AD-A094 581 SYSTEMS CONTROL INC (VT) PALO ALTO CA F/G 17/7

MULTITARGET TRACKING STUDIES.(U)

JUL 80 B FRIEDLANDER N00014-79-C-0743

UNCLASSIFIED SCI-5334-01 NL

1 OF 1

AD A094581

END

DATE FILMED

2-81

DTIC

LEVEL II

12

SCI

SYSTEMS CONTROL, INC. ■ 1801 PAGE MILL ROAD ■ PALO ALTO, CA 94304 ■ TELEX 348-433 ■ (415) 494-1165

Report No. 5334-01

July 1980

AD A094581

MULTITARGET TRACKING STUDIES

Phase I Final Report

Prepared for:

Naval Analysis Program (Code 431)
Office of Naval Research

Under Contract:

N00014-79-C-0743
NR 277-287

Approved by:

H. M. Pearce
General Manager

Prepared by:

B. Friedlander

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: 14 SCI-5334-01	2. GOVT ACCESSION NO.: AD-A094 582
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): Multitarget Tracking Studies, Phase I Final Report	5. TYPE OF REPORT & PERIOD COVERED: Final Report 10/5/79 - 6/3/80
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Ben Friedlander 15	8. CONTRACT OR GRANT NUMBER(s): N00014-79-C-0743
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Systems Control, Inc. 1801 Page Mill Road Palo Alto, CA 94304 1411 17	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61153N RR014-11-01 NR 277-287
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Analysis Program (Code 431) Office of Naval Research Arlington, VA 22217 11 12	12. REPORT DATE: July 1980
	13. NUMBER OF PAGES: 144
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 9 Final rpt. 79-	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for publication; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

autoregressive moving average (ARMA)
parameter estimation
tracking
time difference of arrival
adaptive signal processing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report presents the results of phase I of an investigation of a new concept for tracking multiple targets. This concept is based on modeling the observed data as a multichannel ARMA process. The parameters of the model provide a compact representation of target parameters such as spectrum and TDOA/bearing. These parameters can, therefore, be used as inputs to a tracking algorithm. In phase I the basic algorithm for single target tracking was developed and tested. Results based on simulations of synthetic data → cont

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. are very promising. Preliminary theoretical analysis of the multichannel case has also been performed.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (Continued)

1. INTRODUCTION AND SUMMARY

Multitarget Tracking Studies (MTS) is a research effort which has the objectives of developing and evaluating a new concept for tracking multiple targets. The algorithms developed in this program (which will be referred to as the MTS algorithm or processor) will complement and enhance currently used tracking techniques. While the main goal of the program is directed towards multiple targets, MTS is expected to have a significant impact on the single target case as well.

In this report, we summarize the main theoretical developments and some preliminary performance evaluation results. This work is part of the first phase of the MTS research program in which the single target case was studied. Results so far have been very promising and we expect to adapt the techniques developed in this initial phase to handling multiple targets during the second phase of the program.

An important point here is the following. It has not been the objective of this study to surpass conventional estimation performance of spectrum and TDOA analyzers. The original emphasis was upon demonstration that a framework in which these functions can be carried out naturally for multiple targets is one in which performance on individual targets can be maintained. It was sufficient, therefore, to demonstrate that even for low SNR (-10dB-0dB) the MTS performance compares with conventional approaches. These approaches make their processing gains by integration, a procedure also available to MTS. Because pre-integration performance of MTS was so encouraging, no further pursuit toward comparison was undertaken. However, it turned out that the new approach to adaptive signal processing implicit in the MTS processing could be developed to one offering substantial improvement over current techniques. While our research effort is directed towards the development of tracking algorithms, the signal modeling approach has a much wider applicability to the Navy's signal processing problems. In particular, the MTS algorithm is directly applicable to a number of adaptive signal processing problems, including: line enhancement, high resolution spectral estimation, noise cancelling and channel equalization. Applying the proposed ARMA modeling techniques to some of these problems has already resulted in substantial performance improvements. The analysis of the MTS algorithm as it is used for adaptive signal processing is summarized in [10].

The MTS concept is based on modeling the observed data as an ARMA process. The parameters of the model provide a compact representation of target parameters such as spectrum and TDOA/bearing. These parameters can, therefore, be used as inputs to a tracking algorithm, a target classification program, etc. Section 2 of the report describes the MTS concept and how it fits into an overall system.

A major part of our effort has been directed towards developing and coding the basic MTS algorithm. This algorithm is a parameter estimation technique which recursively computes a set of ARMA parameters from the observed data sequence. This algorithm is now implemented as an interactive computer program and it provides a very powerful and flexible signal processing tool. This program will be the core of our future MTS work. Section 3 of the report describes the algorithm and its main features.

The main issues addressed so far are TDOA estimation and estimation of the spectral parameters of the target under different signal-to-noise ratio conditions. Several synthetic test cases, both narrowband and broadband, were used to evaluate the performance of the MTS algorithm. Results were very encouraging.

For high SNR (20dB and above) the algorithm provided excellent results, and had no problems in converging to the right spectral/TDOA parameters. In moderate SNR (0-20dB), serious convergence problems were initially experienced.

A significant amount of effort was devoted to studying and solving these problems. Our solution provides an important step in extending the range of applicability of recursive parameter estimation algorithms to low SNR situations. A number of publications on this topic are in preparation. Currently, we are able to get good spectral and TDOA estimates for SNR's in the 0-20dB range. Our experience with low SNR (-10dB-0dB) has been that performance matching or exceeding conventional approaches can be achieved. No special difficulties were observed at low SNR, but we feel that some refinements of the algorithm may improve performance even further.

The positive results obtained so far will provide the basis for our next phase of research, in which the multitarget tracking algorithm will be developed. Our research will be performed in two steps:

(i) Complete the single target tracking algorithm.

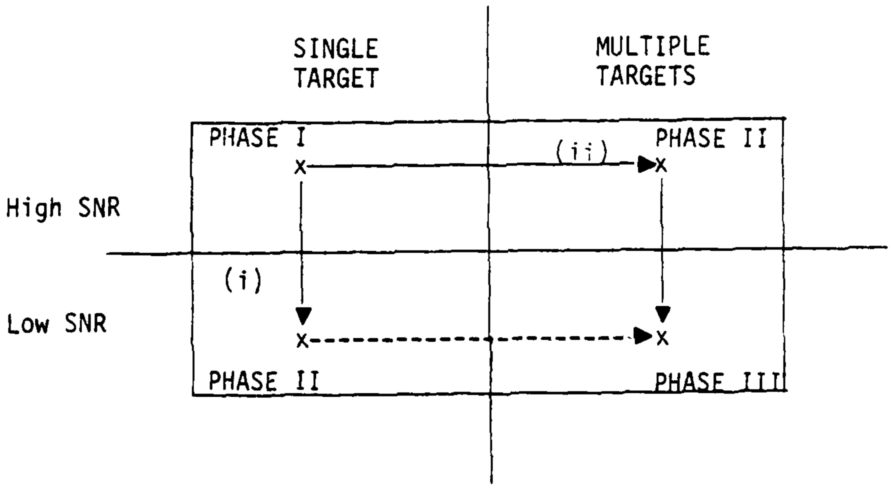
Here, we will concentrate on the issues associated with the operation of the MTS algorithm at low SNR. We will make the algorithm more robust by pre-filtering and other methods, test its tracking capability on synthetic data with time varying parameters and develop performance bounds to evaluate its performance against suitable standards.

(ii) Development and testing of multitarget algorithms for the high SNR case.

Here we will develop and evaluate a candidate algorithm for tracking several targets. The objective will be to demonstrate the capability of an MTS algorithm to provide consistent tracks for several targets. In particular, we will investigate the special structural properties of multi-input multi-output (MIMO) systems of the type used to model the multitarget tracking problem, and study questions of identifiability and uniqueness. The extension of the MTS approach to tracking multiple targets at low SNR will be deferred to a third phase of the project.

Several issues need to be investigated in order to achieve such an extension, including: the convergence of the MIMO RML algorithm, development of pre-filtering and other mechanisms for improved convergence and analysis of the uniqueness and identifiability issues under low SNR conditions.

This plan of work is summarized in the following schematic:



We believe that the next phase of the MTS project will result in significant contributions to the areas of multitarget tracking, adaptive signal processing, multichannel parameter estimation and modeling of vector time-series.

2. SYSTEM DESCRIPTION

The MTS algorithm is a coherent, time-domain signal processing technique for extracting target parameters (spectrum and TDOA/bearing) from multisensor data. The sensors may be the elements of one or several arrays. The MTS algorithm may operate directly on wideband sensor data, as depicted in Figure 1.

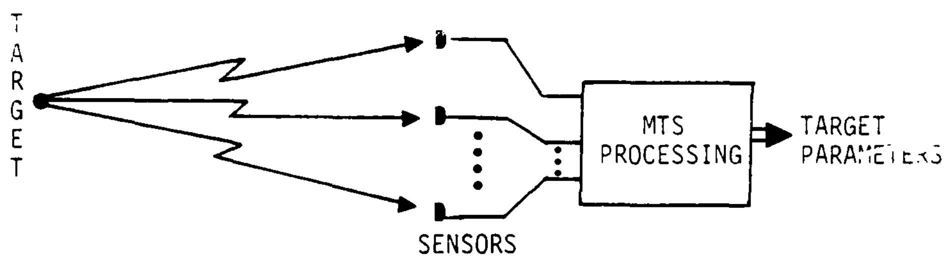


Figure 1: The MTS Processor

However, since the computational requirements of the MTS algorithm increase in proportion to the bandwidth of the input signal (as will be shown later), it is desirable to reduce the bandwidth of the sensor signals before handing them to the MTS algorithm. This can be done by a preprocessing step, in which one or more spectral bands of interest are shifted in frequency to provide a combined, relatively narrowband signal, as depicted in Figure 2.

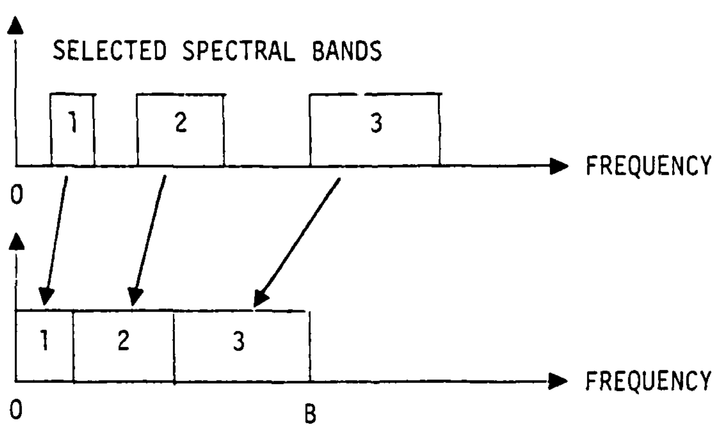
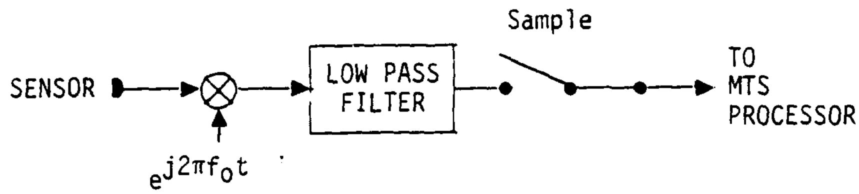
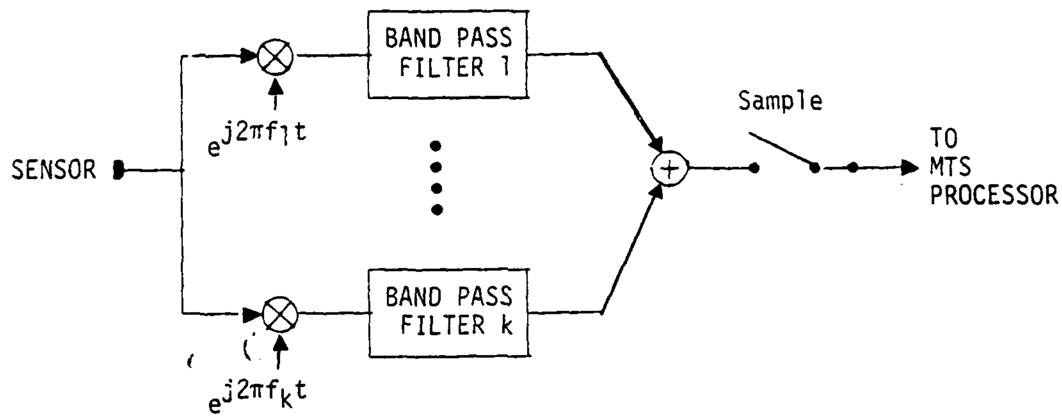


Figure 2: Bandwidth Selection

This bandwidth reduction can be implemented in many different ways. A block diagram of one possible implementation is depicted in Figure 3.



a) A Single Spectral Band (Centered at f_0)



b) Multiple Spectral Bands

Figure 3. Preprocessing for Bandwidth Reduction

In the rest of this report, we will always assume that this preprocessing step has been performed, and that the MTS processor is handed data with a bandwidth of B Hz. The data is sampled at the Nyquist rate, thus

$$\Delta T \triangleq \text{sampling interval} = 1/2B \tag{1}$$

A typical spectrum of the signal at the input of the MTS algorithm will contain several spectral lines in a noise background, as depicted in Figure 4. This spectrum was obtained by performing an N point FFT of the data where

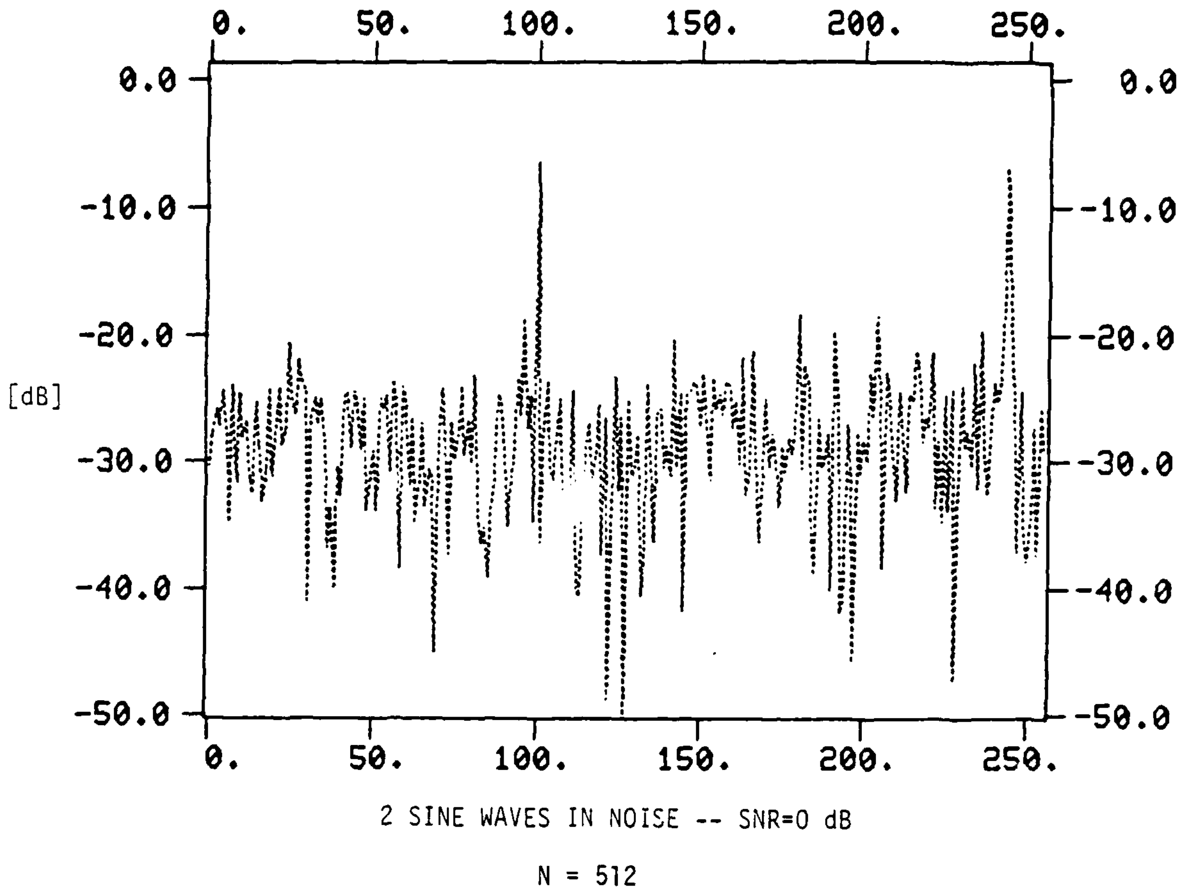


Figure 4: Typical Spectrum of Synthetic Data

$$N = \text{number of data points}, \tag{2a}$$

which corresponds to an integration time of

$$T = N\Delta T = N/2B. \tag{2b}$$

The frequency resolution of this spectral plot is Δf Hz per point, where

$$\Delta f = \frac{1}{N\Delta T} = \frac{2B}{N} . \tag{3}$$

Thus, the i-th point on the plot represents a frequency f_i, where

$$f_i = \frac{i}{N\Delta T} = \frac{2iB}{N} . \tag{4}$$

In this report, we will define the signal-to-noise ratio (SNR) as the ratio of the total signal energy to the total noise energy in the bandwidth B which is provided to the MTS processor. The signal and noise processes are generated by a synthetic data generator which produces for each sensor a data sequence $y_i(t)$.

$$y_i(t) = s_i(t) + n_i(t) \tag{5}$$

$s_i(t)$ = signal arriving at sensor i

$n_i(t)$ = measurement noise at sensor i (white Gaussian noise, independent from sensor to sensor).

The total signal and noise energies (S_i, N_i) are computed by

$$S_i = \frac{1}{N} \sum_{t=1}^{N} s_i^2(t) \tag{6a}$$

$$N_i = \frac{1}{N} \sum_{t=1}^{N} n_i^2(t) \tag{6b}$$

and the corresponding signal-to-noise ratio is given by

$$\mathrm{SNR}_i \stackrel{\Delta}{=} S_i/N_i \tag{7}$$

The noise energy is related to its spectral power density by

$$N = N_o B \tag{8}$$

where N_o is noise energy per Hz.

The MTS algorithm is based on the idea of fitting an autoregressive moving-average (ARMA) model to the observed time series (see Appendix A for a more detailed explanation). The basic model is depicted in Figure 5. The autoregressive (AR) part of the model provides information about the spectrum of the target, while the moving-average (MA) part gives the TDOA information. Thus, once an ARMA model has been fit to the observed data, all the target parameters can be obtained from the ARMA coefficients $\{a_i, B_i\}$. The spectral estimate of the target can be obtained by an FFT of the impulse response of the AR model portion*. More precisely, we can FFT the time series $x(t)$

$$x(t) = - \sum_{i=1}^{n_a} a_i \, x(t-i) + u(t) \qquad (9)$$

$$u(t) = \begin{cases} 1 & t=0 \\ 0 & t \neq 0 \end{cases}$$

The normalized estimate of the target power spectrum $S_x(i)$ is given by

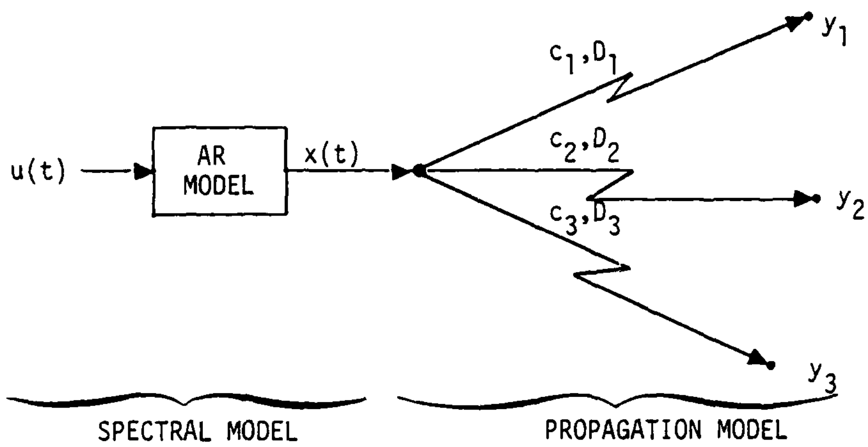
$$S_x(i) = |X_i|^2 \, / \, \left(\frac{2}{N} \sum_{i=1}^{N} |X_i|^2\right) \qquad (10)$$

where $\{X_i\}$ is the FFT of $\{x(t)\}$. The interpretation of the frequency corresponding to the i-th spectral estimate $(S_x(i))$ is given by (4).

The TDOA estimates can be obtained by looking at the b_i coefficients, as depicted in Figure 6. The TDOA corresponding to a difference of one (in order) is ΔT, where ΔT is the sampling rate of the data at the input to the MTS algorithm. This is not necessarily the ultimate resolution of our TDOA estimation, since finer resolution can be achieved by interpolation. A more detailed discussion of this point can be found in Appendix B.

---

*Note that we could also evaluate a z-transform.



c_i = Attenuation, D_i = Delay

Target:

$$x(t) = -\sum_{i=1}^{n} a_i\, x(t-i) + u(t) \qquad X(z) = \frac{1}{A(z)}\, U(z)$$

Receiver:

$$y(t) = \begin{bmatrix} c_1 x(t-D_1) \\ c_2 x(t-D_2) \\ c_3 x(t-D_3) \end{bmatrix} \qquad Y(z) = \underbrace{\begin{bmatrix} c_1 z^{-D_1} \\ c_2 z^{-D_2} \\ c_3 z^{-D_3} \end{bmatrix}}_{B(z)} X(z)$$

The Overall Model:

$$y(t) = -\sum_{i=1}^{n} a_i y(t-i) + \sum_{i=1}^{m} B_i u(t-i) \qquad Y(z) = \frac{B(z)}{A(z)}\, U(z)$$

Figure 5. An ARMA Model for the Single Target Case

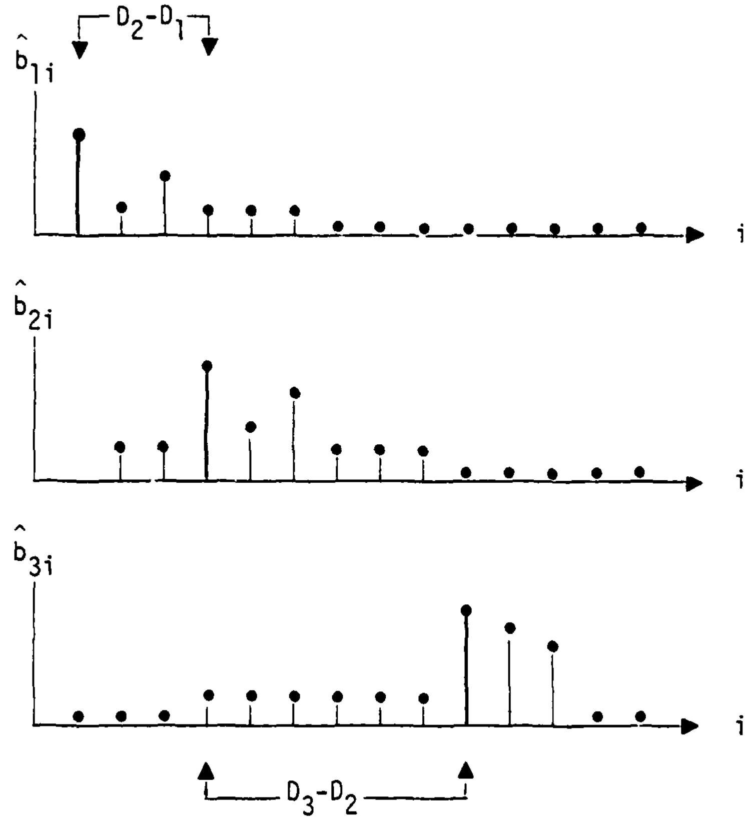


Figure 6. Estimating TDOA-s From the b_i Coefficients

A change of ΔT in the TDOA can be translated into a change of bearing $\Delta\theta$ by (see Figure 7),

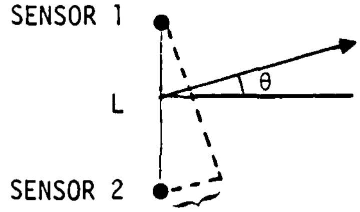
$$\Delta\theta = v\Delta T/L\cos\theta = v/2BL\cos\theta \tag{11}$$

where

L = distance between the two sensors

θ = bearing

v = sound velocity.



$$\text{TDOA} = L \sin\theta / v$$

$$\Delta T = L \cos\theta \; \Delta\theta / v$$

Figure 7: The Geometry for TDOA Computation

The total angular extent over which the MTS processor can be "steered" is given by $\Delta\theta \cdot n_b$ where n_b = the number of b coefficients, i.e., the order of the MA model. This angular extent can, of course, be increased by removing bulk delays prior to the MTS processor. If necessary, several MTS processors can be run in parallel, each covering a section of $\Delta\theta \cdot n_b$ degrees.

Consider, for example, the following representative case:

L = 1200 meters
v = 1490 m/sec
n_b = 20
B = 10 Hz

then: $\Delta\theta = 3.55°$

$\Delta\theta \cdot n_b = 71°$.

The presence of doppler shifts in the received signals will be handled in the MTS algorithm by computing a different set of $\{a_i\}$ for each sensor. In other words, different sensors will observe different (shifted) spectral lines. This feature of the algorithm has not been tested yet, but more details can be found in Section 3.

The estimated target parameters computed by the algorithm will be used as an input to various post processors which extract operational parameters such as target location (coordinates), target signature, target type, etc. An overall block diagram of the processing in an MTS system is depicted in Figure 8.

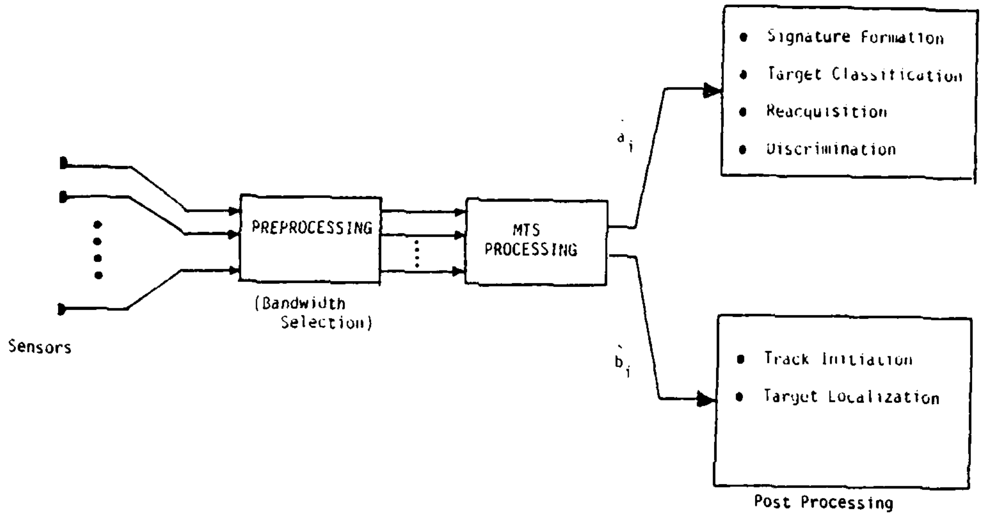


Figure 8. Block Diagram of a Basic MTS System

3. THE MTS ALGORITHM

The core of the MTS processor is a recursive parameter estimation algorithm which estimates ARMA coefficients from an observed data sequence. Algorithms of this type have been developed in the context of adaptive control [1]. Our application of this class of algorithms to acoustic signal processing seems to be a pioneering effort which promises to lead to a whole new class of adaptive signal processing techniques. Some important modifications are required in transforming this type of algorithm from the control context to the signal processing context. A significant part of our research effort was directed to investigation and development of these modifications. A key development, which is described later in this section, was the improvement of the convergence properties of the algorithm.

The Basic Algorithm

Several versions of recursive parameter estimation algorithms have been coded and tested:

(1) Recursive Least Squares (RLS)

(2) Recursive Maximum Likelihood (RML1)

(3) Modified Recursive Maximum Likelihood (RMLP)

(4) Recursive Maximum Likelihood with Prefiltering (RML2)

Initial experiments indicated that the RML2 algorithm is most suitable for our application. We will therefore describe here only the RML2 algorithm. For a more detailed description of all of these algorithms see [2], [3].

The RML2 algorithm estimates the parameters of an ARMA model of the following type:

$$y(t) = -\sum_{i=1}^{n_a} a_i\, y(t) + \sum_{i=1}^{n_b} b_i\, u(t-i) + \sum_{i=0}^{n_c} c_i\, e(t-i) \qquad (12)$$

where $e(t)$ is an (unobservable) white noise process. The presence of the

PRECEDING PAGE BLANK-NOT FILMED

c_i coefficients enables us to handle correlated measurement noise and the case of unknown inputs. It is assumed that $c_o=1$. Equation (12) can be written more compactly as

$$y(t) = \phi^T(t)\theta + e(t) \tag{13}$$

where

$$\phi^T(t) = [-y(t-1),\ldots,-y(t-n_a);\ u(t-1),\ldots,u(t-n_b);\ e(t-1),\ldots,e(t-n_c)]$$

$$\theta^T = [a_1,\ \ldots,\ a_{n_a};\ b_1,\ \ldots,\ b_{n_a};\ c_1,\ \ldots,\ c_{n_c}]$$

The dimension of θ and ϕ is

$$n = n_a + n_b + n_c.$$

Since Eq. (13) is linear in the unknown parameters (the components of θ), a recursive estimation algorithm is obtained by the following set of Kalman filter equations:

$$\hat{\theta}(t+1) = \hat{\theta}(t) + K(t+1)\ \varepsilon(t+1) \tag{14a}$$

$$K(t+1) = P(t)\ \phi(t+1)/(\lambda + \phi^T(t+1)P(t)\phi(t+1)) = P(t+1)\ \phi(t+1) \tag{14b}$$

$$P(t+1) = [P(t)-P(t)\phi(t+1)\phi^T(t+1)P(t)/(\lambda+\phi^T(t+1)P(t)\phi(t+1))]/\lambda = \tag{14c}$$

$$= \text{error covariance of the parameters.}$$

$$\varepsilon(t+1) = y(t+1) - \phi(t+1)^T\ \hat{\theta}(t) = \text{prediction error,} \tag{14d}$$

with initial conditions,

$$P(o) = \alpha I,\ \alpha = \text{a scalar parameter}$$

$$\hat{\theta}(o) = 0 \quad \text{or} \quad \theta_o, \text{ a prior estimate.}$$

The parameter λ represents data windowing, i.e., it is the "forgetting factor" of the algorithm. Various (time-varying as well as fixed) values of this parameter have been tried out. To facilitate the convergence of the algorithm on short data sequences the following was found to give the best results:

$$\lambda(t+1) = \lambda_o \lambda(t) + (1-\lambda_o) \tag{15}$$

where $\lambda(o)$, λ_o are specified parameters.

Other choices for λ are described in [2]. Two additional quantities which are useful to keep track of the numerical behavior of the algorithm are:

$$\text{trace } \{P(t)\} = \sum_{i=1}^{n} P_{ii}(t) \tag{16}$$

and

$$\eta(t) = \frac{1}{n^2} \text{ trace } \{P(t)\} \text{ trace } \{P(t)^{-1}\} = \tag{17}$$

$$= \text{a measure of how close to singular is } P(t).$$

The only difficulty with the algorithm described above is that it requires knowledge of the unobservable noise sequence $e(t)$ (which is required for $\phi(t)$). Since $e(t)$ is unknown, it needs to be replaced by some estimate of $e(t)$. Different versions of the Recursive Maximum Likehihood algorithm are obtained by different choices of the estimate of $e(t)$. For example:

RML1

$$\hat{e}(t) = \varepsilon(t) = y(t) - \phi(t)^T \hat{\theta}(t-1) \tag{18}$$

RML2

$$\hat{e}(t) = y(t) - \phi(t)^T \hat{\theta}(t) \tag{19}$$

In RML2, the unknown $e(t)$ is replaced by a filtered version of the prediction error $\varepsilon(t)$. This filtering is crucial to the proper convergence of the algorithm in MTS applications.

The filtering is accomplished by replacing the $\phi(t)$ vector which is used in Equations (14c), (14d) by a version of $\phi(t)$ filtered by $1/D(z)$ where

$$D(z) = 1 + d_1 z^{-1} + \ldots + d_{n_d} z^{-n_d} \tag{20}$$

Summary of the Filtering Equations

Let $n_{max} = \max \{n_a, n_b, n_c\}$

Define the $n_{max} \times n_{max}$ matrix D

$$D = \begin{bmatrix} -d_1 & -d_2 & \cdots & -d_{n_d} & 0 \cdots 0 & 0 \\ 1 & 0 & & & & 0 \\ & 1 & \ddots & & & \vdots \\ & & & \ddots & & \\ 0 & & & & 1 & 0 \end{bmatrix} \tag{21}$$

Define the $n_{max} x1$ vectors x_1, x_2, x_3 by the following recursions

$$x_1(t+1) = Dx_1(t) + \begin{bmatrix} y(t) \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \quad , \quad x_1(o) = 0 \tag{22a}$$

$$x_2(t+1) = Dx_2(t) - \begin{bmatrix} u(t) \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \quad , \quad x_2(o) = 0 \tag{22b}$$

$$x_3(t+1) = Dx_3(t) - \begin{bmatrix} \varepsilon(t) \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \quad , \quad x_3(o) = 0 \tag{22c}$$

Let

$\bar{x}_1(t)$ be $n_a x1$ consisting of the first n_a entries of $x_1(t)$

$\bar{x}_2(t)$ be $n_b x1$ consisting of the first n_b entries of $x_2(t)$

$\bar{x}_3(t)$ be $n_c x1$ consisting of the first n_c entries of $x_3(t)$

then

$$\phi^T(t) = [\bar{x}_1^T(t), \bar{x}_2^T(t), \bar{x}_3^T(t)] \tag{23}$$

The significance of this filtering has to do with the convergence properties of recursive parameter estimation algorithms. Convergence analysis has shown ([4]-[6]) that without prefiltering, the criterion for convergence is that

$$H(z) = \frac{1}{C(z)} - \frac{1}{2} \tag{24a}$$

be strictly positive real, i.e.,

$$Re\{H(e^{jw})\} > 0 \text{ for all } w. \tag{24b}$$

Unfortunately, as will be discussed later, this condition is not fulfilled for general MTS signals. Since $C(z)$ is a property of the signal, and is not under our control, it is not possible to guarantee convergence in this case. With prefiltering, the condition for convergence becomes

$$H(z) = \frac{D(z)}{C(z)} - \frac{1}{2} \text{ strictly positive real.} \tag{25}$$

The choice of the filter $D(z)$ is under our control and, therefore, there is hope of guaranteeing convergence. A typical choice for $D(z)$ [3] is

$$D(z) = \hat{C}(z) \tag{26}$$

The reasoning behind this choice is that if $\hat{C}(z)$ is a good estimate of $C(z)$, we will get

$$H(z) = \frac{\hat{C}(z)}{C(z)} - \frac{1}{2} \cong 1 - \frac{1}{2} > 0. \tag{27}$$

In our preliminary tests, we discovered that for signals generated by sine waves in white additive noise, this type of filter was inadequate and convergence could not be achieved. This problem was the major stumbling block in our initial research effort and led to a more careful investigation into the convergence of RML2 for MTS signals. A solution to the problem has been found and sucessfully tested. The technique we developed is a significant contribution to the study and application of recursive parameter estimation. The main ideas of our technique are described next.

Improved Pre-Filtering for RML2

To understand the difficulties inherent in the pre-filtering problem, we must first see what the $C(z)$ polynomial means in terms of the target spectrum $A(z)$, the delay structure $B(z)$ and the signal-to-noise ratio. The observed signal $y(t)$ is given by

$$y(t) = \underbrace{\frac{B(z)}{A(z)} u(t)}_{\text{signal}} + \underbrace{v(t)}_{\text{measurement noise}} \tag{28}$$

where

$$A(z) = 1 + a_1 z^{-1} + \ldots + a_{n_a} z^{-n_a}$$

$$B(z) = b_1 z^{-1} + \ldots + b_{n_b} z^{-n_b}$$

$u(t)$, $v(t)$ = independent white noise processes with variance σ_u^2 and σ_n^2 respectively.

Multiplying through by $A(z)$ we get

$$A(z)y(t) = B(z)u(t) + A(z)v(t) \tag{29}$$

Since neither $u(t)$ nor $v(t)$ is directly measureable, there is no way of distinguishing between them and they can be replaced by a white process $e(t)$ with variance σ_e^2 such that

$$A(z)y(t) = C(z)\, e(t), \tag{30}$$

where

$$\sigma_e^2 C(z)\, C(z^{-1}) = \sigma_u^2 B(z)\, B(z^{-1}) + \sigma_n^2 A(z)\, A(z^{-1}). \tag{31}$$

In other words, $C(z)\, e(t)$ will have the same spectrum as $B(z)\, u(t) + A(z)\, v(t)$. To gain some insight into what $C(z)$ may look like, we consider two simple examples. Both examples assume $B(z) = b_D z^{-D}$, i.e., a pure delay propagation model.

(i) SNR = ∞

In this case, $\sigma_n^2 = 0$ and therefore

$$\sigma_e^2 C(z)\ C(z^{-1}) = \sigma_u^2\ b_D^2$$

or

$$C(z) = \sigma_u b_D / \sigma_e = \text{constant} \tag{32}$$

(ii) SNR = 0 (or $-\infty$ db)

In this case, the second term on the right-hand side of (31) dominates, and therefore,

$$\sigma_e^2\ C(z)\ C(z^{-1}) \cong \sigma_n^2\ A(z)\ A(z^{-1})$$

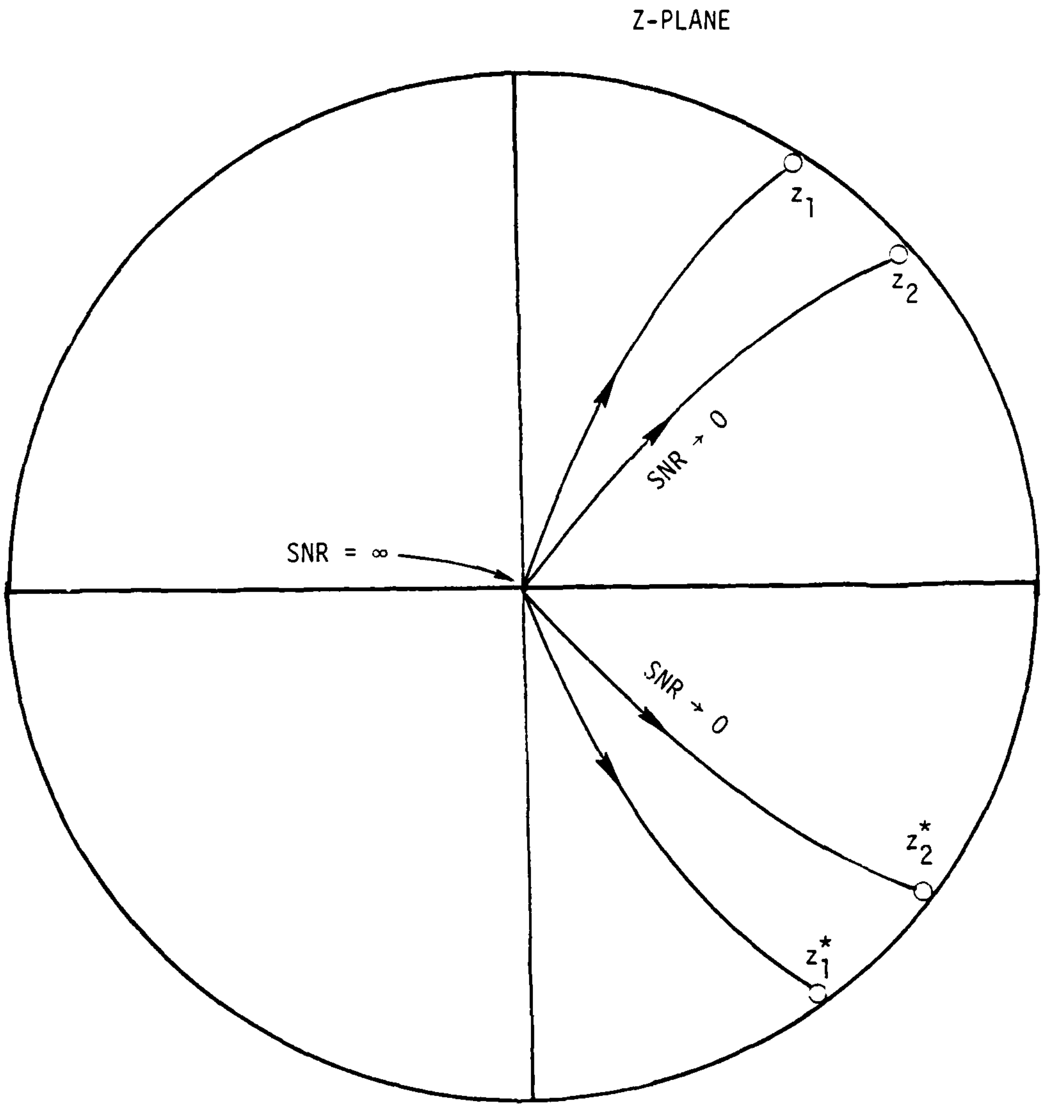
or

$$C(z) \cong A(z). \tag{33}$$

In general, as the SNR decreases, the zeroes of $C(z)$ will move from the origin, towards the zeroes of $A(z)$, as indicated in Figure 9. The exact trajectory of this motion can be plotted using classical root locus techniques [7]. Note that the zeroes of $A(z)$ are shown in Figure 9 to be on or very close to the unit circle. This is to be expected for narrow-band line spectra and for pure sine waves.

Several conclusions can be drawn from the discussion above: (i) For high SNR, no pre-filtering is needed since $C(z)$ = a positive constant; (ii) For very low SNR $C(z)$ has zeroes near the unit circle which means that $1/C(z)$ will most likely not be positive real! Thus, pre-filtering is needed. We many choose either $D(z) = \hat{C}(z)$ or $D(z) = \hat{A}(z)$, since $C(z) \cong A(z)$. The choice $D(z) = \hat{A}(z)$ is usually preferred since the estimates of the AR coefficients $\{a_i\}$ converge much faster than MA coefficients $\{\hat{C}_i\}$.

Preliminary tests of the algorithms essentially confirmed these conclusions. However, serious difficulties were experienced in the case of



z_1, z_2 = zeroes of A(z)

Figure 9: Root Locus of the Zeroes of C(z)

narrowband signals in which case $A(z)$ has zeroes very close to the unit circle. Filtering by $\hat{A}(z)$ led the algorithm to diverge even for reasonably good SNR's. Further investigation indicated at least two possible causes for this phenomenon:

(i) The filter $\hat{A}(z)$ is often unstable, i.e., $\hat{A}(z)$ has poles outside the unit circle. The reason is that since $A(z)$ has poles very near to the unit circle, relatively small estimation errors are sufficient to make $\hat{A}(z)$ unstable, and cause the algorithm to "blow up".

(ii) The assumption that $C(z) \cong A(z)$ and therefore that $(\hat{A}(z)/C(z) - 1/2)$ is positive real is only true for very low SNR's. At moderate SNR's, $C(z)$ may be quite different from $A(z)$ as (31) and Figure 9 clearly indicate. Thus, it would be preferable to find a filter $D(z)$ that is closer to $C(z)$.

A solution which addresses both of these issues is the *following:* let

$$D(z) = \hat{A}(kz) \tag{34}$$

where k is some constant smaller than one. The zeroes of $A(kz)$ are obtained from the zeroes of $A(z)$ by shifting along radial lines, as indicated in Figure 10. The new filter is implemented by setting

$$d_i = k^i \hat{a}_i \tag{35}$$

since

$$\hat{A}(kz) = 1 + \hat{a}_1 kz^{-1} + \hat{a}_2 k^2 z^{-2} + \ldots + \hat{a}_{n_a} k^{n_a} z^{-n_a} \tag{36}$$

The modified filter $\hat{A}(kz)$ is more stable than $\hat{A}(z)$, since its roots are further away from the unit circle. Furthermore, by a proper choice of k, these roots can be brought closer to the roots of $C(z)$, as indicated by a comparison of Figures 9 and 10.

The introduction of this modified pre-filter greatly improved the convergence properties of the algorithms for moderate and low SNR.

Finally we should note that typically the RML2 algorithm is used with $n_b = 0$ and $n_a = n_c$ = twice the number of sine waves expected. Setting $n_b = 0$ is necessary, since the inputs $u(t)$ are not observable by the algorithm. The algorithm is also used in another mode with $n_a = n_c = 0$ when performing TDOA estimation for pure sine waves in noise, as will be discussed later.

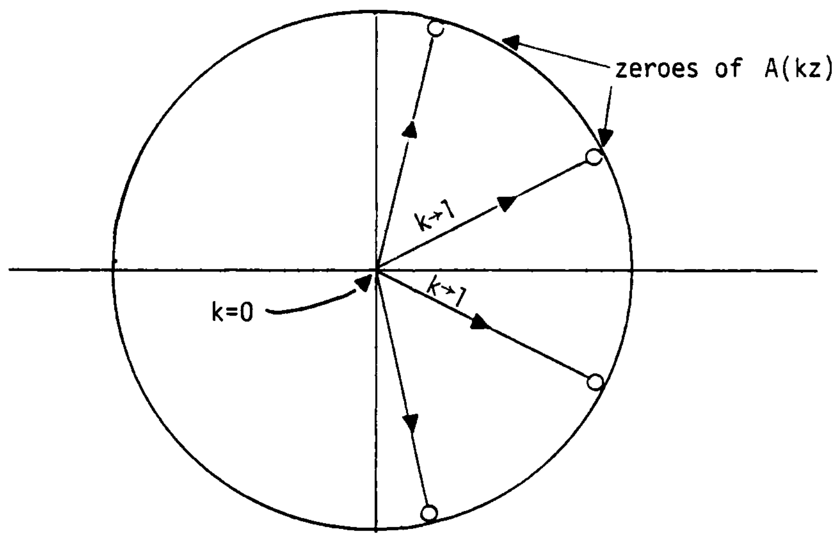


Figure 10. The Root Locus for A(kz)

4. PERFORMANCE EVALUATION

The MTS algorithm was coded and tested to evaluate its performance for different types of signals and different signal-to-noise ratios. The tests so far have been restricted to a single fixed target. Two aspects of the algorithm were studied in these tests: estimation of target spectrum and TDOA estimation. In this section, we present some preliminary results which indicate the type of performance achievable by the MTS algorithm. It should be emphasized, however, that these results are not conclusive; more testing would be needed to establish performance bounds.

4.1 Spectral Estimation

The signals used in our spectral estimation experiments were sine waves in noise, i.e.,

$$y(t) = \sum_{i=1}^{m} A_i \sin(2\pi t/N_i) + v(t) \tag{37}$$

where

A_i = amplitude

N_i = period

$v(t)$ = white gaussian noise

The RML2 algorithm was used to identify the $\{a_i\}$ parameters of the received signal $y(t)$. The final estimates $\hat{a}_i$ of the parameters are then used to generate a spectral estimate. In our simulation, this was done by generating the impulse response of the AR model $1/\hat{A}(z)$, where $\hat{A}(z) = 1 + \hat{a}_1 z^{-1} + \ldots + \hat{a}_{n_a} z^{-n_a}$, and computing its power spectrum. Some typical results for two test cases are shown in Figures 11-20.

Test Case #1

The signal was a single sine wave with a period $N_1 = 5.12$. Assuming that the MTS algorithm operates on a 1 Hz bandwidth (B = 1 Hz),

PRECEDING PAGE BLANK-NOT FILMED

this corresponds to a frequency of 0.390 Hz. It should be remembered that this frequency is really a deviation from the nominal frequency used in the bandwidth selection process depicted in Figures 2 and 3. Thus, we actually are looking at an expanded picture of the spectrum in the range of f_o to f_o+B Hz.

The algorithm used was RML2 with $n_a = n_c = 2$, $n_b = 0$, $\lambda(o) = .95$, $\lambda_o = .99$. In all cases N=512 data points were used. This corresponds to an integration time of $T \cong 256$ sec = 4 minutes (again assuming B=1).

The results are summarized in Table 1 and Figures 11-15. It should be pointed out that for the low SNR cases (-5 dB, -10dB) the algorithm really requires a longer integration time. However, already at N=512 points, or 4 minutes of integration, the true spectrum starts to emerge. For comparison purposes, we have included in the figures a plot of a conventional FFT, using the same number of data points. Hanning windowing was used where indicated.

Test Case #2

The signal consisted of two sine waves with periods N_1=5.12, N_2=3.00, which correspond to 0.390 Hz and 0.667 Hz. The same algorithm was used as in Test Case #1. The results are summarized in Table 1 and Figures 16-20.

Table 1: Test Case Parameter Estimates

Fig. No.			Test Case 1		Test Case 2				Fig. No.
			a_1	a_2	a_1	a_2	a_3	a_4	
	True Value		-.6738	1.0000	.326	1.326	.326	1.00000	
11	SNR (dB)	20	-.6742	1.0000	.3260	1.3257	.3264	1.00000	16
12		10	-.6748	1.0000	.32449	1.32396	.32562	1.00029	17
13		0	-.6746	.9969	.31794	1.29706	.30661	.97536	18
14		- 5	-.6803	.92622	.34322	1.20969	.28624	.95076	19
15		-10	-.5859	.6909	.5297	.7561	.3836	.8787	20

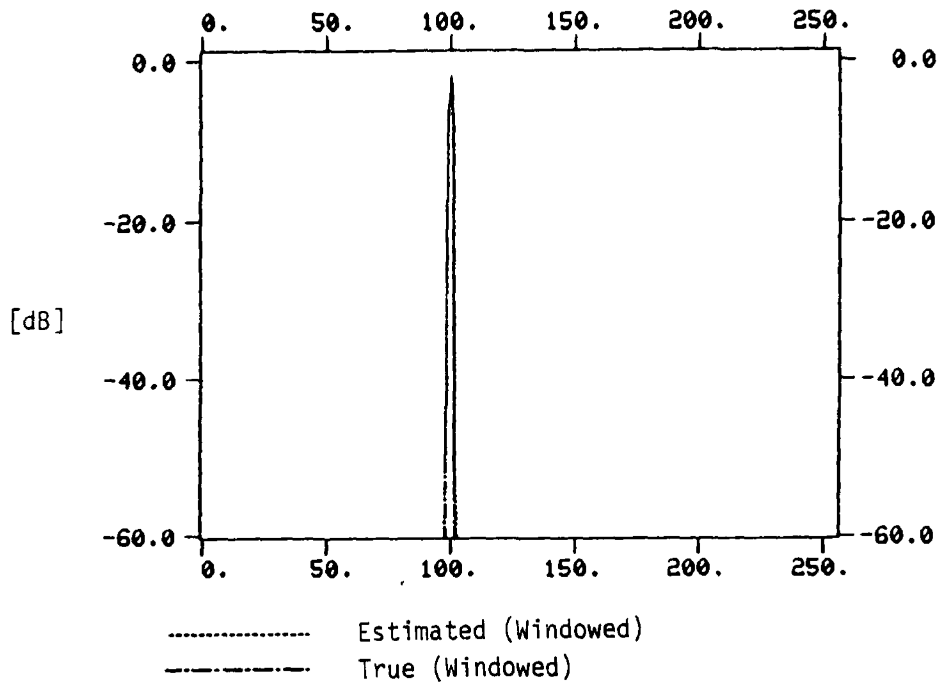


Figure 11: Estimated vs. True Spectrum -- SNR=20 dB
P(o)=1, KA=.9, INIT=Zero

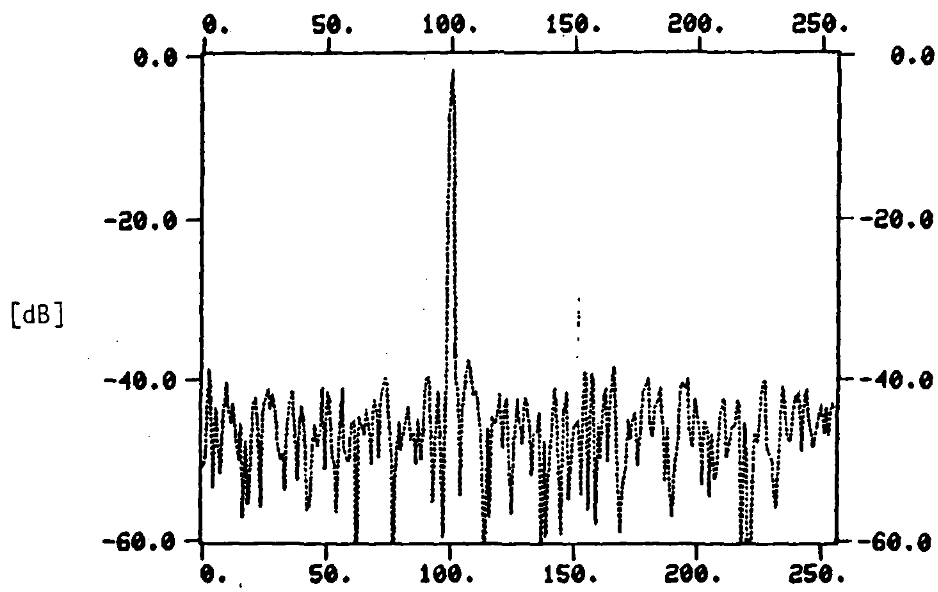
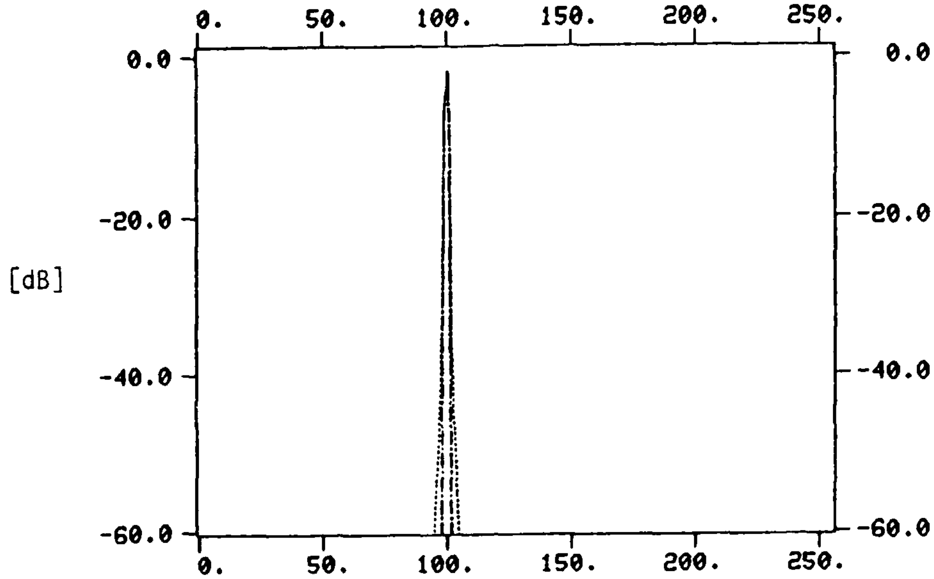


Figure 11B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=20dB, N=512



------------------- Estimated (Windowed)

—·—·—·— True (Windowed)

Figure 12A: Estimated vs. True Spectrum -- SNR=10dB
P(o)=1, KA=.9, INIT=Zero

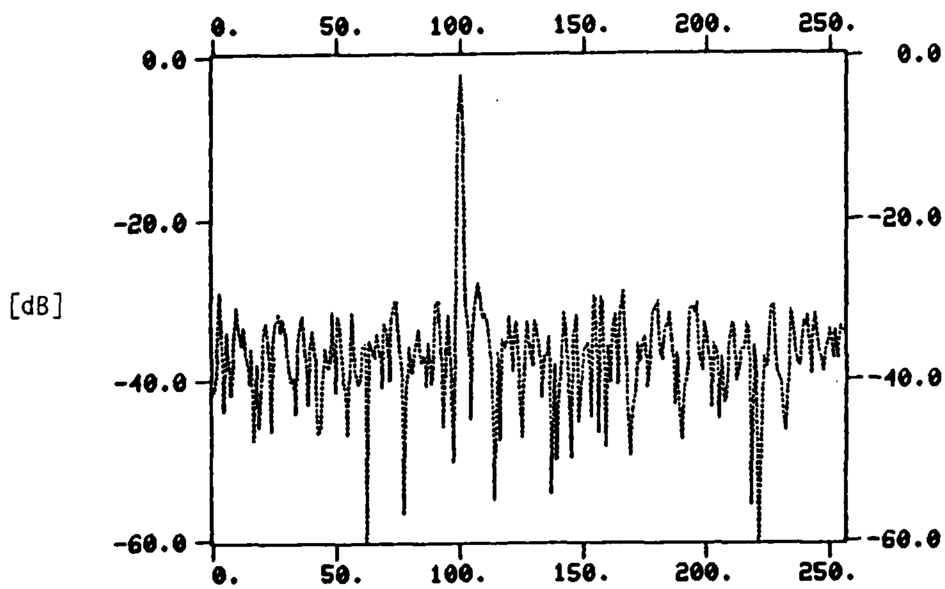


Figure 12B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=10dB, N=512

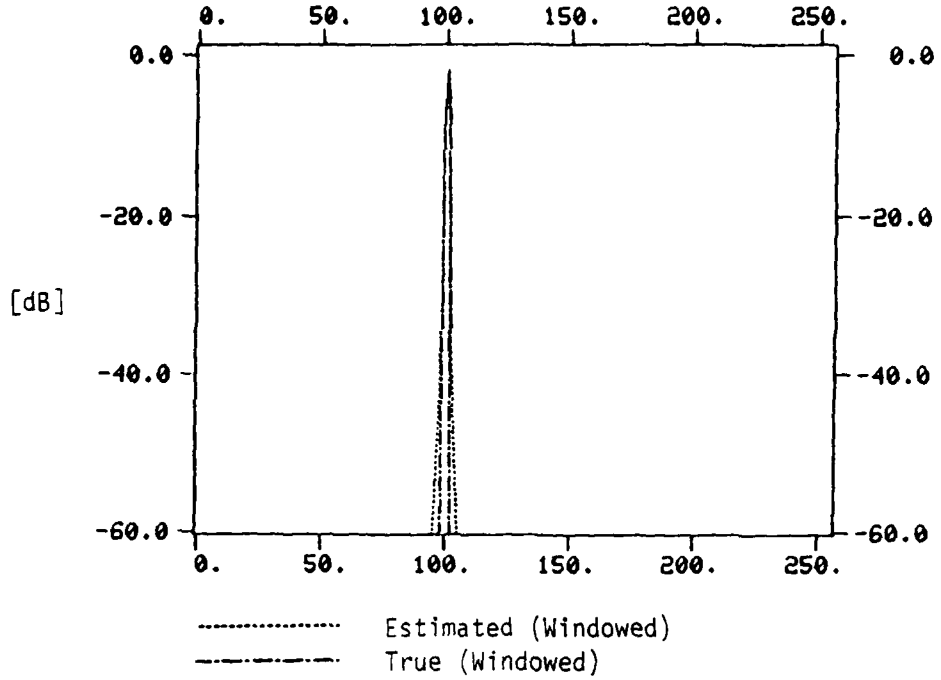


Figure 13A: Estimated vs. True Spectrum -- SNR=0dB
P(o)=1, KA=.9, INIT=Zero

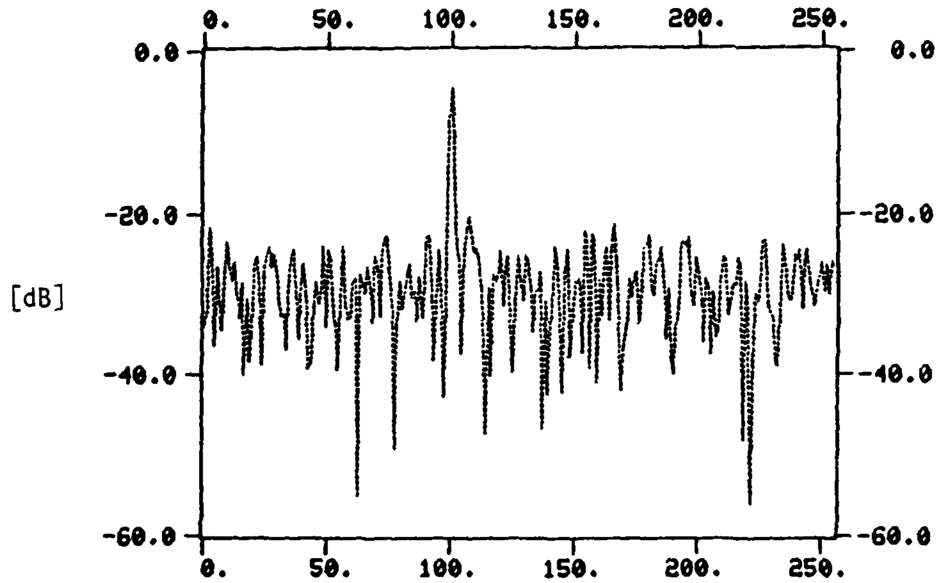


Figure 13B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=0dB, N=512

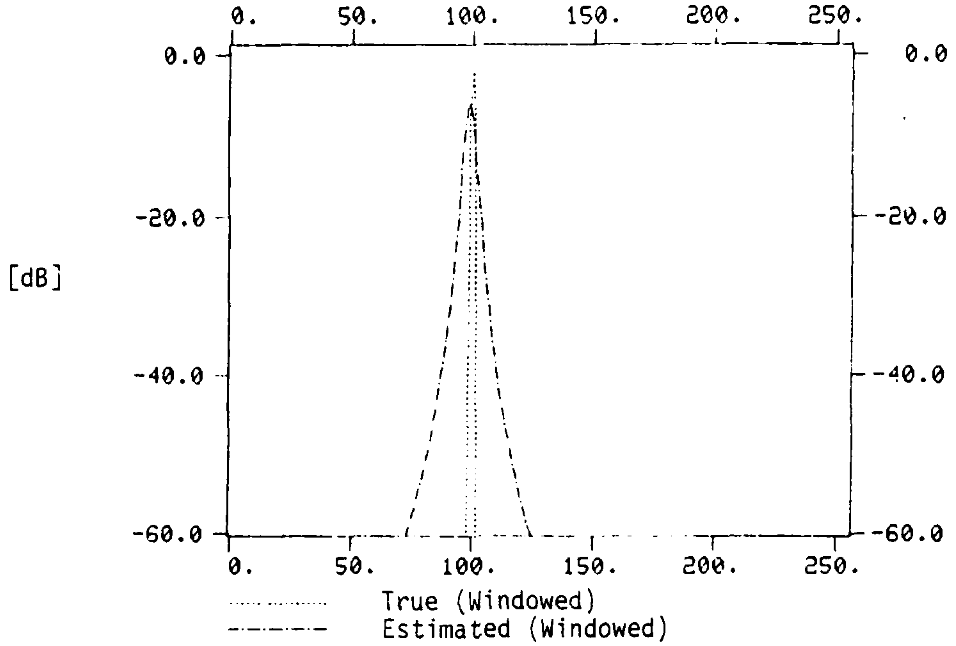


Figure 14A: Estimated and True Spectra -- SNR=-5dB
P(o)=1, KA=.7, INIT=SPEC

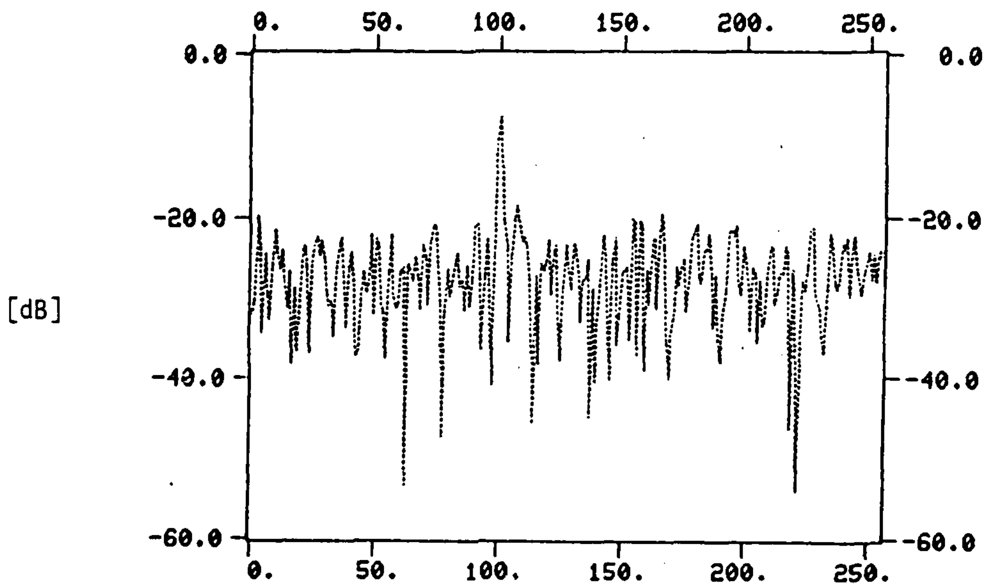
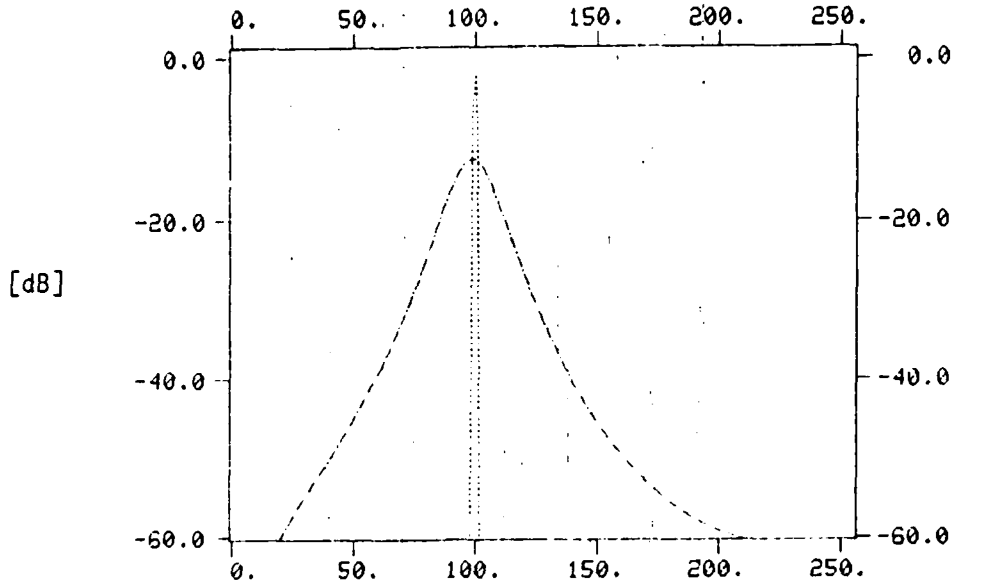


Figure 14B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=-5dB, N=512



True (Windowed)
Estimated (Windowed)

Figure 15A: Estimated and True Spectrum -- SNR=-10dB
P(o)=1, KA=.1, INIT=SPEC

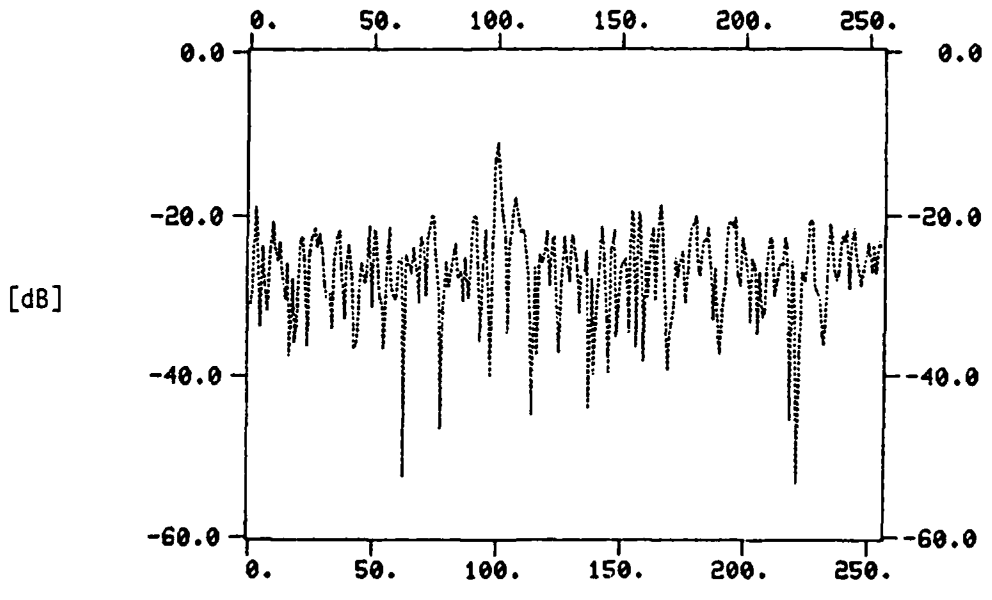


Figure 15B: Spectrum Obtained by FFT of Received
Signal (Windowed) SNR=-10dB, N=512

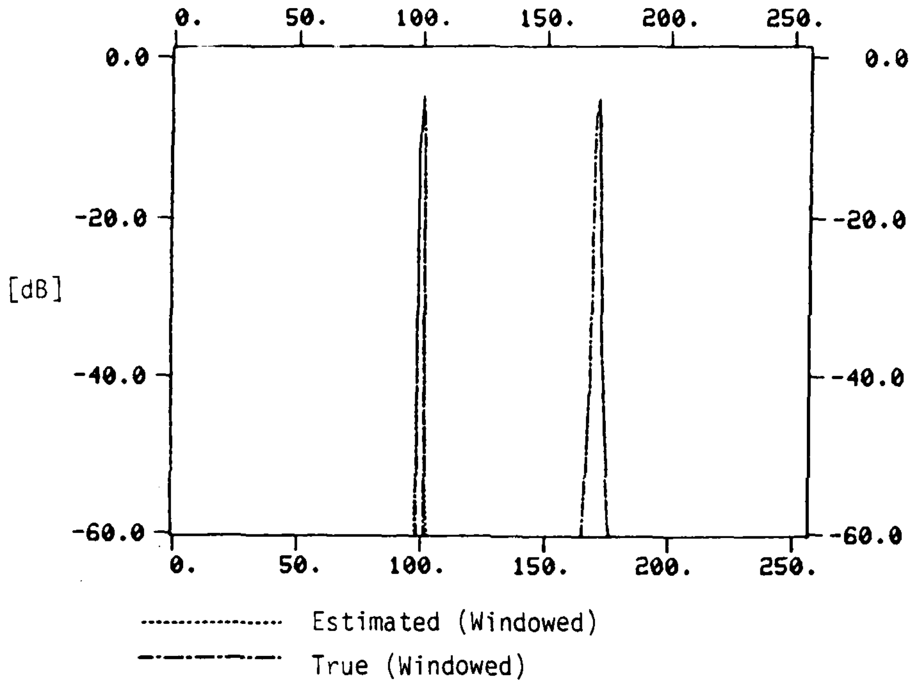


Figure 16A: Estimated vs. True Spectrum -- SNR=20dB
P(o)=.1, KA=.5, INIT=Zero

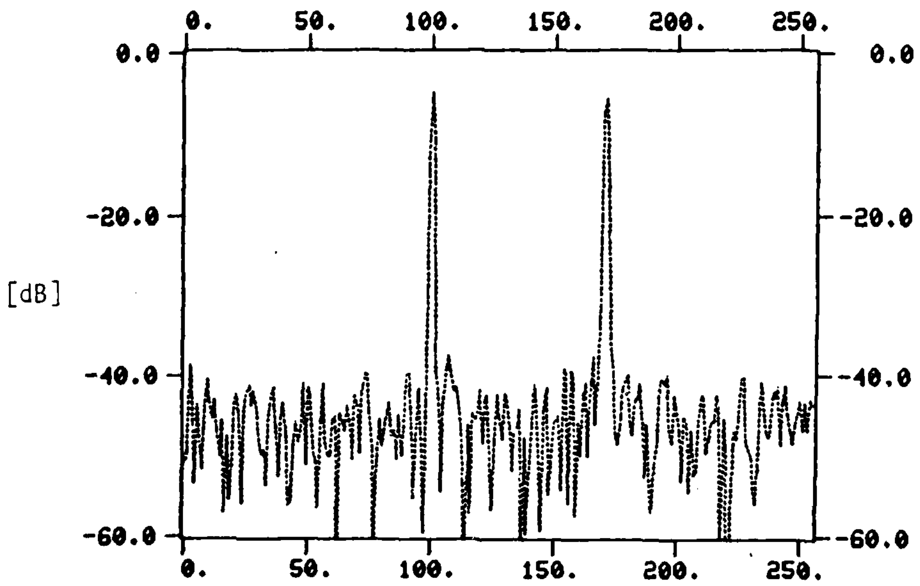
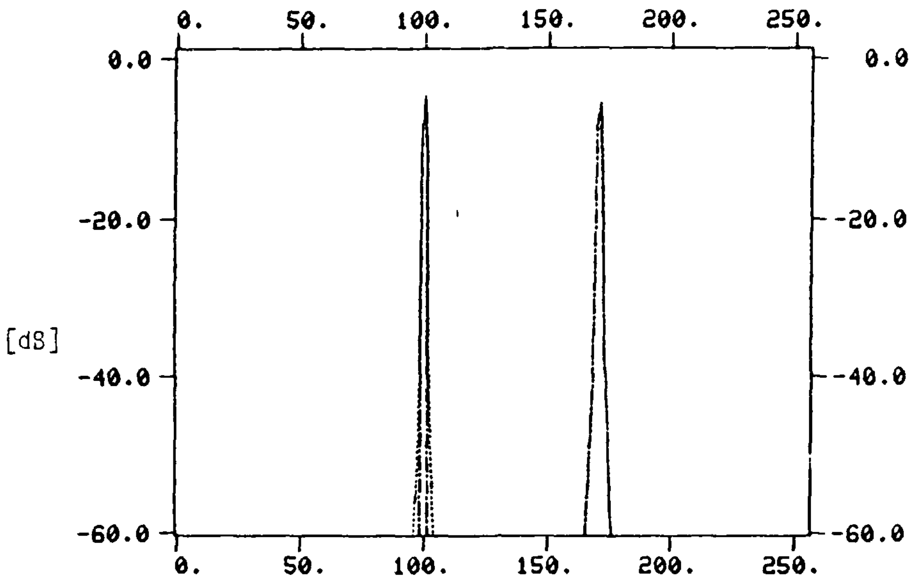


Figure 16B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=20dB, N=512



Estimated (Windowed)
True (Windowed)

Figure 17A: Estimated vs. True Spectrum -- SNR=10dB
P(o)=.1, KA=.5, INIT=Zero

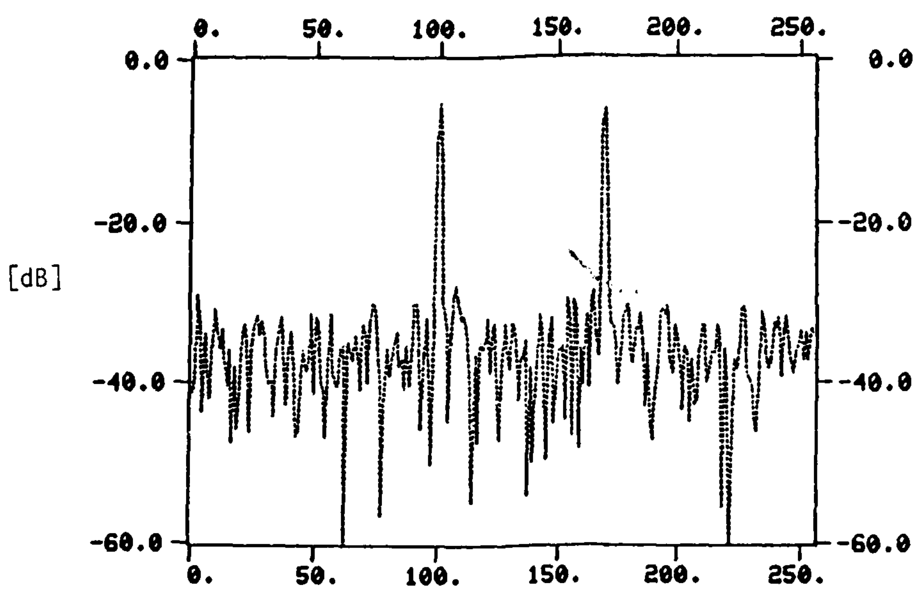


Figure 17B: Spectrum Obtained by FFT of Received Signal (Windowed)
SNR=10dB, N=512

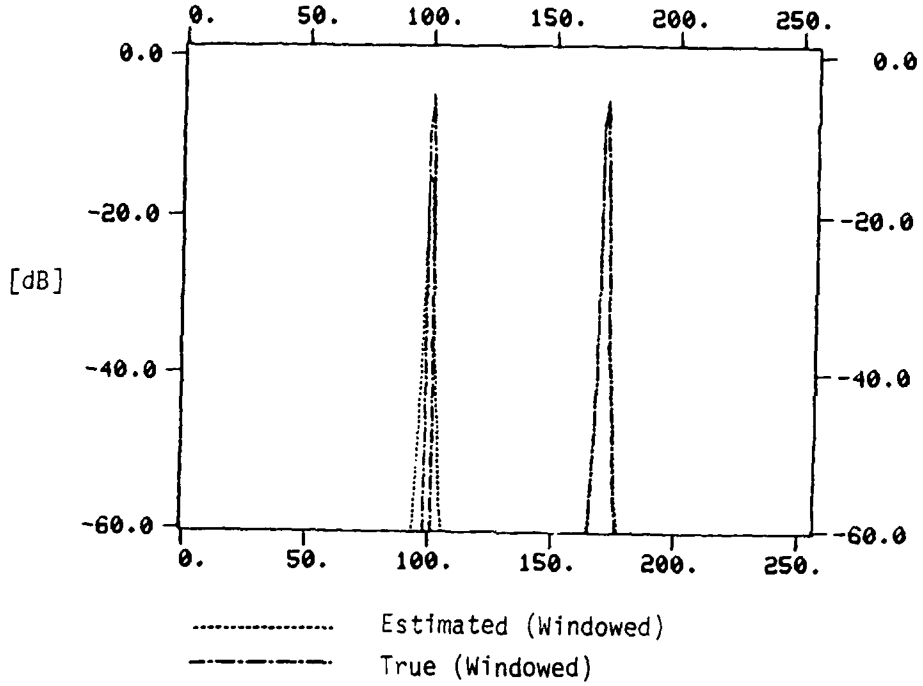


Figure 18A: Estimated vs. True Spectrum -- SNR=0dB
P(o)=.1, KA=.5, INIT=Zero

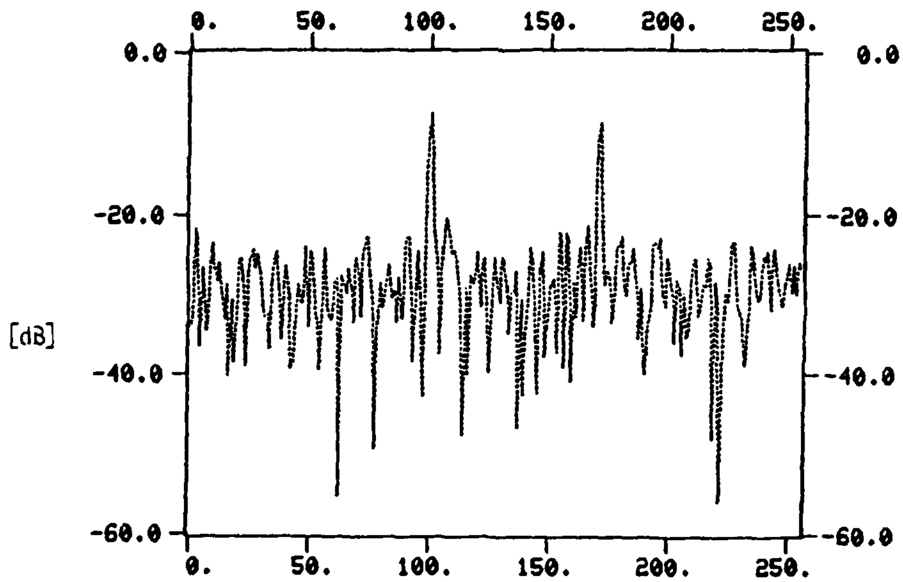


Figure 18B: Spectrum Obtained by FFT of Received Signal (Windowed) SNR=0dB, N=512

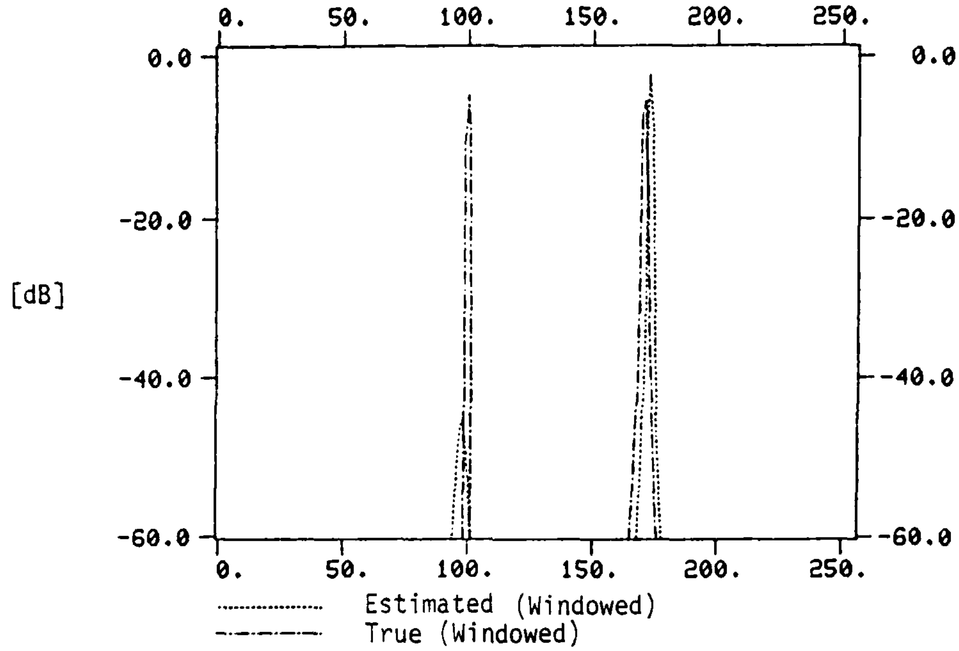


Figure 19A: Estimated vs. True Spectrum -- SNR=-5dB P(o)=.1, KA=.5, INIT=SPEC

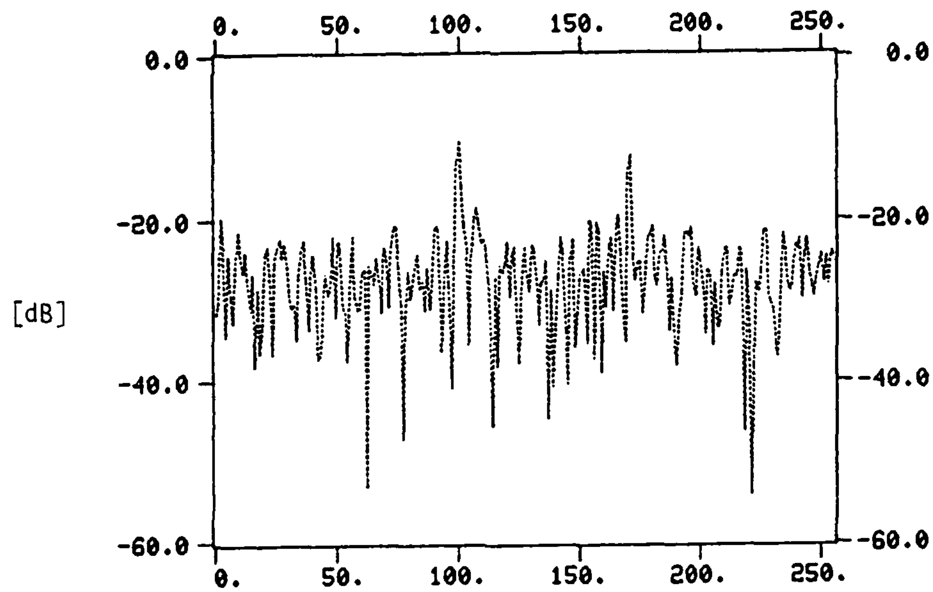


Figure 19B: Spectrum Obtained by FFT of Received Signal (Windowed) SNR=-5dB, N=512

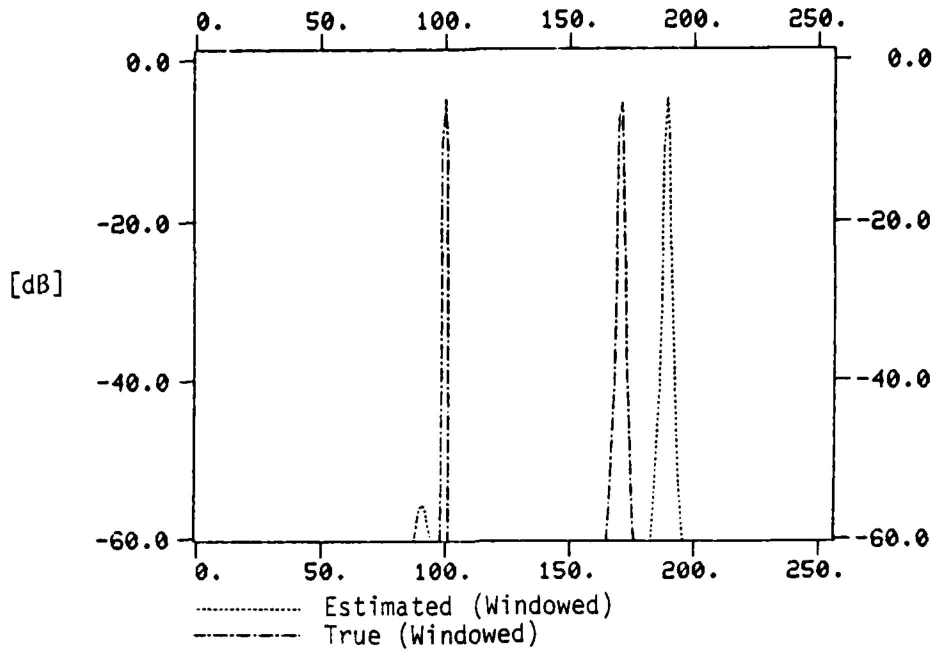


Figure 20A: Estimated vs. True Spectrum -- SNR=-10dB
P(o)=.1, KA=.5, INIT=SPEC

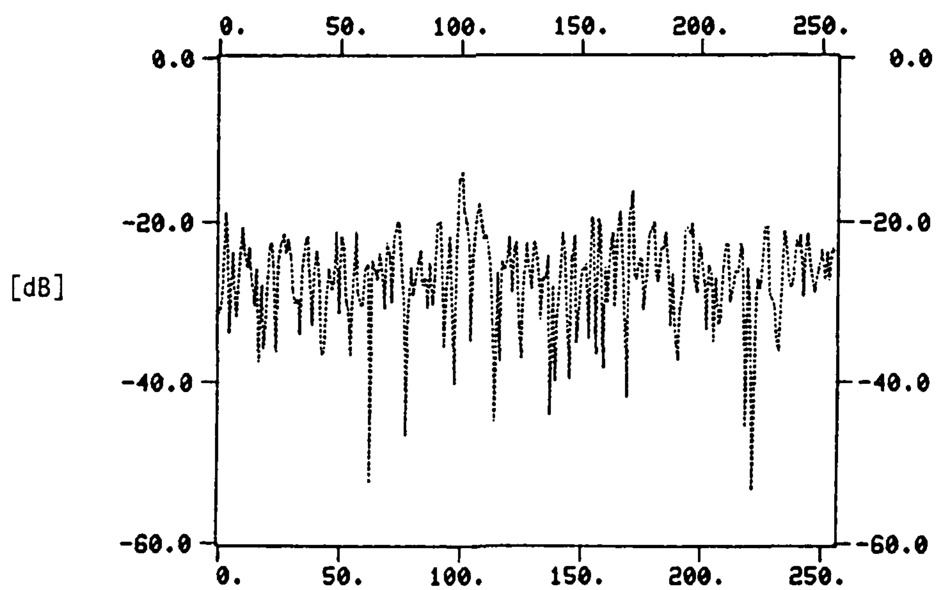


Figure 20B: Spectrum Obtained by FFT of Received Signal
(Windowed) SNR=-10dB, N=512

4.2 TDOA Estimation

Several techniques for TDOA estimation based on the MTS algorithm were implemented and tested. The first and most straightforward approach consists of estimating the coefficients $\{b_i\}$ of an MA model relating the signals $y_1(t)$, $y_2(t)$ at the output of two receivers. Let us assume that the signals in the two receivers are given by

$$y_1(t) = x(t-D_1) + n_1(t) \tag{38a}$$

$$y_2(t) = x(t-D_2) + n_2(t) \tag{38b}$$

where

$x(t)$ = target signal

D_1, D_2 = propagation delays

n_1, n_2 = independent measurement noise processes

This equation can be rewritten as

$$y_2(t) = y_1(t-\tau) + n(t) \tag{39}$$

where

$\tau = D_2 - D_1 = \text{TDOA}$

$n(t) = n_2(t) - n_1(t-\tau)$

Equation (39) represents a special case of a moving average model

$$y_2(t) = \sum_{i=1}^{n_b} b_i y_1(t-i) + n(t) \tag{40}$$

with $b_i \neq 0$ except for $b_\tau = 1$. Thus, estimating the model parameters and looking for the largest $\{b_i\}$ will indicate the value of the TDOA. These parameters can be also used to estimate noninteger values of the TDOA by a proper interpolation technique, as discussed in Appendix B. This interpolation technique was used to provide estimates in two test cases:

Case #1

A second order AR model driven by white noise, with a spectrum given by Figure 21. The algorithm used was RLS with $n_a = n_c = 0$, $n_B = 7$, $\lambda(o) = .95$, $\lambda_o = .99$. Some typical results are summarized in Table 2. The true value of the TDOA was $\tau = 3.00$.

Case #2

A fourth order AR model driven by white noise with a spectrum given by Figure 22. The same algorithm was used as in case #1. The true value of the TDOA was 3.00.

TABLE 2

TDOA Estimation

	SNR (dB)	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$	$\hat{b}_4$	$\hat{b}_5$	$\hat{b}_6$	$\hat{b}_7$	$\hat{\tau}$	P(o)	θ(o)	N
CASE #1	∞	.000	.000	1.000	.000	.000	.000	.000	3.000	10	ZERO	512
	20	-.045	.048	.898	.032	-.024	-.028	.020	2.985	10	ZERO	512
	10	-.158	-.160	.616	.120	-.121	-.074	.041	2.931	10	ZERO	512
	0	-.157	.137	.328	.104	-.153	-.127	.027	2.844	.1	SPEC	512
	- 5	-.095	.074	.177	.024	-.096	-.099	.001	2.772	10	SPEC	2068
	-10	-.045	.039	.080	-.002	-.045	-.051	-.005	2.705	10	SPEC	2048
CASE #2	∞	.000	.000	1.000	.000	.000	.000	.000	3.000	.1	SPEC	512
	20	-.189	.283	.535	.234	-.154	-.026	-.007	2.876	.1	SPEC	512
	10	-.232	.277	.379	.170	-.180	-.035	-.022	2.591	1	SPEC	512
	0	-.160	.192	.232	.110	-.170	-.166	-.026	2.515	.1	SPEC	512
	- 5	-.082	.085	.159	.040	-.089	-.144	-.047	2.715	1	SPEC	2048
	-10	-.040	.044	.076	.010	-.038	-.075	-.033	2.673	1	SPEC	2048

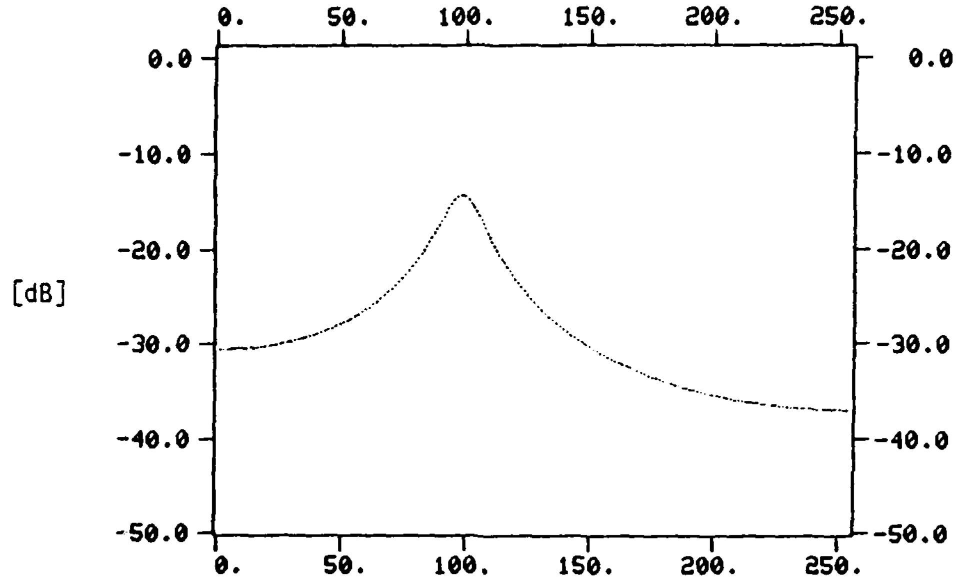


Figure 21: Target Spectrum for TDOA Test Case #1
a_1=.606, a_2=.810

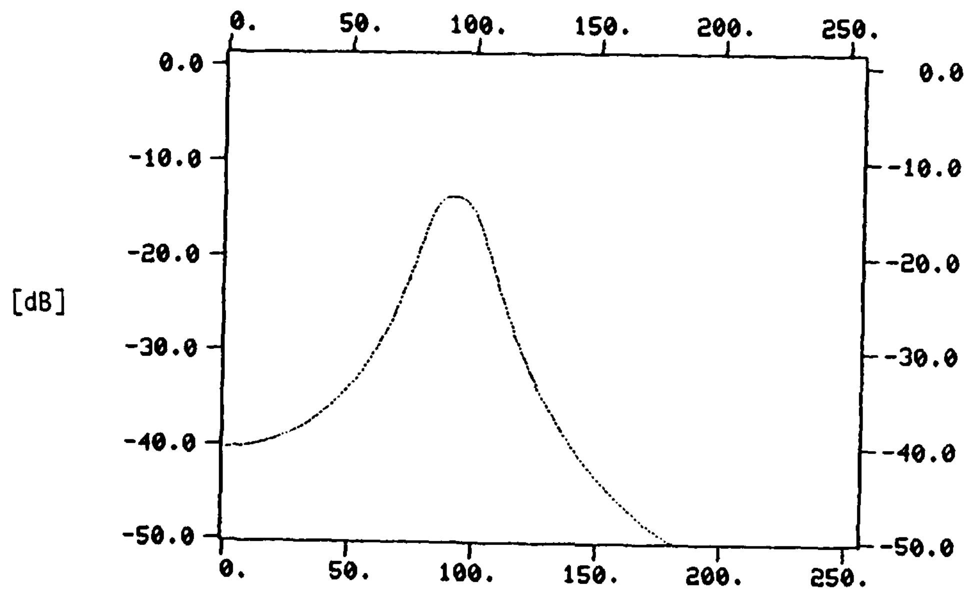


Figure 22: Target Spectrum for TDOA Test Case #2
a_1=-1.506, a_2=2.1654, a_3=-1.2199, a_4=.6561

A somewhat improved version of this approach can be obtained by using the MTS algorithm in an Adaptive Line Enhancer (ALE) mode of operation. The algorithm described in Section 3 provides a predicted estimate of the input signal (see Eq. (14d)),

$$\hat{y}(t+1) = \phi(t+1)^T \hat{\theta}(t) \quad . \tag{41}$$

The estimated signal $\hat{y}(t)$ provides a cleaner, i.e., less noisy version of the received signal $y(t)$. This is illustrated in Figures 24 and 25 which compare the power spectra of y and $\hat{y}$ for two test cases. Note the significant decrease in the noise levels in 24B and 25B.

Thus, the "enhanced" signal $\hat{y}(t)$ can be used as an input to the TDOA estimation algorithm, as depicted in Figure 23. Initial results have indicated some improvement when this method was used, however, more testing is necessary before final conclusions can be drawn

A second approach to TDOA estimation is based on "whitening" the sensor signals using the MTS algorithm and then cross-correlating to obtain the TDOA estimate. The signal whitening is achieved by using the RML2 and obtaining the residual sequence ε_t corresponding to the input signal y_t (See Figure 26.) Correlating the residuals gives a sharp well-defined peak which provides a better indication of the TDOA. Some typical examples are given in Figures 27A, 27B, which compare the correlation function of the residuals with that of a cleaner version of the data obtained by using the predicted signals $\hat{y}_1, \hat{y}_2$. This approach has significant similarities to the coherence techniques now widely employed for target detection and localization [9].

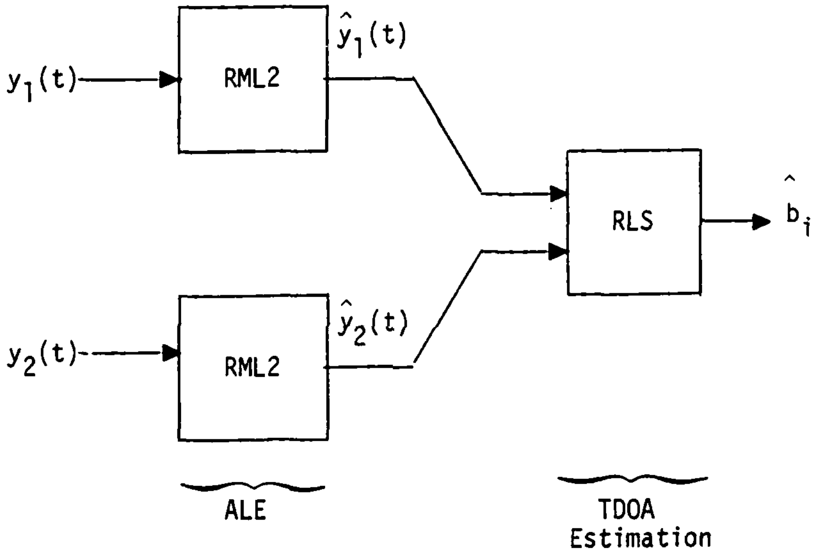


Figure 23: An Improved TDOA Estimator,
Using the Estimated Signals

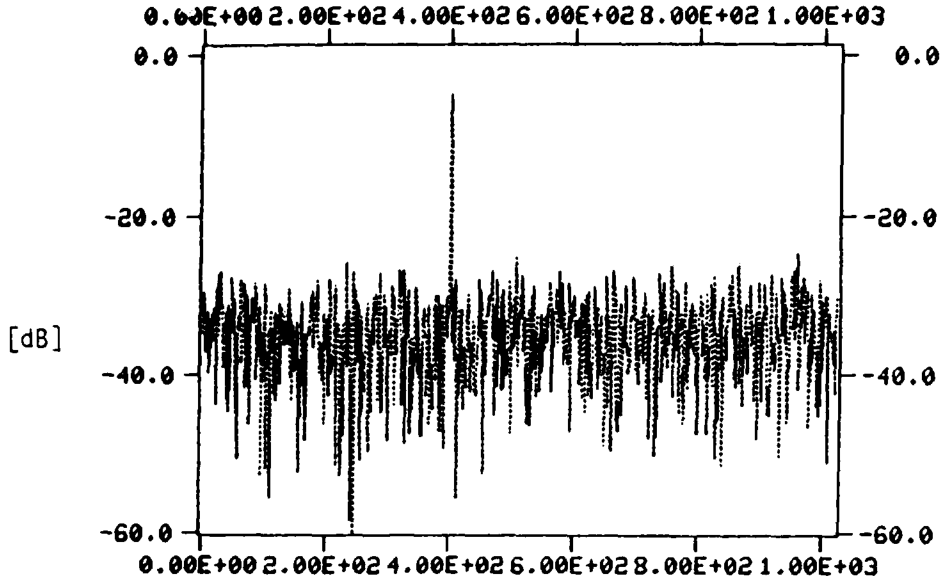


Figure 24A: Power Spectrum of A Single Sine Wave in Noise -- SNR=0dB

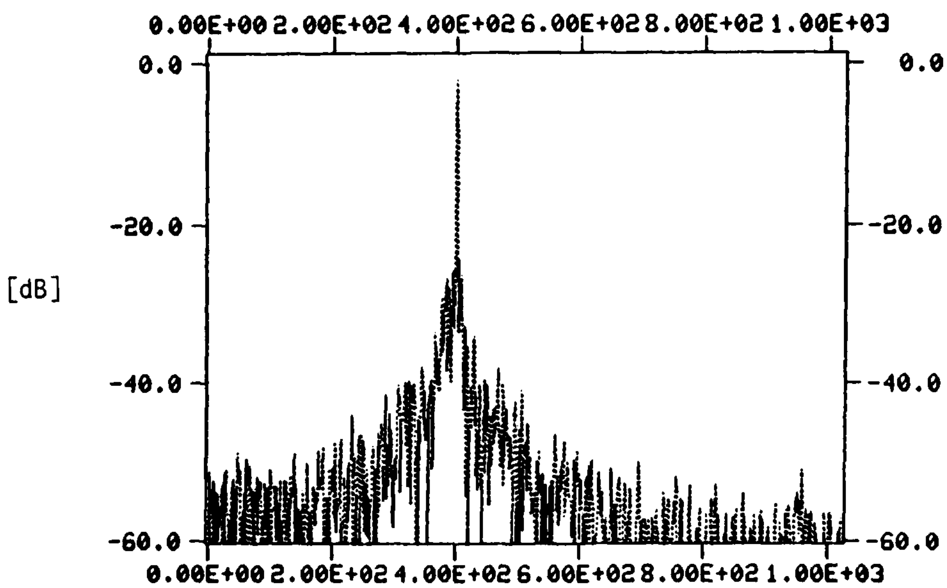


Figure 24B: Power Spectrum of Estimated Signal RML2, N=2048

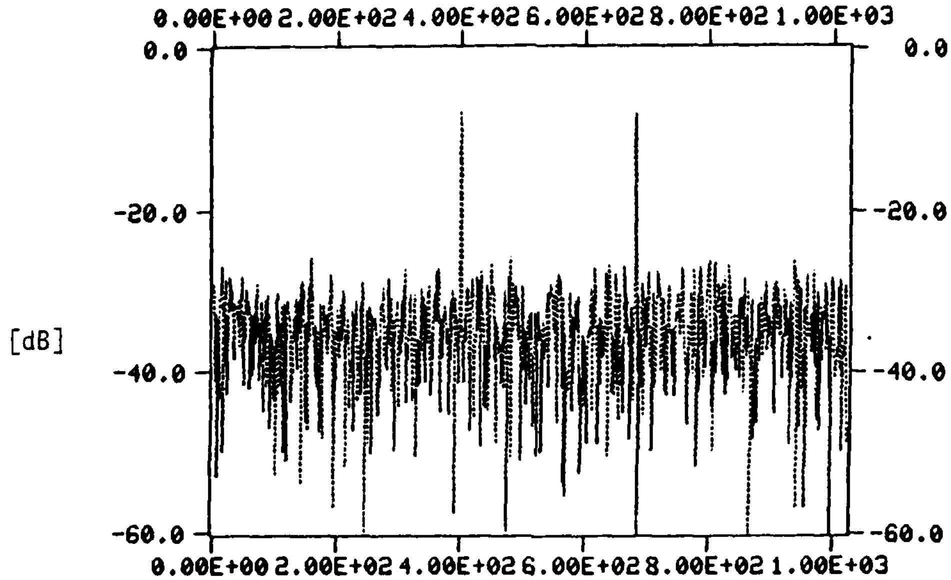


Figure 25A: Power Spectrum of Two Sine Waves in Noise -- SNR=0dB

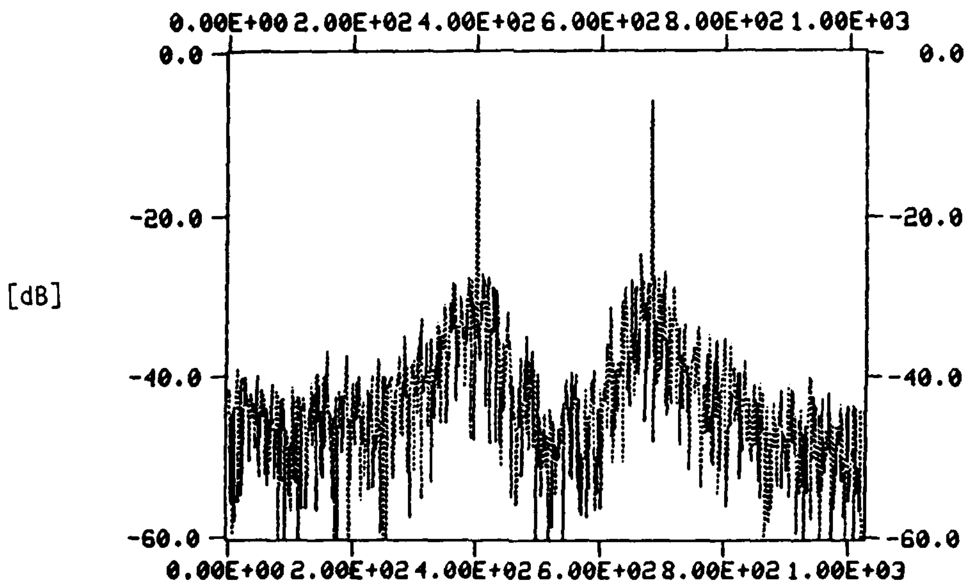


Figure 25B: Power Spectrum of Estimated Signal RML2, N=2048

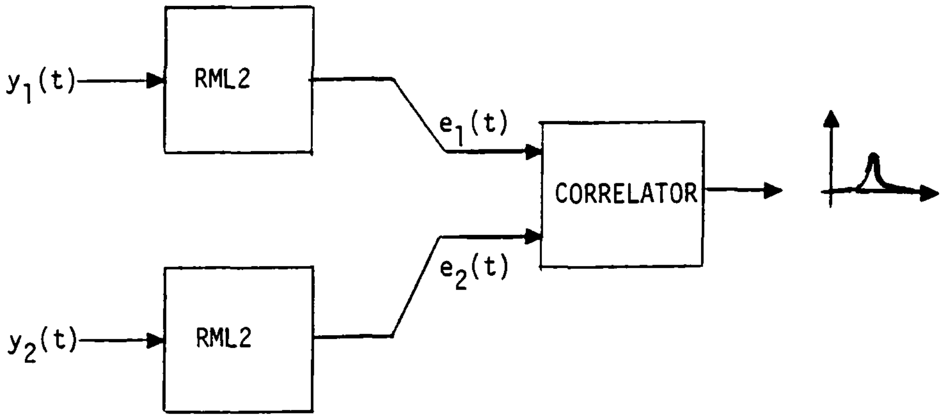


Figure 26: TDOA Estimation by Adaptive "Whitening" and Cross Correlation

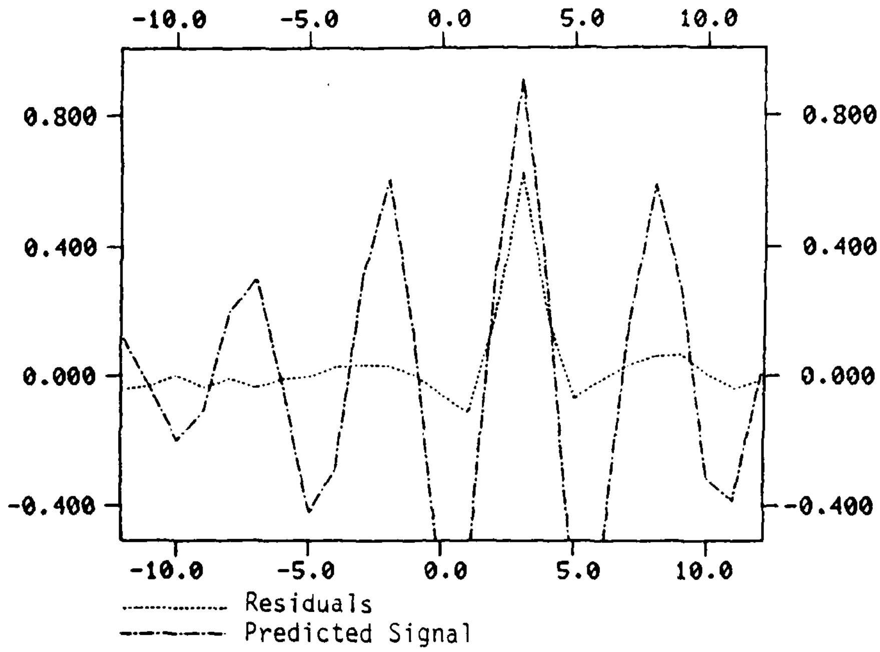


Figure 27A: Correlation of Residuals and Predictions
Signal: a_1=-.606, a_2=.81, SNR=10dB
Algorithm: RML2, n_a=n_c=2, n_B=0, P(o)=10, KA=.8
N=512, λ(o)=.95, λ_o=.99, INIT=SPEC

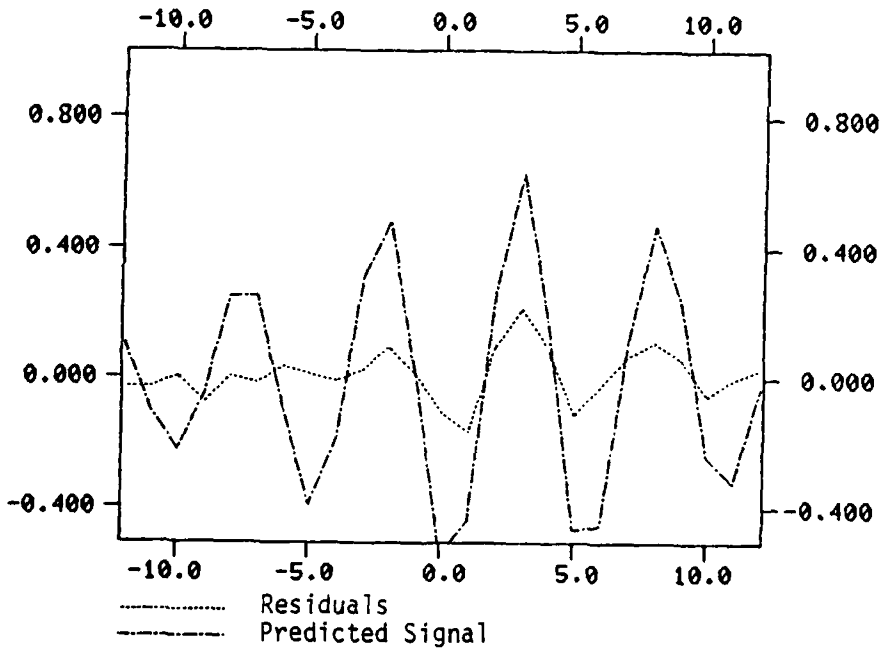


Figure 27B: Correlation of Residuals and Predictions
Signal: a_1=.606, a_2=.81, SNR=-dB
Algorithm: RML2, n_a=n_c=2, N_b=0, P(o)=1, KA=.8,
N=512, λ(o)=.95, λ_o=.99, INIT=SPEC

A third approach to TDOA estimation is based on the idea that the residuals $\varepsilon_i(t)$ provide an estimate of the white driving process $u(t)$. Therefore, it is possible to use the residual $\varepsilon_1(t)$ computed for sensor #1 together with the received data $y_2(t)$ in sensor #2 as the "known" input and output of an ARMA model, and thus apply the RLS estimation algorithm to find its parameters (see Figure 28).

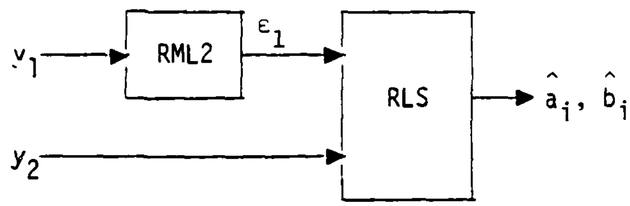


Figure 28. TDOA Estimation by Estimating The Input to the Spectral Model

The residual process $\varepsilon_1(t)$ is in fact a noisy estimate of the input process $u(t)$. It has two components: one due to the measurement noise, and the other due to the unpredictable part of the signal. In wideband signals, the second component is significant and we may expect $\varepsilon_1(t)$ to provide a reasonable estimate of $u(t)$. However, in narrowband signals, which are highly predictable, the second component is small and $\varepsilon_1(t)$ is a very noisy estimate. (In fact, for pure sine waves the second component vanishes!)

These statements are substantiated both by theory and by tests. We found that for pure sine waves, the residuals eventually converge to the measurement noise, and no longer contain information about the signal. For AR processes which are not pure sine waves, the method described above worked satisfactorily in sufficient high SNR. The more narrowband the signal, the worse the performance obtained for a given SNR.

The last approach that was considered for TDOA estimation was to perform multichannel (single input-multiple outputs) parameter estimation using an extension of the RML2 algorithm. One form of the multichannel algorithm, suitable for the no noise case (SNR=∞), was implemented and tested successfully. As expected, no problems occurred in the no-noise case. The algorithm requires some modifications before it can be used on noisy data (see [2]).

Computational Requirements

The computational requirements of the MTS algorithm, as any other algorithm, are difficult to estimate since they depend strongly on a particular implementation. Furthermore, a major part of the computational load is due to data handling, I/O, and the interactive nature of our current program. However, a useful indicator of the amount of computation involved is given by counting the number of operations (multiplies and adds) needed to compute equations (14) and (22), which constitute the basic RML2 algorithm. An approximate count gives $\sim(4n + 5n^2)$ multiplies (where n = the number of estimated parameters) and a comparable number of adds, per single update. If the algorithm operates on M sensors for N data points, the total count becomes

$$\text{No. of operations} \sim(4n + 5n^2)MN = 2(4n + 5n^2)MBT$$

Assuming a typical set of parameters:

$$n = 20, \quad M = 5, \quad B = 10 \text{ Hz}, \quad T = 1 \text{ sec.}$$

we get $2\text{x}10^5$ operations per second. It should be emphasized that this figure is a very rough estimate. Alternative forms of these algorithms are currently available which are more efficient (the so-called "fast" algorithms); however, they were not implemented at this stage of the development.

5. WORK IN PROGRESS

As mentioned earlier, the results presented in this report are only preliminary. We are continuing our investigation in two principal directions:

(i) Algorithm Development/Refinement

The experience gained in testing the MTS algorithm leads us to believe that the performance achieved so far can be further improved. Some of the specific issues which are currently addressed include:

- improved convergence by monitoring the stability of the filter $\hat{C}(z)$, and adjusting the parameter vector $\hat{\theta}$ so that the roots of C(z) will stay inside the unit circle. The results of some initial tests are depicted in Figures 29A-29D. Note the very substantial improvement that was obtained compared to Figures 15A, 19A, 20A, and the fact that Figure 29D corresponds to SNR =-15dB!
- development of algorithms that incorporate structural constraints of the estimated parameters (e.g., the fact that the $\{c_i\}$ parameters are related to the $\{a_i\}$ parameters via equation (31)).

(ii) Algorithm Testing and Performance Evaluation

After developing the core MTS program, we are now in a position to perform a more comprehensive set of tests to study the performance of our algorithms. Specific issues which are being investigated include:

- test the tracking capability of the MTS algorithm on synthetic data with time varying target parameters (TDOA and spectrum).

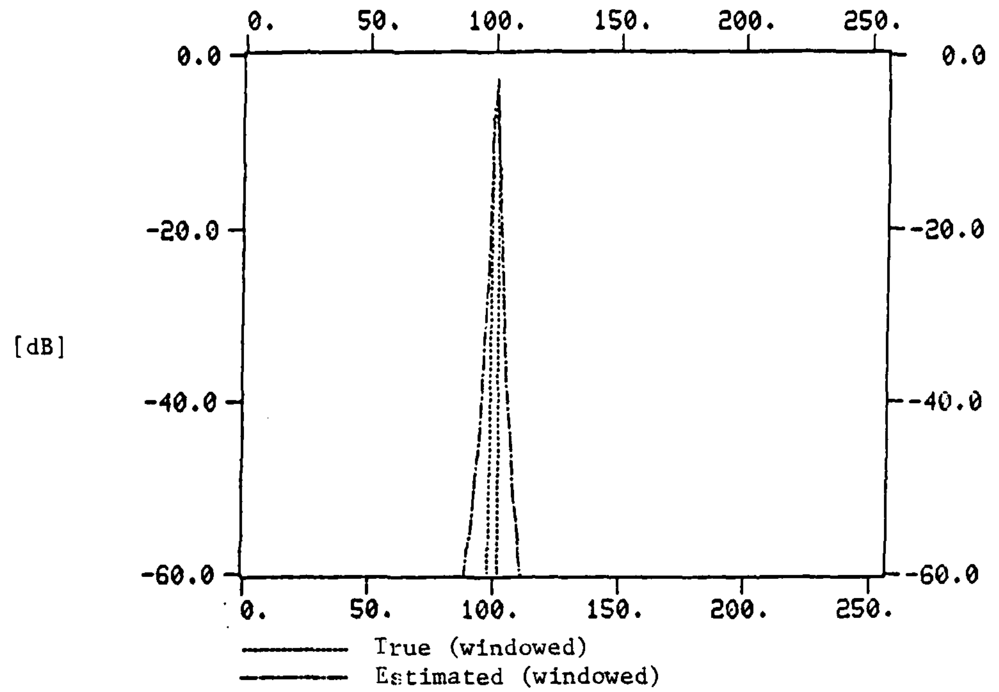


Figure 29A: Estimated and True Spectrum -- SNR=-10dB, N=2048, Stability Monitoring

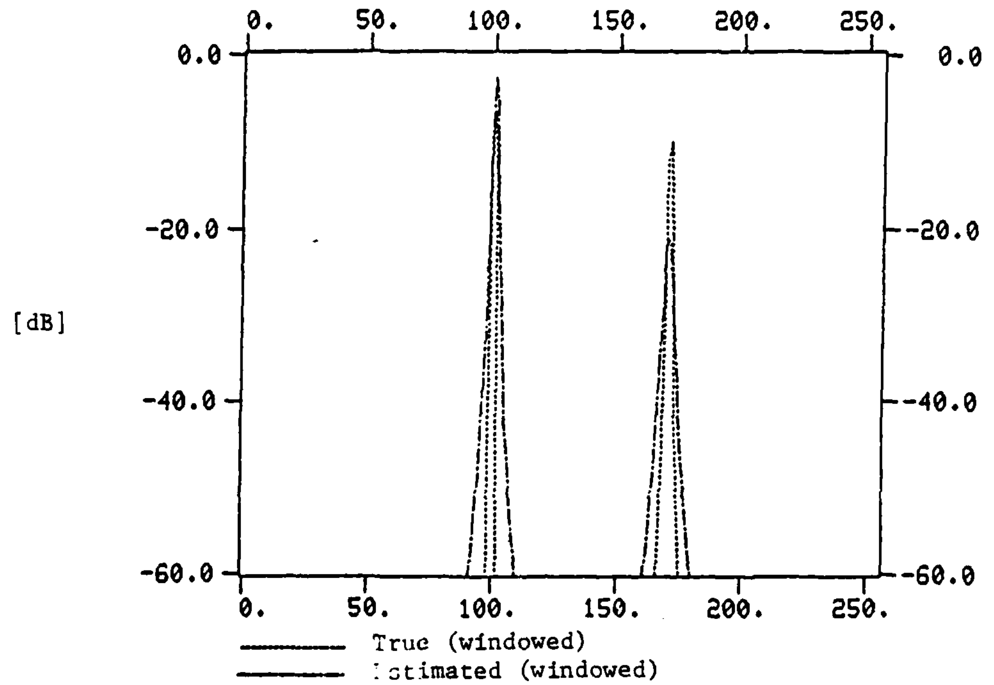


Figure 29B: Estimated and True Spectrum -- SNR=-5dB, N=1024, Stability Monitoring

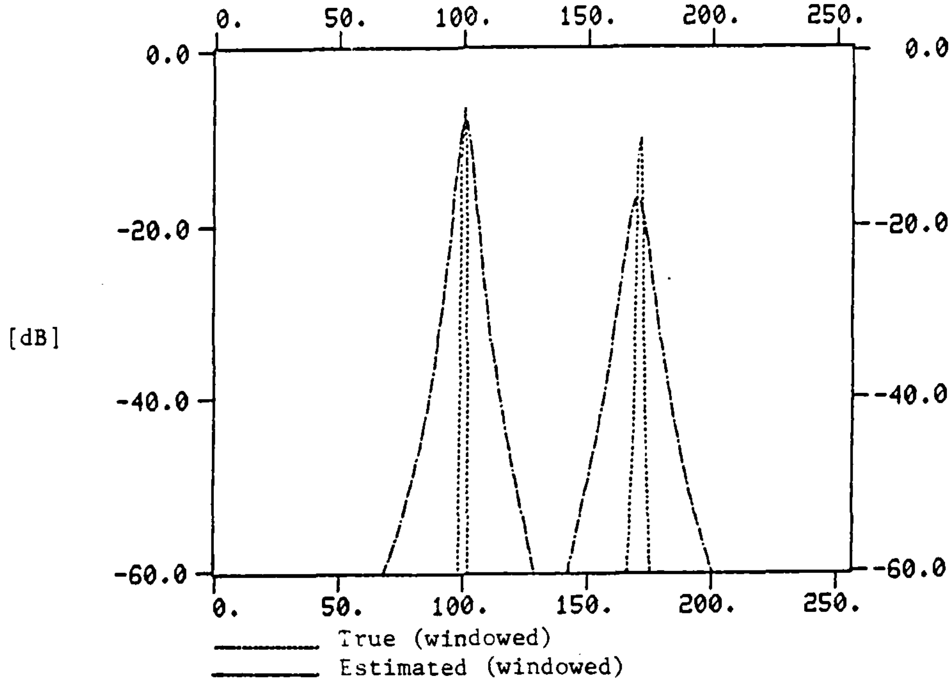


Figure 29C: Estimated and True Spectrum -- SNR=-10dB, N=1024, Stability Monitoring

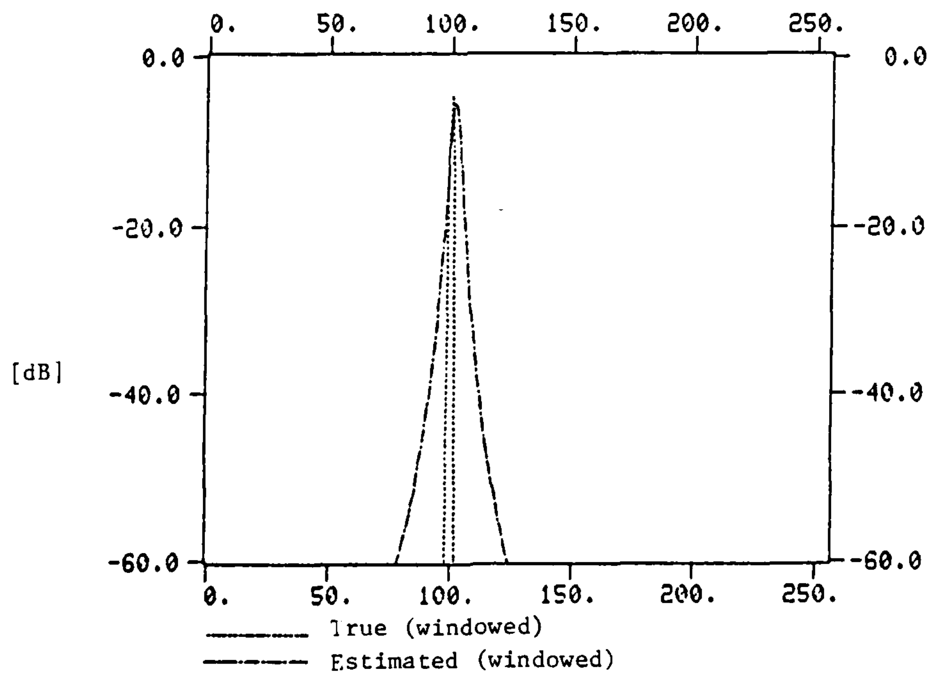


Figure 29D: Estimated and True Spectrum -- SNR=-15dB, N=2048, Stability Monitoring

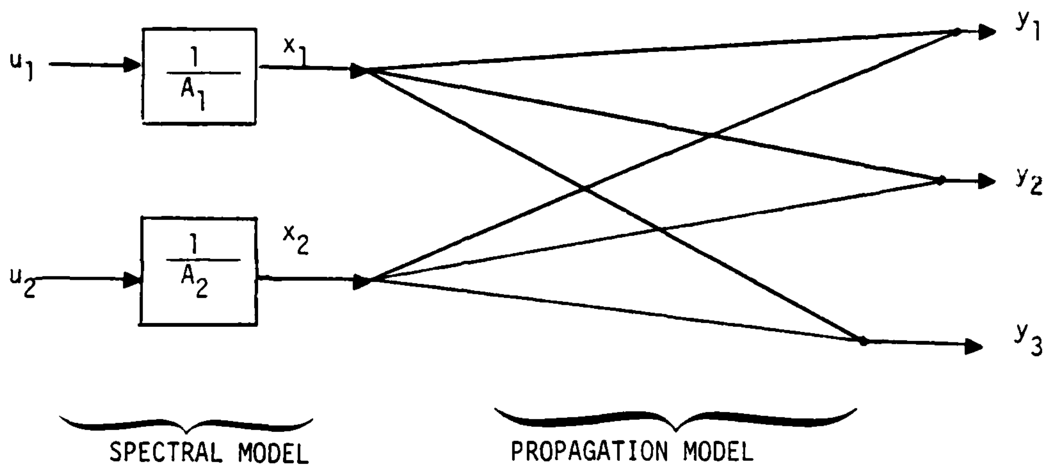
- Test algorithm performance under a variety of conditions including multipath, and more realistic (but still synthetic) data.

In addition to this work which is part of Phase I of the project, we are also studying some of the problems to be addressed in Phase II, i.e., the extension to the multiple target case. This extension will involve fitting a multi-input, multi-output (MIMO) ARMA model to the observed data, as depicted in Figure 30.

We are currently studying some of the basic problems involved in estimating the parameters of MIMO systems and evaluating the modification required to adapt our current MTS algorithm to the multitarget case.

Our approach to the multitarget case will consist of two steps, as mentioned in the introduction. First, we plan to treat the no-noise case. Some of the fundamental issues that need to be addressed are:

- Develop an algorithm for identifying multi-input multi-output systems with unknown inputs. Current techniques are available only for the known input case. Some preliminary work was already performed in the current phase and we do not anticipate any major difficulties.
- Investigate the special structural properties of the MIMO case (e.g., going from Left Matrix Fraction Description to Right Matrix Fraction Description, while preserving the structure (see Appendix A, Equation (18)).
- Study questions of identifiability and uniqueness of the MIMO ARMA model and their relationships to achievable resolutions (e.g., separation of closely spaced targets) and to the discrimination capability of the MTS algorithm.



$$X(z) = A^{-1}(z)\,U(z) \quad , \qquad A(z) = \begin{bmatrix} A_1(z) & 0 \\ 0 & A_2(z) \end{bmatrix}$$

$X(z)$, $U(z)$ are $N\text{x}1$ vectors, N = number of targets

$$y(z) = B(z)\,X(z) = \underbrace{B(z)\,A^{-1}(z)}_{H(z)}\,U(z)$$

$y(z)$ is an $M\text{x}1$ vector, M = number of sensors

Figure 30: Model for the Multitarget Data

- Implement and test a candidate algorithm with emphasis on its tracking ability. The objective will be to demonstrate that after track initiation, the MTS algorithm can provide consistent tracks of several targets.

In the second part of our investigation, we will extend the MTS algorithm to the noisy data case. Some of the basic issues here are:

- How to do proper prefiltering for the MIMO RML2 algorithm.
- Develop the positive real conditions for convergence of the MIMO algorithm and find a way of improving its convergence (as we did in the single target case).
- Implement and test a candidate algorithm. Run a variety of test cases at different SNR's to study convergence behavior.
- Use the experience gained to develop a final version of the MTS algorithm and thoroughly test its tracking capability.

This second step will probably be more difficult than the first, and require more preparation in terms of developing some new theoretical results. However, our experience from the first phase of the project provided us with a clear understanding of the difficulties involved and we feel that the goals of the project can and will be successfully achieved.

The results of the second phase of the MTS project will provide a significant contribution not only to multitarget tracking but also to other areas of interest to the Navy such as: adaptive processing of multichannel signals (noise canceling, adaptive deconvolution, adaptive line enhancement, etc.) and the modeling of vector time-series.

References

1. K. J. Astrom and B. Wittenmark, "On Self-Tuning Regulators," Automatica, Vol. 9, pp. 185-199, 1973.

2. B. Friedlander, "Recursive Identification Algorithms for ARMA Models," Systems Control, Inc., Technical Memorandum TM 5334-01, January 1980.

3. T. Soderstrom, L. Ljung and I. Gustavsson, "A Comparative Study of Recursive Identification Methods," Report 7427, Dept. Automatic Control, Lund, Sweden, 1974.

4. L. Ljung, "On Positive Real Transfer Functions and the Convergence of Some Recursive Schemes," IEEE Trans. on Automatic Control, Vol. AC-22, No. 4, pp. 539-551, August 1977.

5. T. Soderstrom, L. Ljung and I. Gustavsson, "A Theoretical Analysis of Recursive Identification Methods," Automatica, Vol. 14, pp. 231-244, 1978.

6. V. Solo, "The Convergence of AML," IEEE Trans on Automatic Control, Vol. AC-24, No. 6, pp. 958-962, December 1979.

7. S. M. Kay, "The Effects of noise on the Autoregressive Spectral Estimator," IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-27, No. 2, pp. 478-485, October 1979.

8. Y. T. Chan, J. M. Riley and J. B. Plant, "A Parameter Estimation Approach to Time-Delay Estimation and Signal Detection," IEEE Trans. on Acoustics, Speech and Signal Processing," Vol. ASSP-28, No. 1, pp. 8-16, February 1980.

9. G. Clifford, et al., "Coherence Estimation," published by NUSC, Newport Laboratory, Newport, Rhode Island and the New London Laboratory, New London, Connecticut.

10. B. Friedlander, "System Identification Techniques for Adaptive Signal Processing," submitted for publication.

APPENDIX A

System Identification for Multitarget Tracking*

*Published in the Proceedings of the 13th Asilomar Conference on Circuits, Systems and Computers, Pacific Grove, California, November 1979, and was also presented to the National Academy of Sciences Panel on Applied Mathematics Research Alternatives for the U.S. Navy, Washington, D.C., November 2, 1979.

SYSTEM IDENTIFICATION FOR MULTITARGET TRACKING

B. Friedlander and J. J. Anton
Systems Control, Inc.
1801 Page Mill Road
Palo Alto, CA 94304

Abstract

A new approach is presented for locating multiple targets from signals received by a number of spatially distributed sensors. A multi-input, multi-output model is fitted to the observed data. It is shown that the model parameters provide simultaneous estimates of the locations of all targets, as well as their spectra. System identification techniques are applied to perform the model fitting.

INTRODUCTION

Tracking multiple targets represents special difficulties since there can be uncertainties associated with the measurements beyond their inaccuracy usually modeled by some additive noise. This additional uncertainty is related to the origin of the measurements. Since several targets are present, it is necessary to sort out which measurement corresponds to which target. In other words, in addition to the problem of detection and bearing/range estimation, there is a problem of properly labeling the set of measurements. The latter problem is usually referred to as target association or track formation.

Typically, these two facets of multitarget tracking are treated separately. First, a set of potential target locations is obtained. Then some method is used to label these locations by the targets to which they correspond in a manner consistent with previous measurements. Techniques for labeling or multitarget tracking have been developed using various approaches including:

- Kalman filtering (for active sonar [1]; for radar [2])
- Bayesian methods [3]-[7]
- Integer programming [8]
- Track splitting [9], [10], [1], [2]

In all of these techniques the basic detection and location estimation are performed separately for each target. The multitarget aspect of the problem enters only through the labeling procedure.

In this paper we present a radically different approach to multitarget tracking based on simultaneous estimation of multitarget locations. The approach, fundamentally coherent time-domain processing, fits a multi-input, multi-output model to the observed data. The inputs are the signals generated by the targets and the outputs are the sensor measurements. It is shown that the model parameters contain information about the locations of all the targets, as well as other useful information (e.g., the target spectrum). Once these model parameters are labeled by the corresponding targets, that labeling will be consistently maintained from one time scan to the next without need for rechecking or relabeling.

The approach described here has a number of additional features which make it attractive for multitarget and even single target tracking:

- Simultaneous estimation of target location and emitter spectrum.
- Possibility of handling multipath effects.
- Capability of handling nonstationary target and noise statistics.

THE SINGLE TARGET CASE

To illustrate the basic ideas of our approach, we start by looking at a single target. In the next section we will show how to extend the approach to multiple targets.

Consider the following simple problem shown in Figure 1. Two sensors are measuring the signal $x(t)$ propagating from a target located somewhere in the plane. We assume that the propagation involves only some time delays and attenuation. Thus, the outputs y_1, y_2 of the two sensors can be modeled as:

$$y_1(t) = x(t - \tau_1) + n_1(t) \tag{1a}$$

$$y_2(t) = \alpha x(t - \tau_2) + n_2(t) , \tag{1b}$$

where τ_1, τ_2 are the propagation delays from the target to the two sensors, α represents attenuation and n_1, n_2 are independent measurement noise processes. The time sampled version of these outputs will be written as:

$$y_1(k) = x(k - D_1) + n_1(k) \tag{2a}$$

$$y_2(k) = \alpha x(k - D_2) + n_2(k) , \tag{2b}$$

where

$$t = k\Delta t, \quad \tau_1 = D_1 \Delta t, \quad \tau_2 = D_2 \Delta t .$$

Note that the delays τ_1, τ_2 are assumed to be integer multiples of the sampling period. No difficulties arise when τ_1, τ_2 are non-integer multiples provided that the sampling period Δt is properly chosen. This point will be discussed in more detail later.

An assumption that is often made in the context of spectral estimation (e.g., the Maximum Entropy Method [14]-[16]) is that the received signal process $x(t)$ can be represented as an autoregressive process of order n, i.e.,

$$x(k) = -\sum_{i=1}^{n} a_i x(k-i) + u(k) , \tag{3}$$

where $u(k)$ is a white noise driving process. This assumption is not essential to our approach and is introduced here for simplicity. (We will show later how to handle more general emitters, namely, those with rational spectra.)

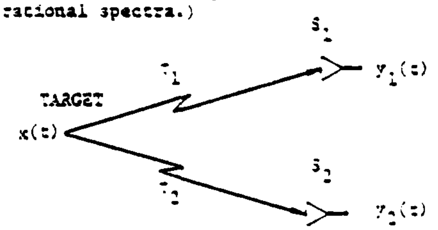


Figure 1. Two Sensors and One Target

Taking the z-transform of Eq. (2) we get

$$Y_1(z) = z^{-D_1}X(z) + N_1(z) = z^{-D_1}\frac{U(z)}{a(z)} + N_1(z) \tag{4a}$$

$$Y_2(z) = cz^{-D_2}X(z) + N_2(z) = cz^{-D_2}\frac{U(z)}{a(z)} + N_2(z) \ , \tag{4b}$$

where

$$a(z) = 1 + \sum_{i=1}^{n} a_i z^{-i} \ . \tag{5}$$

Written in vector form the transfer function from the driving process u to the sensor outputs y_1, y_2 is

$$Y(z) = B(z)X(z) + N(z) = B(z)\frac{1}{a(z)}U(z) + N(z) \ , \tag{6a}$$

where

$$B(z) = \begin{bmatrix} b_1(z) \\ \\ b_2(z) \end{bmatrix} = \begin{bmatrix} z^{-D_1} \\ \\ cz^{-D_2} \end{bmatrix} . \tag{6b}$$

Note that the numerator of this transfer function contains the information about the target location (i.e., the TDOA), while the denominator contains the dynamics of the target emitter signal.

Note that Eq. (7) can be rewritten in a slightly different form. If we multiply through by $a(z)$ and take the inverse z-transform, we get

$$y(k) = -\sum_{i=1}^{n} a_i y(k-i) + B_i u(k-i) + e(k) \ , \tag{7a}$$

where $e(k)$ is a correlated noise process given by

$$e(k) = n(k) + \sum_{i=1} a_i n(k-i) \ . \tag{7b}$$

In other words, the output vector $y(k)$ can be represented as an auto-regressive moving-average (ARMA) process of order n.

Equation (7) immediately suggests a method for estimating TDOA: Find a set of coefficients $\{\hat{a}_i, \hat{B}_i\}$ that best fits the data $y(k)$ in the mean square error sense. This can be done by various parameter estimation techniques to be discussed in a later section. Once the estimates $\{\hat{a}_i, \hat{B}_i\}$ are found, the TDOA can be evaluated by looking at $\hat{B}_i$.

To see this more clearly, consider the example of Figure 1, with $D_1 = 1$, $D_2 = 3$, i.e.,

$$y_1(k) = x(k-1) + n_1(k) \ , \tag{8a}$$

$$y_2(k) = x(k-4) + n_2(k) \ , \tag{8b}$$

which means TDOA = 3. In this case, according to Eq. (6),

$$b_1(z) = z^{-1} \ , \tag{9a}$$

$$b_2(z) = cz^{-4} \ . \tag{9b}$$

Thus, if we estimate the coefficients of $b_1(z)$, $b_2(z)$ by performing an ARMA fitting on the data set $\{y(k)\}$, we will expect to see the situation depicted in Figure 2. Note that the first significant non-zero coefficient of $b_1(z)$ is b_{11}, and the first non-zero coefficient of $b_2(z)$ is $b_{2,4}$. Thus, the TDOA in this case is $4-1 = 3$. The small non-zero values of the other coefficients are due to imperfect modeling because of measurement noise.

Remarks

(1) It should be noted that the numerator polynomial found by such model fitting will not be unique, since without having direct measurements of $x(k)$ the absolute delays (i.e., the degrees D_1, D_2 of the polynomials b_1, b_2) cannot be determined. However, the difference in the degrees of the polynomials b_1, b_2 will be unique and equal to the TDOA, $d_{12} = D_1 - D_2$, which provides the desired information about the source bearing.

(2) Figure 2 also provides an indication of what will happen if the delays are non-integer multiples of the sampling period: instead of having a single non-zero coefficient associated with a given delay, we will have two large coefficients whose relative magnitudes reflect how close the real delay is to the delay represented by that coefficient. (For example, if $k\Delta t \le \tau_1 \le (k+1)\Delta t$, we may expect both $b_{1,k}$ and $b_{1,(k+1)}$ to be non-zero.) Evidently, as long as the sampling rates are sufficiently high compared to the bandwidth of the underlying process, the same approach will work for non-integer time-delays.

(3) The approach described here can also handle multipath. In the case of multipath $b_1(z)$ (and $b_2(z)$) will have several large coefficients corresponding to the direct path delay and the multipath delays. Since the direct path has the shortest delay, the *first* large coefficient of $b_1(z)$ will correspond to the direct-path delay. Thus, the TDOA can be easily evaluated from the $\{B_i\}$ coefficients even in the presence of multipath!

THE MULTITARGET CASE

The ARMA modeling approach can be easily extended to the multitarget case. Here the system consisting of targets and sensors will be represented by a multi-input, multi-output transfer function (i.e., ARMA model). A simple example is depicted in Figure 3.

The equation describing the vector of measured data $y(k)$ is given by

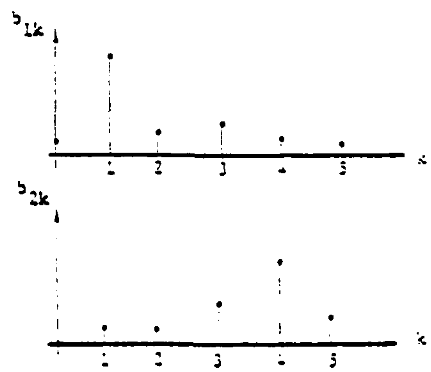


Figure 2. Numerator Coefficients

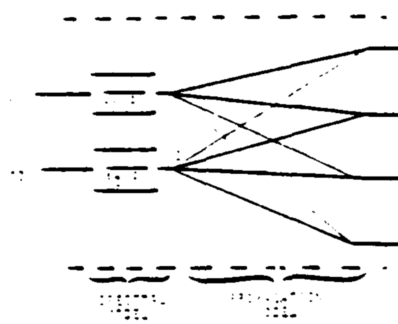


Figure 3. The Multitarget Case

$$Y(z) = B(z)X(z) + N(z) \quad (10a)$$

where

$$X(z) = \begin{bmatrix} x_1(z) \\ \vdots \\ x_n(z) \end{bmatrix}, \quad B(z) = [B_1(z), \ldots, B_n(z)] \quad (10b)$$

If, as before, we assume that the signals emitted by the targets $x_i(z)$ are autoregressive processes, then

$$X(z) = A^{-1}(z)U(z) \quad (11a)$$

where

$$A(z) = \begin{bmatrix} a_1(z) & & 0 \\ & \ddots & \\ 0 & & a_n(z) \end{bmatrix}, \quad U(z) = \begin{bmatrix} u_1(z) \\ \vdots \\ u_n(z) \end{bmatrix} \quad (11b)$$

Thus, the output of the sensors in the multitarget case can be written as

$$Y(z) = B(z)\, A^{-1}(z)\, U(z) + N(z) \quad (12)$$

The question of finding the target TDOA's is equivalent to the problem of estimating the coefficients $\{B_i, A_i\}$ of the transfer function $B(z)A^{-1}(z)$. In other words, find a model $B(z)A^{-1}(z)$ that will best fit the available data $\{y(k)\}$. Once the model has been found, the location of the target will be found by examining the coefficients in the appropriate column of $B(z)$, and the target spectra can be found from the appropriate elements of $A(z)$.

Actually, Eq. (12) is not quite the form we get when ARMA modeling is performed, since the multi-input, multi-output ARMA model has the form

$$y(k) = -\sum_{i=1}^{n} \bar{A}_i y(k-i) + \sum_{i=1}^{n} \bar{B}_i u(k-i) \quad (13)$$

Taking z-transforms will give

$$\bar{A}(z)\, Y(z) = \bar{B}(z)\, U(z) \quad (14)$$

or

$$Y(z) = \bar{A}^{-1}(z)\, \bar{B}(z)\, U(z) \quad (15)$$

where

$$\bar{A}(z) = I + \sum_{i=1}^{n} \bar{A}_i z^{-i} \quad (16)$$

$$\bar{B}(z) = \sum_{i=1}^{n} \bar{B}_i z^{-i} \quad (17)$$

By comparing (14) and (15) we note that the coefficients $\{\bar{A}_i, \bar{B}_i\}$ are not precisely those of the transfer function $B(z)A^{-1}(z)$. In fact, they are related by the equation

$$\bar{A}^{-1}(z)\, \bar{B}(z) = B(z)\, A^{-1}(z) \quad (18)$$

Thus, if we use some parameter estimation technique to fit an ARMA model $\{\bar{A}_i, \bar{B}_i\}$ to the data $\{y(k)\}$, we will have to perform afterwards the step of computing $\{A_i, B_i\}$ from $\{\bar{A}_i, \bar{B}_i\}$, and then evaluate the TDOA's. Various techniques are available for performing these computations [11]-[12].

Note that in the proposed approach, there is no need to "label" the targets at each step. It is necessary, of course, to establish initially the labeling of the columns of $B(z)$ (or $A(z)$), i.e., determine which column shall refer to which target. Afterward, the estimates of the ARMA coefficients are updated at each time step k according to the new data vector $y(k)$.

The proposed approach performs a recursive global estimation process, i.e., for all targets over the chosen time interval. The algorithm automatically tries to fit a set of coefficients which "explain" in the best way all the available data. Since these coefficients contain the TDOA information for all the targets (as well as their spectra), we get a set of consistent estimates of all target locations, based on all the available data. If an optimal parameter estimator is used, the resulting TDOA estimates are truly optimal. In the standard tracking approaches discussed in the Introduction, only suboptimal parameter estimates can be obtained, since each target location is first estimated individually and the tracking algorithm then attempts an appropriate labeling of these estimates.

Finally, it should be noted that finding the estimates of the $\{A_i\}$ coefficients is equivalent to performing simultaneous spectral estimation of all the targets, with automatic line association. The first is true, since the spectrum can be computed directly from the autoregressive coefficients just as for the Maximum Entropy Method [13]-[15]. The latter is true for the same reason that no relabeling of the TDOA's is required.

RECURSIVE PARAMETER ESTIMATION

The approach outlined in the previous sections depends on our ability to compute the ARMA coefficients given a set of measurements. This type of problem has been widely studied in the general context of parameter estimation and in the more specific context of identifying system models from input/output measurements [16]-[24].

The least squares parameter estimation problem is usually formulated as follows: given a set of measurements $\{y(k), u(k)\}_{k=0,K}$ find the coefficients $\{\bar{A}_i, \bar{B}_i\}$ that will minimize the mean square error $\sum_k [y(k) - \hat{y}(k)]^2$ where

$$\hat{y}(k) = -\sum_i \bar{A}_i y(k-i) + \sum_i \bar{B}_i u(k-i) \quad (19)$$

The vector $\hat{y}(k)$ is the value predicted by the ARMA model, for the measurement at time k.

When the inputs u(k) are known, the solution of this problem is fairly straightforward and typical algorithms can be found in [16], [17], [21], [22]. The situation is somewhat more complicated when it is not possible to measure u(k). However, by assuming that u(k) is a sequence of independent "white" random variables with unit variance, it is still possible to estimate the model parameters. This case is usually referred to as "the case of correlated residuals," and several techniques have been suggested for its solution. For details, see the survey by Astrom [16].

More recently a new approach has been developed by Morf [13]-[20] which provides efficient forms of the so-called exact recursive least squares algorithms. These new forms have the added advantages of computational efficiency and fast parameter tracking capability [23],[24]. The last property is important for tracking moving targets, since the ARMA model corresponding to such targets has time varying parameters. These algorithms are also capable of handling nonstationary source and noise processes.

CONCLUSIONS

A new technique for multitarget tracking was outlined in this paper. This approach provides a comprehensive framework for detection, estimation and tracking of multiple targets, based on two key ideas:

- Formulating the multitarget problem as a multichannel estimation problem, thus handling all the targets simultaneously.
- Representing the multisensor data by the parameters of a model which fits all the available data. This results in a global (optimal) estimation of all target parameters.

While the technique has not been fully tested, similar ideas have been applied very successfully in spectral estimation (e.g., the Maximum Entropy Method: single channel and multi-channel), and in speech processing (e.g., the LPC method for speed analysis/synthesis). Thus, we have both conceptual and practical reasons to believe that this is a very promising approach with high potential for improving and extending current tracking capabilities.

We are currently in the process of evaluating the performance of this approach. We are also investigating the extension of this technique to: (a) more general forms of linear propagation models, (b) more general target spectra (ARMA), (c) data with significant doppler shifts, and (d) active sonar and radar applications.

REFERENCES

1. E. J. Fraser and L. Meier, "Mathematical Models and Optimum Computation for Computer-Aided Active Sonar Systems," U.S. Navy Electronic Lab., SRI Final Rep. (First Year), San Diego, Calif., Contract N123-(953)54436A, March 1967.

2. P. Smith and G. Buechler, "A Branching Algorithm for Discriminating and Tracking Multiple Objects," IEEE Trans. Auto. Control, Vol. AC-20, Feb. 1975, pp. 101-104.

3. G. A. Ackerson and K. S. Fu, "On State Estimation in Switching Environments," IEEE Trans. Auto. Control, Vol. AC-15, Feb. 1970, pp. 10-17.

4. H. Akashi and H. Kumamoto, "Random Sampling Approach to State Estimation in Switching Environments," Automatica, Vol. 13, July 1977, pp. 429-434.

5. A. G. Jaffer and S. C. Gupta, "Recursive Bayesian Estimation with Uncertain Observation," IEEE Trans. Inform. Theory, Vol. IT-17, Sept. 1971, pp. 614-616.

6. ______, "Optimal Sequential Estimation of Discrete Processes with Markov Interrupted Observations," IEEE Trans. Auto. Control, Vol. AC-16, Oct. 1971, pp. 471-475.

7. N. E. Nahi, "Optimal Recursive Estimation with Uncertain Observations," IEEE Trans. Inform. Theory, Vol. IT-15, July 1969, pp. 456-462.

8. C. L. Morefield, "Application of 0-1 Integer Programming to Multitarget Tracking Problems," in Proc. IEEE Conf. Decision and Control, Dec. 1975 and IEEE Trans. Auto. Control, Vol. AC-22, June 1977, pp. 302-312.

9. R. A. Singer, R. G. Sea, and K. Housewright, "Derivation and Evaluation of Improved Tracking Filters for Use in Dense Multitarget Environments," IEEE Trans. Inform. Theory, Vol. IT-20, July 1974, pp. 423-432.

10. R. W. Sittler, "An Optimal Data Association Problem in Surveillance Theory," IEEE Trans. Mil. Electron, Vol. MIL-8, April 1964, pp. 125-139.

11. B. W. Dickinson, T. Kailath and M. Morf, "Canonical Matrix Fraction and State-Space Descriptions for Deterministic and Stochastic Linear Systems," IEEE Trans. Auto. Control, Vol. AC-19, No. 6, Dec. 1974, pp. 656-557.

12. B. W. Dickinson, "Properties and Applications on Matrix Fraction Description of Linear Systems," Ph.D. dissertation, Stanford University, Palo Alto, California, 1974.

13. M. Morf, A. Vieira, D. T. L. Lee and T. Kailath, "Recursive Multichannel Maximum Entropy Spectral Estimation," IEEE Trans. Geosci. Electronics, Vol. GE-16, No. 2, April 1978, pp. 85-94.

14. J. P. Burg, "Maximum Entropy Analysis," Ph.D. dissertation, Stanford University, Pal9 Alto, California, 1975.

15. F. J. Harris, "A Maximum Entropy Filter," Report NUC TP441, Naval Undersea Center, San Diego, California, Jan. 1975.

16. K. J. Astrom, and P. Eykhoff, "System Identification - A Survey," Automatica, Vol. 7, Jan. 1971, pp. 123-162.

17. P. Eykhoff, System Identification, Parameter and State Estimation, Wiley, New York, 1974.

18. M. Morf and L. Ljung, "Fast Algorithms for Recursive Identification," Proc. Conf. Decision and Control, Florida, Dec. 1976.

19. M. Morf, "Fast Algorithms for Multivariable Systems," Ph.D. dissertati-on, Stanford University, Palo Alto, California, Aug. 1974.

20. M. Morf, D. T. Lee, J. R. Nickolls and A. Vieira, "A Classification of Algorithms for ARMA Models and Ladder Realizations," Conf. Rec., 1977 IEEE Int. Conf. on Acoustics, Speed and Signal Processing, Hartford, 1977, pp. 13-19.

21. T. Soderstrom, L. Ljung and I. Gustavsson, "A Comparative Study of Recursive Identification Methods," Report 7427, Dept. Automatic Control, Lund Institute of Technology, Lund, Sweden, 1974.

22. G. C. Goodwin and R. L. Payne, Dynamic System Identification. Academic Press, New York, 1977.

23. M. Morf, "Ladder Forms in Estimation and System Identification," IEEE Proc. 11th Annual Asilomar Conf. on Circuits, Systems and Computers, Monterey, California, Nov. 7-9, 1977.

24. M. Morf and D. T. Lee, "Recursive Least Squares Ladder Forms for Fast Parameter Tracking," Proc. 1979 IEEE Conf. on Decision and Control, San Diego, California, Jan. 10-12, 1979.

APPENDIX B

TDOA Estimation

Let

$$y_1(t) = x(t) + n_1(t)$$
$$y_2(t) = x(t+\tau) + n_2(t) \tag{B1}$$

represent signals received by two different sensors. The noise processes $n_1(t)$, $n_2(t)$ are assumed to be white, and independent. The two signals are related by

$$y_2(t) = y_1(t+\tau) + n(t) \tag{B2}$$

where

$$n(t) = n_2(t) - n_1(t+\tau).$$

The noise process $n(t)$ has a variance equal to the sum of the variances of n_1 and n_2. The sampled values of y_1, y_2, n will be denoted by $y_1(k\Delta T)$, $y_2(k\Delta T)$, $n(k\Delta T)$. Assuming that the sampling interval ΔT is adequately small for $x(t)$, we have

$$y_1(t) = \sum_{k=-\infty}^{+\infty} y_1(k\Delta T)\ \mathrm{sinc}(t-k\Delta T) \tag{B3}$$

where

$$\mathrm{sinc}(t) = \sin(\pi t/\Delta T)/(\pi t/\Delta T). \tag{B4}$$

Let

$$\tau = \ell\Delta T + \Delta\tau \quad , \qquad 0 \le \Delta\tau < \Delta T.$$

$$y_2(i\Delta T) = \sum_{k=-\infty}^{+\infty} y_1(k\Delta T)\ \mathrm{sinc}[(i+\ell-k)\Delta T + \Delta\tau] + n(i\Delta T) \tag{B5}$$

Without loss of generality, we can set $\Delta T = 1$, and make a change of variables $i-k = n$, which will give

$$y_2(i) = \sum_{n=-\infty}^{+\infty} b_n y_1(i-n) + n(i) \tag{B6}$$

where

$$b_n = \mathrm{sinc}(n + \ell + \Delta\tau) \tag{B7}$$

Thus, the time series $y_2(i)$ is related to $y_1(i)$ by a moving average (MA) filter with coefficients as given by Equation (B7). In practice, we will consider only a finite number (n_b) of terms in the sum (B6).

The coefficients b_n can be considered as the samples of a function $\mathrm{sinc}(n + \ell + \Delta\tau)$ which achieves a maximum at $n + \ell + \Delta\tau = 0$. Hence, given the coefficients b_n, the delay $\hat{\tau}$ is the value which maximizes the function·

$$b(\tau) = \sum_{n=1}^{n_b} b_n \,\mathrm{sinc}(\tau-n) \quad . \tag{B8}$$

In our experiments, we used a search algorithm to find the value $\hat{\tau}$ which maximizes $b(\tau)$. Some typical results are summarized in Table 2, Section 4. A similar approach, which uses a different type of estimation algorithm, can be found in [8].

APPENDIX C

Program Description and Capabilities

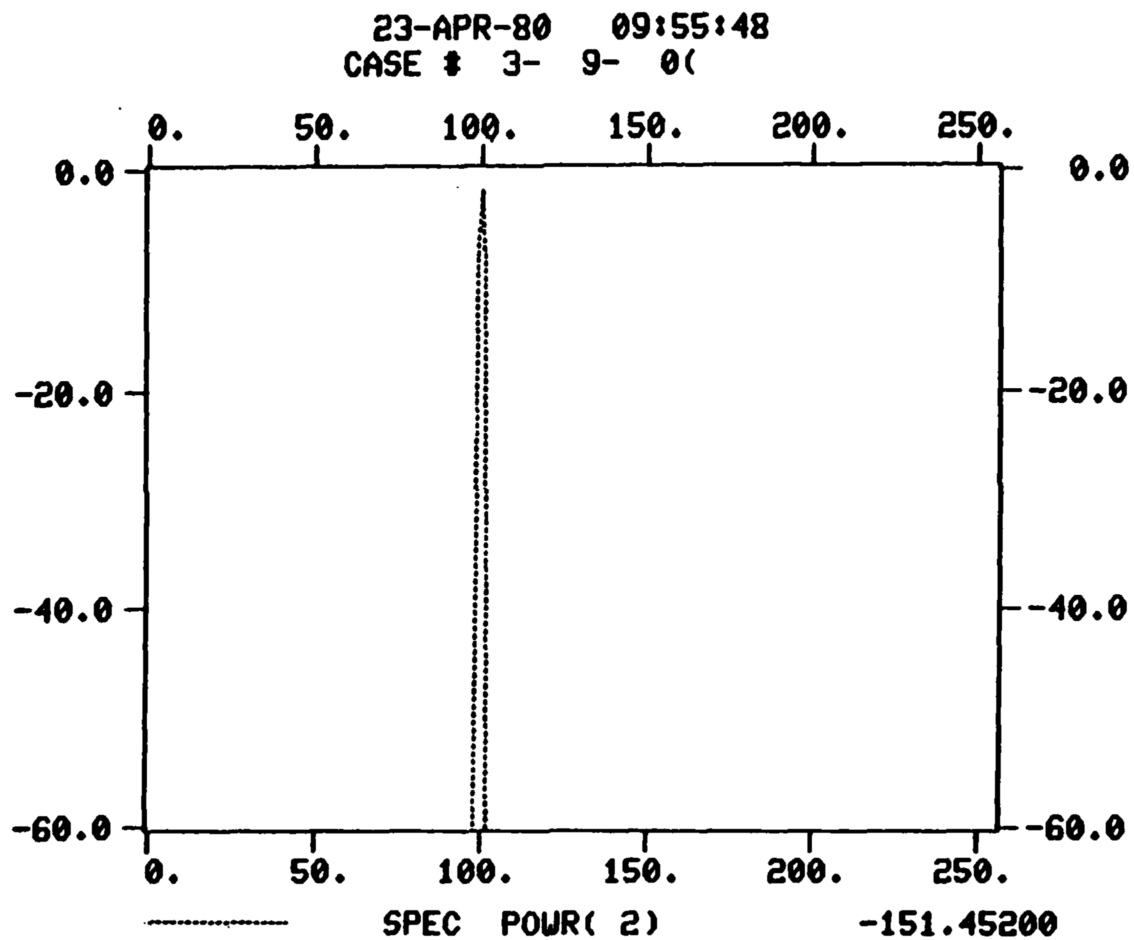
The MTS algorithms were implemented on SCI's VAX. The programs are written in FORTRAN and are fully interactive. Plotting capabilities include a Tektronix display and character displays. The interactive program allows easy changes of test cases (target spectra, signal-to-noise ratio) and algorithm parameters (type of algorithm, model order), as well as convenient program modification.

The following pages present an example of the program parameters under our control and some typical plots obtained for a sample test case.

```
   :
                                        ;
   :
                                            ; ON-LINE HELP COMMAND
   :
                                            ;
   :
                                             HELP

 ALL KEYWORDS MAY BE ABBREVIATED TO 4 LETTERS

             SET        SET VALUE OF ONE OR MORE PARAMETERS
             CORR       PERFORM A CORRELATION
             OPEN       OPEN A TABLE
             SHOW       SHOW CONTENTS OR STRUCTURE OF DATABASE
             STOP       TERMINATE EXECUTION
             SPEC       COMPUTE A SPECTRUM
             DISPLAY    DISPLAY CURVES
             READ       READ COMMANDS FROM A FILE
             DEBUG      SET DEBUG LEVEL

             TARGET     CALCULATE TRANSMITTED SIGNALS
             RCVR       CALCULATE RECEIVED SIGNALS
             IDN1       EXERCISE IDENTIFICATION ALGORITHM #1
             IDN2       EXERCISE IDENTIFICATION ALGORITHM #2
             IDN3       EXERCISE IDENTIFICATION ALGORITHM #3

     TYPE ABORT WHENEVER YOU ARE COMPLETELY FERSHIMMLT

   :
                                        ;
   :
                                        ; READ PRESTORED SCENARIO
   :
                                        ;
   :
                                          READ SETUP.DAT NO
   :
 OPEN TARG OLD TARG.FIL
   :
 OPEN DISP OLD DISP.FIL
   :
 OPEN RCVR OLD RCVR.FIL
   :
 OPEN IDN1 OLD IDN1.FIL
   :
 OPEN IDN2 OLD IDN2.FIL
   :
 OPEN IDN3 OLD IDN3.FIL
   :
 OPEN CORR OLD CORR.FIL
   :
 OPEN SPEC OLD SPEC.FIL
   :
 SPEC ABORT
   :
 CORR ABORT
   :
   :
                                        ;
```

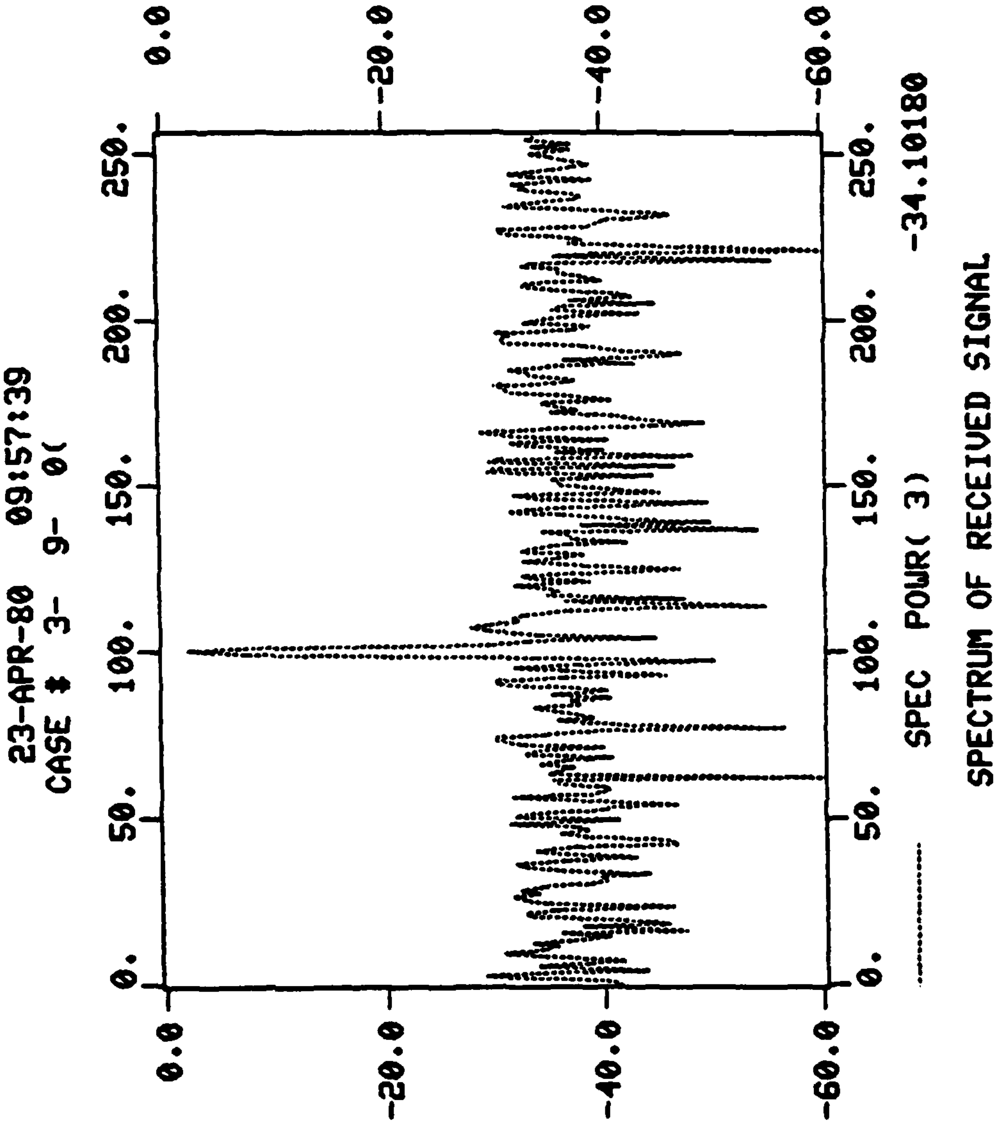
```
:
                                        ; SHOW NAMES OF FILES
:
                                        ;
:
                                         SHOW ALL NULL

   LUN    NAME    STATUS    HITM   DITM   TITM      DESCRIPTION

   21     TARG     OPEN      16      2      0      TARGET PARAMS
   22     RCVR     OPEN      11      8      0      RECEIVER PARAMS
   23     DISP     OPEN      15      0      0      DISPLAY PARAMS
   24     IDN1     OPEN      31     18      0      IDENTIFIER ALG 1
   25     IDN2     OPEN      31     18      0      IDENTIFIER ALG 2
   26     IDN3     OPEN      31     18      0      IDENTIFIER ALG 3
   27     SPEC     OPEN       9      1      0      SPECTRAL DATA
   28     CORR     OPEN      10      1      0      CORRELATIONS
:
                                        ;
:
                                        ; SHOW STRUCTURE OF A FILE
:
                                        ;
:
                                         SHOW TARG ITEMS

   LUN    NAME    STATUS    HITM   DITM   TITM      DESCRIPTION

   21     TARG     OPEN      16      2      0      TARGET PARAMS

          NAME     TYPE    REGN     SIZE    LOC       TITLE

          CASE     INT      HEDR      1       1
          NTGT     INT      HEDR      1       2
          TYPE     CHAR     HEDR      1       3
          MEAN     REAL     HEDR      1       4
          SIGU     REAL     HEDR      1       5
          SIGN     REAL     HEDR      1       6
          NA       INT      HEDR      1       7
          NF       INT      HEDR      1       8
          NTIM     INT      HEDR      1       9
          TIM1     REAL     HEDR      1      10
          DTIM     REAL     HEDR      1      11
          A        REAL     HEDR      8      12
          PER      REAL     HEDR      8      20
          AMP      REAL     HEDR      8      28
          BW       REAL     HEDR      8      36
          PERM     REAL     HEDR      8      44
          U        REAL     DATA      1       1
          X        REAL     DATA      1       2
:
                                        ;
:
                                        ; SHOW CONTENTS OF A FILE
:
                                        ;
:
                                         SHOW TARGET VALUES
```

```
VALUES FOR TARG
   CASE =        3
   NIGT =        1
   TYPE = FMOD
   MEAN =        0.00000
   SIGU =        0.20000
   SIGN =        0.00000
   NA   =        2
   NF   =        1
   NTIM =      512
   TIM1 =        0.00000
   DTIM =        1.00000
   A    =       -0.67300     1.00000     0.00000     0.00000     0.00000
                 0.00000     0.00000     0.00000
   PER  =        5.12000     3.00000     1.00000     1.00000     1.00000
                 1.00000     1.00000     1.00000
   AMP  =        1.41400     1.41400     0.00000     0.00000     0.00000
                 0.00000     0.00000     0.00000
   BW   =        0.00000     0.00000     0.00000     0.00000     0.00000
                 0.00000     0.00000     0.00000
   PERM =     2000.00000  2000.00000 10000.00000     1.00000     1.00000
                 1.00000     1.00000     1.00000
   :
                              ;
   :
                              ; EXECUTE TARGET--1 SINE WAVE UNMODULATED
   :
                              ;
   :
                               TARGET
 MEAN-U, MEAN-X, SIG-U, SIG-X=  0.000E+00 -5.907E-06  0.000E+00  1.001E+00
   :
                              ;
   :
                              ; COMPUTE AND DISPLAY TARGET SPECTRUM
   :
                              ;
   :
                               SPEC 0
   1  TIME  IDN1  PRED(   1)   WINDOW=HANF
   2  TIME  TARG  X   (   1)   WINDOW=HANF
   3  TIME  RCVR  Y1  (   1)   WINDOW=HANF
   4  IMPL  IDN1  A   (   1)   WINDOW=NULL
   5  IMPL  IDN2  A   (   1)   WINDOW=NULL
   6  TBD   TBD   TBD (   1)   WINDOW=NULL
   7  TBD   TBD   TBD (   1)   WINDOW=NULL
   8  TBD   TBD   TBD (   1)   WINDOW=NULL
   9  TBD   TBD   TBD (   1)   WINDOW=NULL
  10  TBD   TBD   TBD (   1)   WINDOW=NULL
          SPECTRUM NUMBER OR 0 FOR HELP:
                              2
   :
                              DISP 2 /
   :
                              ;
   :
                              ; EXECUTE RECEIVER--SNR=10
   :
                              ;
```



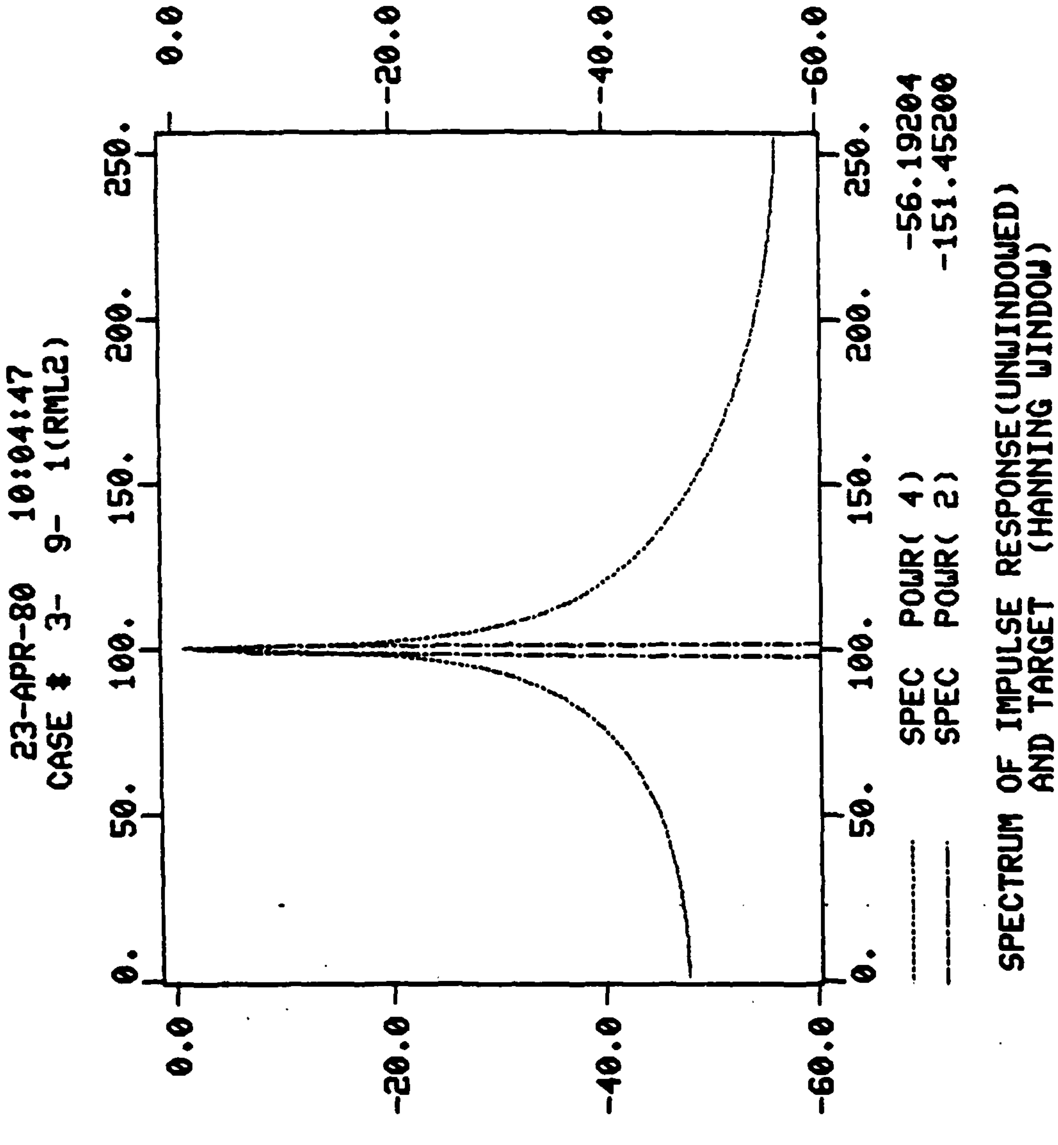
SPECTRUM OF TARGET IN DB

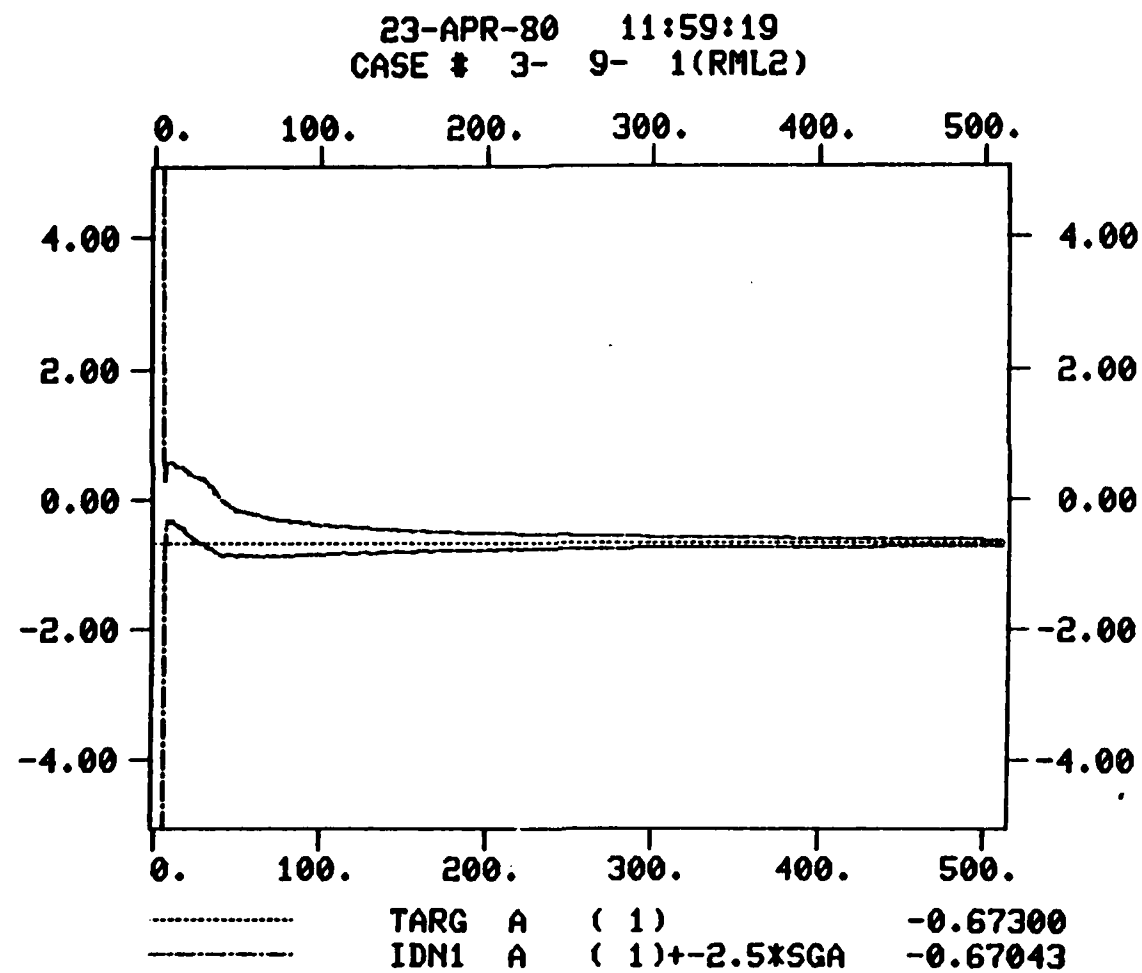
```
:
                              SHOW RCVR VALU

VALUES FOR RCVR
   CASE =       9
   NRCV =       1
   SNR  =      10.00000      1.00000    100.00000    100.00000
   NB   =       4
   NTIM =     512
   TIM1 =       0.00000
   DTIM =       1.00000
   B1   =       1.00000      0.00000      0.00000      0.00000      0.00000
                0.00000      0.00000      0.00000
   B2   =       0.00000      0.00000      0.00000      1.00000      0.00000
                0.00000      0.00000      0.00000
   B3   =       0.00000      0.00000      0.00000      0.00000      0.00000
                0.00000      0.00000      0.00000
   B4   =       0.00000      0.00000      0.00000      0.00000      0.00000
                0.00000      0.00000      0.00000
:
                              RCVR

MEAN-N, MEAN-Y, SIG-N, SIG-Y=  1.461E-02  1.368E-02  3.122E-01  1.058E+00

:
                              ;
:
                              ; DISPLAY RECEIVER SPECTRUM
:
                              ;
:
                              SPEC 3
:
                              DISP 3 /
:
                              ;
:
                              ; EXECUTE IDENTIFIER--RML2
:
                              ;
:
                              SHOW IDN1 VALU

VALUES FOR IDN1
   CASE =       1
   ALG  = RML2
   TYPE = TIME
   INIT = ZERO
   NA   =       2
   NB   =       0
   NC   =       2
   ALPA =     100.00000
   BETA =       0.00000
   FL10 =       0.95000
   FL20 =       1.00000
   FLCN =       0.99000
   TRMN =       0.00000
   RCON =       0.00000
   RA   =       0.60000
```


23-APR-80 09:57:39
CASE # 3- 9- 0(
SPEC POWR(3)
-34.10180
0. 50. 100. 150. 200. 250.
0.0 -20.0 -40.0 -60.0
SPECTRUM OF RECEIVED SIGNAL

```
KC     =          0.00000
NTIM   =       512
T1     =          0.00000
DT     =          1.00000
OTBL   = RCVR
OVBL   = Y1
OSUB   =          1         1
IIBL   = NULL
IVBL   = NULL
ISUB   =          1
A1     =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
B11    =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
B12    =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
B13    =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
B14    =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
C1     =          0.00000        0.00000        0.00000        0.00000        0.00000
                  0.00000        0.00000        0.00000
 :
                                        IDN1
  M,N=       1       4

MEAN-R, SIG-R=       0.03729        1.14272


K,AEST= 512      -0.57043       0.99443

 CEST=            -0.45644       0.60152

TRACE,ETA,RESIDS=        0.04644       15.49508       -0.41171
 :
                                        ;
 :
                                        ; SHOW SPECTRUM OF IMPULSE RESPONSE
 :
                                        ;
 :
                                        SPEC 4
 :
                                        DISP 4 2 /
 :
                                        ;
 :
                                        ; DISPLAY TIME HISTORY OF ESTIMATED A1
 :
                                        ;
 :
                                        SET DISP XVBL TIME /
 :
                                        DISP 0
  1   U    ( 1)+-0.0*        2   X    ( 1)+-0.0*        3   A    ( 1)+-0.0*
  4   A    ( 2)+-0.0*        5   A    ( 3)+-0.0*        6   A    ( 4)+-0.0*
  7   A    ( 5)+-0.0*        8   A    ( 6)+-0.0*        9   A    ( 7)+-0.0*
 10   A    ( 8)+-0.0*       11   N1   ( 1)+-0.0*       12   Y1   ( 1)+-0.0*
 13   B1   ( 1)+-0.0*       14   B1   ( 2)+-0.0*       15   B1   ( 3)+-0.0*
```

```
16  B1   (  4)+-0.0*        17  B1   (  5)+-0.0*        18  B1   (  6)+-0.0*
19  B1   (  7)+-0.0*        20  B1   (  8)+-0.0*        21  TRAC(  1)+-0.0*
22  ETA  (  1)+-0.0*        23  LAM1(  1)+-0.0*        24  PRED(  1)+-0.0*
25  RES  (  1)+-0.0*        26  A    (  1)+-2.5*SGA    27  A    (  2)+-2.5*SGA
28  A    (  3)+-2.5*SGA     29  A    (  4)+-2.5*SGA    30  A    (  5)+-2.5*SGA
31  B1   (  1)+-2.5*SGB1    32  B1   (  2)+-2.5*SGB1   33  B1   (  3)+-2.5*SGB1
34  B1   (  4)+-2.5*SGB1    35  B1   (  5)+-2.5*SGB1   36  C    (  1)+-2.5*SGC
37  C    (  2)+-2.5*SGC     38  C    (  3)+-2.5*SGC    39  C    (  4)+-2.5*SGC
40  C    (  5)+-2.5*SGC     41  TRAC(  1)+-0.0*        42  ETA  (  1)+-0.0*
43  LAM1(  1)+-0.0*         44  PRED(  1)+-0.0*        45  RES  (  1)+-0.0*
46  A    (  1)+-2.5*SGA     47  A    (  2)+-2.5*SGA    48  A    (  3)+-2.5*SGA
49  A    (  4)+-2.5*SGA     50  A    (  5)+-2.5*SGA    51  B1   (  1)+-2.5*SGB1
52  B1   (  2)+-2.5*SGB1    53  B1   (  3)+-2.5*SGB1   54  B1   (  4)+-2.5*SGB1
55  B1   (  5)+-2.5*SGB1    56  C    (  1)+-2.5*SGC    57  C    (  2)+-2.5*SGC
58  C    (  3)+-2.5*SGC     59  C    (  4)+-2.5*SGC    60  C    (  5)+-2.5*SGC
                NEXT CURVE #, / OR 0 FOR HELP:
                                 3 26 /
    :
 STOP
```


23-APR-80 10:04:47
CASE # 3- 9- 1(RML2)
0.
50.
100.
150.
200.
250.
0.0
-20.0
-40.0
-60.0
SPEC POWR(4) -56.19204
SPEC POWR(2) -151.45200
SPECTRUM OF IMPULSE RESPONSE(UNWINDOWED)
AND TARGET (HANNING WINDOW)


23-APR-80 11:59:19
CASE # 3- 9- 1(RML2)
0. 100. 200. 300. 400. 500.
4.00 2.00 0.00 -2.00 -4.00
TARG A (1) -0.67300
IDN1 A (1)+-2.5*SGA -0.67043

Distribution List
for
"Multi-Target Tracking Studies"

All addressees receive one copy unless otherwise specified

Dr. Thomas O. Mottl
The Analytic Sciences Corporation
Six Jacob Way
Reading, MD 01867

Naval Ocean Systems Center
Code 6212
San Diego, CA 92152

Naval Surface Weapons Center
White Oak Laboratory
Code U-20
Silver Spring, MD 20910 2 copies

Dr. Yaakov Bar-Shalom
The University of Connecticut
Department of Electrical Engineering
and Computer Science
Box U-157
Storrs, CT 06268

Mr. Conrad
Naval Intelligence Support Center
Code 20
Suitland, MD 20390

Dr. V. T. Gabriel
General Electric Company
Sonar Systems Engineering
Farrell Road Plant
Building 1, Room D6
Syracuse, NY 13201

Naval Air Development Center
Warminster, PA 18974

Naval Electronic Systems Command
Washington, DC 20360
Code 320
PME-124

Naval Research Laboratory
Washington, DC 20375
Code 2627, Code 5308, Code 7932

Naval Sea Systems Command
Washington, DC 20360
Code 63R-1, Code 63R-16

Defense Technical Information Center
Cameron Station
Alexandria, VA 22314 12 copies

Center for Naval Analyses
2000 North Beauregard Street
Alexandria, VA 22311

Office of Naval Research
800 N. Quincy Street
Arlington, VA 22217
Code 431 2 copies

Dr. Byron D. Tapley
The University of Texas at Austin
Dept. of Aerospace Engineering
and Engineering Mechanics
Austin, TX 78712

Dr. Fred W. Weidmann
TRACOR, Inc.
Tracor Sciences and Systems
6500 Tracor Lane
Austin, TX 78721

Dr. C. Carter
Naval Underwater Systems Center
New London Laboratory
Code 313
New London, CT 06320

Naval Underwater Systems Center
Code 352
Newport, RI 02840

Office of Naval Research Western
Regional Office
1030 East Green Street
Pasadena, CA 91106

Dr. R. Cavanagh
Planning Systems, Inc.
Suite 600, 7900 West Park Drive
McLean, VA 22102

Distribution List (Cont'd)

Naval Postgraduate School
Monterey, CA 93940
Technical Library
Dr. H. Titus
Dr. N. Forrest
Dr. G. Sackman

Applied Physics Laboratory
Johns Hopkins University
Johns Hopkins Road
Laurel, MD 20810

Summit Research Corporation
1 West Deer Park Avenue
Gaithersburg, MD 20760

Massachusetts Institute of Technology
Department of Ocean Engineering
Cambridge, MA 02139
Dr. Psaraftis

Dr. M. Hinich
Department of Economics
Virginia Polytechnic Institute
and State University
Blacksburg, VA 24061

Dr. T. Fortmann
Bolt, Beranek and Newman, Inc.
10 Moulton Street
Cambridge, Massachusetts 02138

Laboratory for Information and
Decision Sciences
Massachusetts Institute of Technology
Cambridge, MA 02139

DATE
ILME