterey, CA 93943-5000
8a Name of Funding/Sponsoring Organization / 8b Office Symbol (if applicable)	9 Procurement Instrument Identification Number
8c Address (city, state, and ZIP code)	10 Source of Funding Numbers: Program Element No / Project No / Task No / Work Unit Accession No

11 Title (include security classification) NATURAL RESONANCE EXTRACTION AND ANNIHILATION FILTERING METHODS FOR RADAR TARGET IDENTIFICATION

12 Personal Author(s) Timothy J. Murphy

13a Type of Report	13b Time Covered	14 Date of Report (year, month, day)	15 Page Count
Master's Thesis	From To	September, 1990	120

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (continue on reverse if necessary and identify by block number)
Field	Group	Subgroup	Pole extraction, Cadzow-Solomon, Annihilation filters

19 Abstract (continue on reverse if necessary and identify by block number)

This thesis represents an initial attempt to demonstrate aspect independent target identification of complex radar targets using annihilation filters based on the natural resonances of the targets. The Cadzow-Solomon signal processing algorithm is tested to determine its suitability for the task of extracting the poles from complex targets to a degree of accuracy required for successful implementation of an annihilation filtering target identification system. This testing was conducted through the use of noise polluted synthetic data as well as measured transient scattering data from thin-wire and silver coated scale model aircraft targets. The testing revealed that the Cadzow-Solomon algorithm can return pole clusters at false pole locations when processing the scattered returns from complex targets. Properties of annihilation filters which may affect their ability to discriminate complex targets are examined.

20 Distribution/Availability of Abstract ☒ unclassified/unlimited ☐ same as report ☐ DTIC users	21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual M. A. Morgan	22b Telephone (include Area code) (408) 646-2677	22c Office Symbol EC/Mw

DD FORM 1473,84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete — security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

Natural Resonance Extraction and Annihilation Filtering Methods for Radar Target Identification

by

Timothy J. Murphy
Captain, United States Marine Corps
B.A.E.M., University of Minnesota

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author: Timothy J. Murphy

Approved by: Michael A. Morgan, Thesis Advisor

Ralph Hippenstiel, Second Reader

Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

This thesis represents an initial attempt to demonstrate aspect independent target identification of complex radar targets using annihilation filters based on the natural resonances of the targets. The Cadzow-Solomon signal processing algorithm is tested to determine its suitability for the task of extracting the poles from complex targets to a degree of accuracy required for successful implementation of an annihilation filtering target identification system. This testing was conducted through the use of noise polluted synthetic data as well as measured transient scattering data from thin-wire and silver coated scale model aircraft targets. The testing revealed that the Cadzow-Solomon algorithm can return pole clusters at false pole locations when processing the scattered returns from complex targets. Properties of annihilation filters which may affect their ability to discriminate complex targets are examined.

Accession For
NTIS CRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

TABLE OF CONTENTS

I. INTRODUCTION 1

A. THE PROBLEM 2

B. HISTORICAL BACKGROUND 3

II. POLE EXTRACTION ALGORITHMS 10

A. EARLY METHODS 10

1. Direct Minimization 11

2. Prony's Method 12

3. Kumaresan-Tufts Algorithm 13

a. System Model 14

b. Singular Value Decomposition 14

c. Bias Compensation 16

d. Earlier Results 18

B. CADZOW-SOLOMON ALGORITHM 18

1. System Model 18

2. Singular Value Decomposition 21

3. Bias Compensation 21

C. DETERMINING SYSTEM ORDER 22

1. Information Theoretic Criteria 23

2. Use Of AIC in Pole Extraction Algorithms 24

III. ALGORITHM TESTING 26

A. SYNTHETIC SIGNAL MODEL 27

B. SYNTHETIC SIGNAL TESTING RESULTS 29

C. THIN WIRE SIGNAL TESTING 52

1. Thin Wire Integral Equation Generated Data 53

2. Thin Wire Measured Data 57

IV. POLES OF SCALE MODEL AIRCRAFT 62

A. COMPLEXITY OF THE RESPONSE OF COMPLEX TARGETS 62

1. False Pole Clustering 63

2. A Look at the Spectra 68

B. ANNIHILATION FILTERING LIMITATIONS 73

V. SUMMARY AND CONCLUSIONS 77

A. SUMMARY 77

B. CONCLUSIONS AND RECOMMENDATIONS 78

APPENDIX A. S-PLANE POLES USED IN SYNTHETIC TESTING 81

APPENDIX B. ARMA COEFFICIENT GENERATOR 83

A. PROGRAM DESCRIPTION 83

B. PROGRAM LISTING 83

APPENDIX C. ARMA SIGNAL GENERATOR 86

A. PROGRAM DESCRIPTION 86

B. PROGRAM LISTING 86

APPENDIX D. THE CADZOW-SOLOMON POLE EXTRACTION ALGORITHM 89

A. PROGRAM DESCRIPTION 89

B. PROGRAM LISTING 89

APPENDIX E. DETERMINING SYSTEM ORDER 106

A. SUBROUTINE DESCRIPTION 106

B. SUBROUTINE LISTING 106

LIST OF REFERENCES 107

INITIAL DISTRIBUTION LIST 110

LIST OF FIGURES

Figure 1. Transient Electromagnetic Scattering 7

Figure 2. Incident Double Gaussian Pulse 30

Figure 3. Spectrum of Incident Pulse 31

Figure 4. Synthetic Medium Q Test Signal S/N=20 dB 32

Figure 5. Pole Extraction from High Q Synthetic Signal, S/N=90 dB 33

Figure 6. Pole Extraction from High Q Synthetic Signal, S/N=30 dB 34

Figure 7. Pole Extraction from High Q Synthetic Signal, S/N=20 dB 35

Figure 8. Pole Extraction from High Q Synthetic Signal, S/N=10 dB 36

Figure 9. Pole Extraction from High Q Synthetic Signal, S/N=7 dB 37

Figure 10. Pole Extraction from Medium Q Synthetic Signal, S/N=90 dB . . . 38

Figure 11. Pole Extraction from Medium Q Synthetic Signal, S/N=30 dB . . . 39

Figure 12. Pole Extraction from Medium Q Synthetic Signal, S/N=20 dB . . . 40

Figure 13. Pole Extraction from Medium Q Synthetic Signal, S/N=10 dB . . . 41

Figure 14. Pole Extraction from Medium Q Synthetic Signal, S/N=7 dB 42

Figure 15. Pole Extraction from Low Q Synthetic Signal, S/N=90 dB 43

Figure 16. Pole Extraction from Low Q Synthetic Signal, S/N=30 dB 44

Figure 17. Pole Extraction from Low Q Synthetic Signal, S/N=20 dB 45

Figure 18. Pole Extraction from Low Q Synthetic Signal, S/N=10 dB 46

Figure 19. Pole Extraction from Low Q Synthetic Signal, S/N=7 dB 47

Figure 20. Test of Pole Clustering Using Synthetic Signals 49

Figure 21. Magnitude of the FFT of One of the Synthetic Signals Used in the Cluster Test 51

Figure 22. Poles Extracted from TDIE Generated Thin Wire Response S/N=90 dB 54

Figure 23. First Quadrant View of Poles Extracted from TDIE Generated Thin Wire Response S/N=90 dB 55

Figure 24. Poles Extracted from TDIE Generated Thin Wire Response S/N=20 dB 58

Figure 25. First Quadrant View of Poles Extracted from TDIE Generated Thin Wire Response S/N=20 dB 59

Figure 26. Poles Extracted from Thin Wire Measured Responses for Various Angles of Incident Excitation 60

Figure 27. Clustering of Extracted poles for Target No. 1 65

Figure 28. Clustering of Extracted Poles for Target No. 1 Using New Processing Parameters. 66

Figure 29. Spectra of the Responses of a Single Scale Model Aircraft to Electromagnetic Excitation Incident at Various Angles 69

Figure 30. Spectra of the Responses of a Second Scale Model Aircraft to Electromagnetic Excitation Incident at Various Angles 71

Figure 31. Spectra of the Responses of a Thin Wire to Electromagnetic Excitation Incident at Various Angles 72

Figure 32. Spectrum of a Kill Pulse Used in the Attempt at Target Identification Using Annihilation Filtering. 74

Figure 33. Close up of the Lower Frequencies of the Spectrum of the Kill Pulse 75

I. INTRODUCTION

Radar has long been recognized as an effective tool for determining such information as a target's location in range and angle. In general however, radar systems are not capable of identifying the type of target being illuminated. In some situations this information may be as important as the target's location. A radar target scattering an incident electromagnetic wave can be considered to be a single input, single output, linear time invariant system. Because of this, the target can be described by a transfer function with poles and zeros. In his work at the Air Force Weapons Laboratory, Baum [1] showed through the development of the singularity expansion method (SEM) that these identifiable poles are determined by the target's geometry and composition. According to Moffatt and Mains [2], these poles are independent of the angle of incidence and polarization of the exciting waveform. Morgan [3] has shown theoretically that, after the last driven response is received from the target, its scattering response consists of a weighted superposition of natural resonances, each of which is independent of the incident excitation.

The use of these resonances for radar target identification was first proposed in 1974 by Moffatt and Mains [2]. Early attempts at demonstrating the feasibility of such a system were disappointing due to the high noise sensitivity of the signal processing algorithms. Recently, several signal processing algorithms have been used

to locate poles in a target's noisy response to a degree of accuracy which could make target identification with this technique viable [4]. The Cadzow-Solomon algorithm in particular, seems well suited for this application. This thesis represents an initial attempt to use radar returns taken from scale model aircraft in a scattering range to develop a database of target pole locations. These pole locations were then used to attempt to demonstrate the use of an annihilation filtering scheme for target identification. During this work, it was discovered that the Cadzow-Solomon algorithm may return pole clusters at false pole locations. Possible reasons for this are examined.

A. THE PROBLEM

Classifying radar targets based on their natural resonances requires the use of several signal processing functions. The first step is to identify the poles of each target class of interest. For simple targets such as spheres or thin wires, it may be possible to derive these poles analytically. However, for more complicated targets such as aircraft, the poles will more likely need to be extracted from actual measurements of the target's response to incident electromagnetic excitation. This information forms a database which will become the basis for target classification.

The second step in classifying a target of interest is to compare its poles with those contained in the database and to classify the target based on the closest match. One possible method for accomplishing this would be to extract the resonances of the target using the same signal processing techniques employed in the development of

the database and then to compare these poles directly against the database. However, because the signal processing algorithms for pole extraction tend to be computationally intensive, the time required to extract an unknown target's poles may exceed that which would make the system useful. It is possible to perform the database comparison without explicitly determining the target's poles. In the continuous time domain, the K-pulse method of Kennaugh [5] makes this possible, while in the discrete time domain the annihilation filter used by Dunnavin [6], Morgan and Dunnavin [7] and Chen *et. al.* [8] is used. An annihilation filter is an "all-zero" filter with its zeros located to correspond to the poles of an expected target. When the response of the corresponding target is applied to the filter, the energy corresponding to the target's poles is canceled and the filter's output consists only of energy due to the driven portion of the response and any noise present in the system.

A system to classify radar targets by their natural resonances using these techniques would require a separate annihilation filter corresponding to each target classification of interest. In using the system, the response of a target of concern would be fed to each of the filters concurrently. In its simplest form, the filter exhibiting the lowest output energy would be selected as that corresponding to the classification of the unknown target.

B. HISTORICAL BACKGROUND

The concept of radar target classification through the use of natural resonances is based on assumptions about the nature of a target's return signal to a radar pulse. In 1971, Baum [1] developed the SEM, wherein a target's impulse response induced currents are considered to be the sum of natural modes. A pulse illuminating a target induces currents on the target's surface. In the time domain these currents occur in modes of the form, $\bar{J}_n(\bar{r})\exp(s_n t)$, where s_n represents a natural resonance of the target. These natural resonances occur in the left-hand portion of the s-plane. Since the time domain currents are real, they must occur in complex conjugate pairs. The natural resonances can be represented as

$$s_n = \sigma_n + j\omega_n \tag{1}$$

where σ_n is the damping rate in Nepers/sec and ω_n is the frequency in radians/sec. The location of these poles in the s-plane is determined by the requirement to satisfy the boundary conditions which, in turn, are determined by the physical properties of the target including its shape, size, and composition.

Although an infinite number of these resonances exist in any object, only a finite set of them will be measurably excited by an incident electromagnetic wave of finite bandwidth. Because certain resonances are strongly associated with certain sections of a target's structure, the aspect at which the target is illuminated will affect the amplitude and phase at which each of these current modes are excited. However, the frequency and damping rate of each mode is not determined by the aspect at

which the target is illuminated or at which it radiates with respect to the receiving antenna. In 1974, Moffatt and Mains [2] first proposed that the extraction of these resonances from a target's response could be used for target identification. This proposal built on earlier work by Kennaugh and Moffatt [9], who observed that a target could be characterized by its impulse response which would include the exponentially damped terms of the form discussed above.

Unfortunately, the scattered field response of a target cannot be represented simply as a sum of complex exponentials occurring at the resonance frequencies. Early attempts at target classification based on natural resonances were disappointing not only due to the noise sensitivity of the signal processing algorithms, but also due to the use of an incomplete signal model. The current induced on the surface of a highly conducting target illuminated by an electromagnetic field must satisfy the magnetic field integral equation (MFIE) [10]

$$\bar{J}(\bar{r},t)=2\hat{n}\times\bar{H}^{i}(\bar{r},t)+\int_{S_{pv}}\int\tilde{K}(\bar{r},\bar{r}',t)\bar{J}(\bar{r}',t-\frac{|\bar{r}-\bar{r}'|}{c})dS \tag{2}$$

where $\bar{J}$ is the surface current density, $\hat{n}$ is the outward unit vector normal to the surface of the object, $\bar{H}^{i}$ is the incident magnetic field, and $\tilde{K}$ is a Green's function dyadic. The principal-value (PV) type integral excludes the point $r = r'$. The cross product term represents the physical optics portion of the induced current while the surface integral term represents the "feedback" currents to each point on the scatterer due to the induced currents at all other points on the object. With $\bar{H}^{i} = 0$ in (2), it

is possible to represent the response as a sum of natural modes of the form $\bar{J}_n(\bar{r})\exp(s_n t)$, where s_n has the form of (1). Figure 1 illustrates the relationships in (2).

The scattering response of a target illuminated by a pulsed radar signal will consist of an early time driven response, caused in part by induced currents driven by $\bar{H}^i \neq 0$, followed by a late-time natural mode response. The early time response can be envisioned as the scattering of the target as the radar pulse is passing over it, while the late time response is that due to the decaying currents present once the pulse is no longer directly illuminating the target. As the pulse moves across the target, the surface current consists of the physical optics term added to the Green's function integral contribution from all points previously illuminated by the pulse. Because of causality, there is no induced current at points on the scatterer ahead of the incident wavefront. For a monostatic radar the transition from early to late time will occur at $\Delta t = T + 2D/c$ seconds after the leading edge of the scattered pulse arrives back at the radar antenna. Here T is the pulse duration, D is the target's dimension along the direction of wave propagation, and c is the speed of light.

In the far-field, the back-scattered response of the target due to the surface currents induced by the incident pulse takes the form

$$\bar{H}^s(-r\hat{p},t) = \frac{1}{4\pi c r}\frac{\partial}{\partial t}\iint_S \hat{p}\times\bar{J}(\bar{r}',t-|\bar{r}-\bar{r}'|/c)dS' \qquad (3)$$

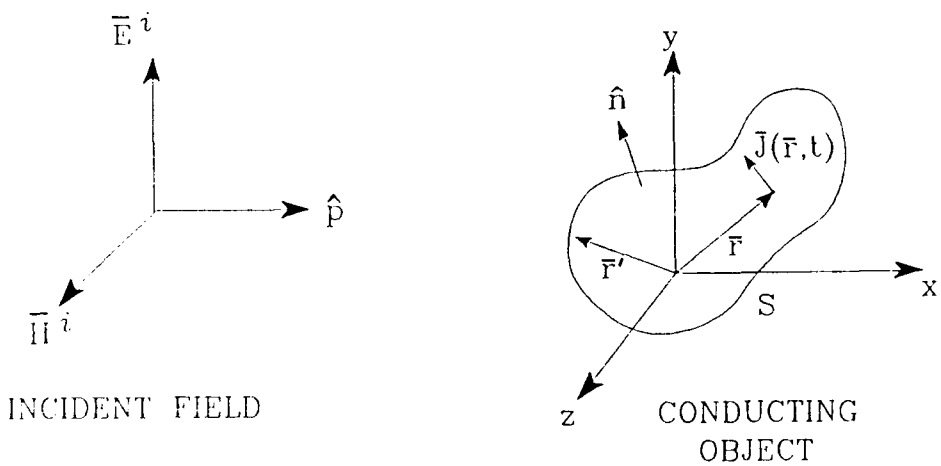

Figure 1. Transient Electromagnetic Scattering

where $\hat{p}$ is the unit vector in the direction of the plane wave's propagation. This equation gives the value of the field at a point in the far-field by integrating the current at each point on the target's surface. To find the back-scattered far-field, substitute the currents given by (2) into (3):

$$\bar{H}^s(-r\hat{p},t) = u(t-r/c)\left\{\bar{H}_{po}(-r\hat{p},t)+\sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \bar{H}_n(-r\hat{p},t)\exp(s_n t)\right\}. \tag{4}$$

The first term in (4) describes that portion of the field generated by the $2\hat{n}\times\bar{H}^i$ term in (2), the physical optics term. The second term represents the contribution of the feedback currents in the Green's function integral to the back-scattered field. For any particular point on the object, the infinite number of paths connecting it to all other points on the object is the same as for any other point on the object. These paths are unique to the geometry of the object and correspond to the paths taken by currents as they feedback to the particular point. The Green's function kernel accounts for this in (2). Thus, this term in (2), and the resultant field in (4) are unique to the structure of the target.

Additional insight is provided by the singularity expansion method developed by Baum [1]. SEM uses the singularities of the response in the complex frequency (s) plane as a method for viewing a scatterer's response. Early attempts at target identification using natural resonances employed a SEM "class 1" expansion as a model for the target's scattering response. A "class 1" expansion in the singularity

expansion method describes the response as a sum of fixed weighted terms involving the resonant frequencies. Morgan [3] has shown that this is an accurate model only in the late time portion of the response. In the early time the response contains the physical optics term and is further complicated by the fact that the surface area over which the integration in (3) must be conducted is continually changing as the pulse moves across the target. Thus the coefficients $\bar{H}_n$ in (4) are time varying. In the late time portion of the response, after the pulse has passed completely over the object, the surface integral in (3) must be conducted over the entire surface of the object, and the coefficients in (4) are constant. A "class 2" form of the SEM expansion can take into account the time varying coefficients. The early time response of a scatterer is therefore composed of a physical optics term and a class 2 SEM expansion.

II. POLE EXTRACTION ALGORITHMS

This chapter begins with a brief discussion of several of the signal processing algorithms which have been important in the history of radar target identification using natural resonances. It then continues with a more in-depth look at the Kumaresan-Tufts and the Cadzow-Solomon algorithms.

A. EARLY METHODS

The first problem to be solved in radar target identification using natural resonances is that of accurately locating the target's poles. A degree of precision is necessary in the determination of these locations due to the possibility of different targets having poles which are relatively close, causing difficulties in selecting between the two targets in an identification scheme. Examination of a signal's spectrum is typically done using the fast Fourier transform (FFT) due to this algorithm's efficiency and reasonable results for a large class of signal processes. However, due to one of its main limitations as given by Kay and Maple [11], the FFT is inappropriate for use in locating radar target natural resonances. The frequency resolution in Hertz is roughly the reciprocal of the time interval in seconds over which the sampled data is available. A radar return pulse from the 1/72 scale model aircraft used in this thesis cannot typically be expected to be more than 5ns in duration giving a frequency resolution of no better than 200MHz and typically is much worse. Full-size radar targets will have responses of several μs, but it would

still be advantageous to achieve a greater frequency resolution than the FFT can provide. A second limitation is that the FFT is used only to determine the frequencies contained within a signal; for target identification schemes a complete pole location in either the s-plane or the z-plane is required.

1. Direct Minimization

As presented by Morgan [10], the late-time response of a radar target can be represented as a sum of damped sinusoids oscillating at the target's resonant frequencies.

$$\hat{y}(t) = \sum_{i=1}^{\infty} A_i e^{\sigma_i t} \cos(\omega_i t + \theta_i) \tag{5}$$

In the digital domain, this representation becomes:

$$\hat{y}(n\Delta t) = \hat{y}_n = \sum_{i=1}^{N} A_i e^{\sigma_i n\Delta t} \cos(\omega_i n\Delta t + \theta_i) \tag{6}$$

Each of the sinusoids in the representation is defined by its amplitude, A_i, damping rate, σ_i, frequency, ω_i, and phase, θ_i. These parameters can be adjusted to minimize the mean square error between the modeled signal $\hat{y}_n$ and the actual received discrete signal y_n, where the squared error at each point is $e_n^2=(y_n-\hat{y}_n)^2$. Use of this technique is computionally inefficient due to the high degrees of dimensionality and non-linearity; however, Chong was able to use it to process synthetically generated data down to 15 dB signal-to-noise ratios [12].

2. Prony's Method

Prony's method is a technique for modeling data of equally spaced samples by a linear combination of exponentials. This technique can be used to model the late-time response of a radar target. This autoregressive model is given by

$$y_n = \sum_{i=1}^{K_D} a_i y_{n-i} \qquad for\ n = 0, \cdots, N-1 \tag{7}$$

Here y_n is the n-th sample of the received signal and K_D is the system order. Taking the z transform gives

$$z^{K_D} - a_1 z^{K_D-1} - a_2 z^{K_D-2} - \cdots - a_{K_D} = 0 \tag{8}$$

The roots of this polynomial are the poles of the system model in the z-plane. By first finding the coefficients a_i, these poles can be located. These coefficients can be found by using equation (7) in a system of M equations:

$$\begin{bmatrix} y_0 & \cdots & y_{K_D-1} \\ \vdots & & \vdots \\ y_{M-1} & \cdots & Y_{K_D+M-2} \end{bmatrix} \begin{bmatrix} a_{K_D} \\ \vdots \\ a_i \end{bmatrix} = \begin{bmatrix} y_{K_D} \\ \vdots \\ y_{K_D+M-1} \end{bmatrix} \tag{9}$$

The original Prony's method required that the data matrix be exactly determined with $M = K_D =$ the system order. For resonance based target

identification this is not likely to be possible since the system order will not be known *a priori*. The extended Prony method [11] seeks an approximate fit with K_D exponentials by setting $M > K_D$ and minimizing the squared error. With this extension the technique can be used with noisy data. In practice however, the noise tends to perturb the extracted pole positions and one is still left with the problem of not knowing the system order K_D. If the order is estimated below the actual value, poles will be lost and those extracted will not be accurately located. Overestimating generates spurious poles due to the noise with no means of separating them from the true poles.

3. Kumaresan-Tufts Algorithm

Kumaresan and Tufts modified Prony's method in an attempt to alleviate some of its shortcomings, including its sensitivity to noise and the need for *a priori* knowledge of the system order. The first of these modifications was to deliberately overestimate the system order. This provides the model with the flexibility to compensate for the errors caused by noise. Second, singular value decomposition (SVD) was used to partially alleviate the ill-conditioning of the data matrix. Also, the causality of the system is used to separate the computed poles into orthogonal signal and noise spaces. Kumaresan has demonstrated [13] that the use of singular value decomposition in conjunction with a non-causal system model tends to force the extra poles of an overestimated signal inside the unit circle on the z-plane, with the signal poles remaining outside.

a. System Model

The system model used by Kumaresan and Tufts is an autoregressive type model and is therefore applicable only to the late time portion of a radar return signal. The non-causal model is given by

$$y_n = \sum_{i=1}^{K_D} a_i y_{n+K_D+1-i} \tag{10}$$

Here K_D is greater than the actual system order. In matrix form, with M such equations, this becomes

$$\begin{bmatrix} y_1 & \cdots & y_{K_D} \\ \vdots & & \vdots \\ y_M & \cdots & y_{K_D+M-1} \end{bmatrix} \begin{bmatrix} a_{K_D} \\ \vdots \\ a_1 \end{bmatrix} = \begin{bmatrix} y_0 \\ \vdots \\ y_{M-1} \end{bmatrix} \tag{11}$$

Or, in matrix notation,

$$D_y \cdot a = y \tag{12}$$

Here the coefficients a_i correspond to those in (8) in that they define the polynomial the roots of which are the z-plane poles.

b. Singular Value Decomposition

The use of singular value decomposition is at the heart of the Kumaresan-Tufts pole extraction method. Its use allows solution of the system of equations in (11) despite ill-conditioning of the data matrix, as well as separating the

signal poles from the noise poles. The following discussion of this technique is taken principally from Golub [14].

Singular value decomposition factors the $M \times K_D$ data matrix D_y into the product of three matrices:

$$D_y = U \Sigma V^T \tag{13}$$

The columns of U $(M \times M)$ are eigenvectors of $D_y D_y^T$ and the columns of V^T $(K_D x K_D)$ are eigenvectors of $D_y^T D_y$. If the data matrix has rank r, then the $M \times K_D$ matrix Σ will consist of r singular values on its diagonal, the roots of which are the eigenvalues of both $D_y^T D_y$ and $D_y D_y^T$. When used with an over-determined system the diagonal of the Σ matrix splits into a signal subspace

$$u_1, u_2, \cdots, u_{K_D'} \quad \textit{with eigenvalues} \quad \eta_1 \geq \eta_2 \geq \cdots \geq \eta_{K_D'} \tag{14}$$

and a noise subspace

$$u_{K_{D+1}'}, u_{K_{D+2}'}, \cdots, u_M \quad \textit{with eigenvalues} \quad \eta_{K_{D+1}'} \geq \eta_{K_{D+2}'} \geq \cdots \geq \eta_M. \tag{15}$$

With no noise present, all the eigenvalues in (15) are zero and the rank of Σ reduces to K_D', the actual system order.

In order to solve the system of equations in (12), the pseudoinverse of D_y can be found as

$$D_y^+ = V\Sigma^+ U^T \tag{16}$$

where Σ^+ is a $K_D \times M$ matrix whose singular values are the reciprocals of those in the Σ matrix. The coefficient vector $\boldsymbol{a}$, of minimum Euclidian norm, is then given by

$$\boldsymbol{a} = D_y^+ \boldsymbol{y} \tag{17}$$

The coefficient vector $\boldsymbol{a}$ in (17) provides the best possible (least-squares) solution to (12).

c. *Bias Compensation*

Kumaresan and Tufts [15] noticed the partioning in (14) and (15) and modified the algorithm in an attempt to reduce the effects of noise. Kumaresan and Tufts averaged the values of the singular values spanning the noise space (15) and then subtracted this value from each of the singular values spanning the signal space (14). The noise singular values were then set to zero and the new Σ matrix which resulted was used in computing the pseudoinverse. Although no analytical justification for this technique was provided, it dramatically reduced the effects of noise and allowed an increase in achievable frequency resolution.

A second scheme for bias compensation was derived by Norton [16]. The noisy data matrix can be described by $D_y = S + N$, where N is composed of the wide-sense stationary white noise process v_i, and can be represented as

$$N = \begin{bmatrix} v_i & \cdots & v_{K_D} \\ \vdots & & \vdots \\ v_M & \cdots & v_{M+K_D} \end{bmatrix} \tag{18}$$

The expected value of $D_y D_y^T$ can be determined as

$$D_y D_y^T = E[(S+N)(S+N)^T] = E[SS^T] + E[SN^T] + E[NS^T] + E[NN^T] \tag{19}$$

With the assumption of wide-sense stationary white noise, the noise has zero mean and the two cross product terms are zero. Also, $E[NN^T]=\sigma_v^T I$ and, since S is deterministic, $E[SS^T]=SS^T$. The expected value of $D_y D_y^T$ can then be written

$$E[D_y D_y^T] = SS^T + \sigma_v^2 I \tag{20}$$

According to the eigenvalue shifting theorem, if the eigenvalues of SS^T are λ_i, the eigenvalues of the matrix $E[D_y D_y^T]$ are $\lambda_i + \sigma_v^2$. Therefore, in the mean, the eigenvalues of $D_y D_y^T$ are increased by the variance of the noise. This led Norton to propose a bias compensation method by squaring the singular values of the Σ matrix which correspond to the power of the noise, and then take the average in order to obtain an estimate of the noise variance σ_v^2. The noise singular values are then set to zero. The first K_D singular values of the Σ matrix correspond to the system poles and they are next squared and the estimate of the noise variance is subtracted from each. The square root of this result is then used as the new set of singular values

corresponding to the system poles. As in the Kumaresan-Tufts bias compensation scheme, the pseudoinverse can then be found in the normal manner.

d. Earlier Results

The Kumaresan-Tufts algorithm was tested using synthetic data, thin wire integral equation data, thin wire scattering measured data and scale model aircraft measured data by Larison [4]. He was able to demonstrate reasonable pole extraction performance for low frequency poles. Because the algorithm can only be used in the late time portion of a target's response, the algorithm had difficulty extracting higher frequency poles with their corresponding higher damping rate. Larison's results also suggest that the two most critical parameters in using the algorithm are selecting the appropriate starting point at which to begin processing the data sequence as well as selecting the appropriate system order so that the bias compensation scheme will provide the best possible results.

B. CADZOW-SOLOMON ALGORITHM

The Cadzow-Solomon algorithm [17] has shown a greater degree of promise for use in natural resonance extraction from radar target return signals than any of the earlier described algorithms and was used exclusively for the construction of a target library in this thesis. This section describes the algorithm.

1. System Model

The Cadzow-Solomon algorithm is based on an autoregressive moving-average (ARMA) type model and, as such, requires knowledge of both the system's

input and output. It is capable of estimating both the poles and zeros of the system. The governing equation is given by

$$y_n = \sum_{i=1}^{K_D} a_i y_{n-i} + \sum_{i=0}^{K_N} b_i x_{n-i} \tag{21}$$

Here K_D is the order of the denominator of the system's transfer function (poles), K_N is the order of the numerator (zeros), and x_n is the exciting waveform. A set of M such equations in matrix form would be represented by

$$\begin{bmatrix} y_0 & \cdots & y_{K_D-1} & \vdots & x_0 & \cdots & x_{K_N} \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ y_{M-1} & \cdots & y_{K_D+M-2} & \vdots & x_{M-1} & \cdots & x_{K_N+M-1} \end{bmatrix} \begin{bmatrix} a_{K_D} \\ \cdot \\ \cdot \\ \cdot \\ a_1 \\ \cdots \\ b_{K_N} \\ \cdot \\ \cdot \\ \cdot \\ a_0 \end{bmatrix} = \begin{bmatrix} y_{K_D} \\ \cdot \\ \cdot \\ \cdot \\ y_{K_D+M-1} \end{bmatrix} \tag{22}$$

Following the technique used in the Kumaresan-Tufts algorithm, it is possible to overestimate the order of both the zeros and the poles in the system in order to provide a degree of noise immunity for the input and output waveforms respectively. If the actual system order is K_D' ($\leq K_D$) and the actual order of the numerator in the system transfer function is K_N' ($\leq K_N$), then a necessary and sufficient condition for the model equations (22) to have a solution is for the data matrix to have a rank of $K_D + K_N + 1$. Cadzow and Solomon state that this condition will be ensured by

taking $n = 0$ to correspond to that time instant at which the excitation first becomes nonzero for a transient type excitation.

Equation (21) can be modified for backward prediction [16]

$$y_n = \sum_{i=1}^{K_D} a_i y_{K_D+n-i+1} + \sum_{i=0}^{K_N} b_i x_{n-i} \tag{23}$$

In matrix form this is

$$\begin{bmatrix} y_{K_N+1} & \cdots & y_{K_N+K_D} & \vdots & x_0 & \cdots & x_{K_N} \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ \cdot & & \cdot & \vdots & \cdot & & \cdot \\ y_{K_N+M} & \cdots & y_{K_N+K_D+M-1} & \vdots & x_{M-1} & \cdots & x_{K_N+M-1} \end{bmatrix} \begin{bmatrix} a_{K_D} \\ \cdot \\ \cdot \\ \cdot \\ a_1 \\ \cdots \\ b_{K_N} \\ \cdot \\ \cdot \\ \cdot \\ a_0 \end{bmatrix} = \begin{bmatrix} y_{K_D} \\ \cdot \\ \cdot \\ \cdot \\ y_{K_D+M-1} \end{bmatrix} \tag{24}$$

In matrix notation

$$[D_{yx}] \begin{bmatrix} a \\ \cdots \\ b \end{bmatrix} = y \quad \textit{where } [D_{yx}] = [D_y \vdots D_x] \tag{25}$$

2. Singular Value Decomposition

The use of singular value decomposition to solve the system of equations (24) again provides the minimum-norm solution. Its use with the non-causal model again separates the noise poles from the signal poles across the unit circle.

3. Bias Compensation

Cadzow and Solomon have shown that if the numerator and denominator orders K_N' and K_D' are overestimated as K_N and K_D, singular value decomposition will return at least $s=1+\min\{K_D-K_D', K_N-K_N'\}$ of the eigenvalues with a value of zero for noiseless data. In the noisy case, these eigenvectors can be expected to take on some low values which may allow them to be distinguished from the preceding eigenvalues. However, because of the composite form of the data matrix, and because the eigenvalues are returned in standard nondecreasing order, it does not appear possible to directly partion the eigenvalues in the form of (14) and (15) saying, for example, that a certain subset corresponds to the signal poles, another to the signal zeros, and the remaining to the extraneous poles and zeros. This is the first drawback of the Cadzow-Solomon algorithm in terms of bias compensation. For a compensation scheme such as that used by Kumaresan and Tufts [15], the number of singular values that should be set to zero cannot be readily determined.

Another drawback of the composite data matrix in terms of bias compensation is that the additive noise is different for the input and output data. Norton's eigenvalue compensation scheme is theoretically valid only if the input and output noise variances are equal [16]. In constructing the target library in this thesis the

input data is not noisy. The same is not true for the output data. Nonetheless, Larison [4] made the assumption of equal noise variance and processed data using eigenvalue compensation with the Cadzow-Solomon algorithm achieving good results. Testing of the algorithms prior to construction of the target library (Chap. III) confirmed the consistently superior results obtained making this assumption and using eigenvalue compensation. For this reason, eigenvalue compensation was used throughout this thesis in attempts at the construction of a target library.

C. DETERMINING SYSTEM ORDER

The use of either of the bias compensation schemes presented requires an estimation of the order of the system. Larison [4] used a trial and error technique, systematically varying his estimate of the order and observing the effect of a particular estimate on the arrangement of the poles. As the correct order is approached the noise poles assume an orderly, even arrangement about the unit circle. Algorithms such as the information theoretic criteria by Akaike [18] have been proposed which can determine the system order by a statistical examination of the eigenvalues returned by singular value decomposition. These algorithms look for the partioning which is present as in (14) and (15). This section will examine the algorithm behind the Akaike information theoretic criteria (AIC) and then examine considerations for its use with algorithms for radar target identification.

1. Information Theoretic Criteria

The earlier discussion on singular value decomposition within the Kumaresan-Tufts algorithm explained the properties of the eigenvalues $\eta_1 \geq \eta_2 \geq \ldots \geq \eta_M$ of the $M \times K_D$ matrix Σ: they are nonnegative and the largest ones $\eta_1, \eta_2, \ldots, \eta_{K_D'}$ ($K_D' < K_D$) have corresponding eigenvectors which span the signal subspace. The remaining eigenvalues and their corresponding eigenvectors represent the noise subspace. The information theoretic criteria seeks to estimate the integer rank of the signal subspace. This value can then be used when applying either of the bias compensation schemes. The following discussion of the criteria follows that in Aurand [19].

Singular value decomposition returns the Σ matrix which consists of K_D singular values on its diagonal, which are the roots of the eigenvalues of interest. Then for an index $p = 0,1,\ldots,K_D\text{-}1$, the information theoretic criteria is calculated as

$$AIC(p) = LR(p) + p(2K_D - p) \tag{26}$$

where $LR(p)$ is a log-likelihood ratio of a representation of the correlation matrix. For a given size data matrix with the Kumaresan-Tufts algorithm the total number of data points processed, N, is $M+K_D\text{-}1$ and $LR(p)$ is given by

$$LR(p) = -\ln\left[\frac{\prod_{i=p+1}^{K_D} \eta_i^{1/(K_D-p)}}{\frac{1}{K_D-p}\sum_{i=p+1}^{K_D} \eta_i}\right]^{(K_D-p)N} \qquad p=0,1,\ldots,K_D-1 \ . \tag{27}$$

The best estimate of the rank of the signal subspace, K_D', is the value of p for which the AIC criterion is minimized. Aurand went on to simplify the expressions in order to facilitate computer coding arriving at

$$AIC(p) = (K_D-p)N \ln[\frac{1}{K_D-p}\sum_{i=p+1}^{K_D} \eta_i] - N \ln[\prod_{i=p+1}^{K_D} \eta_i] + p(2K_D-p) \tag{28}$$

for $p = 0,1,\ldots,K_D-1$.

The first term in the above equation is the logarithm of the arithmetic mean of the $(M-p)$ smallest eigenvalues. The second term is the logarithm of the geometric mean of these same eigenvalues. As the smallest eigenvalues become more uniform, the ratio of the geometric mean to the arithmetic mean approaches unity and the sum of the first two terms, $LR(p)$, approaches zero.

2. Use Of AIC in Pole Extraction Algorithms

The AIC algorithm is an extremely effective means of estimating system order when using an algorithm such as Kumaresan-Tufts which uses overestimation to reduce the effects of noise and is based on an autoregressive model. The partioning represented by (14) and (15) seems readily susceptible to exploitation by

this type of algorithm. Aurand [Ref. 19: Chap. V] states that order estimators of this type are most effective when the algorithm is run several times and the value of the system order which recurs most frequently is selected as the most reliable estimate. In using a pole finding algorithm to determine the natural resonances of a complex target such as a scale model aircraft, it is usually necessary to run the algorithm several times before the poles are accurately located due to the need to determine the optimum values of parameters such as the starting point and the row and column dimensions of the data matrix. An algorithm such as AIC reduces the number of parameters which must be determined since the best value of the system order will tend to surface as the remaining parameters are being identified.

As discussed earlier, use of singular value decomposition with the Cadzow-Solomon algorithm does not result in a Σ matrix which consists of singular values which are as easily partioned as is the case with the Kumaresan-Tufts algorithm. As such, it is not as susceptible to analysis by algorithms such as AIC for determining the correct order of the system. Nonetheless it was found to be helpful at determining an upper end for an estimated system order for use in bias compensation. It seems likely that use of this algorithm, which is intended for use with data governed by an AR type model, is effective at determining the location of the $1+\min\{K_D-K_D', K_N-K_N'\}$ break in the eigenvalues returned following singular value decomposition when using the Cadzow-Solomon algorithm. Kay [20] discusses a modification to the AIC algorithm for use with ARMA systems, but the existence of this algorithm was discovered to late for incorporation into this thesis.

III. ALGORITHM TESTING

The original objective of this thesis was the construction of a library of poles extracted from scale model aircraft. The attempt at this goal took place in a two phase process. The first phase consisted of testing of the Cadzow-Solomon pole extraction algorithm using synthetic data as well as data from a simple target in the form of a thin wire. This phase was necessary in order to gain proficiency in the use of the algorithm prior to attempting to extract the poles from a complex target. It also provided the opportunity to gain an appreciation for what the strengths and weaknesses of the algorithm would be in extracting these poles. As such, the synthetic data testing phase of this work was more extensive in terms of its attempt to simulate the conditions which could be expected from the response of a complex target than had been the case in previous works. The second phase of the process was to extract the poles from the measured scattering response of the scale model aircraft. As discussed in the previous chapter, the extracted poles are a least squares solution to the governing equations. Noise, unknowns such as the actual system order, and the fact that certain poles may be more or less excited depending on the angle of incidence of the exciting waveform all tend to cause slight variations in the location of the pole as extracted by the algorithm. Repeated runs of the code tend to display clusters of extracted poles. These clusters were expected to correspond to the true poles of the target. For simple targets and simple synthetic data, this

appears to be the case as will be explained in this chapter. For more complex targets, such as the scale model aircraft targets used in this thesis, this was found to not necessarily be true, as will be explained in the next chapter.

A. SYNTHETIC SIGNAL MODEL

The Cadzow-Solomon algorithm is based on an autoregressive model. As discussed in Chapter I, a portion of the scattered field of a real target is due directly to the driving incident wave and the remainder of the field is due to feedback currents occurring on the surface of the target. Thus an autoregressive moving average (ARMA) structure is appropriate for modeling this phenomenon. The generating equation for the synthetic signals used for testing the algorithm is given by

$$y_n = \sum_{i=1}^{N} a_i y_{n-i} + \sum_{i=0}^{L} b_i x_{n-i} \tag{29}$$

where x_n is the digitized exciting waveform, y_n is the scatterer response, a_i are the coefficients which correspond to the scatterer's poles in the fashion of equations (7) and (8). Similarly the coefficients b_i correspond to the zeros of the transfer function describing the scatterer. N is the order of the denominator of the system's transfer function and L is the order of the numerator.

Three separate signals were generated, each based on ten pole pairs covering a frequency range from 1-10 GHz. Each of the three base signals represented a

different level of damping (Q factor). The corresponding pole pairs in each of the signals were related in the s-plane by

$$\frac{\sigma_n}{\omega_n} = \frac{1}{2\pi}[\ln(k_n)] \tag{30}$$

where $k_1 = 0.5$ for the low Q or highly damped case, $k_2 = 0.7$ for the medium Q case and $k_3 = 0.8$ for the high Q case. Appendix A contains a listing of the s-plane poles used in the synthetic signals. The line of thought followed in creating these signals was to have the medium Q case approximate as closely as possible the expected level of damping from the actual measured scattering responses of the scale model aircraft targets.

The s-plane poles were then converted to z-plane poles based on 1024 samples over 20 ns. Multiplying terms $(z\text{-}z_1)(z\text{-}z_2)...(z\text{-}z_{20})$ where z_i is the i-th z-plane pole results in a polynomial of the form of (8). The coefficients of this polynomial are the coefficients a_i in (29). The coefficients b_i used to create the synthetic signals, were arrived at through an inverse partial fraction expansion using the $1/(z\text{-}z_i)$ terms. They were all generated using an amplitude value of 1.0 and a phase difference of 0.0 for each of the signal poles. Program listings for the coefficient generator and the recursive signal generator appear in the appendices.

The exciting incident waveform chosen to generate these test signals was the double Gaussian pulse depicted in Figure 2. This pulse is a wide Gaussian pulse with

a ten percent width of 0.3 nanoseconds subtracted from a narrow Gaussian pulse with a ten percent width of 0.15 nanoseconds, resulting in a bandwidth covering approximately 1-10 GHz. The spectral content of the double Gaussian pulse is illustrated in Figure 3.

Once the three base test signals were generated, they were each run through a additive noise program, generating signals with signal-to-noise ratios (measured in terms of signal energy to noise energy) of 90 dB, 30 dB, 20 dB, 10 dB, and 7 dB. A total of fifteen test signals were thereby generated. Figure 4 illustrates a typical synthetic test signal generated through this process.

B. SYNTHETIC SIGNAL TESTING RESULTS

Using the Cadzow-Solomon pole extraction program written by Larison [4], with modifications to allow use of the AIC algorithm, the poles were extracted from each of the fifteen test signals. In each case the input parameters to the program were varied in an attempt to achieve a minimum error distance between the true pole and the extracted poles. Figures 5-19 illustrate the results of this effort. Table I lists the average error distance between the true pole and the nearest extracted pole, measured on the z-plane for each of the fifteen synthetic signals tested.

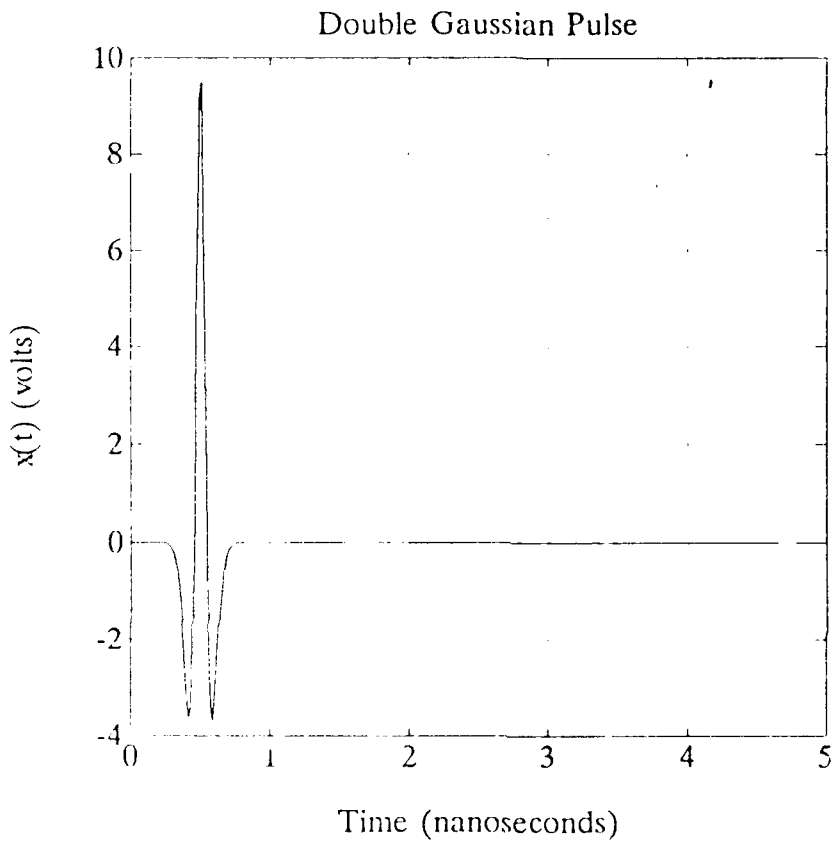

Figure 2. Incident Double Gaussian Pulse

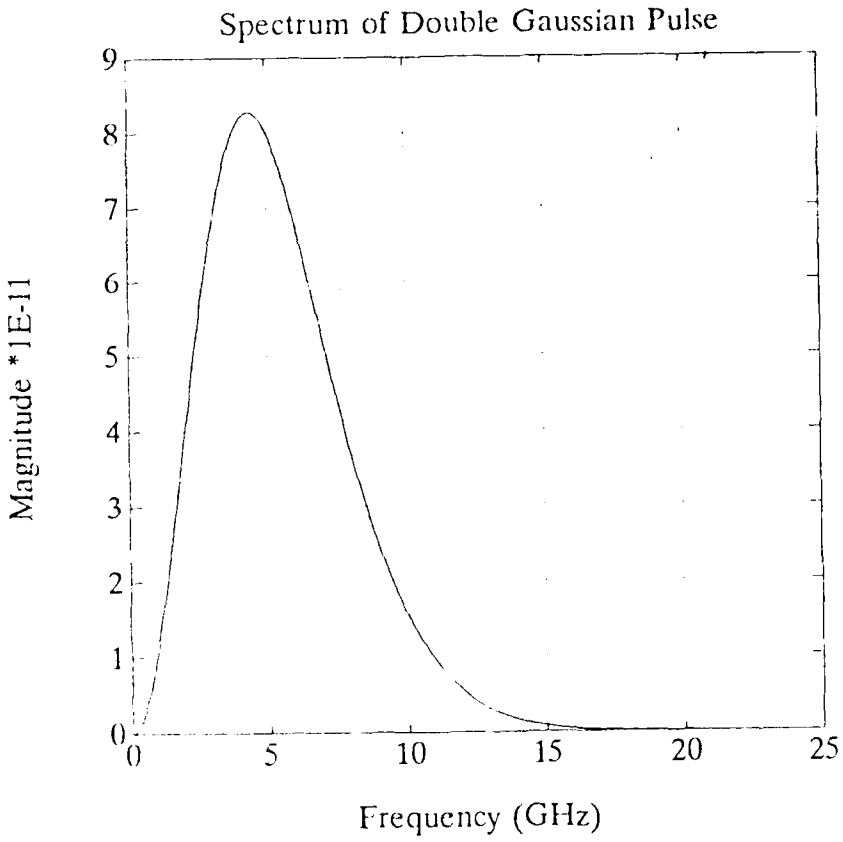

Figure 3. Spectrum of Incident Pulse

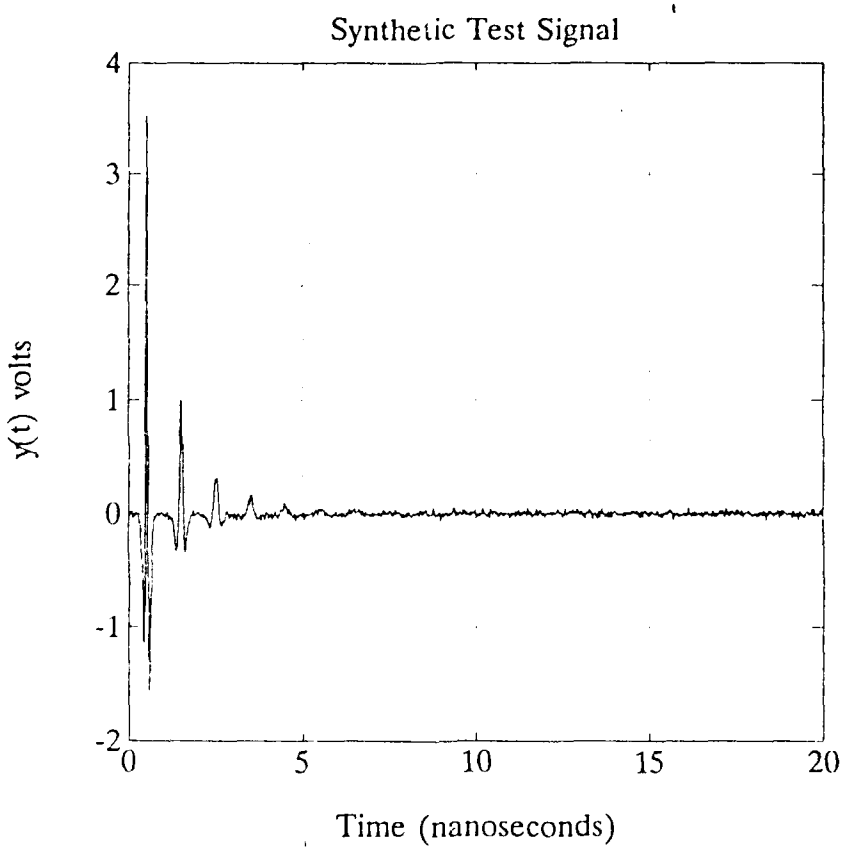

Figure 4. Synthetic Medium Q Test Signal S/N=20 dB

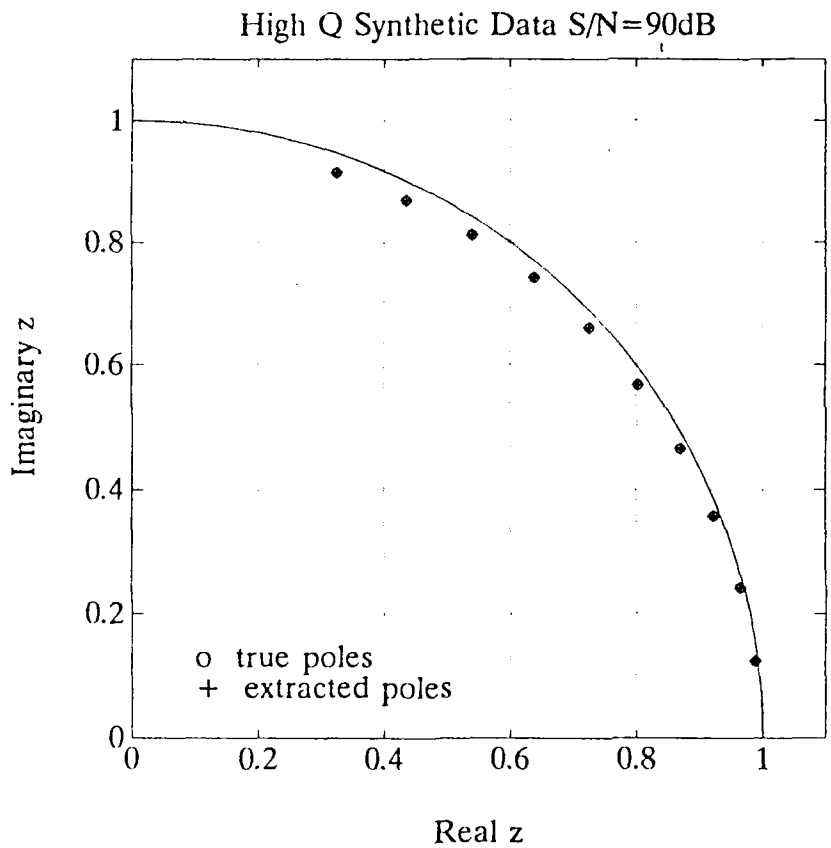

Figure 5. Pole Extraction from High Q Synthetic Signal, S/N=90 dB

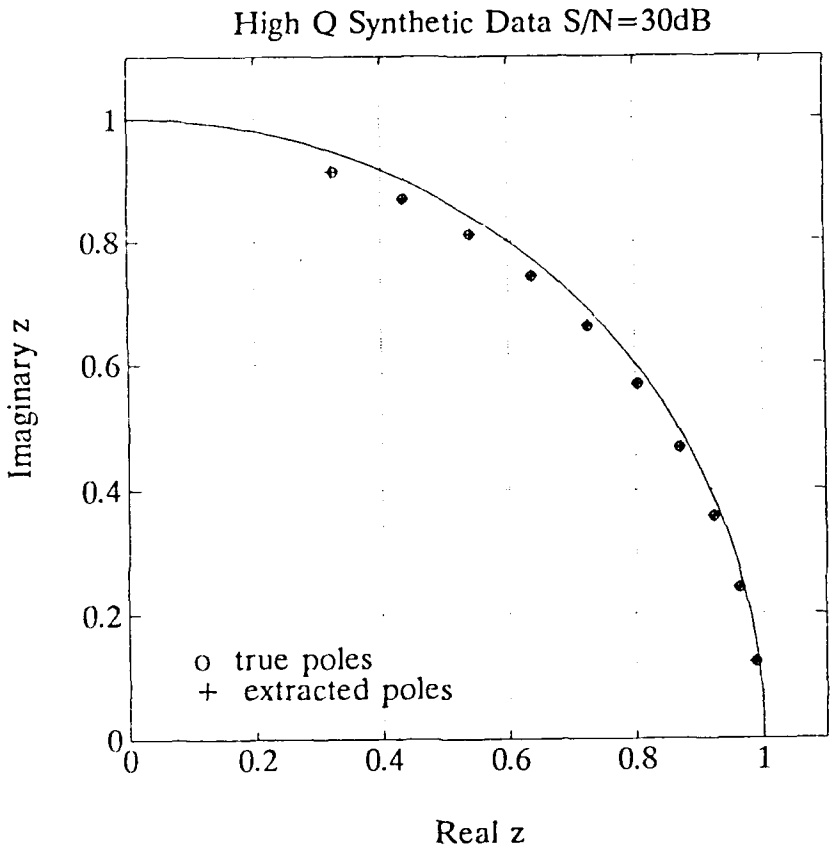

Figure 6. Pole Extraction from High Q Synthetic Signal, S/N=30 dB

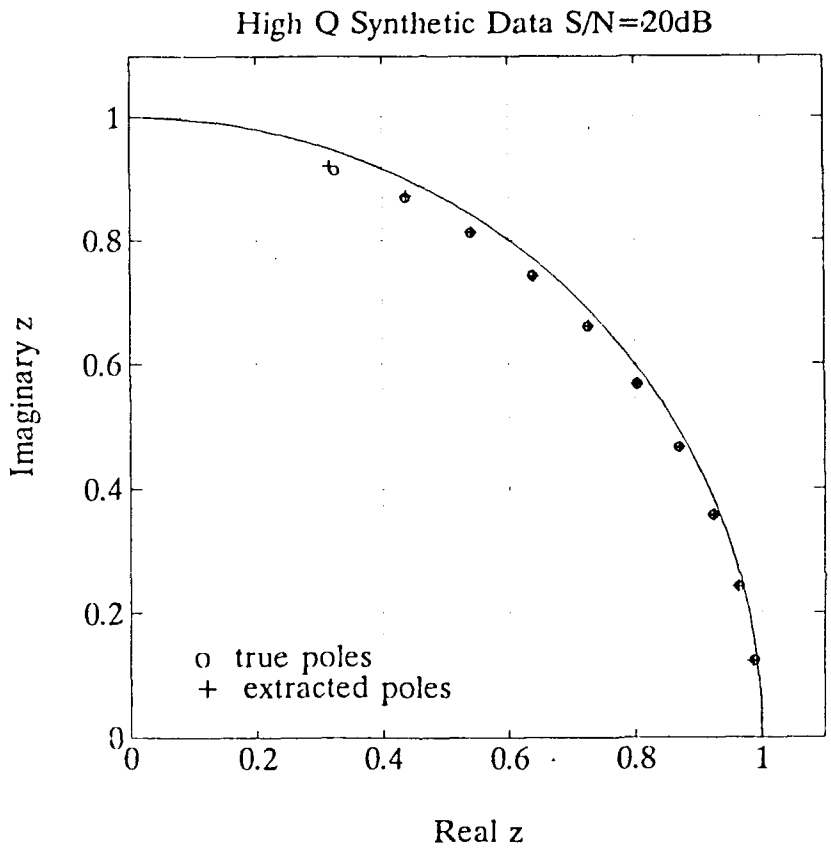

Figure 7. Pole Extraction from High Q Synthetic Signal, S/N=20 dB

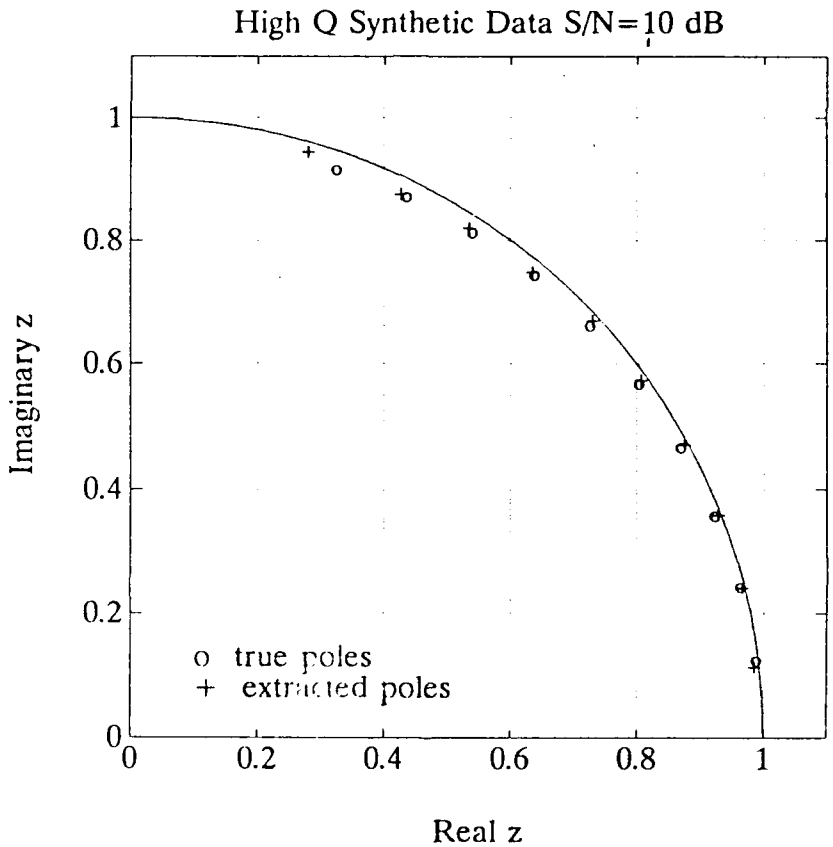

Figure 8. Pole Extraction from High Q Synthetic Signal, S/N=10 dB

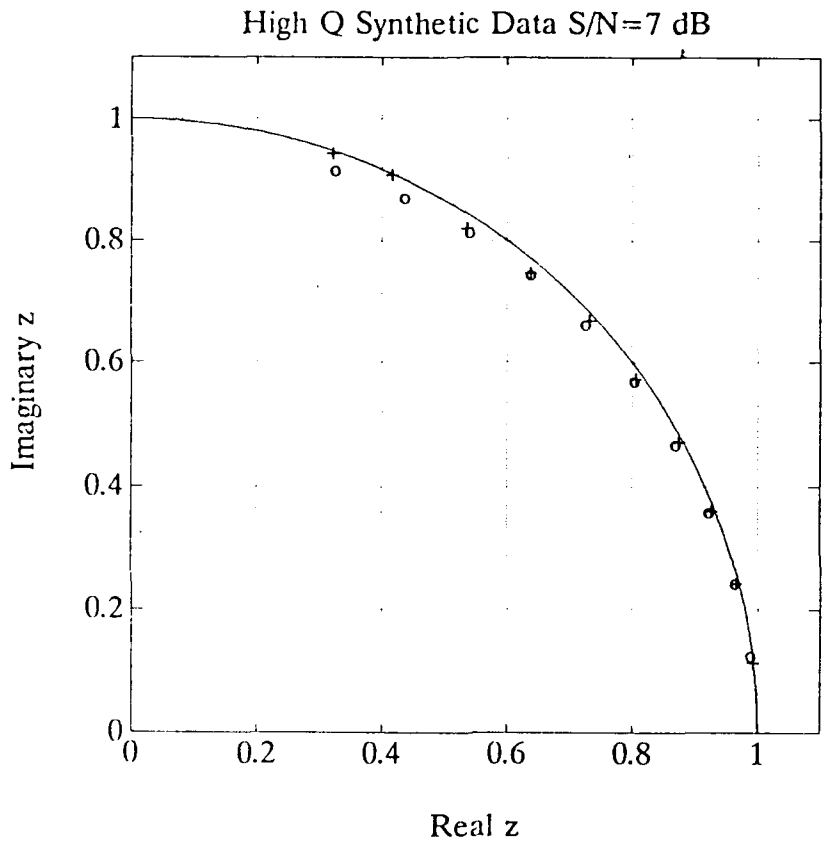

Figure 9. Pole Extraction from High Q Synthetic Signal, S/N=7 dB

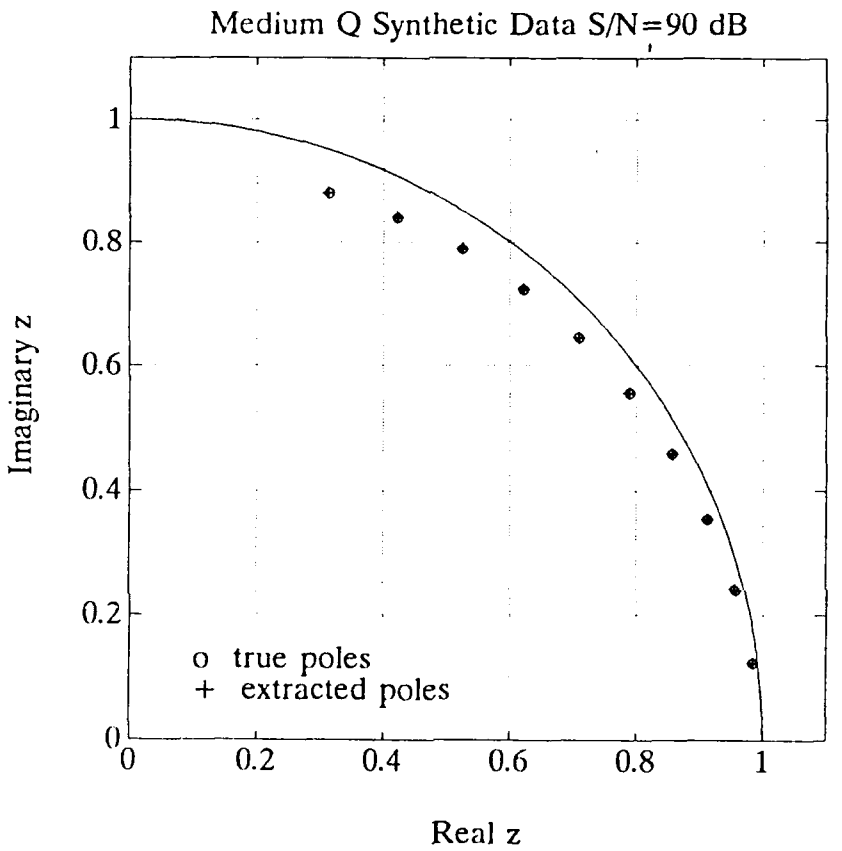

Figure 10. Pole Extraction from Medium Q Synthetic Signal, S/N=90 dB

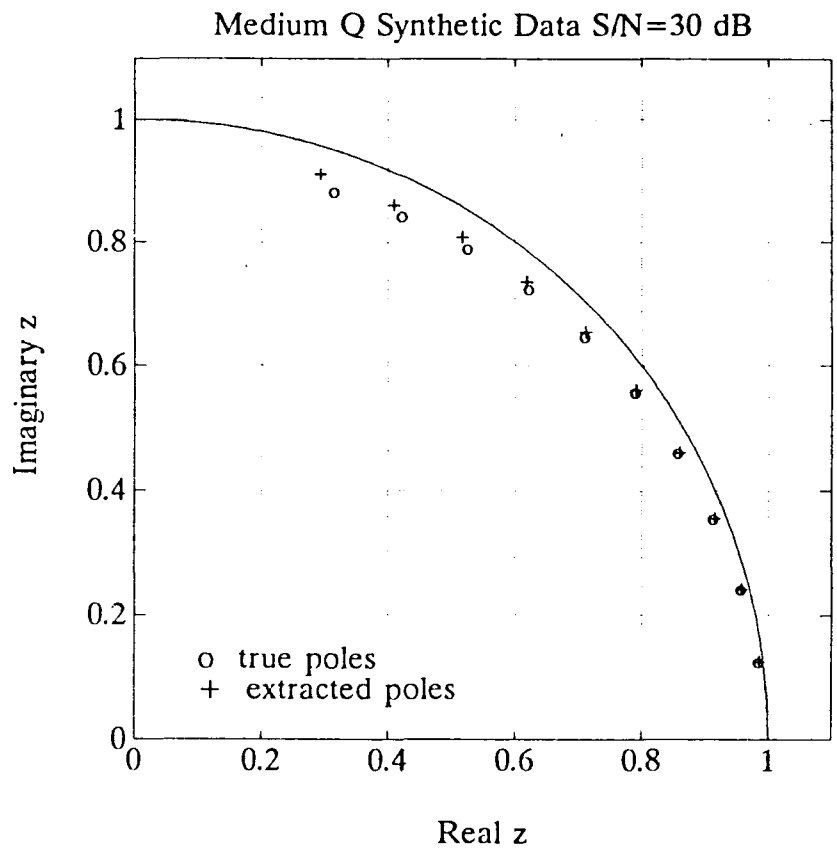

Figure 11. Pole Extraction from Medium Q Synthetic Signal, S/N=30 dB

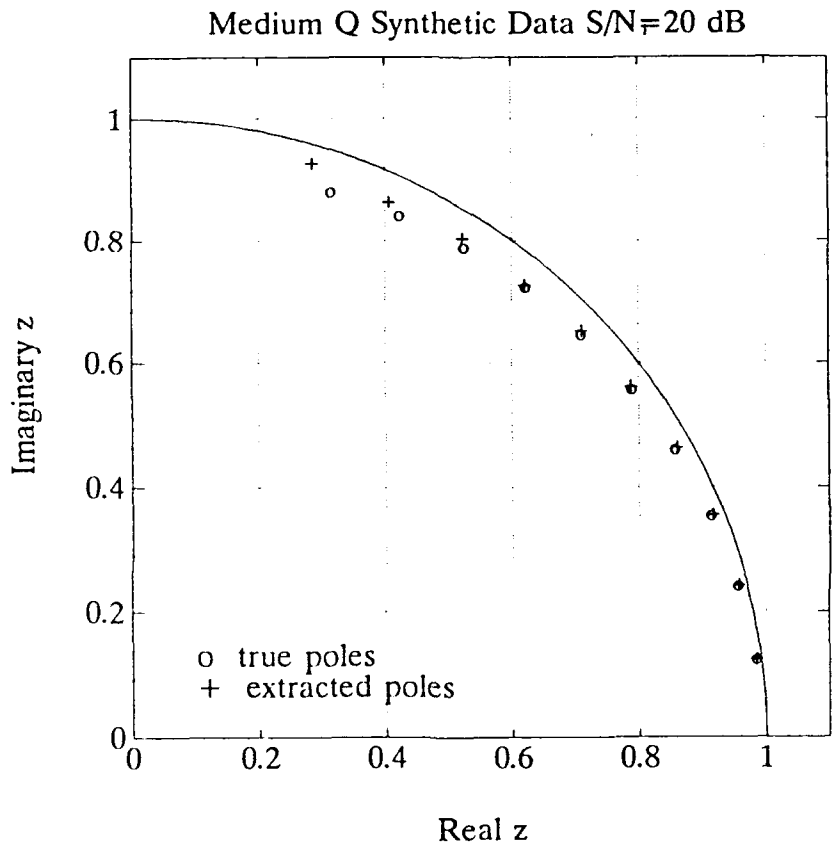

Figure 12. Pole Extraction from Medium Q Synthetic Signal, S/N=20 dB

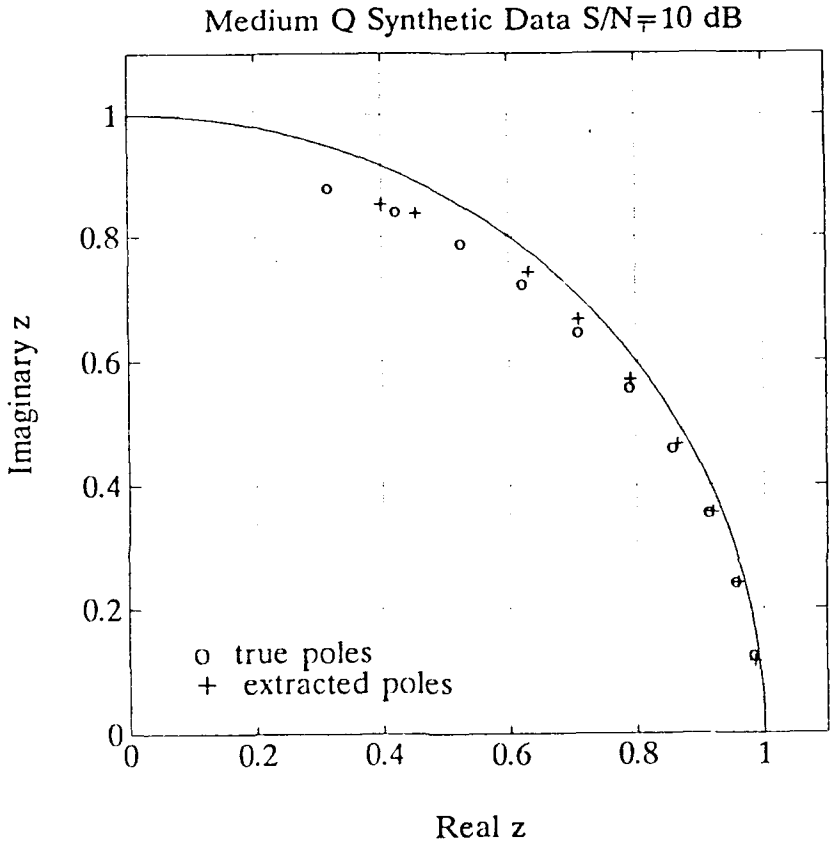

Figure 13. Pole Extraction from Medium Q Synthetic Signal, S/N=10 dB

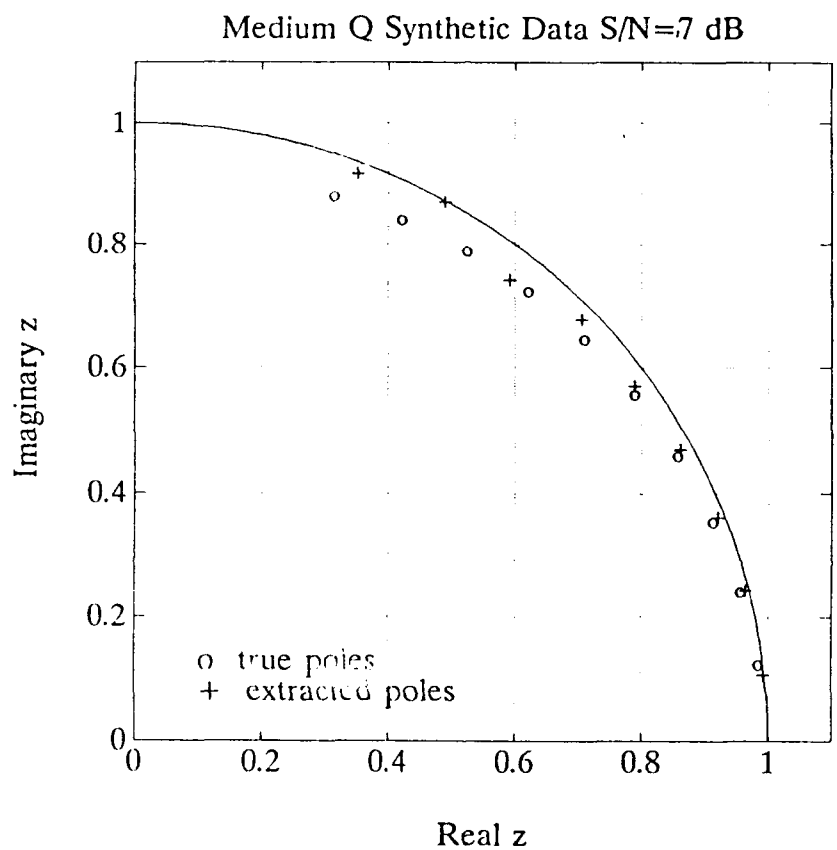

Figure 14. Pole Extraction from Medium Q Synthetic Signal, S/N=7 dB

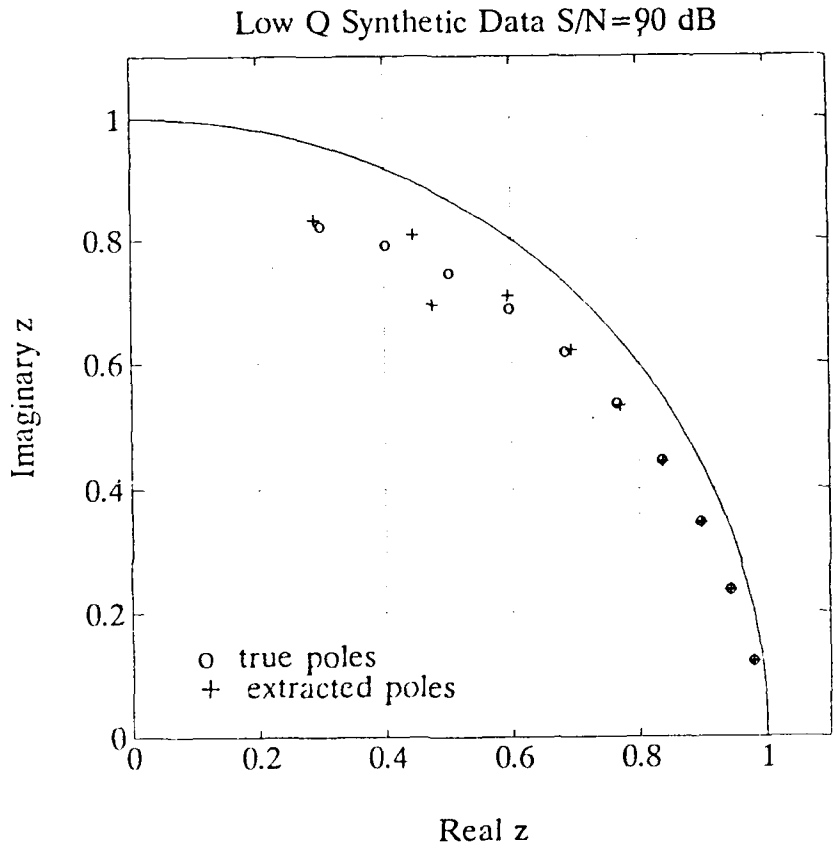

Figure 15. Pole Extraction from Low Q Synthetic Signal, S/N=90 dB

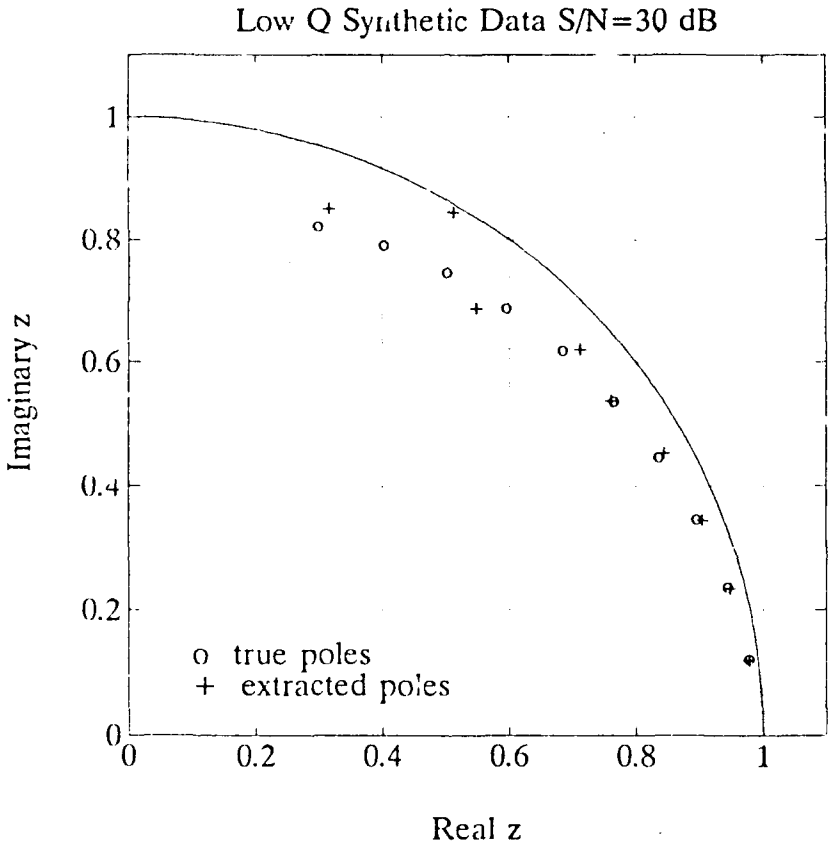

Figure 16. Pole Extraction from Low Q Synthetic Signal, S/N=30 dB

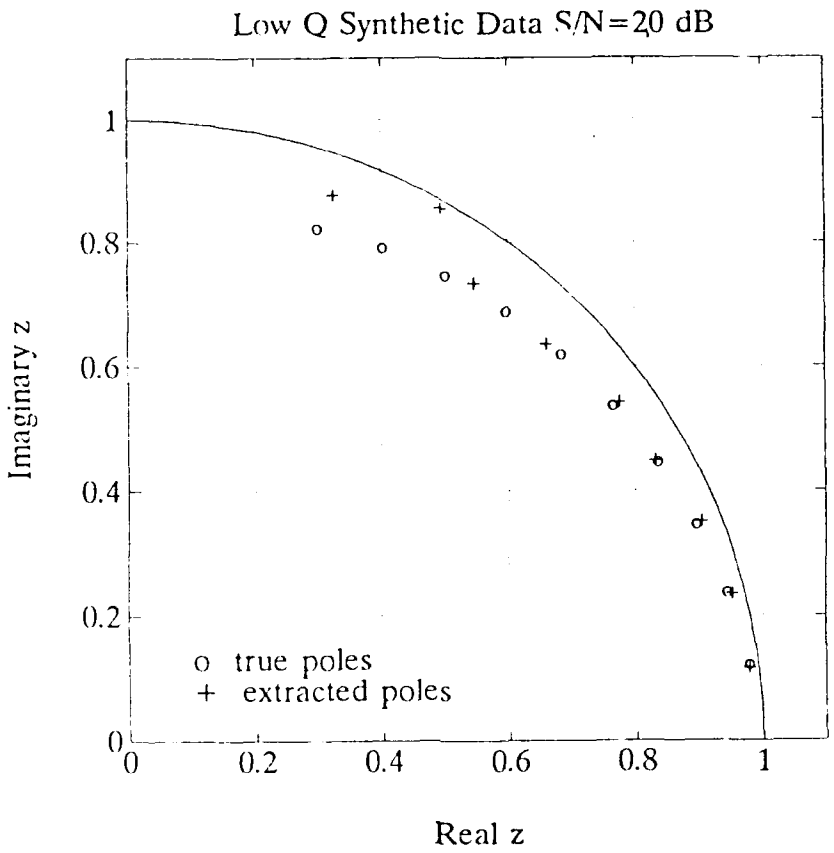

Figure 17. Pole Extraction from Low Q Synthetic Signal, S/N=20 dB

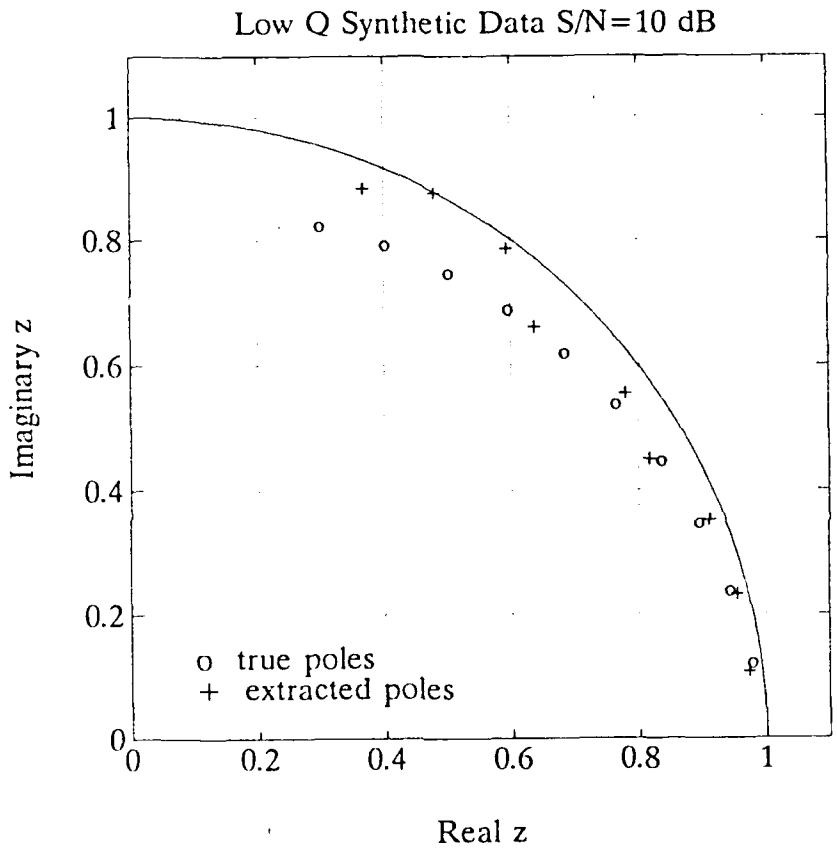

Figure 18. Pole Extraction from Low Q Synthetic Signal, S/N=10 dB

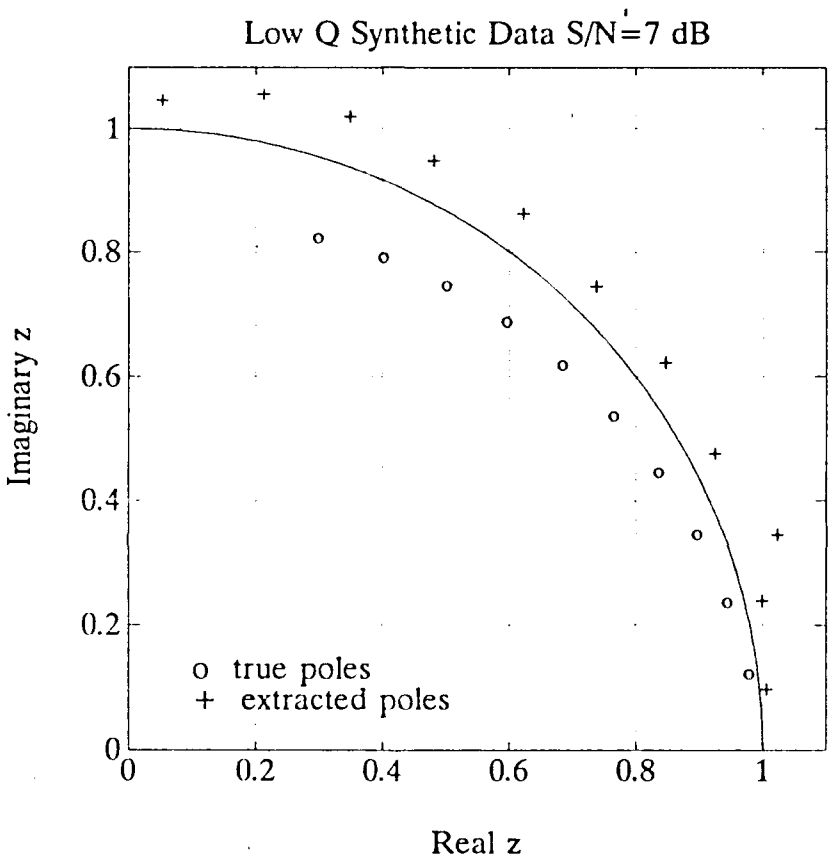

Figure 19. Pole Extraction from Low Q Synthetic Signal, S/N=7 dB

TABLE 1. EXTRACTED POLE ERROR DISTANCES

S/N (dB)	HIGH Q	MEDIUM Q	LOW Q
90	2.7423E-5	1.5821E-3	2.8017E-2
30	7.2861E-4	5.9002E-3	2.7849E-2
20	2.2292E-3	1.3615E-2	3.8008E-2
10	1.0582E-2	2.2484E-2	5.0115E-2
7	1.2609E-2	2.4227E-2	1.2668E-1

In moving from synthetic data to measured data the problem becomes one of attempting to locate unknown poles rather than attempting to minimize the error distance between a known pole location and an extracted pole. True poles were expected to tend to assert themselves repeatedly despite slight variations in the parameters used in processing or despite different noise sequences. Poles to be used for filtering were then going to be taken to be the centroids of a cluster of extracted poles. In an attempt to simulate the techniques which would be used in extracting poles from measured data, one further test using synthetic data was conducted. Fourteen different medium Q data sequences were created. To each of these noise was added, using a different seed for the noise generator, to a signal to noise ratio of 20 dB. Each of these sequences was processed and the extracted poles have all been plotted in Figure 20. The clustering is very apparent in this plot. The lower frequency poles have the extracted poles from each of the fourteen different signals

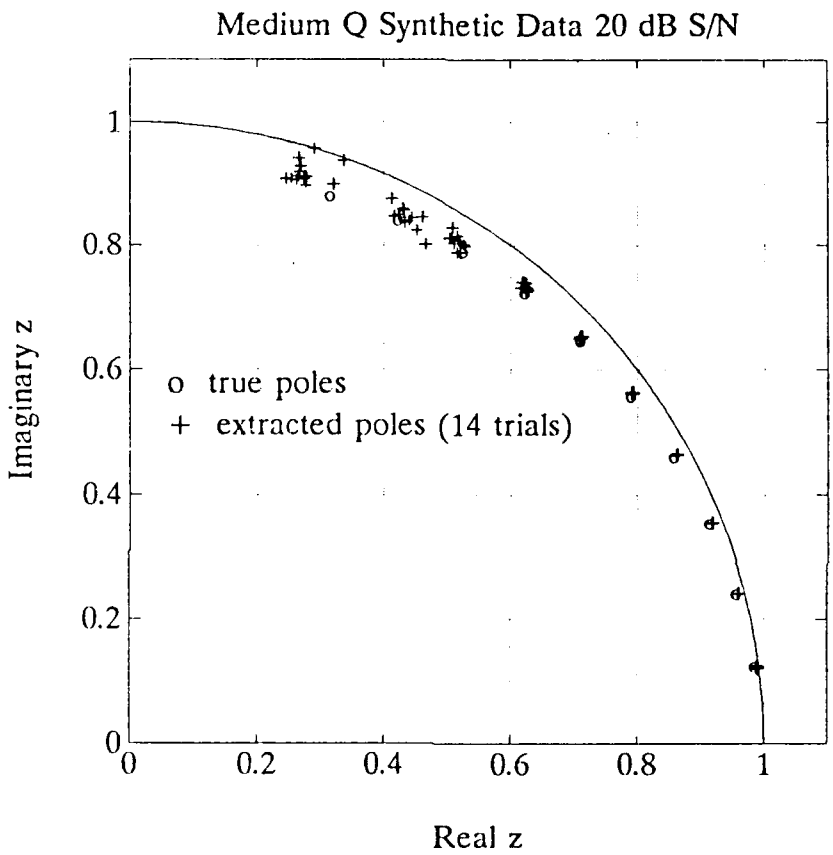

Figure 20. Test of Pole Clustering Using Synthetic Signals

so tightly grouped that they are indistinguishable. Observing that the lowest frequency pole is not quite as tightly grouped as the next few poles despite the fact that it has less damping illustrates the effect of the spectrum of the exciting waveform (Figure 3). The energy contained in the exciting pulse is significantly lower at the frequency of the lowest pole compared to the next several poles. Thus this mode is less strongly excited and the pole itself is more difficult for the algorithm to extract. Figure 21 displays the magnitude of the Fast Fourier Transform of one of these fourteen test sequences and it serves to further illustrate this observation.

Based on all of the results using synthetic data, several observations were made. The first of these is that best results were obtained by choosing a start point located within several points of the zero crossing nearest to the first obvious response to the excitation. This is in accordance with observations made by Larison [4].

The second observation made is that best results were normally obtained using a data matrix which was as large as possible. During the synthetic data testing of this work, the program being used allowed a data matrix with maximum dimensions of 70x70. Within this framework, best results were obtained by setting K_N=20, the actual order of the numerator of the system transfer function, and setting K_D=48, thereby filling out the data matrix. Underestimating the value of K_N would result in inaccurate results. Overestimating K_N did not increase accuracy. This is due to the fact that the input waveform was noise free and therefore did not require overestimation. Conversely, the output waveform was not noise free and by

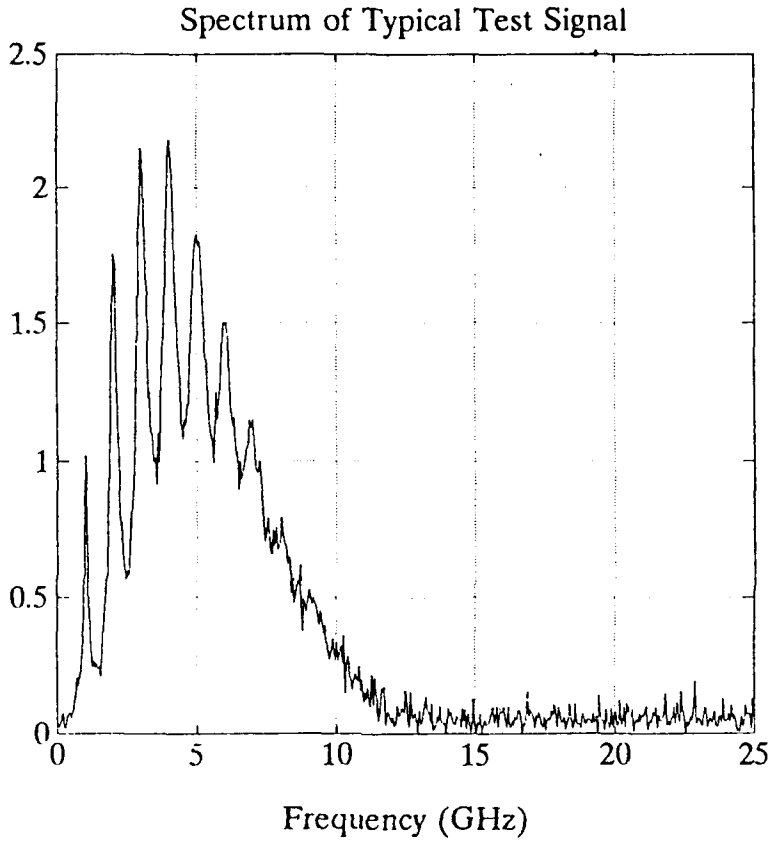

Figure 21. Magnitude of the FFT of One of the Synthetic Signals Used in the Cluster Test

overestimating K_N, there was less space available in the data matrix for overestimating K_D and eliminating some of the effects of noise. It was also noted that if the signal was of short duration, it was not always advantageous to use the maximum possible size of the data matrix. Once the signal becomes completely buried in the noise it is better not to include any further points in the data matrix. Thus it is best to include as many full cycles of data contained in the output waveform as was possible within the limitations of matrix size and noise.

The best results were not always obtained when using the bias compensation scheme by setting the estimated system order to the true value of 20. In particular at high signal-to-noise ratios more favorable results were obtained by setting the estimated system order to higher than the true value. The Akaike Information Criterion (AIC) routine was extremely useful for identifying this optimum value.

Figure 20 illustrates that despite the use of the bias compensation scheme, an element of radial (damping level) bias may be expected throughout the bandwidth of the measurement system. At higher frequencies, where the damping is higher, an element of axial (frequency) bias appears as well.

C. THIN WIRE SIGNAL TESTING

The algorithm was further tested using both Morgan's time-domain thin wire integral equation (TDIE) program [21] and thin wire measured scattering data. For a detailed explanation of the techniques used for measuring the scattering from the thin wire as well as the scale model aircraft targets see [22].

1. Thin Wire Integral Equation Generated Data

The TDIE program was used to compute the backscattered response of a thin wire to an incident double Gaussian pulse for several different incident angles. These responses were then perturbed by noise to signal-to-noise ratios of 90 and 20 dB. Figure 23 illustrates the results of the pole extraction for the 90 dB S/N case. The first five poles appear with very strong clustering for all angles of incidence. It should be noted that these true poles appeared quite consistently in these locations despite variations in the parameters used in processing the signal with the Cadzow-Solomon algorithm. Noise poles and the superfluous signal poles located closer to the center of the unit circle were much more volatile in their location as parameters were varied. Although the signal may contain more heavily damped poles than those principle poles located near the unit circle, accurate locations would be much more difficult to determine and those pole locations deep inside the unit circle should not be trusted. Due to the high level of damping, knowing their location would not be necessary for use in an annihilation filtering scheme. Figure 23 displays just the first quadrant of the previous plot in order to better illustrate the high degree of clustering of the principle signal poles. As expected, broadside excitation failed to elicit a response from every other mode. Excitation at an angle of 70 degrees from the axis of the wire apparently failed to excite every third mode.

The TDIE program generated a response which consisted of 960 points over 20 ns. This sampling rate is different from that used in the synthetic ARMA data and the measured data, thus these z-plane plots are not directly comparable.

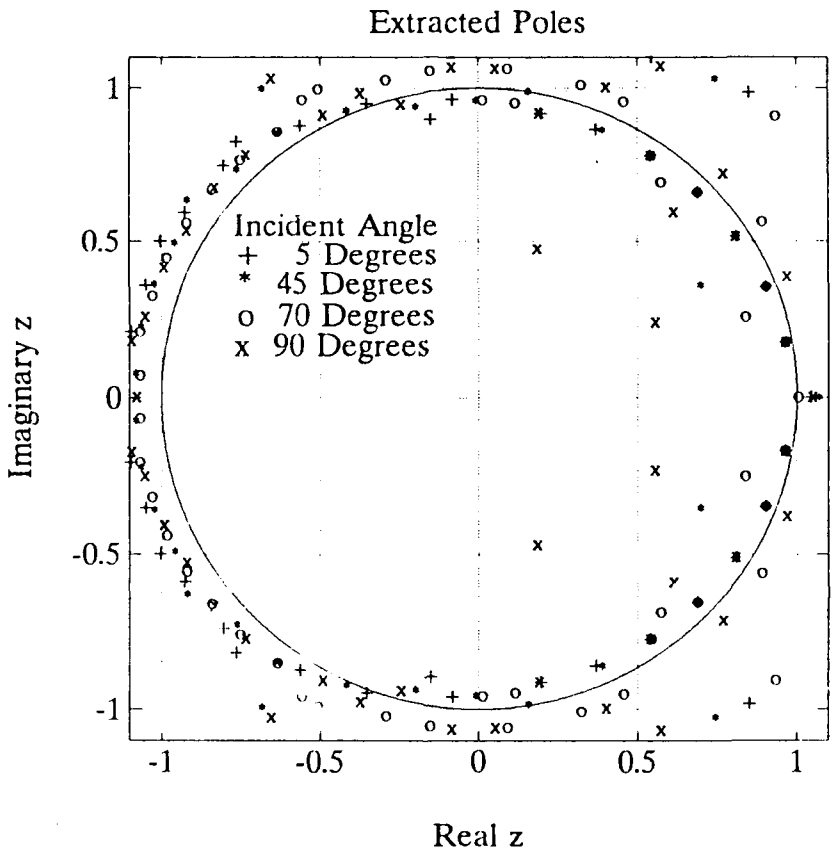

Figure 22. Poles Extracted from TDIE Generated Thin Wire Response S/N=90 dB

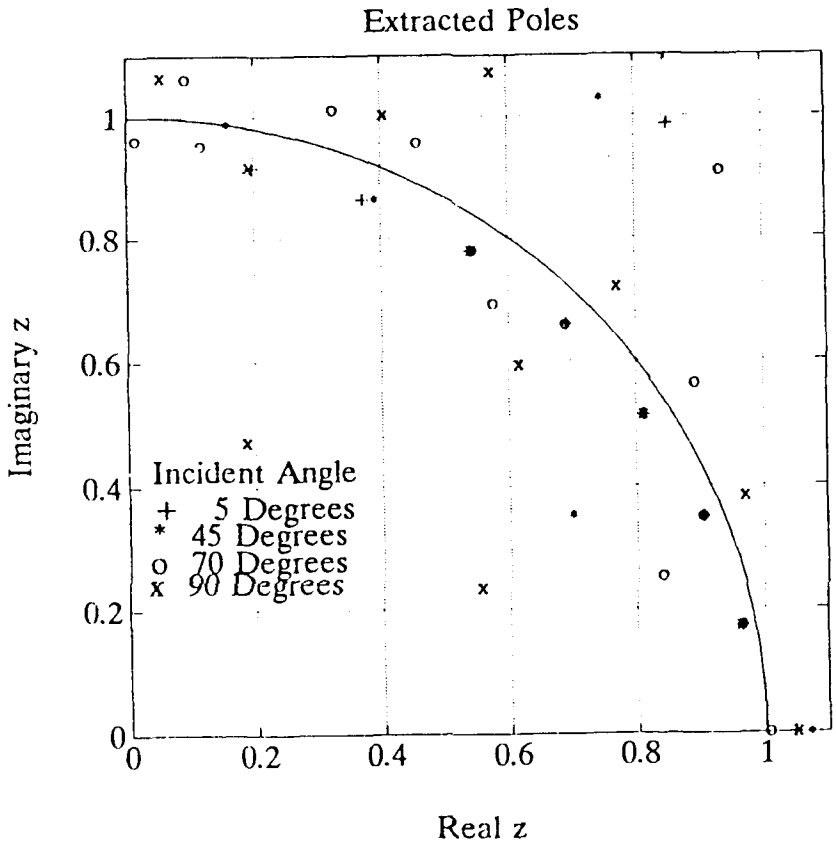

Figure 23. First Quadrant View of Poles Extracted from TDIE Generated Thin Wire Response S/N=90 dB

However, the ability to extract poles at certain frequencies corresponded well to the level of excitation provided by the double Gaussian, as was the case with the synthetic ARMA data.

In processing the thin wire data, the feed-forward order of the system, K_N', is calculated by determining the length of early-time as $2L/c$. The duration of the pulse width need not be included in this calculation because the pulse itself is included in the data matrix which is processed. This time value is then converted to the appropriate number of time points based on the sampling rate used. This number of time points is the minimum value which can be used for K_N, since it represents the number of delays in a z-transform which would be necessary to represent the early-time of the system. Depending on how large this value is, it may be more efficient to back off some number of points from the earliest possible processing point in order to process less of the early-time data and to keep the value of K_N low. In general it was found that best results could be obtained if at least 20 points of early-time were processed. For thin wire data at near broadside angles of incidence, early-time will be very short when calculated in this manner. Nonetheless, it is necessary to retain some minimum value of K_N in order to allow the necessary information regarding the excitation to be included in the model of the system.

Another important consideration in the processing of this data which was observed was the scaling of the input waveform. Although the data was generated using a double Gaussian pulse with a peak amplitude of 1 volt, better pole extraction results could be achieved if the double Gaussian waveform used in the data matrix

for processing had a peak value which was on the same order of magnitude as the peak value of the response waveform. Such scaling in no way changes the frequency content of the exciting waveform but it does apparently provide better results by minimizing some of the effects of ill-conditioning in the data matrix.

Figure 24 illustrates the results of the pole extraction at a signal-to-noise ratio of 20 dB. Figure 25 is again a view of the first quadrant of the previous plot. Reasonable clustering is still evident in the first five poles, although the effects of the noise are obvious.

The efforts at pole extraction in this thesis were done in concert with work on annihilation filtering by Reddy [23]. Reddy used the poles extracted here from the TDIE thin wire data to build an annihilation filter. He built two additional filters, one of which had its pole locations perturbed above those extracted by five percent in both frequency and damping, as well as one in which the poles were perturbed below the extracted location by five percent. In passing the TDIE thin wire signals through these filters, he found that the filter which consisted of zeros corresponding to the poles extracted here consistently exhibited lower output energy than did the filters with perturbed zero locations.

2. Thin Wire Measured Data

Figure 26 illustrates the results of pole extraction from the actual measured response of a thin wire. The results of this extraction are roughly comparable to that achieved for the TDIE thin wire data at a signal-to-noise ratio of 20 dB. Reddy [23] used these extracted poles again to build three filters, one of which used zeros

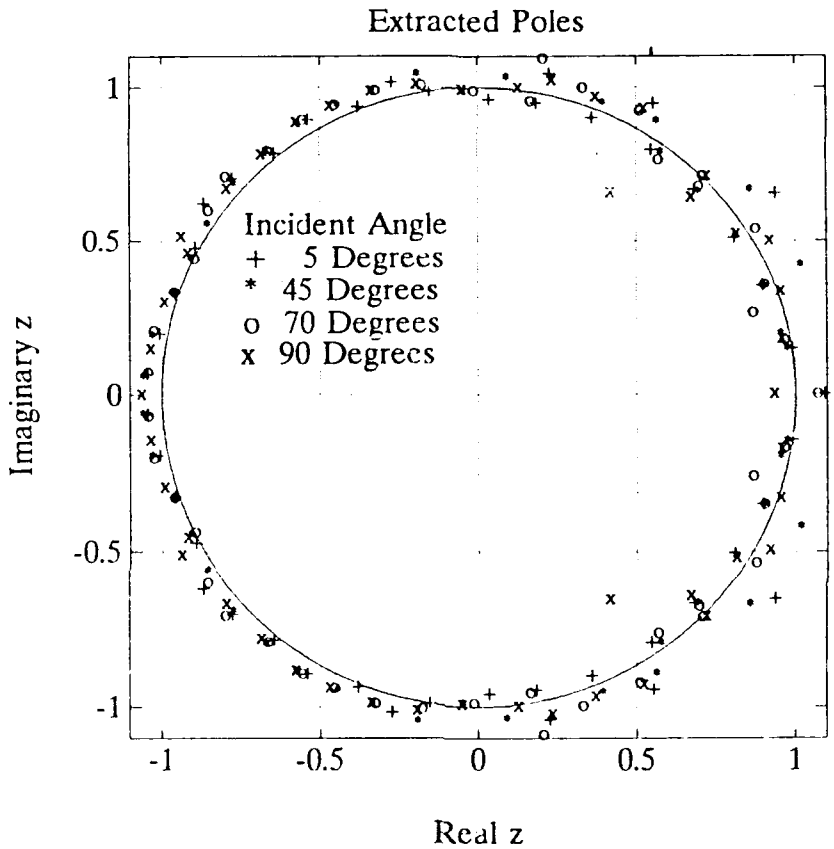

Figure 24. Poles Extracted from TDIE Generated Thin Wire Response S/N=20 dB

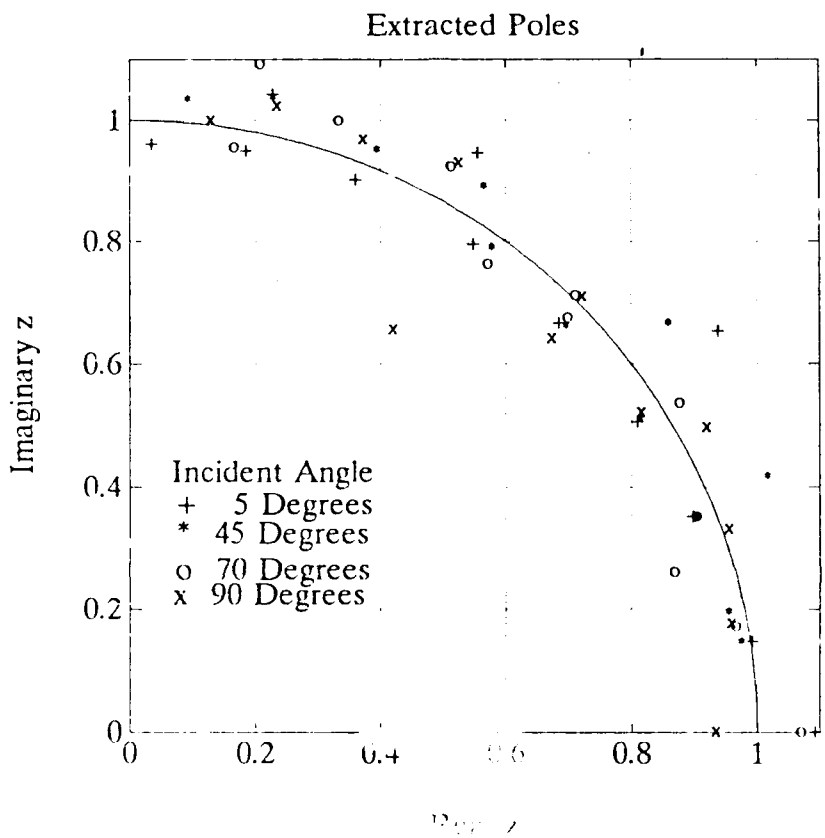

Figure 25. First Quadrant View of Poles Extracted from TDIE Generated Thin Wire Response S/N=20 dB

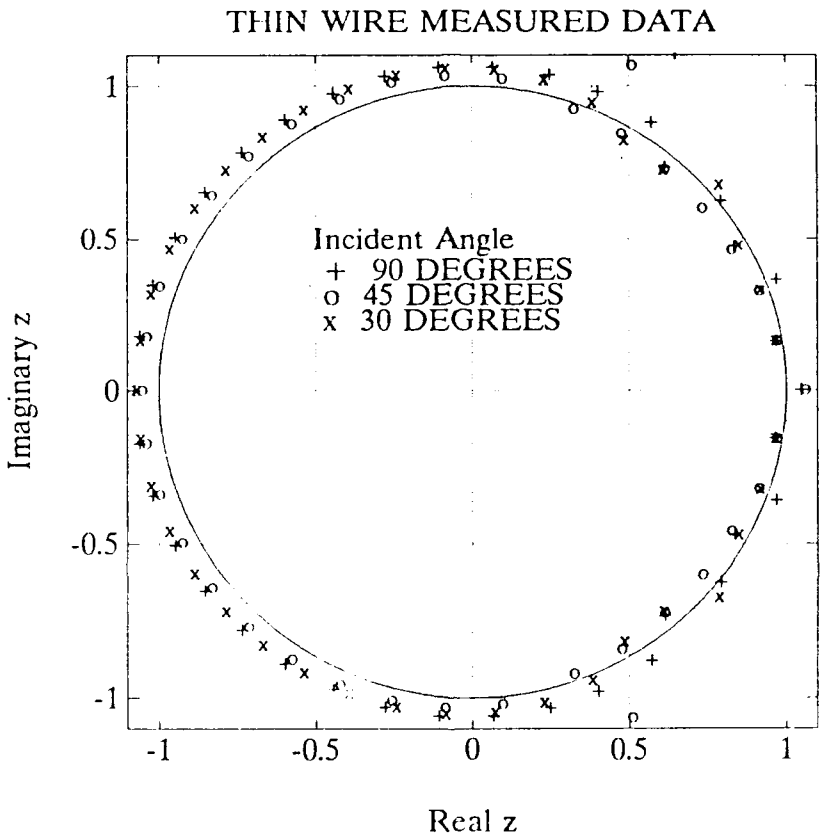

Figure 26. Poles Extracted from Thin Wire Measured Responses for Various Angles of Incident Excitation

corresponding to the extracted poles and the others based on five percent variations. The correct filter again consistently exhibited the lowest output energy when the measured thin wire responses were fed through them.

IV. POLES OF SCALE MODEL AIRCRAFT

The original objective of the work done for this thesis was the construction of a library of poles corresponding to a number of different scale model aircraft targets in order that these poles could be used to demonstrate the identification of complex targets in an annihilation filtering scheme. This was attempted after poles had been extracted from two aircraft models, but the corresponding filters were unable to properly identify the correct target with any degree of consistency. This led to an investigation of the potential problems. The principle reason for the failure was apparently the complexity of the spectrum of the responses of these complex targets to the exciting pulse. This complexity was much greater than expected. As will be shown, this complexity can lead to clustering at incorrect pole locations. A consequence of this complexity, as discovered by Reddy [23], is that annihilation filters with enough selectivity to discriminate between similar targets are difficult to build. This chapter will examine some of the problems which occurred when attempting to build a library of scale model aircraft target poles.

A. COMPLEXITY OF THE RESPONSE OF COMPLEX TARGETS

As mentioned, the complexity of the response of the scale model aircraft targets in the frequency domain was more complex than was expected, and may be more complex than is appropriate for using any of the discussed pole extraction methods as they currently stand. This section will examine this complexity through the use of

the Fast Fourier Transform. It will also be demonstrated that observations of well defined clusters when using the Cadzow-Solomon algorithm, as it was used during testing, are no guarantee that poles have been accurately located. Difficulties arise not only in accurately locating the poles of these complex targets, but in determining which of the target's poles can be used most successfully in an annihilation filtering target identification scheme.

1. False Pole Clustering

Data for two different scale model aircraft targets were processed using the Cadzow-Solomon algorithm, while following the lessons learned during the algorithm testing. Poles were extracted from the measured responses of the aircraft to electromagnetic excitation incident at 0, 30, 90, and 180 degrees from nose-on as well as two cases with the excitation incident onto the top of the aircraft, one case with the wings parallel to the incident electric field and the other with the fuselage parallel to it. In processing the thin wire data, the principle poles consistently exerted themselves despite the use of a wide range of parameters in processing. With the scale model aircraft targets, there were no poles which appeared nearly as consistently. Nonetheless, it was possible to achieve excellent clustering with slight variations in the parameters used. It was *assumed* that if poles repeatedly exerted themselves despite slight variations in the parameters used in processing, they are very likely to be true poles. Thus, in processing the data, the set of parameters which allowed clusters of poles to exhibit this characteristic was searched out. The tell-tale

sign of orderly spacing of the noise poles was also an important characteristic which was watched for in attempting to locate the aircraft poles.

Figure 27 illustrates the extracted poles from one of the responses of one of the scale model targets. The clusters are the result of seven different runs of the algorithm in processing the signal. Each of the seven runs had one of the principle parameters (starting point for processing, estimated system order used for bias compensation, and data matrix size) varied slightly about some initial value. Although the figure shows only those poles inside or very near the unit circle in the first quadrant of the z-plane, the noise poles did appear in an orderly fashion as has been mentioned. Plots similar to this were generated for each of the six measured responses of the target. The centroid of each of the clusters in each plot was calculated and was expected to be a true pole for the response at that particular angle of incidence. Because it was desired to demonstrate aspect independent target classification, the poles for each of the angles of incidence were compared and those which appeared with a degree of consistency were averaged and used in the initial filtering attempt. In this manner, filters for two different aircraft were constructed. These filters were then used in an unsuccessful attempt at discriminating between the two aircraft [23].

In investigating the reason for the inability to discriminate between the two targets, one of the things which came out was that it is possible to have pole clustering at false pole locations. This was the case for the poles in Figure 27. Figure 28 illustrates pole clustering for the same aircraft and angle of incident

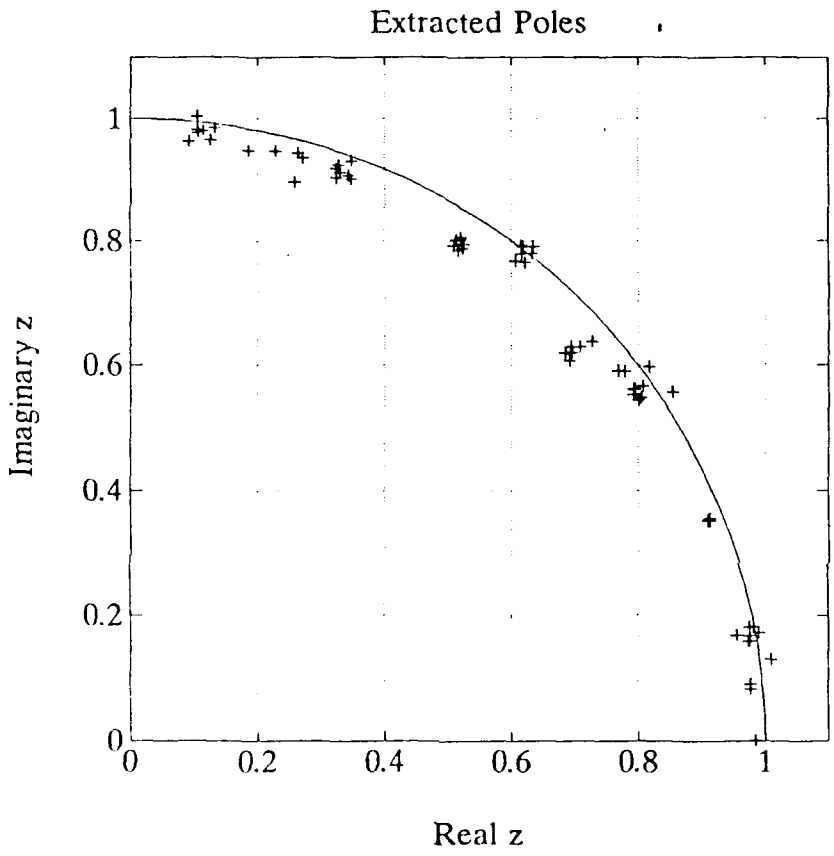

Figure 27. Clustering of Extracted poles for Target No. 1

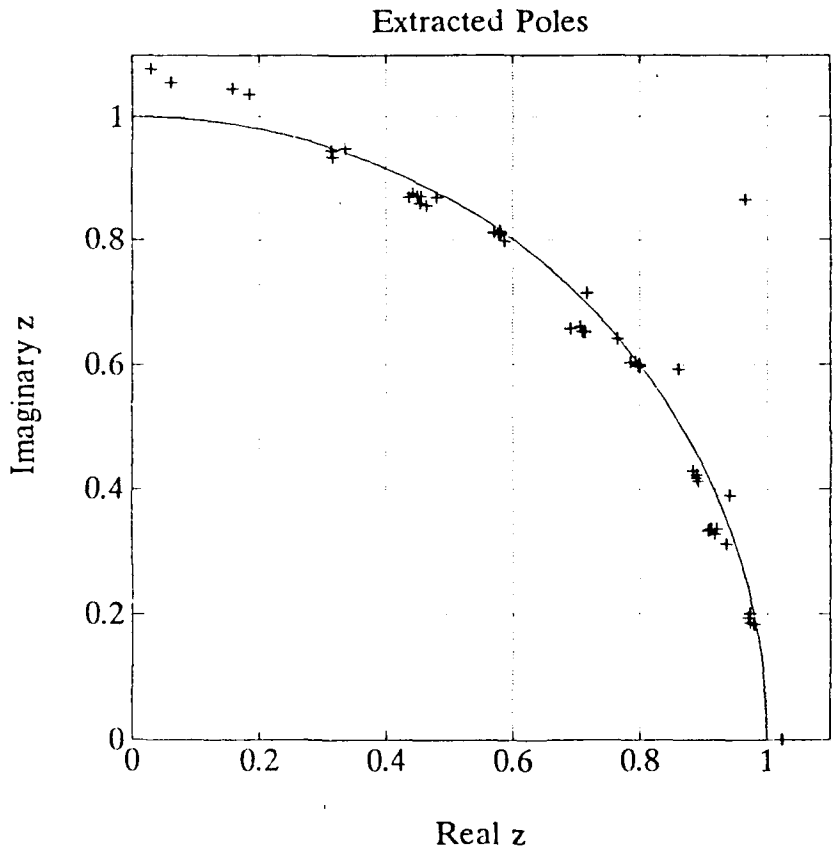

Figure 28. Clustering of Extracted Poles for Target No. 1 Using New Processing Parameters.

excitation as in Figure 27. For this figure a different set of parameters was used in processing the data for the initial case, but the clustering was achieved by varying the parameters in the same manner as was done for Figure 27. The array of parameters presents the processor with a huge number of degrees of freedom in processing the data. With these targets it seemed that by choosing the parameters appropriately, clusters could appear almost anywhere. A consequence of this is that clusters could also appear in comparing poles extracted from the returns of the target illuminated from various aspects without. These cluster locations can also be false in that they may not lead to success when used in an annihilation filter.

In order to guard against the possibility of false clustering, steps were taken to ensure that the model used in processing the data was appropriate. As discussed in the previous chapter with regards to the processing of thin wire scattering data, calculations regarding the feed-forward order of the system were made based on the size of the target and the resulting duration of early-time. Scaling of the input waveform was used to prevent numerical ill-conditioning.

The possibility that the complexity of the current modes during the earliest portions of late-time induced numerical errors was also investigated. During this period, there may be feed-forward modes which are incomplete and by attempting to incorporate them into the signal model, errors in pole locations may result. In an attempt to avoid this problem the data was processed using only the later portion of early-time, with a corresponding reduction in the value of K_N used. No direct

improvement was noted in the ability to extract poles which would provide better results in an annihilation filtering scheme.

2. A Look at the Spectra

Further investigation revealed that the problem may not have been so much that the algorithm was incapable of locating poles in the target's response as the response having just too many closely located poles. The FFT was used to study the spectra of the response of the scale model aircraft targets. Although the FFT cannot provide pole locations which could be used for target identification, it can provide some insight into the nature of the response of scale model aircraft to incident electromagnetic excitation.

Figure 29 illustrates the magnitude of the FFT of the response of a single aircraft for three different angles of incident excitation. These FFTs were taken using the points which were expected to best fit the model used in the Cadzow-Solomon processing algorithm. The last twenty points of the calculated early-time were included as well all of the late-time response until the signal appears have decayed well into the level of the noise. The data file was then zero padded to provide a reasonable degree of resolution prior to taking the FFT. Several other possible windows of the data were used in attempts to gain more insight into the frequency domain, but that presented here appeared to provide the most insight. Including more late-time points resulted in a plot having a larger variance and including much more of the early-time tended to smooth the plot out to the point where it provided little useful information.

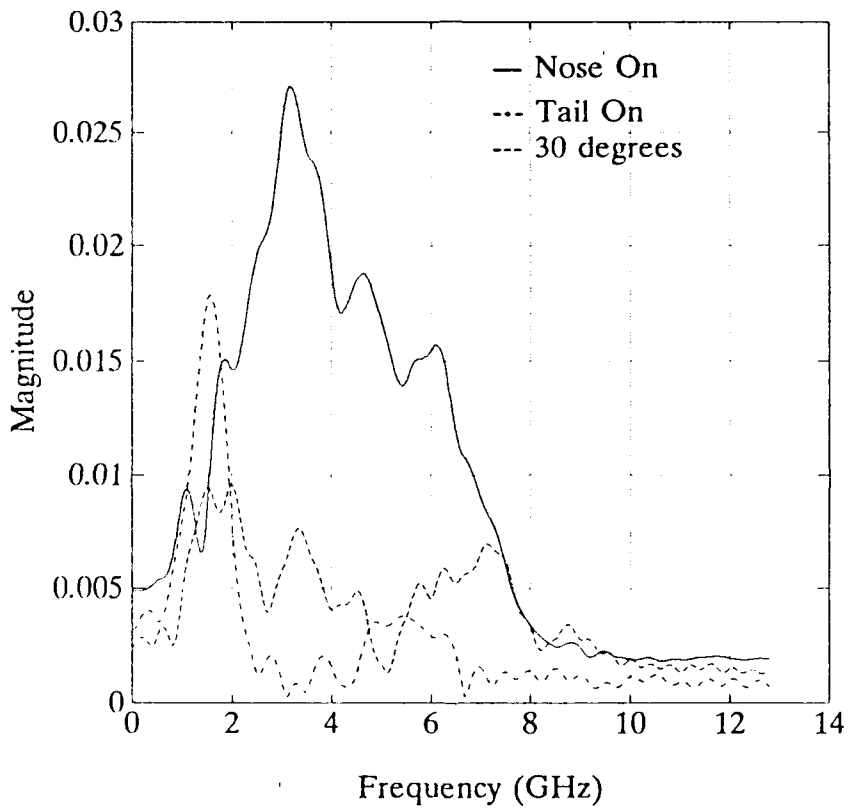

Figure 29. Spectra of the Responses of a Single Scale Model Aircraft to Electromagnetic Excitation Incident at Various Angles

Examining Figure 29 reveals that there is not a single frequency which appears to be resonant to any strong degree for all three aspects. The figure also shows that there may be poles which are closely located and that the poles may not appear with any regularity in terms of their spacing in frequency. The three aspects which are included in this figure were expected to provide relatively similar responses to the excitation. Viewed electromagnetically, the target was expected to look similar directly from the front and from the rear; that is the principle structures which will resonate (wings, fuselage, etc.) were illuminated with fields which were orientated in the same direction in each case.

Figure 30 displays the same situation for another of the scale model aircraft targets. The problem of the large number of poles present within the bandwidth excited by the double Gaussian pulse is again visible as well as the lack of a strong correlation between different angles of incidence. Comparing the two figures reveals another problem. Poles which seem to correspond closely between the two targets appear to have been excited for some of the angles of incidence. If the pole finding algorithm can be refined to locate poles in the response of complex targets to a high degree of accuracy it should still be possible to discriminate between these closely corresponding poles. Poles located this closely would still present problems if they are to be used in an annihilation filtering scheme for target identification as well be examined in the next section.

In contrast to the two preceding figures, Figure 31 displays the spectra of the measured response of the thin wire using the same rules to decide which points

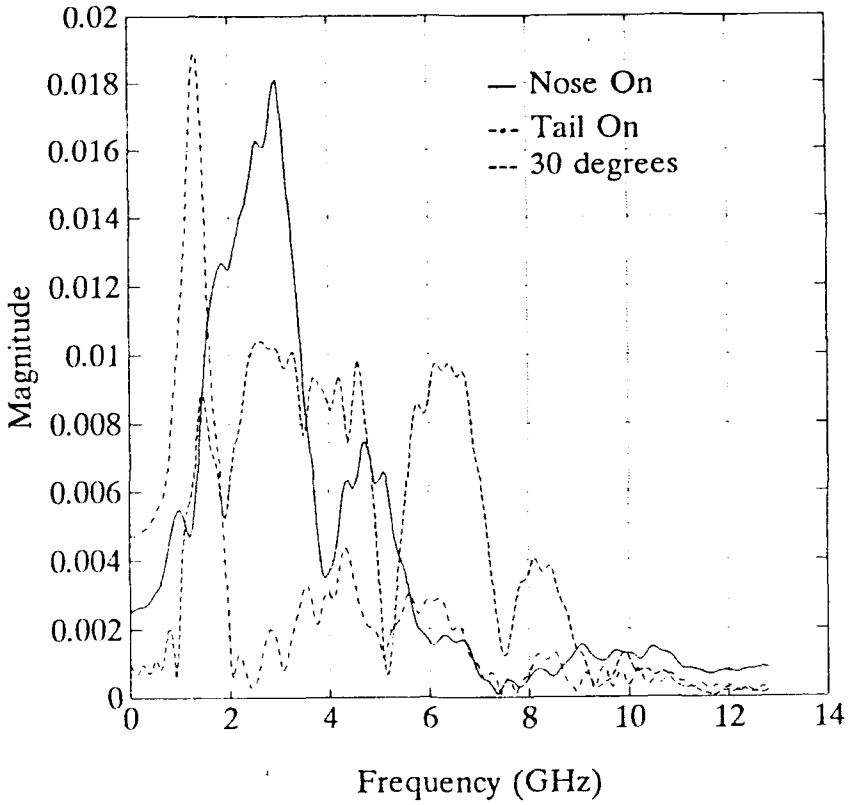

Figure 30. Spectra of the Responses of a Second Scale Model Aircraft to Electromagnetic Excitation Incident at Various Angles

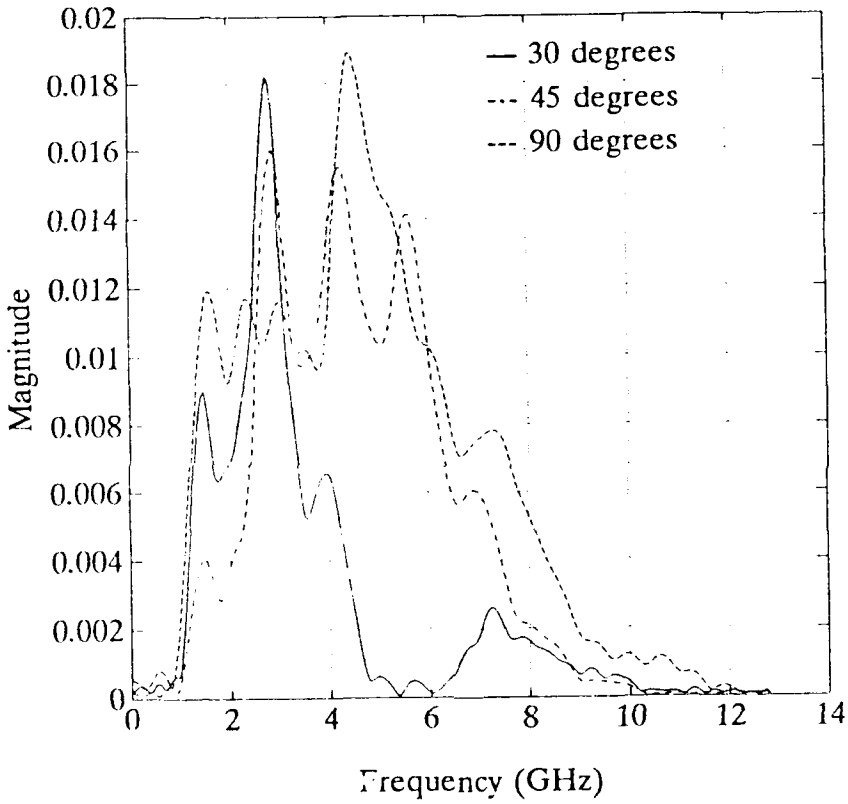

Figure 31. Spectra of the Responses of a Thin Wire to Electromagnetic Excitation Incident at Various Angles

would be incorporated in the FFTs. Here the correspondence in resonances between the different angles of excitation, particularly for the lower frequencies. Comparing this figure with Figure 26 reveals good correlation between the appearance of a strong resonance in the FFT and the ability to extract a corresponding pole with the Cadzow-Solomon algorithm.

B. ANNIHILATION FILTERING LIMITATIONS

The inability to demonstrate target identification using a resonance based annihilation filtering scheme appears to be due not only to problems with pole extraction caused by the complexity of the spectrum, but also to limitations of the annihilation filter when attempting to discriminate targets with closely located poles. As utilized by Reddy [23], an annihilation filter consists of a pulse, known as a "kill-pulse," whose zeros correspond to the poles of the target whose response is to be annihilated. This pulse is convolved with the target response and the late-time energy of the resulting waveform is measured. Figure 32 illustrates the spectrum of one of the kill-pulses used in the attempt to demonstrate target identification using annihilation filters. Zeros appear in the spectrum at approximately 1.5, 3.0, 3.5, and 4.5 GHz. Figure 33, which is a close up of the lower frequencies of the spectrum of this same kill-pulse, helps to reveal that this pulse actually was built with a fifth zero at 1.2 GHz. This figure illustrates that annihilation filters, as they have been implemented here, are incapable of providing sufficient resolution to discriminate between closely located poles. The figure also reveals that any targets which had

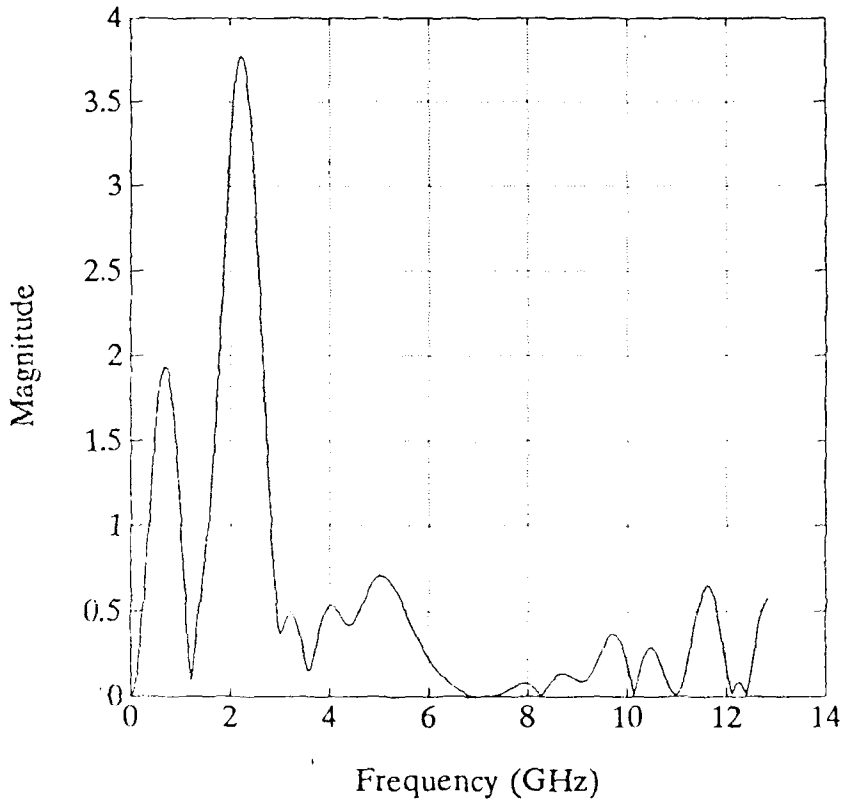

Figure 32. Spectrum of a Kill Pulse Used in the Attempt at Target Identification Using Annihilation Filtering. From Ref. 23

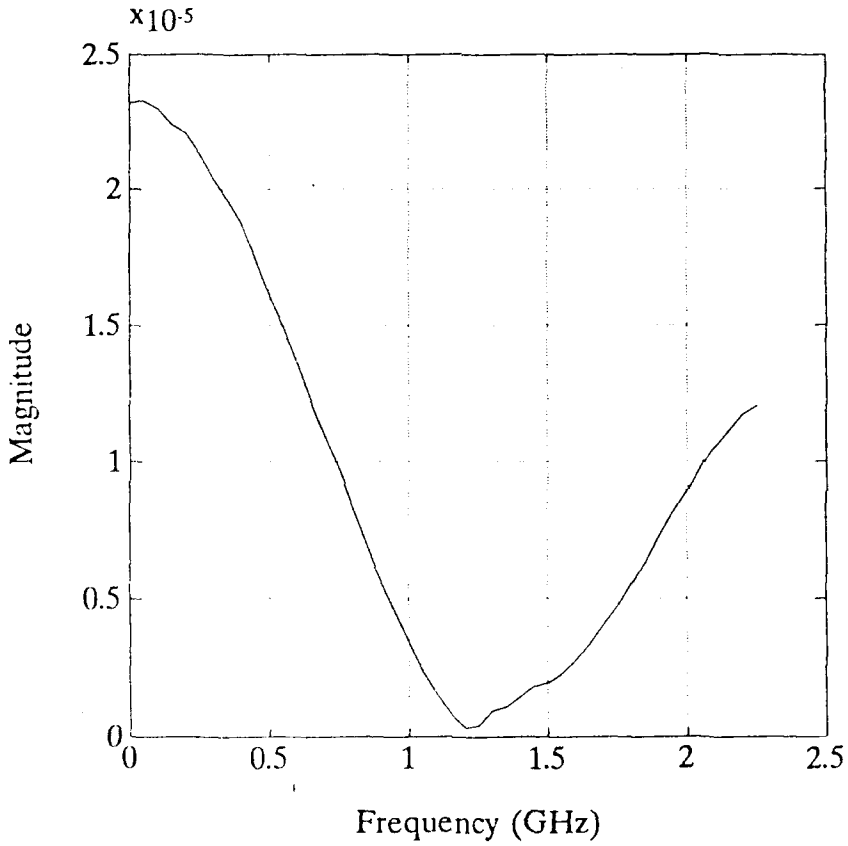

Figure 33. Close up of the Lower Frequencies of the Spectrum of the Kill Pulse From Ref. 23

poles located in the frequency range of approximately 0.7-1.7 GHz would have these poles effectively annihilated by the filter, decreasing the likelihood that the filter would provide a high degree of capability in properly identifying targets.

The problem with constructing these annihilation filters is thus twofold; poles of a single target are often closely located, and poles of other targets can also be closely located to poles of the first. If a filter is constructed with enough zeros to result in a highly annihilated late-time waveform, this same filter may also strongly annihilate the energy in the response of scale model aircraft other than the one for which it was designed. If only poles which do not correspond with any degree of proximity to poles in the response of any of the other targets are used in constructing the filter, very little of the energy in the response of the target for which it was designed will be annihilated by the filter.

V. SUMMARY AND CONCLUSIONS

This chapter reviews the findings of this thesis. In light of the fact that the original objective of developing a library of poles for scale model aircraft and using these poles to demonstrate target identification through annihilation filtering was not met, approaches which may lead to greater success in the future will be suggested.

A. SUMMARY

This thesis has presented an initial attempt to demonstrate radar target identification by building on the earlier work of Norton [16] and Larison [4]. The first portion of the work consisted of testing the Cadzow-Solomon algorithm for pole extraction. The synthetic testing phase of this work was more comprehensive than in previous efforts for two reasons: (1) the synthetic signals were generated by a true ARMA type signal generator, and; (2) there was a relatively large number (i.e., 10) of pole pairs contained within these signals. These pole pairs also covered the complete frequency range which could be expected to be fully excited in the scattering data taken from measurements in the anechoic chamber. Excellent pole extraction capabilities were noted for synthetically generated data, integral equation computed data, and for thin wire scattering data.

When analyzing the scattering data of scale model aircraft targets it was found that the Cadzow-Solomon pole extraction algorithm was unable to extract poles which could be used to implement a successful radar target identification scheme using

annihilation filters. The principle reason for this appears to be that the natural mode structure of the scattered radar signals from one of these complex targets is more complicated than was anticipated and, as a result, more complicated than was prepared for in the algorithm testing portion of this work. An additional complication caused by the complexity of the scattered signal in the frequency domain is that poles which are located close to one another, whether for a single target or for separate targets, can present difficulties in discrimination when using annihilation filtering.

B. CONCLUSIONS AND RECOMMENDATIONS

The spectra of the returned radar signals from the scale model aircraft were expected in each case to be highly dominated by a few poles resulting from the resonances of major structures on the target, such as the wings. It appears that the situation is more complicated than was expected. Further work needs to be done using targets of intermediate complexity. Investigating the scattering from multi-element thin wire targets, for example two thin wires joined in the shape of a T or a cross, could provide more insight into the nature of the response of scale model aircraft targets. Such testing would also serve to demonstrate the abilities of algorithms such as Cadzow-Solomon to extract the poles of these more complicated responses.

The algorithm testing conducted in this work was done using signals which turned out to be much less complicated than that of the scale model aircraft targets.

Future work on this subject should include attempts to extract the poles of signals which have poles which are not regularly spaced. It could also be useful to test the algorithm using a synthetic signal composed of irregularly spaced poles which are unknown to the program operator, but which could later be checked, in order to determine more completely the validity of associating extracted pole clusters with true poles.

The principle difficulty with the Cadzow-Pole extraction algorithm, as it was employed in this thesis, is the large number of degrees of freedom which the operator has in selecting the values of the parameters to be used in processing the data. It is possible that some of these degrees of freedom could be removed, resulting in more accurate and efficient pole extraction. Cadzow [24] has proposed a method of signal enhancement which could potentially eliminate the need to conduct bias compensation. The use of the ARMA based AIC system order estimator discussed in chapter II, could be useful in estimating the optimal values of K_D and K_N for processing the data.

Annihilation filters are an effective means for discriminating targets whose radar scattering returns exhibit relatively simple spectral content. Difficulties arise when attempting to discriminate more complex targets since the filters exhibit a significant width in their nulls about a zero point. In light of the apparent complexity of the frequency domain scattering response of aircraft type targets, a means to reduce the impact of this width will need to be implemented. The use of multiple zeros at each identified pole location is one possibility which may sharpen the ability of the filters

to discriminate between closely located poles. If this is not possible some other means of exploiting the natural resonances of targets may need to be discovered if they are to be used for successful target identification.

APPENDIX A. S-PLANE POLES USED IN SYNTHETIC TESTING

The following three tables list the *s*-plane poles which were used to generate the synthetic signals used in the algorithm testing portion of this work. In generating these signals each of the pole pairs was input to the ARMA coefficient generator program with an amplitude of one and a phase of zero. These poles were developed in accordance with equation (30).

TABLE A1. LOW Q SYNTHETIC POLES

f_n (GHz)	σ_n (GNp/s)	ω_n (GRad/s)
1	-0.6892	6.2474
2	-1.3784	12.4948
3	-2.0676	18.7422
4	-2.7568	24.9895
5	-3.4460	31.2369
6	-4.1352	37.4843
7	-4.8244	43.7317
8	-5.5136	49.9791
9	-6.2028	56.2264
10	-6.8919	62.4738

TABLE A2. MEDIUM Q SYNTHETIC POLES

f_n (GHz)	σ_n (GNp/s)	ω_n (Grad/s)
1	-0.3562	6.2752
2	-0.7124	12.5504
3	-1.0687	18.8256
4	-1.4249	25.1007
5	-1.7811	31.3759
6	-2.1373	37.6511
7	-2.4935	43.9263
8	-2.8498	50.2015
9	-3.2060	56.4767
10	-3.5622	62.7518

TABLE A3. HIGH Q SYNTHETIC POLES

f_n (GHz)	σ_n (GNp/s)	ω_n (Grad/s)
1	-0.1652	6.2832
2	-0.3250	12.5664
3	-0.4876	18.8496
4	-0.6501	25.1327
5	-0.8126	31.4159
6	-0.9751	37.6991
7	-1.1376	43.9823
8	-1.3002	50.2655
9	-1.4626	56.5487
10	-1.6252	62.8319

APPENDIX B. ARMA COEFFICIENT GENERATOR

A. PROGRAM DESCRIPTION

This program was used to generate the a_k and b_k coefficients of a scatterer transfer function given by

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_L z^{-L}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} \tag{A.1}$$

after reading a data file containing the desired s-plane poles for each of the N pole pairs. The program also reads in the sampling information to allow conversion of the input s-plane values to the z-plane. Additionally, the program allows input of a complex multiple of each z-plane pole pair to allow simulation of the relative amplitude and phase of the transfer function poles.

This program was written by Capt. T. J. Murphy and Capt. P. C. Reddy and uses the subroutine POLY written by Prof. M. A. Morgan.

B. PROGRAM LISTING

```
      PROGRAM HCOEF
C
      REAL*8 T, A(0:100), B(0:2),NUM1(0:100),ATMP(0:100),NUM2(0:100)
      REAL*8 SPOLR(30), SPOLI(30),PI,MAG(30),PHASE(30),C(30),D(30)
      INTEGER  NPTS, NR, N
      COMPLEX*16  SPOL(30), ZPOL(30)
      CHARACTER*16 FNAME, PFNAME

      PI=3.14159265
      WRITE(*,*) 'Enter filename for s-plane poles '
      READ(*,120) PFNAME
```

```
      OPEN(1,FILE=PFNAME)
      READ(1,130) NR
      WRITE(*,*) 'Enter time interval'
      READ(*,*) T
      WRITE(*,*) 'Enter number of points'
      READ(*,*) NPTS
      DO 10 I= 1,NR
      READ(1,*) SPOLR(I)
      READ(1,*) SPOLI(I)
      SPOL(I)=DCMPLX(SPOLR(I),SPOLI(I))
      ZPOL(I)=CDEXP(DCMPLX(T/FLOAT(NPTS-1))*SPOL(I))
      write(*,*) 'z-plane pole =',zpol(i)
      WRITE(*,*) 'Enter magnitude of pole',I
      READ(*,*) MAG(I)
      WRITE(*,*) 'Enter phase (in degrees) of pole',I
      READ(*,*) PHASE(I)
      PHASE(I)=0.0
      MAG(I)=1.0
C     CONVERT MAG AND PHASE TO RECTANGULAR COORD
      C(I)=MAG(I)*COS(PHASE(I))
      D(I)=MAG(I)*SIN(PHASE(I))
10    CONTINUE
      A(0)=1.0
      A(1)=-2.0*REAL(ZPOL(1))
      A(2)=(REAL(ZPOL(1)))**2+(AIMAG(ZPOL(1)))**2
      NUM1(0)=2.0*C(1)
      NUM1(1)=-2.0*(REAL(ZPOL(1))*C(1)+AIMAG(ZPOL(1))*D(1))
      NUM1(2)=0.0
      N=2
      DO 15 I=2,NR
      NUM2(0)=2.0*C(I)
      NUM2(1)=-2.0*(REAL(ZPOL(I))*C(I)+AIMAG(ZPOL(I))*D(I))
      NUM2(2)=0.0
      B(0)=1.0
      B(1)=-2.0*REAL(ZPOL(I))
      B(2)=(REAL(ZPOL(I)))**2+(AIMAG(ZPOL(I)))**2
      DO 17 K=0,N
      ATMP(K)=A(K)
17    CONTINUE
C     CALCULATE NEW DENOMINATOR
      CALL POLY(A,B,N)
C     CALCULATE NEW NUMERATOR
      N=N-2
      CALL POLY(NUM1,B,N)
      N=N-2
      CALL POLY(ATMP,NUM2,N)
      DO 18 L=0,N
      NUM1(L)=NUM1(L)+ATMP(L)
18    CONTINUE
15    CONTINUE
```

```
      WRITE(*,*) 'Enter name of denominator coefficient output file'
      READ(*,120) FNAME
      OPEN(3,FILE=FNAME)
      WRITE(3,130) N
      DO 20 I=1,N
      WRITE(3,*) A(I)
20    CONTINUE
      CLOSE(3)
      WRITE(*,*) 'Enter name of numerator coefficient output file'
      READ(*,120) FNAME
      OPEN(3,FILE=FNAME)
      WRITE(3,130) N
      DO 21 I=0,N
      WRITE(3,*) NUM1(I)
21    CONTINUE
      CLOSE(3)
110   FORMAT(E16.10)
120   FORMAT(A)
130   FORMAT(I5)
      END
C
      SUBROUTINE POLY(A,B,N)
C
C     Multiplying  {B(0)z**2 + B(1)*z + B(2)} x
C     {A(0)z**N + A(1)*z**(N-1) + A(2)*z**(N-2) + .... + A(N)} =
C     C(0)z**(N+2) + C(1)*z**(N+1) + C(2)*z**N + ....  + C(N+2)
C
C     Computing C(n) coefficients and storing in A(n) while
C     incrementing N --> N + 2
C
      REAL*8 A(0:100),B(0:2),C(0:100)
C     A(0)=1.0
C     Initialize on first call to routine
C     IF(N.GE.2) GO TO 11
C     N=2
C     A(1)=B(1)
C     A(2)=B(2)
C     GO TO 44
11    C(0)=B(0)*A(0)
      C(1)=B(0)*A(1)+B(1)*A(0)
      DO 22 I=2,N
22    C(I)=B(0)*A(I)+A(I-1)*B(1)+A(I-2)*B(2)
      C(N+1)=A(N)*B(1)+A(N-1)*B(2)
      C(N+2)=A(N)*B(2)
      N=N+2
      DO 33 I=0,N
33    A(I)=C(I)
44    CONTINUE
      RETURN
      END
```

APPENDIX C. ARMA SIGNAL GENERATOR

A. PROGRAM DESCRIPTION

This programs generates the time response of an ARMA system via the equation

$$y(n) = \sum_{k=1}^{k=N} a_k y(n-k) + \sum_{k=0}^{k=L} b_k x(n-k) \tag{C.1}$$

where $x(n)$ is the input excitation record, a_k and b_k are the coefficients determined by the program in Appendix B, and $y(n)$ is the output data record.

This program was written by Prof. M. A. Morgan.

B. PROGRAM LISTING

```
      Program ARMA2
C
C     Computing y[n] response for N-th order ARMA filter due to x[n]
C     input, with coefficients:
C
C               IIR  a(k) for k=1,N
C               FIR  b(k) for k=0,L
C
C     by M.A. Morgan 5/23/90
C
      REAL*8 x(0:2047),y(0:2047),a(1:30),b(0:32)
      CHARACTER*64 TITLE,TITL
      CHARACTER*16 FNAME
C     Entering D.E. Coefficients
      WRITE(*,*) 'Enter Filename For Recursive a(k):'
      WRITE(*,*) 'Use 0 for FIR Filter'
      READ(*,100) FNAME
      IF(FNAME.EQ.'0') GO TO 15
      OPEN(2,FILE=FNAME)
C     Assuming that a(k) are reversed polarity in file.
      READ(2,110) N
```

```
      write(*,*) 'Write k, a(k):'
      DO 11 k=1,N
      READ(2,*) a(k)
      a(k)=-a(k)
      write(*,*) k,a(k)
11    CONTINUE
      CLOSE(2)
15    CONTINUE
      WRITE(*,*) 'Enter Filename for Non-Recursive b(k):'
      READ(*,100) FNAME
      OPEN(2,FILE=FNAME)
      READ(2,110) L
      write(*,*) 'Write k, b(k):'
      DO 22 k=0,L
      READ(2,*) b(k)
      write(*,*) k,b(k)
22    CONTINUE
      CLOSE(2)
      WRITE(*,*) 'Enter Filename for x[n] Plot File:'
      READ(*,100) FNAME
      OPEN(2,FILE=FNAME)
      READ(2,100) TITL
      READ(2,110) NX
      READ(2,120) XQ
      READ(2,120) XQ
      write(*,*) 'Write m, x(m):'
      DO 24 m=0,NX-1
      READ(2,120) x(m)
      write(*,*) m,x(m)
24    CONTINUE
      CLOSE(2)
C     Setting up Plot File for y[n]
      WRITE(*,*) 'Enter Number of Points: (.le. 2048)'
      READ(*,*) NY
      WRITE(*,*) 'Enter time interval (nanoseconds)'
      READ(*,*) Tmax
      Tmin=0.0
      WRITE(*,*) 'Enter filename for y[n] response output'
      READ(*,100) FNAME
      TITLE= 'ARMA Filter y[n] Response'
      OPEN(1,FILE=FNAME)
      WRITE(1,100) TITLE
      WRITE(1,110) NY
      WRITE(1,120) Tmin
      WRITE(1,120) Tmax
C     Initializing then iterating to form y[m]
      y(0)=b(0)*x(0)
      write(1,120) y(0)
      DO 44 m=1,NY-1
      y(m)=0.0
```

```
      Lmax=MIN(m,L)
      DO 27 k=0,Lmax
27    y(m)=y(m)+b(k)*x(m-k)
      Nmax=MIN(m,N)
      DO 33 k=1,Nmax
33    y(m)=y(m)+a(k)*y(m-k)
      write(*,*) y(m)
      WRITE(1,120) y(m)
44    CONTINUE
      WRITE(*,*) 'Thats All Folks !'
      CLOSE(1)
100   FORMAT(A)
110   FORMAT(I5)
120   FORMAT(E12.6)
      STOP
      END
```

APPENDIX D. THE CADZOW-SOLOMON POLE EXTRACTION ALGORITHM

A. PROGRAM DESCRIPTION

This program implements the Cadzow-Solomon pole finding algorithm as described in Chapter II. The program was written by Capt. P.D. Larison with modifications by Capt. T.J. Murphy. The system order estimating subroutine is included in the next appendix. For information on the remaining subroutines see [4].

B. PROGRAM LISTING

```
$LARGE
      INTEGER I,J,K,L/1/,N,IERR,kd,m,MN,magpol(2),NSTRTPT,DELTAY
      INTEGER IER,NCAUS,NMENU,NZPOL,INSTRTPT
      INTEGER*2 KdPLT
      REAL*8 A(70,70),W(70),U(70,70),V(70,70),RV1(70)
      REAL*8 VS(70,70),UT(70,70),icomp(70),VT(70,70)
      REAL*8 AINV(70,70),dist(20),X(70),US(70,70)
      REAL*8 XP(70),B(70),SIGMA(70,70),SIG(70,70)
      REAL*8 COF(70)
      REAL*8  ROOTR(70),ROOTI(70),RRTMAX,IRTMAX,RRTMIN,IRTMIN
      REAL    CRTR(70),CRTI(70),NRTR(70),NRTI(70),MAG
      REAL*8 D(1024),avg,machep/1.0E-16/,Dy(140),Dx(1024)
      REAL*8 avgdist/0.0d0/,distmin/1000.d0/,dmin/1000.0d0/
      COMPLEX*16 S(70),truzpol(20),pdmin(10)
      LOGICAL MATU/.TRUE./,MATV/.TRUE./,CAUSAL/.TRUE./,LONG/.TRUE./
      LOGICAL DSET/.FALSE./,NUFILE/.TRUE./
      CHARACTER TITLE*16,header*64,yn*1,dc*1,TITLER*16,TITLEI*16
      CHARACTER TITL*16,TITLD*16,RPOL*16,IPOL*16,AUTORD*1

14    IF (DSET) CLOSE(10)
      NOVERLAY=0
      OPEN(10,FILE='PLOT')
      IF (DSET) GO TO 232
      WRITE (*,*) 'Welcome to signal processing using the'
      WRITE (*,*) 'Cadzow-Solomon method'
```

```
      WRITE (*,*) ' '
      WRITE (*,*) 'Do you want '
      WRITE (*,*) ' '
      WRITE (*,*) '1. The long version for beginners'
      WRITE (*,*) '2. The short version for pros'
      WRITE (*,*) ' '
118   WRITE (*,*) 'Please enter 1 or 2 '
      READ (*,*) N
      IF (N .EQ. 1) THEN
      LONG=.TRUE.
      ELSEIF (N .EQ. 2) THEN
      LONG=.FALSE.
      ELSE
      GO TO 118
      ENDIF
      WRITE (*,*) 'Enter name of data file with s-plane poles'
      READ(*,100) TITL
      OPEN(9,FILE=TITL)
      READ(9,*) NZPOL
      DO 77 I=1,NZPOL
      READ(9,*) ICOMP(1)
      READ(9,*) ICOMP(2)
      TRUZPOL(I)=CDEXP(DCMPLX(20.0d0/1023.0d0)*DCMPLX(ICOMP(1),ICOMP(2))
     + )
      WRITE(*,*) 'True z-plane pole #',I
      WRITE(*,*) TRUZPOL(I)
77    CONTINUE
      CLOSE(9)
      WRITE (*,*) 'Session will begin with entry of parameters needed fo
     + r processing'
      WRITE (*,*)
      WRITE (*,*) 'Do you want to enter parameters from'
      WRITE (*,*) ' '
      WRITE (*,*) '1. The keyboard'
      WRITE (*,*) '2. A previously created file of parameters'
      WRITE (*,*) ' '
19    WRITE (*,*) 'Please enter 1 or 2 '
      READ (*,*) N
      IF (N .EQ. 1) THEN
      GO TO 8
      ELSEIF (N .EQ. 2) THEN
13    WRITE (*,*) 'Enter title of file containing parameters'
      READ (*,100) TITL
      OPEN(1,FILE=TITL)
      READ(1,100) TITLE
      READ(1,110) NPTS
      READ(1,110) NRT
      READ(1,110) Kd
      READ(1,110) M
      READ(1,110) DELTAY
```

```
      READ(1,110) NSTRTPT
      READ(1,110) NCAUS
      READ(1,100) TITLD
      READ(1,110) NDPTS
      READ(1,110) Kn
      READ(1,110) INSTRTPT
      CLOSE(1)
      GO TO 232
      ELSE
      GO TO 19
      ENDIF
      WRITE (*,*) ' '

8     WRITE (*,*) 'Enter title of file containing excitation waveform'
      READ (*,100) TITLD
      OPEN(8,FILE=TITLD)
      READ(8,100) HEADER
      READ(8,110) N
      IF (N .GT. 1024) THEN
      WRITE (*,*) 'Number of points in data file exceeds the dimension'
      WRITE (*,*) 'of the array used in the program to store the file'
      STOP
      ENDIF
      CLOSE(8)
      IF ((N .GE. NDPTS) .AND. DSET) THEN
      NDPTS=N
      GO TO 232
      ENDIF
      NDPTS=N

9     WRITE (*,*) 'Enter estimated feed forward order'
      IF (DSET) THEN
      MAXIMUM=NDPTS-M
      IF (MAXIMUM .GT. M-Kd-1) MAXIMUM=M-Kd-1
      IF (MAXIMUM .GT. NDPTS-INSTRTPT-Kn-M+1) THEN
      MAXIMUM=NDPTS-INSTRTPT-Kn-M+1
      ENDIF
      ELSE
      MAXIMUM=66
      ENDIF
      IF (MAXIMUM .EQ. 1) THEN
      WRITE (*,*) 'The estimated feed forward order can only be 1'
      IF (DSET) GO TO 232
      GO TO 10
      ELSE
      IF (DSET) THEN
      WRITE (*,*) 'Given the other parameters chosen thus far,'
      ENDIF
411   WRITE (*,*) 'the order may range from                1'
      WRITE (*,*) '                          to',MAXIMUM
```

```
      READ (*,*) Kn
      IF (Kn .GE. 1 .AND. Kn .LE. MAXIMUM) THEN
      IF (DSET) GO TO 232
      GO TO 10
      ENDIF
      WRITE (*,*) 'Enter estimated order again'
      WRITE (*,*) ' '
      GO TO 411
      ENDIF
      IF (DSET) GO TO 232

10    INSTRTPT=1
412   IF (INSTRTPT+Kn+M-1 .GT. NDPTS) THEN
      INSTRTPT=INSTRTPT-1
      ELSE
      INSTRTPT=INSTRTPT+1
      GO TO 412
      ENDIF
      MSTRT=INSTRTPT

      IF (INSTRTPT .EQ. 1) THEN
      WRITE (*,*) 'The first point can only be 1'
      GO TO 232
      ELSE
      WRITE (*,*) 'Enter first point in waveform file to be processed'
413   WRITE (*,*) 'Given the other parameters chosen thus far,'
      WRITE (*,*) 'the starting point may range from          1'
      WRITE (*,*) '                                  to',MSTRT
      READ (*,*) INSTRTPT
      IF (INSTRTPT .GE. 1 .AND. INSTRTPT .LE. MSTRT) THEN
      IF (DSET) GO TO 232
      GO TO 1
      ENDIF
      WRITE (*,*) 'Enter starting point again'
      WRITE (*,*) ' '
      GO TO 413
      ENDIF
      IF (DSET) GO TO 232

1     IF (.NOT. DSET) NUFILE=.TRUE.
      IF (.NOT. DSET) NSTRTPT=1
      WRITE (*,*) 'Enter title of data file to be read'
      READ (*,100) TITLE
      OPEN(12,FILE=TITLE)
      READ(12,100) HEADER
      READ(12,110) NPTS
      IF (NPTS .GT. 1024) THEN
      WRITE (*,*) 'Number of points in data file exceeds the dimension'
      WRITE (*,*) 'of the array used in the program to store the file'
      STOP
```

```
      ENDIF
      CLOSE(12)

      IF (NUFILE) THEN
      GO TO 3
      ELSEIF (NSTRTPT+(Kd+M-1)*DELTAY .LE. NPTS) THEN
      GO TO 232
      ELSE
      GO TO 6
      ENDIF


3     IF (NUFILE) THEN
      MAXIMUM=69-Kn-1
      IF (MAXIMUM .GT. NPTS-69) MAXIMUM=NPTS-69
      MIN=2
      IF (MIN .EQ. MAXIMUM) THEN
      Kd=MIN
      WRITE (*,*) 'Given the other parameters chosen thus far,'
      WRITE (*,*) 'Kd must be ',MIN
      GO TO 4
      ENDIF

      WRITE (*,*) 'Enter Kd, >= the estimated order of the system '
      WRITE (*,*) 'Given the other parameters chosen thus far,'
34    WRITE (*,*) 'Kd may range from',MIN
      WRITE (*,*) '                 to',MAXIMUM
      READ (*,*) Kd
      IF (Kd .GE. MIN .AND. Kd .LE. MAXIMUM) GO TO 4
      GO TO 34

      ELSEIF (DSET) THEN
      MAXIMUM=M-Kn-1
      IF (MAXIMUM .GT. NPTS-M) MAXIMUM=NPTS-M
      MIN=2
      N=MAXIMUM
17    IF (NSTRTPT+(N+M-1)*DELTAY .LE. NPTS) THEN
      MAXIMUM=N
      IF (MIN .EQ. MAXIMUM) THEN
      Kd=MIN
      GO TO 232
      ELSEIF (MAXIMUM .LT. MIN) THEN
      DELTAY=1
      IF (1+(2+M-1)*DELTAY .LE. NPTS) THEN
      Kd=2
      GO TO 137
      ENDIF
      WRITE (*,*) 'Error. Kd must be less than 2'
      Kd=2
      GO TO 232
```

```
      ENDIF
      WRITE (*,*) 'Given the other parameters chosen thus far,'
22    WRITE (*,*) 'Kd may range from                ',MIN
      WRITE (*,*) '                          to',MAXIMUM
      WRITE (*,*) 'Enter Kd'
      READ (*,*) Kd
      IF (Kd .GE. MIN .AND. Kd .LE. MAXIMUM) GO TO 232
      GO TO 22
      ELSE
      N=N-1
      GO TO 17
      ENDIF
      ENDIF

C     Determine M
4     IF (NUFILE) THEN
      WRITE (*,*) 'Enter M, the row dimension of the data matrix'
      IF (.NOT. DSET .AND. LONG) THEN
      WRITE (*,*) ' '
      WRITE (*,*) 'Note: Kd+M points in ',title
      WRITE (*,*) '      will be processed '
      WRITE (*,*) ' '
      ENDIF
320   WRITE (*,*) 'M may range from',Kd
      IF (NPTS-Kd .GT. 69) THEN
      WRITE (*,*) '              to        69'
      ELSE
      WRITE (*,*) '              to',NPTS-Kd
      ENDIF
      READ (*,*) M
      IF (M .GT. 69) THEN
      WRITE (*,*) 'M must also be less than 70'
      GO TO 320
      ELSEIF (M .LT. Kd) THEN
      WRITE (*,*) 'M must be greater than or equal to Kd, Kd= ',Kd
      GO TO 320
      ELSEIF (Kd+M .GT. NPTS) THEN
      WRITE (*,*) 'Kd+M must be less than or equal to',NPTS,','
      WRITE (*,*) 'the number of data points in',TITLE
      WRITE (*,*) ' '
      GO TO 320
      ENDIF
C     Begin part for data already set
      ELSE
      N=Kd
122   IF (NSTRTPT+(Kd+N-1)*DELTAY .LE. NPTS) THEN
      N=N+1
      GO TO 122
      ELSE
      N=N-1
```

```
      ENDIF
      IF (N .EQ. Kd) THEN
      WRITE (*,*) 'M must equal',Kd
      M=Kd
      GO TO 232
      ENDIF
      MAXIMUM=N
      IF (MAXIMUM .GT. 69) MAXIMUM=69
      IF (Kd+Kn+1 .EQ. MAXIMUM) THEN
      M=Kd+Kn+1
      GO TO 232
      ELSEIF (Kd+Kn+1 .GT. MAXIMUM) THEN
      WRITE (*,*) 'Kd must be reduced'
      GO TO 3
      ELSE
      MIN=Kd+Kn+1
      ENDIF
      IF (MIN .LT. Kn+Kd+1) MIN=Kn+Kd+1
18    WRITE (*,*) 'M may range from',MIN
      WRITE (*,*) '              to',MAXIMUM
      WRITE (*,*) 'Enter M'
      READ (*,*) M
      IF (M .GE. MIN .AND. M .LE. MAXIMUM) GO TO 232
      GO TO 18
      ENDIF

c     Determine DELTAY
137         IF (.NOT. NUFILE) GO TO 232
5     N=1
99    IF (NSTRTPT+N*(Kd+M-1) .LE. NPTS) THEN
      N=N+1
      GO TO 99
      ELSE
      N=N-1
      ENDIF
      IF (N .EQ. 1) THEN
      WRITE (*,*) 'Given the other parameters chosen thus far,'
      WRITE (*,*) 'Spacing can only be 1'
      DELTAY=1
      IF (NUFILE) THEN
      GO TO 577
      ELSE
      GO TO 232
      ENDIF
      ENDIF
      IF (.NOT. DSET .AND. LONG) THEN
      WRITE (*,*) 'Enter spacing between the ',Kd+M
      WRITE (*,*) 'data points of ',TITLE
      WRITE (*,*) 'to be processed '
      WRITE (*,*) ' '
```

```
      WRITE (*,*) 'If, for example, one is chosen, then ',Kd+M
      WRITE (*,*) 'consecutive points in ',TITLE
      WRITE (*,*) 'will be processed '
      WRITE (*,*) ' '
      ELSE
      WRITE (*,*) 'Enter spacing '
      WRITE (*,*) ' '
      ENDIF
199   WRITE (*,*) 'Spacing may range from          1 '
      WRITE (*,*) '                    to',N
      READ (*,*) DELTAY
      IF (DELTAY .GE. 1 .AND. DELTAY .LE. N) THEN
      IF (NUFILE) THEN
      GO TO 577
      ELSE
      GO TO 232
      ENDIF
      ELSE
      GO TO 199
      ENDIF

577   WRITE (*,*) 'Do you wish to adjust eigenvalues? (y/n)'
      READ (*,150) YN
      IF (YN .EQ. 'N' .OR. YN .EQ. 'n') THEN
      IF (NUFILE) GO TO 6
      GO TO 232
      ENDIF
      IF (YN .NE. 'Y' .AND. YN .NE. 'y') GO TO 577
2     WRITE (*,*) 'Discard or compensate eigenvalues? (d/c)'
      READ (*.150) DC
      IF (DC .EQ. 'D' .OR. DC .EQ. 'd') GO TO 73
      IF (DC .NE. 'C' .AND. DC .NE. 'c') GO TO 2
73    WRITE (*,*) 'Do you want computer estimation of system order (y/n)
     +'
      READ (*,150) AUTORD
      IF (AUTORD .EQ. 'Y' .OR. AUTORD .EQ. 'y') THEN
      IF (NUFILE) GO TO 6
      GO TO 232
      ENDIF
      IF (AUTORD .NE. 'N' .AND. AUTORD .NE. 'n') GO TO 73
      WRITE (*,*) 'Enter estimate of the actual order of the system'
      WRITE (*,*) ' '
      IF (LONG) THEN
      WRITE (*,*) 'This estimate will be used to determine the '
      WRITE (*,*) 'number of eigenvalues compensated or discarded '
      ENDIF
71    WRITE (*,*) 'the estimate may range from            2'
      WRITE (*,*) '                          to',Kd+Kn+1
      READ (*,*) NRT
      IF (NRT .GT. Kd+Kn+1 .OR. NRT .LT. 2) THEN
```

```
      GO TO 71
      ELSEIF (.NOT. NUFILE) THEN
      GO TO 232
      ENDIF

6     NSTRTPT=1
74    IF (NSTRTPT+(Kd+M-1)*DELTAY .LE. NPTS) THEN
      NSTRTPT=NSTRTPT+1
      GO TO 74
      ELSE
      NSTRTPT=NSTRTPT-1
      ENDIF
      IF (NSTRTPT .EQ. 1) THEN
      WRITE (*,*) 'Given the other parameters chosen thus far,'
      WRITE (*,*) 'the starting point for processing the data'
      WRITE (*,*) 'must be the first point in the data file'
      GO TO 232
      ENDIF
      WRITE (*,*) 'Enter desired starting point in data file'
      IF (.NOT. DSET .AND. LONG) THEN
      WRITE (*,*) '1 indicates the first point in the data file '
      ENDIF
      WRITE (*,*) ' '
      WRITE (*,*) 'Given the other parameters chosen thus far,'
747   WRITE (*,*) 'the starting point may range from          1'
      WRITE (*,*) '                                 to',NSTRTPT
      READ (*,*) N
      IF (N .GE. 1 .AND. N .LE. NSTRTPT) THEN
      NSTRTPT=N
      ELSE
      WRITE (*,*) 'Enter starting point again'
      WRITE (*,*) ' '
      GO TO 747
      ENDIF
      IF (.NOT. NUFILE) GO TO 232

7     IF (DSET) THEN
      IF (NCAUS .EQ. 1) THEN
      NCAUS=2
      GO TO 232
      ELSE
      NCAUS=1
      GO TO 232
      ENDIF
      ENDIF
      WRITE (*,*) 'Do you want the data matrix arrangement to be'
      WRITE (*,*) ' '
      WRITE (*,*) '1. Causal'
      WRITE (*,*) '2. Non-causal'
      WRITE (*,*) ' '
```

```
181   WRITE (*,*) 'Please enter 1 or 2 '
      READ (*,*) NCAUS
      IF (NCAUS .EQ. 1) THEN
      CAUSAL=.TRUE.
      ELSEIF (NCAUS .EQ. 2) THEN
      CAUSAL=.FALSE.
      ELSE
      GO TO 181
      ENDIF
      GO TO 232


12    IF (AUTORD .EQ. 'Y' .OR. AUTORD .EQ. 'y') THEN
      IF (NUFILE) THEN
      WRITE(*,*) 'System order must be calculated first'
      GO TO 232
      ENDIF
      ENDIF
      WRITE (*,*) 'Enter title of file to contain parameters'
      READ (*,100) TITL
      OPEN(1,FILE=TITL)
      WRITE(1,100) TITLE
      WRITE(1,110) NPTS
      WRITE(1,110) NRT
      WRITE(1,110) Kd
      WRITE(1,110) M
      WRITE(1,110) DELTAY
      WRITE(1,110) NSTRTPT
      WRITE(1,110) NCAUS
      WRITE(1,100) TITLD
      WRITE(1,110) NDPTS
      WRITE(1,110) Kn
      WRITE(1,110) INSTRTPT
      CLOSE(1)
      IF (DSET) GO TO 232

15    IF (DSET) THEN
      CLOSE(2)
      CLOSE(3)
      CALL SUBPLT(NOVERLAY,MAG)
      ENDIF

232   DSET=.TRUE.
      NUFILE=.FALSE.
      WRITE(*,*) ' '
      WRITE(*,*) '1. Data file to be processed                    ',T
     + ITLE
      WRITE(*,*) '  Number of data points in data file          ',NPTS
      IF(AUTORD .EQ. 'Y' .OR. AUTORD .EQ. 'y') THEN
      WRITE(*,*) '2. Automated system order determination'
```

```
      ELSE
      WRITE(*,*) '2. Estimated order of the system              ',NRT
      ENDIF
      WRITE(*,*) '3. Kd, the number of columns in the data matrix',Kd
      WRITE(*,*) '4.  M, the number of  rows   in the data matrix',M
      WRITE(*,*) '5. Spacing between data points being processed ',DELTA
     + Y
      WRITE(*,*) '6. First point in the data file to be processed',NSTRT
     + PT
      WRITE(*,*) '    Last point in the data file to be processed',NSTRT
     + PT+Kd+M-1
      IF (NCAUS .EQ. 1) THEN
      WRITE(*,*) '7. Data matrix arrangement for processing          CA
     + USAL   '
      ELSE
      WRITE(*,*) '7. Data matrix arrangement for processing       NON-CA
     + USAL   '
      ENDIF
      WRITE(*,*) ' '

      WRITE(*,*) '8. File containing excitation waveform            ',T
     + ITLD
      WRITE(*,*) '   Number of data points in above file         ',NDPTS
      WRITE(*,*) '9. Estimated order of the waveform              ',Kn

      WRITE(*,*) '10. First point in the file to be   '
      WRITE(*,*) '    input into the data matrix                 ',INSTR
     + TPT
      WRITE(*,*) ' '

      WRITE(*,*) '11. Begin processing using above settings'
      WRITE(*,*) '12. Store parameters 1-10 in a file'
      WRITE(*,*) '13. Retrieve parameters 1-10 from a previously created
     +  file'
      WRITE(*,*) '14. Reset overlays'
      WRITE(*,*) '15. Re-plot overlays'
      WRITE(*,*) '16. End this session of Cadzow-Solomon signal processi
     + ng'
      WRITE(*,*) ' '
      WRITE(*,*) 'Enter an integer from 1 to 16 to make changes as often
     +  as you desire'
200   READ (*,*) NMENU
      IF (NMENU .LT. 1 .OR. NMENU .GT. 16) THEN
      WRITE(*,*) 'Enter an integer from 1 to 16'
      GO TO 200
      ENDIF

      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16),NMENU

11    OPEN(12,FILE=TITLE)
```

```
      READ(12,100) HEADER
      READ(12,110) NPTS
      READ(12,120) XQ
      READ(12,120) XQ
      DO 2337 I=1,NPTS
      READ(12,120) D(I)
2337      CONTINUE
      CLOSE(12)

      OPEN(8,FILE=TITLD)
      READ(8,100) HEADER
      READ(8,110) NDPTS
      READ(8,120) XQ
      READ(8,120) XQ
      DO 414 I=1,NDPTS
      READ(8,120) Dx(I)
414       CONTINUE
      CLOSE(8)

      KdPLT=Kd
      WRITE(*,*) 'enter title of file to contain real part of poles'
      READ(*,100) TITLER
      OPEN(2,file=TITLER)


      WRITE(*,*)'enter title of file to contain imaginary part of poles'
      READ(*,100) TITLEI
      OPEN(3,file=TITLEI)

      WRITE(10,130) (KdPLT)
      WRITE(10,100) TITLER
      WRITE(10,100) TITLEI
130   FORMAT(I2)

      MN=MAX(M,Kd+Kn+1)

100   FORMAT(A)
110   FORMAT(I5)
120   FORMAT(E12.6)
150   FORMAT(A1)

      DO 39 I=1,Kd+M
      Dy(I)=D((I-1)*DELTAY+NSTRTPT)
39    CONTINUE

237   DO 227 I=1,M
      DO 127 J=1,Kd+Kn+1
      A(I,J)=Dy(I+J+1-IJ)
      IF (J .GE. Kd+1) A(I,J)=Dx(I+J+INSTRTPT-2-Kd)
127   CONTINUE
```

```
227   CONTINUE

      B(1)=Dy(1)
      DO 449 I=2,M
      B(I)=a(I-1,1)
449   CONTINUE

      N=Kd+Kn+1

      CALL SVD(MACHEP,M,N,MN,A,W,MATU,U,MATV,V,IERR,RV1)

C     ANY ERRORS?
      IF (IERR .GT. 0.0) THEN
      WRITE (*,*) 'Error in singular value number ',IERR,STOP
      ENDIF
      IF (YN .EQ. 'N') GO TO 2227

      DO 1127 I=1,Kd+Kn+1
      XP(I)=0.0
1127  CONTINUE

c     COMPENSATE EIGENVALUES
c     ORDER SINGULAR VALUES
      XP(1)=W(1)
      DO 119 I=2,Kd+Kn+1
      DO 1179 J=1,I
      IF (W(I) .GT. XP(J)) then
      DO 123 K=I+1,J,-1
123   XP(K)=XP(K-1)
      XP(J)=W(I)
      GO TO 119
      ENDIF
1179  CONTINUE
      XP(I+1)=W(I)
119   CONTINUE
c     XP( ) now contains ordered singular values: xp(1) is the largest


      IF (AUTORD .EQ. 'Y' .OR. AUTORD .EQ. 'y') CALL FNDAIC(Kd,xp,m,nrt)
      IF (DC .EQ. 'D') THEN
      DO 112 J=NRT+1,Kd+Kn+1
112   W(j)=(0.0)
      ELSE
c     COMPENSATE
      AVG=0.0
      DO 111 J=NRT+1,Kd+Kn+1
      AVG=AVG+XP(j)**2
111   CONTINUE
      IF (Kd+Kn+1 .GT. NRT) AVG=AVG/DBLE(FLOAT(Kd+Kn+1-NRT))
      DO 132 J=1,Kd+Kn+1
```

```
      DO 177 K=1,Kd+Kn+1
      IF ( W(J) .EQ. XP(K) ) THEN
      IF ( K .GT. NRT ) THEN
      W(J)=0.0
      ELSE
      W(J)=DSQRT(DABS(W(J)*W(J)-AVG))
      ENDIF
      GO TO 132
      ENDIF
177   CONTINUE
132   CONTINUE
      ENDIF


c     Calculate UT, the transpose of U, an M x M matrix
2227  DO 500 I=1,M
      DO 600 J=1,M
      UT(I,J)=(U(J,I))
600   CONTINUE
500   CONTINUE


c     Form SIGMA+ (Kd+Kn+1 x M)
      DO 70 I=1,Kd+Kn+1
      DO 80 J=1,M
      SIGMA(I,J)=0.0
      IF (I .EQ. J .AND. W(J) .NE. 0.0) THEN
      SIGMA(I,J)=1.0D0/W(j)
      ELSE
      SIGMA(I,J)=0.0D0
      ENDIF
80    CONTINUE
70    CONTINUE

c     Form SIGMA (M x Kd+Kn+1)
      DO 700 I=1,M
      DO 800 J=1,Kd+Kn+1
      SIG(I,J)=0.0
      IF (I .EQ. J) SIG(I,J)=W(J)
800   CONTINUE
700   CONTINUE


c     Calculate matrix multiplication of V x SIGMA+=VS, where
c     V=Kd+Kn+1xKd+Kn+1,SIGMA+=Kd+Kn+1xM,VS=Kd+Kn+1xM
      CALL MXMUL(V,SIGMA,Kd+Kn+1,Kd+Kn+1,M,VS)

c     Caiculate matrix multiplication of VS x UT=AINV, where
c     VS=Kd+Kn+1xm,UT=mxm,AINV=Kd+Kn+1xm
      CALL MXMUL(VS,UT,Kd+Kn+1,M,M,AINV)
```

```
c       Calculate matrix multiplication of AINV x B, where
c       AINV=Kd+Kn+1xm,B=mx1,XP=Kd+Kn+1x1
        CALL MXMUL(AINV,B,Kd+Kn+1,M,L,XP)

c       Compute autoregressive coefficients from prediction coefficients
        IF (XP(Kd) .EQ. 0.0) THEN
        WRITE (*,*) 'ERROR, avoiding division by zero'
        STOP
        ELSE
        B(Kd)=1.0D0/XP(Kd)
        ENDIF
        DO 347 I=2,Kd
        b(i-1)=-b(Kd)*xp(Kd-i+1)
347     CONTINUE
c       rearrange prediction coefficients for call to POLRT

        DO 357 I=1,Kd
        X(I)=-B(Kd-I+1)
        IF (NCAUS .EQ. 1) X(I)=-XP(Kd-I+1)
        WRITE (*,*) I,XP(i),X(i)
357     CONTINUE
        X(Kd+1)=1.0

        CALL POLRT(X,COF,KD,ROOTR,ROOTI,IER)

        IF (IER .NE. 0) WRITE (*,*) 'ERROR with polrt, ier=',IER,STOP

        DO 777 I=1,Kd
        WRITE(2,120) ROOTR(I)
        WRITE(3,120) ROOTI(I)
        S(I)=DCMPLX(ROOTR(I),ROOTI(I))
777     CONTINUE

        MAGPOL(1)=0
        DO 647 I=1,Kd
        IF (CDABS(S(I)) .GE. 1.0D0) MAGPOL(1)=magpol(1)+1
647     continue

        WRITE(*,*) '# of poles with magnitude >= 1',magpol(ij)
        WRITE (*,*) 'HIT ANY KEY TO CONTINUE'
        READ (*,100) HEADER

        NOVERLAY=NOVERLAY+1
        CLOSE(2)
        CLOSE(3)
        CALL SUBPLT(NOVERLAY,MAG)
        DO 45 I=1,NZPOL
        DIST(I)=CDABS(TRUZPOL(I)-S(1))
        PDMIN(I)=S(1)
45      CONTINUE
```

```
      DO 400 I=2,Kd
      DO 401 J=1,NZPOL
      IF (CDABS(TRUZPOL(J)-S(I)) .LT. DIST(J) ) THEN
      PDMIN(J)=S(I)
      DIST(J)=CDABS(TRUZPOL(J)-S(I))
      ENDIF
401   CONTINUE
400   CONTINUE
      AVGDIST=0.0
      J=0
      DO 402 I=1,NZPOL
      J=J+1
      IF(J .EQ. 4) then
      WRITE(*,*) 'hit any key to continue'
      READ(*,100) HEADER
      J=0
      ENDIF
      WRITE(*,*) 'True z-pole    ',TRUZPOL(I)
      WRITE(*,*) 'Obtained z-pole',PDMIN(I)
      WRITE(*,*) 'Distance from true pole',DIST(I)
      AVGDIST=AVGDIST+DIST(I)
402   CONTINUE
      IF (NZPOL .EQ. 0) GO TO 404
      WRITE(*,*) 'Average distance from true poles',AVGDIST/NZPOL
      WRITE(*,*) ' '
404   WRITE(*,*) 'Poles with magnitude less than one'
      WRITE(*,*) ' '
      J=0
      K=0
      WRITE(*,*) 'Enter file to contain real poles'
      READ(*,100) RPOL
      WRITE(*,*) 'Enter file to contain imaginary poles'
      READ(*,100) IPOL
      OPEN(20,FILE=RPOL)
      OPEN(21,FILE=IPOL)
      DO 403 I=1,Kd
      IF (CDABS(S(I)) .LT. 1.0) THEN
      WRITE (*,*) S(I),CDABS(S(I))
      WRITE (20,120) REAL(S(I))
      WRITE (21,120) IMAG(S(I))
      J=J+1
      K=K+1
      ENDIF
      IF (J .EQ. 20) THEN
      WRITE (*,*) 'Hit any key to continue'
      READ (*,100) HEADER
      J=0
      ENDIF

403   CONTINUE
```

```
      CLOSE(20)
      CLOSE(21)
      WRITE (*,*) 'Poles with magnitude less than one ',K
      WRITE (*,*) 'Hit any key to continue'
      READ (*,100) HEADER

      GO TO 232

16    STOP
      END
```

APPENDIX E. DETERMINING SYSTEM ORDER

A. SUBROUTINE DESCRIPTION

This subroutine implements the Akaike Information Criterion (AIC) in order to determine the order of a system. The routine is based on the algorithm as described in Chapter II.

B. SUBROUTINE LISTING

```
      SUBROUTINE FNDAIC(Kd,XP,M,NRT)
C
C     ESTIMATES SYSTEM ORDER USING AIC CRITERION
C
      INTEGER J,Kd,NRT,STOP,M
      REAL*8 XP(70),SUM(0:69),PROD(0:69),TERM1,AIC,AICM
C
      STOP=Kd-1
      DO 20 J=0,STOP
      SUM(J)=XP(J+1)
      PROD(J)=XP(J+1)
      DO 30 K=(J+2),Kd
      SUM(J)=SUM(J)+XP(K)
      PROD(J)=PROD(J)*XP(K)
30    CONTINUE
      TERM1=FLOAT(Kd-J)*FLOAT(Kd+M-1)*ALOG(1/(FLOAT(Kd-J))*SUM(J))
      AIC=TERM1-FLOAT(Kd+M-1)*ALOG(PROD(J))+FLOAT(J*(2*Kd-J))
      IF(J .EQ. 0) THEN
      AICM=AIC
      NRT=0
      ELSEIF(AIC .LT. AICM) THEN
      AICM=AIC
      NRT=J
      ENDIF
20    CONTINUE
      WRITE(*,*) 'SYSTEM ORDER =   ', NRT
      RETURN
      END
```

LIST OF REFERENCES

1. C. E. Baum, *On the Singularity Expansion Method for the Solution of Electromagnetic Interaction Problems,* Air Force Weapons Laboratory Interaction Note 88, December 1971.

2. D. L. Moffatt and R. K. Mains, "Detection and discrimination of radar targets," *IEEE Trans. on Antennas and Propagation*, AP-23 May 1975, pp 358-367.

3. Michael A. Morgan, "Singularity expansion representations of fields and currents in transient scattering," *IEEE Trans. on Antennas and Propagation*, AP-23, May 1984, pp. 466-473.

4. Peter D. Larison, "Evaluation of system identification algorithms for aspect-independent radar target classification," Master's Thesis, Naval Postgraduate School, Monterey, CA, Dec 1989.

5. E. M. Kennaugh, "The K-pulse concept," *IEEE Trans. on Antennas and Propagation*, AP-29, March 1981 pp. 327-331.

6. James B. Dunavin, "Identification of scatterers based upon annihilation of complex natural resonances," Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1985.

7. Michael A. Morgan and James B. Dunavin, "Discrimination of scatterers using natural resonance annihilation," *Abstracts of 1986 National Radio Science Meeting*, Philadelphia, PA, June 1986.

8. K. M. Chen, D. P. Nyquist, E. J. Rothwell, L. L. Webb, and B. Drachman, "Radar target discrimination by convolution of radar returns with extinction-pulses and single-mode extraction signals," *IEEE Trans. on Antennas and Propagation*, AP-34, July 1986, pp. 896-904.

9. E. M. Kennaugh and D. L. Moffatt, "Transient and impulse response approximations," *Proc. IEEE*, 53, August 1965, pp. 893-901.

10. Michael A. Morgan, "Scatterer discrimination based upon natural resonance annihilation," *Journal of Electromagnetic Waves and Applications*, 2, 2, 1987, pp. 155-176.

11. Steven M. Kay and Stanley Lawrence Marple, Jr., "Spectrum Analysis-A Modern Perspective," *Proc. IEEE*, 69, November 1981, pp. 1380-1415.

12. Choong Y. Chong, "Investigation of non-linear estimation of natural resonances in target identification," Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1983.

13. Ramdas Kumaresan, "Estimating the parameters of exponentially damped or undamped sinusoidal signals in noise," Ph.D. Dissertation, University of Rhode Island, Kingston, RI, October 1982.

14. G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numer. Math.*, 14,1970, pp. 403-420.

15. R. Kumaresan, and D. W. Tufts, "Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise," *IEEE Trans. Acoustics, Speech, and Sig. Proc.*, December 1982, pp. 833-840.

16. S. A. Norton, "Radar target classification by natural resonances: signal processing algorithms," Engineer's Thesis, Naval Postgraduate School, Monterey, CA, Dec 1989.

17. James A. Cadzow and Otis M. Solomon, Jr., "Algebraic approach to system identification," *IEEE Trans. Acoustics, Speech, and Sig. Proc.*, AP-29, March 1981 pp. 462-469.

18. Hirotugu Akaike, "A new look at the statistical model identification," *IEEE Trans. on Automatic Control*, December 1974, pp. 716-723.

19. John Finney Aurand, "An antenna array processing system for multiple source bearing estimation," Ph.D. Dissertation, Iowa State University, Ames, IA, 1987.

20. Steven M. Kay, *Modern Spectral Estimation Theory & Application*, p. 297, Prentice Hall, Englewood Cliffs, NJ, 1987.

21. Michael A. Morgan, *Time-Domain Thin-Wire Integral Equation Program*, Naval Postgraduate School, Monterey, CA, 1989.

22. Norman J. Walsh, "Bandwidth and signal to noise ratio enhancement of the NPS transient scattering laboratory," Master's Thesis, Naval Postgraduate School, Monterey, CA, Dec 1989.

23. Peter C. Reddy, "Radar target classification by natural resonances: system analysis," Master's Thesis, Naval Postgraduate School, Monterey, CA, Sept. 1990.

24. James A. Cadzow, "Signal enhancement: a useful signal processing tool," *ASSP Workshop on Spectrum Estimation and Modeling*, pp. 162-167, IEEE Publishing Services, New York, IEEE Catalog No. 88CH2633-6, 1988.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Defense Logistic Studies Information Exchange U.S. Army Logistics Management College Fort Lee, Virginia 23801-6043	1
4.	Department Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5002	1
5.	Professor Michael A. Morgan, Code EC/Mw Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5002	5
6.	Professor Ralph Hippenstiel, Code EC/Hi Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5002	1
7.	Commandant of the Marine Corps Code TE 06 Headquarters, U.S. Marine Corps Washington, D.C. 20380-0001	1
8.	Dr. John N. Entzminger Director, Tactical Technology Office Defense Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, Virginia 22209	1