HIGH-RESOLUTION METHODS
FOR ESTIMATING
OCEAN WAVE SPECTRA

FINAL REPORT

by

Lee E. Freitag, M.S.

Department of Electrical Engineering
University of Alaska, Fairbanks
Fairbanks, Alaska 99775

May 1987

Prepared for:

STATE OF ALASKA
DEPARTMENT OF TRANSPORTATION AND PUBLIC FACILITIES
DIVISION OF PLANNING AND PROGRAMMING
RESEARCH SECTION
2301 Peger Road
Fairbanks, Alaska 99701

The contents of this report reflect the views of the author who is responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Alaska Department of Transportation and Public Facilities. This report does not constitute a standard, specification or regulation.
# Contents

List of Figures ................................................................. 2  
Glossary ................................................................. 3  

1. Introduction ................................................................. 4  
   1.1 Background and Motivation ......................................... 5  
   1.2 Organization ........................................................... 7  

2. Scalar Spectral Estimation ................................................. 8  
   2.1 Fourier Transform Methods ........................................ 8  
   2.2 Autoregressive Spectral Analysis ................................ 13  
   2.3 The Maximum-Likelihood Method ................................. 22  
   2.4 Eigenvector Methods for Spectral Analysis ..................... 25  

3. Summary and Conclusions ................................................ 30  
   3.1 Algorithm Summary ................................................ 30  
   3.2 Comments .............................................................. 32  

Appendix A–Derivation of Autoregressive Analysis ..................... 33  
Appendix B–Derivation of the Maximum-Likelihood Method .......... 35  
Appendix C–Real-Time Systems ............................................ 38  
References ................................................................. 41
# Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>FFT of one segment of an AR(2) process</td>
</tr>
<tr>
<td>2.2</td>
<td>Average of seven transformed segments</td>
</tr>
<tr>
<td>2.3</td>
<td>Segmenting using the triangular window</td>
</tr>
<tr>
<td>2.4</td>
<td>A second order AR process</td>
</tr>
<tr>
<td>2.5</td>
<td>Ideal spectrum of the above process</td>
</tr>
<tr>
<td>2.6</td>
<td>Result of assuming an AR(4) process</td>
</tr>
<tr>
<td><strong>2.7</strong></td>
<td>Result of assuming an AR(6) process</td>
</tr>
<tr>
<td>2.8</td>
<td>Two AR(2) process added together</td>
</tr>
<tr>
<td>2.9</td>
<td>AR(5) fit to Figure 2.8</td>
</tr>
<tr>
<td>2.10</td>
<td>FFT of Figure 2.8</td>
</tr>
<tr>
<td>2.11</td>
<td>ML spectrum of an AR(2) process</td>
</tr>
<tr>
<td>2.12</td>
<td>Ideal spectrum for AR(2) process</td>
</tr>
<tr>
<td><strong>2.13</strong></td>
<td>Eigenvector spectrum of AR(2) process</td>
</tr>
</tbody>
</table>
Glossary

\[ \mathbf{A}, \mathbf{B} \ldots \mathbf{Z} \] Matrices and frequency-domain vectors
\[ \mathbf{a}, \mathbf{b}, \ldots \mathbf{z} \] Vectors
\[ \mathbf{a}, \mathbf{b}, \ldots \mathbf{z} \] Scalars
\[ \mathbf{I} \] Identity matrix
\[ \mathbf{A}^*, \mathbf{a}^* \] Complex conjugate transpose
\[ \mathbf{\Lambda} \] Diagonal eigenvalue matrix
\[ \lambda_i \] \(i\)th eigenvalue of a eigenvalue sequence
\[ \mathbf{E} \] Column matrix of eigenvectors (a modal matrix)
\[ \mathbf{E}_n \] Column matrix of noise subspace eigenvectors
\[ \mathbf{E}_s \] Column matrix of signal subspace eigenvectors
\[ \mathbf{e}_i \] \(i\)th eigenvector of an eigenvector matrix
\[ \mathbf{R}(m) \] Correlation sequence
\[ \mathbf{R} \] Correlation matrix
\[ \hat{\mathbf{R}} \] Estimate of the correlation matrix
\[ \sigma^2 \] Noise variance or power
\[ E[\cdot] \] Expected value
\[ \Delta t \] Sampling period
\[ \Delta f \] Frequency increment
\[ \text{ML} \] Maximum-Likelihood
\[ \text{ME} \] Maximum-Entropy
\[ \text{AR} \] Autoregressive
\[ \text{FFT} \] Fast Fourier Transform
CHAPTER 1

Introduction

Estimating the power spectrum associated with ocean waves is an important part of any engineering effort which involves the sea. Whether the data is being collected in a highly controlled wave tank or on a buoy far from shore, the techniques which may be used for analysis are the same. As the ability to collect and process wave data has advanced, so has the need for sophisticated analysis. Offshore structures, shore protection systems and general naval architecture all require information on the type and strength of the waves which will be encountered. With proper wave information designers can make better decisions and produce reliable, yet cost-effective, designs.

In this paper several types of estimators which may be used on wave data are presented and compared. The primary object is to survey modern spectral estimation techniques with respect to their possible application as wave analysis methods. Although in one sense a train of wave data is just another time series, as with any specific application there are some differences or special points which must be considered. A few assumptions will simplify the analysis and make it quite general as far as the whole field of signal processing is concerned. Special considerations which affect particular methods will be discussed as they appear. More global concerns will be mentioned in the introduction to spectral analysis which begins this chapter. After this motivational and introductory section, the organization of the paper is presented.
1.1 Background and motivation

Spectral analysis has always been a topic of interest for those involved with wave research. As new methods of time series analysis are developed, they are applied to wave data. The major advances in spectral analysis: the fast Fourier transform (FFT), the maximum-likelihood (ML) method of Capon and autoregressive (AR) or maximum-entropy (ME) analysis have all had an impact on wave studies. A more recent advance in spectral analysis which utilizes a signal subspace approach has not yet been applied to waves. It may be that this method is not entirely appropriate for determining wave spectra. Irregardless, it is important to explore the possibilities associated with any promising technique.

Because wave analysis is a relatively small subfield of signal processing, the extensive and specific literature base associated with more general signal analysis does not exist. However, over the past twenty years enough articles have appeared to give an indication of techniques being employed in wave analysis. One of the reasons for the modest selection of wave spectra articles is that most researchers are using established techniques and publish results, not methods. Exceptions are Regier and Lloyd who in 1977 published a pair of articles ([DR77] and [RD77]), one on methods and the other on results of a wave study done off the coast of Barbados. Much of the article on methods [DR77] comes from [Reg75], Regier’s doctoral dissertation. In both [DR77] and [Reg75] the concentration is on frequency-wavenumber \( (f - \kappa) \) analysis, i.e. directional spectra.

A 1979 paper from the Norwegian Institute of Technology [HH79] introduced autoregressive (AR) and maximum-entropy (ME) spectral analysis to ocean wave research. Both of these techniques (and they can be shown to be equivalent for the single-channel case) originated in geophysics, and have found wide applications in signal processing. One of the major advantages of these methods is that
data is reduced to a set of coefficients which are a parametric description of the measured process.

The maximum-likelihood method, originated by Capon [Cap69] for array \((f - \kappa)\) analysis, has gained a wide following. It is non-parametric, and though formulated for plane waves still works well when the spectrum is broad rather than purely harmonic. An application of that method to determination of wave spectra in a wave tank may be found in [JWRP81]. This article also included a brief presentation of ML theory.

There are a number of other articles, some of them not well-distributed, which are concerned with estimating wave spectra. However, the articles above, and the references which they cite, make up a substantial body of work (though it is not exhaustive by any means). In what follows we shall attempt to present some of these methods, usually working from the theoretical side of signal processing in order to produce the most general results.
1.2 Organization

The core of this paper lies in the next chapter, which contains descriptions of several analysis techniques and some comparisons between them. The emphasis is on putting them all together, and for that reason, some of the derivations have been relegated to Appendices A and B. Extensive simulations are not done, though some simple demonstrations which point out advantages or disadvantages are performed.

In Chapter 3 the relative merits of the estimators are discussed and computational aspects presented. Appendices A and B contain derivations of the AR and ML methods respectively, and Appendix C has descriptions of possible real-time processing systems, one with very long term storage and another with a self-contained communications system.
CHAPTER 2
Scalar Spectral Estimation

SPECTRAL estimation (SE) methods for the scalar or single-channel data case merit discussion for two main reasons. Firstly, they provide a basis for the more complex directional or multi-channel spectral estimation methods, and second, in some cases only a single sensor is available or necessary for the desired analysis. In this chapter several popular methods for scalar spectral estimation are presented.

Methods involving the Fourier transform are discussed first because they form an initial step in other, more complex, algorithms and because they provide a “benchmark” which may be used for the purpose of comparison. The autoregressive (AR) spectral analysis technique is then described. AR spectral analysis is important because it allows a parametric description of the process in question to be computed. In addition, it is related to the maximum-entropy (ME) method which is of much interest. The second half of this chapter is devoted to two so-called high-resolution methods. The maximum-likelihood (ML) method is presented first and shown to perform well in the narrowband and broadband signal cases. An eigenvector method based on the signal subspace representation of sinusoids in white noise is also discussed and shown to have desirable properties under specific circumstances. Finally, conclusions are drawn about the use of these various spectral estimators.

2.1 Fourier transform methods

Discrete Fourier transform (DFT) methods have been popular, as well as feasible, for real-time spectral analysis since the development of the fast Fourier
transform (FFT) algorithm. This algorithm reduces the computation load inherent to the DFT and the result is an efficient means of spectral estimation. While the Fourier transform equation may be easily written as

$$X(m) = \Delta t \sum_{n=0}^{N-1} x_n e^{-j2\pi mn/N},$$

there are several specific considerations which an analyst must attend to in its implementation.

Of primary importance is the question of resolution. The resolution in the frequency domain is directly related to the number of points used to compute the spectrum. The frequency increment $A_f$ is

$$A_f = \frac{1}{N \Delta t}$$

where $N$ is the number of points in the data set. Obviously, as the number of points used in computing the transform is increased, the frequency resolving ability of the algorithm is increased. Before arbitrarily increasing the length of the collected data records, one must consider the ramifications of doing so. One problem with extremely long Fourier transforms is computational: the processor available must be able to perform the operation in what may be a limited window of time. Second, the nature of the time series itself must be considered. It is assumed that the time series is weakly stationary, that is, its first and second moments (the mean and variance) are constant over the analysis interval. As this interval is increased the likelihood that the assumption of a stationary process will be violated is also increased. The effect may or may not be significant, though bias in the frequency estimate is usually the result.

One difficulty with the standard FFT is that it often requires smoothing before it begins to resemble the actual process being analyzed. The accepted means
of reducing the variance of this spectral estimator is to average over successive estimates. This is termed the Bartlett estimate and is given by

$$X_{BART}(m) = \frac{1}{K} \sum_{i=1}^{K} X_i(m).$$

This smoothing technique has a tendency to decrease the variability at frequency taps where noise is predominate (assuming of course that the noise is white). This averaging requires significant quantities of data, but is quite effective in reducing the variance. As illustration, note Figures 2.1 and 2.2. The first is the magnitude plot of the FFT of one segment of a data suite generated from a second order AR process. A peak at the center of this broadband process is evident at $0.15$ Hz, but in the remainder of the plot oscillations of $10$ to $15$ dB are present. By averaging over seven segments (Figure 2.2) this is significantly reduced and the resulting plot is very smooth in comparison.

The shorter the length of the data set used to compute the spectrum using the Fourier transform the more important the selection of a window with which to pass over the data before transforming. Using the rectangular window (i.e. no window) results in significant sidelobes in the frequency domain. In attempts to reduce the magnitude of the sidelobes many different windows have been developed and studied (see [Har78] for a comprehensive selection). Because the primary idea is to reduce the effects of truncation the windows basically perform a tapering in the time domain. One result of the window is a "spreading out" of the spectrum, since as relative sidelobe height is decreased the mainlobe width is increased. Increasing the mainlobe width may not be detrimental, in fact it may be a positive side effect of windowing depending on the process being analyzed, and whether averaging around adjacent frequency bins is desired. Selection of an appropriate window function should depend on the application and the length of the data sets involved.
Figure 2.1  FFT of one segment of an AR(2) process

Figure 2.2  Average of seven transformed segments
Because the use of a window makes points near the ends of a particular data segment less significant, it is also common to reuse the data by segmenting with an overlap. A 50 percent overlap will put the peak of one segment at the null point of the previous and following segments. This is illustrated in Figure 2.3 where 256 points are divided up into 64 point segments overlapped by 50 percent. This yields seven segments rather than just four and averaging over the seven allows a further reduction in variance and makes for better overall use of the data.

Although much more could be said about spectral analysis using the Fourier transform, other sources such as [OS75] and [KM81] and references cited therein provide much more information than space allows here.

Figure 2.3  Segmenting using the triangular window
2.2 Autoregressive spectral analysis [Mak75]

Autoregressive analysis assumes that the process in question may be represented by a weighted sum of previous values plus a white noise element. The object is to determine the correct values of the weighting coefficients $a_k$ which characterize the process. At any given instant the current value of the signal is given by

$$x_n = \sum_{k=1}^{p} a_k x_{n-k} + Gw_n$$  \hspace{1cm} (2.3)$$

where $p$ is the order of the process, $w_n$ is the white noise component and $G$ is a gain factor.

Of primary interest in AR spectral estimation are: determination of correct process order, stability of the resulting process and correctness of the actual spectral estimate. There are many different ways of computing process order, although sometimes a heuristic approach is as reasonable as any other. Some of these are discussed in [Hay83]. One method falls out of the computation of the coefficients $a_k$ and will be discussed in Appendix A. Stability of an AR process is guaranteed as long as its poles lie within the unit circle. Thus, any computational scheme should ensure that this occurs.

We may write the transfer function of an AR($p$) (AR process of order $p$) in the frequency domain by taking the $z$ transform of equation (2.3). The result is

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}}.$$  \hspace{1cm} (2.4)$$

The roots of the polynomial formed by the denominator of (2.4) are the poles of the process and their location in the $z$-plane characterizes the process. As the pole locations approach the unit circle, the process tends to the purely harmonic. In contrast, the poles of a white noise process will cluster at the origin. When white noise is added to a harmonic process the poles will be perturbed from the unit circle toward the origin. The detection of sinusoids in noise is a complete
topic in itself and we only mention here that when it is known that the process is completely harmonic a method such as Pisarenko Harmonic Decomposition (PHD) may be used with some advantage over general AR spectral estimation. In fact, even if a process is known not to be purely harmonic this method or a similar one may be used. The consequence is a harmonic (narrowband) fit to a non-harmonic (broadband) process. This may be sufficient if only the major frequency component of the broadband process is desired.

A derivation of the AR equations from a one-step Wiener filter is included in Appendix A. Here we simply present the system of equations which must be solved in order to determine the coefficients and the resulting AR spectrum. First recall that the estimated autocorrelation sequence of a time series may be computed from

\[ \hat{R}_{zz}(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} x_{n+m} x_n^* \]  \hspace{1cm} (2.5)

which although biased, provides less mean-square error than the unbiased estimator for \( R_{xx} \) [KM81]. Note that \( x^* \) is the conjugate of \( x \). Now form the autocorrelation matrix from the autocorrelation sequence by extending each lag term diagonally. The result is the symmetric, Toeplitz matrix

\[
\begin{bmatrix}
R_{zz}(0) & R_{zz}(-1) & \cdots & R_{zz}(-p) \\
R_{zz}(1) & R_{zz}(0) & \cdots & R_{zz}(-(p-1)) \\
\vdots & \vdots & \ddots & \vdots \\
R_{zz}(p) & R_{zz}(p-1) & \cdots & R_{zz}(0)
\end{bmatrix}
\]

where Toeplitz simply means that the entries along each diagonal are equal [GS58]. The relation between the autoregressive coefficients and the autocorrelation matrix is

\[ \hat{R}_{zz} \mathbf{a} = \sigma^2 \delta_{i0} \]  \hspace{1cm} (2.6)

where \( \delta_{i0} \) is the Kroneker delta:

\[ \delta_{i0} = \begin{cases} 
0 & \text{if } i \neq 0 \\
1 & \text{if } i = 0 
\end{cases} \]
Writing equation (2.6) in matrix form yields

\[
\begin{bmatrix}
R_{xx}(0) & R_{xx}(-1) & \cdots & R_{xx}(-p) \\
R_{xx}(1) & R_{xx}(0) & \cdots & R_{xx}(-(p-1)) \\
\vdots & \vdots & \ddots & \vdots \\
R_{xx}(p) & R_{xx}(p-1) & \cdots & R_{xx}(0)
\end{bmatrix}
\begin{bmatrix}
1 \\
ap_1 \\
pm_2 \\
p\cdots p
\end{bmatrix}
= \begin{bmatrix}
\sigma^2 \\
0 \\
\vdots \\
0
\end{bmatrix}.
\]

This system has \( p + 1 \) equations and \( p + 1 \) unknowns: the coefficients \( a_k : k = 1, \ldots, p \) and noise variance \( \sigma^2 \). These unknowns may be computed from equation (2.6) and the AR spectrum found from

\[
S_{AR}(f) = \frac{\sigma^2 \Delta t}{|1 + \sum_{k=1}^{p} a_k e^{-j2\pi f k \Delta t}|^2}.
\]

Since the equation for the Fourier transform may be written as

\[
X(\ell) = \mathbf{A}t \sum_{k=0}^{N-1} x_n e^{-j2\pi \ell k \Delta t}
\]

we immediately see that once the coefficients are found the spectrum may be computed using an FFT. Generally this involves padding the coefficient record with zeros to get the \( 2^n \) data points required by the fast Fourier transform algorithm. Zero-padding provides interpolating points between spectral lines and yields a smoother spectrum in the case where there are only a few data points. The result is a representation of the continuous spectrum.

An example of an AR process is shown in Figure 2.4. The transfer function

\[
H(z) = \frac{1}{1 - 1.097z^{-1} + 0.87z^{-2}}
\]

was used to generate the time series. In this case \( a_1 = -1.097 \) and \( a_2 = 0.87 \). An advantage of using autoregressive analysis is the resulting reduction of the data to a short series of coefficients which may be stored in a relatively small amount of computer memory. The coefficients may then be used to generate a time series with statistical properties similar to the original [HH79]. This may be useful for
simulation purposes. To generate such time series one may use equation (2.3) or equivalently,
\[ x_n = Gw_n - \sum_{k=1}^{p} a_k x_{n-k} \]
where the \( w_n \) represent white noise, and \( G \) is a gain or scaling factor.

In Figure 2.5 the ideal spectrum of the process is plotted. It was computed using the actual coefficients \( a_k \) and equation (2.7). The ideal spectrum shows a broadband process centered at 0.15 Hz. The next figure is an example of the actual use of both equations (2.6) and (2.7). The autocorrelation of the time series in Figure 2.5 was found and the coefficients found using Durbin’s recursion (see Appendix A). The result was \( a_1 = -0.93 \) and \( a_2 = 0.59 \). It is instructive to observe the location in the z-plane of one of the poles for the ideal and experimental cases. For the ideal case (in polar notation), the magnitude is 0.93 at an angle of 144 degrees (0.15 Hz). The experimental data yields 0.782 at 142.7 degrees (0.146 Hz). The center frequency location is quite accurate, but we see that the pole has been perturbed significantly from its location near the unit circle. This comes about from noise added in generating the time series, the noise has forced the poles further away from the unit circle and made the process less harmonic.

Adding noise has increased the order of the process; one way to get a better fit is to assume that a higher order is needed and re-calculate accordingly. The result of assuming a fourth order process is shown in Figure 2.6. The plot is almost identical to the ideal spectrum except for the rise near the folding frequency. This comes about from the higher order fit, the extra poles are showing their presence. When the order is increased to six, an ambiguity becomes visible near 0.4 Hz (Figure 2.7). This is one of the problems with AR spectral analysis, determining the correct order and interpreting the resulting spectrum. As an example of a higher order process observe Figure 2.8, where two AR(2) time series have been
added together. A fifth order fit yields the spectrum of Figure 2.9, very smooth compared to the Fourier transform (using a Hamming window) of Figure 2.10.
Figure 2.4 A second order AR process

Figure 2.5 Ideal spectrum of the above process
Figure 2.7 Result of assuming an AR(6) process
Figure 2.8  Two AR(2) processes added together

Figure 2.9  AR(5) fit to Figure 2.8
Figure 2.10 FFT of Figure 2.8
2.3 The maximum-likelihood method

The maximum-likelihood (ML) method was originally developed by Capon for use in processing data from a large aperture seismic array in eastern Montana [Cap69]. While it is both a scalar spectral analysis technique as well as a directional estimator we shall consider only the single-channel case in this section.

The maximum-likelihood method results from optimizing the output of a filter at a specific frequency. The general idea behind maximum-likelihood is to select a particular direction of look and minimize the variance associated with a filter which has unity response at the frequency of interest. This amounts to minimizing

$$\sigma^2 = \mathbf{a}^* \mathbf{R}_{xx} \mathbf{a}$$

subject to the constraint that

$$\mathbf{a}^* \mathbf{c} = 1$$

where \( \mathbf{a} \) is the column vector of filter coefficients and \( \mathbf{c} \) is the vector

$$\mathbf{c}(f) = \begin{bmatrix} e^{2\pi f_i \Delta t} \\ \vdots \\ e^{2\pi f_i(p-1)\Delta t} \end{bmatrix}.$$ 

For the non-complex case the vector \( \mathbf{c} \) is

$$\mathbf{c}(f) = \begin{bmatrix} \cos 2\pi f_i \Delta t \\ \vdots \\ \cos 2\pi f_i(p-1)\Delta t \end{bmatrix}.$$ 

We may use the Lagrange multiplier technique to find an expression for the maximum-likelihood spectrum (see Appendix B). The result is

$$S_{ML}(f) = \frac{1}{\mathbf{c}^*(f) \mathbf{R}^{-1} \mathbf{c}(f)}$$

where \( \mathbf{R}^{-1} \) is the inverse of the autocorrelation matrix.
While the ML method is quite good it is important to note that the problem is formulated for a plane wave situation. In an article by Davis and Regier [DR77] it is noted that this method might not be optimum for the continuous broadband spectra associated with ocean waves. Davis and Regier then propose an alternate method which uses a constraint different than \( a^c = 1 \). In essence the modified method integrates over a certain wavenumber region rather than simply working at a single wavenumber. One of the difficulties of this scheme is determining the optimum region over which to integrate. Because of the computational load a real-time method might have to simply assume a value for this parameter, while off-line analysis may utilize a search for an optimum value. The performance of this estimator (termed DASE for Data-Adaptive Spectral Estimator) is excellent, but may be too computationally intensive for some real-time implementations.

An example of an ML spectrum is shown in Figure 2.11, where the same AR transfer function used in the previous sections is used again. In comparing the ML spectrum of Figure 2.11 with the Bartlett estimate in Figure 2.2 one immediately notes that the ML spectrum is much smoother. Recall that the spectrum in Figure 2.2 was the result of smoothing over seven segments and used 512 data points. The advantage of the ML method, which required only 128 points to produce the spectrum seen in Figure 2.11, becomes obvious: it takes much less data and results in a much smoother spectral representation than the Fourier transform methods. However, the AR spectrum of the last section is significantly smoother than the ML spectrum here. This does not mean that ML is not as good as AR. Maximum-likelihood is very good with plane waves, whereas AR (as implemented above) does not always accurately represent a narrowband signal.
Figure 2.11  ML spectrum of an AR(2) process

Figure 2.12  Ideal spectrum for AR(2) process
2.4 Eigenvector methods for spectral analysis

Eigenvector methods for signal processing have attracted much interest recently. While it is not possible to do justice to the extensive theory surrounding vector space concepts and generalized eigenanalysis in this short section, it is nonetheless possible to explain briefly how these methods may be used for spectral analysis. It is assumed that the reader has some knowledge of basic linear algebra. For general background on linear algebra see any basic text such as [Leon80].

Pisarenko Harmonic Decomposition [Pisar73], which was alluded to previously, is the forerunner of the signal subspace approach as developed by Schmidt [Sch81] for array processing. Signal subspace algorithms make use of the eigenstructure of the correlation (or spectral) matrices to determine the subspaces which correspond to the signal and the noise. Knowledge of these subspaces, which are spanned by the eigenvectors of the estimated correlation matrix, allows a search to be done through all possible parameter vectors. This search is for those parameter vectors which lie in the signal subspace. The parameter vectors characterize the signals except for their associated power which may be computed by solving a series of linear equations. A brief outline of the method for spectral analysis is given below.

A discrete time model for harmonics in additive noise may be written in matrix form for n samples as

\[ \mathbf{x} = \mathbf{C}(f)\mathbf{a} + \mathbf{n} \]

where

\[ \mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}, \quad \text{and} \quad \mathbf{n} = \begin{bmatrix} n_0 \\ n_1 \\ \vdots \\ n_{n-1} \end{bmatrix}. \]
The matrix $C(f)$ is

$$
\begin{bmatrix}
\vdots & \vdots & \vdots & \vdots \\
 c(f_1) & c(f_2) & \cdots & c(f_p) \\
\vdots & \vdots & \ddots & \vdots \\
\end{bmatrix}
$$

where

$$
c(f_i) = \begin{bmatrix}
1 \\
\cos 2\pi f_i \\
\vdots \\
\cos 2\pi f_i(n-1)
\end{bmatrix}.
$$

The columns of $C(f)$ are called the parameter vectors of the system (model) because they characterize any given signal.

The autocorrelation matrix $R_{xx}$ is found from

$$
R_{xx} = E[x x^*]
$$

where $x^*$ denotes the conjugate transpose of $x$, and $E[\cdot]$ is the expected value. Taking the expected values yields

$$
R_{xx} = C(f)R_{aa}C^*(f) + R_{nn}
$$

where $R_{xx}$ is the autocorrelation matrix of signal strengths and $R_{nn}$ is the noise autocorrelation matrix. If the noise is uncorrelated with itself then only the zeroth lag of the noise autocorrelation function is greater than zero and

$$
R_{nn} = \sigma^2 I
$$

where $\sigma^2$ is the noise variance (power) and $I$ is the identity matrix of the appropriate order. Thus, in the uncorrelated noise case, the model for the autocorrelation matrix is

$$
R_{xx} = C(f)R_{aa}C^*(f) + \sigma^2 I.
$$

The autocorrelation matrix is now decomposed so that

$$
R_{xx} = \mathbf{E} \Lambda \mathbf{E}^{-1}
$$
where $\mathbf{E}$ is the matrix made up of the $n$ eigenvectors of $\mathbf{R}_{zz}$ and $\mathbf{A}$ is a diagonal matrix made up of the $n$ eigenvalues $\lambda$. We wish to examine the eigenvectors which make up $\mathbf{E}$ and determine which correspond to the signal and which to the noise.

Because the number of sinusoids is directly related to process order, this is identical to determining the order of the process as in autoregressive analysis and the techniques associated with AR order determination may be applied here. The interested reader is referred to [Hay83], and in what follows it is assumed that the process order is available. Given that the order is $p$, the $p$ largest eigenvalues indicate the $p$ “largest” eigenvectors which span the signal subspace. The remaining $n - p$ eigenvectors span what is called the noise subspace. Some basic linear algebra reveals that these two subspaces must be orthogonal. The set of all possible parameter vectors is a linearly independent set which we shall term the parameter manifold. These parameter vectors which represent the sinusoids of the process form a spanning set for the signal subspace.

At first glance it might seem that the eigenvectors determined from the auto-correlation matrix would be the same as the actual signal parameter vectors. In actuality all we may say about the two sets is that they span the same subspace. Thus a method must be found to determine the specific parameter vectors that span the signal subspace and represent the signals. To do so the orthogonality between the signal and noise subspaces is exploited. Because the inner product of two orthogonal vectors is zero, a plot of

$$S(f) = \frac{1}{c^*(f)\mathbf{E}_n\mathbf{E}_n^*c(f)}$$

will tend to infinity if a particular parameter vector is orthogonal to the noise subspace. The matrix $\mathbf{E}_n$ is the column matrix of the noise eigenvectors and the peaks of this sequence reveals the parameter vectors (and thus the frequencies) of the sinusoids.
This search through the parameter manifold may be done to whatever resolution is desired. The method becomes asymptotically exact as the length of the data sequence goes to infinity; hence the appellation “high-resolution” used in describing this estimator. Because of the sinusoidal assumption, this estimator works best for narrowband sources. When used on broadband signals the main peaks of the process will be detected depending, of course, on the order assumed in determining the dimensionality of the subspaces.

In Figure 2.13 an example of a plot generated by the signal subspace algorithm is presented. The spectrum pictured (more appropriately termed a pseudo-spectrum) is of the same AR(2) process as in the previous sections. We note that this method has successfully detected a signal at the center of this wide band process. However, it does not give an indication of bandwidth, only the location of the strongest spectral peak.

Whether or not the eigenvector method would be useful for wave analysis remains to be seen. It depends on the nature of the wave spectra: if it is relatively narrowband this estimator might work quite well, if broadband the method would provide a narrowband fit to the center frequency of the process.
Figure 2.13  Eigenvector spectrum of AR(2) process
CHAPTER 3

Summary and Conclusions

In the last chapter the various methods of analysis were discussed from a rather theoretical standpoint. But for these methods to be of use they must be somehow implemented. The implementation of an analysis system raises some practical issues which must be dealt with before any data is collected. One of these critical issues is the selection of an algorithm. This will be discussed in the first part of this concluding chapter.

In the last section the paper is briefly summarized and some ideas about the use and verification of algorithms presented. The first two appendices which follow this chapter contain derivations which were left out of the body of the paper for clarity of presentation. In the last appendix proposals for real-time processing systems are presented.

3.1 Algorithm selection

One of the difficulties in any data analysis project is selecting an algorithm or technique with which to process the data. The problem is particularly acute if the processing system must be coded into a remote instrumentation system where it is fixed for the duration of the project or the life of the instrument. It is possible however to discuss selection in terms of two specific areas.

Computational Assuming a microprocessor-based system, there is a definite limit on algorithm complexity. While the processing associated with wave analysis can generally be done in real-time because of the low sampling rate, very
sophisticated methods such as DASE and eigenanalysis may require clever coding in machine language or auxiliary processors. But because of the high costs of low-level software development, high-level languages should always be used if possible. For this reason, adding data acquisition equipment capable of Direct Memory Access (DMA) and additional processors which can work quasi-independently of the “supervisor” may actually be less expensive than writing more complex code.

Algorithmic Though speed is obviously important, the type of processing done and the form of the resulting output affects the entire project. If there is no way of preserving the original data, either by saving records if they pass some criteria which makes them worthy, or by transmitting them to a central data gathering system, then the algorithm must provide all information required by the analyst. In the case of AR spectral estimation only the coefficients and the noise variance are required to completely characterize the spectrum. For ML analysis the magnitude at selected points in the active spectrum would have to be recorded. Note that the computational burden of the ML algorithm can be eased by only computing the spectrum at frequencies of interest. For Fourier analysis this would probably not be the case since computing frequencies selectively would require that a discrete Fourier transform be done rather than a fast Fourier transform. If very few frequencies were of interest then this might be satisfactory, but usually enough points to represent a continuous spectrum are desired and it is best to use the FFT.

In using the FFT there are several more decisions to make: what type of window should be used on the data? How should the appropriate data set length (and thus resolution) be determined? Should the estimate be averaged over successive samples as in the Bartlett scheme? Some of these questions can be
answered analytically by experimenting on actual data. Others are judgement calls and in most cases experience is the best guide.

### 3.2 Comments

The intent throughout this paper has been the presentation of ideas and techniques which constitute a basic repertoire of spectral analysis methods. In this presentation we have: briefly reviewed the background literature available for both wave analysis and general spectral analysis, examined the use of four major spectral analysis methods and considered the problem of actually selecting a method.

The four estimators range from the standard Fourier transform to the rather esoteric eigenvector or signal subspace method. One of the difficulties with making a decision about which estimator to use comes about because of the lack of detailed information on these estimators as applied to wave analysis. No matter how many simulations are performed, there is always a gap in the knowledge gained through contrived trials. What is needed is a long-term study of these spectral estimators (and others not mentioned here) in actual use and under diverse conditions. With this information it will be possible to make a more informed choice of analysis method.

In conclusion it must be pointed out that none of these estimators is distinctly inferior or superior to the others. Each has advantages and disadvantages which surface under various conditions. An analyst should make an effort to explore and gain some understanding of all methods in order to derive the most from the collected data and further the advancement of wave science.
Appendix A

Autoregressive Analysis

Let us begin with the matrix form of the Wiener-Hopf equation

\[ R_{zz} h_0 = r_{dz}. \]  (A1)

Here \( R_{zz} \) is the autocorrelation matrix, \( h_0 \) is the impulse response vector or Wiener filter, and \( r_{dz} \) may be thought of as the correlation between the actual output and the desired output.

A special case of the Wiener filter is the Prediction Error Filter (also known as the one-step Wiener filter) which is designed to predict the best next value of a time series. The correlation sequence on the right hand side of (A1) then becomes \( r_{zz} \) and, if we let \( a = -h_0 \), the resulting matrix is \([KM81]\)

\[
\begin{bmatrix}
R_{zz}(0) & R_{zz}(-1) & \ldots & R_{zz}(-(p-1)) \\
R_{zz}(1) & R_{zz}(0) & \ldots & R_{zz}(-(p-2)) \\
\vdots & \vdots & \ddots & \vdots \\
R_{zz}(p-1) & R_{zz}(p-2) & \ldots & R_{zz}(0)
\end{bmatrix}
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_p
\end{bmatrix}
=
\begin{bmatrix}
R_{zz}(1) \\
R_{zz}(2) \\
\vdots \\
R_{zz}(p)
\end{bmatrix}. \]  (A2)

The error associated with this filter may be written as \([Hay83]\)

\[ \sigma^2 = \sum_{k=0}^{p} a_k R_{zz}(-k) \]

and this equation may be used to augment (A2) which then yields

\[
\begin{bmatrix}
R_{zz}(0) & R_{zz}(-1) & \ldots & R_{zz}(-(p-1)) \\
R_{zz}(1) & R_{zz}(0) & \ldots & R_{zz}(-(p-1)) \\
\vdots & \vdots & \ddots & \vdots \\
R_{zz}(p) & R_{zz}(p-1) & \ldots & R_{zz}(0)
\end{bmatrix}
\begin{bmatrix}
1 \\
a_1 \\
\vdots \\
a_p
\end{bmatrix}
=
\begin{bmatrix}
\sigma^2 \\
0 \\
\vdots \\
0
\end{bmatrix}. \]  (A3)
A more compact version of equation (A3) is

$$R_{zz}a = \sigma^2 \delta_{i0}$$

(A4)

where $\delta_{i0}$ is the Kroneker delta ($\delta_{i,j} = 1$ if $i = j$ and zero otherwise).

Traditional AR spectral estimation involves solving equation (A4) for the vector $a$ and the variance $\sigma^2$ which may then be used to compute the spectrum of the process using

$$S_{AR}(f) = \frac{\sigma^2 \Delta t}{1 + \sum_{k=1}^{p} a_k e^{-j2\pi fk\Delta t}}.$$  \hspace{1cm} (A5)

Because of the structure of equation (A4) it is possible to solve the system recursively for successively higher order autoregressive fits to the process. Durbin's recursion (which is about twice as fast as a similar method due to Levinson) is given by the following set of equations [Mak75].

Initialize: $\sigma_0^2 = R(0)$

Recursion:

$$k_i = - \left[ R(i) + \sum_{j=1}^{i-1} a_{j(i-1)}R(i-j) \right] / \sigma_{i-1}^2$$

$$a_i^{(i)} = k_i$$

$$a_j^{(j)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}; \hspace{0.5cm} 1 \leq j \leq i - 1$$

$$\sigma_i^2 = (1 - k_i^2)\sigma_{i-1}^2.$$  \hspace{1cm} (A6)

In solving this set of equations each step of the recursion yields a new set of coefficients and a new value for the variance $\sigma^2$. An indication of order may be obtained by observing how $\sigma^2$ changes. In general, as the variance approaches a minimum or levels off, the order is equal to or greater than the actual order of the system. Thus it is possible to obtain a best AR fit by tracking $\sigma^2$ and terminating the regression when its rate of decrease levels off.
Appendix B

The Maximum-Likelihood Method

The derivation of the ML method is a simple constrained optimization problem which may be developed in either the frequency or time domain. Here we consider the problem of determining the spectra from an estimate of the auto-correlation matrix $R_{xx}$. 

The variance or power associated with a given filter vector $a$ may be written as

$$\sigma^2 = a^* R_{xx} a$$

where $a$ is the column vector

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}.$$ 

We wish to minimize $\sigma^2$ subject to the constraint that $a^* c = 1$ where $c$ is the column vector

$$c = \begin{bmatrix} \frac{1}{e^{2\pi f_i \Delta t}} \\ \vdots \\ e^{2\pi f_i (p-1) \Delta t} \end{bmatrix}.$$ 

This constraint ensures unity frequency response at the frequency of interest $f_i$. Note that in the real-data, time domain case the vector $c$ effectively reduces to

$$c = \begin{bmatrix} \cos 2\pi f_i \Delta t \\ \vdots \\ \cos 2\pi f_i (p - 1) \Delta t \end{bmatrix}.$$ 

where $c$’s dependence on the frequency $f_i$ is implicit. Let us form the Lagrangian

$$L = a^* R_{xx} a + \alpha (1 - a^* c)$$

35
where \( a \) is the Lagrange multiplier. Differentiating with respect to \( a \) and equating the result to zero gives

\[
2R_{zz} a - ac = 0.
\]

Solving for \( a \) yields

\[
a = \frac{2}{2} R_{zz}^{-1} c. \tag{B1}
\]

Premultiply both sides by \( c^* \) and note that \( c^*a = a^*c = 1 \) so

\[
a c^* = 1 = \frac{2}{2} c^* R_{zz}^{-1} c.
\]

Solving for \( a \) gives

\[
a = \frac{R_{zz}^{-1} c}{c^* R_{zz}^{-1} c}.
\]

which, when substituted into equation (B1), will yield

\[
a = \frac{R_{zz}^{-1} c}{c^* R_{zz}^{-1} c}. \tag{B2}
\]

Observe that this optimum filter equation is simply \( R_{zz}^{-1} c \) multiplied by the scalar gain factor \( [c^* R_{zz}^{-1} c]^{-1} \). These gain factors throughout the sample spectrum \( f_i \) are, in effect, a measure of the relative power spectral density. This is easily seen by simple manipulation of equation (B2):

\[
a = \frac{R_{zz}^{-1} c}{c^* R_{zz}^{-1} c},
\]

\[
R_{zz} a = \frac{c}{c^* R_{zz}^{-1} c}
\]

\[
a^* R_{zz} a = \frac{1}{c^* R_{zz}^{-1} c}.
\]

But \( a^* R_{zz} a = \sigma^2 \) so that

\[
\sigma^2 = \frac{1}{c^* R_{zz}^{-1} c}.
\]

To emphasize the dependence of \( c \) on frequency we may write

\[
\sigma^2(f) = \frac{1}{c^*(f) R_{zz}^{-1} c(f)}.
\]
This expression then gives the power at a given frequency. It is more common to write $\sigma^2(f)$ as $S(f)$ so that the resulting equation for the ML spectrum is

$$S_{\text{ML}}(f) = \frac{1}{c^*(f)R_{zz}^{-1}c(f)}.$$  \hfill (B3)
Appendix C
Real-Time Systems

The advent of fast, low-power microcomputers has brought about a revolution in instrumentation systems. However, this increase in available computing power has been followed by a tendency toward more and more sophisticated analysis techniques. While the computational load associated with the maximum-likelihood method is well within the capability of a 16-bit microprocessor assisted by a numeric co-processor (such Intel's 8087), eigenanalysis and the DASE algorithm of Regier and Davis may exceed the processing capacity of a personal computer used in real-time analysis.

One solution to this shortfall in processing power involves adding additional specialized processors to take over some of the "number-crunching". Currently add-on boards for an IBM-PC or compatible are available which contain specialized microprocessors developed especially for high-speed signal processing. One such board is based on Texas Instrument's TMS32010 and is capable of performing a 64-point fast Fourier transform in several milliseconds. Signal processing chips such as the TMS32010 generally have separate program memory which contains code downloaded from the host and written specifically for a particular processing task. While the cost of developing such a system might be high if the specific code required for the co-processor was not already commercially available, it might be the only way of actually implementing a real-time system.

If the goal is only to collect the data and not actually process it, then it may be possible in the near future to install systems which can collect time domain data over long periods. Optical storage devices which can handle over 200 megabytes
have recently been announced, and such mass storage could provide the basis for an instrumentation system that could collect raw data for over a year and store it for later retrieval. Sampling at 2 Hz continuously for one year would accumulate 63 megabytes (assuming eight bit data). While much of the data might not be of much use at a later date, continuous monitoring would ensure that unusual, once-a-year type sea-states would be recorded and could be examined as desired.

A new mobile telephone system based on portable microwave stations for satellite communication might obviate the need for massive storage at the instrumentation site and provide immediate access to data collected in areas out of the range of normal communications. Though this system will not be available for some years, it might provide a cost-effective way of monitoring remote locations which might otherwise be impossible to instrument. This type of system would have to be “intelligent” because it would be quite expensive to transmit raw data. The data would have to be reduced to a compact form, and ideally, would only be sent when the sea-state changed in some fashion. Given that the quality of the circuit was good, off-the-shelf 1200 or 2400 baud modems could be used and would not be difficult to interface with the computer and the portable telephone system.

A system using this or some other type of communications link from a remote site to a central location could be used to provide a “dial-up” service for those who require information about sea conditions and have access to a telephone. A microcomputer system could be implemented which receives the data from the instrumentation system, analyzes it and then synthesizes a message based on that information. The system could even be interactive, not requesting information from the remote site until someone called, and then providing that information to subsequent callers for a specified period of time. This would limit toll charges for an under-utilized mobile satellite system. In general however, a remote system
which provided information to the central location on a transmit-on-delta basis would probably be easiest to set up. This would also provide a history of the sea conditions which would be compact and relatively easy to store for a permanent record. Information such as this gathered over several years might be useful in predicting sea-states at locations where previously no information was available.

In addition, microcomputer based systems can be programmed to collect and store time series data for unusual conditions which are of interest to the analyst. Certain combinations of winds, waves and currents might lead to unusual effects which a researcher might wish to study in detail using several different analysis techniques. The data could be stored on-board, or transmitted to the monitoring site for storage.

It should be noted that while the hardware for most of these types of systems is readily available, the software which would run such a dial-up monitoring system would be non-trivial to write. A program which was powerful enough to be easily modified and provided the capability for some interactive use could take months to write and test. Development costs such as this should be taken seriously when sophisticated monitoring systems are proposed.
References


