N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

Reports of the Department of Geodetic Science

Report No. 297

SPECTRAL ANALYSIS OF GEOS-3 ALTIMETER DATA AND FREQUENCY DOMAIN COLLOCATION

by

Kamil Eren

The Ohio State University
Department of Geodetic Science
1958 Neil Avenue
Columbus, Ohio 43210

February, 1980

Abstract

Spectral Analysis in Geodetic Science and a method for the solution of least-squares collocation in the frequency domain as well as a fast Toeplitz inversion algorithm in the space domain are outlined.

The mathematical background in spectral analysis as applied to geodetic applications is summarized. In geodetic problems, we usually have discrete and non-periodic data. The resolution (cut-off frequency) of the GEOS-3 altimeter data has been examined by determining the shortest wavelength (corresponding to the cut-off frequency) recoverable. The data from some 18 profiles were used in this study. The total power (variance) in the sea surface topography with respect to the reference ellipsoid as well as with respect to the GEM-9 surface was computed.

A fast inversion algorithm for matrices of simple and block Toeplitz matrices and its application to least-squares collocation is explained. This algorithm yields a considerable gain in computer time and storage in comparison with conventional least-squares collocation.

Frequency domain least-squares collocation techniques are also introduced and applied to estimating gravity anomalies from GEOS-3 altimeter data. These techniques substantially reduce the computer time and requirements in storage associated with the conventional least-squares collocation. Numerical examples given in this paper demonstrate the efficiency and speed of these techniques. The number of numerical operations required to calculate the signals is proportional to $N \log_2 N$ (where N is the number of observations) rather than N^2 with the fast Toeplitz inversion algorithm or N^3 with classical collocation.

Foreword

Table of Contents

1. Spectral Analysis and Applications

1.1 Introduction

Spectral analysis can be defined as the process of calculating and interpreting a spectrum. It actually yields a deeper understanding of data as well as the systems which produced those data. The latter property has been very useful in geophysical applications.

Spectral analysis is a good tool for the analysis of time or space series. This approach is commonly known as "the frequency domain method". While using spectral methods one always has to keep in mind that the frequency domain approach is not the only way to study time series. Time series can also be studied using the time domain approach by parametric models such as autoregressive (AR) or autoregressive moving average (ARMA) models etc. But the frequency domain approach has great advantages over the time domain approach as we will explain in this chapter.

The transformation of a record to the frequency domain is termed its "spectrum". The frequency and/or wavenumber, which is the reciprocal of frequency, is in many respects a more significant and more useful variable to use than the time (or space) functions. It is to be noted that transforming a data vector, say $x(t)$, into the frequency domain does not mean the addition of anything new but only rearrangement of the given data according to frequency (or wavenumber) instead of according to time sequence. The advantage of representation in the frequency domain comes from the fact that most geophysical phenomena are expressed theoretically in frequency dependent form. The independent variable frequency or wavenumber provides a reliable and unique check on the data in the time domain and on the systems which produce those data. In the frequency domain, comparisons of different records are referred to the same value of a certain parameter. Spectral analysis utilizes the whole signal, so no information is left out. For details see Bath (1974).

1.2 Application Fields of Spectral Analysis

1.2.1 Prediction and Interpolation

As it is very well known, time series analysis is mainly concerned with the study of the time (or space) variations of physical processes. When the state of the process is represented by measurements with one or more components at the time point, then we can represent the variations of the process over time by a vector of real-valued functions. These functions can be expressed as a sum of an infinite number of sinusoidal terms, i.e. Fourier series. This is nothing else but fitting to the data a trigonometric polynomial in the least-squares sense so that the value of the function can be predicted everywhere along the profile or surface.

1.2.2 Analysis of Sample Data

We reduce a complicated function to a series of simple trigonometric functions, in other words to its fundamental wave functions which provide a different and rather revealing information for the sample sequence.

A time function $x(t)$ can be characterized statistically when the mean, variance, and covariances between the values of $x(t)$ at different times are given. But in the analysis of a finite length of record the spectrum, which is the Fourier transform of the autocovariance function as shown later, is often preferable to the autocovariance function mainly for the following reasons:

a) Estimates of the spectrum at neighboring frequencies are generally uncorrelated. Thus we can interpret the sample spectrum easier than the sample autocovariance function.

b) As it was mentioned before, in many physical problems the spectrum is of direct physical interest to us.

1.2.3 Aid to Computations

We can use spectral analysis as an aid to computations, because some calculations can be performed more easily in the spectral domain than in the time (or space) domain. The frequency-domain representations are often simpler to handle computationally. This feature will be applied in Chapter 4 for the recovery of gravity anomalies from geoid heights which can be called the solution by frequency domain least-squares collocation.

1.2.4 Filtering and Control

By spectral methods we can compute the wave components of a time (or space) series and remove some unwanted components from the estimates. Filtering represents a weighted average in the time domain and a single multiplication in the frequency domain. So filtering is easier to apply in the frequency domain than in the time domain.

1.2.5 Differentiation and Integration

By means of differentiation the relative distribution of power can be emphasized in the high frequency (Mayhan, 1978, p. 5-27), for example:

$$x(t) = \sum_{n=-\infty}^{\infty} X_n \cdot e^{i n \omega_0 t}$$

-2-

$$x'(t) = \frac{d\,x(t)}{dt} = \sum_{n=-\infty}^{\infty} (i\,n\,\omega_0)\,X_n\,e^{i\,n\,\omega_0 t}$$

$$x''(t) = \frac{d^2 x(t)}{dt^2} = \sum_{n=-\infty}^{\infty} (i\,n\,\omega_0)^2\,X_n\,e^{i\,n\,\omega_0 t}$$

where X_n are complex Fourier coefficients, $\omega_0 = 2\pi/T$ is the fundamental angular frequency, and $i = \sqrt{-1}$. Here we can see that $x''(t)$ would have more high-frequency content than $x(t)$ because of the appearance of n^2.

If we are interested in deemphasizing the high-frequency content of a signal, then we can use integration, for example:

$$x_1(t) = \int x(t)\,dt = \sum_{n=-\infty}^{\infty} \left(\frac{1}{i\,n\,\omega_0}\right) X_n\,e^{i\,n\,\omega_0 t}$$

$$x_2(t) = \int x_1(t)\,dt = \sum_{n=-\infty}^{\infty} \left(\frac{1}{i\,n\,\omega_0}\right)^2 X_n\,e^{i\,n\,\omega_0 t}$$

provided the average (mean) value is zero. In this case $x_2(t)$ would have less high-frequency content than $x(t)$.

In addition to the applications above, the frequency domain methods have been successively applied for the estimation of transfer functions for simulation and optimization of the data vector, for generating new physical theories, for pattern recognition, for studying periodic solutions to physical problems described by differential equations, for approximating non-periodic functions, as an operational device for solving differential equations, etc.

1.3   Fourier Series and Fourier Transforms

1.3.1 The Standard Form of Fourier Series

In order to represent a data vector in the frequency domain we shall study Fourier series and Fourier transforms first. According to the Fourier theorem (Bath, 1974 , p. 26) a periodic function $x(t)$ having a fundamental period T and satisfying the Dirichlet's conditions can be represented by an infinite Fourier series

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \tag{1.1}$$

where $\omega_0 = 2\pi/T$ is the fundamental angular frequency, a_n and b_n are Fourier coefficients. The derivation of eqn. (1.1) is given in Appendix 1. The Dirichlet's conditions can be summarized as follows:
- $x(t)$ is at least sectionally continuous with finite jumps
- $x(t)$ possesses a finite number of maxima and minima
- The integral $\int_T x(t)\,dt$ should be convergent.

As an example a typical periodic function is shown in Figure 1.1 below.
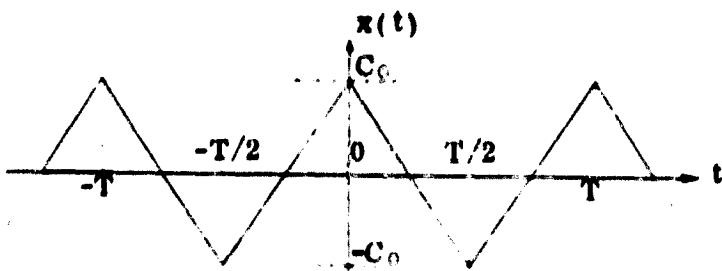
Figure 1.1. A Periodic Function.

We can determine the Fourier coefficients of eqn. (1.1) through the following steps:

- To find the coefficients a_n multiply eqn. (1.1) by $\cos n\omega_0 t$, $n = 0, 1, 2, \ldots$ from both sides and integrate over the period T with respect to t.
- To find the coefficients b_n multiply eqn. (1.1) by $\sin n\omega_0 t$, $n = 1, 2, \ldots$ from both sides and integrate over T with respect to t.

In the integration procedure above the following orthogonality relations have been used:

$$\int_T \sin m\omega_0 t \, \sin n\omega_0 t \, dt = \begin{cases} T/2 & \text{for } m = n \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{1.2.a}$$

$$\int_T \cos m\omega_0 t \, \cos n\omega_0 t \, dt = \begin{cases} T/2 & \text{for } m = n \neq 0 \\ T & \text{for } m = n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{1.2.b}$$

$$\int_T \sin m\omega_0 t \, \cos n\omega_0 t \, dt = 0 \quad \text{for all } m \text{ and } n \tag{1.2.c}$$

Following the steps above, we have for the Fourier coefficients of eqn. (1.1)

$$a_n = \frac{2}{T} \int_T x(t) \cos n\omega_0 t \, dt, \qquad n = 0, 1, 2, \ldots \tag{1.3.a}$$

$$b_n = \frac{2}{T} \int_T x(t) \sin n\omega_0 t \, dt, \qquad n = 1, 2, \ldots \tag{1.3.b}$$

As can be seen from the expressions above: a_n is the cosine transform of $x(t)$ and b_n is the sine transform of $x(t)$. It is common to call a_n "co-spectrum" and b_n "quad-spectrum" (Bath, 1974, p. 40).

1.3.2 The Complex Form of the Fourier Series

We can also express the Fourier series in complex form by substituting the following identities

$$\cos n\omega_0 t = (e^{i n \omega_0 t} + e^{-i n \omega_0 t})/2$$

$$\sin n\omega_0 t = (e^{i n \omega_0 t} - e^{-i n \omega_0 t})/(2i)$$ (1.4)

in eqn. (1.1) to obtain

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \frac{1}{2}(a_n - ib_n)e^{i n \omega_0 t} + \sum_{n=-\infty}^{-1} \frac{1}{2}(a_n - ib_n)e^{i n \omega_0 t}$$ (1.5)

$$x(t) = \sum_{n=-\infty}^{\infty} X_n e^{i n \omega_0 t}$$ (1.6)

where, $\quad X_n = \frac{1}{2}(a_n - ib_n), \quad n = \ldots, -1, 0, 1, \ldots,$ (1.7)

are the complex Fourier coefficients. This is the complex form of the Fourier series. If we substitute eqn. (1.3.a-b) in eqn. (1.7), we have

$$X_n = \frac{1}{T}\int_T x(t)\, e^{-i n \omega_0 t}\, dt$$ (1.8)

The eqn. (1.8) is called the "frequency domain representation of $x(t)$". The resulting complex Fourier coefficients are orthogonal (Papoulis, 1965) such that

$$E\{X_n\, X_m\} = \begin{cases} 0 & \text{for } n \neq m \\ \alpha_n & \text{for } n = m \end{cases}$$ (1.9)

where E denotes the expectation over T, and $E(X_0) = E[x(t)]$, $E(X_n) = 0$ for $n \neq 0$. We also define a quantity α_n as follows:

$$\alpha_n = \frac{1}{T}\int_T E\{x(t)\, x(t+\tau)\}\, e^{-i n \omega_0 \tau}\, d\tau = \frac{1}{T}\int_T C(\tau)e^{-i n \omega_0 \tau}\, d\tau$$ (1.10)

Equation (1.9) above can be proven as follows:

a) $\quad E\{X_n\} = \frac{1}{T}\int_T E\{x(t)\}\, e^{-i n \omega_0 t}\, dt$ (1.11)

since $E[x(t)]$ is a constant this expression is zero for $n \neq 0$

b) $\quad E\{X_n^* x(t)\} = \frac{1}{T}\int_T E\{x(t)x^*(u)\}e^{i n \omega_0 t}\, du = \frac{1}{T}\int_T C(t-u)e^{i n \omega_0 u}\, du$ (1.12)

where $C(t-u)$ is the autocovariance of the function $x(t)$ and the superscript $(*)$ denotes the complex conjugate. If $x(t)$ is periodic in the mean square sense, which will be explained in Section 1.3.5, then $C(\tau)$ is also periodic, i.e. $C(\tau) = C(\tau + T)$. Then it follows

$$E\{X_n^* \cdot x(t)\} = \alpha_n\, e^{i n \omega_0 t}$$ (1.13)

Now we can prove that

$$E\{X_k\, X_n^*\} = \frac{1}{T}\int_T E\{X_n^* \cdot x(u)\}\, e^{-i k \omega_0 u}\, du$$

$$= \frac{\alpha_n}{T}\int_T e^{-i(k-n)\omega_0 u}\, du = \begin{cases} 0 & \text{for } n \neq k \\ \alpha_n & \text{for } n = k \end{cases}$$ (1.14)

-5-

Using a similar approach we can also show the complex form of the Fourier series

$$E\left\{\left|x(t) - \sum_{n=-\infty}^{\infty} X_n \, e^{i n \omega_0 t}\right|^2\right\} = 0 \tag{1.15}$$

Let us take the square of the homogeneous equation above in order to see the cancellations of its terms.

$$E\{x(t)\,x(t)\} - \sum_{n=-\infty}^{\infty} E\{X_n^* \, x(t)\} \, e^{-i n \omega_0 t} - \sum_{n=-\infty}^{\infty} E\{X_n \, x^*(t)\} e^{i n \omega_0 t} + \sum_{k=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} E\{X_n \, X_k^*\} \, e^{i(k-n)\omega_0 t}$$

$$= C(0) - \sum_{n=-\infty}^{\infty} \alpha_n \, e^{i n \omega_0 t} \, e^{-i n \omega_0 t} - \sum_{n=-\infty}^{\infty} \alpha_n^* \, e^{-i n \omega_0 t} \, e^{i n \omega_0 t} + \sum_{n=-\infty}^{\infty} \alpha_n$$

$$= C(0) - \sum_{n=-\infty}^{\infty} \alpha_n^* = 0 \tag{1.16}$$

The last equality follows from eqn. (1.30) in section 1.3.5.

Eqn. (1.8) can also be derived directly from eqn. (1.6) via the orthogonality relationships for the exponential function defined as

$$\int_T e^{i m \omega_0 t} \, e^{-i n \omega_0 t} \, dt = \begin{cases} T & \text{for } n = m \\ 0 & \text{otherwise} \end{cases} \tag{1.17}$$

Multiplying both sides of eqn. (1.6) by $e^{-i n \omega_0 t}$ and integrating on $(0, T)$ we have

$$\int_T x(t) \, e^{-i n \omega_0 t} \, dt = \int_T \sum_{m=-\infty}^{\infty} X_m e^{i m \omega_0 t} \, e^{-i n \omega_0 t} \, dt$$

Interchange the order of summation and integration

$$\int_T x(t) \, e^{-i n \omega_0 t} \, dt = \sum_{m=-\infty}^{\infty} X_m \int_T e^{i m \omega_0 t} \, e^{-i n \omega_0 t} \, dt = T \, X_n \tag{1.18}$$

and

$$X_n = \frac{1}{T} \int_T x(t) \, e^{-i n \omega_0 t} \, dt, \quad n = -\infty, \ldots, 0, \ldots, \infty \tag{1.19}$$

There is a unique one-to-one correspondence between $x(t)$ and X_n called a Fourier transform pair, usually denoted by

$$x(t) \longleftrightarrow X_n$$

So for a given $x(t)$, there is one and only one set of X_n, and vice versa. The set of X_n corresponding to $x(t)$ is called the "spectrum" of $x(t)$.

Since X_n is complex one needs two graphs to display X_n. One is the graph of the $|X_n|$ versus n, or frequency which is called the amplitude spectrum and the other is the graph of the angle of X_n versus n or frequency which is called the phase spectrum. The expressions for amplitude $|X_n|$ and phase angle φ_n are:

$$|X_n| = \tfrac{1}{2} (a_n^2 + b_n^2)^{\frac{1}{2}} \tag{1.20}$$

$$\varphi_n = -\tan^{-1}(b_n/a_n) = \text{Arg } X_n \tag{1.21}$$

-6-

From these relations we can write

$$X_n = |X_n| e^{i\varphi_n} \tag{1.22}$$

As readers already have noticed, so far (for example from eqn. (1.1) or (1.6)) the spectrum of a periodic function $x(t)$ is a discrete distribution made up of a finite number of frequencies when observational data are used.

1.3.3 The Cosinewave Form of the Fourier Series

Still another form for eqn. (1.1) can be derived using the identity

$$a_n \cos n\omega_0 t + b_n \sin n\omega_0 t = A_n \cos (n\omega_0 t + \varphi_n)$$

In eqn. (1.1) to obtain

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} A_n \cos (n\omega_0 t + \varphi_n) \tag{1.23}$$

where $A_n = (a_n^2 + b_n^2)^{\frac{1}{2}}$ is the amplitude of the cosine wave and $\varphi_n = -\tan^{-1}(b_n/a_n)$ is the phase angle of the wave.

1.3.4 Average Power and RMS Value

There are some cases where we are only interested in the average power contained in any frequency component which comprises $x(t)$. So here we speak about the power of $x(t)$. Average power for a certain wavenumber or frequency is defined as $2|X_n|^2$ and plotted against wavenumber or frequency. This function is an even function of n, and contains no phase information.

The total average power is expressed as

$$P_{AVG}^2 = \frac{1}{T} \int_T x^2(t) \, dt \tag{1.24}$$

and the RMS value of $x(t)$ is defined to be

$$[x(t)]_{RMS} = P_{AVG} = [\frac{1}{T} \int_T x^2(t) \, dt]^{\frac{1}{2}} \tag{1.25}$$

By Parseval's theorem (the proof is given in Appendix 1.A) the eqn. (1.24) can be written

$$P_{AVG}^2 = \left(\frac{a_0}{2}\right)^2 + \sum_{n=1}^{\infty} (a_n^2 + b_n^2)/2 \tag{1.26}$$

or identically in terms of amplitudes of cosinewaves or by complex Fourier coefficients

$$P_{AVG}^2 = \left(\frac{a_0}{2}\right)^2 + \frac{1}{2} \sum_{n=1}^{\infty} A_n^2 \tag{1.27}$$

-7-

$$P_{AVG}^2 = X_0^2 + 2\sum_{n=1}^{\infty} |X_n|^2 = \sum_{n=-\infty}^{\infty} |X_n|^2 \tag{1.28}$$

In some reference textbooks eqn. (1.26) is expressed in terms of average degree powers

$$P_{AVG}^2 = \sum_{n=0}^{\infty} P_n^2 \tag{1.29}$$

where $P_n^2 = (a_n^2 + b_n^2)/2 = A_n^2/2 = 2|X_n|^2$, $n = 1, 2, \ldots$, and $P_0^2 = (a_0/2)^2$.

1.3.5 Mean Square Periodicity

A function $x(t)$ is periodic in the mean square sense if the autocovariance $C(\tau)$ is periodic (Papoulis, 1965), p. 367), i.e. if $C(\tau) = C(\tau + T)$ for every τ then we can transform $C(\tau)$ to the frequency domain;

$$C(\tau) = \sum_{n=-\infty}^{\infty} \alpha_n \, e^{i n \omega_0 \tau} \tag{1.30}$$

where α_n are the complex Fourier coefficients of $C(\tau)$ and defined as

$$\alpha_n = \frac{1}{T} \int_T C(\tau) \, e^{-i n \omega_0 \tau} d\tau \tag{1.31}$$

The power spectrum (also called spectral density) of a function $x(t)$ is the Fourier transform of its autocovariance function given as

$$S(\omega) = \int_{-\infty}^{\infty} C(\tau) e^{-i \omega \tau} \, d\tau \tag{1.32.a}$$

and the inverse Fourier transform can be written

$$C(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) \, e^{i \omega \tau} \, d\tau \tag{1.32.b}$$

If we substitute eqn. (1.30) in (1.32.a) we obtain

$$S(\omega) = \sum_{n=-\infty}^{\infty} \alpha_n \int_{-\infty}^{\infty} e^{i n \omega_0 \tau} \, e^{-i \omega \tau} d\tau = \sum_{n=-\infty}^{\infty} \alpha_n \, z(\omega_n) \tag{1.33}$$

where $z(\omega_n) = \int_{-\infty}^{\infty} e^{i n \omega_0 \tau} e^{-i \omega \tau} d\tau$ is the Fourier transform of $z(t) = \exp(i n \omega_0 \tau)$.

If we combine the exponential terms of $z(\omega_n)$ then we get

$$z(\omega_n) = \int_{-\infty}^{\infty} 1 \, e^{-i(\omega - n \omega_0)\tau} \, d\tau \tag{1.34.a}$$

This is the Fourier transform of $z(t) = 1$ with the angular frequency $(\omega - n\omega_0)$. Now denote $\lambda = \omega - n\omega_0$ in order to write

$$z(\lambda) = \int_{-\infty}^{\infty} 1 \, e^{-i \lambda \tau} \, d\tau \tag{1.34.b}$$

The equation above is the classical Fourier transform of $z(t) = 1$ with angular

frequency λ. To evaluate the integral above we will show that

$$1 \longleftrightarrow 2\pi \cdot \delta(\omega)$$

i.e. $2\pi \cdot \delta(\omega)$ and 1 are the Fourier transform pairs, where $\delta(\omega)$ is the F Dirac delta function (unit area impulse) defined as follows:

$$\delta(t) = 0 \text{ for } t \neq 0, \quad \int_{-\infty}^{\infty} \delta(t) \, dt = 1, \text{ and finally, } \int_{-\infty}^{\infty} \delta(t) \, x(t) = x(0).$$

For the proof, let us compute the inverse Fourier transform of $2\pi \cdot \delta(\omega)$.

$$x(t) = 1 = \frac{1}{2\pi} \int_{-\infty}^{\infty} \cdots \, e^{i\omega t} d\omega$$

$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} [2\pi \, \delta(\omega)] \, e^{i\omega t} \, d\omega = 1 \qquad (1.34.c)$$

Hence, eqn. (1.34.b) can be written as

$$1 \longleftrightarrow z(\lambda) = 2\pi \cdot \delta(\lambda) = 2\pi \cdot \delta(\omega - n\omega_0)$$

So we have proved that

$$z(\omega_n) = \int_{-\infty}^{\infty} e^{in\omega_0 \tau} \, e^{-in\omega\tau} \, d\tau = 2\pi \, \delta(\omega - n\omega_0) \qquad (1.35.a)$$

If we substitute this equation in (1.33) we obtain

$$S(\omega) = 2\pi \sum_{n=-\infty}^{\infty} \alpha_n \, \delta(\omega - n\omega_0) \qquad (1.35.b)$$

Eqn. (1.35.b) shows that the spectral density $S(\omega)$ of $x(t)$ is a sequence of equidistant impulses.

1.4 Finite Fourier Representations of Arbitrary Functions and Windows

By a Fourier series we can approximately represent an arbitrary function $x(t)$ as a harmonic polynomial of finite $x_N(t)$ with finite degree N. We have to truncate a Fourier series at some finite degree, because in practice we cannot carry the series expansion to infinity. But how large need N be chosen to have a reasonable approximation to $x(t)$? In this section we will attempt to answer such a question. We will also examine whether

$$\lim_{N \to \infty} x_N(t) = x(t)$$

for each t in the interval $(0, T)$.

Finite representations of arbitrary functions lead us to the concept of a "window", which is described as follows; If $x(t)$ and $y(t)$ are periodic

functions with period T such that

$$y(t) = \int_T W(t-\tau) x(\tau) d\tau$$

then $W(t)$ is called the window through which $y(t)$ views $x(t)$. Some important properties, which a window should possess, can be stated as follows:

1. A window $W(t)$ should be an even function to give equal weight to the values of $x(\cdot)$ about a given point.

$$W(t) = W(-t)$$

2. $W(t)$ should integrate to 1.

$$\int_T W(t) \, dt = 1$$

so that if $x(t) = c$ a constant, then we would have $y(t) = c$, the same constant.

3. $W(t)$ should have its maximum value at $t = 0$.

$$|W(t)| \leq W(0) \quad \text{for all } t$$

4. $W(t)$ should concentrate as much as possible around the origin, say $t = 0$. In order to clarify the concept of concentration, let us consider the area, A, defined as

$$A = \int_{-T/2}^{T/2} W(t) \, dt, \quad \text{and} \quad A_1 = \int_{-\Delta T}^{\Delta T} W(t) \, dt, \quad \Delta T < T/2,$$

such that A_1 should be as close to A as possible. Thus $y(t)$ should reflect the behavior of $x(t)$ in the neighborhood of t.

1.4.1 Dirichlet and Fejer Kernels

Let us define the discrete Fourier representation of $x_N(t)$ as

$$x_N(t) = \frac{a_0}{2} + \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t), \quad \omega_0 = 2\pi/T \tag{1.36}$$

and substitute eqns. (1.3.a-b) in the above equation and interchange the order of summation and integration to obtain

$$x_N(t) = \frac{2}{T} \int_0^T \left\{ \sum_{n=1}^{N} x(\lambda) [\cos n\omega_0\lambda \cos n\omega_0 t + \sin n\omega_0\lambda \sin n\omega_0 t] \right\} d\lambda + \frac{1}{T} \int_0^T x(\lambda) d\lambda \tag{1.37}$$

which can be shortened to

$$x_N(t) = \frac{2}{T} \int_0^T [0.5 + \cos \omega_0(\lambda - t) + \dots + \cos N\omega_0(\lambda - t)] x(\lambda) \, d\lambda \tag{1.38}$$

or as given by many authors

$$x_N(t) = \frac{1}{T} \int_0^T D_N[\omega_0(\lambda - t)] \, x(\lambda) \, d\lambda \tag{1.39}$$

-10-

where $\qquad D_N(z) = 2(0.5 + \cos z + \cos 2z + \ldots + \cos Nz), \quad z = \omega_0 t$ $\qquad$ (1.40)

Eqn. (1.40) is identical to

$$D_N(z) = \sin(N + 0.5)\, z/\sin(0.5\, z) \qquad (1.41)$$

The eqn. (1.41) can be shown by multiplying both sides of it by $(\sin z/2)$ and then substituting eqn. (1.40) for $D_N(z)$. This function is even and periodic with period 2π. The function $D_N(z)$ is called "Dirichlet's kernel" having the following roots and maximum value

$$D_N(z) = \begin{cases} (2N+1) & \text{for } z = 0 \\ 0 & \text{for } z = \pm \dfrac{2\pi}{2N+1},\ \dfrac{4\pi}{2N+1}, \ldots \end{cases} \qquad (1.42)$$

From eqn. (1.39) it can be seen that $D_N(z)$ is the window through which $x_n(t)$ views $x(t)$. Dirichlet's kernel has some disadvantages such as not concentration satisfactorily about $t = 0$, and slow convergence etc.

Parzen (1967, pp. 212-213) suggests, as an alternative to eqn. (1.36), the Fejer's arithmetic mean $\overline{x}_N(t)$ defined as

$$\overline{x}_N(t) = \frac{1}{N+1}\,[x_0(t) + x_1(t) + \ldots + x_N(t)] \qquad (1.43)$$

This converges uniformly to $x(t)$, provided $x(t)$ is a continuous periodic function. $\overline{x}_N(t)$ can be expanded into a finite Fourier series as follows

$$\overline{x}_N(t) = \sum_{n=0}^{N} (1 - \frac{n}{N+1})\,(a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \qquad (1.44)$$

or with integral representation (proof is given in Appendix 1.B)

$$\overline{x}_N(t) = \frac{1}{T} \int_0^T F_N(2\pi\,\frac{\lambda - t}{T})\, x(\lambda)\, d\lambda \qquad (1.45)$$

where

$$F_N(z) = \frac{1}{N+1} \left[\frac{\sin(N+1)\frac{1}{2}z}{\sin\frac{1}{2}z}\right]^2, \quad z = 2\pi t/T \qquad (1.46)$$

$F_N(z)$ is called the Fejer kernel.

1.4.2 The Modified Truncated Fourier Series

A modified expression for the truncated Fourier series is defined by

$$x_N^*(t) = x_N(t) - \frac{1}{2}[a_N \cos N\omega_0 t + b_N \sin N\omega_0 t] \qquad (1.47)$$

or with integral representation

$$x_N^*(t) = \frac{1}{T} \int_0^T D_N^*(2\pi\,\frac{\lambda - t}{T})\, x(\lambda)\, d\lambda \qquad (1.48)$$

-11-

where
$$D_N^*(z) = D_N(z) - \cos \tfrac{1}{2} z = \sin N z / \tan(\tfrac{z}{2})$$
(1.49)

1.4.3. Tukey Means

Tukey suggested the following mean to form a harmonic approximation to $x(t)$ (Parzen, 1967, p. 215)

$$x_N^{\cdot}(t) = 0.25 \, x_N^*(t - T/2N) + 0.5 \, x_N^*(t) + 0.25 \, x_N^*(t + T/2N)$$
(1.50)

We can find the window corresponding to Tukey means by

$$x_N^{\cdot}(t) = \frac{1}{T}\int_0^{\cdot\tau} \{0.25 \, D_N^*[\tfrac{2\pi}{T}(\lambda - t) - \tfrac{\pi}{N}] + 0.5 \, D_N^*[\tfrac{2\pi}{T}(\lambda - t)]$$
$$+ 0.25 \, D_N^*[\tfrac{2\pi}{T}(\lambda - t) + \tfrac{\pi}{N}]\} \, x(\lambda) \, d\lambda$$
(1.51)

$$= \int_c^{\tau} \frac{1}{T} \, T_N [\tfrac{2\pi}{T}(\lambda - t)] \, x(\lambda) \, d\lambda$$
(1.52)

where
$$T_N(z) = 0.25 \, D_N^*(z - \tfrac{\pi}{N}) + 0.5 \, D_N^*(z) + 0.25 \, D_N^*(z + \tfrac{\pi}{N}), \quad z = 2\pi t/T \ (1.53)$$

and it is called the "Tukey kernel".

For comparison purposes, we show in Figures 1.2, 1.3 the Dirichlet, Fejer and Tukey kernels. It can be seen that the Tukey kernel provides a better harmonic approximation, since it is more concentrated around the origin than either the Dirichlet or the Fejer kernel.

1.5 Fourier Expansion Using Discrete Data

Lecause of the following reasons we may have to use the values of the function $x(t)$ at the N equispaced points;

- The determination of the Fourier coefficients of the periodic continuous function $x(t)$, with period T, might be difficult because of the difficulty in evaluating the integrals defining the coefficients,

- $x(t)$ might not be known at all points $\underline{t}$ in the interval $0 \leq t \leq T$ but may be known only at equally spaced points.

Suppose the values of the function $x(t)$ are known at N equispaced points $t_0, t_1, t_2, \ldots, t_{N-1}$ defined by

$$t_n = n \Delta t, \quad n = 0, 1, \ldots, (N-1)$$
(1.54)

where $\Delta t = T/N$.

Eqns. (1.3.a-b), which are the Fourier coefficients of eqn. (1.1), are
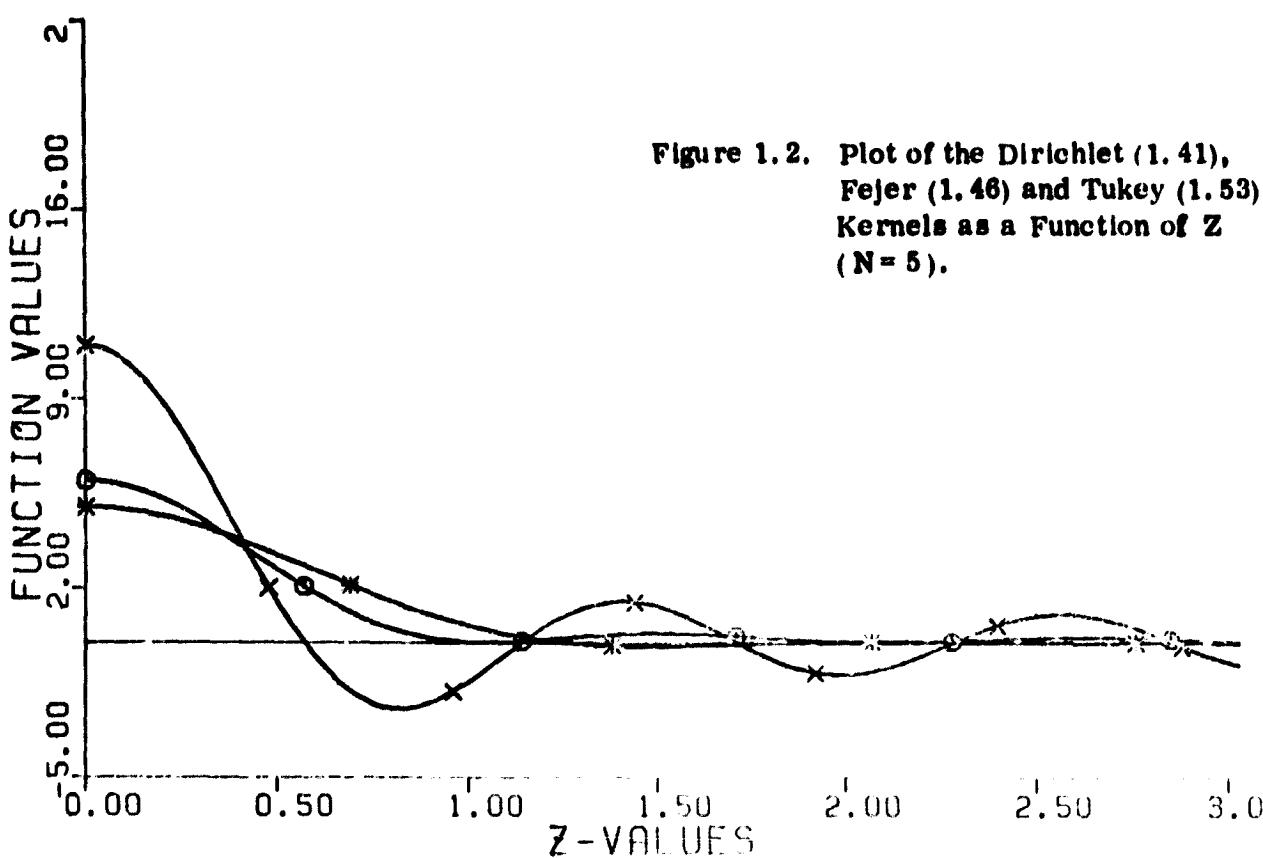
Figure 1.2. Plot of the Dirichlet (1.41), Fejer (1.46) and Tukey (1.53) Kernels as a Function of Z (N = 5).

X = Dirichlet Kernel
☉ = Fejer Kernel
∗ = Tukey Kernel
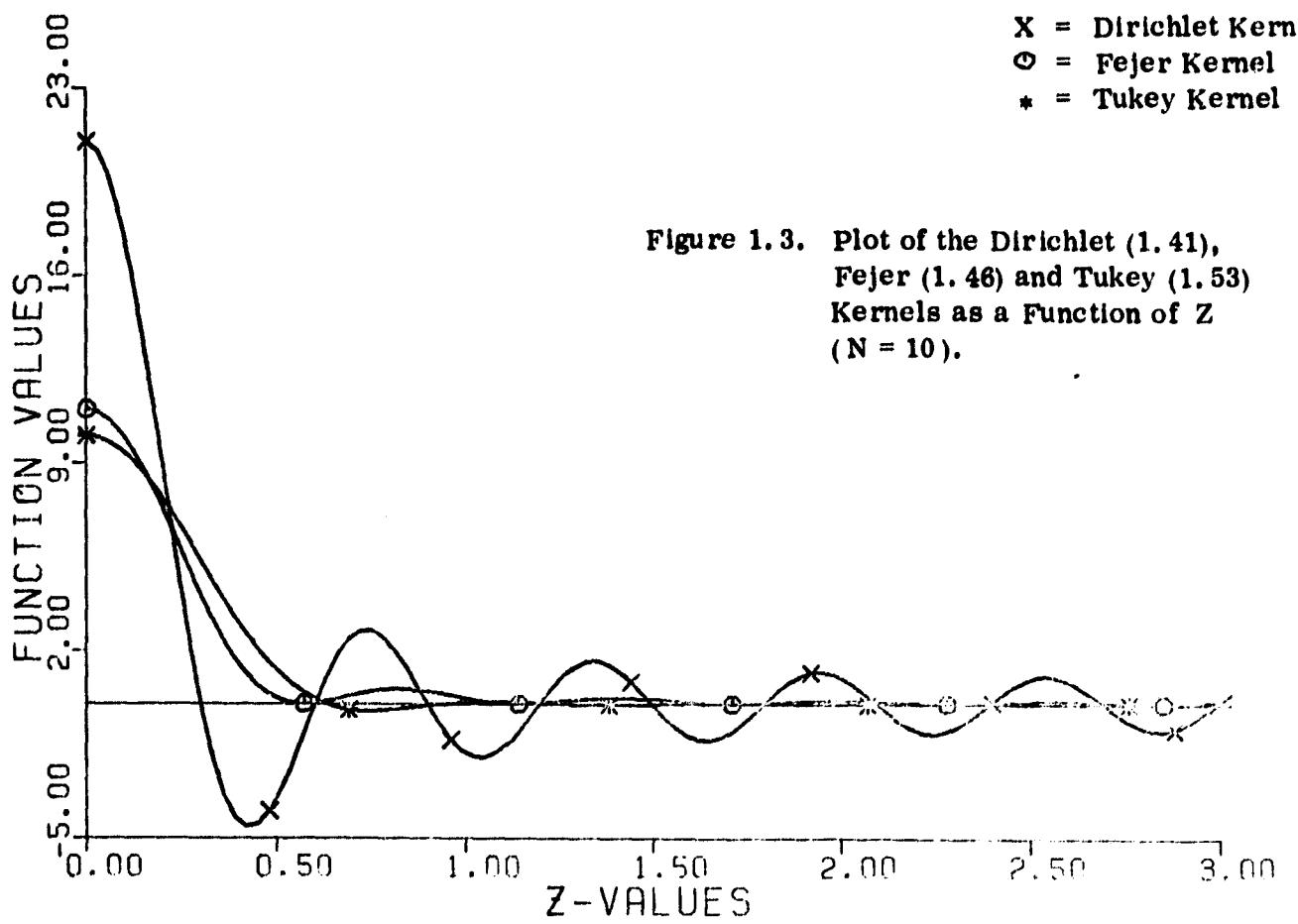
Figure 1.3. Plot of the Dirichlet (1.41), Fejer (1.46) and Tukey (1.53) Kernels as a Function of Z (N = 10).

represented in discrete form, as will be proved later, as follows

$$a_n = \frac{2}{N} \sum_{j=0}^{N-1} x(j \cdot \Delta t) \cos (2\pi j n/N), \quad n = 0, 1, \ldots, N_H \qquad (1.55.a)$$

$$b_n = \frac{2}{N} \sum_{j=0}^{N-1} x(j \cdot \Delta t) \sin(2\pi j n/N), \quad n = 1, 2, \ldots, N_H \qquad (1.55.b)$$

where

$$N_H = \begin{cases} (N-1)/2 & \text{if } N \text{ odd} \\ N/2 & \text{if } N \text{ even} \end{cases} \qquad (1.56)$$

In eqns. (1.55.a-b) we let n go up to N_H, because in order to know the values of the discrete Fourier coefficients (a_n, b_n) to $n = \infty$, it suffices to know them up to N_H. We can justify this conclusion by using the following identies:

$$\begin{aligned} \cos (kN + n) \omega_0 t_j &= \cos (n \omega_0 t_j) \\ \cos (kN - n) \omega_0 t_j &= \cos (n \omega_0 t_j) \\ \sin (kN + n) \omega_0 t_j &= \sin (n \omega_0 t_j) \\ -\sin (kN - n) \omega_0 t_j &= \sin (n \omega_0 t_j) \end{aligned} \qquad (1.57)$$

where $\omega_0 = 2\pi/T = 2\pi/N \cdot \Delta t$ and $t_j = j \cdot \Delta t$ and substituting them in eqns (1.55.a-b) above we can see the periodicity of the Fourier coefficients at basic interval $(-N_H, N_H)$, in other words

$$\left. \begin{aligned} a_n &= a_{kN+n} \\ b_n &= b_{kN+n} \end{aligned} \right\} \quad -N_H \le n \le N_H \text{ and } k = \text{any integer.}$$

Moreover, we notice that $a_n = a_{-n}$ and $b_n = -b_{-n}$, where $0 \le n \le N_H$.

As can be seen from eqn (1.1) that a_n, b_n $n = 0, 1, 2, 3, \ldots, N_H$ are the only distinct Fourier coefficients for N equispaced data points, since from N equispaced observations we can only determine N other quantities namely a_n, b_n, $n = 0, 1, \ldots, N_H$.

If we define frequency as the number of cycles in the data interval $T (=1)$ then N_H is known as "Nyquist frequency or folding frequency". And from discrete observations we can only determine the Fourier coefficients up to the Nyquist frequency. So the eqn. (1.1) takes the following form for the discrete expansion:

$$x(k \cdot \Delta t) = \frac{a_0}{2} + \sum_{n=1}^{N_H} (a_n \cos 2\pi kn/N + b_n \sin 2\pi kn/N), \text{ for } N \text{ odd} \qquad (1.58)$$

$$x(k \cdot \Delta t) = \frac{a_0}{2} + \sum_{n=1}^{N_H-1} (a_n \cos 2\pi kn/N + b_n \sin 2\pi kn/N) + \frac{a_{N_H}}{2} \cos \pi k, \text{ for } N \text{ even} \qquad (1.59)$$

where N_H is defined as in eqn. (1.56).

Now consider the following trigonometric identities

$$\sum_{k=0}^{N-1} \cos 2\pi km/N \cos 2\pi kr/N = \begin{cases} N & \text{for } m = r = 0 \text{ or } N/2 \\ 0 & \text{for } m \ne r \\ N/2 & \text{for } m = r \ne 0 \text{ or } N/2 \end{cases}$$

$$\sum_{k=0}^{N-1} \sin 2\pi km/N \cos 2\pi kr/N = 0 \quad \text{for all } m \text{ and } r$$

$$\sum_{k=0}^{N-1} \sin 2\pi km/N \sin 2\pi kr/N = \begin{cases} N/2 & \text{for } m = r \ne 0 \text{ or } N/2 \\ 0 & \text{otherwise} \end{cases} \qquad (1.60)$$

and suppose the values of the function $x(t)$ at N discrete points are given, then from eqns. (1.58) and (1.59) we can determine the Fourier coefficients a_n, and b_n, which are defined by eqns. (1.55.a-b), i.e.

$$a_n = \frac{2}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \cos 2\pi k n/N, \quad n = 0, 1, \ldots, N_H$$

$$b_n = \frac{2}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \sin 2\pi k n/N, \quad n = 1, 2, \ldots, N_H$$

(1.61)

It is also possible to write the complex and cosinewave forms of the discrete finite Fourier series (corresponding to eqns. (1.6), (1.16), and (1.23)). Suppose the continuous function $x(t)$ with square integrable derivative is defined on the interval $(0, T)$ and evaluated at N points

$$k \cdot \Delta t = k \cdot T/N, \quad k = 0, 1, \ldots, (N-1)$$

then the complex form of the Fourier coefficients are given by

$$X_n^{(N)} = \frac{1}{2}(a_n - i b_n) = \frac{1}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \, e^{-i2\pi k n/N}$$

(1.62)

and the inverse transform $x(k \cdot \Delta t)$ can be expresses as follows

$$x(k \cdot \Delta t) = \begin{cases} \sum_{n=-N_H+1}^{N_H} X_n^{(N)} \, e^{2\pi i k n/N} & \text{for } N \text{ even} \\ \sum_{n=-N_H}^{N_H} X_n^{(N)} \, e^{2\pi i k n/N} & \text{for } N \text{ odd} \end{cases}$$

(1.63)

where N_H is defined in eqn. (1.56).

The expression for the cosinewave form of the function $x(k \cdot \Delta t)$ defined at N equispaced points (corresponding to eqn. (1.23) of continuous function $x(t)$) is

$$x(k \cdot \Delta t) = \frac{a_0}{2} + \sum_{n=1}^{N_H} A_n \cos(2\pi k n/N - \varphi_n), \quad k = 0, 1, \ldots, (n-1)$$

(1.64)

where $A_n = (a_n^2 + b_n^2)^{\frac{1}{2}}$, $\varphi_n = -\tan^{-1}(b_n/a_n)$.

The computation of the complex Fourier coefficients defined by eqn. (1.62) or the computation of a data vector from the complex Fourier coefficients defined by eqn. (1.63) requires roughly N^2 complex multiplications and additions. Since the time of computation of the transforms above increases by the square of N data points, computationally it becomes inefficient or even impossible for very large N. For this reason spectral analysis requiring the discrete Fourier transform (DFT) has not been so common in the past.

In 1924 Runge and Konig discovered a method for efficiently computing DFT. This method is known as the fast Fourier transform (FFT) algorithm. However, Runge and Konig's FFT algorithm has gone unnoticed and FFT was "rediscovered" by Cooley and Tukey in 1965 (Cooley, 1965). FFT takes advantage of the fact that

DFT can be carried out iteratively, which results in a considerable savings of computation time and in an improvement of round-off errors associated with these computations. FFT requires roughly $N \log_2 N$ complex multiplications and additions of the classical discrete Fourier transform method. A detailed explanation of FFT is given in Appendix 1.D.

The discrete Fourier transform pair for the fast Fourier transform (FFT) is given as follows:

$$X_n^{(N)} = \frac{1}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \, e^{-i 2\pi k n / N} \qquad \text{direct Fourier transform} \qquad (1.65)$$

$$x(k \cdot \Delta t) = \sum_{n=0}^{N-1} X_n^{(N)} \, e^{i 2\pi i k n / N} \qquad \text{inverse Fourier transform} \qquad (1.66)$$

This transform pair is usually denoted

$$X_n^{(N)} \longleftrightarrow x(k \cdot \Delta t)$$

We see that eqn. (1.62) can be directly transformed into the frequency domain using FFT, but not eqn. (1.63), where the sum is not from zero to (N-1), in the present form. By rearranging eqn. (1.63) we have,

$$x(k \cdot \Delta t) = \sum_{n=0}^{N_H} Y_n^{(N)} \, e^{i 2\pi k n / N} + \sum_{n=-N_H}^{N_H} Y_n^{(N)} \, e^{i 2\pi k n / N} \qquad (1.67)$$

where

$$Y_n^{(N)} = \begin{cases} X_n^{(N)}/2 & \text{for } n = 0 \\ X_n^{(N)}/2 & \text{for } n = N_H \text{ and } N = \text{even} \\ X_n^{(N)} & \text{otherwise} \end{cases}$$

Eqn. (1.67) is equivalent to

$$x(k \cdot \Delta t) = \sum_{n=0}^{N_H} Y_n^{(N)} \, e^{i 2\pi k n / N} + \sum_{n=0}^{N_H} (Y_n^{(N)})^* \, e^{i 2\pi k n / N} \qquad (1.68)$$

where the superscript (*) shows the complex conjugate of the function inside the parentheses.

Now let the sum go to (N-1) such that

$$x(k \cdot \Delta t) = \sum_{n=0}^{N-1} Z_n^{(N)} \, e^{i 2\pi k n / N} + \sum_{n=0}^{N-1} (Z_n^{(N)})^* \, e^{i 2\pi k n / N} \qquad (1.69)$$

where

$$Z_n^{(N)} = \begin{cases} Y_n^{(N)} & \text{for } 0 \leq n \leq N_H \\ 0 & \text{otherwise} \end{cases} \qquad (1.70)$$

Finally we can write,

$$x(k \cdot \Delta t) = x_1(k \cdot \Delta t) + x_1^*(k \cdot \Delta t) \qquad (1.71)$$

where

$$x_1(k \cdot \Delta t) = \sum_{n=0}^{N-1} Z_n^{(N)} \, e^{i 2\pi k n / N}$$

which can be calculated by FFT, and $x_1^*(k \cdot \Delta t)$ is just the complex conjugate of $x_1(k \cdot \Delta t)$.

Further, we can show the relationship between the complex Fourier representations of discrete data and continuous function, i.e. the integral form. The sampled function $x(k \cdot \Delta t)$ is expressed from eqn. (1.63) as (suppose N is even)

$$x(k \cdot \Delta t) = \sum_{n = -N_H + 1}^{N_H} X_n^{(N)} e^{i2\pi k n/N} \tag{1.72}$$

If we compute the Fourier coefficients of the continuous function $x(t)$ using the integral form eqn. (1.16), we get

$$X_n = \frac{1}{T} \int_T x(t) \, e^{-in\omega_0 t} \tag{1.73}$$

Suppose we evaluate $x(t)$ at the points $(k \cdot \Delta t)$, $k = 0, 1, \ldots, (N-1)$ using X computed by eqn. (1.73). We then obtain

$$x(k \cdot \Delta t) = \sum_{n = -\infty}^{\infty} X_n \, e^{in\omega_0 k \cdot \Delta t}$$
$$= \sum_{n = -\infty}^{\infty} X_n \, e^{i2\pi k n/N} \tag{1.74}$$

By the periodicity of the complex exponential we can write

$$x(k \cdot \Delta t) = \sum_{n = -N_H + 1}^{N_H} e^{i2\pi k n/N} \left[X_n + (X_{n-N} + X_{n+N}) + (X_{n-2N} + X_{n+2N}) + \ldots \right] \tag{1.75}$$

Equating the coefficients of $\exp(2\pi i k n/N)$ of eqn. (1.72) and (1.75), we get

$$X_n^{(N)} = X_n + \sum_{s=1}^{\infty} (X_{n-sN} + X_{n+sN}) \tag{1.76}$$

From now on $X_n^{(N)}$ will be denoted simply as X_n.

It is very important to note that the coefficient for the n th frequency of the function defined at N points is the sum of the coefficients of the continuous function at the $(n, n+N, n-N, n+2N, n-2N, \ldots)$ frequencies. The frequencies $(n \pm N, n \pm 2N, \ldots)$ are called the "ALIASES" of the n th frequency (Fuller, 1976, p. 119). We already know that the frequency $(N/2)$ is called the Nyquist frequency (see p. 114). The aliases of an observed frequency are those frequencies which are obtained by adding or subtracting integer multiples of twice the Nyquist frequency.

1.6 Breakdown of Variance from Sampled Data (at N Points) and Periodogram

1.6.1 Breakdown of Variance

The total average power (or the variance) of the function $x(k \cdot \Delta t)$ is defined as follows

-17-

$$P_{AVG}{}^2 = \frac{1}{N} \sum_{k=0}^{N-1} [x(k \cdot \Delta t)]^2 \tag{1.77}$$

or in terms of Fourier coefficients, i.e. in the frequency domain (from the Parseval's theorem)

$$P_{AVG}{}^2 = (a_0/2)^2 + \frac{1}{2} \sum_{n=1}^{N_M-1} (a_n{}^2 + b_n{}^2) + P_{N_M}{}^2 \tag{1.78}$$

$$= X_0{}^2 + 2 \sum_{n=1}^{N_M-1} |X_n|^2 + P_{N_M}{}^2, \text{ where } P_{N_M}{}^2 = \begin{cases} (a_{N_M}{}^2 + b_{N_M}{}^2)/2 & \text{if N odd} \\ a_{N_M}{}^2/4 & \text{if N even} \end{cases}$$

If we substitute $(a_n{}^2 + b_n{}^2) = A_n{}^2$, where A_n is the amplitude of the cosinewave as defined by eqn. (1.64), we get

$$P_{AVG}{}^2 = A_0{}^2 + \frac{1}{2} \sum_{n=1}^{N_M-1} A_n{}^2 + P_{N_M}{}^2 \tag{1.79}$$

The degree power of this function is given by

$$P_n{}^2 = \begin{cases} (a_n{}^2 + b_n{}^2)/2 = A_n{}^2/2 = 2|X_n|^2 & \text{for } n = 1, 2, \ldots, (N_M - 1) \\ a_0{}^2/4 & \text{for } n = 0 \\ (a_{N_M}{}^2 + b_{N_M}{}^2)/2 & \text{for } n = N_M \text{ and N odd} \\ a_{N_M}{}^2/4 & \text{for } n = N_M \text{ and N even} \end{cases} \tag{1.80}$$

so finally it is possible to write

$$P_{AVG}{}^2 = \sum_{n=0}^{N_M} P_n{}^2 \tag{1.81}$$

1.6.2 Periodogram

The periodogram is a function of frequency (n) and used very frequently in practice. As we will show later it is the discrete Fourier transform (DFT) of the autocovariance function of the data vector in question. Moreover, it is directly related to the degree power defined in the previous section.

The Fourier coefficients of eqns. (1.58-59) can be considered as regression coefficients (Fuller, 1976, p. 276). Then by the standard regression analysis we can partition the total variance $(N \cdot P_{AVG}{}^2)$ of the N observations as follows:

$$I_N(n) = \begin{cases} N \cdot (a_n{}^2 + b_n{}^2)/2 & \text{for } n = 1, 2, \ldots, (N_M - 1) \\ N \cdot a_0{}^2/4 & \text{for } n = 0 \\ N \cdot a_{N_M}{}^2/4 & \text{for } n = N_M \text{ and N even} \\ N \cdot (a_{N_M}{}^2 + b_{N_M}{}^2)/2 & \text{for } n = N_M \text{ and N odd} \end{cases} \tag{1.82}$$

where the Fourier coefficients are given by eqns. (1.55.a-b). Notice that eqn. (1.82) is equivalent to $I_N(n) = N \cdot P_n{}^2$.

$I_N(n)$ is usually known as the "peridogram". If we assume that $x(t)$ is a

-18-

sequence of normal, independent $(0,\sigma^2)$ random variables, then the Fourier coefficients (a_n, b_n), being linear combinations of $x(t)$, will be normally distributed. These coefficients are independent, because the sine and cosine functions are orthogonal in the basic interval. As a result we can write

$$
\begin{aligned}
I_x(n)/\sigma^2 &\sim \chi_2^2 \qquad \text{for } n = 1, 2, \ldots, (N_H - 1)\\
I_x(N_H)/\sigma^2 &\sim \chi_1^2 \qquad \text{for } N \text{ even}\\
I_x(0)/\sigma^2 &\sim \chi_1^2\\
I_x(N_H)/\sigma^2 &\sim \chi_2^2 \qquad \text{for } N \text{ odd}
\end{aligned}
\tag{1.83}
$$

Let us substitute eqns. (1.55. a-b) in eqn. (1.82) in order to get an alternate expression for the periodogram

$$
I_x(0) = \frac{1}{N} \sum_{k=0}^{N-1} [x(k \cdot \Delta t)]^2
\tag{1.84}
$$

$$
I_x(n) = \frac{2}{N} \left\{ \left[\sum_{k=0}^{N-1} x(k \cdot \Delta t) \cos 2\pi kn/N \right]^2 + \left[\sum_{k=0}^{N-1} x(k \cdot \Delta t) \sin 2\pi kn/N \right]^2 \right\}
\tag{1.85}
$$
$$
n = 1, 2, \ldots, N_H
$$

If N even, then $I_x(N_H)$ is given by

$$
I_x(N_H) = \frac{1}{N} \left[\sum_{k=0}^{N-1} x(k \cdot \Delta t) \cos \pi k \right]^2
\tag{1.86}
$$

If we consider the following trigonometric identities

$$
\sum_{k=1}^{N} x(k \cdot \Delta t) \cos 2\pi kn/N = \sum_{k=1}^{N} [x(k \cdot \Delta t) - \mu] \cos 2\pi kn/N
\tag{1.87}
$$

$$
\sum_{k=1}^{N} x(k \cdot \Delta t) \sin 2\pi kn/N = \sum_{k=1}^{N} [x(k \cdot \Delta t) - \mu] \sin 2\pi kn/N
\tag{1.88}
$$

then eqn. (1.85) can be written as

$$
\begin{aligned}
I_x(n) &= \frac{2}{N} \left\{ \left[\sum_{k=1}^{N} (x(k \cdot \Delta t) - \mu) \cos 2\pi kn/N \right]^2 + \left[\sum_{k=1}^{N} (x(k \cdot \Delta t) - \mu) \sin 2\pi kn/N \right]^2 \right\}\\
&= \frac{2}{N} \left\{ \sum_{k=1}^{N} \sum_{j=1}^{N} [x(k \cdot \Delta t) - \mu][x(j \cdot \Delta t) - \mu] \cos 2\pi n(k-j)/N \right\}
\end{aligned}
\tag{1.89}
$$

Now let $k - j = p$, to yield

$$
I_x(n) = \frac{2}{N} \sum_{p=-(N-1)}^{N-1} \sum_{j} [x(j \cdot \Delta t) - \mu][x(j \cdot \Delta t + p \cdot \Delta t) - \mu] \cos 2\pi np/N
\tag{1.90}
$$

where $1 \leq j + p \leq N$ such that

$$
\sum_{j=1}^{N-p} \quad \text{for } p > 0 \quad \text{and} \quad \sum_{j=1}^{N+p} \quad \text{for } p < 0
$$

substitute these in eqn. (1.90) in order to get

$$
I_x(n) = 2 \sum_{p=-(N-1)}^{N-1} \cos 2\pi np/N \left\{ \frac{2}{N} \sum_{j=1}^{N-|p|} [x(j \cdot \Delta t) - \mu][x(j \cdot \Delta t + |p| \cdot \Delta t) - \mu] \right\}
$$

$$
I_x(n) = 2 \sum_{p=-(N-1)}^{N-1} \gamma(p) \cos 2\pi np/N, \quad n = 1, 2, \ldots, N_H
\tag{1.91}
$$

-19-

$$I_N(0) = N \cdot \overline{X}_N^2$$

where $\overline{X}_N = [x(\Delta t) + x(2\Delta t) + \ldots + x(N\Delta t)]/N$ is the mean value of the function.

It can be proven (Fuller, 1976, p. 279) that

$$\lim_{N \to \infty} E[I_N(\omega)] = 4\pi f(\omega), \quad \omega \neq 0 \qquad (1.92)$$

where

$$f(\omega) = \frac{1}{2\pi} \sum_{j=-\infty}^{\infty} \gamma(j) \cos j\omega$$

and

$$\lim_{N \to \infty} E\{I_N(0) - 2N\mu^2\} = 4\pi f(0)$$

Having found the Fourier coefficients, total average power, and degree power, now we can compute the uncertainties of these quantities. If the discrete observations $x(k \cdot \Delta t)$ are independent and normally distributed $N(\mu, \sigma^2)$ with Fourier coefficients (a_n, b_n) defined as in eqn. (1.61), i.e.

$$a_n = \frac{2}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \cos 2\pi k n/N$$
$$b_n = \frac{2}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \sin 2\pi k n/N \qquad (1.93)$$

then, as derived in Appendix 1.C

$$\text{var}(a_n) = E\{a_n - E(a_n)\}^2 = \frac{4\sigma^2}{N} \sum_{k=0}^{N-1} \cos^2 2\pi k n/N \qquad (1.94)$$

using the eqn. (1.60) we find

$$\text{var}(a_n) = \begin{cases} 2\sigma^2/N & \text{for } n \neq 0 \\ 4\sigma^2/N & \text{for } n = 0 \\ 4\sigma^2/N & \text{for } n = N_H \text{ and } N \text{ even} \end{cases} \qquad (1.95)$$

similarly,

$$\text{var}(b_n) = E\{b_n - E(b_n)\}^2 = \begin{cases} 2\sigma^2/N & \text{for } n \neq 0 \\ 0 & \text{for } n = 0, \text{ or} \\ & n = N_H \text{ and } N \text{ even} \end{cases} \qquad (1.96)$$

Finally, we can compute the variance of the average degree power, P_n, which is a function of the Fourier coefficients as given in eqn. (1.80).

$$\text{var}(P_n) = \sigma^2/N \qquad (1.97)$$

1.7 Non-Periodic Functions and the Fourier Transform

1.7.1 Mathematical Definition

In previous sections we dealt with periodic functions exclusively. Now we wish to find out if it is possible to express non-periodic functions as a sum of fundamental sinusoidal functions. The answer is positive, but the "sum" is uncountably infinite (a continuous sum or integral) and each sinusoid has essentially zero amplitude. However the sum of all these infinitesimal sinusoids will produce a non-zero, non-periodic signal.

To present the decomposition of a non-periodic $x(t)$, (Mayhan, 1978, pp. 5-30) we may start with a periodic function $x_T(t)$ and allow the period to extend to infinity so that a non-periodic $x(t)$ results. If we consider the periodic function shown in Figure 1.4
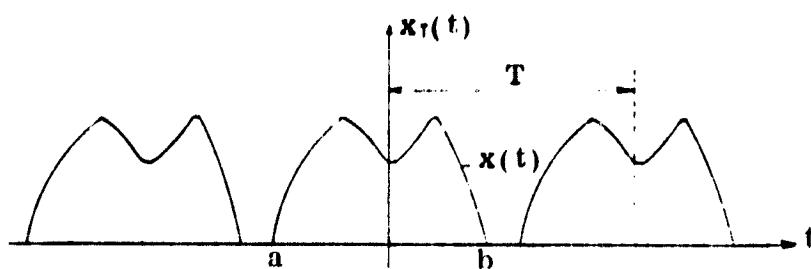
Figure 1.4. A Periodic Function $x_T(t)$, $-\infty < t < \infty$ and a Non-Periodic Function $x(t)$, $a < t < b$.

from eqn. (1.6) we have

$$x_T(t) = \sum_{n=-\infty}^{\infty} X_n \, e^{i n \omega_0 t} \tag{1.98}$$

with complex Fourier coefficients,

$$X_n = \frac{1}{T} \int_{-T/2}^{T/2} x_T(t) \, e^{-i n \omega_0 t} \, dt \tag{1.99}$$

As T increases the number of important frequencies in the expansion of $x_T(t)$ will also increase, but each having smaller and smaller amplitude. If we define frequency as the number of cycles in a specified interval T (period) then we have for the fundamental frequencies f_0 and ω_0

$$f_0 = 1/T, \qquad \omega_0 = 2\pi/T \tag{1.100}$$

then, we have for frequencies

$$
\begin{aligned}
f &= n f_0 \sim 1,\ 2,\ 3,\ 4,\ldots & \text{for } T = 1 \\
f &= n f_0 \sim \tfrac{1}{2},\ 1,\ \tfrac{3}{2},\ 2,\ldots & \text{for } T = 2
\end{aligned}
\qquad n = 1, 2, \ldots
$$

-21-

Thus, as $T \to \infty$ we have $x_r(t) \to x(t)$ and

$$\lim_{r \to \infty} \frac{2\pi n}{T} = \omega, \quad -\infty < n < \infty \tag{1.101}$$

so a general non-periodic $x(t)$ will result.

The harmonic content of $x(t)$ will consist of all angular frequencies ω, and a continuous sum, i.e. in integral, will replace the discrete sum. Since ω_0 is the fundamental angular frequency, in other words the frequency spacing, the spacing approaches a small infinitesimal amount denoted by $d\omega$ as $T \to \infty$.

$$d\omega = \lim_{r \to \infty} \frac{2\pi}{T} \tag{1.102}$$

This being the case, the amplitude at any frequency will tend to zero. But the expansion of $x_r(t)$ will not be zero, since we deal here with an uncountably infinite sum of quantities. Now using the relations above we can write the complex Fourier coefficients corresponding to eqn. (1.16) as

$$X_n = \frac{d\omega}{2\pi} \int_{-\infty}^{\infty} x(t) \ e^{-i\omega t} \ dt \tag{1.103}$$

It is common to define

$$X(\omega) = \int_{-\infty}^{\infty} x(t) \ e^{-i\omega t} \ dt \tag{1.104}$$

to obtain for eqn. (1.103)

$$X_n = \frac{d\omega}{2\pi} \ X(\omega) \tag{1.105}$$

or in case of discrete data with N observations (or sampled values)

$$X_n = \frac{1}{N \cdot \Delta t} \ X(\omega) \tag{1.106}$$

where Δt is the data (or sampling) interval; and as $T \to \infty$, so does N. If we substitute eqn. (1.105) in the Fourier (complex) expansion of $x_r(t)$, then we end up with a continuous sum for the expansion of $x(t)$ defined as follows

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) \ e^{i\omega t} \ d\omega \tag{1.107}$$

Eqns. (1.104) and (1.107) are the Fourier transform pair (also known as Fourier integrals) for the non-periodic function $x(t)$ and usually denoted as

$$x(t) \iff X(\omega)$$

There is a one-to-one correspondence between $x(t)$ and $X(\omega)$. It is also common practice to use the standard frequency instead of angular frequency. So if we substitute $2\pi f$ for ω we have from eqns. (1.104) and (1.107),

$$X(f) = \int_{-\infty}^{\infty} x(t) \; e^{-i2\pi f t} dt \tag{1.108}$$

$$x(t) = \int_{-\infty}^{\infty} X(f) \; e^{i2\pi f t} \; df \tag{1.109}$$

In the case of periodic functions we had related the complex Fourier coefficients X_n to the amplitudes of the sinusoidal function. For a non-periodic function $x(t)$, it is possible to interpret $X(\omega)$ as the amplitude density. That is to say, the amplitude of the sinusoid is zero at any one frequency ω, however when it is considered over any finite frequency interval $d\omega$ (or $\Delta\omega$), the amplitude is not zero but equal to

$$X_n = \frac{d\omega}{2\pi} X(\omega)$$

Thus, the harmonic content of $x(t)$ can be considered as distributed over all frequencies with a density $X(\omega)$.

For non-periodic functions, the Fourier coefficients corresponding to (1.08) and (1.09) are given below

$$a(f) = \int_{-\infty}^{\infty} x(t) \cos(2\pi f t) \; dt \tag{1.110}$$

$$b(f) = \int_{-\infty}^{\infty} x(t) \sin(2\pi f t) \; dt \tag{1.111}$$

The equations above can be derived from eqns. (1.3.a-b) of a periodic function as follows:

$$a(f) = \lim_{T \to \infty} \frac{T}{2} \; a_n = \int_{-\infty}^{\infty} x(t) \cos 2\pi n t/T \; dt$$

substitute $f = n/T$, to obtain

$$a(f) = \int_{-\infty}^{\infty} x(t) \cos(2\pi f t) \; dt \tag{1.112}$$

similarly $b(f)$ can be derived. It can also be seen from eqns. (1.108) and (1.110) that

$$X(f) = a(f) - i \, b(f) \tag{1.113}$$

If the function is known only at discrete points, then the relation between the complex Fourier coefficients of a periodic function and non-periodic function can be derived through eqns. (1.55.a-b), namely

$$\hat{a}(n) = \lim_{N \to \infty} \frac{N}{2} \; a_n = \lim_{N \to \infty} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \cos 2\pi k n/N \tag{1.114.a}$$

$$\hat{b}(n) = \lim_{N \to \infty} \frac{N}{2} \; b_n = \lim_{N \to \infty} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \sin 2\pi k n/N \tag{1.114.b}$$

We know that a_n is an amplitude, $1/T$ is the frequency increment, then, $(T \cdot a_n)$ is the amplitude per frequency increment of amplitude density, and finally, $a(f)$ is half the amplitude density at frequency "f" (Rayner, 1971, p.

56). Rayner says "The relationship between amplitude density and amplitude is analogous to that between probability density and probability."

In section 1.3.2 we expanded a periodic function $x(t)$ with period T into the complex Fourier series and proved that the complex Fourier coefficients X_n are orthogonal, i.e.

$$E\{X_n X_m\} = \begin{cases} 0 & \text{for } n \neq m \\ \alpha_n & \text{for } n = m \end{cases}$$

where X_n and α_n are defined by eqns. (1.8) and (1.10) respectively.

If the function is not periodic, to be denoted by $\hat{x}(t)$, then neither is its autocovariance, say $\hat{C}(\tau)$. Hence for $\tau \neq 0$ we always have $\hat{C}(\tau) < \hat{C}(0)$. If for some $\tau = \tau_1$, we have $\hat{C}(\tau_1) = C(0)$, then $\hat{C}(\tau)$ is certainly periodic. Suppose a non-periodic $\hat{C}(\tau)$ is given. For a fixed t shown in Figure 1.5 we can expand it into the Fourier series in the interval $(-P/2, P/2)$, where P is the given interval length (Papoulis, 1965).
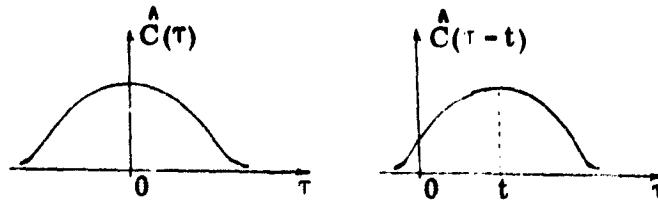
Figure 1.5. A Non-Periodic Function.

The coefficients of the expansion would depend on t in the following way,

$$\hat{C}(\tau - t) = \sum_{n=-\infty}^{\infty} \beta_n(t) \, e^{in\omega_0 t}, \qquad |\tau| < P/2 \qquad (1.115)$$

$$\beta_n(t) = \frac{1}{P} \int_P C(\tau - t) \, e^{-in\omega_0 t} \, d\tau \qquad (1.116)$$

The non-periodic function $\hat{x}(t)$ can be expanded into a Fourier series in the interval $(-P/2, P/2)$ as follows

$$\hat{x}(t) = \sum_{n=-\infty}^{\infty} X_n \, e^{in\omega_0 t}, \qquad \begin{cases} \omega_0 = 2\pi/T \\ |t| < P/2 \end{cases} \qquad (1.117)$$

We can show that $\hat{x}(t) = x(t)$ in the defined interval. That is to say

$$E\{|x(t) - \hat{x}(t)|^2\} = 0 \qquad \text{for } |t| < P/2 \qquad (1.118.a)$$

where E denotes the expectation. Eqn. (1.118.a) is equivalent to

$$E\{|x(t)|^2\} - E\{x(t)\,\hat{x}^*(t)\} - E\{x^*(t)\,\hat{x}(t)\} + E\{|\hat{x}(t)|^2\} \qquad (1.118.b)$$

and we know that

-24-

$$E[|x(t)|^2] = E[|\hat{x}(t)|^2] = \hat{C}(0) \qquad \text{for } |t| < P/2$$

We can also show that

$$E[x(t) x^*(t)] = E[x^*(t) x(t)] = \hat{C}(0) \quad \text{for } |t| < P/2$$

For the proof: first multiply both sides of eqn. (1.117) by $x^*(t)$, then take the expectation

$$E\{\hat{x}(t) x^*(t)\} = \sum_{n=-\infty}^{\infty} E\{X_n x^*(t)\} e^{i n \omega_0 t} \qquad (1.119)$$

and use eqn. (1.12) to obtain

$$E\{X_n x^*(t)\} = \frac{1}{P} \int_P E\{x(\tau) x^*(t)\} e^{-i n \omega_0 \tau} d\tau = \frac{1}{P} \int_P C(\tau-t) e^{-i n \omega_0 \tau} d\tau = \beta_n(t) \quad (1.120)$$

finally substitute eqn. (1.120) in (1.119) to conclude the proof.

$$E\{\hat{x}(t) x^*(t)\} = \sum_{n=-\infty}^{\infty} \beta_n(t) e^{i n \omega_0 t} dt = \hat{C}(0) \qquad (1.121)$$

Similarly we can show that

$$E\{\hat{x}^*(t) x(t)\} = \hat{C}(0) \qquad (1.122)$$

so we see that each term of eqn. (1.118.b) is equal to $\hat{C}(0)$. Thus the proof is completed.

As to the non orthogonality of the complex Fourier coefficients of non-periodic functions; we will show that they are approximately orthogonal in a finite domain.

First, let us compute the expectation of the products of complex Fourier coefficients, $[X_n X_m^*]$. We can define this product as follows:

$$X_n X_m^* = \frac{1}{P} \int_P X_n x^*(t) e^{i m \omega_0 t} dt \qquad (1.123)$$

now use eqn. (1.120) to obtain

$$E\{X_n X_m^*\} = \frac{1}{P} \int_P \beta_n(t) e^{i m \omega_0 t} dt \qquad (1.124)$$

We see that eqn. (1.124) is not zero for $n \neq m$, i.e. The complex Fourier coefficients are no longer orthogonal. If $\hat{C}(\tau)$ were periodic with period T, then we could easily show that

$$\beta_n(t) = \alpha_n \exp(-i n \omega_0 t)$$

and we could write

$$E[X_n X_m^*] = \begin{cases} \alpha_n & m = n \\ 0 & m \neq n \end{cases}$$

Since $\overset{\Lambda}{C}(\tau)$ is not periodic the coefficients X_n are not orthogonal. However, in eqn. (1.120) if we replace the integration limits by $\pm\infty$, then we commit an error $\delta(t, P)$ which is small for large P, so we can write

$$P \cdot \beta_n(t) = \int_{-\infty}^{\infty} \overset{\Lambda}{C}(\tau - t)\, e^{-in\omega_0\tau}\, d\tau + \delta(t, P) = e^{-in\omega_0 t}\, s(n\omega_0) + \delta(t, P) \qquad (1.125)$$

Substitute eqn. (1.125) in (1.124) to obtain

$$P \cdot E\{X_n\, X_a^*\} = \frac{s(n\omega_0)}{P} \int_P e^{i(a-n)\omega_0 t}\, dt + \frac{1}{P} \int_P \delta(t, P)\, e^{ia\omega_0 t}\, dt \qquad (1.126)$$

From the equation above we can write

$$\lim_{P \to \infty} P \cdot E\{X_n\, X_a^*\} = \begin{cases} s(n\omega_0) & \text{for } n = m \\ 0 & \text{otherwise} \end{cases} \qquad (1.127)$$

We see that $E\{|X_n|^2\}$ tends to zero as $(1/P)$ for $P \to \infty$, whereas the autocovariance $E\{X_n\, X_a^*\}$ tends to zero faster, i.e. the correlation coefficients of X_n and X_a tend to zero also. So we can say that the coefficients X_n's of eqn. (1.117), for large P, are approximately uncorrelated.

1.7.2 Properties of the Fourier Transform

The Fourier transform has some very important properties such as linearity, scaling, time shifting, modulation, symmetry, differentiation, convolution, etc., which are well explained in every text book in this topic (e.g. see Bath, 1974, 11. 42-48). As a result of these properties, some important relations are given in Table 1.1.

Table 1.1. Summary of Fourier Transforms.

Function	Fourier Transform
$x(t)$	$X(\omega)$
$X(t)$	$2\pi x(-\omega)$
$a_1 x_1(t) + a_2 x_2(t)$	$a_1 X_1(\omega) + a_2 X_2(\omega)$
$x(at)$	$(1/|a|)\, X(\omega/a)$
$x(t-t_0)$	$\exp(-i\omega t_0)\, X(\omega)$
$\exp(i\omega_0 t)\, x(t)$	$X(\omega - \omega_0)$
$x(t) \cdot \cos \omega_0 t$	$(X(\omega - \omega_0) + X(\omega + \omega_0))/2$
$x_1(t) \cdot x_2(t)$	$X_1(\omega) * X_2(\omega)/2\pi$ [convolution]
$x_1(t) * x_2(t)$	$X_1(\omega) \cdot X_2(\omega)$
$p^n x(t)$ [n^{th} derivative $x(t)$]	$(i\omega)^n\, X(\omega)$
$x(-t)$	$X(-\omega)$

In the following sections the convolution property will be applied. The convolution of arbitrary functions $x(t)$ and $y(t)$ is defined as follows:

$$x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)\, y(t-\tau)\, d\tau$$

$$= \int_{-\infty}^{\infty} x(t-\tau)\, y(\tau)\, d\tau$$

We can apply the convolution both in the time domain and in the frequency domain.

1.7.2.1 Convolution in the Time Domain

If

$$x(t) \longleftrightarrow X(\omega)$$

$$y(t) \longleftrightarrow Y(\omega')$$

then

$$x(t) * y(t) \longleftrightarrow X(\omega)\, Y(\omega)$$

The above transform pair can be proven by the following procedure:

$$z(t) = x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)\, y(t-\tau)\, d\tau$$

The Fourier transform of $z(t)$ is

$$Z(\omega) = F[x(t) * y(t)] = \int_{-\infty}^{\infty} e^{-i\omega t} \int_{-\infty}^{\infty} x(\tau)\, y(t-\tau)\, d\tau\, dt$$

$$= \int_{-\infty}^{\infty} x(\tau)\, d\tau \int_{-\infty}^{\infty} y(t-\tau)\, e^{-i\omega t}\, dt$$

Now let $\lambda = t - \tau$, $d\lambda = dt$ and with τ held constant

$$Z(\omega) = \int_{-\infty}^{\infty} x(\tau)\, d\tau \int_{-\infty}^{\infty} y(\lambda)\, e^{-i\omega\lambda}\, e^{-i\omega\tau}\, d\lambda$$

$$= \int_{-\infty}^{\infty} x(\tau)\, e^{-i\omega\tau}\, d\tau \int_{-\infty}^{\infty} y(\lambda)\, e^{-i\omega\lambda}\, d\lambda$$

Finally we can write

$$Z(\omega) = F[x(t) * y(t)] = X(\omega)\, Y(\omega)$$

1.7.2.2 Convolution in the Frequency Domain

If

$$x(t) \longleftrightarrow X(\omega)$$

$$y(t) \longleftrightarrow Y(\omega)$$

then

$$x(t)\, y(t) \longleftrightarrow 1/2\pi\ X(\omega) * Y(\omega)$$

We can prove this transform pair by defining $z(t) = x(t) y(t)$ and taking the Fourier transform of $z(t)$

$$Z(\omega) = F[x(t) y(t)] = \int_{-\infty}^{\infty} x(t) y(t) e^{-i\omega t} dt$$

For $x(t)$ substitute the following inverse Fourier transform

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\lambda) e^{i\lambda t} d\lambda$$

to obtain

$$Z(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\lambda) d\lambda \int_{-\infty}^{\infty} y(t) e^{-i(\omega-\lambda)\cdot} dt$$

Since the second integral is the Fourier transform of $y(t)$, $(\omega-\lambda)$ being the angular frequency, we have

$$Z(\omega) = F[x(t) y(t)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\lambda) Y(\omega-\lambda) d\lambda$$

or equivalently

$$Z(\omega) = \frac{1}{2\pi} X(\omega) * Y(\omega)$$

1.7.3 The Finite Record Length and Windows

1.7.3.1 Time Domain Windows

In practice we cannot record the results of an experiment of infinite length, so the use of limited record lengths are inevitable in spectral analysis of observational series. This forces us to truncate the record at some length, say T, such that

$$y(t) = \begin{cases} x(t) & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \tag{1.128}$$

where $x(t)$ is a non-periodic function defined for $(-\infty, \infty)$ but recorded only in the interval $(-T/2, T/2)$, this record being denoted by $y(t)$. Actually this truncation is nothing else but applying the rectangular (box-car) window. Then we can write

$$y(t) = x(t) u_T(t), \quad \text{where } u_T(t) = \begin{cases} 1 & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \tag{1.129}$$

thus, the Fourier transform of $y(t)$ is

$$Y(\lambda) = \int_{-\infty}^{\infty} y(t) e^{-i2\pi f t} dt = \int_{-T/2}^{T/2} y(t) e^{-i2\pi f t} dt \tag{1.130.a}$$

or equivalently

$$Y(\lambda) = \int_{-\infty}^{\infty} x(t) u_T(t) e^{-i\lambda t} dt \tag{1.130.b}$$

If we substitute eqn. (1.107) for $x(t)$ above and evaluate the integral, in other words, use the convolution technique, then we have

$$Y(\lambda) = \frac{1}{2\pi} X(\lambda) * U(\lambda) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) U_T(\omega-\lambda) d\omega \tag{1.131}$$

-28-

where, $U_T(\omega - \lambda)$ is the Fourier transform of $u_T(t)$ defined by

$$U_T(\omega) = (\sin \omega T/2)/(\omega/2) \tag{1.132}$$

Therefore,

$$Y(\lambda) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) \frac{\sin[(\omega - \lambda)(T/2)]}{(\omega - \lambda)/2} \tag{1.133}$$

In the limit as $T \to \infty$, the right-hand side of eqn. (1.132) reduces to $X(\omega)$. Our main goal is to determine the true spectrum $X(\omega)$, but because of limited data we can only estimate $Y(\omega)$. So we must design a window, say $w(t)$, in such a way that $Y(\omega)$ will be very close to $X(\omega)$. Even with the application of the most optimum windows we will get a distorted spectrum. The main task, then, is to keep the spectral distortion close to a minimum.

Windows can be applied either in the time domain or in the frequency domain. **In the time domain:**

$$y(t) = y(t) \cdot w(t) = x(t)[u_T(t) \cdot w(t)] = x(t) \cdot w(t)$$

In the frequency domain:

$$Y(\omega) = \frac{1}{2\pi} X(\omega) * W(\omega) = \frac{1}{2\pi} Y(\omega) * W(\omega)$$

That is to say, either we compute $Y(\omega)$ and convolve it with $W(\omega)$, or instead we multiply the data vector by a window vector in the time domain and compute the Fourier transform.

We can state some of the properties, which a spectral window corresponding to the applied time window must possess:

- small or insignificant side-lobes, i.e. smooth time window without sharp corners
- high concentration at the main (central) lobe
- symmetry with respect to the y-axis, in other words, the window function must be even.

Designing an optimum window for a specific purpose has been a major challenge to scientists in this subject.

Bath (1974, p. 157) classifies windows as follows:

(1) Trigonometric windows, which use trigonometric functions of time
(2) Power windows
(3) Exponential windows

Some of the most commonly used windows will be explained below.

(1) Rectangular (Box-car or Barlett) Window

$$w(t) = \begin{cases} 1 & \text{for } |t| \le T/2 \\ 0 & \text{otherwise} \end{cases} \tag{1.134.a}$$

with spectral representation

$$W(\omega) = (\sin \omega T/2)/(\omega/2) = T \text{ sinc } (\omega T/2\pi) \tag{1.134.b}$$

where sinc $(x) = \sin \pi x/(\pi x)$ is how the sinc function is defined. A rectangular window for a Discrete Fourier Transform (DFT) is defined as (Harris, 1978, p. 58)

$$w(n) = \begin{cases} 1 & \text{for } n = 0, 1, \dots, (N-1) \\ 0 & \text{otherwise} \end{cases} \tag{1.135.a}$$

then the spectral window for the DFT window is given by

$$W(\theta) = \frac{\sin(\theta(N/2))}{\sin(\theta/2)} e^{-i\theta\frac{N-1}{2}}, \quad \theta = 0, 1, \dots, (N-1) \tag{1.135.b}$$

Since the sinc-function has large side-lobes, this window is not very suitable for a true representation of a spectrum. In addition to this, the magnitude of these side-lobes decreases slowly and half of them are negative causing displeasing results, because the power is positive by definition.

Notice that a large T will lead to more details, i.e. to better resolution in the computed spectrum than a small T. But a small T will lead to better stability and reliability of the computed spectral estimates, since for small T's the spectral smoothing extends over a larger frequency interval.

In order to eliminate the effect of the rectangular window and the effect of noise, Bath (1974, p. 180) suggests to use smoothing in the frequency domain, e.g.

$$X_0' = \begin{Bmatrix} 0.54 \\ 0.50 \end{Bmatrix} X_0 + \begin{Bmatrix} 0.46 \\ 0.50 \end{Bmatrix} X_1$$

$$X_k' = \begin{Bmatrix} 0.23 \\ 0.25 \end{Bmatrix} X_{k-1} + \begin{Bmatrix} 0.54 \\ 0.50 \end{Bmatrix} X_k + \begin{Bmatrix} 0.23 \\ 0.25 \end{Bmatrix} X_{k+1}$$

$$X_{N/2}' = \begin{Bmatrix} 0.46 \\ 0.50 \end{Bmatrix} X_{(N/2)-1} + \begin{Bmatrix} 0.54 \\ 0.50 \end{Bmatrix} X_{N/2}$$

where X_n are complex raw Fourier coefficients, whereas X_n' are smoothed complex Fourier coefficients. The above expressions are equivalent to applying the Hanning and Hamming lag windows in time domain respectively.

(2) Triangular (Fejer) Window, $\Delta_T(t)$

$$\Delta_T(t) = \begin{cases} 1 - 2|t|/T & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \qquad (1.136.a)$$

with the spectral representation

$$\Delta(\omega) = \frac{T}{2} \frac{\sin^2(\omega\frac{T}{4})}{(\omega\frac{T}{4})^2} = \frac{T}{2} \operatorname{sinc}^2 \frac{\omega T}{4\pi} \qquad (1.136.b)$$

The triangular window for a DFT is defined as (Harris, 1978, p. 59)

$$\Delta(n) = \begin{cases} n/(N/2) & \text{for } n = 0, 1, \ldots, (N/2) \\ \Delta(N-n) & \text{for } n = (N/2), \ldots, (N-1) \end{cases} \qquad (1.137.a)$$

with the corresponding spectral window

$$\Delta(\theta) = \frac{2}{N} \frac{\sin(\theta\frac{\pi}{4})^2}{\sin(\frac{\theta}{2})^2} e^{-i\theta\frac{N-2}{2}}, \qquad \theta = 0, 1, \ldots, (N-1) \qquad (1.137.b)$$

This spectral window has no negative side-lobes, but it has large main lobes, which do not have high concentration.

(3) Hanning (Tukey or Cosine) Window

$$h_1(t) = \begin{cases} (1 + \cos 2\pi t/T)/2 & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \qquad (1.138.a)$$

with the spectral representation

$$\begin{aligned} H_1(\omega) &= \tfrac{1}{2} \frac{\sin\omega(T/2)}{(\omega/2)} + \frac{T}{4}\left[\frac{\sin(\omega(T/2)+\pi)}{\omega(T/2)+\pi} + \frac{\sin(\omega(T/2)-\pi)}{\omega(T/2)-\pi} \right] \\ &= \frac{T}{2} \operatorname{sinc} \frac{\omega T}{2\pi} + \frac{T}{4}\left[\operatorname{sinc}(\omega(T/2\pi)+1) + \operatorname{sinc}(\omega(T/2\pi)-1) \right] \end{aligned} \qquad (1.138.b)$$

or as a function of $W(\omega)$ of rectangular window

$$H_1(\omega) = \frac{1}{4} W(\omega - \frac{2\pi}{T}) + \tfrac{1}{2} W(\omega) + \frac{1}{4} W(\omega + \frac{2\pi}{T}) \qquad (1.138.c)$$

The Hanning window for a DFT is defined as (Harris, 1978, p. 60)

$$h_1(n) = \begin{cases} \tfrac{1}{2}(1 - \cos 2\pi n/N) & \text{for } n = 0, 1, \ldots, (N-1) \\ 0 & \text{otherwise} \end{cases} \qquad (1.139.a)$$

with the corresponding spectral window

$$H_1(\theta) = \tfrac{1}{2} W(\theta) + \frac{1}{4} [W(\theta - \frac{2\pi}{N}) + W(\theta + \frac{2\pi}{N})] \qquad (1.139.b)$$

where $W(\theta)$ is defined in eqn. (1.135.b). Thus, we see that $H_1(\omega)$ is a sum of three sinc-functions displaced relative to each other and the side-lobes from the three sinc-functions cancel each other to a large extent, which is pleasing.

(4) Hamming Window

$$h_3(t) = \begin{cases} (0.54 + 0.46 \cos 2\pi t/T) & \text{for } |t| < T/2 \\ 0 & \text{otherwise} \end{cases} \qquad (1.140.a)$$

with the spectral representation

$$H_3(\omega) = 0.54\, T\, \frac{\sin \omega(T/2)}{\omega(T/2)} + 0.23\, T\left[\frac{\sin(\omega(T/2)+\pi)}{\omega(T/2)+\pi} + \frac{\sin(\omega(T/2)-\pi)}{\omega(T/2)-\pi}\right]$$

$$= 0.54\, T \sin c\, \frac{\omega T}{2\pi} + 0.23 T\left[\sin c(\frac{\omega T}{2\pi}+1) + \sin c(\frac{\omega T}{2\pi}-1)\right] \qquad (1.140.b)$$

or in terms of $W(\omega)$ of the rectangular window

$$H_3(\omega) = 0.54\, W(\omega) + 0.23\,[\,W(\omega+(2\pi/T)) + W(\omega-(2\pi/T))\,] \qquad (1.140.c)$$

The Hamming window for a DFT is defined as (Harris, 1978, p. 62)

$$h_3(n) = \begin{cases} 0.54 - 0.46 \cos 2\pi n/N & \text{for } n = 0, 1, \ldots, (N-1) \\ 0 & \text{otherwise} \end{cases} \qquad (1.141.a)$$

with the corresponding spectral window

$$H_3(\theta) = 0.54\, W(\theta) + 0.23\,[\,W(\theta-(2\pi/N)) + W(\theta+(2\pi/N))\,], \qquad (1.141.b)$$
$$\theta = 0, 1, \ldots, (N-1)$$

where $W(\theta)$ is defined in eqn. (1.135.b). Notice that the Hanning and Hamming spectral windows correspond to a weighted average over three consecutive values of the spectral function of the box-car window. The Hamming window also has small side-lobes.

(5) Cosine-Tapered Rectangular Window

This window is flat over most of the data, but tapers off near the two ends of the data. For example, it may consist of cosine window at the ends (say 10% at each end) and a rectangular window in between defined as follows:

$$c(t) = \begin{cases} (1 + \cos 10\pi t/T)/T & \text{for } -(T/2) \le t < -0.4\,T \\ 2/T & \text{for } -0.4\,T \le t < 0.4\,T \\ (1 + \cos 10\pi t/T)/T & \text{for } 0.4\,T < t < T/2 \end{cases} \qquad (1.142.a)$$

with the spectral representation

$$C(\omega) = \frac{1}{2}\, \frac{\sin \omega(T/2) + \sin 4\omega(T/10)}{\omega(T/2)\,[1 - (\omega(T/10\pi))^2]} \qquad (1.142.b)$$

The above window for a DFT is defined as (Rayner, 1971, p. 83)

$$c(n) = \begin{cases} \frac{1}{2}[\,1 - \cos(10\pi n)/N\,] & , \quad 0 < n < N/10 \\ 1 & , \quad N/10 < n < 9N/10 \\ \frac{1}{2}[\,1 - \cos(10\pi(N-n))/N\,], & 9N/10 < n < (N-1) \end{cases} \qquad (1.143)$$

-32-

An extensive study of various windows designed for different purposes is given in Harris (1978).

1.7.3.2 Frequency Domain Windows

It is also possible to apply windows in the frequency domain and compute the effects in the time domain. As an example, suppose that $X(\omega)$ of a function $x(t)$ is truncated outside $(-P, P)$ yielding $X_r(\omega)$ defined as

$$X_r(\omega) = X(\omega) W_p(\omega), \quad \text{where } W_p(\omega) = \begin{cases} 1 & \text{for } |\omega| \cdot p \\ 0 & \text{otherwise} \end{cases} \tag{1.144}$$

Since eqn. (1.144) corresponds to a convolution of $x(t)$ and $w(t)$ in the time domain, we have

$$x_b(t) = x(t) * w_p(t) = \int_{-\infty}^{\infty} x(\tau) \frac{\sin(T/2)(t-\tau)}{\pi(t-\tau)} d\tau \tag{1.145}$$

Instead of the rectangular truncation above, if we eliminate the components of $|\omega| > P$ and favor linearly the low-frequency components, then we have

$$X_\Delta(\omega) = X(\omega) \cdot \Delta_p(\omega) \tag{1.146.a}$$

where $\Delta_p(\omega)$ is the triangular pulse defined by

$$\Delta_p(\omega) = \begin{cases} (1 - |\omega|/p) & \text{if } |\omega| \leq p \\ 0 & \text{otherwise} \end{cases} \tag{1.146.b}$$

The corresponding equation in the time domain can be expressed via the convolution theorem as

$$X_\Delta(t) = \int_{-\infty}^{\infty} x(\tau) \frac{2 \sin^2\left[(T/4)(t-\tau)\right]}{\pi(T/2)(t-\tau)} d\tau \tag{1.147}$$

1.7.4 Power Spectrum Variance Breakdown of Non-Periodic Functions

The energy content of a non-periodic function $x(t)$ is defined by

$$E^2 = \int_{-\infty}^{\infty} x^2(t) dt \tag{1.148}$$

or equivalently in the frequency domain by,

$$E^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) X(-\omega) d\omega = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\omega)|^2 d\omega \tag{1.149}$$

If $x(t)$ does not have a finite energy, then we have to consider the total average power P^2 expressed as follows

$$P^2 = \lim_{\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x^2(t) dt \tag{1.150}$$

Papoulis (1962, p. 240) classifies functions into the following three classes according to the magnitude of the power:

-33-

(1) $P^2 = 0$, obviously contains all functions with finite energy content.

(2) $0 < P^2 < \infty$, contains all periodic functions, i.e. $P^2 = \sum_{n=-\infty}^{\infty} |X_n|^2$ where X_n are complex Fourier coefficients of the function $x(t)$, and of some non-periodic functions as well.

(3) $P^2 = \infty$, i.e. $x(t)$ has infinite power.

Power Spectrum from Sequenced Data

(a) Using Autocovariance Function

For the statistical analysis of dependent, sequenced data, say $x(t)$, the mean variance and the autocovariance function characterize the distribution completely. But generally the autocovariance function is difficult to interpret and adjacent estimates are not independent, so the confidence intervals are difficult to calculate. On the other hand, the statistical estimates in the frequency domain are relatively stable and independent.

The power spectrum $S(\omega)$ of a function $x(\psi)$ is defined as the Fourier transform of its autocovariance function $\gamma(\psi)$, which is expressed as

$$\gamma(\psi) = \lim_{T \to \infty} \frac{1}{T} \int_T x(\tau) \, x(\psi + \tau) \, d\tau \tag{1.151}$$

or in case of discrete observations

$$\gamma(\psi) = \lim_{N \to \infty} \frac{1}{N - |r|} \sum_{\tau = 0}^{N-1-|r|} x(\tau) \, x(\psi + \tau), \qquad \begin{array}{l} \psi = r \cdot \Delta \psi \\ r = 0, 1, \ldots, (N-1) \end{array} \tag{1.152}$$

Then, the power spectrum can be written

$$S(\psi) = \int_{-\infty}^{\infty} \gamma(\psi) \, e^{-i\omega\psi} \, d\psi \tag{1.153}$$

Actually $S(\omega)$ and $\gamma(\psi)$ are the Fourier transform pairs usually shown as

$$\gamma(\psi) \longleftrightarrow S(\omega)$$

The inverse Fourier transform of eqn. (1.122) gives the autocovariance function

$$\gamma(\psi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) \, e^{i\omega\psi} \, d\psi \tag{1.154}$$

The power spectrum $S(\omega)$ can be directly expressed as a function of $x(\psi)$, (Rayner, 1971, p. 77),

$$S(\omega) = \lim_{T \to \infty} \frac{1}{T} \, |X(\omega)|^2 \tag{1.155}$$

The relation between eqns. (1.152) and (1.155) is known as the Wiener-Khintchine relation.

The eqn. (1.152) cannot be applied in its present form, because $\gamma(\psi)$ is not defined outside ψ in the case of observed data vector $x(\psi)$. In the derivation of eqn. (1.152) we implicitly assumed that $x(\psi)$ was zero for $|\psi| > T/2$, but it is not necessarily the case. Suppose ψ is in the range of $(-m, m)$ and let $C(\psi)$ be an estimate of the ensemble value of the true auto-covariance function $\gamma(\psi)$, i.e.

$$\text{Average } (C(\psi)) = \gamma(\psi) \tag{1.156}$$

then, we can make eqn. (1.156) apply to all values of ψ through windowing,

$$C'(\psi) = C(\psi) \ W(\psi)$$

where $W(\psi)$ is the window function. Therefore, eqn. (1.156) becomes

$$\text{Ave}[C'(\psi)] = \text{Ave}[C(\psi)] \cdot W(\psi) = \gamma(\psi) \cdot W(\psi)$$

and the frequency domain representation

$$\text{Ave}[S(\omega)] = S(\omega) * W(\omega) \tag{1.157}$$

where $S(\omega)$ is the spectrum (power) of the true autocovariance function of $x(\psi)$ (ψ) is the window through which $\gamma(\psi)$ is viewed, and $W(\omega)$ is the corresponding spectral window.

The major steps to compute the power spectrum from a series of N equally spaced observations can be stated as follows:

(1) Remove mean and non-desired effects of low frequencies (trends).
(2) Compute the unbiased sample autocovariance function $C(\psi)$ through

$$C(\psi) = \frac{1}{N - |r|} \sum_{k=0}^{N-1-|r|} x(k) \ x(k+\psi), \ \text{where } \begin{cases} \psi = r \cdot \Delta\psi \\ 0 \le r \le m \end{cases} \tag{1.158.a}$$

It is also common to use the biased sample autocovariance function of the following form,

$$C(\psi) = \frac{1}{N} \sum_{k=0}^{N-1-|r|} x(k) \ x(k+\psi) \tag{1.158.b}$$

(3) Apply the selected window to get the windowed autocovariance function

$$C'(\psi) = C(\psi) \ W(\psi)$$

(4) Transform $C'(\psi)$ into the frequency domain as follows

$$S'(f_n) = \sum_{r=-m}^{m} C'(r) \cos(2\pi f_n r \cdot \Delta\psi) \ \Delta\psi \tag{1.159}$$

where $\Delta\psi = \text{constant} * \Delta s$ the lag increment, and $\Delta s = $ observational

interval, $f_n = n/(2m \cdot \Delta\psi)$ with $n = 0, 1, \ldots, $, $(2m \cdot \Delta\psi)$ = the length of the basic interval.

Since $C'(r \cdot \Delta\psi)$ and the cosine function are even, it is possible to write

$$S'(f_k) = 4 \sum_{r=0}^{m} C'(r \cdot \Delta\psi) \cos(2\pi f_k r \cdot \Delta\psi) \Delta\psi \tag{1.160.a}$$

We have to make adjustments for the end data (recall the trapezoidal integration) (for details see Bath, 1974, pp. 171-173), therefore,

$$S'(f_k) = 2\Delta\psi [C'(0) + 2 \sum_{r=1}^{m-1} C'(r \cdot \Delta\psi) \cos\frac{\pi r f_k}{m} + C'(m \cdot \Delta\psi) \cos \pi f_k] \tag{1.160.b}$$

with the inverse transform (Otnes, 1972, p. 197)

$$C'(r \Delta\psi) = \frac{1}{4m \cdot \Delta\psi} [S'(0) + 2 \sum_{n=1}^{m-1} S'(f_n) \cos\frac{\pi n r}{m} + S'(m) \cos \pi r] \tag{1.160.c}$$

The eqn. (1.160.a) can also be expressed in terms of average variance per prequency band (Rayner, 1971, p. 81), by dividing it by the length of the basic interval, i.e. $(2m \cdot \Delta\psi)$

$$S'(f_n) = \frac{2}{m} \sum_{r=0}^{m} C'(r \cdot \Delta\psi) \cos 2\pi f_n r \Delta\psi \tag{1.160.d}$$

The relation between eqns. (1.160.a), the variance density estimate, and (1.160.d), the average per frequency band, are very important and they must be used in the right places.

Still further we have to make adjustments for the spectrum ends thus finally we obtain for the power spectrum

$$S'(0) = \frac{1}{2m} C'(0) + \frac{1}{m} \sum_{r=1}^{m-1} C'(r \cdot \Delta\psi) + \frac{1}{2m} C'(m \cdot \Delta\psi)$$

$$S'(f_n) = \frac{1}{m} C'(0) + \frac{2}{m} \sum_{r=1}^{m-1} C'(r \cdot \Delta\psi) \cos\frac{\pi n r}{m} + \frac{1}{m} C'(m \cdot \Delta\psi) \cos \pi n, \quad 0 < n < m$$

$$S'(m) = \frac{1}{2m} C'(0) + \frac{1}{m} \sum_{r=1}^{m-1} (-1)^r C'(r \cdot \Delta\psi) + \frac{(-1)^m}{(2m)} C'(m \cdot \Delta\psi) \tag{1.161}$$

(5) Convolve the raw powers above with $W(r)$, i.e. with the spectral window, to get $\overline{S}(r)$. As an example, suppose we decide to use Hanning window. Then there are two options: (1) we can multiply the data by window function and transform to the frequency domain, or (2) we can transform the data vector to the frequency domain, in other words, a rectangular window is used in the time domain, and then apply convolution in the frequency domain, which takes the following form for this particular example.

$$\overline{S}(0) = \tfrac{1}{2} S'(0) + \tfrac{1}{2} S'(1)$$
$$\overline{S}(r) = \tfrac{1}{4} S'(r-1) + \tfrac{1}{2} S'(r) + \tfrac{1}{4} S'(r+1)$$
$$\overline{S}(m) = \tfrac{1}{2} S'(m-1) + \tfrac{1}{2} S'(m) \tag{1.162}$$

The above estimates are averages for bands $1/(2m \cdot \Delta\psi)$ wide, except at $r = 0, m$ where they are $1/(4m \cdot \Delta\psi)$ wide centered at $\omega = \pi r/(m \cdot \Delta\psi)$.

(b) Using Direct Data Transform

In order to compute the power spectrum we can also transform the data directly. If $x(t)$ is a non-periodic function and only defined in the interval $(-T/2, T/2)$, then we have to use a window, say $w(t)$, to obtain

$$\hat{S}(f) = \frac{1}{T} \left| \int_{-T/2}^{T/2} x(t)\, w(t)\, e^{-i 2\pi f t}\, dt \right|^2$$

$$= \frac{1}{N} | X(f) * W(f) |^2 \tag{1.163}$$

where $f = \omega/2\pi$, and $N =$ the total number of observations. Now recall the following relationship

$$\hat{X}(f) = X(f) * W(f) = \hat{a}(f) - i\hat{b}(f) \tag{1.164}$$

in order to write

$$\hat{S}(f) = \frac{1}{N} [\hat{a}^2(f) + \hat{b}^2(f)], \qquad |f| < \frac{1}{2\Delta\psi} \tag{1.165}$$

Since eqn. (1.165) is an even function, for the one-sided spectrum we have

$$\hat{S}(f) = \frac{2}{N} [\hat{a}^2(f) + \hat{b}^2(f)], \qquad 0 \le f < \frac{1}{2\Delta\psi} \tag{1.166}$$

If we convert the power density estimates into average variance per frequency band, multiplying by $1/N$, and substitute eqns. (1.114.a-b) above we obtain

$$\hat{S}(k) = \tfrac{1}{2} [\hat{a}^2(k) + \hat{b}^2(k)] \tag{1.167}$$

This equation is equivalent to the expression for the degree variance of the periodic function. But here we have used windowed data and the estimate above is an average for a band centered at "k" rather than a discrete estimate at k.

We see that $\hat{S}(k)$ contains $(N/2)$ full bands and covers the same range of frequencies from zero to f_N (Nyquist frequency) as $\hat{S}(r)$, which contains $\underline{m}$ full bands. So we can make $\hat{S}(k)$ equivalent to $\hat{S}(r)$ by summing over $(N/2m)$ bands.

Now we can give the steps for the estimation of the power spectrum from a set of observations, $x(t)$, as follows:

(1) Remove mean and trends.

(2) Apply the window function $w(t)$, e.g. use a cosine-tapered rectangular window given by eqn. (1.142.a), thus

$$\hat{x}(t) = x(t) \cdot w(t) \tag{1.168}$$

(3) Compute the Fourier coefficients $\hat{a}(f)$, $\hat{b}(f)$ through eqns. (1.110) and (1.111) or the coefficients $\hat{a}(k)$, $\hat{b}(k)$ by eqns. (1.55.a-b), which are related to $\hat{a}(f)$ and $\hat{b}(f)$ through eqns. (1.114.a-b), using FFT in the calculations.
(4) Calculate the power spectrum $\hat{S}(k)$ by eqn. (1.167) and sum over blocks, say 5 to 50 (Rayner, 1971, p. 83), to obtain $\bar{S}(r)$. If we assume N is even such that $N/2m = 2z + 1$, then

$$\bar{S}(0) = \left(\frac{\hat{a}_0}{2}\right)^2 + \sum_{k=1}^{z} [\hat{a}^2(k) + \hat{b}^2(k)]/2$$

$$\bar{S}(r) = \sum_{k=r(2z+1)-z}^{r(2z+1)+z} [\hat{a}^2(k) + \hat{b}^2(k)]/2, \qquad 0 < r < m \tag{1.169}$$

$$\bar{S}(m) = \sum_{k=N/2-z}^{N/2-1} (\hat{a}^2(k) + \hat{b}^2(k)]/2 + \left(\frac{\hat{a}_{N/2}}{2}\right)^2$$

frequency bands are centered at

$$0, \ 1/2m \cdot t, \ 2/2m \cdot t, \ldots, \ r/2m \cdot t, \ldots, \ 1/2t$$

(c) Using the Filtering Method

This method will not be discussed here. In this method data are passed through a band-pass filter, squared, and then summed with a final normalization. For this method, readers are referred to Otnes (1972, pp. 311-315).

1.7.5 The Confidence Intervals

The chi-square distribution is given by the formula below,

$$f(\chi^2) = \frac{(\chi^2)^{\nu/2-1} \cdot e^{-\chi^2/2}}{2^{\nu/2} \ \Gamma(\nu/2)} \tag{1.170}$$

where $\Gamma(\nu/2)$ is the gamma function, and ν = degrees of freedom.

It is well known that

$$\left(\frac{DF * \text{sample variance}}{\text{true variance}}\right)$$

has χ^2 distribution, DF being degrees of freedom, i.e. in other words,

$$\nu \cdot \hat{\sigma}^2 / \sigma^2 = \chi^2(\nu) \tag{1.171}$$

By rearranging this equation we can write

-33-

$$\frac{\upsilon \cdot \hat{\sigma}^2}{\chi^2 [\upsilon, \text{ smaller probability level}]} \leq \sigma^2 \leq \frac{\upsilon \cdot \hat{\sigma}^2}{\chi^2 [\upsilon, \text{ larger probability level}]} \quad (1.172)$$

Now let us see the application of the χ^2 distribution in case of spectral analysis, by going back to the spectrum of the sample variance

$$\hat{\sigma}^2(r) = \bar{S}(r) = \sum_{k=r(2z+1)-z}^{r(2z+1)+z} [\hat{a}^2(k) + \hat{b}^2(k)]/2 \quad (1.173)$$

Since $\hat{a}(k)$ and $\hat{b}(k)$ are defined from an orthogonal transformation and are generally assumed to be normally distributed, $\hat{\sigma}^2(r)$ of eqn. (1.173) has $2(2z+1)$ degrees of freedom in the interval $0 < r < m$, and $2(z+1)$ degrees of freedom at $r = 0, m$.

But using a window reduces the degrees of freedom (Tukey, 1967) and must be considered in computations, e.g. if we use a cosine-tapered rectangular window, then

$$DF = \upsilon = 2(2z+1)(N-G)/N \quad (1.174)$$

where G is the percentage of tapering, and N is the number of total observations.

1.7.6 Cross Spectral Analysis and Coherence

1.7.6.1 Cross Spectral Analysis

Consider the following linear regression,

$$y = ax + b$$

where y, a, b, x are functions of "t". Let y also be a function of all the x's in the region, then we can write

$$y(t) = b(t) + \int_{-\infty}^{\infty} a(q) \cdot x(t-q) \, dq \quad (1.175)$$

If $x(t)$ and $b(t)$ are uncorrelated, then by eqn. (1.175) we obtain

$$C_{xy}(\psi) = \int_{-\infty}^{\infty} a(q) \cdot C_{xx}(\psi-q) \, dq \quad (1.176)$$

with the frequency domain representation

$$S_{xy}(f) = A(f) \cdot S_{xx}(f) \quad (1.177)$$

and

$$A(f) = S_{xy}(f)/S_{xx}(f) \quad (1.178)$$

$A(f)$ is known as the response function of the system.

Recall the definition of the correlation in the time domain, i.e.

-39-

$$\rho_{xy} = (\Sigma\, xy/N)/[\,(\Sigma\, x^2/N)^{\frac{1}{2}}(\Sigma\, y^2/N)^{\frac{1}{2}}\,] \tag{1.179}$$

The corresponding correlation in the frequency domain is defined as

$$R_{xy}(f) = S_{xy}(f)/[\,(S_{xx}(f))^{\frac{1}{2}}(S_{yy}(f))^{\frac{1}{2}}\,] \tag{1.180}$$

which is called "the coherence or coherency" and varies between $(-1, 1)$. In the equation above, $S_{xy}(f)$ is the Fourier transform of the cross-correlation function $C_{xy}(\psi)$ and defined as follows

$$S_{xy}(f) = \int_{-\infty}^{\infty} C_{xy}(\psi) \cdot e^{-i2\pi f \psi}\, d\psi, \quad -\infty < f < \infty \tag{1.181}$$

or equivalently,

$$S_{xy}(f) = \lim_{T \to \infty} \frac{1}{T}\, Y(f) \cdot X^*(f) \tag{1.182.a}$$

and for one-sided spectrum we have

$$S_{xy}(f) = 2\int_{-\infty}^{\infty} C_{xy}(\psi) \cdot e^{-i2\pi f \psi} d\psi = \lim_{T \to \infty} \frac{2}{T}\, Y(f) \cdot X^*(f) \tag{1.182.b}$$

where $Y(f)$ and $X(f)$ are defined as in eqn. (1.108).

For discrete and finite data the eqn. (1.182.a-b) can be written (Rayner, 1971, p. 82)

$$S_{xy}(f) = \frac{1}{N}\hat{Y}(f) \cdot \hat{X}^*(f) = \frac{1}{N}[\,Y'(f) * H(f)\,][\,X'(f) * H(f)\,]^*, \quad -\infty < f < \infty \tag{1.183}$$

and for the one-sided cross spectrum we write

$$\hat{S}_{xy}(f) = \frac{2}{N}\hat{Y}(f) \cdot \hat{X}^*(f) = \frac{2}{N}[\,Y'(f) * H(f)\,][\,X'(f) * H(f)\,] \tag{1.184}$$

If cross-covariance density estimates are converted into average cross spectral density (CSD) per frequency band, then we have

$$\hat{S}_{xy}(k) = \frac{1}{N}\left\{\frac{2}{N}\left[\frac{N}{2}\hat{a}_y(k) - i\frac{N}{2}\hat{b}_y(k)\right]\left[\frac{N}{2}\hat{a}_x(k) + i\frac{N}{2}\hat{b}_x(k)\right]\right\}$$

$$= \tfrac{1}{2}[\,\hat{a}_y(k) - i\hat{b}_y(k)\,][\,\hat{a}_x(k) + i\hat{b}_x(k)\,] \tag{1.185}$$

The real parts of eqns. (1.183) and (1.185) are called cospectra, and imaginary parts of those are called quadrature spectra, that is to say

$$\hat{S}_{xy}(f) = \hat{R}_{xy}(f) - i\hat{Q}_{xy}(f)$$

or equivalently, $\tag{1.186}$

$$\hat{S}_{xy}(k) = \hat{R}_{xy}(k) - i\hat{Q}_{xy}(k)$$

where $\hat{R}_{xy}(f)$ and $\hat{R}_{xy}(k)$ are cospectra, and $\hat{Q}_{xy}(f)$ and $\hat{Q}_{xy}(k)$ are quadrature spectra.

We can express the cross-covariances between x and y (assuming means are zero)

(a) in continuous case:

$$C_{xy}(\psi) = \lim_{T \to \infty} \frac{1}{T} \int_{-\infty}^{\infty} y(t) \cdot x(t + \psi) \, dt \qquad (1.187)$$

(b) in discrete case:

$$C'_{xy}(\psi) = \frac{1}{N - |r|} \sum_{t = -N + |r|}^{N - |r|} y(t) \cdot x(t + \psi), \qquad \begin{array}{l} \psi = r \cdot \Delta\psi \\ |r| \le m \end{array} \qquad (1.188)$$

Notice that eqns. (1.187) and (1.188) are not even, therefore, we break them into even and odd series defined as follows:

$$C'_{xy_E}(\psi) = \tfrac{1}{2} [C'_{xy}(\psi) + C'_{xy}(-\psi)] \qquad (1.189)$$

$$C'_{xy_O}(\psi) = \tfrac{1}{2} [C'_{xy}(\psi) - C'_{xy}(-\psi)] \qquad (1.190)$$

Eqn. (1.189) is transformed into frequency domain through eqn. (1.160) and (1.162) to get $S'_{xy_E}(r)$ which is known as "the cospectrum of $y(t)$ and $x(t)$", and defined by

$$S'_{xy_E}(r) = 2 \Delta\psi [C'_{xy_E}(0) + 2 \sum_{k=1}^{m-1} C'_{xy}(r) \cos \frac{\pi r k}{m} + C'_{xy_E}(m) \cos \pi r] \qquad (1.191.a)$$

The power of the eqn. (1.190) is computed similarly,

$$S'_{xy_O}(r) = \frac{2}{m} \sum_{\psi=1}^{m-1} C'_{xy_O}(r) \sin r \pi \psi / m \qquad 0 < r < m$$

$$= 2\Delta\psi [C'_{xy_O}(0) + 2 \sum_{k=1}^{m-1} C'_{xy_O}(k) \sin \pi r k / m] \qquad (1.191.b)$$

$S'_{xy_O}(r)$ is known as "the quadrature spectrum of $y(t)$ and $x(t)$". Eqns. (1.191. a-b) must be smoothed for the finite length to yield $\overline{S}_{xy_E}(r)$ and $\overline{S}_{xy_O}(r)$. Finally we can write for the cross-covariance

$$|\overline{S}_{xy}(r)| = [\overline{S}_{xy_E}^2(r) + \overline{S}_{xy_O}^2(r)]^{\frac{1}{2}} \qquad (1.192)$$

and for the phase

$$\overline{\Phi}_{xy}(r) = \tan^{-1} [\overline{S}_{xy_O}(r) / \overline{S}_{xy_E}(r)] \qquad (1.193)$$

Now, we may summarize the main steps for the calculation of the cross spectrum:

(1) Remove mean and trends.
(2) Apply window function so that $\hat{x}(t) = x'(t) \cdot h(t)$, $\hat{y}(t) = y'(t) \cdot h(t)$ with the Fourier transforms $\hat{X}(f) = \hat{X}'(f) * H(f)$, $\hat{Y}(f) = \hat{Y}'(f)$

-41-

(3) Compute the Fourier coefficients $\hat{a}_x(f)$, $\hat{b}_x(f)$, $\hat{a}_y(f)$, $\hat{b}_y(f)$ from eqns. (1.110) and (1.111), or better the coefficients $\hat{a}_x(k)$, $\hat{b}_x(k)$, $\hat{a}_y(k)$, $\hat{b}_y(k)$) from eqns. (1.55.a-b) using FFT.

(4) Compute the cospectrum and quad-spectrum by rearranging eqn. (1.183) into real and imaginary parts to get

$$\hat{S}_{xy_{\ell}}(k) = [\hat{a}_x(k)\ \hat{a}_y(k) + \hat{b}_x(k)\ \hat{b}_y(k)]/2$$

$$\hat{S}_{xy_0}(k) = [\hat{a}_x(k)\ \hat{b}_y(k) - \hat{a}_y(k)\ \hat{b}_x(k)]/2$$

(1.194)

(5) Sum the results above over $(2z + 1)$ elementary bands to obtain

and

$$\bar{S}_{xy_{\ell}}(r) = \sum_{k=r(2z+1)-z}^{r(2z+1)+z} [\hat{a}_x(k)\ \hat{a}_y(k) + \hat{b}_x(k)\ \hat{b}_y(k)]/2$$

$$\bar{S}_{xy_0}(r) = \sum_{k=r(2z+1)-z}^{r(2z+1)+z} [\hat{a}_x(k)\ \hat{b}_y(k) - \hat{a}_y(k)\ \hat{b}_x(k)]/2$$

(1.195)

(6) Compute the cross spectrum $|\bar{S}_{xy}(r)|$ and the phase $\bar{\theta}_{xy}(r)$ from eqns. (1.192) and (1.193) respectively.

1.7.6.2 Coherence

The following expression

$$\text{arc tan h}\,[\,|\bar{S}_{xy}(r)|\,] = \frac{1}{2}\,\ell n\left[\frac{1+|\bar{S}_{xy}(r)|}{1-|\bar{S}_{xy}(r)|}\right]$$

(1.196)

is normally distributed with a variance approximately equal to $(1/DF)$ (Rayner, 1971, p. 98). Then the approximate limits of eqn. (1.196) are given by

$$\text{arc tan h}\,[\,|\bar{S}_{xy}(r)|\,] \pm \omega_\alpha\left(\frac{1}{\upsilon}\right)^{\frac{1}{2}}$$

(1.197)

where α = significance level, and ω_α = the expected limit, and ν = degrees of freedom.

1.8. Summary

The computation of a spectrum from continuous and discrete data and spectral analysis has been explained in this chapter. In geodetic applications we generally have discrete and finite-length data. The selection of finite-length data causes undesirable effects on the spectrum known as the spectral leakage (Harris, 1978, pp. 51-52). In order to minimize the spectral leakage windows are applied as explained in this chapter. We believe that this chapter is a good reference for geodesists, who use discrete and finite-length data.

2. Spectral Analysis of GEOS-3 Altimeter Data

2.1 Satellite Altimetry

The determination of the gravity field and the shape of the earth has been the main goal of Geodesy for centuries. If we were able to measure gravity over the entire earth, then a considerable number of geodetic problems could be solved. However, in practice, we do not have enough observations in many parts of the earth, mainly in ocean areas.

Recently the Geodynamics Experimental Ocean Satellite (GEOS-3) observations have made it possible to determine gravity anomalies from altimeter height observations. By satellite altimetry we hope to measure the spectrum of the sea surface to a very high frequency [λ (wavelength) < 1000 km] so that we can determine small scale changes in addition to the Earth's gravity field, luni-solar effects, atmospheric tides etc. In fact, oceanographers would like to know sea topography at the 10 cm level from the mean sea level in order to detect ocean currents.

The geometry of an altimeter borne satellite with respect to the earth's center is illustrated in Figure 2.1. Here we simply assume that the line OA connecting the geocenter and the satellite is perpendicular to the instantaneous sea surface (ISS) and the mean sea surface (MSS). The altimeter measures the distance (h' = AP') between the satellite and the satellite's footprint of ISS. We can compute the coordinates of the satellite through the observations made at the tracking stations distributed in a world-wide network. Hence we can derive the geodetic (φ, λ) or Cartesian (X, Y, Z) coordinates of the satellite's footprint with respect to a reference ellipsoid as well as its distance (r' = OA) from the center of this ellipsoid, and consequently the height (h = AQ) of the altimeter above the ellipsoid. The sea surface height (SSH), which is the separation between the reference ellipsoid and ISS, is computed as follows (Rapp, 1977, p. 2)

$$SSH = h - (h' - R + b) \tag{2.1}$$

where R is the refraction correction
b is the a priori altimeter bias (if any)
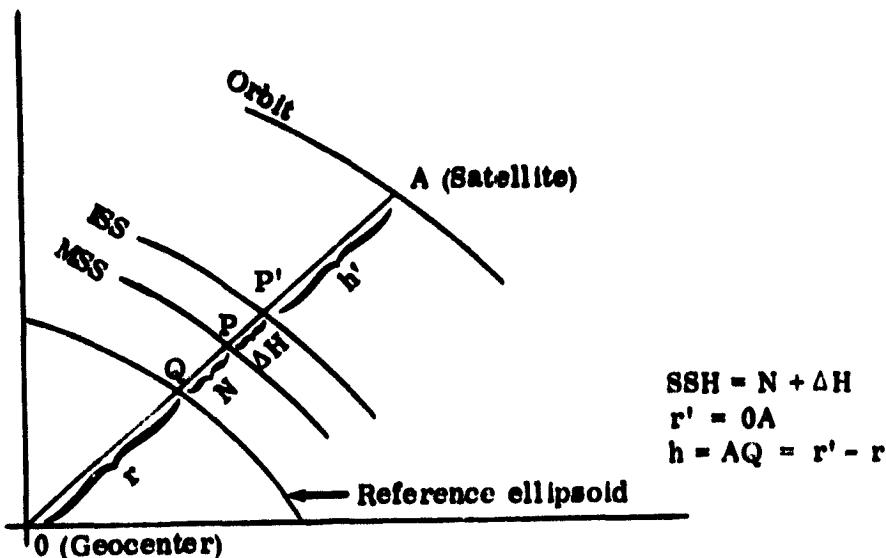
Figure 2.1. The Geometry of Satellite Altimetry.

The geoid is usually defined as the equipotential surface coinciding with MSS. The distance (N) between the reference ellipsoid and MSS is known as the geoid height or undulation. There exists a separation (ΔH) between SS and MSS due mainly to tides, winds, storms, currents, etc. Therefore, SSH should be corrected for the factors mentioned above in order to obtain the geoid height (N) Hence we have

$$N = h - (h' - R + b) - \Delta H \qquad (2.2)$$

Eqn. (2.2) is nothing else but an observation equation, the undulation N being considered as the observation. For one or more arcs Rapp (Ibid, p. 12) uses the following mathematical model for the adjustment of geoid heights:

$$\underline{N} + \underline{B}\,\underline{\Delta} = \underline{N}_R + \underline{V} \qquad (2.3)$$

where $\underline{N}$; column vector of geoid heights implied by altimeter data

$\underline{B}$; the design matrix of error model parameters

$\underline{\Delta}$; column vector of parameters of the error model

$\underline{N}_R$; column vector of geoid heights implied by the reference surface (in our case GEM 9 surface to degree 20)

$\underline{V}$; column vector of residuals.

For the adjustment purpose a data set of 419294 geoid heights implied by GEOS-3 altimeter data, in 2003 arc segments, was considered. First a primary adjustment was carried out, followed by regional adjustments. The details of the adjustment are described in Rapp (Ibid, pp. 1-24).

-44-

2.2 Gravity Anomaly Recovery

As we mentioned at the beginning of this chapter, we are mainly interested in the determination of point and mean gravity anomalies from the GEOS-3 altimeter data. There are various techniques for the anomaly recovery. Rummel, Sjöberg and Rapp (1977) have used (1) the inverse Stokes' equation or the Molodenskii equation (by Rummel); (2) the spectral analysis approach (by Sjöberg); (3) the least-squares collocation (by Rapp). In addition to these techniques, two new methods will be introduced in this paper (1) the least-squares collocation using Toeplitz matrices; (2) the frequency domain least-squares collocation, in the third and fourth chapter respectively.

In the next sections we will examine the power spectrum of the adjusted GEOS-3 altimeter data and predicted point anomalies along various arc segments. Point anomalies have been computed from altimeter data by using least-squares collocation (Moritz, 1975) as follows:

$$\underline{\Delta g} = \underline{C}_{\Delta g N} \left(\underline{C}_{NN} + \underline{D} \right)^{-1} \underline{N} \tag{2.4}$$

where $\underline{\Delta g}$; predicted point anomalies
$\underline{N}$; vector of given geoid heights
$\underline{C}_{\Delta g N}$; row-vector of the covariance between the anomaly predicted and the geoid heights
$\underline{C}_{NN}$; square and symmetric covariance matrix of the given geoid heights
$\underline{D}$; the noise matrix of the given geoid heights (taken diagonal).

The point anomalies predicted as described above do not reflect the contributions of the high-frequency components to the variance (total average power) due to the following facts;

(1) Altimeter observations used in the adjustment procedure described above represent averages of two seconds in the low data rate, imposing a 28-km limit on the shortest wavelength of information.

(2) We use a global covariance model ($\underline{C}_{\Delta g N}$ and $\underline{C}_{NN}$ above), which acts like a low-pass filter, in the least-squares collocation.

The spectral analysis approach will enable us to point out the contribution of every existing frequency or wavelength component to the variance (total average power). Then we can make a reasonable conclusion about the resolution of altimeter data and predicted anomalies.

2.3 Power Spectrum from Geoid Heights

In this and following sections we will examine the breakdown of variance with respect to the frequencies (or wavelengths) and the contribution of each frequency component to the total average power. The power spectra of some ten

profiles (numbered one to ten) shown in Figure 2.2 have been considered in this section. From the adjusted GEOS-3 altimeter data we have computed geoid heights along these profiles at the intersections of a latitude-longitude graticule of one degree or arc (Rapp, 1979). Each profile has been considered independently for the removal of the mean value along the particular profile from computed undulations and for the computation of the Fourier transform, which was carried out by FFT algorithms of IMSL libraries at OSU, by eqn. (1.62), i.e.

$$X_n = \frac{1}{N}\sum_{k=0}^{N-1} x(k \cdot \Delta t)\, e^{-i 2\pi k n / N}, \qquad \begin{array}{l} n = 0, 1, \ldots, N_H \\ N_H = \begin{cases} N/2 & \text{if } N \text{ even} \\ (N-1)/2 & \text{if } N \text{ odd} \end{cases} \end{array} \qquad (2.5)$$

where X_n ; the complex Fourier coefficients
  $x(k \cdot \Delta t)$; the geoid height at the $(k \cdot \Delta t)$th point after the removal of the mean value along the profile
  N ; the number of undulations along the profile.

In section 1.6 we have defined the total average power (variance) of a function $x(k \cdot \Delta t)$; $k = 0, 1, \ldots, (N-1)$ as

$$P_{AVG}^2 = \frac{1}{N}\sum_{k=0}^{N-1} [x(k \cdot \Delta t)]^2 \qquad (2.6)$$

in the time domain and by Parseval's theorem as

$$P_{AVG}^2 = \sum_{k=0}^{N_H} P_n^2 \qquad (2.7)$$

where

$$P_n^2 = \begin{cases} X_0^2 & \text{for } n = 0 \\ 2|X_n|^2 & \text{for } n = 1, 2, \ldots, (N_H - 1) \\ X_{N_H}^2 & \text{for } n = N_H \text{ and } N \text{ even} \\ 2|X_{N_H}|^2 & \text{for } n = N_H \text{ and } N \text{ odd} \end{cases} \qquad (2.8)$$

in the frequency domain. P_n^2 , $n = 0, 1, \ldots, N_H$ is known as the nth degree power of the periodic function $x(k \cdot \Delta t)$. $P_0 = X_0$ is the average value (also called dc value) of the function $x(k \cdot \Delta t)$. Since we have removed mean values before transformations, they are equal to zero in all our computations. Even if there exists a mean value during the transformation, the power of zeroth degree is excluded from the variance to obtain

$$\hat{P}_{AVG}^2 = P_{AVG}^2 - P_0^2 = \sum_{n=1}^{N_H} P_n^2 \qquad (2.9)$$

The contribution of the nth degree power to the variance is given by

$$C_n = P_n^2 / \hat{P}_{AVG}^2 \qquad (2.10)$$

and the power implied by frequencies up to $M \leq N_H$ (from now on we will call Mth cumulative power) is given by

GEOS 3 GROUND TRACKS
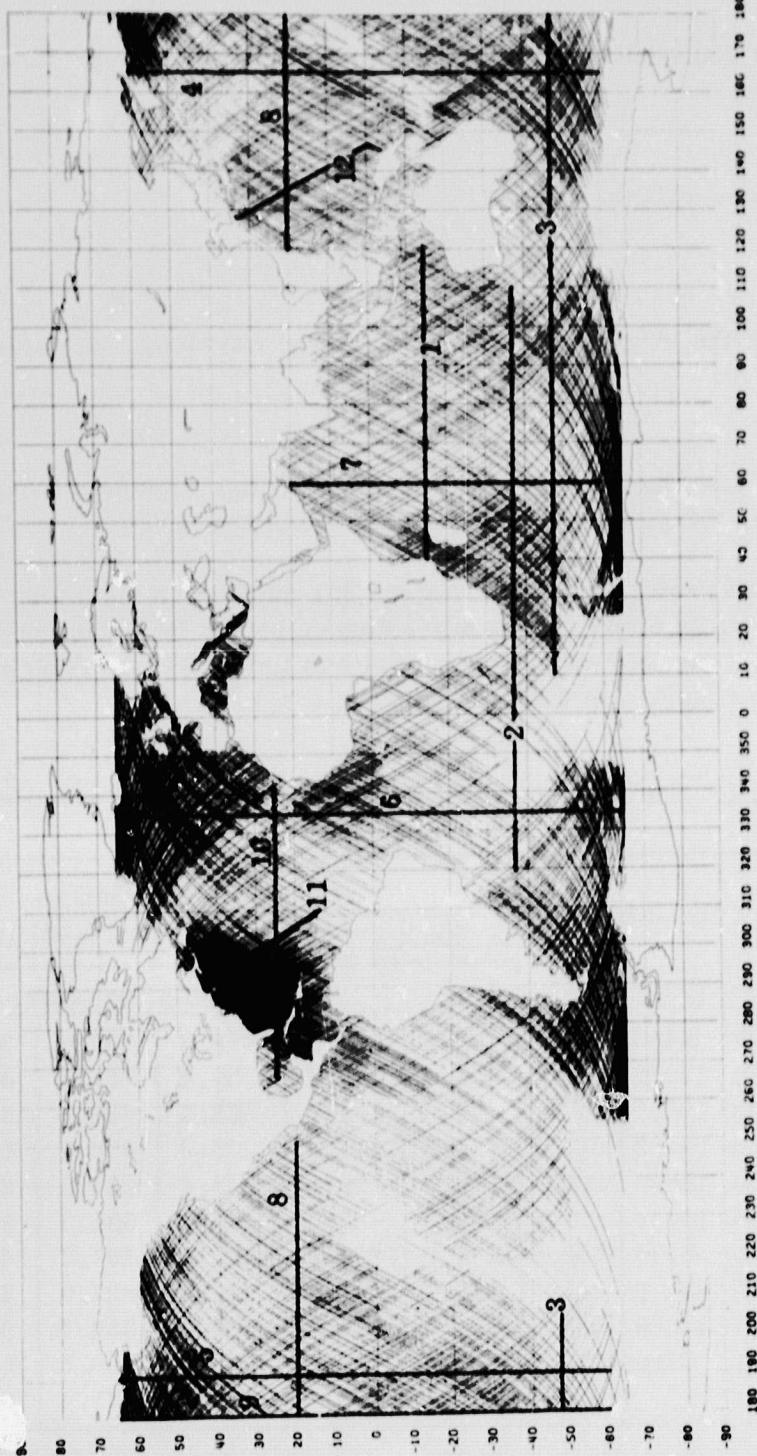
Figure 2.2. GEOS-3 Ground Tracks and Some Selected Sub-Profiles.

$$S_M = \sum_{n=1}^{M} P_n^2 \qquad (2.11.a)$$

finally, the contribution of the Mth-cumulative power is given by

$$R_M = S_M / \hat{P}_{AVG}^2 \qquad (2.11.b)$$

We know that the frequency f_n is defined as

$$f_n = n/S \qquad (2.12.a)$$

where S is the length of the profile, and the wavelength is defined as

$$\lambda_n = S/n \qquad (2.12.b)$$

Hence P_n^2 can also be called the power of nth-wavelength component, and we can consider the wavelength as the variable of the power.

The ten profiles shown in Figure 2.1 have been transformed into the frequency domain in order to obtain the degree power P_n^2, its contribution c_n to the variance, the Mth-cumulative power to the variance. The profiles and the computation results are described in Table 2.1.

Table 2.1. The Breakdown of Variance According to Wavelengths.

Profile No.	φ_1, φ_2 (Deg.)	λ_1, λ_2 (Deg.)	Length (km)	P_{AVG}^2 (m^2)	$\lambda>6000$	$\lambda>3000$	$\lambda>2000$	$\lambda>1500$	$\lambda>1000$	$\lambda>500$	$\lambda>200$
					Wavelength						
					S, cumulative power in m^2						
1	-15, -15	40, 120	8485	342.8	294.8	318.1	328.3	331.9	337.5	339.7	342.8
2	-37, -37	320, 110	13232	367.6	330.4	357.1	359.4	361.1	363.8	365.9	367.4
3	-48, -48	10, 208	14658	619.2	577.9	601.4	608.4	610.3	614.0	616.3	618.6
4	-60, 60	165, 165	13232	678.2	603.4	641.4	652.3	658.7	665.8	673.5	678.2
5	-60, 65	190, 190	13899	269.9	160.8	236.8	245.6	254.1	258.0	265.1	269.4
6	-65, 65	335, 335	14344	521.8	437.3	487.5	500.3	504.7	508.8	516.2	521.8
7	-60, 20	60, 60	8784	1520.6	1295.8	1399.3	1451.4	1463.7	1485.6	1505.4	1520.6
8	20, 20	120, 250	13479	841.3	756.2	801.5	816.8	827.9	830.1	836.3	841.3
9	-60, 65	180, 180	13899	562.7	458.9	520.2	534.2	544.4	550.9	558.0	562.7
10	25, 25	265, 341	7760	713.2	605.8	618.8	642.9	659.6	682.0	700.1	713.2
Mean				643.7	552.1	598.2	614.0	621.1	629.7	637.7	643.7

From the table above we see that the long-wavelength components of undulations contribute the greater part of the variance (power) P_{AVG}^2, e.g. $\lambda>6000$, $\lambda>3000$, $\lambda>2000$, $\lambda>1000$, $\lambda>500$, contribute 86%, 93%, 95%, 98%, 99%, of the variance respectively.

2.4 Power Spectrum of Geoid Heights with Respect to the GEM 9 Reference Surface

In the adjustment of GEOS-3 altimeter data a reference surface implied by GEM 9 potential coefficients to degree and order 20 has been used. Therefore, we have decided to examine the power spectrum of geoid heights with respect to the GEM 9 reference surface, say ΔN. In order to compute ΔN first we have to compute reference surface, i.e. GEM 9 surface, undulations and then subtract it from adjusted altimeter undulation at the same point. The GEM 9 surface undulations are computed as follows (Rapp, 1977):

$$N_R = \frac{GM}{\gamma r} \sum_{\ell=2}^{20} \left(\frac{a}{r}\right)^\ell \sum_{m=0}^{\ell} (\overline{C}_{\ell m} \cos m\lambda + \overline{S}_{\ell m} \sin m\lambda)\, \overline{P}_{\ell m}(\sin \overline{\varphi}) \qquad (2.13)$$

where

GM	;	the geocentric gravitational constant
r	;	the geocentric distance to the computation point
a	;	an equatorial radius
$\overline{C}_{\ell m}, \overline{S}_{\ell m}$	;	fully normalized potential coefficients
$\overline{P}_{\ell m}$	;	fully normalized Legendre functions
$\overline{\varphi}, \lambda$	;	the geocentric latitude and longitude
γ	;	the normal gravity at $\overline{\varphi}$.

Thus we can write for the undulations with respect to the GEM 9 surface

$$\Delta N = N - N_R \qquad (2.14)$$

where N is the adjusted GEOS-3 geoid height at which N_R is being computed.

Seven arcs of about 13000 km length have been selected for the spectral analysis. These arcs are described in Table 2.2.

Table 2.2. Arcs 2, 3, 4, 5, 6, 8, 9 and Statistics.

Arc No. (See Fig. 2.2)	Latitude φ_1, φ_2 (Deg.)	Longitude λ_1, λ_2 (Deg.)	Length (km)	Mean ($\overline{\Delta N}$) (m)	RMS
2	−37, −37	320, 107	13054	0.3	1.6
3	−48, −48	10, 186	13021	−0.6	2.3
4	−60, 57	165, 165	13010	−0.5	2.1
5	−60, 57	190, 190	13010	−0.6	1.9
6	−65, 52	335, 335	13010	0.7	2.6
8	20, 20	120, 245	13061	0.9	2.9
9	−60, 57	180, 180	13010	−0.4	2.6

2.4.1 Power Spectrum when Data are Assumed to be Periodic

If we assume data are periodic, i.e. data repeat themselves indefinitely, then we will have a discrete spectrum made up of a finite number of frequencies. The geoid heights given with respect to the GEM 9 reference surface, which are defined in eqn. (2.14), have been transformed into the frequency domain using FFT algorithms mentioned before. Finally the breakdown of variance, i.e. the degree power, has been computed by eqn. (2.6). The results are described in Table 2.3.

Table 2.3. The Breakdown of Variance and Degree Powers with Reference to GEM 9 Surface from Periodic Data.

Wavenumber or Frequency	Corresponding Global Degree	Wavelength (km)	Arc No.							
			2	3	4	5	6	8	9	Mean
			Degree Power (m ** 2)							
1	(3)	13020	0.17	0.99	0.20	0.13	0.55	0.18	0.11	0.33
2	(6)	6510	0.44	0.22	0.03	0.76	0.44	1.86	0.54	0.61
3	(9)	4340	0.17	0.30	0.90	0.54	0.80	1.31	0.19	0.60
4	(12)	3255	0.19	0.52	0.24	..16	1.46	0.54	0.95	0.58
5	(15)	2604	0.33	1.78	0.51	0.10	0.41	0.14	0.15	0.49
6	(18)	2170	0.08	0.15	0.05	0.06	0.03	0.48	0.18	0.15
7	(21)	1860	0.29	0.01	0.49	0.00	0.69	0.01	0.52	0.29
.	.	.	+	+	+	+	+	+	+	+
.			1.67	3.97	2.42	1.75	4.38	4.52	2.64	3.05
.		.	.	.	.	.	.	.	.	.
.		.	.	.	.	.	.	.	.	.
$\hat{P}^2_{AVG}$			2.70	5.21	4.49	3.71	6.76	8.48	6.81	5.52

From Table 2.3 we see that the mean of variances of geoid heights is 5.52 m^2. But there is an uncertainty on each given geoid height, which is about ± 1 m in our case. Then under the assumption of white noise, i.e. no correlation, we can compute the reduced power as follows (Wagner, 1977, p. 14)

$$\overline{P}^2 \text{ (reduced)} = P^2 \text{ (measured)} - P^2 \text{ (noise)} \qquad (2.15)$$
$$\overline{P}^2 \text{ (reduced)} = 5.52 - 1.00 = 4.42$$

and finally $\overline{P} = 2.1$ m.

So we have recovered about 2.1 m undulation information in addition to the GEM 9 undulations. From Table 2.3 we also see that $\lambda > 2000$ km contribute about 1.6 m, and $\lambda < 2000$ km contribute about 1.3 m of this extra 2.1 m undulation information.

The results given in Table 2.3 also yield a global power for harmonic degrees

-50-

$l = 3, 6, 9, \ldots$. However, we have to keep in mind that we assumed periodicity of data and used only seven profiles, which are about $1/3$ of a great circle around the earth. So in the next section we will consider non-periodic data and derive a global power spectrum.

2.4.2 Power Spectrum from Non-Periodic Data

In this section we are after a global degree power using GEOS-3 altimeter data. However, we do not have any complete altimeter data around the earth. So we assume that the power spectrum of each profile, which is a part of a great circle around the earth, is an estimate of the global power spectrum after scaling the frequencies and power. Since the variance of a sub-profile is a good estimate for the global variance, the degree power of each sub-profile contains about (Wagner 1977, pp. 89-93)

$$C_i = S \quad \text{(length of great circle)}/S_i \quad \text{(length of sub-profile)} \qquad (2.15)$$

times the global degree power of the equivalent global frequency.

Since the data of a great circle around the earth are periodic we can use a window to obtain a periodic function, say $y(n)$, $n = 1, 2, \ldots, N_{max}$, from a non-periodic function, $x(n)$, $n = 1, 2, \ldots, N < N_{max}$. We can explain the above procedure by considering the example above. We are given 120 observations defined as

$$x(k \cdot \Delta\psi), \quad k = 0, 1, \ldots, 119$$
$$\text{and}$$
$$\Delta\psi = 1°$$

Now let $y(n \cdot \Delta\psi)$ be defined as

$$y(n \cdot \Delta\psi) = w(n) \cdot x(n \cdot \Delta\psi), \qquad (2.16)$$

$n = -120, -119, \ldots, 0, \ldots, 119, 120, \ldots, 240$

such that

$$w(n) = \begin{cases} 1 & \text{for } 0 \leq n \leq 119 \\ 0 & \text{otherwise} \end{cases}$$

then, $y(n \cdot \Delta\psi)$ is defined along the whole great circle. So $y(n \cdot \Delta\psi)$ is periodic and it can be transformed into the frequency domain and interpreted as global. However, in order to maintain the variance (or average power) of $y(n \cdot \Delta\psi)$ equal to the variance of $x(k \cdot \Delta\psi)$, the degree powers of $y(n \cdot \Delta\psi)$ should be multiplied by C_1 defined by eqn. (2.15), which will be explained later.

In order to compute a global power spectrum we have selected some four sub-profiles with observations one degree apart, described in Table 2.4.

Table 2.4. Arcs 4, 5, 6, 9 and Statistics.

Arc No. (See Fig. 2.2)	Latitude φ_1, φ_2 (Deg.)	Longitude λ_1, λ_2 (Deg.)	Length (km)	Mean $(\overline{\Delta N})$ (m)	RMS
4	-60, 60	165, 165	13343	-0.5	2.0
5	-60, 65	190, 190	13899	-0.4	2.1
6	-65, 65	335, 335	14455	0.6	2.5
9	-60, 65	180, 180	13899	-0.3	2.5

The global power spectrum of each sub-profile of Table 2.4 has been computed as follows:

(1) The mean of each arc was removed from the given data to obtain

$$\hat{x}(k \cdot \Delta\psi) = x(k \cdot \Delta\psi) - \frac{1}{N} \sum_{i=0}^{N-1} x(k \cdot \Delta\psi), \quad k = 0, 1, \ldots, (N-1)$$

where N is the total number of observations along the particular arc and $\Delta\psi = 1°$ in our examples.

(2) $\hat{x}(k \cdot \Delta\psi)$ has been multiplied by a cosine-tapered window, which is explained in section 1.7.3.1.5, to obtain

$$y(k \cdot \Delta\psi) = c(k \cdot \Delta\psi) \cdot x(k \cdot \Delta\psi), \quad -\left(\frac{C-1}{2} N\right) \leq k \leq \left(\frac{C+1}{2} N\right) \qquad (2.17)$$

where

$$c(k \cdot \Delta\psi) = \begin{cases} \frac{1}{2}[1-\cos(25\pi k/N)] & \text{if } 0 \leq k \leq N/25 \\ 1 & \text{if } N/25 \leq k \leq 24N/25 \\ \frac{1}{2}[1-\cos(25\pi(N-k)/N)] & \text{if } 26N/25 < k \leq (N-1) \\ 0 & \text{otherwise} \end{cases} \qquad (2.18)$$

and $C = S$ (the length of great circle)/s (the length of sub-profile).

(3) The variance of $\hat{x}(n \cdot \Delta\psi)$, $n = 0, 1, \ldots, (N-1)$ is defined by eqn. (2.6) as

$$P_{AVG}{}^2 = \frac{1}{N} \sum_{k=0}^{N-1} [x(k \cdot \Delta t)]^2 \tag{2.19}$$

or equivalently

$$P_{AVG}{}^2 = \frac{C}{N_{max}} \sum_{k=0}^{N_{max}-1} [y(k \cdot \Delta\psi)]^2, \quad \text{where } N_{max} = C \cdot N \tag{2.20}$$

if we neglect the effect of the tapering of the cosine-tapered window. The Fourier transform of $y(k \cdot \Delta\psi)$ has been computed by eqn. (1.62), i.e.

$$Y_n = \frac{\sqrt{C}}{N_{max}} \sum_{k=0}^{N_{max}-1} y(k \cdot \Delta\psi) \, e^{-i 2\pi k n / N_{max}} \tag{2.21}$$

where Y_n ; the global complex Fourier coefficients (unscaled)

$\quad\quad N_{max}$; the dimension of periodic data vector $y(k \cdot \Delta\psi)$

$\quad\quad C$; the coefficient as defined above.

Equation (2.21) is multiplied by $\sqrt{C}$ in order to maintain the average power defined by eqn. (2.19) or (2.20).

(4) The breakdown of the variance according to wavenumber (or frequency) has been computed by eqn. (2.8), namely

$$P_n{}^2 = \begin{cases} Y_0{}^2 & \text{for } n = 0 \\ 2|Y_n|^2 & \text{for } n = 1, 2, \ldots, (N_s-1) \\ Y_{N_M}{}^2 & \text{for } n = N_M \text{ and } N_{max} \text{ even} \\ 2|Y_{N_M}|^2 & \text{for } n = N_M \text{ and } N_{max} \text{ odd} \end{cases} \tag{2.22}$$

where $\quad N_M = \begin{cases} N_{max}/2 & \text{if } N_{max} \text{ even} \\ (N_{max}-1)/2 & \text{if } N_{max} \text{ odd} \end{cases}$

and finally the variance (total average power) has been computed,

$$P_{AVG}{}^2 = \sum_{n=0}^{N_M} P_n{}^2 \tag{2.23}$$

The results are described in Table 2.5 for the sub-profile given in Table 2.4.

Table 2.5. The Breakdown of Variance and Degree Powers with Reference to GEM 9 Surface from Non-Periodic Data.

Wavenumber or Frequency	Wavelength (km)	Arc No.				
		4	5	6	9	Mean
		Power (m ** 2)				
1	40 000	0.01	0.12	0.15	0.12	0.10
2	20 000	0.03	0.21	0.32	0.19	0.19
3	13 333	0.04	0.12	0.18	0.04	0.10
4	10 000	0.03	0.16	0.03	0.06	0.07
5	8 000	0.01	0.39	0.16	0.41	0.24
6	6 667	0.01	0.34	0.19	0.67	0.30
7	5 714	0.06	0.05	0.03	0.62	0.19
8	5 000	0.17	0.06	0.11	0.48	0.18
9	4 444	0.23	0.27	0.29	0.41	0.30
10	4 000	0.14	0.19	0.11	0.35	0.20
11	3 636	0.04	0.01	0.04	0.24	0.08
12	3 333	0.05	0.09	0.42	0.13	0.17
13	3 077	0.09	0.20	0.53	0.06	0.22
14	2 857	0.10	0.12	0.18	0.05	0.11
15	2 667	0.12	0.04	0.12	0.06	0.09
16	2 500	0.14	0.02	0.31	0.10	0.14
17	2 353	0.10	0.00	0.20	0.16	0.11
18	2 222	0.04	0.04	0.01	0.16	0.06
19	2 105	0.02	0.11	0.09	0.09	0.08
20	2 000	0.03	0.06	0.18	0.00	0.07
21	1 905	0.10	0.01	0.19	0.05	0.09
22	1 818	+0.14	+0.03	+0.21	+0.16	+0.13
.		1.70	2.64	4.05	4.61	3.25
.		.	.	.	.	.
.		.	.	.	.	.
$\hat{P}^{2}_{AVG}$		3.99	4.31	6.27	8.07	5.66

As we see from Table 2.5, the mean of variances (average powers) of undulations with reference to the GEM 9 surface is about $5.66\,\text{m}^2$. If we consider a 1 m uncertainty on each given data point, then with the assumption of white noise for the data, we can compute the reduced power as follows:

$$\bar{P}^2 \text{ (reduced)} = P^2 \text{ (measured)} - P^2 \text{(noise)}$$
$$\bar{P}^2 \text{ (reduced)} = 5.66 - 1.00 = 4.66 \text{ m}^2 \qquad (2.24)$$

and further $\bar{P} = 2.16$ m.

Thus we have recovered about 2.16 m undulation information in addition

to the GEM 9 undulations. From Table 2.5 we also see that $\lambda \geq 2000\,km$ contribute about 1.7 m , and $\lambda < 2000\,km$ contribute about 1.3 m of this extra 2.16 m undulation information. Notice the almost identical results given in Table 2.3, where we had assumed periodicity, for reduced, measured powers and extra undulation information.

2.5 Power Spectrum of Gravity Anomalies

As we mentioned before the main goal of Geodesy is the determination of the gravity field and the shape of the earth. In sections 2.1 and 2.2 we have explained how to obtain geoid heights from satellite altimeter data and how to recover gravity anomalies from these geoid heights. In this section we will examine the power spectrum of these recovered anomalies. We are mainly interested in pointing out the resolution of these anomalies. In other words, we will try to point out the minimum anomaly wavelength recoverable from these anomalies.

In our computations throughout this section the N-data vector has been transformed into the frequency domain by eqn. (1.62) after the removal of mean and discontinuities at the ends. Discontinuities at the ends have been removed by a cosine-tapered window. Degree powers P_n^2 and the total average power P_{AVG}^2 have been computed from eqns. (1.80) and (1.81) respectively. Frequencies and wavelengths have been computed from eqns. (2.12.a) and (2.12.b).

2.5.1 Power Spectrum from Recovered Anomalies Only

Two GEOS-3 arcs, one in the Calibration area and the other in the Philippines area, have been selected. These arcs are shown in Figure 2.2. Along these arcs point anomalies have been computed from adjusted undulations at the data points using least-squares collocation. The data vectors of point anomalies have been transformed into the frequency domain in order to obtain degree powers, their contribution to the variance, and finally the cumulative contribution by eqns. (2.8), (2.10) and (2.11.b) respectively. The results are described in Table 2.6.

The altimeter data used in the adjustment are approximately 2-second averages. So the shortest wavelength of information recoverable from the adjusted GEOS-3 altimeter data of OSU is roughly 28 km if we assume the data are noise-free. In fact we know that they are not noise free. These undulations have about 1 m standard deviations.

Table 2.6. The Breakdown of the Variance of Recovered Point Gravity Anomalies.

ARC NO: 11 (N = 268)

$\varphi_1 = 12°.85$, $\varphi_2 = 41°.26$, $s = 3705$ km
$\lambda_1 = 309°.96$, $\lambda_2 = 289°.75$

Mean: -20.8 mgals, Variance: 605. mgal²

Wave-number (n)	Wavelength λ_n (km)	Power P_n^2 (mgal**2)	Contribution C_n	Cumulative Contribution R_n
1	3705.	50.00	0.292	0.292
2	1852.	34.87	0.204	0.495
3	1235.	20.36	0.119	0.614
4	926.	6.57	0.038	0.653
5	741.	5.98	0.035	0.688
6	617.	10.73	0.063	0.750
7	529.	3.25	0.019	0.769
8	463.	3.92	0.023	0.792
9	412.	2.17	0.015	0.805
10	371.	2.19	0.013	0.818
14	264.	3.33	0.019	0.871
20	185.	0.47	0.003	0.900
29	128.	1.40	0.008	0.960
50	74.	0.03	0.000	0.986
100	37.	0.02	0.000	0.998
134	28.	0.01	0.000	1.000

ARC NO: 12 (N = 300)

$\varphi_1 = -0°.02$, $\varphi_2 = 32°.26$, $s = 4162$ km
$\lambda_1 = 146°.53$, $\lambda_2 = 126°.90$

Mean: 13.4 mgals, Variance: 970. mgal²

Wave-number (n)	Wavelength λ_n (km)	Power P_n^2 (mgal**2)	Contribution C_n	Cumulative Contribution R_n
1	4162.	16.13	0.020	0.020
2	2081.	22.86	0.029	0.049
3	1387.	0.03	0.000	0.049
4	1041.	121.48	0.154	0.203
5	833.	4.09	0.005	0.208
6	694.	79.41	0.100	0.308
7	595.	16.29	0.021	0.329
8	520.	86.29	0.109	0.438
9	463.	26.44	0.033	0.471
10	416.	74.99	0.095	0.566
20	208.	8.53	0.011	0.904
30	139.	0.33	0.000	0.968
36	115.	4.13	0.005	0.990
40	104.	0.84	0.001	0.994
60	69.	0.03	0.000	0.998
100	42.	0.02	0.000	0.999
150	28.	0.00	0.000	1.000

From the results given in Table 2.6 we can point out the shortest anomaly wavelength recoverable from point anomalies implied by GEOS-3 altimeter data. First we will assume the standard deviations of point anomalies to be about 27 mgals, this having been found in the solution of least-squares collocation. Secondly we will also assume that the noise spectrum is equally distributed on each degree. Then we have, with (1.97)

$$\text{var}(P_n) = \sigma^2/N = \begin{cases} 729 \text{ mgal}^2/268 = 2.72 \text{ mgal}^2 \text{ for the } 11^{th} \text{ arc} \\ 729 \text{ mgal}^2/300 = 2.43 \text{ mgal}^2 \text{ for the } 12^{th} \text{ arc} \end{cases} \quad (2.25)$$

We see from Table 2.6 that the signal-to-noise ratio is greater than <u>one</u> above the 264 km wavelength for the 11th arc and above the 115 km wavelength for the 12th arc.

2.5.2 Power Spectrum from Measured (by Ship) and Recovered Point Anomalies

In the previous section we have examined the power spectrum of computed point anomalies along two arcs separately. In this section we will examine the power spectrum of anomalies using predicted point anomalies and measured point anomalies, which were kindly provided to us by Mr. W.H. Chapman, U.S. Geological Survey. The anomalies are along the sub-profiles 13, 14, and 15, which are described in Table 2.7 and shown in Figure 2.3.

Table 2.7. Statistics of Recovered (Predicted) and Measured Point Gravity Anomalies Along the Arcs 13, 14, 15.

Arc No.	N	Length (km)	Latitude φ_1, φ_2 (Deg.)	Longitude λ_1, λ_2 (Deg.)	Mean (mgals)		Variance (mgal ** 2)		Mean Diff. (mgals)	RMS Diff.
					Pred.	Meas.	Pred.	Meas.	(Meas-Pred)	(mgals)
13	132	383	25.05, 22.36	122.11, 124.45	-43.8	-45.4	3782.	5776.	-1.6	39.9
14	78	203	22.54, 24.39	124.53, 124.71	-58.5	-49.2	4529.	4109.	9.3	23.4
15	168	473	18.08, 13.88	127.15, 126.28	17.1	13.9	372.	303.	-3.2	13.0

The data vectors of predicted point anomalies and measured point anomalies have been transformed into the frequency domain separately along each sub-profile, given in Table 2.7, in order to compute degree powers, their contribution to the variance, and finally the cumulative contribution by eqns. (2.8), (2.10), and (2.11.b) respectively. The results are described in Table 2.8.

Table 2.8. The Breakdown of Variance and Degree Powers of Point Anomalies Along Arcs 13, 14, 15.

ARC NC: 13

ARC NO: 14

ARC NC: 15

If we assume the standard deviations of predicted anomalies to be 27 mgals
as in section 2.5.1, then we can calculate the uncertainties on each degree power
to be

$$var(P_n)_{Pred} = \frac{\sigma_{Pred}^2}{N} = \begin{cases} 729/132 = 5.52 \text{ mgal}^2 & \text{for the } 13^{th} \text{ arc} \\ 729/\ 78 = 9.35 \text{ mgal}^2 & \text{for the } 14^{th} \text{ arc} \\ 729/148 = 4.92 \text{ mgal}^2 & \text{for the } 15^{th} \text{ arc} \end{cases} \quad (2.26)$$

We see from Table 2.8 that for predicted anomalies the signal-to-noise
ratio is greater than one above the 127 km wavelength for the 13th arc and above
the 68 km wavelength for the 14th arc, and finally above the 208 km wavelength
for the 15th arc. The 208 km shortest recoverable wavelength of the 15th arc is
much higher than 127 and 68 km of the 13th and 14th arcs respectively. Assuming
the same precision for predicted anomalies along each arc we can state that the
big difference above is most likely due to smooth ocean bottom topography along
the 15th arc and rough ocean bottom topographies along the 13th and 14th arcs.

Thus, after all these analyses, we can conclude that the shortest point
anomaly wavelength implied by GEOS-3 altimeter data is about 70 km when the
standard deviation of predicted anomalies is about 27 mgals.

Now consider the standard deviation of each measured point anomaly to be
5 mgals. Then we can compute uncertainties on each degree power as follows:

$$var(P_n)_{Meas} = \frac{\sigma_{Meas}^2}{N} = \begin{cases} 25/132 = 0.19 \text{ mgal}^2 & \text{for the } 13^{th} \text{ arc} \\ 25/\ 78 = 0.32 \text{ mgal}^2 & \text{for the } 14^{th} \text{ arc} \\ 25/148 = 0.17 \text{ mgal}^2 & \text{for the } 15^{th} \text{ arc} \end{cases} \quad (2.27)$$

We see from Table 2.8 that for measured point anomalies, the signal-to-
noise ratio is greater than one above the 12.3 km wavelength for the 13th arc
and above the 6.3 km wavelength for the 14th arc and finally above the 21.5 km
wavelength for the 15th arc. We conclude from the results above that the meas-
urements (ship) have a much greater high-frequency contribution to the power
than predicted point anomalies; this we had anticipated beforehand.

2.5.3 Power Spectrum from Measured (by Ship) and Recovered Point Anomalies, Bathymetry, and Filtering

In previous sections we have examined the power spectrum from predicted
anomalies, as well as measured (by ship) and predicted anomalies. In this section
we have selected three more profiles, numbered 16, 17, and 18, which are shown
in Figure 2.3. Along these, measured point anomalies and bathymetry were
provided to us by A.B. Watts of Columbia University. On these profiles we have
computed point anomalies using least-squares collocation from GEOS-3 altimeter
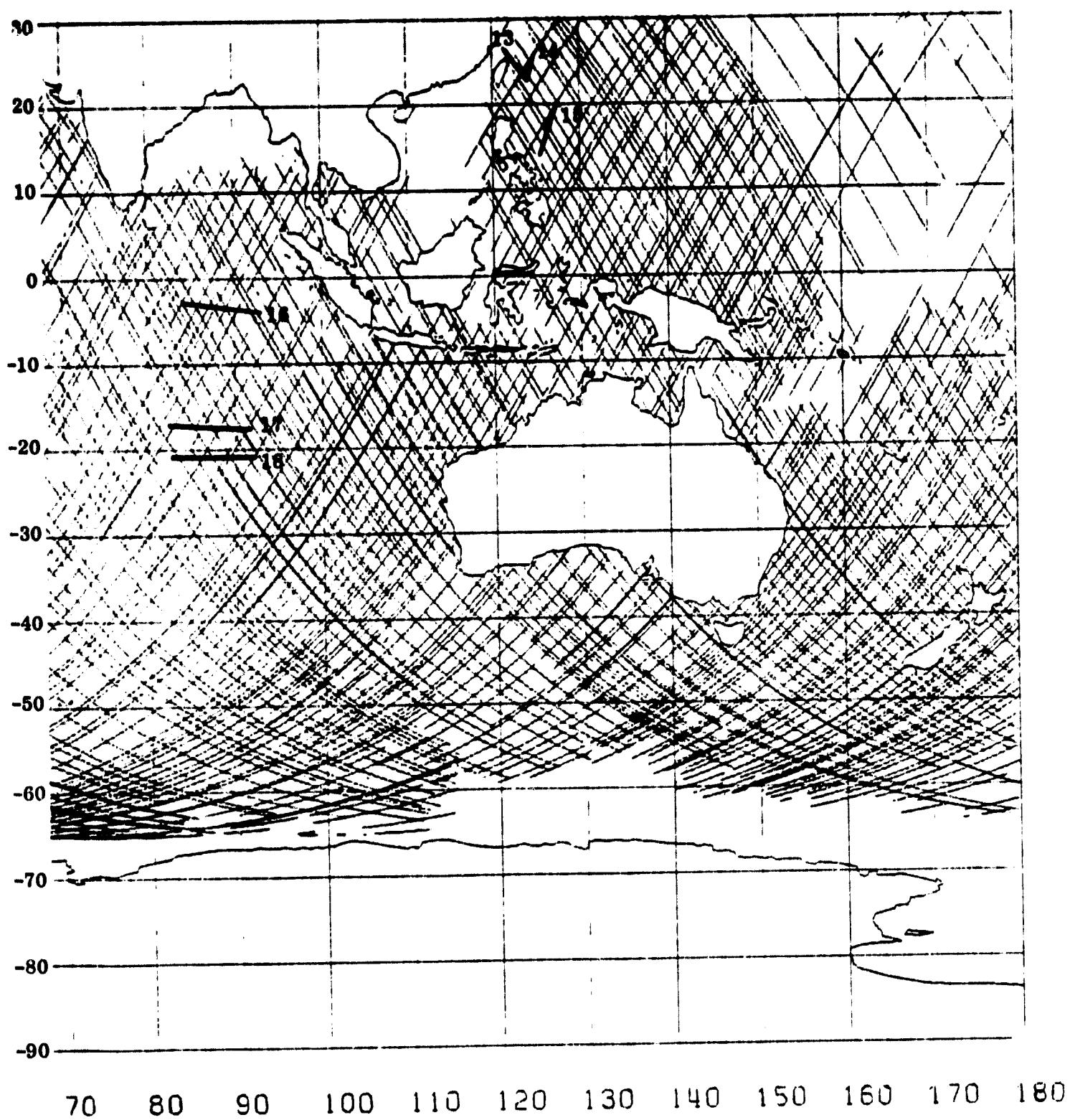
Figure 2.3. GEOS-3 Ground Tracks and Some Selected Sub-Profiles.

data. Predictions have been made at those points where measured anomalies have their local maxima or minima and are still not further than 30 km from neighboring prediction points. These three profiles are described in Table 2.9.

Table 2.9. Statistics of Recovered (Predicted) and Measured Point Gravity Anomalies Along the Arcs 16, 17, 18.

Arc No.	N		Length (km)	Latitude φ_1, φ_2 (Deg.)	Longitude λ_1, λ_2 (Deg.)	Mean (mgals)		Variance (mgal^2)		Mean Diff. (Meas-Pred)	RMS Diff. (mgals)
	Pred.	Meas.				Pred.	Meas.	Pred.	Meas.		
16	62	390	1202.	- 3.19, - 4.00	85.12, 93.52	-19.5	-18.3	689.	707.5	-1.2	11.5
17	54	178	1012.	-16.76, -17.51	83.23, 91.87	-24.5	-31.0	1240.	1791.	6.5	19.7
18	66	210	1013.	-20.53, -20.64	83.27, 92.14	- 8.1	-22.2	153.	760.	14.1	14.7

We have also computed geoid undulations by least-squares collocation at those points where point anomalies are predicted along the profiles 16, 17, and 18. All these, the bathymetry, measured anomaly, predicted anomaly, and geoid undulation profiles (corresponding to the profiles 16, 17, and 18) are plotted in Figures 2.4, 2.5, and 2.6 respectively.

The data vectors of predicted point anomalies and measured point anomalies have been transformed into the frequency domain along each sub-profile, given in Table 2.9, in order to obtain degree powers, their contribution to the variance, and finally the cumulative contribution by eqns. (2.8), (2.10), and (2.11.b) respectively. The results are described in Table 2.10.

Figure 2.4. Altimeter Anomaly, Measured Anomaly, Undulation and Bathymetry.
Along Arc 16.

-62-
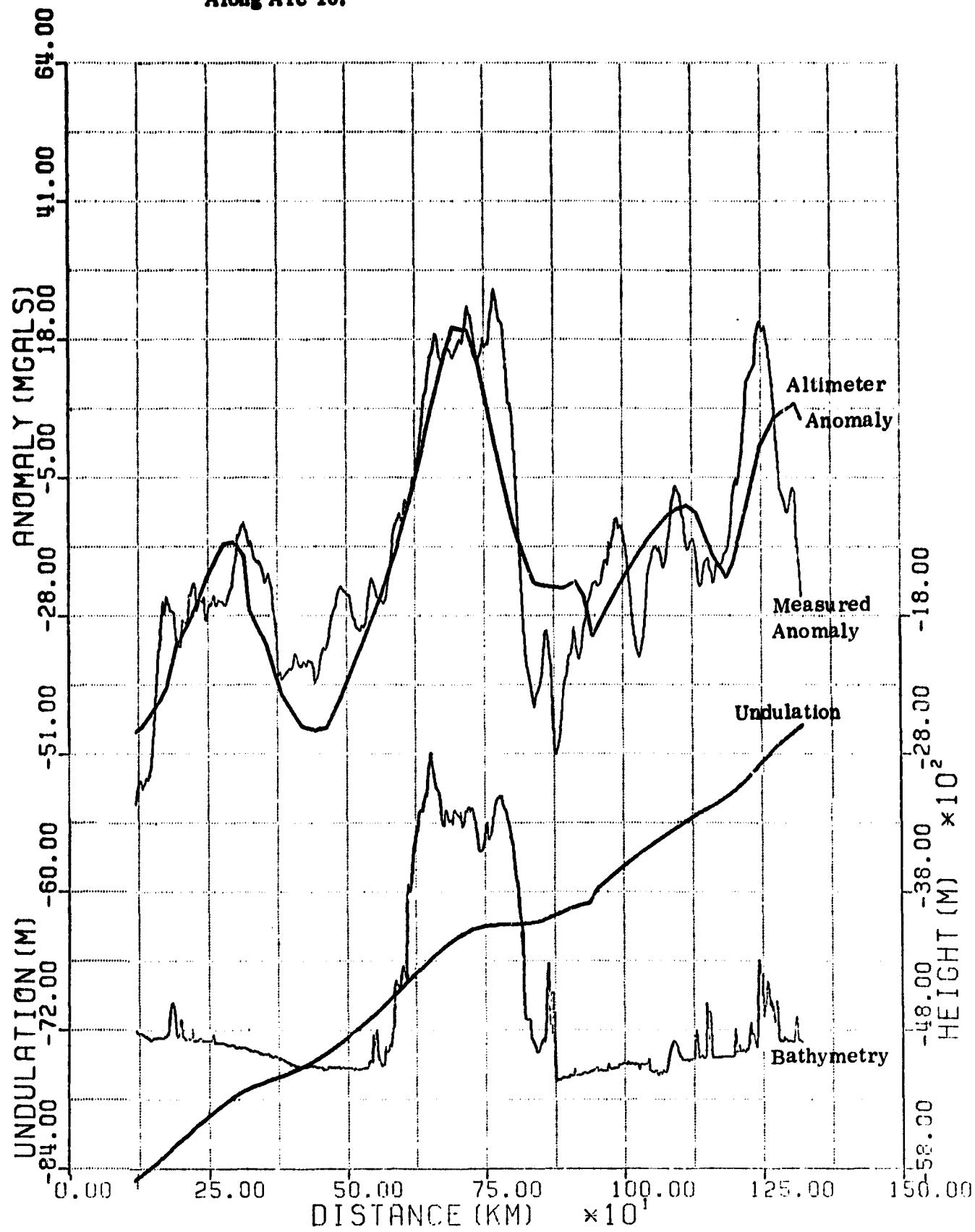
Figure 2.5. Altimeter Anomaly, Measured Anomaly, Undulation and Bathymetry. Along Arc 17.
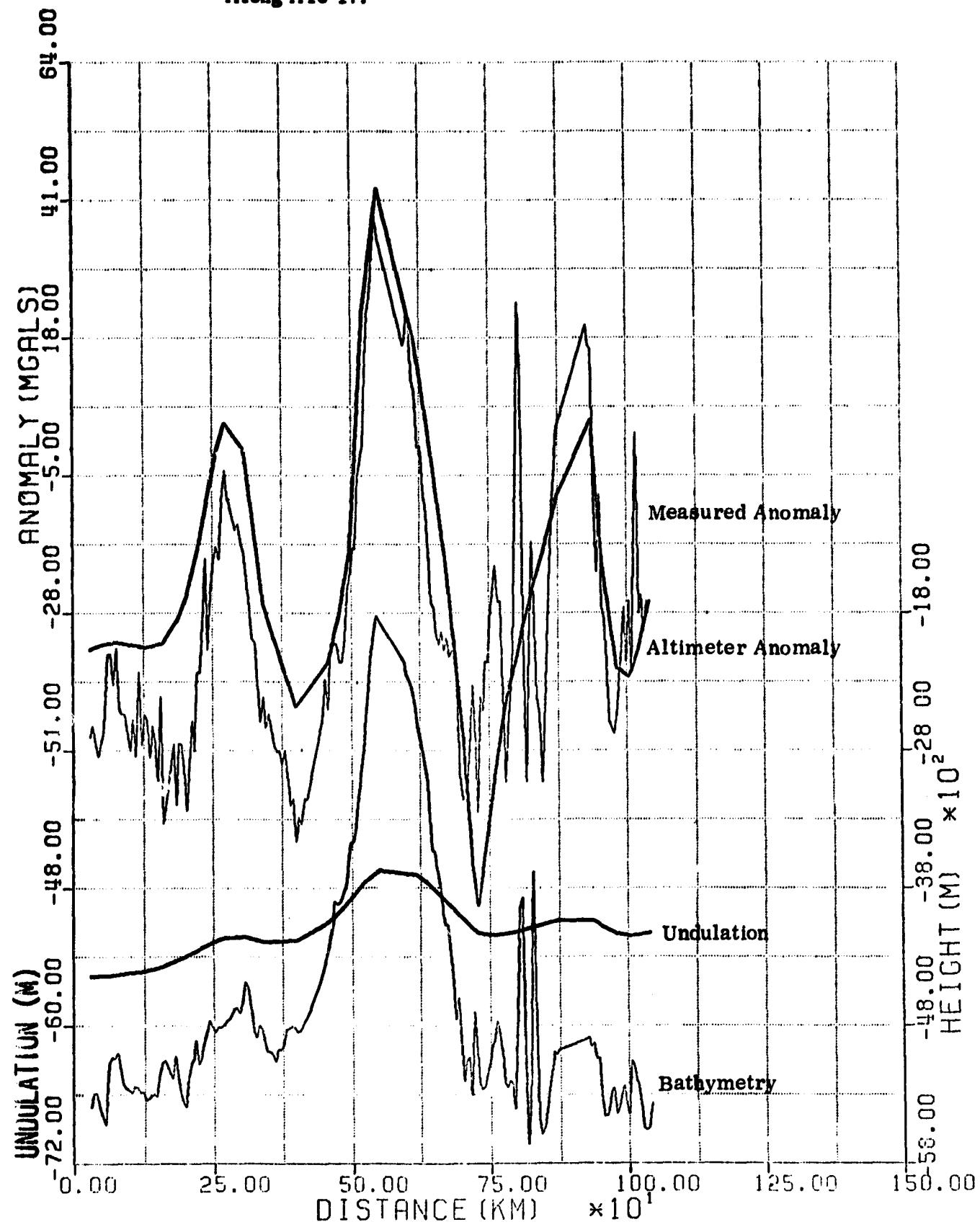
Figure 2.6. Altimeter Anomaly, Measured Anomaly, Undulation and Bathymetry. Along Arc 18.

-64-
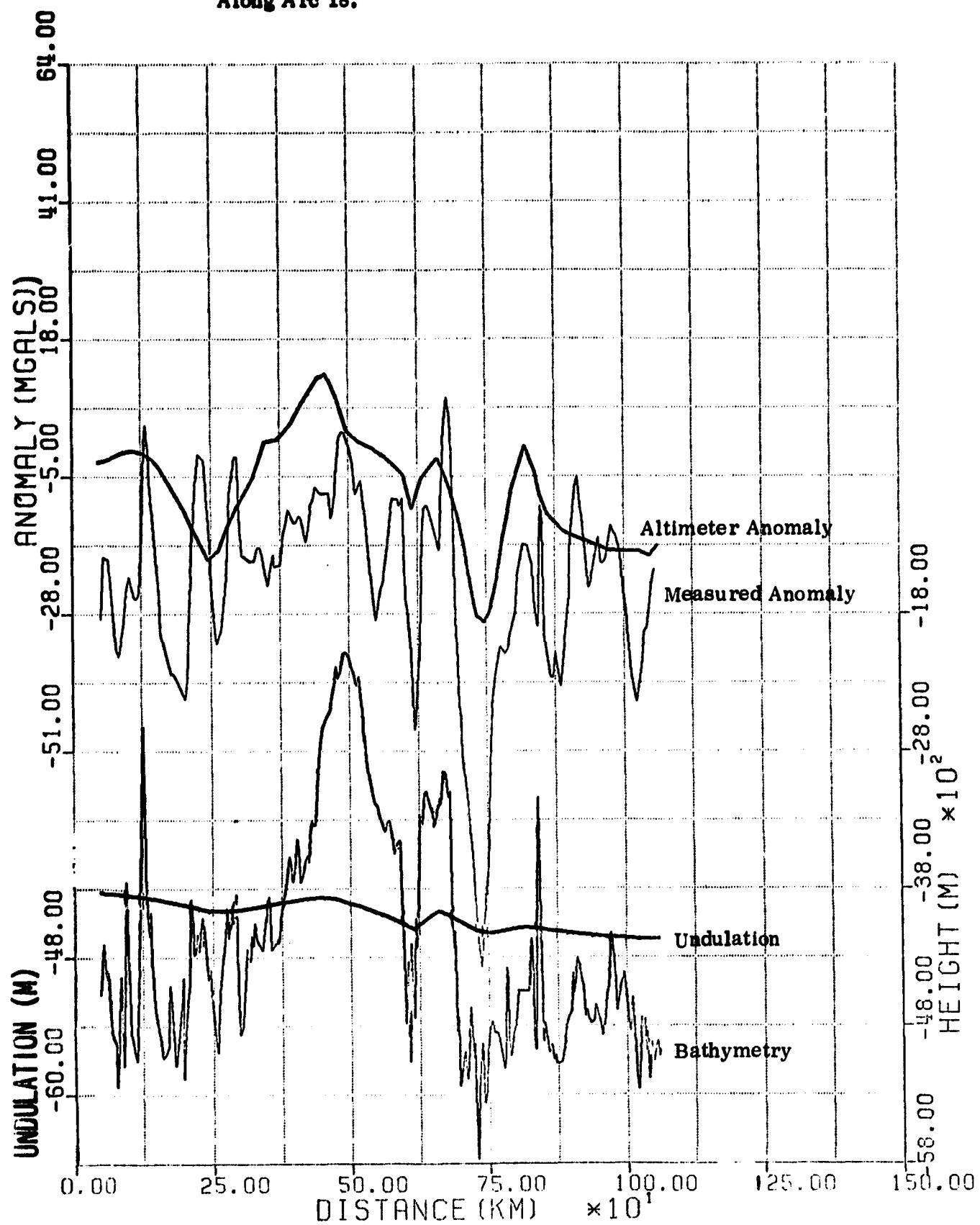
Table 2.10. The Breakdown of Variance and Degree Powers of Point Anomalies Along Arcs 16, 17, 18.

ARC NO: 16

Wave number (n)	Wavelength λ_n (km)	Power P_n (mgal²) 1. Pred. 2. (Meas.)	Contribution C_n 1. Pred. 2. (Meas.)	Cumulative Contribution R_n 1. Pred. 2. (Meas.)
1	1201.6	53.18 (30.98)	0.172 (0.085)	0.172 (0.085)
2	600.8	103.12 (113.74)	0.334 (0.311)	0.506 (0.395)
3	400.5	57.28 (72.61)	0.186 (0.198)	0.692 (0.593)
4	300.4	25.80 (30.14)	0.084 (0.082)	0.775 (0.676)
5	240.3	17.95 (50.36)	0.058 (0.138)	0.834 (0.813)
6	200.3	25.54 (1.16)	0.083 (0.003)	0.916 (0.816)
7	171.7	6.75 (8.80)	0.022 (0.024)	0.938 (0.840)
8	150.2	3.14 (29.47)	0.010 (0.080)	0.948 (0.921)
9	133.5	2.27 (2.68)	0.007 (0.006)	0.956 (0.928)
10	120.2	1.32 (2.13)	0.004 (0.006)	0.960 (0.934)
..20	66.1	0.25 (0.58)	0.001 (0.002)	0.986 (0.976)
..30	40.1	0.26 (0.22)	0.001 (0.001)	1.000 (0.991)
..53	22.7	0.00 (0.10)	0.000 (0.000)	1.000 (0.996)

ARC NO: 17

Wave number (n)	Wavelength λ_n (km)	Power P_n (mgal²) 1. Pred. 2. (Meas.)	Contribution C_n 1. Pred. 2. (Meas.)	Cumulative Contribution R_n 1. Pred. 2. (Meas.)
1	1097.1	51.07 (100.81)	0.000 (0.146)	0.000 (0.146)
2	593.6	44.79 (36.69)	0.078 (0.053)	0.159 (0.199)
3	335.7	393.92 (300.18)	0.616 (0.435)	0.765 (0.634)
4	251.8	114.98 (69.15)	0.178 (0.100)	0.944 (0.734)
5	201.4	3.86 (23.17)	0.006 (0.034)	0.950 (0.768)
6	167.9	2.53 (28.85)	0.004 (0.042)	0.954 (0.810)
7	143.9	0.06 (9.25)	0.000 (0.013)	0.954 (0.823)
8	125.9	21.85 (29.53)	0.034 (0.043)	0.988 (0.866)
9	111.9	3.34 (14.21)	0.005 (0.021)	0.993 (0.887)
10	100.7	0.13 (3.87)	0.000 (0.006)	0.993 (0.892)
..20	50.4	0.10 (0.36)	0.000 (0.001)	1.000 (0.943)
..88	11.4	0.06 (0.17)	0.000 (0.000)	1.000 (1.000)

ARC NO: 18

Wave number (n)	Wavelength λ_n (km)	Power P_n (mgal²) 1. Pred. 2. (Meas.)	Contribution C_n 1. Pred. 2. (Meas.)	Cumulative Contribution R_n 1. Pred. 2. (Meas.)
1	1064.0	30.14 (30.25)	0.380 (0.140)	0.380 (0.140)
2	582.0	11.06 (34.67)	0.139 (0.130)	0.067 (0.270)
3	334.7	15.29 (14.61)	0.177 (0.064)	0.685 (0.333)
4	251.0	13.15 (6.57)	0.153 (0.025)	0.817 (0.359)
5	200.8	1.91 (11.22)	0.022 (0.041)	0.840 (0.397)
6	167.3	4.58 (27.32)	0.053 (0.102)	0.893 (0.499)
7	143.4	2.08 (29.72)	0.020 (0.111)	0.923 (0.599)
8	125.5	4.00 (9.73)	0.047 (0.036)	0.970 (0.646)
9	111.6	0.71 (14.93)	0.000 (0.055)	0.973 (0.701)
10	100.4	0.35 (1.46)	0.004 (0.005)	0.963 (0.707)
..20	50.2	0.04 (2.56)	0.001 (0.010)	0.995 (0.925)
..30	33.5	0.02 (0.29)	0.000 (0.001)	0.999 (0.981)
..56	17.9	0.00 (0.13)	0.000 (0.000)	1.000 (0.999)

As we did in previous sections let us assume that the standard deviations of predicted anomalies are 27 mgals each. Then we can compute uncertainties on each degree power:

$$\text{var}(P_n)_{Pred} = \sigma^2_{Pred}/N = \begin{cases} 729/62 = 11.75 \text{ mgal}^2 \text{ for the } 16^{th}\text{arc} \\ 729/54 = 13.50 \text{ mgal}^2 \text{ for the } 17^{th}\text{arc} \\ 729/66 = 11.05 \text{ mgal}^2 \text{ for the } 18^{th}\text{arc} \end{cases} \quad (2.28)$$

We see from Table 2.10 that for predicted anomalies the signal-to-noise ratio is greater than <u>one</u> above the 200.3 km wavelength for the 16th arc and above the 125.9 km wavelength for the 17th arc and finally above the 251.0 km wavelength for the 18th arc.

Let us again consider, as we did in previous sections, the standard deviation of each measured point anomaly to be roughly 5 mgals. Then we can compute uncertainties on each degree power as follows:

$$\text{var}(P_n)_{Meas} = \sigma^2_{Meas}/N = \begin{cases} 25/390 = 0.06 \text{ mgal}^2 \text{ for the } 16^{th}\text{arc} \\ 25/178 = 0.14 \text{ mgal}^2 \text{ for the } 17^{th}\text{arc} \\ 25/210 = 0.12 \text{ mgal}^2 \text{ for the } 18^{th}\text{arc} \end{cases} \quad (2.29)$$

We see from Table 2.10 that for measured anomalies, the signal-to-noise ratio is greater than <u>one</u> above the 22.3 km wavelength for the 16th arc and above the 11.4 km wavelength for the 17th arc and finally above the 17.6 km wavelength for the 18th arc.

We see again that measurements reflect greater high-frequency contribution to the total power than predicted anomalies.

Since measurements reflect the shorter wavelength features of anomalies, the RMS difference between predictions and observations along a particular profile should get smaller when we use a low-pass filter for:

(1) measurements
(2) both measurements and predictions.

The low-pass filter used here can be described as follows: The data vector of the spatial (or time) domain representation along a particular arc has been transformed into the frequency domain. Then the Fourier coefficients have been truncated above $M \leq N_H$, where N_H is the Nyquist frequency and M is the cut-off frequency in the basic interval, i.e. the length of the arc in question. Truncating the frequency domain series above the frequency M corresponds to low-pass filtering. After having generated the complex Fourier coefficients X_n, applying a filter only involves the truncation of the coefficients above $M \leq N_H$. Then by an inverse Fourier transform we can regenerate the spatial domain

representation. In order to make our approach more understandable let us give an example: Suppose we have an arc of 1000 km length with 100 equally spaced observations ($N = 100$) on it. Then the Nyquist frequency is:

$$N_H = N/2 = 50 \text{ (in the interval of 1000 km)}$$

and the frequency, with eqn. (2.12.a), is

$$f_n = n \text{ (in the interval of 1000 km)}, \quad n = 0, 1, \ldots, 50$$

and finally the wavelength, with eqn. (2.12.b), is

$$\lambda_n = 1/f_n = 1000 \text{ km}/n, \quad n = 1, 2, \ldots, 50$$

Thus we can compute the shortest wavelength recoverable, which corresponds to the Nyquist frequency, for this particular example,

$$\lambda_{N_H} = 1000 \text{ km}/50 = 20 \text{ km}$$

The data vector of observations, say $x(t)$, $t = 0, 1, \ldots, 99$, is transformed into the frequency domain by eqn. (2.5), i.e.

$$X_n = \frac{1}{100} \sum_{k=0}^{99} x(k) e^{-i 2\pi kn/N} \qquad n = 0, 1, \ldots, 50$$

The inverse Fourier transform (c.f. eqn. (1.71)) yields

$$x(k) = x_1(k) + x_1^*(k), \qquad k = 0, 1, \ldots, 99$$

where $\quad x_1(k) = \sum_{n=0}^{99} Z_n e^{i 2\pi kn/100}$, such that $\quad Z_n = \begin{cases} X_n & \text{for } 0 \leq n \leq 50 \\ 0 & \text{otherwise} \end{cases}$

$$\lambda_M = 50 \text{ km} = 1000 \text{ km}/M \rightarrow M = 20$$

Now suppose we want to filter the power contribution of any wavelength smaller than 50 km. Then we have

$$\lambda_M = 50 \text{ km} = 1000 \text{ km}/M \rightarrow M = 20$$

Now the truncated complex Fourier coefficients, say Y_n, are defined as

$$Y_n = \begin{cases} X_n & \text{for } n = 0, 1, \ldots, 19, 20 \\ 0 & \text{otherwise} \end{cases}$$

Then, finally, we can regenerate the spatial domain representation, say $y(t)$, corresponding to the frequency domain representation Y_n. The function $y(t)$ does not have any high-frequency components above 20. The definition of $y(t)$ is as follows:

$$y(t) = y_1(t) + y_1^*(t)$$

where

$$y_1(t) = \sum_{n=0}^{\infty} Y_n \, e^{i 2\pi k n / 100}$$

By using low-pass filtering as described above we have filtered out any power contributions from wavelengths below 50 km, 100 km, and 200 km for the data vectors of measured anomalies and bathymetry along the sub-arcs 16, 17 and 18 shown in Figure 2.3. The regenerated spatial domain representations are shown in Figures 2.7, 2.8, 2.9 for $\lambda \le 50$ km, $\lambda \le 100$ km, $\lambda \le 200$ km, respectively. In addition to the three sub-profiles above we applied low-pass filtering to the data vector of measured anomalies along the sub-profiles 13, 14, 15 of Figure 2.3 for $\lambda \le 50$ km, $\lambda \le 100$ km, and $\lambda \le 200$ km. The results are described in Table 2.11.

In the procedures above we have only filtered out the high-frequency components of the power of measured anomalies. In addition we also filtered out the high-frequency components of the predicted anomalies. The previously defined low-pass filter has been applied for the data vector of measured anomalies, bathymetry, as well as predicted anomalies along the profiles 16, 17, and 18. The regenerated spatial domain representations are shown in Figure 2.10, 2.11, and 2.12 for $\lambda \le 50$ km, $\lambda \le 100$ km, $\lambda \le 200$ km respectively. We also applied the low-pass filter to the data vector of measured anomalies and predicted anomalies along the sub-profiles 13, 14, 15, of Figure 2.3 for $\lambda \le 50$ km, $\lambda \le 100$ km, and $\lambda \le 200$ km. The results are described in Table 2.11.

Table 2.11. RMS Differences when Low-Pass Filtering is Applied to Point Anomalies Along the Arcs 13, 14, 15, 16, 17, 18.

Arc No.	Mean Diff. (Pred-Meas) (mgals)	RMS Diff. (mgals)	RMS when $\lambda \le 50$ km filtered		RMS when $\lambda \le 100$ km filtered		RMS when $\lambda \le 200$ km filtered	
			Meas.	Both Pred & Meas.	Meas.	Both Pred & Meas.	Meas.	Both Pred & Meas.
13	6.5	41.7	40.4	40.3	32.0	31.5	18.9	18.9
14	0.0	26.6	23.5	23.2	17.7	17.4	17.5	12.7
15	0.9	13.6	12.6	12.7	11.8	11.7	8.6	8.4
16	-1.2	11.5	11.7	11.3	11.9	10.9	10.3	8.2
17	6.5	18.6	15.3	15.1	13.0	12.5	13.2	12.0
18	14.1	14.4	13.1	13.1	11.1	11.1	8.6	8.2
Mean	4.5	21.1	19.4	19.3	16.3	15.9	12.9	11.4

We see from Table 2.11 that the RMS differences between the cases (i) when only the data vector of measured anomalies are filtered and (ii) when both data vectors of measured anomalies and predicted anomalies are filtered,
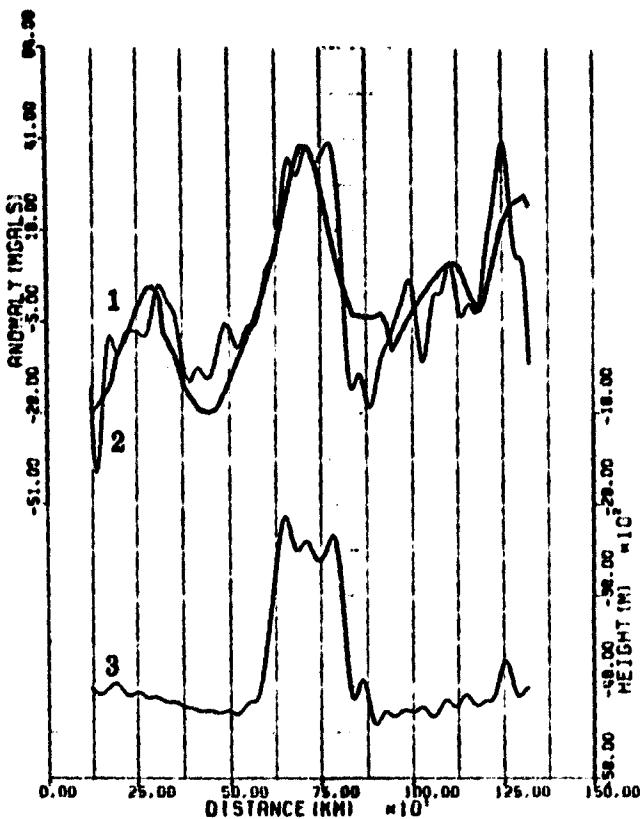
Arc 16
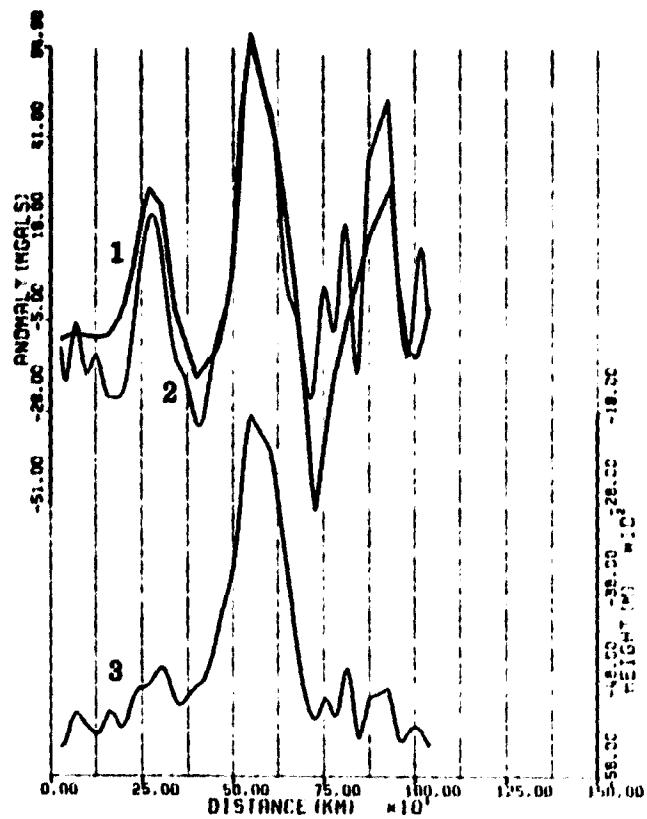
Arc 17
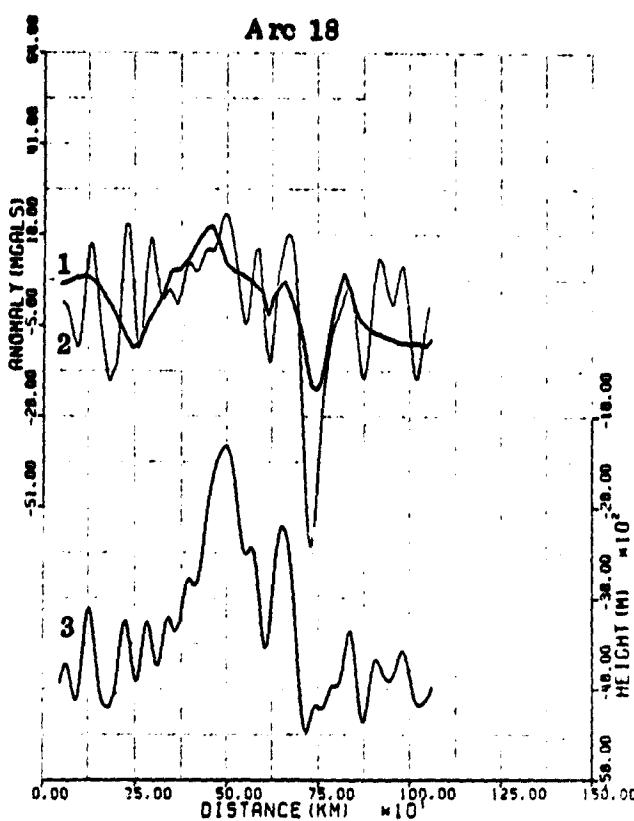
Arc 18

Figure 2.7. The Low-Pass Filtering
($\lambda \leq 50$ km) on Measured Anomalies
and Bathymetry Along Arcs 16, 17, 18.

1. Altimeter Anomaly (without filtering)
2. Measured Anomaly (low-pass filtering)
3. Bathymetry (low-pass filtering)

-69-
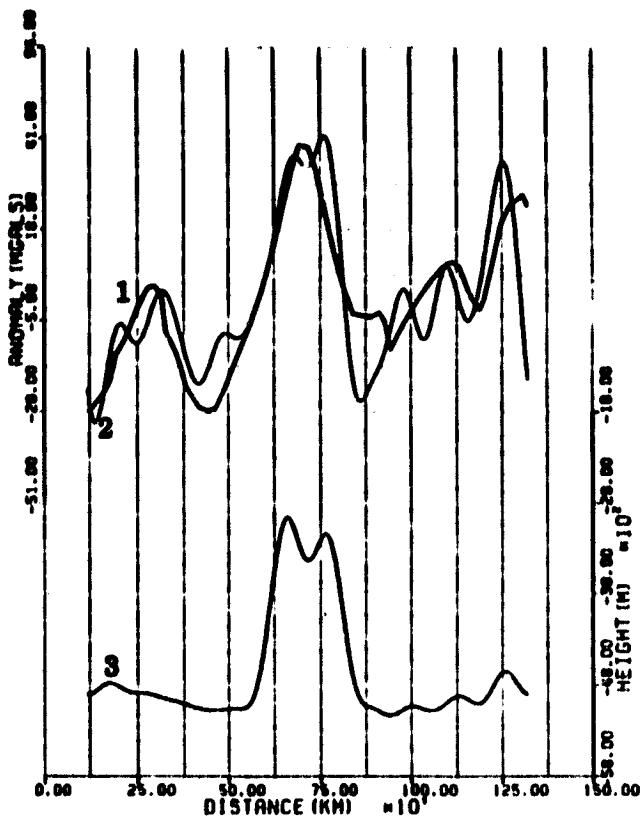
Arc 16
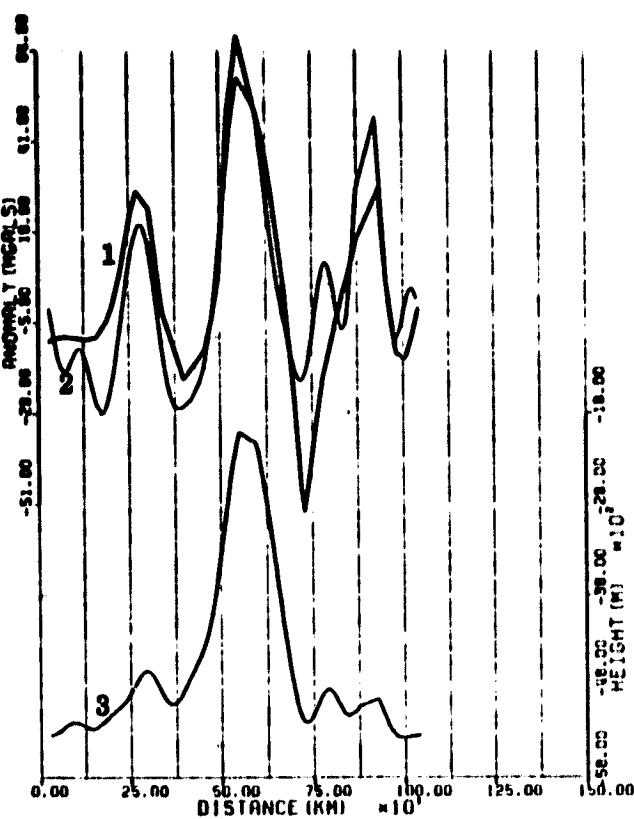
Arc 17
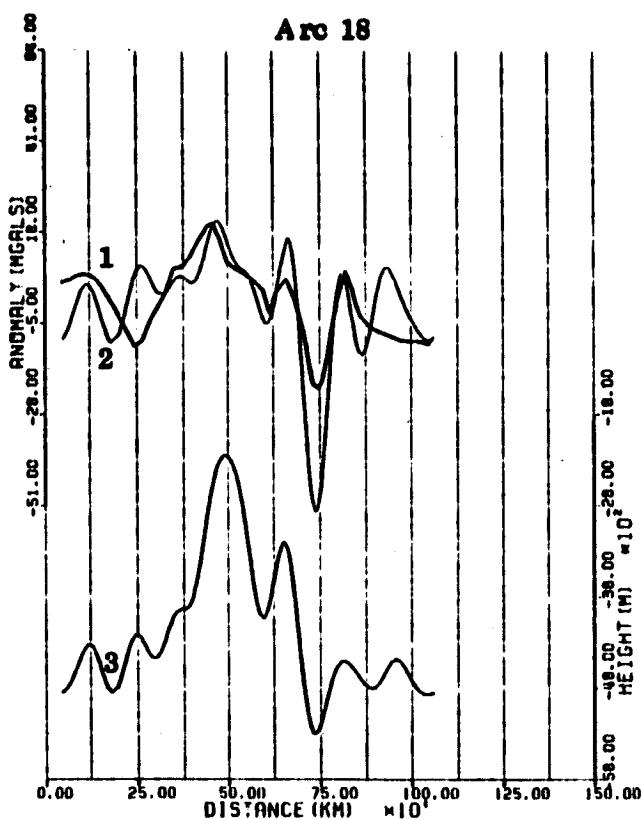
Arc 18

Figure 2.8. The Low-Pass Filtering
($\lambda \leq 100$ km) on Measured Anomalies
and Bathymetry Along Arcs 16, 17, 18.

1. Altimeter Anomaly (without filtering)
2. Measured Anomaly (low-pass filtering)
3. Bathymetry (low-pass filtering)
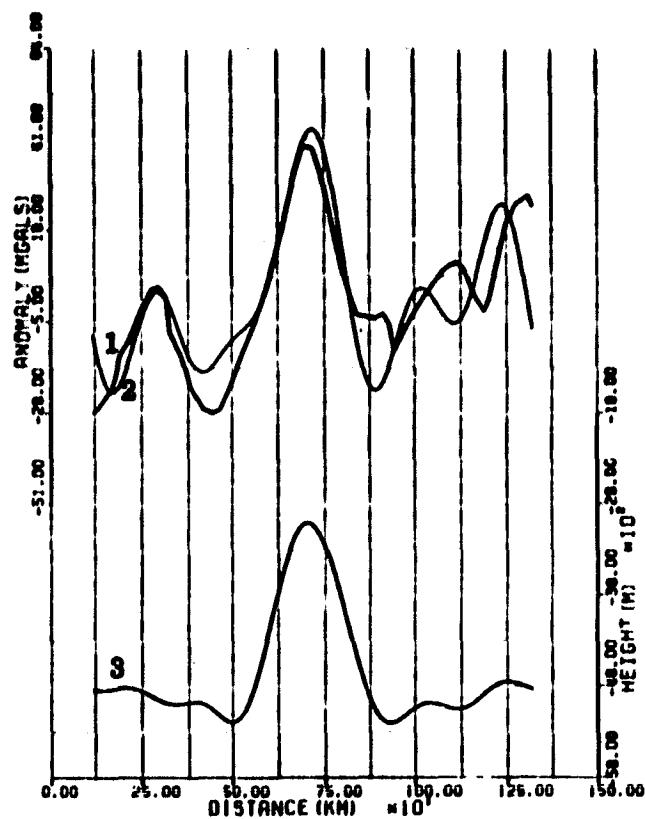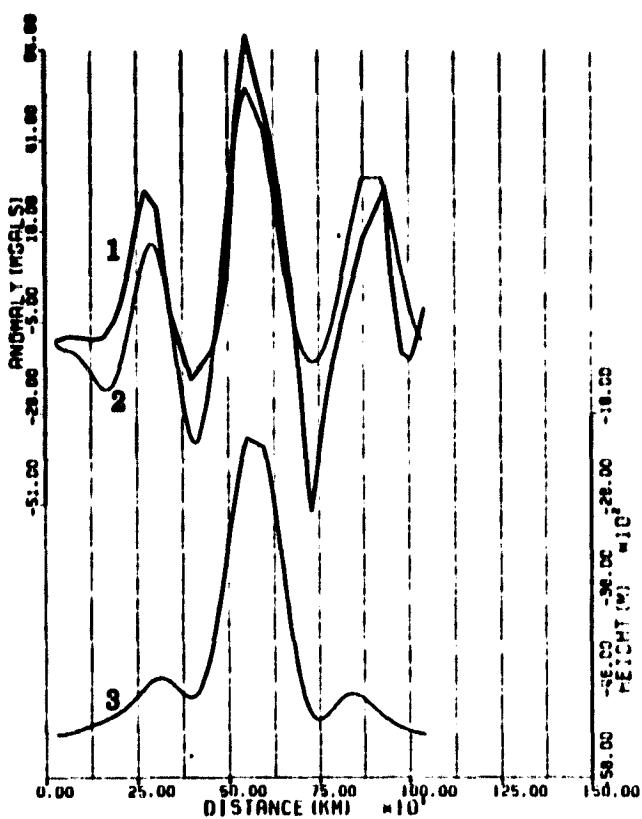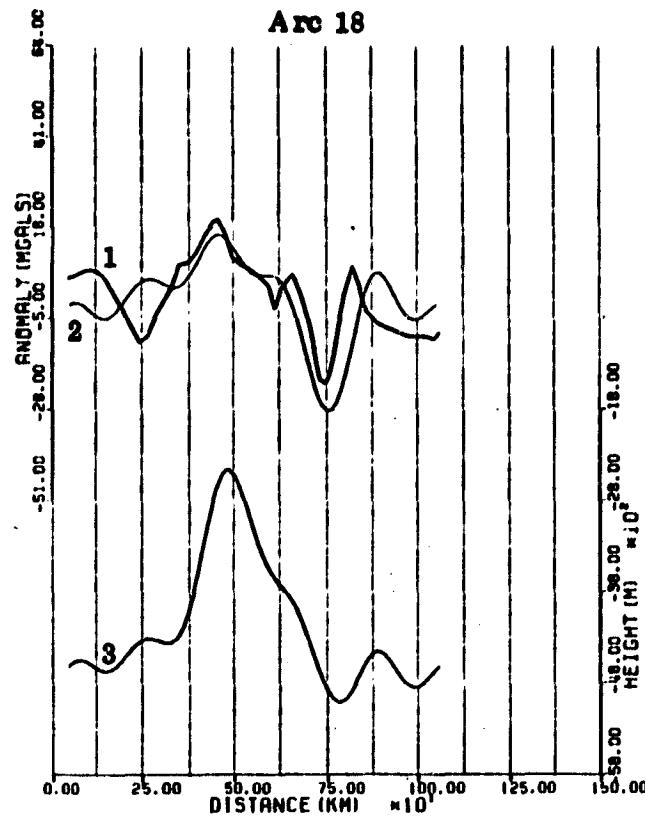
Arc 16

Arc 17

Arc 18

Figure 2.9. The Low-Pass Filtering
($\lambda \le 200$ km) on Measured Anomalies
and Bathymetry Along Arcs 16, 17, 18.

1. Altimeter Anomaly (without filtering)
2. Measured Anomaly (low-pass filtering)
3. Bathymetry (low-pass filtering)
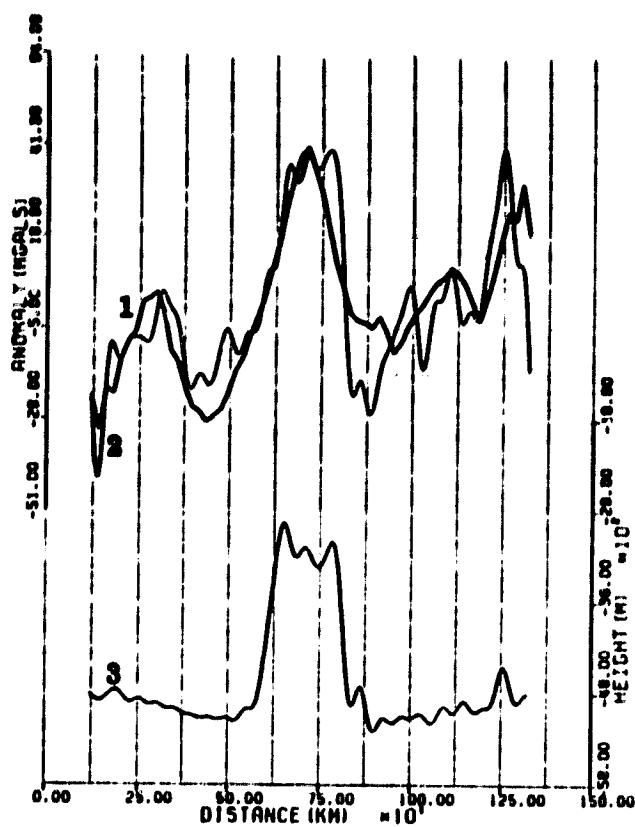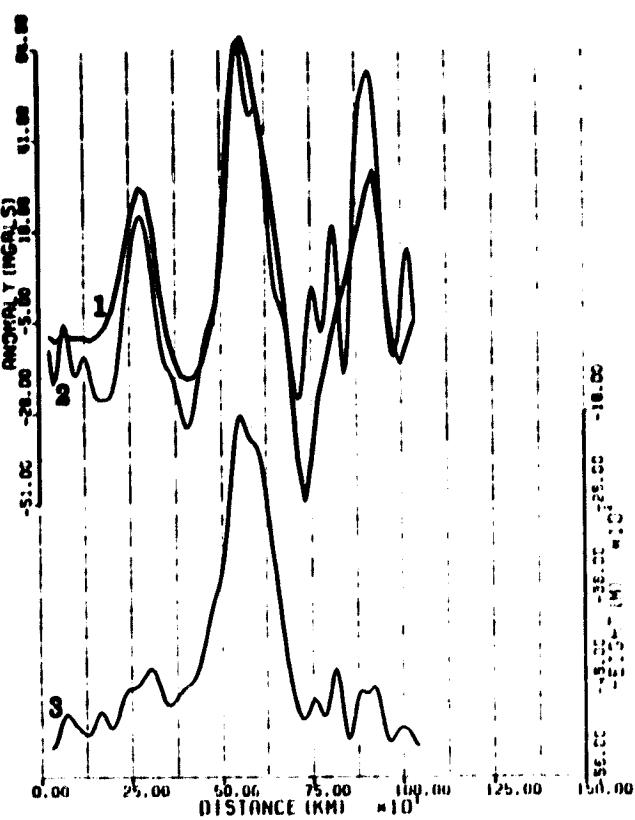
Arc 16

Arc 17

Arc 18
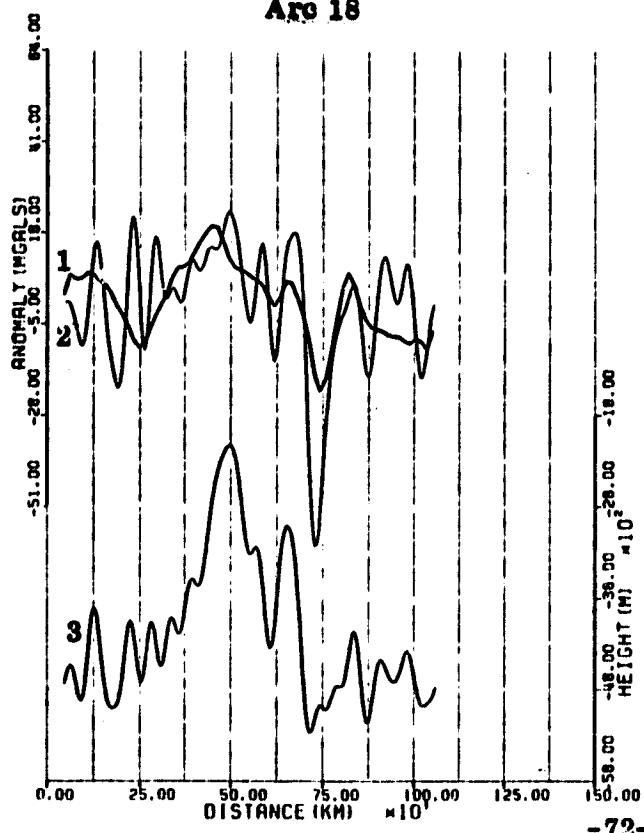
Figure 2.10. The Low-Pass Filtering
($\lambda \leq 50$ km) on Altimeter Anomalies,
Measured Anomalies and Bathymetry
Along Arcs 16, 17, 18.

1. Altimeter Anomaly
2. Measured Anomaly
3. Bathymetry

Arc 16

Arc 17

Arc 18
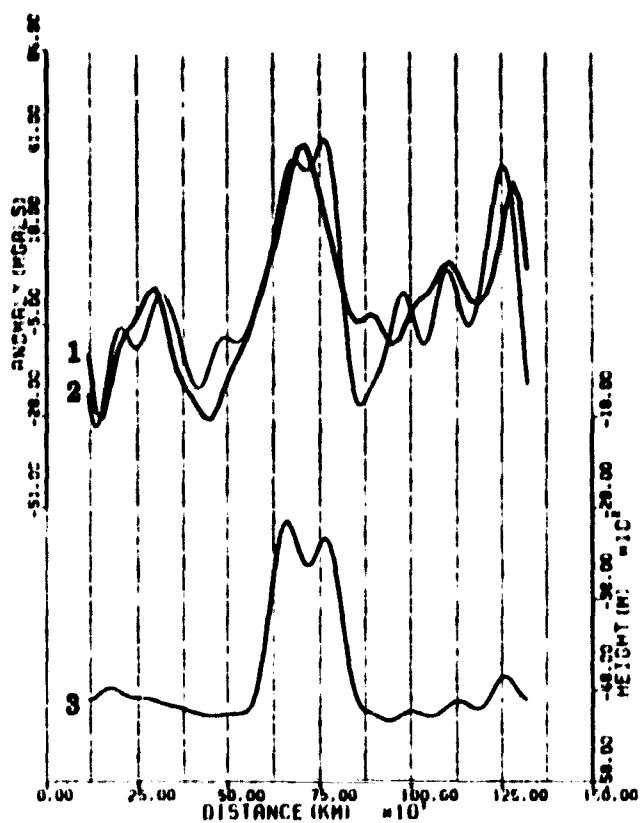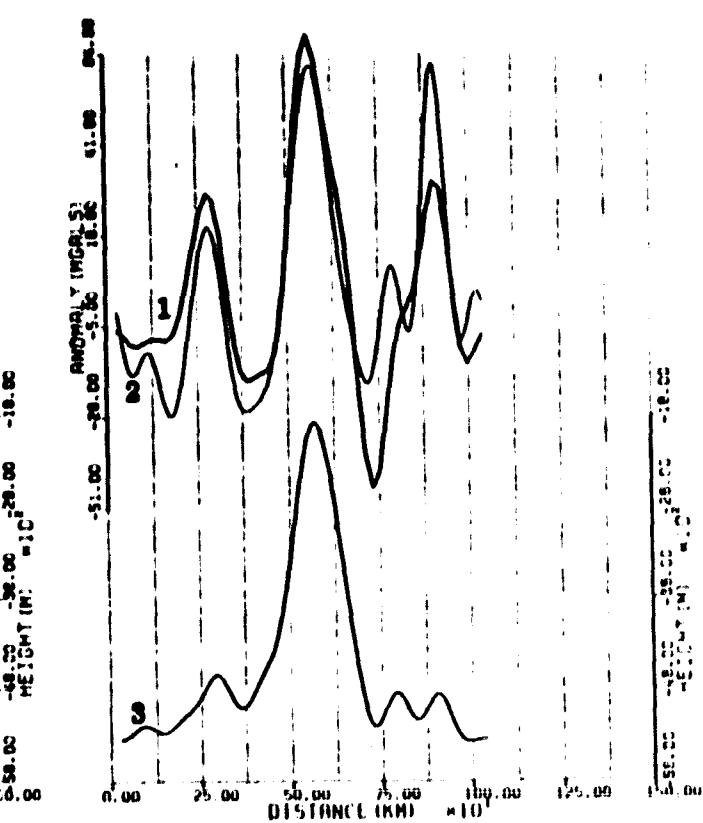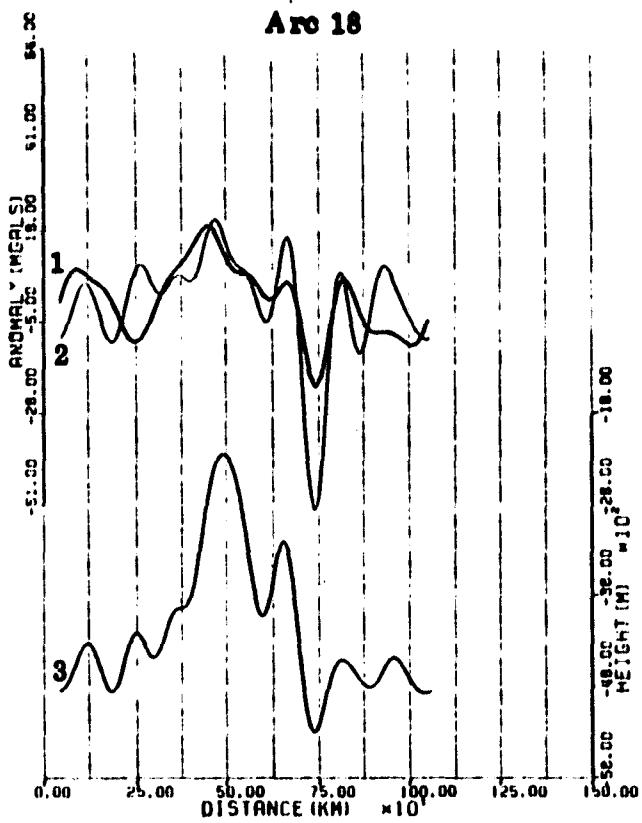
Figure 2.11. The Low-Pass Filtering
($\lambda \le 100$ km) on Altimeter Anomalies,
Measured Anomalies and Bathymetry
Along Arcs 16, 17, 18.

1. Altimeter Anomaly
2. Measured Anomaly
3. Bathymetry

Arc 16

Arc 17

Arc 18
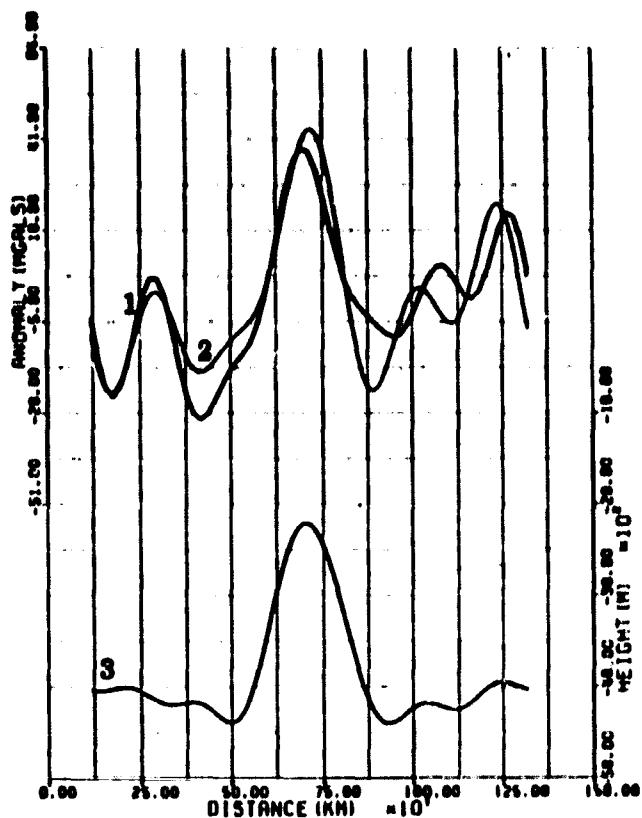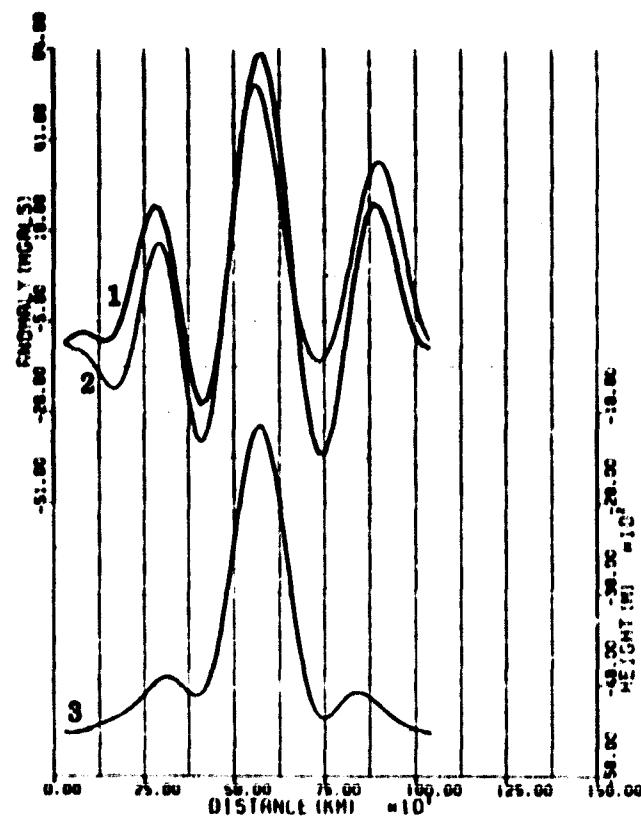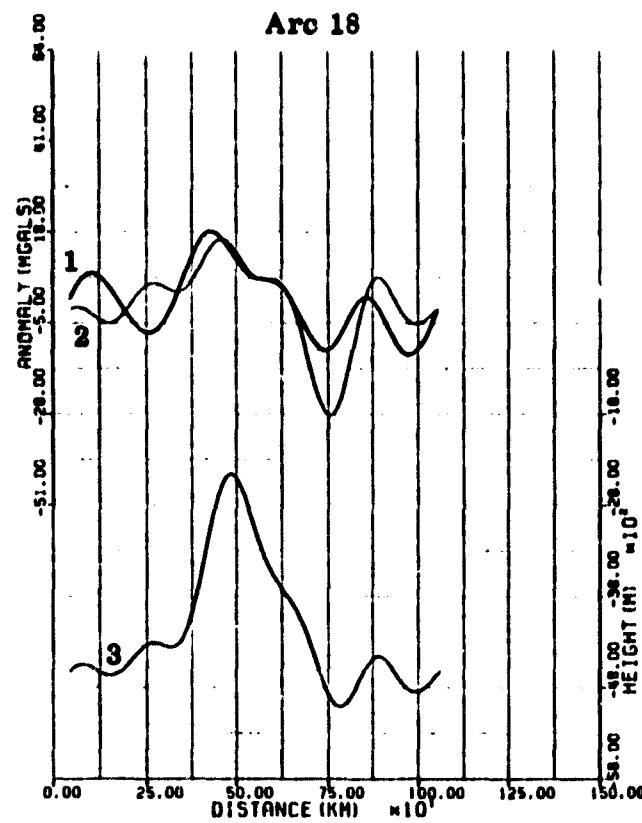
Figure 2.12. The Low-Pass Filtering
($\lambda \leq 200\,km$) on Altimeter Anomalies,
Measured Anomalies and Bathymetry
Along Arcs 16, 17, 18.

1. Altimeter Anomaly
2. Measured Anomaly
3. Bathymetry

are very small. The differences are small because the data vector of predictions reflects very little power contributions of high frequencies. We also see from Table 2.11 that the mean of RMS differences between predicted and measured anomalies is 21.1 mgals out of 6 sub-profiles 13, 14, 15, 16, 17, and 18. It is about 11.4 mgals when the power contributions from wavelengths below 200 km are filtered out.

3. Simple and Block Toeplitz Matrices

3.1 Simple Toeplitz Matrices

A Toeplitz matrix is an $(N+1, N+1)$ matrix $T_N = t_{kj}$, where $t_{kj} = t_{k-j}$ $k, j = 0, 1, \ldots, N$, that is to say that t_{kj} is a function of $(k-j)$ rather than of (k, j) separately. So T_N is a matrix of the following form

$$
T_N = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & \ldots & t_{-N} \\ t_1 & t_0 & t_{-1} & \ldots & t_{-(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ t_N & t_{N-1} & t_{N-2} & & t_0 \end{bmatrix} \tag{3.1}
$$

Covariance matrices of weakly stationary stochastic time series, and matrix representations of linear time-invariant discrete time filters are of Toeplitz form (Gray, 1977).

The very special structure of the T_N matrix can be exploited during the process of inversion, which yields significant savings in computational time and storage. In the past, several authors [(Levinson, 1947), (Trench, 1964), (Kutikov, 1967), (Zohar, 1969), etc.] have approached this problem and given algorithms to carry out the inversion of general Toeplitz matrices.

Since we generally deal with 'positive definite and symmetric normal equations' in least-squares adjustment and/or collocation, only the inversion of symmetric and positive definite Toeplitz matrices will be considered here.

Let us now consider the least-squares collocation model

$$
x = A X + s' + n \tag{3.2.a}
$$

which yields the following expression for the s-signal vector (Moritz, 1975, p. 15)

$$
s = C_{sx} \overline{C}^{-1}(x - A X), \quad \overline{C} = C_{s's'} + C_{nn} \tag{3.2.b}
$$

or equivalently, when the trend is removed before hand

$$
s = C_{sx} \overline{C}^{-1} x \tag{3.2.c}
$$

where s is the p-signal vector, x is the $(N+1)$-measurement vector, X is the u-parameter vector with the design matrix A, C_{sx} is the $[p, (N+1)]$-crosscovariance matrix between s and x, $C_{s's'}$ is the covariance matrix between s' signals, C_{nn} is the covariance matrix of the noise, and $\overline{C}$ is the $[(N+1), (N+1)]$ covariance matrix of the measurements x. If the number of observations is large, then the classical inversion of $\overline{C}$ in eqn. (3.2) causes the greatest problem. The $\overline{C}$ matrix is of Toeplitz form if we assume:

- observations are at equally spaced points along a profile
- noises are equal
- the covariance function used in the computation of $\bar{C}$ has the property $(C_{s's'})_{jk} = (C_{s's'})_{j-k}$, this implies the stationarity of the covariance function.

With the assumptions above, it follows that:

(1) $T_N = \bar{C}$ is non-singular

(2) T_N is symmetric, i.e. $t_{kj} = t_{jk}$

(3) $\sum\limits_{k,j=0}^{n} a_k\, t_{kj}\, a_j > 0$, $\quad n = 0, 1, \ldots, N$ for any set a_m, $\;m = 0, 1, \ldots, n$ such that at least one of the a_k is not zero and finite.

A recursive procedure for the inversion of symmetric and positive definite T_N will be given here. This recursive inversion requires numerical computations proportional to $(N+1)^2$ compared to $(N+1)^3$ of classical bordering methods and the storing of only one row of the input T_N and the output T_N^{-1}.

3.2 Inversion of Simple Toeplitz Matrices

It is well known that the inverse of a symmetric and positive definite matrix is also symmetric and positive definite. Then it is possible to express the inverse T_N^{-1} of T_N as (Faddeyev, 1963)

$$T_N^{-1} = B^{\tau}\, A\, B \tag{3.3}$$

where

$$B = \begin{bmatrix} 1 & 0 & 0 & \ldots\ldots0 \\ b_{10} & 1 & 0 & \ldots\ldots0 \\ b_{20} & b_{21} & 1 & \ldots\ldots0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ b_{N0} & b_{N1} & b_{N2} & \ldots\ldots1 \end{bmatrix} \qquad A = \begin{bmatrix} a_{00}^2 & 0 & \ldots\ldots0 \\ 0 & a_{11}^2 & \ldots\ldots0 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ 0 & 0 & \ldots\ldots a_{NN}^2 \end{bmatrix} \tag{3.4}$$

In complete agreement with the equality above we can write (Kutikov, 1966)

$$A^{-1} = B\, T_N\, B^{\tau} \tag{3.5}$$

or equivalently,

$$\sum_{k=0}^{n} b_{nk} \cdot t_{kj} = 0, \quad j = 0, 1, \ldots, (n-1), \quad n = 1, \ldots, N \tag{3.6.a}$$

$$b_{nn} = 1, \quad n = 0, 1, \ldots, N \tag{3.6.b}$$

$$\sum_{k=0}^{n} b_{nk} \cdot t_{kn} = a_{nn}^{-2}, \quad n = 0, 1, \ldots, N \tag{3.6.c}$$

If we substitiute the following condition

$$t_{kj} = t_{k-j}, \quad k, j = 0, 1, \ldots, N$$

in the above equations, we obtain

$$\sum_{k=0}^{n} b_{nk} \cdot t_{k-j} = 0, \quad j = 0, 1, \ldots, (n-1), \quad n = 1, \ldots, N \tag{3.7.a}$$

$$b_{nn} = 1, \qquad n = 0, 1, \ldots, N \tag{3.7.b}$$

$$\sum_{k=0}^{n} b_{nk} \cdot t_{k-n} = a_{nn}^{-2}, \quad n = 0, 1, \ldots, N \tag{3.8}$$

Since T_N is non-degenerate by definition, b_{nk} values are uniquely defined by the relations above. We will show that (Ibid, p. 63) under conditions (1), (2), (3) we have the following recurrence relations:

$$b_{nn} = 1, \qquad\qquad n = 0, 1, \ldots, N \tag{3.9.a}$$

$$e_n = b_{nk} \cdot t_{k+1}, \qquad n = 0, 1, \ldots, (N-1) \tag{3.9.b}$$

$$a_{nn}^{-2} = b_{nk} \cdot t_{n-k} \qquad n = 0, 1, \ldots, N \tag{3.9.c}$$

$$b_{n+1,0} = -e_n \cdot a_{nn}^2, \qquad n = 0, 1, \ldots, (N-1) \tag{3.9.d}$$

$$b_{n+1,k} = b_{n,k-1} + b_{n+1,0} \cdot b_{n,n-k}, \quad n = 1, 2, \ldots, (N-1), \quad k = 1, \ldots, (n-1) \tag{3.9.e}$$

In order to show this relation let us write the left hand sides of eqn. (3.7.a) replacing n by $(n+1)$:

$$X_s = \sum_{k=0}^{n+1} b_{n+1,k} \cdot t_{k-s}, \qquad s = 0, 1, \ldots, n \tag{3.10}$$

and let us also assume that for the b_{mk} $k, m = 0, \ldots, n$ defined by conditions (3.9) the relations (3.7)-(3.8) are satisfied. Using eqn. (3.9) we can write

$$X_s = b_{n+1,0} \cdot t_{-s} + \sum_{k=1}^{n} b_{n+1,k} \cdot t_{k-s} + b_{n+1,n+1} \cdot t_{n+1-s}$$

$$= b_{n+1,0} \cdot t_{-s} + \sum_{k=1}^{n} b_{n,k-1} \cdot t_{k-s} + b_{n+1,0} \sum_{k=1}^{n} b_{n,n-k} \cdot t_{k-s} + b_{n+1,n+1} \cdot t_{n+1-s}$$

$$= b_{n+1,0} \left[t_{-s} + \sum_{k=1}^{n} b_{n,n-k} \cdot t_{k-s} \right] + \left[\sum_{m=1}^{n} b_{n,m-1} \cdot t_{m-s} + b_{n+1,n+1} \cdot t_{n+1-s} \right]$$

using the equality (3.9.a) and substituting $m = k + 1$, we obtain

$$X_s = b_{n+1,0} \left[b_{0,0} t_{-s} + \sum_{k=1}^{n} b_{n,n-k} \cdot t_{k-s} \right] + \left[\sum_{k=0}^{n-1} b_{n,k} t_{k-s+1} + b_{n,n} t_{n+1-s} \right]$$

$$X_s = b_{n+1,0} \sum_{k=0}^{n} b_{n,n-k} \cdot t_{k-s} + \sum_{k=0}^{n} b_{n,k} \cdot t_{k-s+1}, \quad s = 0, 1, \ldots, n \tag{3.11}$$

By virtue of eqn. (3.7.a), the right-hand sides of the above equalities are equal to zero for $s = 1, \ldots, n$ and by (3.9.b)-(3.9.d) for $s = 0$ also.

Thus when we consider eqn. (3.7.a), then the eqns. (3.7) - (3.8) follow from the conditions (2.9) by induction. Since b_{nk} $n = 0, 1, \ldots, N$ and $k = 0, \ldots, n$ are

unique, we conclude that eqns. (3.7) and (3.8) are equivalent to eqn. (3.9).

Let r_{kj}^N $k, j = 0, 1, \ldots, N$ denote the elements of the inverse T_N^{-1} such that

$$
T_N^{-1} = \begin{bmatrix}
r_{00}^N & r_{01}^N & \cdots\cdots & r_{0N}^N \\
r_{10}^N & r_{11}^N & \cdots\cdots & r_{1N}^N \\
\cdot & \cdot & & \cdot \\
\cdot & \cdot & & \cdot \\
\cdot & \cdot & & \cdot \\
r_{N0}^N & r_{N1}^N & \cdots\cdots & r_{NN}^N
\end{bmatrix}
\tag{3.12}
$$

then the following equalities hold (Kutikov, p. 66)

$$
r_{kj}^{(N+1)} = r_{kj}^N + b_{n+1,k}\, a_{N+1,N+1}^2\, b_{N+1,j} \qquad k, j = 0, 1, \ldots, (N+1) \tag{3.13}
$$

This follows from eqn. (3.3), where $r_{N+1,j}^N$ is taken to be equal to zero.

Since the Toeplitz matrix T_N satisfying the conditions (1), (2) and (3) is symmetric and persymmetric (i.e. it has symmetry both its diagonal and cross-diagonal), in other words

$$
t_{kj} = t_{jk}, \quad t_{kj} = t_{N-j,N-k}
$$

we can show that the inverse T_N^{-1} is also symmetric and persymmetric (Zohar, 1969). The symmetricity of the inverse is apparent, so, we will only show that T_N^{-1} is persymmetric. For the proof let us introduce the Exchange matrix E defined as a square matrix with units along the cross-diagonal, and zeros elsewhere. Now suppose A is an arbitrary $(N \times N)$ matrix. If we examine the matrix product $(E\, A^T\, E)$, we see that the overall effect is to exchange elements of A which are located symmetrically with respect to the cross-diagonal. So a persymmetric matrix is a matrix H satisfying $E\, H^T\, E = H$. Now from this formulation it is easy to show that T_N^{-1} is persymmetric. Let us start with

$$
(T_N^{-1})(T_N^T) = I
$$

and note that $E\, E = I$ to obtain

$$
E\,(T_N^T)^{-1}\,(E\,E)\,(T_N^T)\,E = E\,E = I
$$

$$
[\,E\,(T_N^T)^{-1}\,E\,]\,[\,E\,T_N^T\,E\,] = I
$$

but $E\, T_N^T\, E = T_N$. Hence $[\,E\,(T_N^T)^{-1}\,E\,] = T_N^{-1}$ and T_N^{-1} is persymmetric.

Toeplitz inverses have a very useful property: If the last row of the inverse, say r_{Nj}^N, $j = 0, 1, \ldots, N$, is known, then all other elements of the inverse can be computed iteratively from r_{Nj}^N. First let us compute r_{Nj}^N by writing from eqn. (3.13)

$$
r_{kj}^N = r_{kj}^{(N-1)} + b_{N,k}\, a_{NN}^2\, b_{N,j}
$$

-79-

and letting $k = N$ and also noting $r_{N,j}^{(N-1)} = 0$, $j = 0, 1, \ldots, n$, $b_{NN} = 1$, we obtain

$$r_{N,j}^{N} = a_{NN}^2 \, b_{N,j} \qquad\qquad j = 0, 1, \ldots, N \qquad\qquad (3.14)$$

In order to compute the remaining elements of the inverse we have from eqn. (3.13)

$$r_{k-1,j-1}^{N} = r_{k-1,j-1}^{N-1} + b_{N,k-1} \, a_{NN}^2 \, b_{N,j-1} \qquad\qquad (3.15)$$

and also considering persymmetry

$$r_{k,j}^{N} = r_{N-j,N-k}^{N} = r_{N-j,N-k}^{N-1} + b_{N,N-j} \, a_{NN}^2 \, b_{N,N-k}$$

$$= r_{k-1,j-1}^{N-1} + b_{N,N-j} \, a_{NN}^2 \, b_{N,N-k} \qquad\qquad (3.16)$$

Substitute the equality of $r_{k-1,j-1}^{N-1}$ from eqn. (3.16) in eqn. (3.15) to obtain

$$r_{k-1,j-1}^{N} = r_{k,j}^{N} - b_{N,N-j} \, a_{NN}^2 \, b_{N,N-k} + b_{N,k-1} \, a_{NN}^2 \, b_{N,j-1} \qquad\qquad (3.17)$$

By considering symmetry and persymmetry, the algorithm can be summarized as follows:

(1) Compute the last row of the B matrix and a_{NN}^2 through the recursive procedure defined by eqn. (3.9);

$$b_{nn} = 1, \qquad\qquad\qquad\qquad n = 0, 1, \ldots, N$$

$$e_n = \sum_{k=0}^{n} b_{nk} \cdot t_{k+1}, \qquad\qquad n = 0, 1, \ldots, (N-1)$$

$$a_{nn}^{-2} = \sum_{k=0}^{n} b_{nk} \cdot t_{n-k}, \qquad\qquad n = 0, 1, \ldots, N \qquad\qquad (3.18)$$

$$b_{n+1,0} = -e_n \cdot a_{nn}^2, \qquad\qquad n = 0, 1, \ldots, (N-1)$$

$$\left. \begin{aligned} b_{n+1,j} &= b_{n,j-1} + b_{n+1,0} \, b_{n,n-j} \\[4pt] b_{n+1,n-j+1} &= b_{n,n-j} + b_{n+1,0} \, b_{n,k-1}, \end{aligned} \right\} \quad \begin{aligned} &n = 1, 2, \ldots, (N-1) \\ &j = 1, 2, \ldots, ((n-1)/2) \end{aligned}$$

By the formulae above the reduction in computer storage is considerable, because $b_{n+1,j}$, $b_{n+1,n-j+1}$ following the calculation can be stored in the place of $b_{n,j-1}$, $b_{n,n-k}$, which are not used in any further calculation.

(2) Compute $r_{N,j}^{N}$, $j = 0, 1, \ldots, N$ and the remaining elements of the inverse by eqns. (3.14) and (3.17);

$$r_{N,j}^{N} = a_{NN}^{2} \cdot b_{N,j}$$

$$r_{jN}^{N} = r_{N-j,0}^{N} = r_{0,N-j}^{N} = r_{N,j}^{N} \text{ from symmetry and}$$
$$\left. \right\} \quad j = 0, 1, \ldots, N \qquad (3.19)$$
persymmetry

$$r_{k-1,j-1}^{N} = r_{k,j}^{N} - b_{N,N-k} \cdot a_{NN}^{2} \cdot b_{N,N-j} + b_{N,k-1} \cdot a_{NN}^{2} \cdot b_{N,j-1}, \quad k = (N_{H}+1), \ldots, N, \quad j = (N-k), \ldots, k$$

$$\text{where } N_{H} = \begin{cases} (N+1)/2 & \text{if } N \text{ odd} \\ N/2 & \text{if } N \text{ even} \end{cases}$$

finally $r_{j-1,k-1}^{N} = r_{N-j+1,N-k+1}^{N} = r_{N-k+1,N-j+1}^{N} = r_{k-1,j-1}^{N}$ from symmetry and persymmetry.

As we see from the expressions above, we do not have to store all the elements of the inverse at once. As soon as an element of the inverse is computed it is multiplied by the corresponding element in x-vector of our model (3.2.2). So we only have to store one row of the inverse in order to solve the recursive equation (3.17). A computer program written in FORTRAN language is given in Appendix 3.A.

Using this algorithm some test runs have been made in double precision arithmetic on Amdahl-470 computer. The results are given in Table 3.1. for the solution of the linear equality

$$T_{N} F = X \qquad (3.20)$$

where T_{N} is a $[(N+1),(N+1)]$ Toeplitz matrix whose inverse has been computed row by row, X is the data vector of length $(N+1)$ and F is the solution vector of length $(N+1)$ we desire.

Table 3.1. CPU Time and Core Storage for the Inversion of Simple Toeplitz Matrices.

Dimension (N+1)	C P U Time (sec.)	Storage (K)
100	0.12	3.2
200	0.48	6.4
400	2.86	12.8
1000	18.21	32.0

From the Table 3.1 we see that the simple Toeplitz inversion is very efficient as far as time and storage are concerned. If we have equally spaced data with equal noises, then the covariance matrix of the data is of Toeplitz form and we obtain identical solutions with the classical inversion algorithms. But if the observations are not equally spaced and do not have equal noises, then we have to use interpolated data and give equal noises to the observations in order to use the algorithm above. Thus we introduce some approximations to our solution. We can proceed and use this algorithm if we can show that the degree of approximation has negligible effect on the solution vector. To demonstrate the efficiency of this algorithm and to see the effect of approximation on the solution, two arcs

of altimetry described in Table 3.2 and shown in Figure 2.2 from GEOS-3 have been examined.

Table 3.2. The Description of Arcs 11 and 12.

Arc No.	Area	Latitudes (Deg.) φ_1	φ_2	Longitudes (Deg.) λ_1	λ_2	Length (km)	No. of Observations
11	Calibration	12.8480	39.9526	309.9578	291.0051	3540	256
12	Philippines	-0.0241	32.4655	146.5301	126.8229	4162	300

As the first step, the point gravity anomalies have been predicted at observation points by eqn. (3.2.c) using the original observations and their uncertainties. As the second step, equally spaced data have been created from the original observations and the mean uncertainty of the original observations along the particular arc has been given to each predicted data point as the uncertainty. As the third step the point gravity anomalies have been computed at the original observation points by eqn. (3.2.c), but using the simple Toeplitz inversion algorithm to invert $\overline{C}$. The statistics of this study are given in Table 3.3 and anomalies are plotted in Figures 3.1 and 3.2 for the 11th and 12th arc respectively.

Table 3.3. The Statistics of Arcs 11 and 12.

Arc No.	RMS (mgals)	Max. Diff. (mgals)	Mean Diff. (mgals)
11	1.1	5.5	-0.1
12	4.8	32.4	-0.3

The average standard deviation of the predicted point anomalies is about 27 mgals. The RMS differences of 1.1 mgals for the eleventh arc, and 4.8 mgals for the twelfth arc above are much small than the standard deviations of the predictions. So it would not be unfair to say that the approximations introduced in the examples above have negligible effects on the computed point anomalies.

The twelfth arc of the Table 3.2 was also used for ($1° \times 1°$) mean anomaly recovery of some 25 blocks, which are located in a ($5° \times 5°$) block whose coordinates are:

$$\varphi_N = 10°, \quad \varphi_S = 5°, \qquad \lambda_E = 137°, \quad \lambda_W = 132°$$

The RMS difference between a rigorous least-squares collocation solution and Toeplitz solution has been found to be only 0.1 mgals with a maximum difference 0.2 mgals. The difference between the two solutions is negligible should we consider the standard deviations to the predictions, which is about 15 mgals.

Figure 3.1. Altimeter Anomaly Using Toeplitz Algorithm and Conventional Collocation Along Arc 11.

O Conventional Collo-
cation (Dark Line)
✱ Toeplitz Algorithm
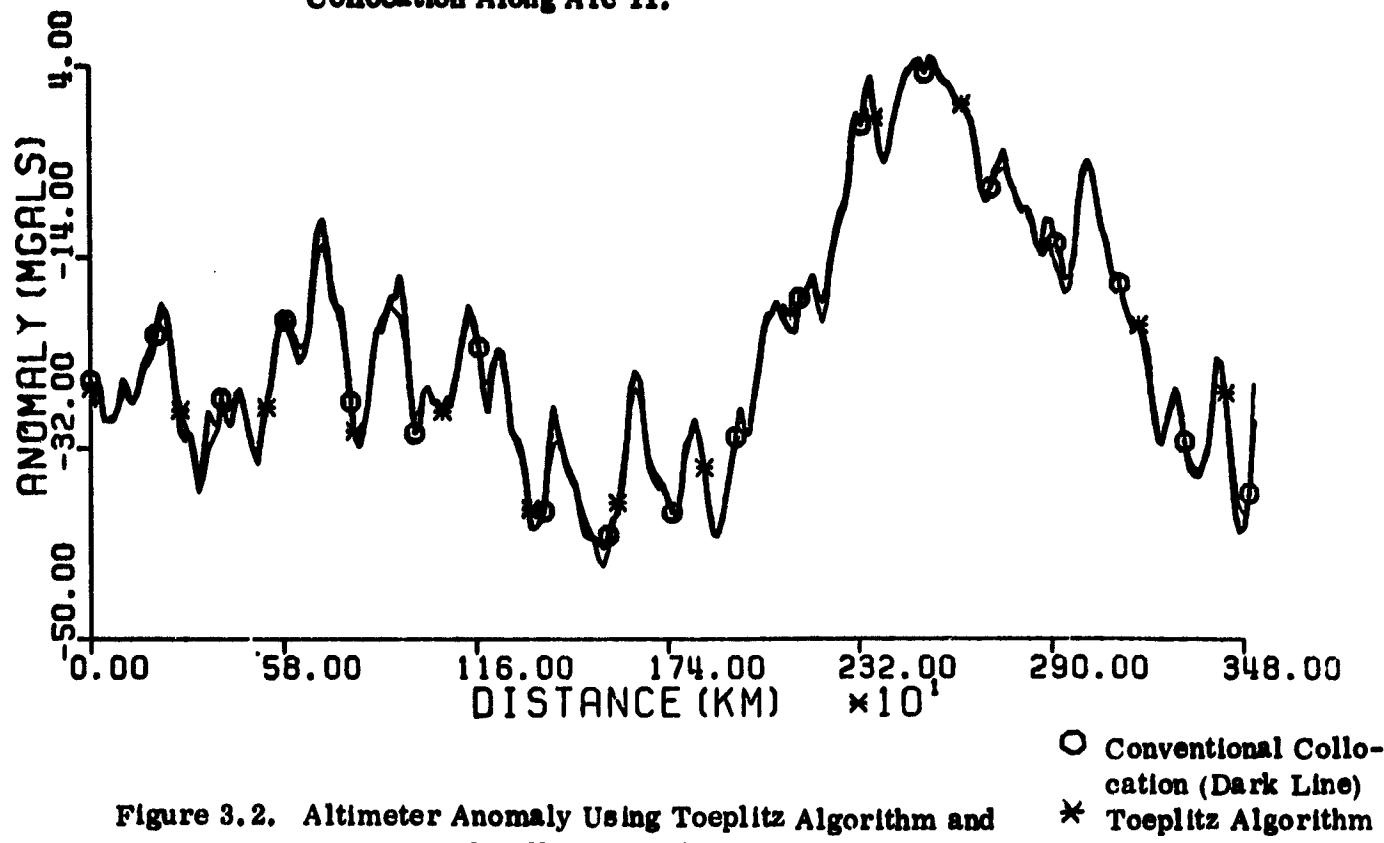
Figure 3.2. Altimeter Anomaly Using Toeplitz Algorithm and Conventional Collocation Along Arc 12.
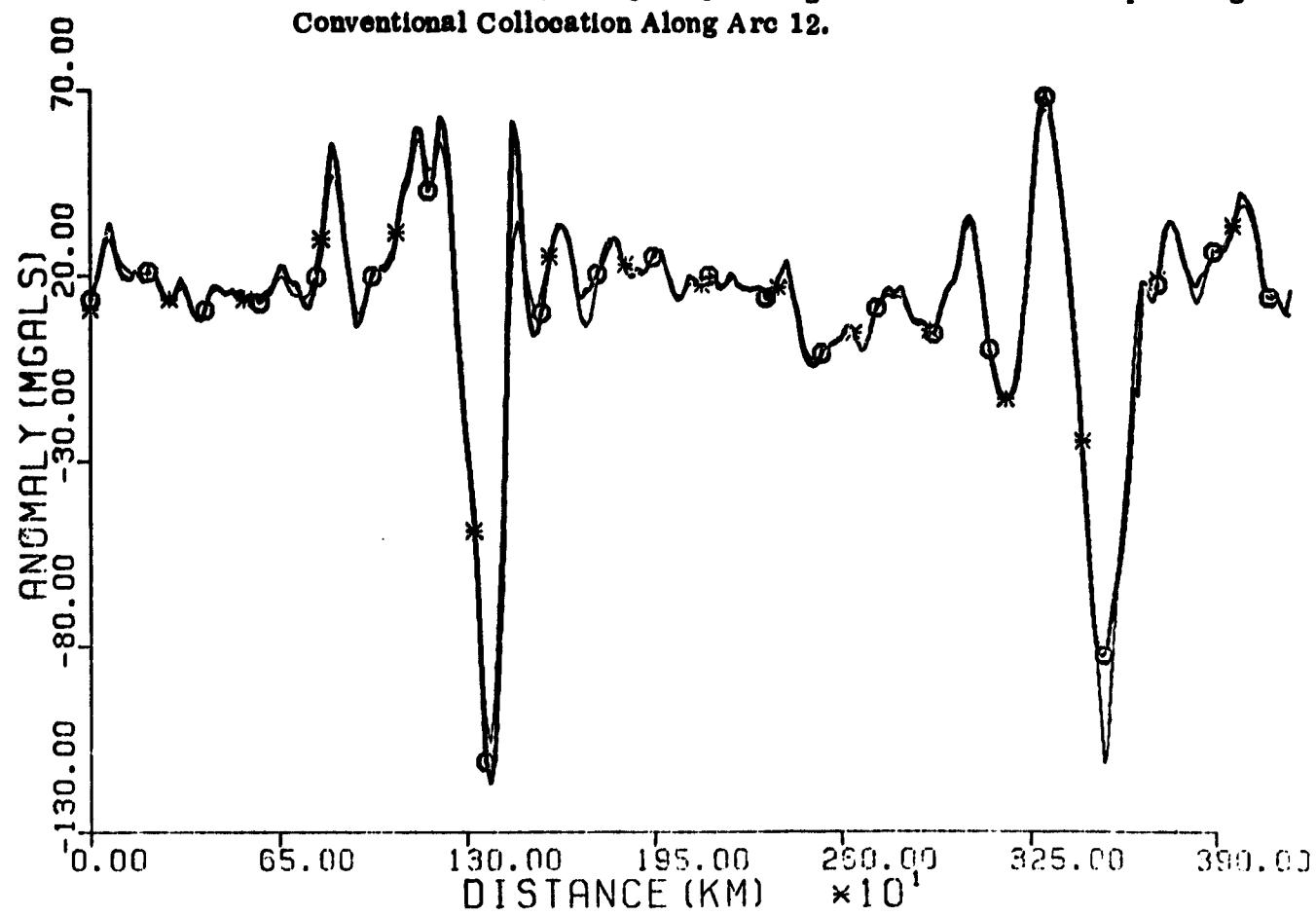
3.3 Block-Toeplitz Matrices

We have defined the simple Toeplitz matrix as a matrix T_N whose (k,j)th element t_{kj} is a function of $(k-j)$. When t_{kj} itself is a matrix, then T_N is called a block-Toeplitz matrix. In this section our attention will be restricted to the case when the block-Toeplitz is real, symmetric, persymmetric and positive definite. The derivations are very similar with the simple Toeplitz case, the only difference being the replacement of scalars by matrices. Thus we generalize the algorithm derived in the previous section. A complete derivation for the inversion of positive definite block-Toeplitz matrices can be found in Kutikov (1966).

Block-Toeplitz matrices appear quite often in time series analysis. The simple Toeplitz forms usually arise in one-dimensional Wiener filtering, whereas block-Toeplitz forms arise in two-dimensional filtering. Suppose x_s, $s = 0, 1, \ldots, p$ are random vectors, each vector having length of $(q+1)$, then the autocovariance matrix (stationarity also assumed) is of block-Toeplitz form with dimensionality $[(p+1)(q+1), (p+1)(q+1)]$. We can see that $(q+1)$ is the dimension of the square subblocks and $(p+1)$ is the number of random vectors.

3.4 Inversion of Block-Toeplitz Matrices

Development of the Algorithm

The algorithm given for the simple Toeplitz case is generalized here as follows:

Simple Toeplitz Case		Block Toeplitz Case	
	T_N		T_N^s
$t_{kj} = t_{k-j},$	$k, j = 0, 1, \ldots, N$	$T_{st} = T_{s-t},$	$s, t = 0, 1, \ldots, p$
	B		B^s
$b_{kj},$	$k, j = 0, 1, \ldots, N$	$B_{st},$	$s, t = 0, 1, \ldots, p$
$a_{nn}^{-2}, a_{nn}^{2},$	$n = 0, 1, \ldots, N$	A_{ss}^{-2}, A_{ss}^{2}	$s = 0, 1, \ldots, p$
$e_n,$	$n = 0, 1, \ldots, (N-1)$	$E_s,$	$s = 0, 1, \ldots, (p-1)$
r_{kj}^{N}	$k, j = 0, 1, \ldots, N$	$R_{st},$	$s, t = 0, 1, \ldots, p$

Now we can write the iterative solution algorithm for the inversion of the block Toeplitz matrices corresponding to the eqns. (3.18) and (3.19) of simple Toeplitz inversion.

(1) $\quad B_{ss} = I \qquad\qquad s = 0, 1, \ldots, p$

$$E_s = \sum_{t=0}^{s} B_{st} T^s_{t+1}, \quad s = 0, 1, \ldots, (p-1)$$

$$A^{-s}_{ss} = \sum_{t=0}^{s} B_{st} T^s_{s-t}, \quad s = 0, 1, \ldots, p$$

$$B_{s+1,0} = -E_s A^s_{ss} \qquad s = 0, 1, \ldots, (p-1)$$ (3.21)

$$\left.\begin{array}{l} B_{s+1,t} = B_{s,t-1} + B_{s+1,0} B_{s,s-t} \\[4pt] B_{s+1,s-t+1} = B_{s,s-t} + B_{s+1,0} B_{s,t-1} \end{array}\right\} \begin{array}{l} s = 1, 2, \ldots, (p-1) \\[4pt] t = 1, 2, \ldots, (s-1)/2 \end{array}$$

(2) $\quad R_{pt} = A^s_{pp} B_{pt} \qquad t = 0, 1, \ldots, p$

$$\left.\begin{array}{l} R_{tp} = R'_{pt} \\[6pt] R_{p-t,0} = R'_{pt} \\[6pt] R_{0,p-t} = R_{pt} \end{array}\right\} \quad \text{[from symmetry and persymmetry]}$$ (3.22)

$$R_{s-1,t-1} = R_{st} - B'_{p,p-s} A^s_{pp} B_{p,p-t} + B'_{p,s-1} A^s_{pp} B_{p,t-1}$$

$$s = (p_M+1), \ldots, p, \quad t = (p-s), \ldots, s \quad \text{where} \quad p_M = \begin{cases} (p+1)/2 & \text{if } p \text{ odd} \\ p/2 & \text{if } p \text{ even} \end{cases}$$

$$\left.\begin{array}{l} R_{t-1,s-1} = R_{p-t+1,p-s+1} = R'_{s-1,t-1} \\[6pt] R_{p-s+1,p-t+1} = R_{s-1,t-1} \end{array}\right\} \quad \text{[from symmetry and persymmetry]}$$

Similar to the simple Toeplitz case the saving in computational time and in computer storage is very significant. We need only store one block-row of the T^s_N and one block-row of the inverse $(T^s_N)^{-1}$ in cases where we are not interested in the inverse itself but rather in the solution. A computer program for the solution of (3.23) written in FORTRAN language is given in Appendix 3.B.

The computational time by this algorithm is proportional to $p^2 (q+1)^3$ compared to the $p^3 (q+1)^3$ of the classical method of bordering. Some test calculations on the Amdahl-470 yielded the results given in Table 3.4 in the solution of the linear equation

$$T^s_N F = X, \quad N+1 = (q+1) \cdot (p+1) \tag{3.23}$$

where T^s_N is the block-Toeplitz matrix of dimensionality $[(n+1), (N+1)]$ consisting of $[(q+1), (q+1)]$ subblocks, F is the solution vector of length $(N+1)$ and X is the data vector of length $(N+1)$. Thus for the solution vector we have,

$$F = (T_N^s)^{-1} X \qquad (3.24)$$

Table 3.4. CPU Time and Core Storage for the Inversion
of Block-Toeplitz Matrices.

Dimension			CPU Time	Storage
P+1	Q+1	N+1	(sec.)	(K)
25	2	50	0.24	3
100	2	200	4.42	10
200	2	400	13.25	20
80	5	400	15.80	40
20	20	400	48.85	93
400	2	800	49.68	40

We will now show that the covariance matrix of equally spaced gridded data is of block-Toeplitz form. If data are gridded and observations on each profile are equally spaced, then distances between profiles do not have to be equal to each other in order to obtain a block-Toeplitz form. As an example let us consider the three profiles shown in Figure 3.3. The covariance matrix of this model is of block-Toeplitz form and is given below.
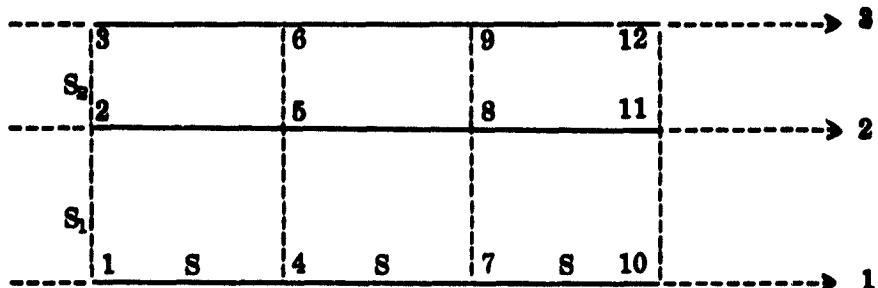
Figure 3.3. Three Parallel Profiles with Equally Spaced
and Gridded Observations.

$$T = \begin{bmatrix} T_0 & T_1 & T_2 & T_3 \\ T_1 & T_0 & T_1 & T_2 \\ T_2 & T_1 & T_0 & T_1 \\ T_3 & T_2 & T_1 & T_0 \end{bmatrix}$$

where

$$T_0 = \begin{bmatrix} C(0) & C(s_1) & C(s_1 + s_2) \\ & C(0) & C(s_2) \\ \text{symmetric} & & C(0) \end{bmatrix}$$

$$T_1 = \begin{bmatrix} C(s) & C(\sqrt{s_1^2 + s^2}) & C(\sqrt{s^2 + (s_1 + s_2)^2}) \\ & C(s) & C(\sqrt{s^2 + s_2^2}) \\ \text{symmetric} & & C(s) \end{bmatrix}$$

$$T_2 = \begin{bmatrix} C(2s) & C(\sqrt{4s^2 + s_1^2}) & C(\sqrt{4s^2 + (s_1 + s_2)^2}) \\ & C(2s) & C(\sqrt{4s^2 + s_2^2}) \\ \text{symmetric} & & C(2s) \end{bmatrix}$$

$$T_3 = \begin{bmatrix} C(3s) & C(\sqrt{9s^2 + s_1^2}) & C(\sqrt{9s^2 + (s_1 + s_2)^2}) \\ & C(3s) & C(\sqrt{9s^2 + s_2^2}) \\ \text{symmetric} & & C(3s) \end{bmatrix}$$

In order to elaborate the applicability of the block-Teoplitz inversion in practice, it was used in the recovery of point and $(1° \times 1°)$ mean anomalies from geoid heights. Point anomalies predicted from gridded geoid heights, which are created from the adjusted satellite altimeter data, are shown in Tables 3.5 and 3.6. The computation points of Table 3.5 are located on 90-E Ridge, and those of Table 3.6 are located in the Bovin Trench Area. These locations have been selected due to their large anomaly variations. $(1° \times 1°)$ mean anomalies predicted from gridded geoid heights are shown in Tables 3.7, 3.8, and 3.9. The $(1° \times 1°)$ computation blocks of the Tables 3.7 and 3.8 are located in the Philippines Trench, and those of Table 3.9 are located in the Calibration area.

GANOM and CANOM are common in all the Tables and have the following meanings:

GANOM: Gravity anomaly computed from GEM 9 potential coefficients,
CANOM: Gravity anomaly computed from original adjusted satellite altimeter data using the rigorous least-squares collocation.

We see from Table 3.5 that the RMS differences between the gravity anomalies computed from the original adjusted altimeter data and those from the gridded data (predicted) are 8.0 and 3.0 mgals respectively, when using (6 x 65) and (10 x 40) grids. These RMS differences are well below the standard deviations of the predictions themselves if we consider the standard deviations of predictions to be 27 mgals as before. However, the RMS differences in Table 3.6 are 27.8, 16.9, and 17.8 mgals in the case of (5 x 72), (10 x 36), and (18 x 20) gridded data, respectively. Only one of the RMS differences is slightly higher than the 27 mgals standard deviation of predictions considered before.

We also computed $(1° x 1°)$ mean anomalies of blocks 710, 711, and 343 given in Tab'es 3.7, 3.8, and 3.9 respectively. The RMS differences between mean anomalies computed from original altimeter data and (10 x 40) gridded data are 5.9, 6.2, 2.6 mgals respectively. The standard deviation of $(1° x 1°)$ mean anomaly prediction is about 7 mgals. So the RMS differences above are smaller than the standard deviation.

From the results above, we can conclude that, in the case of altimeter data, the RMS differences between predictions from original data and gridded data are generally less than the standard deviation of predictions. Thus, we can create a gridded data and predict signals by the fast Toeplitz algorithms described in this chapter.

Table 3.5. Point Anomalies from GEM 9 Coefficients, Original Geoid Heights and Gridded Geoid Heights.

LAT	LON	CANOM	TANOM1	TANOM2	CANOM	CANOM-TANOM1	CANOM-TANOM2
-11.00	87.00	-50.4	-34.5	-26.4	-23.2	11.1	3.4
-11.00	87.20	-20.8	-38.6	-31.6	-27.6	11.0	4.5
-11.00	87.40	-20.2	-41.9	-37.8	-38.9	0.0	1.3
-11.00	87.60	-20.1	-42.3	-41.3	-41.4	0.0	
-11.00	87.80	-20.0	-38.9	-40.2	-41.8	-2.3	
-11.00	88.00	-19.9	-32.4	-32.7	-34.4	-2.0	
-11.00	88.20	-19.8	-20.2	-18.7	-17.2	0.0	1.3
-11.00	88.40	-19.7	-13.4	-2.1	4.1	6.6	5.3
-11.00	88.60	-19.6	-0.6	10.0	15.8	9.1	
-11.00	88.80	-19.5	-1.4	11.8	11.6	10.0	
-11.00	89.00	-19.4	-7.7	1.5	-0.6	6.8	
-11.00	89.20	-19.3	-19.5	-18.1	-15.4	3.1	
-11.00	89.40	-19.2	-31.5	-37.2	-33.7	-1.5	
-11.00	89.60	-19.1	-40.6	-43.4	-47.4	-6.0	
-11.00	89.80	-19.1	-36.6	-33.8	-34.7	-0.1	
-11.00	90.00	-19.0	-30.3	-33.6	-34.8	-4.5	
-11.00	90.20	-18.9	-31.3	-34.1	-33.8	-2.0	
-11.00	90.40	-18.8	-49.6	-33.7	-49.0	0.6	
-11.00	90.60	-18.7	-45.0	-33.6	-33.5	1.8	
-11.00	90.80	-18.6	-37.8	-36.4	-37.7	0.1	
-11.00	91.00	-18.5	-30.0	-33.2	-31.7	-1.5	

RMS Difference Between CANOM and TANOM1 is 8.0 mgals
RMS Difference Between CANOM and TANOM2 is 3.0 mgals

where, TANOM1 : the point anomaly computed from (6 x 65) gridded geoid heights.

TANOM2 : the point anomaly computed from (10 x 40) gridded geoid heights.

The grid limits for prediction described in this table are:

$$\varphi_N = -8^\circ, \quad \varphi_S = -14^\circ, \quad \lambda_E = 92^\circ, \quad \lambda_W = 86^\circ$$

Table 3.6. Point Anomalies from GEM 9 Coefficients, Original Geoid Heights and Gridded Geoid Heights.

LAT	LON	CANOM	TANOM1	TANOM2	TANOM3	CANOM	CANOM-TANOM1	CANOM-TANOM2	CANOM-TANOM3
28.00	141.00	13.7	-14.7	-12.8	-5.4	0.5	15.2	13.3	5.9
28.00	141.10	13.6	-22.5	-13.7	-5.0	9.7	32.2	23.4	14.7
28.00	141.20	13.6	-24.0	-5.5	1.4	25.2	49.2	30.7	23.8
28.00	141.30	13.6	-14.1	14.0	16.8	45.6	59.7	31.6	28.8
28.00	141.40	13.6	11.5	45.3	49.3	70.2	58.7	24.9	20.9
23.00	141.50	13.5	54.1	87.1	92.9	98.9	44.8	11.8	6.0
23.00	141.60	13.5	109.7	134.5	138.8	131.3	21.6	-3.2	-7.5
23.00	141.70	13.5	170.0	181.0	183.4	167.0	-3.0	-14.0	-16.4
28.00	141.80	13.4	225.6	221.8	223.6	204.5	-.1	-17.3	-19.1
28.00	141.90	13.4	269.8	254.1	257.1	241.0	-27.8	-13.1	-16.1
28.00	142.00	13.4	298.7	274.6	277.3	272.2	-26.5	-2.4	-5.1
23.00	142.10	13.3	308.1	281.6	281.9	290.6	-17.5	9.0	8.7
28.00	142.20	13.3	298.5	273.6	275.2	289.4	-9.1	15.8	14.2
28.00	142.30	13.2	265.7	251.2	254.3	272.4	6.7	21.2	18.1
28.00	142.40	13.2	212.4	216.1	213.9	243.3	30.9	27.2	29.4
23.00	142.50	13.2	131.6	172.3	164.4	203.7	52.1	31.4	39.3
28.00	142.60	13.1	103.1	124.0	113.5	155.1	52.0	31.1	41.6
28.00	142.70	13.1	68.3	75.6	73.5	100.0	31.7	24.4	26.5
23.00	142.80	13.0	34.3	27.2	35.7	41.7	7.4	14.5	6.0
28.00	142.90	13.0	-5.7	-20.7	-3.4	-15.7	-10.0	5.0	-12.3
23.00	143.00	12.9	-55.7	-66.0	-48.7	-67.2	-11.5	-1.2	-18.5
23.00	143.10	12.8	-103.0	-103.4	-91.2	-109.0	-6.0	-5.6	-17.8
28.00	143.20	12.8	-133.9	-129.2	-124.1	-138.2	-4.3	-9.0	-14.1
23.00	143.30	12.7	-148.0	-141.9	-139.9	-152.7	-4.7	-10.8	-12.8
23.00	143.40	12.6	-149.9	-143.0	-142.9	-151.7	-1.8	-8.7	-8.8
23.00	143.50	12.6	-141.4	-134.3	-136.2	-138.4	3.0	-4.1	-2.2
23.00	143.60	12.5	-123.9	-118.0	-120.3	-117.7	6.2	6.3	2.6
28.00	143.70	12.4	-99.1	-95.8	-97.4	-94.5	4.6	1.3	2.9
23.00	143.80	12.4	-68.8	-70.4	-72.4	-71.2	-2.4	-0.8	1.2
28.00	143.90	12.3	-36.6	-44.8	-47.7	-49.1	-12.5	-4.3	-1.4
23.00	144.00	12.2	-8.8	-22.2	-25.0	-29.2	-20.4	-7.0	-4.2

RMS Difference Between CANOM and TANOM1 is 27.8 mgals
RMS Difference Between CANOM and TANOM2 is 16.9 mgals
RMS Difference Between CANOM and TANOM3 is 17.8 mgals

where, TANOM1 : the point anomaly computed from (5 x 72) gridded geoid heights.

TANOM2 : the point anomaly computed from (10 x 36) gridded geoid heights.

TANOM3 : the point anomaly computed from (18 x 20) gridded geoid heights.

The grid limits for prediction described in this table are:

$$\varphi_N = 30^\circ, \quad \varphi_S = 26^\circ, \quad \lambda_E = 145^\circ.75, \quad \lambda_W = 139^\circ.75$$

C-2

Table 3.7. ($1^{\circ} \times 1^{\circ}$) Mean Anomalies from GEM 9 Coefficients, Original Geoid Heights and Gridded Geoid Heights.

BLOCK	LATN	LATS	LONE	LONW	GEM	CANOM	TANOM	CANOM-TANOM
719	10.00	9.00	128.00	127.00	29.1	-19.6	-30.4	10.0
	10.00	9.00	129.00	128.00	27.2	51.7	50.4	1.3
	10.00	9.00	130.00	129.00	25.4	12.3	15.4	-3.1
	10.00	9.00	131.00	130.00	20.7	23.2	23.6	-0.4
	10.00	9.00	132.00	131.00	22.1	6.8	8.2	-1.4
	9.00	8.00	128.00	127.00	30.4	-60.9	-59.0	-1.9
	9.00	8.00	129.00	128.00	23.6	61.5	53.8	7.7
	9.00	8.00	130.00	129.00	10.6	20.3	23.2	-2.9
	9.00	8.00	131.00	130.00	23.3	23.1	19.9	3.2
	9.00	8.00	132.00	131.00	23.7	16.1	18.3	-2.4
	8.00	7.00	128.00	127.00	30.0	-57.4	-56.9	-0.5
	8.00	7.00	129.00	128.00	29.2	59.2	55.9	5.3
	8.00	7.00	130.00	129.00	27.6	33.0	35.4	-3.4
	8.00	7.00	131.00	130.00	26.2	21.3	23.5	-1.2
	8.00	7.00	132.00	131.00	24.8	29.0	29.1	-0.1
	7.00	6.00	128.00	127.00	30.8	-141.6	-162.8	21.2
	7.00	6.00	129.00	128.00	29.2	65.4	78.5	-13.1
	7.00	6.00	130.00	129.00	27.8	61.4	58.8	2.6
	7.00	6.00	131.00	130.00	26.5	27.4	28.4	-1.0
	7.00	6.00	132.00	131.00	25.3	15.3	15.9	-0.6
	6.00	5.00	128.00	127.00	30.0	-159.2	-158.2	-1.0
	6.00	5.00	129.00	128.00	23.6	40.2	42.1	-1.9
	6.00	5.00	130.00	129.00	27.4	60.7	57.4	3.3
	6.00	5.00	131.00	130.00	26.3	39.3	41.7	-2.4
	6.00	5.00	132.00	131.00	25.4	10.8	11.5	-0.7

RMS DIFFERENCE BETWEEN CANOM AND TANOM IS   5.9 MGALS

where, TANOM: the ($1^{\circ} \times 1^{\circ}$) mean anomaly computed from (10×50) gridded geoid heights.

The grid limits for prediction described in this table are:

$$\varphi_N = 11^{\circ}, \quad \varphi_S = 4^{\circ}, \quad \lambda_E = 128^{\circ}, \quad \lambda_W = 121^{\circ}$$

-91-

Table 3.8. (1°x 1°) Mean Anomalies from GEM 9 Coefficients, Original Geoid Heights and Gridded Geoid Heights.

BLOCK	LATN	LATS	LONE	LONW	GEM	CANOM	TANOM	CANOM-TANOM
711	10.00	9.00	133.00	132.00	20.6	7.4	7.9	-0.5
	10.00	9.00	134.00	133.00	19.2	15.1	17.1	-2.0
	10.00	9.00	135.00	134.00	17.8	7.6	3.9	3.7
	10.00	9.00	136.00	135.00	16.5	23.9	32.8	-8.9
	10.00	9.00	137.00	136.00	16.1	6.6	7.8	-1.2
	9.00	8.00	133.00	132.00	22.3	23.6	21.8	1.8
	9.00	8.00	134.00	133.00	20.9	13.1	13.7	-0.6
	9.00	8.00	135.00	134.00	19.5	47.3	41.5	5.8
	9.00	8.00	136.00	135.00	18.1	-15.3	-16.2	0.9
	9.00	8.00	137.00	135.00	16.6	53.9	48.7	5.2
	8.00	7.00	133.00	132.00	23.5	30.2	30.0	0.2
	8.00	7.00	134.00	133.00	26.2	27.8	24.4	2.9
	8.00	7.00	135.00	134.00	25.8	36.9	43.7	-6.8
	8.00	7.00	135.00	137.00	19.5	-31.0	-16.1	-14.9
	8.00	7.00	137.00	136.00	18.0	36.6	24.7	13.9
	7.00	6.00	133.00	132.00	24.2	30.9	29.7	1.2
	7.00	6.00	134.00	133.00	23.1	47.1	41.1	6.0
	7.00	6.00	135.00	134.00	21.9	-34.1	-23.1	-11.0
	7.00	6.00	136.00	135.00	25.6	37.9	30.5	7.4
	7.00	6.00	137.00	135.00	19.3	4.8	1.4	3.3
	6.00	5.00	133.00	132.00	24.5	40.8	38.4	7.4
	6.00	5.00	134.00	133.00	29.6	30.8	31.0	-0.2
	6.00	5.00	135.00	134.00	22.6	1.8	-3.1	4.9
	5.00	4.00	135.00	133.00	21.8	35.7	29.7	-4.0
	5.00	4.00	137.00	135.00	20.3	28.6	31.0	-2.4

RMS DIFFERENCE BETWEEN CANOM AND TANOM IS   6.2 MGALS

where, TANOM: the (1°x 1°) mean anomaly computed from (10 x 40) gridded geoid heights.

The grid limits for prediction described in this table are:

$$\varphi_N = 11°, \quad \varphi_s = 4°, \quad \lambda_\varepsilon = 138°, \quad \lambda_w = 131°$$

Table 3.9. (1°x 1°) Mean Anomalies From GEM 9 Coefficients, Original Geoid Heights and Gridded Geoid Heights.

BLOCK	LATN	LATS	LONE	LONW	GEM	CANOM	TANOM	CANOM-TANOM
343	40.00	39.00	292.00	291.00	-15.4	-32.4	-30.3	-2.1
	40.00	39.00	293.00	292.00	-15.4	-35.4	-38.4	3.0
	40.00	39.00	294.00	293.00	-15.3	-31.1	-31.4	0.3
	40.00	39.00	295.00	294.00	-14.9	-26.7	-28.2	1.5
	40.00	39.00	296.00	295.00	-14.4	-33.8	-35.6	1.2
	40.00	39.00	297.00	296.00	-13.8	-32.3	-31.1	-1.2
	39.00	38.00	292.00	291.00	-16.6	-29.1	-31.4	3.3
	39.00	38.00	293.00	292.00	-16.6	-37.7	-34.6	-3.1
	39.00	38.00	294.00	293.00	-16.4	-37.6	-31.8	-5.8
	39.00	38.00	295.00	294.00	-16.0	-23.6	-27.1	-1.5
	39.00	38.00	296.00	295.00	-15.4	-30.6	-32.5	1.9
	39.00	38.00	297.00	296.00	-14.7	-3.3	-4.1	-5.1
	38.00	37.00	292.00	291.00	-18.0	-37.6	-37.5	-0.1
	38.00	37.00	293.00	292.00	-18.0	-32.7	-36.7	3.0
	38.00	37.00	294.00	293.00	-17.7	-25.6	-25.4	0.8
	38.00	37.00	295.00	294.00	-17.3	-15.4	-17.6	1.3
	37.00	36.00	292.00	295.00	-16.5	-11.9	-9.0	-2.9
	37.00	36.00	297.00	296.00	-15.8	-15.6	-12.5	-3.1
	37.00	36.00	292.00	291.00	-19.8	-23.8	-23.4	1.5
	36.00	35.00	291.00	292.00	-19.7	-23.2	-25.4	1.3
	36.00	35.00	293.00	293.00	-19.3	-19.1	-18.3	-0.3
	36.00	35.00	294.00	293.00	-18.7	-9.9	-11.3	1.9
	36.00	35.00	295.00	294.00	-17.9	-13.8	-14.5	0.3
	35.00	34.00	297.00	295.00	-17.0	-18.5	-9.5	-5.0
	35.00	34.00	292.00	291.00	-21.7	-23.1	-23.5	3.4
	35.00	34.00	293.00	292.00	-21.6	-22.1	-19.3	-3.3
	35.00	34.00	294.00	295.00	-21.1	-12.9	-12.3	-1.1
	35.00	34.00	295.00	294.00	-20.8	-9.4	-5.3	-3.6
	35.00	34.00	296.00	295.00	-19.4	-16.3	-14.0	-0.3
	35.00	34.00	297.00	296.00	-16.4	-14.3	-15.1	0.3

RMS DIFFERENCE BETWEEN CANOM AND TANOM IS   2.6 MGALS

where,   TANOM: the (1°x 1°) mean anomaly predicted from (10 x 40) gridded geoid heights.

The grid limits for prediction described in this table are:

$$\varphi_N = 40^\circ.5, \quad \varphi_S = 34^\circ.5, \quad \lambda_E = 297^\circ.5, \quad \lambda_W = 291^\circ.5$$

4. Fourier Transforms and Frequency Domain Collocation

Satellite altimeters such as those on Geos-3 and Seasat-1 satellites introduced in the last several years have provided multitudes of mearurements of gvoid heights. In Chapter 2 we analyzed some of this Geos-3 altimeter data. In the near future other measurements such as gravity gradients by satellite borne gradiometers, range and/or range-rate measurements by satellite-to-satellite tracking (SST) techniques may be available in large quantities. The processing and interpretation of this data by conventional methods is very costly and time consuming, even sometimes being impossible. The analysis of such data usually involves convolutioi in the space domain, e.g. conventional colloc..tion, but only simple multiplications in the frequency domain (see sections 1.7.2.1 and 1.7.2.2 for convolution) as we will show later in this chapter. Thus, frequency domain methods can be considered as efficient tools in our computations and analyses. We compute frequency domain representations via Fourier transforms of space domain representations.

Fourier transforms can be used for many different purposes. Jordan (1978) shows how to use Fourier transforms for upward continuation, for computing anomalies from geoid heights and for solving Stokes' integral, etc. as an application of Wiener filtering (Moritz, 1967), which is a special case of our approach to be explained in this chapter. Fourier transforms will be applied on the expressions in the space domain. Therefore, the theory of least-squares collocation in the space domain will be briefly explained below just for the purpose of completeness. For details readers are referred to Moritz (1975).

4.1 Least-Squares Collocation

Least-squares collocation is a method utilizing minimum variance estimation for a model of the form (Moritz, 1975, p. 7)

$$x = AX + s' + n \qquad (4.1)$$

where
- x : the N-vector of observations (measurements)
- X : the u-vector of parameters or unknowns
- n : the N-vector of the measuring errors (noise)
- A : the design matrix of dimension ($N \times u$)
- s' : the N-vector of signals measured

The vectors s' and n are purely random and they have zero expectation (average or mean value), i.e.

$$E\{s'\} = E\{n\} = 0 \qquad (4.2)$$

Let "s" denote the signal vector of length P to be predicted defined as

$$s = [s_1, s_2, \ldots, s_p]^T \tag{4.3}$$

and let us assume zero expectation for signal s and no correlation between the signal measured or the signal to be predicted and the noise, i.e.

$$E\{s\} = 0, \quad E\{s' \cdot n^T\} = 0, \quad E\{s \cdot n^T\} = 0 \tag{4.4}$$

then we can proceed as follows.

Eqn. (4.1) can be written in the following way

$$x = A X + [0 \quad I] \begin{bmatrix} s \\ s' + n \end{bmatrix} \tag{4.5}$$

where 0 denotes $(N \times P)$ zero matrix and I denotes $(N \times N)$ unit matrix. It is convenient to substitute

$$v = \begin{bmatrix} s \\ s' + n \end{bmatrix} \tag{4.6}$$

playing the role of "residuals", and

$$B = [0 \quad I] \tag{4.7}$$

so that eqn. (4.5) can be written as

$$x = A X + B v \tag{4.8}$$

Eqn. (4.8) has the form of condition equations with parameters (Uotila, 1967).

Here, for least-squares adjustment, we minimize

$$v^T Q^{-1} v = \text{minimum} \tag{4.9}$$

where Q is the covariance matrix of the v-vector and may be defined as a partitioned matrix

$$Q = \begin{bmatrix} C_{ss} & C_{ss'} \\ C_{s's} & C_{s's'} + C_{nn} \end{bmatrix} \tag{4.10}$$

where $\quad C_{ss} = \text{cov}(s, s), \quad C_{ss'} = \text{cov}(s, s')$
$\qquad\quad C_{s's} = \text{cov}(s', s), \quad C_{nn} = \text{cov}(n, n)$
$\qquad\quad C_{s's'} = \text{cov}(s', s')$

The covariance between two functions, say f and g, is defined as

$$\text{cov}(f, g) = E\{[f - E(f)][g - E(g)]^T\} \tag{4.11}$$

Thus using the definition above and recalling the assumptions given by eqns. (4.2) and (4.4), we obtain

-95-

$$C_{ss} = E\{s\,s'\}, \quad C_{s's'} = E\{s'\,s'^T\}, \quad C_{nn} = E\{n\,n'\}$$

$$C_{ss'} = E\{s\,s'^T\}, \quad C_{s's} = E\{s'\,s'\} \tag{4.12}$$

The minimization of eqn (4.9) is equivalent to

$$\phi = \tfrac{1}{2} v' Q^{-1} v - K^T (A X + B v - x) \tag{4.13}$$

where K is the column vector of correlates or Lagrange multipliers. In order to minimize eqn (4.13) we proceed as follows:

$$\frac{\partial \phi}{\partial v} = v' Q^{-1} - K^T B = 0$$

$$\frac{\partial \phi}{\partial X} = \phantom{v'Q^{-1}} - K^T A = 0 \tag{4.14}$$

$$A X + n - x = 0$$

The solution of the equations above yield:

$$X = [A^T (B Q B^T)^{-1} A]^{-1} A^T (B Q B^T)^{-1} x$$

$$v = Q B^T (B Q B^T)^{-1} (x - A X)$$

Substitute eqns. (4.7) and (4.10) above for B and Q respectively to obtain

$$X = (A^T \overline{C}^{-1} A)^{-1} A^T \overline{C}^{-1} x \tag{4.15}$$

$$v = \begin{bmatrix} s \\ s' + n \end{bmatrix} \quad \begin{bmatrix} C_{ss'} \overline{C}^{-1} (x - A X) \\ x - A X \end{bmatrix} \tag{4.16}$$

where
$$\overline{C} = C_{s's'} + C_{nn} \tag{4.17}$$

By using the definition (4.11), with eqns. (4.2) and (4.4), it can be easily seen

$$C_{xx} = \overline{C} = C_{s's'} + C_{nn}$$

$$C_{sx} = C_{ss'} , \quad C_{xs} = C_{s's} \tag{4.18}$$

so we can write for s given by eqn. (4.16)

$$s = C_{sx} \overline{C}^{-1} (x - A X) \tag{4.19}$$

The errors of estimations are (Moritz, 1975, pp. 32-33)

$$E_{xx} = (A^T C_{xx} A)^{-1} \tag{4.20}$$

$$E_{ss} = C_{ss} - C_{sx} C_{xx}^{-1} C_{xs} + H A E_{xx} A^T H^T \tag{4.21}$$

where $H = C_{sx} C_{xx}^{-1}$.

If we assume there are no parameters (i.e. $X = 0$) in the adjustment, as we assumed in the second and third chapters in the determination of gravity anomalies and undulations from Geos-3 altimeter data, then we can write (c.f. eqn. (2.4)) for the signal

$$s = C_{sx} C_{xx}^{-1} x \qquad (4.22)$$

and for the errors on signal

$$E_{ss} = C_{ss} - C_{sx} C_{xx}^{-1} C_{xs} \qquad (4.23)$$

The equations given in this section solve our problem in the space domain. The inversion of the covariance matrix of observations C_{xx}, which has a dimension equal to the number of observations (N), is very costly and time consuming for large N. Therefore, we seek another solution, which is faster and cheaper than the space domain solution, i.e. the frequency-domain solution. It is this alternative which is to be discussed in the next sections.

4.2 Frequency Domain Collocation

In the previous section we derived the space domain equations of least-squares collocation. In what follows, we will derive the corresponding equations in the frequency domain. Thus the least-squares collocation is carried out in the frequency domain and solved for the desired quantities. The application of the inverse transform (Fourier) yields the corresponding desired quantities in the space domain. All the required transforms are carried out by the fast Fourier transform (FFT) to facilitate and speed up the computations.

We have defined the discrete Fourier transform (DFT) pairs by eqns. (1.65) and (1.66) in section 1.5. Here we use a slightly different definition for the DFT pairs expressed as follows:

$$X_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x(k) \exp(-i2\pi kn/N), \qquad \text{(direct Fourier transform)} \qquad (4.24)$$
$$n = 0, 1, \ldots, (N-1)$$

$$x(k) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_n \exp(i2\pi kn/N), \qquad \text{(inverse Fourier transform)} \qquad (4.25)$$
$$k = 0, 1, \ldots, (N-1)$$

The equations above are derived from eqns. (1.65) and (1.66), first by multiplying both sides of (1.65) by $\sqrt{N}$ in order to obtain

$$Y_n = \sqrt{N} X_n' = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x(k) \exp(-i2\pi kn/N) \qquad (4.26)$$

and then substituting $X_n' = Y_n/\sqrt{N}$ in eqn. (1.66) to obtain

$$x(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} Y_n \exp(i 2\pi kn/N) \qquad (4.27)$$

Eqns. (4.26) and (4.27) are equivalent to (4.24) and (4.25) respectively, and can be seen following the replacement of Y_n by X_n above.

Eqn. (4.24) can be written in a matrix form as

$$X = A_1 \cdot X \qquad (4.28)$$

where $\quad X = [X_0, X_1, \ldots, X_{N-1}]^T, \quad x = [x(0), x(1), \ldots, x(N-1)]^T$

$$A_1 = \frac{1}{\sqrt{N}} \begin{bmatrix} e^{-2\pi i 00/N} & e^{-2\pi i 10/N} & \ldots\ldots & e^{-2\pi i(N-1)0/N} \\ e^{-2\pi i 01/N} & e^{-2\pi i 11/N} & \ldots\ldots & e^{-2\pi i(N-1)1/N} \\ \vdots & \vdots & & \vdots \\ e^{-2\pi i 0(N-1)/N} & e^{-2\pi i 1(N-1)/N} & \ldots\ldots & e^{-2\pi i(N-1)(N-1)/N} \end{bmatrix} \qquad (4.29)$$

Here A_1 is the transform matrix and equal to the Fourier matrix defined by

$$F = [F_{kn}] = \frac{1}{\sqrt{N}} [\exp(-2\pi i kn/N)], \qquad 0 \le k, n \le (N-1) \qquad (4.30)$$

so we can write $\qquad A_1 = F \qquad (4.31)$

As we mentioned before, we used the space domain equations (4.22) and (4.23) for the determination of gravity anomalies and their errors from geoid heights. Now we can derive the corresponding equations in the frequency domain. The Fourier transform pairs for observations, signals and covariance matrices will be denoted as follows.

$$x \longleftrightarrow X, \qquad s \longleftrightarrow S, \qquad C_{xs} \longleftrightarrow C_{XS}$$
$$C_{sx} \longleftrightarrow C_{SX}, \qquad C_{xx} \longleftrightarrow C_{XX}, \qquad E_{ss} \longleftrightarrow E_{SS} \qquad (4.32)$$

The vector of measurements (x) and the vector if signals (s) are transformed into the frequency domain through

$$X = A_1 x, \qquad S = A_1 s \qquad (4.33)$$

where the transform matrix A_1 is defined by eqn. (4.29) being equal to the Fourier matrix defined by (4.30). In order to obtain the frequency representations of the covariance and cross-covariance matrices C_{xx}, C_{ss}, C_{sx}, we recall the definition of covariances and cross-covariances given by eqn. (4.11)

$$C_{XX} = E\{[X - E(X)][X - E(X)]^+\} \qquad (4.34)$$

where ($\dagger$) denotes the complex conjugate and transpose operation. Substitute eqn. (4.28) for X in (4.34) to obtain

$$C_{XX} = E\{[A_1 x - E(A_1 x)] [A_1 x - E(A_1 x)]^\dagger\}$$

$$= A_1 E\{[x - E(x)] [x - E(x)]\} A_1^\dagger$$

Since the expectation of the terms inside the braces is equal to C_{xx}, the covariance matrix of observations, we can write

$$C_{XX} = A_1 C_{xx} A_1^\dagger \tag{4.35}$$

Similarly, we can prove that

$$C_{SX} = A_1 C_{sx} A_1^\dagger \tag{4.36}$$

$$C_{XS} = A_1 C_{xs} A_1^\dagger \tag{4.37}$$

$$C_{SS} = A_1 C_{ss} A_1^\dagger \tag{4.38}$$

$$E_{SS} = A_1 E_{ss} A_1^\dagger \tag{4.39}$$

Having transformed the space domain quantities into the frequency domain, we can write the frequency domain collocation solutions corresponding to eqns. (4.22) and (4.23) of the space domain solutions

$$S = C_{SX} C_{XX}^{-1} X \tag{4.40}$$

$$E_{SS} = C_{SS} - C_{SX} C_{XX}^{-1} C_{XS} \tag{4.41}$$

Following the computation of the signal vector S and its error E_{SS} in the frequency domain, we can find the signal vector s and its error E_{ss} in the space domain by applying the inverse transform as explained below. From eqn. (4.33), we can write

$$s = A_1^{-1} S$$

Since $A_1 = F$ is an orthogonal matrix (recall the orthogonality relations of exponential functions as explained in the first chapter), we have for the signal

$$s = A_1^\dagger S \tag{4.42}$$

and similarly for the error matrix

$$E_{ss} = A_1^\dagger E_{SS} A_1 \tag{4.43}$$

We see from eqns. (4.40) and (4.41) that we still have to invert the matrix C_{xx}, which has dimension $(N \times N)$, N being equal to the total number of observations. We transformed everything into the frequency domain to avoid the inversion of the covariance matrix C_{xx}, which also has dimension $(N \times N)$. The computations of C_{xx}, C_{sx}, and C_{xs} from eqns. (4.35), (4.36) and (4.37) respectively, require the product of three $(N \times N)$ full matrices in each case. So if we go blindly, we will have to perform many more computations compared to the direct collocation solution in the space domain. However, we will show later that a frequency domain matrix (FDM) is diagonal if the corresponding space domain matrix (SDM) is circular, and asymptotically diagonal if the SDM is of the Toeplitz form. In addition, we will show that the diagonal elements of the FDM is the Fourier transform of the first column of the corresponding SDM. Hence, we do not have to compute the FDM by the direct transformation of the corresponding SDM, for example, as in eqn. (4.35), which requires the product of three $(N \times N)$ matrices.

4.3 Diagonalization of Toeplitz and Circulant Matrices

Suppose we have a stationary covariance function and equally spaced (or sampled) observations along a profile, then, as we explained in the third chapter, the covariance matrix of observations are of the Toeplitz form. In addition, if this profile forms a complete circle, then the resulting covariance matrix is circular. A circulant matrix, say T_c, is one having the form

$$T_c = \begin{bmatrix} t_0 & t_1 & t_2 & \cdots & t_{N-1} \\ t_{N-1} & t_0 & t_1 & \cdots & t_{N-2} \\ t_{N-2} & t_{N-1} & t_0 & \cdots & t_{N-3} \\ \vdots & \vdots & \vdots & & \vdots \\ t_1 & t_2 & t_3 & \cdots & t_0 \end{bmatrix} \qquad (4.44)$$

The circulant matrix T_c is a special type of the simple Toeplitz matrix T_{N-1} given by eqn. (3.1) such that $t_{-k} = t_{N-k}$, $k = 1, \ldots, (N-1)$. The diagonalization of T_c and T_{N-1} will be given in order to see that the eigenvalues of T_c and T_{N-1} (asymptotically) are the diagonal elements of their frequency domain representations.

4.3.1 Diagonalization of Toeplitz Matrices

As in the third chapter, we denote a simple Toeplitz matrix by T_{N-1}. The dimension of T_{N-1} is $(N \times N)$ and defined as in eqn. (3.1) such that its elements on symmetric diagonals are identical and its diagonal elements are equal. Assume the covariance function t_k of T_{N-1} is absolutely summable, i.e.,

$$\sum_{k=-\infty}^{\infty} |t_k| < \infty \qquad (4.45)$$

then, for large N, Fuller (1976, pp. 133-138) proves that

$$F^{\dagger} T_{N-1} F \cong \text{Diag} (\lambda_0, \lambda_1, \ldots, \lambda_{N-1}) \tag{4.46}$$

whore F is defined by eqn. (4.30) and λ_n are the eigenvalues of T_N. The eigervalues (λ_n) are defined as follows:

$$\lambda_n = 2\pi t(\omega_n), \quad \omega_n = 2\pi n/N, \quad n = 0, 1, \ldots, (N-1)$$

where $t(\omega_n)$ is the spectral density of the t_k expressed as

$$t(\omega_n) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} t_k \exp(-i\omega_n k), \quad n = 0, 1, \ldots, (N-1) \tag{4.47}$$

Thus, we can write for eigenvalues

$$\lambda_n = \sum_{k=-\infty}^{\infty} t_k e^{-i2\pi k n/N}, \quad n = 0, 1, \ldots, (N-1) \tag{4.48}$$

Since we cannot carry the summation in eqn. (4.48) to infinity, and becasue t_k is absolutely summable, i.e. t_k gets smaller with increasing $|k|$, we will assume:

(1) $t_k = 0$ for $k \geq N$: we can write for the eigenvalues of T_{N-1}

$$\lambda_n \cong \sum_{k=-(N-1)}^{N-1} t_k e^{-i2\pi k n/N} = t_0 + \sum_{k=1}^{N-1} (t_k + t_{N-k}) e^{-i2\pi kn/N}$$

Hence $\quad \underline{\lambda} = \sqrt{N} \cdot F^{\dagger} \underline{t}^{\blacksquare}$ \hfill (4.49)

where $\underline{\lambda} = [\lambda_0, \lambda_1, \ldots, \lambda_{N-1}]^{\mathsf{T}}$, $\underline{t}^{\blacksquare} = [t_0, (t_1 + t_{N-1}), (t_2 + t_{N-2}), \ldots, (t_{N-1} + t_1)]$, and F is the Fourier matrix defined by eqn. (4.30).

In the following sections, we will come across products such as $F T_{N-1} F^{\dagger}$ instead of $F^{\dagger} T_{N-1} F$ as given in eqn. (4.46). Therefore, we will define the vector of the eigenvalues of $F T_{N-1} F^{\dagger}$ as follows:

$$\underline{\lambda} = \sqrt{N} F^{\dagger} \underline{t}^{\blacksquare} \tag{4.50}$$

(2) $t_k = 0$ for $k > m$, where $m \leq N/2$: with this assumption, T_{N-1} becomes (to be denoted as $\overline{T}_{N-1}$)

$$\overline{T}_{N-1} \begin{bmatrix} t_0 & t & \cdots & t_{\blacksquare} & & & & \\ t_1 & t_0 & \cdots & \cdots & t_{-\blacksquare} & & \underline{0} & \\ \vdots & & \ddots & & & \ddots & & \\ t_{\blacksquare} & \cdots & t_1 & t_0 & t_{-1} & \cdots & t_{-\blacksquare} & \\ & \ddots & & & & & \ddots & t_{-\blacksquare} \\ & & \ddots & & & & & \vdots \\ \underline{0} & & & \ddots & & & & t_{-\blacksquare} \\ & & & & t_{\blacksquare} & t_{\blacksquare-1} & \cdots & t_0 \end{bmatrix} \tag{4.51}$$

If we fill the zero elements of $\bar{T}_{N-1}$ in such a way that the resulting matrix, say $\bar{T}_c$, is circulant, i.e.

$$T_c = \begin{bmatrix} t_0 & t_{-1}\ldots\ldots t_{-m} & 0 & t_m\ldots\ldots t_1 \\ t_1 & t_0 & & t_{-m}\cdot 0\ldots\ldots t_2 \\ \vdots & & & \ddots \\ t_m\ldots\ldots\ldots\ldots t_0 & t_{-1}\ldots\ldots\ldots 0 \\ 0\ldots\ldots\ldots\ldots t_1 & t_0\ldots\ldots\ldots t_{-m} \\ \vdots & \ddots \\ t_{-1}\ldots\ldots\ldots 0 & t_m\ldots\ldots\ldots t_0 \end{bmatrix} = \begin{bmatrix} c_0 & c_1\ldots\ldots c_{N-1} \\ c_{N-1} & c_0\ldots\ldots c_{N-2} \\ \vdots \\ \vdots \\ \vdots \\ c_1 & c_2\ldots\ldots c_0 \end{bmatrix} \qquad (4.52)$$

where

$$c_k = \begin{cases} t_{-k} & \text{for } k = 0,1,\ldots,m \\ t_{N-k} & \text{for } k = (N-m),\ldots,(N-1) \\ 0 & \text{otherwise} \end{cases} \qquad (4.53)$$

then $\bar{T}_{N-1}$ and $\bar{T}_c$ are asymptotically equivalent which can be proven as follows (Gray, 1977, pp. 27-29):

$$\lim_{N \to \infty} |\bar{T}_{N-1} - \bar{T}_c|^2 = \lim_{N \to \infty} \frac{1}{N} \sum_{k=0}^{m} k [|t_k|^2 + |t_{-k}|^2]$$

$$\le \lim_{N \to \infty} \frac{m}{N} \sum_{k=0}^{m} [|t_k|^2 + |t_{-k}|^2] = 0$$

from the summability of covariance functions. Thus

$$\lim_{N \to \infty} |\bar{T}_{N-1} - \bar{T}_c| = 0, \quad i.e \quad \lim_{N \to \infty} \bar{T}_N = \bar{T}_c \qquad (4.54)$$

In what follows, we will see that the diagonalization of a circular matrix is simple and its eigenvalues can be found easily. Since $\bar{T}_{N-1}$ is asymptotically equivalent to the circular matrix $\bar{T}_c$, the eigenvalues of T_{N-1} are equal to those of $\bar{T}_c$ and $\bar{T}_{N-1}$ can be diagonalized similar to T_c.

4.3.2 Diagonalization of Circular Matrices

We defined a circulant matrix T_c by eqn. (4.44). The eigenvalues, λ_n, and eigenvectors, Y_n, of T_c satisfy the following equation,

$$T_c \cdot Y_n = \lambda_n \cdot Y_n , \qquad n = 0,1,\ldots,(N-1) \qquad (4.55)$$

where $Y_n = [y_{1n}, y_{2n}, \ldots, y_{(N-1)n}]^T$. Let r_n be a root of the equation $r^N = 1$ such that

$$r_n = e^{-i2\pi n/N} , \qquad n = 0,1,\ldots,(N-1) \qquad (4.56.a)$$

and let us set

$$y_{jn} = r_n^j , \qquad j,n = 0,1,\ldots,(N-1) \qquad (4.56.b)$$

then eqn. (4.55) can be written

$$T_c \begin{bmatrix} r_n^0 \\ r_n^1 \\ \vdots \\ r_n^{N-1} \end{bmatrix} = \lambda_n \begin{bmatrix} r_n^0 \\ r_n^1 \\ \vdots \\ r_n^{N-1} \end{bmatrix} \tag{4.57}$$

Now substitute eqn. (4.44) for T_c to obtain

$$c_0 \; r_n^0 + c_1 \; r_n^1 + \ldots\ldots + c_{N-1} r_n^{N-1} = \lambda_n \, r_n^0$$
$$c_{N-1} r_n^0 + c_\lambda r_n^1 + \ldots\ldots + c_N \; r_n^{N-1} = \lambda_n \, r_n^1$$
$$\vdots \qquad \vdots \qquad\qquad\qquad \vdots \qquad\quad \vdots \tag{4.58}$$
$$c_1 \; r_n^0 + c_2 \; r_n^1 + \ldots\ldots + c_0 \; r_n^{N-1} = \lambda_n \, r_n^{N-1}$$

and multiply the first equation of (4.58) by r_n^N, the second by r_n^{N-1}, and so forth. Then use $r_n^{N+k} = r_n^k$ in order to see the equality for each equation if

$$\lambda_n = \sum_{k=0}^{N-1} c_k \cdot r_n^k \tag{4.59}$$

The substitution of eqn. (4.56.a) in (4.59) yields:

$$\lambda_n = \sum_{k=0}^{N-1} c_k \, e^{-12\pi kn/N}, \qquad\qquad n = 0, 1, \ldots, (N-1) \tag{4.60}$$

Hence, the characteristic vectors are given as

$$Y_n = [\, 1, \, 2^{-12\pi n/N}, \, e^{-12\pi(N-1)n/N} \,]^T, \qquad n = 0, 1, \ldots, (N-1) \tag{4.61}$$

and the orthonormal matrix, which also diagonalizes the original matrix T_c, of eigenvectors is given as

$$F = \frac{1}{\sqrt{N}} [\, Y_0, \, Y_1, \ldots, \, Y_{N-1} \,] \tag{4.62}$$

Eqn. (4.62) is equivalent to eqn. (4.30) defined previously, i.e. the Fourier matrix. Since F is the orthonormal matrix of eigenvectors, we can write (Byron et. al., 1969, pp. 120-124)

$$F^\dagger \, T_c \, F = \text{Diag}\,(\lambda_0, \, \lambda_1, \ldots, \, \lambda_{N-1}) \tag{4.63.a}$$

Hence,

$$\underline{\lambda} = [\, \lambda_0, \, \lambda_1, \ldots, \, \lambda_{N-1} \,]^T = \sqrt{N} \cdot F \cdot \underline{c} \tag{4.63.b}$$

Here $F^\dagger$ denotes the complex conjugate transpose of F as usual and $\underline{c}$ is the first row (or column) of T_c. Similarly, the diagonal elements of $F T_c F^\dagger$ are expressed as follows

$$\underline{\lambda} = \sqrt{N} \; F^\dagger \, \underline{c} \tag{4.63.c}$$

Thus we have proved that the Fourier matrix (F) given by eqn. (4.30) is the orthonormal diagonalizing matrix formed by the eigenvectors of either a Toeplitz matrix or a circular matrix. The eigenvalues can be computed simply by eqn. (4.50) for Toeplitz matrices and by eqn. (4.60) for circular matrices.

4.4 Frequency Domain Collocation on One-Dimensional Region

4.4.1 Large N and the Classical Wiener Filtering

In section 4.2 we derived the least-squares collocation expressions in the frequency domain, and in the previous section we showed that the transform matrix $A_1 = F$ defined by eqn. (4.29) is, in fact, the diagonalizing matrix of the covariance matrix C_{xx}, C_{xs}, C_{sx}. Therefore, we can write for the frequency domain collocation covariance matrices C_{xx}, C_{xs}, C_{sx}, and C_{ss}.

$$C_{XX} = \text{Diag}(\lambda_0^{xx}, \lambda_1^{xx}, \ldots, \lambda_{N-1}^{xx})$$

$$C_{XS} = \text{Diag}(\lambda_0^{xs}, \lambda_1^{xs}, \ldots, \lambda_{N-1}^{xs})$$

$$C_{SX} = \text{Diag}(\lambda_0^{sx}, \lambda_1^{sx}, \ldots, \lambda_{N-1}^{sx})$$

$$C_{SS} = \text{Diag}(\lambda_0^{ss}, \lambda_1^{ss}, \ldots, \lambda_{N-1}^{ss})$$

(4.64)

where λ_n^{xx}, λ_n^{xs}, λ_n^{sx}, λ_n^{ss}, $n = 0, 1, \ldots, (N-1)$ are the eigenvalues of C_{xx}, C_{xs}, C_{sx}, and C_{ss} respectively. Thus, we can write for the frequency domain collocation solutions (expressed by eqns. (4.40) and (4.41))

$$S_n = \lambda_n^{sx} \cdot X_n / \lambda_n^{xx} \qquad\qquad (4.65)$$

$$n = 0, 1, \ldots, (N-1)$$

$$(E_{ss})_n = \lambda_n^{ss} - \lambda_n^{sx} \lambda_n^{xs} / \lambda_n^{xx} \qquad\qquad (4.66)$$

where S_n is the n-th signal of S vector and $(E_{ss})_n$ is the corresponding error, and X_n is the n-th value of X vector, i.e. $S = [S_0, S_1, \ldots, S_{N-1}]^T$, $X = [X_0, X_1, \ldots, X_{N-1}]^T$.

The eigenvalues in eqn. (4.64) are computed from eqn. (4.50) when the corresponding space domain matrix is of the Toeplitz form and from eqn. (4.63.c) when it is a circular matrix. The equations given by (4.65) and (4.66) lead us to the very well known "the classical Wiener Kolmogorov filter".

Having found S in the frequency domain we can easily obtain the corresponding s-signal vector in the space domain applying the inverse transform as expressed by eqn. (4.42). The computation of the error matrix E_{ss} from E_{ss} through eqn. (4.43) is also very simple: Consider the diagonalization of the circular matrix T_c by eqn. (4.63.a). Now suppose the diagonal elements of $F^\dagger T_c F$, i.e. the eigenvalues λ_n, are given and the circular matrix T_c is asked. Then the first row $\underline{c}$ of T_c is found from (4.63.b) by multiplying both sides of the equation from left side by $F^\dagger$ to obtain

-104-

$$\underline{c} = \frac{1}{\sqrt{N}} \; F^{\dagger} \; \underline{\lambda}$$

Since $E_{ss} = F E_{ss} F^{\dagger}$ has a similar form with $(F T_c F^{\dagger})$ in addition to its diagonality, the error covariance matrix of signals (E_{ss}) is circulant and its first row $\underline{S}_{\bullet}$ is computed as follows

$$\underline{S}_{\bullet} = \frac{1}{\sqrt{N}} \; F \; d^{ss} \tag{4.67}$$

where $d^{ss} = [d_0 , d_1 , \ldots, d_{N-1}]$ are the diagonal elements of E_{ss}. The computation of the remaining elements of E_{ss} is straightforward by recalling the definition of circular matrices given by eqn. (4.44).

Thomas et. al. (1976, pp. 3-4) relates the overall estimation error to a single quantity, namely to the rms estimation error, σ_s defined by

$$\sigma_s^2 = \frac{1}{N} \; E\{s \, s^T\} = \frac{1}{N}\sum_{k=1}^{N} \; E\{(s_T - s)^2\} \tag{4.68}$$

where N is the length of the s-signal vector to be predicted

s_T is the vector of true signals.

Eqn. (4.68) is equivalent to

$$\sigma_s^2 = \frac{1}{N} \; \text{Trace} \, (E_{ss}) \tag{4.69.a}$$

br by the Parsenval's theorem mentioned in section 1.3.4, it is equivalent to

$$\sigma_s^2 = \frac{1}{N} \; \text{Trace} \, (E_{ss}) \quad = \quad \sum_{k=0}^{N-1} \lambda_k^{ss} \tag{4.69.b}$$

Thus we have completed the solution of the s-signal vector and its error. The total number of computations for the frequency domain least-squares collocation explained above is proportional to $N \log_2 N$ multiplications and additions compared to N^3 in the case of the space domain least-squares collocation.

Moritz (1967) introduced the frequency domain method explained above as "the least-squares filtering", and Sjoberg et al. (1977) applied it for the prediction of mean free-air gravity anomalies from altimeter data along one arc at a time. In his computations, Sjoberg assumed the covariance or cross-covariance t_k to be equal to zero for $k \geq M$ (M = 70) compared to $t_k = 0$ for $k > N$ in our solutions.

Following the derivation of the frequency-domain algorithm, it was necessary to test it. In order to do so, we selected two arcs, which are the 11-th and 12-th arcs of Figure 2.2. Our aim was to determine free-air point gravity anomalies from altimeter data at data points along one arc at a time. We followed the computational steps as outlined below:

(1) The data vector of geoid heights, say $x(k)$, $k = 0, 1, \ldots, N-1$, was transformed into the frequency domain by eqn. (4.33).

(2) The covariance matrix C_{xx} of geoid heights and the cross-covariance matrix C_{sx} between observations (geoid heights) which were computed by using the empirical covariance functions given by the subroutine COVA of Tscherning and Rapp (1974) and signals were transformed into the frequency domain by eqn. (4.50).

(3) The frequency domain representation of the signal vector was computed by eqn. (4.65).

(4) Finally, the desired s-signal ⋯⋯ in the space-domain was computed by eqn. (4.42).

For our examinations, we first considered a 682 km long segment of the 11-th arc and denoted it ARC 11-A, and a 1380 km long segment of the 12-th arc and denoted it ARC 12-A. Secondly, we considered Arcs 11 and 12 in their total lengths. These arcs and the statistics between the space domain collocation solution (SDCS) and the frequency domain collocation solution (FDCS) are described in Table 4.1. The free-air point anomalies determined by the SDCS and FDCS are shown in Figures 4.1, 4.2, 4.3, and 4.4 for the arcs 11-A, 12-A, 11, and 12 respectively. Table 4.1 and Figures 4.1 to 4.4 do not include about 1.5% of predictions made at the beginning and at the end of each arc. We deleted them due to their large differences from those of SDCS. This big difference is caused by the negligence of off-diagonal terms in FDCS known as "edge effects" (Thomas et. al., 1976).

Table 4.1. The Statistics Between SDCS and FDCS Along
The Arcs 11-A, 12-A, 11, and 12.

ARC No.	Latitude φ_1, φ_2 (Deg.)	Longitude λ_1, λ_2 (Deg.)	Length (km)	No. of Obsns.	RMS Diff. (mgals)	Mean Diff. (FDCS-SDCS)	Max. Diff. (FDCS-SDCS)
11-A	12.848, 18.197	309.957, 306.844	681.9	50	1.8	0.1	- 4.3
12-A	-0.024, 10.881	146.530, 140.565	1380.2	100	11.7	-0.3	-42.3
11	12.848, 39.751	309.957, 291.194	3511.2	254	2.6	-0.2	8.4
12	-0.024, 32.466	149.530, 126.823	4162.3	300	5.3	-0.6	-33.6
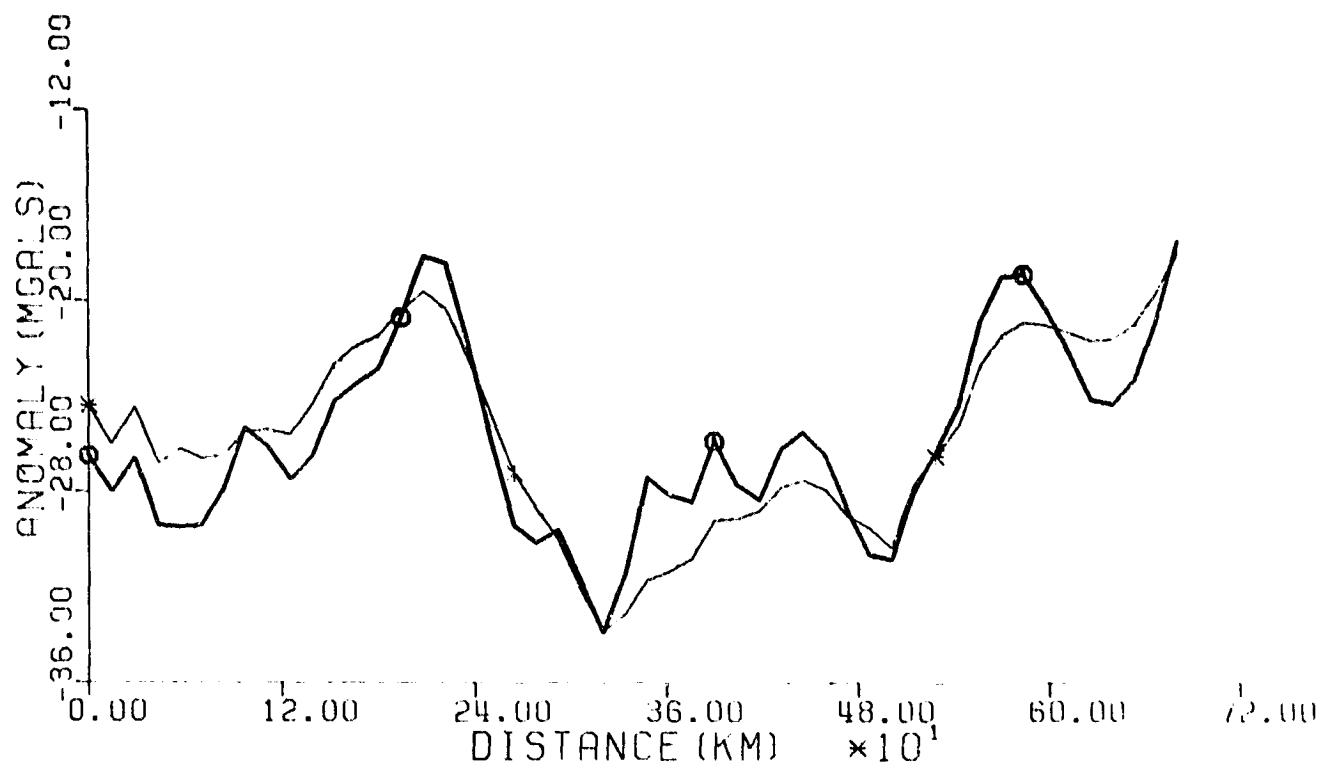
Figure 4.1. Free-air point anomalies from SDCS and FDCS along the Arc 11-A.
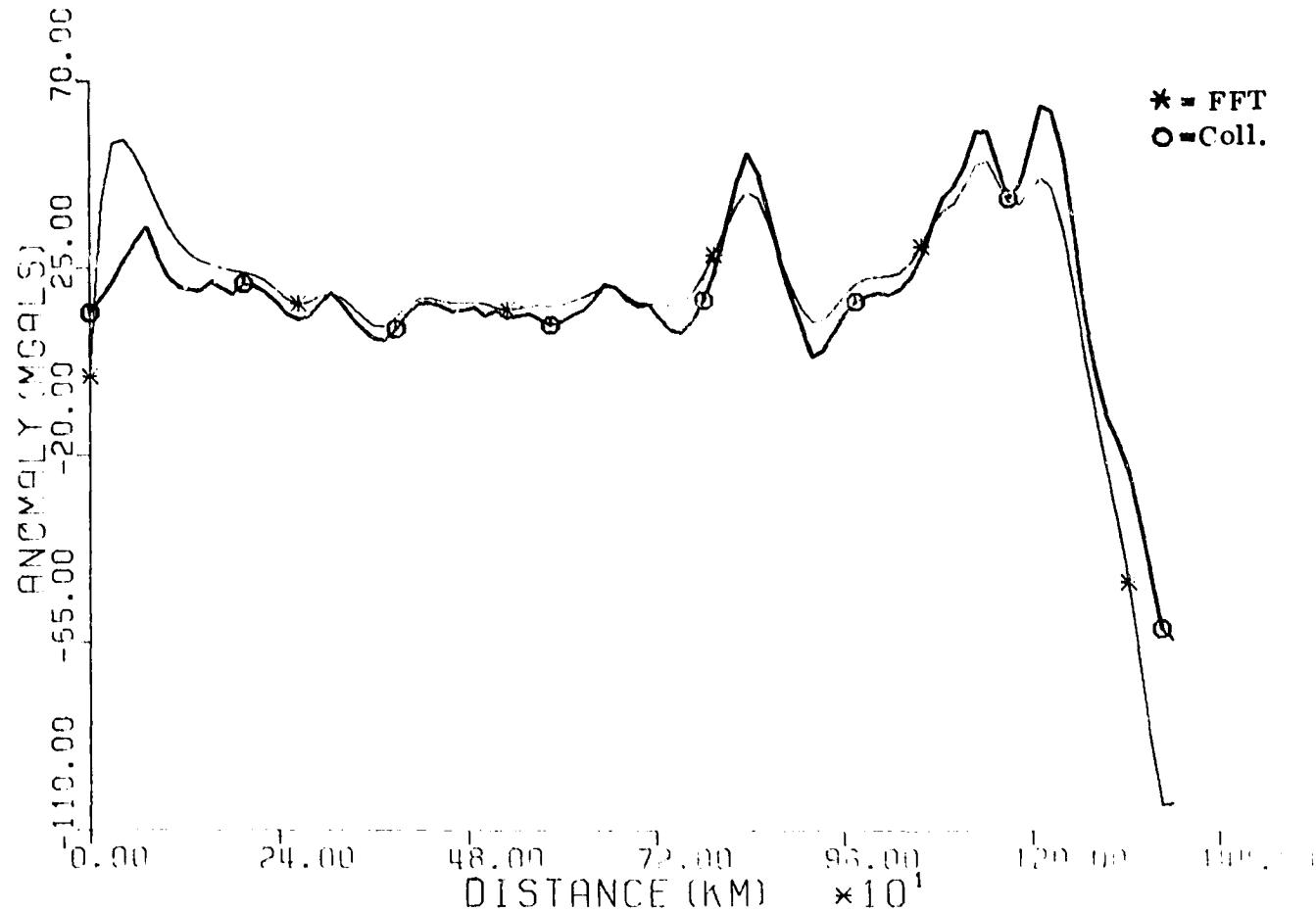
Figure 4.2. Free-air point anomalies from SDCS and FDCS along the Arc 12-A.
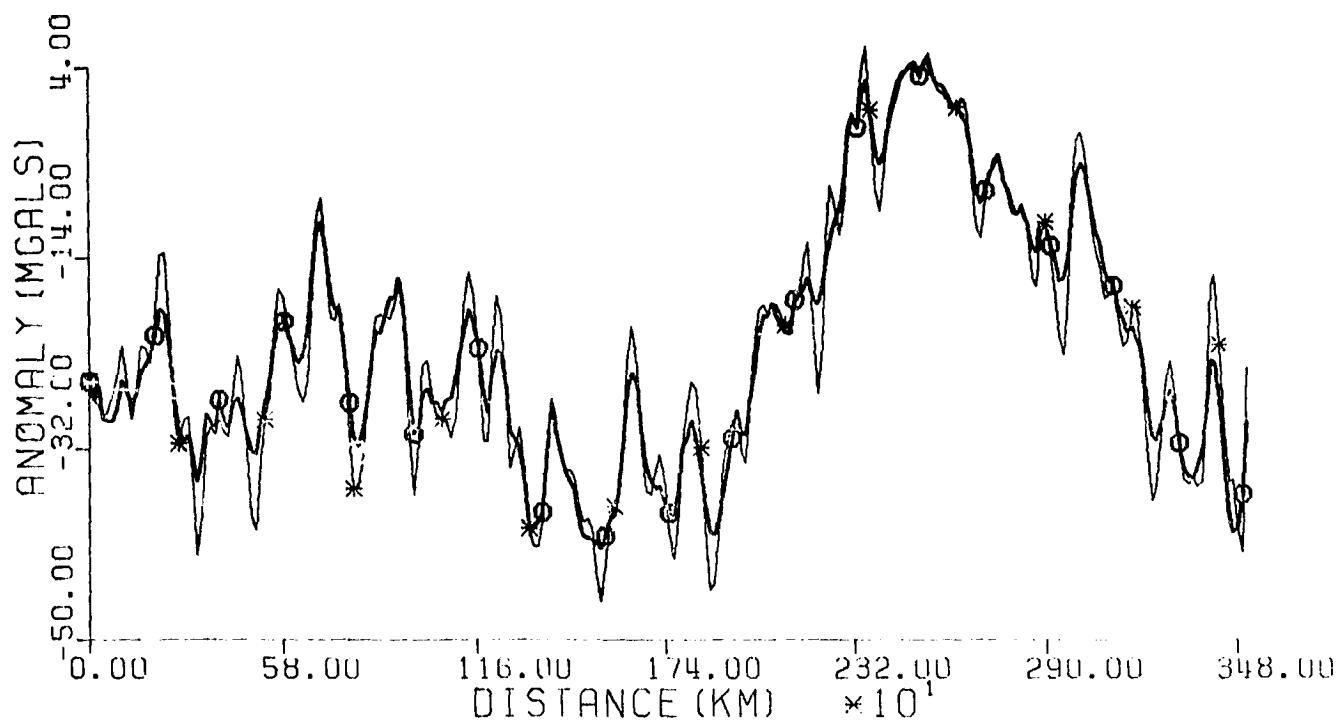
Figure 4.3. Free-air point anomalies from SDCS and FDCS along the Arc 11.
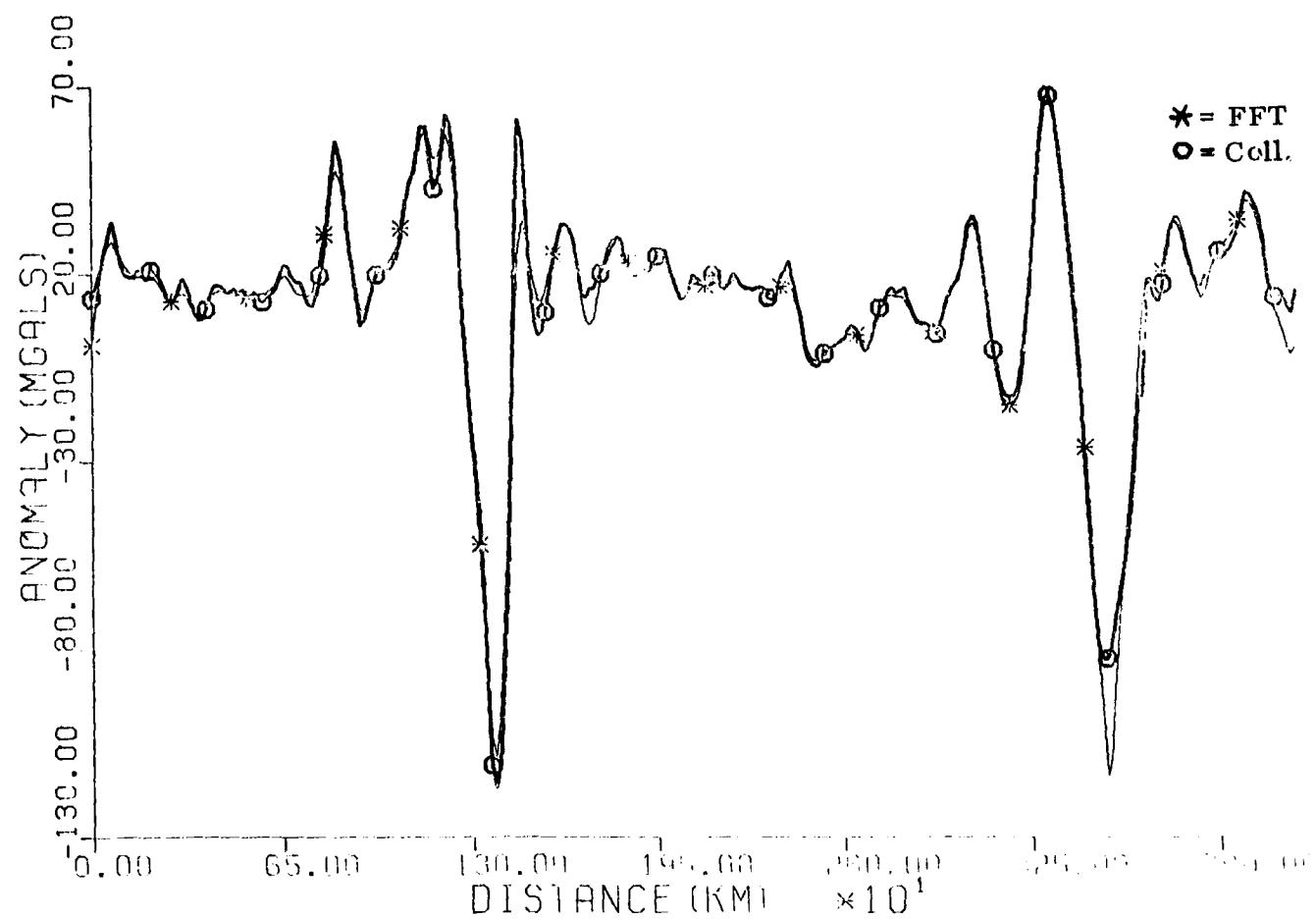
Figure 4.4. Free-air point anomalies from SDCS and FDCS along the Arc 12.

4.4.2 Small N and the Windowed Frequency Domain Solution

In the preceeding section, we as. .ned a long data length (large N) over an interval longer than correlation distances and used asymptotically equivalent expressions for the eigenvalues of the space domain covariance and cross-co-variance matrices in order to compute the corresponding asymptotically diagonal frequency domain covariance and cross-covariance matrices. In the case of a short data length and small N the asymptotic equalities mentioned above are no longer valid. Therefore, in this section, we will introduce the windowed frequency domain collocation algorithm. This algorithm retains the necessary computational speed to solve large problems but uses data given over an interval shorter than correlation distances. The frequency domain representations of covariance and cross-covariance matrices become band-diagonal here and diagonal in the limit as N gets larger and larger. By "windows" we try to minimize the number of super-diagonal bands, which is proportional to the side lobe energy (for the frequency domain representations of windows see section 1.7.3.1) of the data spectrum (Heller et al., 1977, p. 13). In what follows, we transform the "windowed data" into the frequency domain and perform our solutions there in order to compute the s-signal vector and the E_{ss}-error covariance matrix. The solution of the s-signal vector given by eqn. (4.22) is achieved in two steps:

- the solution of $\quad y = C_{xx}^{-1} x$
- the solution of $\quad s = C_{sx} C_{xx}^{-1} x = C_{sx} y$

4.4.2.1. The Computation of $y = C_{xx}^{-1} x$

We want to solve a system of equations

$$C_{xx} \cdot y = x \tag{4.70}$$

for the y-vector. Since C_{xx} is usually a full matrix, this solution would require approximately N^3 computer operations. In order to reduce the number of computer operations, we first multiply the data vector, x, by an appropriate window matrix to control the sidelobe energy of the spectrum of the data vector, then transform it into the frequency domain, i.e.

$$X = A_2 \cdot x \tag{4.71.a}$$

and for the frequency domain representation, Y, of the solution vector, y

$$Y = (A_2^\dagger)^{-1} \cdot y \tag{4.71.b}$$

where A_2 is the transform matrix defined as

$$A_2 = F w \tag{4.72}$$

-109-

Here F is defined in eqn. (4.30) and w is a window matrix (usually diagonal). For the covariance matrix of X (analogous to the derivation of eqn. (4.35)) we can write

$$C_{XX} = A_2 \ C_{xx} \ A_2^\dagger$$

$$= F \ w \ C_{xx} \ w \ F^\dagger \tag{4.73}$$

Hence eqn. (4.70) takes the following form in the frequency domain

$$C_{XX} \ Y = X \tag{4.74}$$

We solve the system of linear equation above for Y, and we compute the y-vector in the space domain from eqn. (4.71.b),

$$y = A_2^\dagger \ Y \tag{4.75}$$
$$= w \ F^\dagger \ Y$$

Why solve the system of linear equations in the frequency domain such as defined by eqn. (4.74) instead of solving the corresponding equations in the space domain such as defined by eqn. (4.70)? The reason is that C_{XX} is nearly band-diagonal and the solution of eqn. (4.74) can be carried out inexpensively by an efficient algorithm such as the banded-Cholesky decomposition (Forsythe et al., 1967).

In order to show the almost band-diagonal form of C_{XX} we write eqn. (4.73) in the following form (Heller et al., 1977, pp. 55-59)

$$C_{XX} = 2 \ K \ F_{2N} \ \hat{w} \ \hat{C}_{xx} \ \hat{w} \ F_{2N}^\dagger \ K^\intercal \tag{4.76}$$

where: K is the sampling matrix defined as

$$K = \begin{bmatrix} 1 & 0 & 0\ldots\ldots0 & 0 & 0 \\ 0 & 0 & 1\ldots\ldots0 & 0 & 0 \\ \vdots & & & & \\ 0 & 0 & 0\ldots\ldots0 & 1 & 0 \end{bmatrix}_{(N \times 2N)} \tag{4.77.a}$$

F_{2N} is $(2N \times 2N)$ Fourier matrix defined as

$$F_{2N} = \{(F_{2N})_{kn}\} = \frac{1}{\sqrt{2N}} \ [e^{-i2\pi kn/2N}], \quad 0 \le k, n \le (2N-1) \tag{4.77.b}$$

$\hat{w}$ is the $(2N \times 2N)$ extended version of the w window matrix and defined as

$$\hat{w} = \begin{bmatrix} w & 0 \\ 0 & 0 \end{bmatrix}_{(2N \times 2N)} \tag{4.77.c}$$

Here, w is a diagonal matrix with diagonal elements $(w_0, w_1, \ldots, w_{N-1})$ defined by the window function, and $\underline{0}$ is $(N \times N)$ zero matrix.

$\hat{C}_{xx}$ is a $(2N \times 2N)$ circular matrix defined as

$$\hat{C}_{xx} = \begin{bmatrix} C_{xx} & C'_{xx} \\ C'_{xx} & C_{xx} \end{bmatrix} \tag{4.77.d}$$

Here C'_{xx} is a Toeplitz matrix with elements

$$C'_{xx}(\ell \cdot \Delta t) = \begin{cases} C_{xx}[(\ell - N) \cdot \Delta t] & \text{for } \ell > 0 \\ 0 & \text{for } \ell = 0 \\ C_{xx}[(\ell + N) \cdot \Delta t] & \text{for } \ell < 0 \end{cases}$$

By substituting the definitions of K, F_{2N}, $\hat{w}$, and $\hat{C}_{xx}$ above in eqn. (4.76), we can see that eqn. (4.73) and (4.76) are equivalent. Eqn. (4.76) can also be expressed as

$$C_{xx} = 2 K F_{2N} \hat{w} F_{2N}^{\dagger} F_{2N} \hat{C}_{xx} F_{2N}^{\dagger} F_{2N} \hat{w} F_{2N}^{\dagger} K^{\mathsf{T}}$$

$$= 2 K \hat{W} \hat{C}_{xx} \hat{W} K^{\mathsf{T}} \tag{4.78}$$

where

$$\hat{W} = F_{2N} \hat{w} F_{2N}^{\dagger}, \qquad \hat{C}_{xx} = F_{2N} \hat{C}_{xx} F_{2N}^{\dagger} \tag{4.79}$$

We have shown in section 4.3.2 that the frequency domain representation of a circular matrix is diagonal as given, for example, by eqns. (4.63.a) and (4.63.b) From eqn. (4.63.a) we can write for the circular matrix T_C

$$T_C = F \cdot \text{Diag}(\lambda_0, \lambda_1, \ldots, \lambda_{N-1}) F^{\dagger} \tag{4.80}$$

If we let N go to $2N$, then T_C of eqn. (4.80) and $\hat{W}$ of eqn. (4.79) have similar forms. Therefore, $\hat{W}$ is also circulant with the elements

$$[\hat{W}]_{kn} = \Omega_{k-n}, \qquad\qquad 0 \le k, n \le (2N-1) \tag{4.81.a}$$

we can write for the first row, say $\underline{\Omega}$, of $\hat{W}$

$$\underline{\Omega} = (\Omega_0, \Omega_1, \ldots, \Omega_{2N-1})^{\mathsf{T}} = \frac{1}{\sqrt{2N}} F_{2N}^{\dagger} \begin{pmatrix} w \\ \underline{0} \end{pmatrix} \tag{4.81.b}$$

as can be derived from eqn. (4.63.b), where $\begin{pmatrix} w \\ \underline{0} \end{pmatrix}$ is the vector of the diagonal elements of $\hat{w}$ given by eqn. (4.77.c). It also follows directly from eqns. (4.63.a-c) that $\hat{C}_{xx}$ given by eqn. (4.79) (recall that $\hat{C}_{xx}$ is circulant) is diagonal as expressed below

$$\hat{C}_{xx} = \text{Diag}(\omega_0, \omega_1, \ldots, \omega_{2N-1}) \tag{4.82.a}$$

Actually, the ω_n's are the eigenvalues of $\hat{C}_{xx}$ given in eqn. (4.77.d) and they can be computed using eqn. (4.63.c)

$$\underline{\omega} = \sqrt{2N} F_{2N}^{\dagger} \underline{c} \tag{4.82.b}$$

-111-

where $\underline{\omega} = (\omega_0, \omega_1, \ldots, \omega_{2N-1})^T$ is the power spectrum of $\hat{C}_{xx}$, and $\underline{c}$ is the first column of the circular matrix $\hat{C}_{xx}$.

Now substitute eqns. (4.77.a), (4.81.a), and (4.82.a) for K, $\hat{W}$, and $\hat{C}_{xx}$ respectively, in eqn. (4.78) and use some simplifications to obtain

$$(\hat{C}_{xx})_{jk} = 2 \sum_{\ell=0}^{2N-1} \Omega_{\ell-2j} \, \Omega_{\ell-2k} \, \omega_\ell \qquad \begin{array}{l} 0 \leq j,k \leq (N-1) \\ \Omega_{-\ell} = \Omega_{2N-\ell} \end{array} \qquad (4.83)$$

where Ω_ℓ and ω_ℓ are defined by eqns. (4.81.b) and (4.82.b), respectively, and $\overline{\Omega}_\ell$ represents the complex conjugate of Ω_ℓ. The equation (4.78) is now a very efficient formula for the calculation of C_{xx}. This comes about because when we select a window which has a mainlobe and a series of very small sidelobes, then C_{xx} has an almost band-diagonal structure with some elements in the upper right and lower left corners. Hence, the solution of eqn. (4.74) can be carried out by the band-diagonal and corner implementation of the Cholesky decomposition, which requires numerical operations proportional to $(N_B^2 \cdot N)$, where N_B is the width of the band, compared to N^3 for conventional solutions.

The out-of-band elements of C_{xx} can be made very small by selecting an appropriate window and matrix bandwidth (usually less than 10) so that we can neglect the out-of-band elements in our solutions. If C_{xx} with zero out-of-band elements is denoted by C_δ, then we can state that C_δ is an approximation to the exactly transformed matrix C_{xx}. This approximation affects the solution vector y of eqn. (4.70). The magnitude of the error on y can be controlled by the choice of window and matrix bandwidth as mentioned above. The approximation above is the only error introduced in the calculation of the y-vector.

There are many windows one can use for the purpose of minimizing the approximation affects. Here we will only examine the Kaiser window which has proved to be very useful in signal processing applications. The Kaiser window is expressed in the space domain as follows:

$$w_k(n) = I_0(\beta\sqrt{1-(n/N_H)^2})/I_0(\beta), \qquad -N_H \leq n \leq N_H \qquad (4.84)$$

where $\quad N_H = \begin{cases} (N-1)/2 & \text{if } N \text{ odd} \\ N/2 & \text{if } N \text{ even} \end{cases}, \qquad I_0(x) = \sum_{k=0}^{\infty} \frac{1}{(k!)^2}\left(\frac{x}{2}\right)^{2k}$

and $\quad \beta$ is a constant that specifies a frequency response trade-off between the peak height of the sidelobe ripples and the width or the energy of the mainlobe.

The space domain and frequency domain representations of the Kaiser window for $\beta = 2, 4, 6$, and 8 are shown in Figure 4.5.

Having examined the Kaiser window, we return to the elements of C_{xx} defined by eqn. (4.83). The right-hand side of this equation is the convolution of ω_ℓ with the weighting function
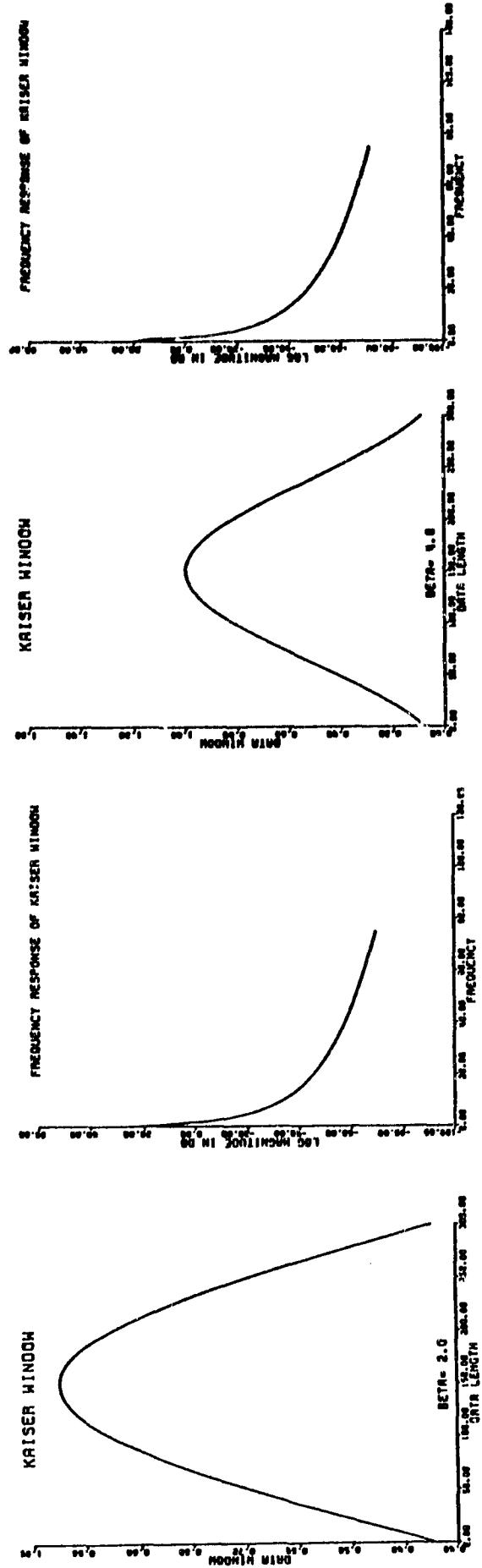
Figure 4.5. The space domain and frequency domain representations of the Kaiser window for $\beta = 2, 4, 6,$ and 8.
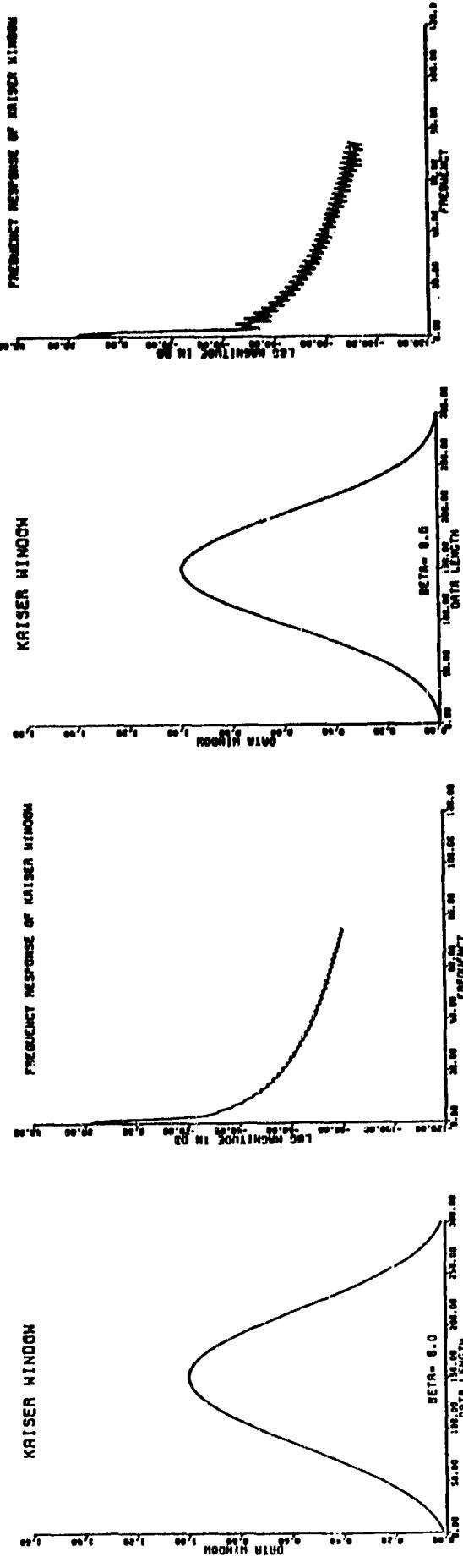
$$P_\ell(j,k) = \Omega_{\ell-2j} \, \overline{\Omega}_{\ell-2k} \tag{4.85}$$

This weighting function is very small and consequently negligible when $|k-j| > N_8$ and $|k-j| < (N-N_8)$ which correspond to the product of a mainlobe and sidelobe or two sidelobes. It is for this reason that we have tried to find a window function with very small sidelobes. As a result, we only compute the band-diagonal and upper right and lower left corner elements of C_{xx} shown in Figure 4.6. Hence, we have for the super-diagonal elements

$$[C_{xx}]_{k,k+m} = 2 \sum_{\ell=0}^{2N-1} \Omega_{\ell-2k} \, \overline{\Omega}_{\ell-2(k+m)} \; \omega_\ell \,, \qquad \left\{ \begin{array}{l} 0 \le k \le (N-m-1) \\ 0 \le m \le N_8 \end{array} \right. \tag{4.86.a}$$

and for the upper right corner elements

$$[C_{xx}]_{k,j} = 2 \sum_{\ell=0}^{2N-1} \Omega_{\ell-2k} \, \overline{\Omega}_{\ell-2j} \; \omega_\ell \,, \qquad \left\{ \begin{array}{l} 0 \le k \le N_8 \\ (N-N_8+k) \le j \le N \end{array} \right. \tag{4.86.b}$$

The lower diagonal elements of C_{xx} are just the complex conjugates of the corresponding symmetrical elements of the upper diagonal elements.

If we let w (window matrix) go to I (identity matrix) and N_8 go to zero, i.e. only the elements of the main diagonal are retained, then the algorithm presented above reduces to "the Generalized Wiener Filtering" described in section 4.4.1.
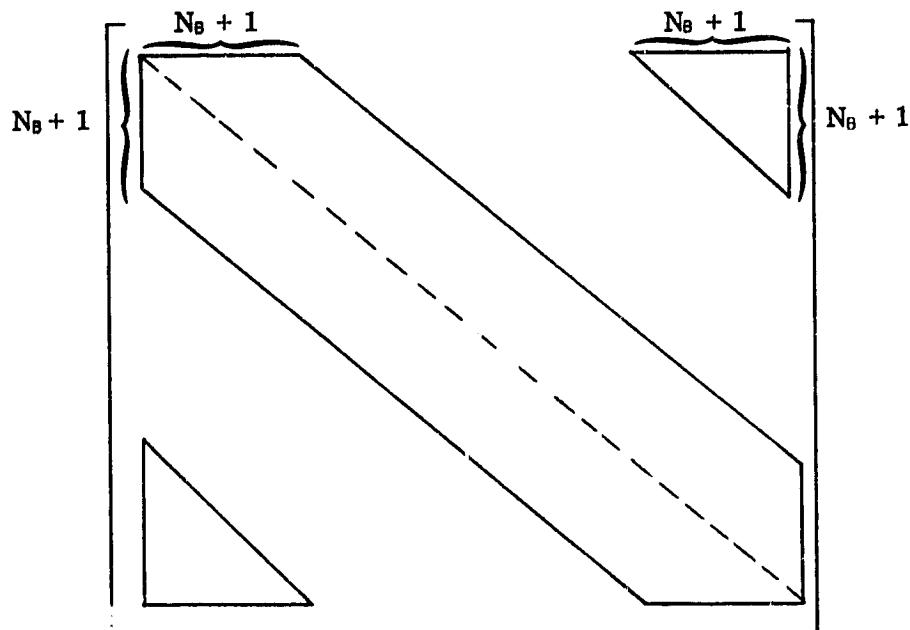
Figure 4.6. The structure of C_{xx} under Fourier transformation.

4.4.2.2 The Computation of $s = C_{sx} \cdot y$

For the computation of $(C_{sx} \cdot y)$ we will utilize the fast Toeplitz matrix multiplication. Thus, the y-vector will be multiplied by the Toeplitz matrix C_{sx} such that this algorithm will take many less operations than the standard way of multiplying the y-vector by C_{sx}. Suppose the cross-covariance matrix C_{sx} has dimension $(M \times N)$, then the matrix-vector multiplication requires roughly $(M \cdot N)$ multiplications and additions. If M is very small compared to N then any classical matrix-vector multiplication can be used. However, if M is large then the fast algorithm presented below would be more efficient.

In the preceeding sections we assumed $M = N$ so that C_{sx} is a Toeplitz matrix. If $M < N$, then C_{sx} is a partition of a Toeplitz matrix. In the case of $M < N$ we extend C_{sx} in such a way that the resulting matrix, to be denoted as $\overline{C}_{sx}$, is a square Toeplitz matrix. The first M elements of the product $(\overline{C}_{sx} \cdot y)$ contains the desired product $(C_{sx} \cdot y)$.

In order to achieve a fast multiplication of $(\overline{C}_{sx} \cdot y)$, we imbed $\overline{C}_{sx}$ in a $(2N \times 2N)$ circulant matrix, say T_c, as explained below. The cross-covariance function $\gamma_{sx}(k \cdot \Delta\psi)$ between signals and measurements is extended to the range $-(2N-1) \leq k \leq (2N-1)$ by the definition

$$
E_{sx}(k \cdot \Delta\psi) = \begin{cases} \gamma_{sx}(k \cdot \Delta\psi) & \text{for } 0 \leq k \leq (N-1) \\ 0 & \text{for } k = N \\ \gamma_{sx}[(k-2N) \cdot \Delta\psi] & \text{for } (N+1) \leq k \leq (2N-1) \end{cases} \tag{4.87}
$$

and $\quad E_{sx}(-k \cdot \Delta\psi) = E_{sx}[(2N-k) \cdot \Delta\psi] \quad$ for $1 \leq k \leq (2N-1)$

where $\Delta\psi$ is the spacing between s and x. The matrix T_c with the elements

$$
(T_c)_{jk} = E_{sx}[(k-j) \cdot \Delta\psi] \tag{4.88}
$$

is circulant with the property.

$$
T_c \begin{pmatrix} y \\ 0 \end{pmatrix} = \begin{pmatrix} \overline{c}_{sx} \cdot y \\ d \end{pmatrix} \tag{4.89}
$$

For the fast matrix-vector multiplication, we transform the vector $\begin{pmatrix} y \\ 0 \end{pmatrix}$ and T_c into the frequency domain as explained in section 4.2 (analogous to eqn. (4.33) and eqn. (4.63) for $\begin{pmatrix} y \\ 0 \end{pmatrix}$ and T_c respectively) to obtain

$$
Y = [Y_0, Y_1, \ldots, Y_{2N-1}]^T = F_{2N} \begin{pmatrix} y \\ 0 \end{pmatrix} \tag{4.90}
$$

and

$$
T_c^f = F_{2N} \, T_c \, F_{2N}^{\dagger} = \text{Diag}(\lambda_0, \lambda_1, \ldots, \lambda_{2N-1})
$$

$$
\underline{\lambda} = [\lambda_0, \lambda_1, \ldots, \lambda_{2N-1}]^T = \sqrt{2N} \, F_{2N}^{\dagger} \, \underline{c} \tag{4.91}
$$

where F_{2N} is defined in eqn. (4.77.b) and $\underline{C}$ is the first column of T_C. Hence, we can write

$$T_C' \cdot Y = F_{2N}\begin{pmatrix} \overline{C}_{sx} \, y \\ d \end{pmatrix} = [Y_0\lambda_0, Y_1\lambda_1, \ldots, Y_{2N-1}\lambda_{2N-1}] \tag{4.92}$$

Finally, we apply the inverse Fourier transform to obtain

$$\begin{pmatrix} \overline{C}_{sx} \, y \\ d \end{pmatrix} = F_{2N}^{+} \begin{bmatrix} \lambda_0 \, Y_0 \\ \lambda_1 \, Y_1 \\ \vdots \\ \lambda_{2N-1} Y_{2N-1} \end{bmatrix} \tag{4.93}$$

The first M elements of the product $(\overline{C}_{sx} \cdot y)$ is the desired product $(C_{sx} \cdot y)$, so multiplication is completed. The number of operations of the entire procedure amounts ot three FFT applications, therefore, it is proportional to $(N \log_2 N)$.

4.4.2.3 An Application of the Windowed Frequency Domain Collocation

In order to demonstrate the efficiency of the windowed frequency domain collocation algorithm described above, we selected an **arc** of altimetry (12 th arc of Figure 2.2), which is 4162 km long and has 300 observations ($N = 300$), and computed the ($5° \times 5°$) free-air mean gravity anomaly of block 711 ($\varphi_N = 10°$, $\varphi_S = 5°$, $\lambda_E = 137°$, $\lambda_W = 132°$) shown in Figure 4.5. We also computed the ($1° \times 1°$) anomalies inside 711 from the geoid heights along Arc 12 by the following techniques.
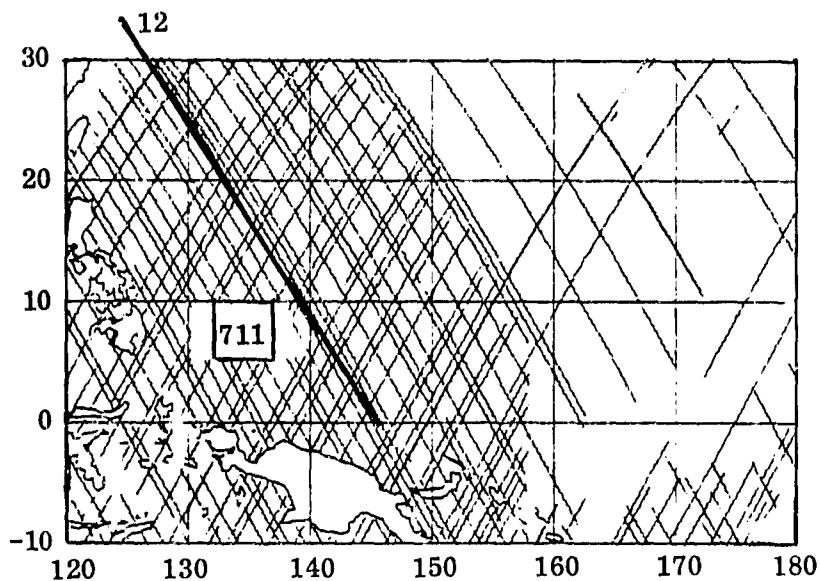
Figure 4.5. The locations of Block 711 and Arc 12.

(1) C̲o̲n̲v̲e̲n̲t̲i̲o̲n̲a̲l̲ L̲e̲a̲s̲t̲-̲S̲q̲u̲a̲r̲e̲s̲ C̲o̲l̲l̲o̲c̲a̲t̲i̲o̲n̲

Here mean anomalies were computed from eqn. (4.22) as explained in section 4.1.

(2) C̲l̲a̲s̲s̲i̲c̲a̲l̲ W̲i̲e̲n̲e̲r̲ F̲i̲l̲t̲e̲r̲i̲n̲g̲ (i.e. Frequency Domain Collocation with Large N)

The y-vector and C_{xx} matrix of eqn. (4.22) were transformed into the frequency domain by eqns. (4.33) and (4.50) respectively, to obtain

$$Y = C_{xx}^{-1} X = [\text{Diag}(\lambda_0, \lambda_1, \ldots, \lambda_{N-1})]^{-1} X$$

$$= \begin{bmatrix} 1/\lambda_0 & 0 & \ldots\ldots\ldots & 0 \\ 0 & 1/\lambda_1 & \ldots\ldots\ldots & 0 \\ 0 & 0 & \ldots\ldots\ldots & 1/\lambda_{N-1} \end{bmatrix} X \qquad (4.94)$$

then an inverse fast Fourier transform was applied to Y to yield

$$y = C_{xx}^{-1} x = F^{\dagger} Y \qquad (4.95)$$

Finally, we computed the s-signal vector by

$$s = C_{sx} y$$

where s represents the signal vector of mean anomalies and C_{sx} represents the cross-covariance between mean anomalies and observations (geoid heights).

(3) W̲i̲n̲d̲o̲w̲e̲d̲ F̲r̲e̲q̲u̲e̲n̲c̲y̲ D̲o̲m̲a̲i̲n̲ C̲o̲l̲l̲o̲c̲a̲t̲i̲o̲n̲

The x-vector and C_{xx} matrix of eqn. (4.22) was transformed into the frequency domain by eqns. (4.71.a) and (4.73) respectively, to form the following system of linear equations

$$C_{xx} Y = X \qquad (4.96)$$

We solved the linear equations above for Y using the Cholesky's decomposition as explained in section 4.4.2, then we computed $y = C_{xx}^{-1} x$ vector by an inverse transform defined by eqn. (4.75), i.e.

$$y = w F^{\dagger} Y$$

Finally, we computed the s-signal vector of mean anomalies by

$$s = C_{sx} y$$

The results of the three solutions explained above are described in Table 4.2.

Table 4.2. Mean Anomalies ($1° \times 1°$) from the GEM-9 Coefficients, from Least Squares Collocation, Wiener Filtering, and Windowed Frequency Domain Collocation by Using Undulations Along Arc 12.

BLOCK No. = 711						
Size (Deg.)	LAT (NW) (Deg.)	LON (NW) (Deg.)	Δg (mgals) GEM-9	Δg (mgals) Collocation	Δg (mgals) Wiener Filtering	Δg (mgals) Windowed Freq. Dom. Collocation
5 x 5	10	132	20.5	17.3	16.9	17.2
1 x 1	10	132	20.6	17.7	17.7	17.5
		133	19.2	15.4	15.1	15.5
		134	17.8	13.2	12.1	13.5
		135	16.5	11.1	10.5	11.4
		136	15.1	8.9	9.3	8.8
	9	132	22.3	20.0	19.1	19.7
		133	20.9	17.6	15.9	17.5
		134	19.5	15.2	13.9	15.2
		135	18.1	13.0	12.3	13.0
		136	16.6	11.0	10.8	10.8
	8	132	23.5	22.0	21.8	21.7
		133	22.2	19.6	19.4	19.4
		134	20.8	17.3	17.2	17.2
		135	19.5	15.3	15.1	15.2
		136	18.0	13.7	13.5	13.6
	7	132	24.2	23.5	21.0	23.3
		133	23.1	21.3	20.7	21.1
		134	21.9	19.2	18.9	19.1
		135	20.6	17.3	17.7	17.3
		136	19.3	16.0	17.3	16.1
	6	132	24.5	24.4	23.8	24.3
		133	23.6	22.5	22.3	22.5
		134	22.6	20.7	21.0	20.6
		135	21.5	19.0	20.0	19.0
		136	20.3	17.8	19.7	17.8

The RMS difference of 25 ($1° \times 1°$) mean anomalies between least-squares collocation and Wiener filtering is 0.87 mgals compared to 0.16 mgals between least-squares collocation and windowed frequency domain collocation. However, both RMS differences can be considered small compared to the 15 mgals standard deviation of the ($1° \times 1°$) anomaly predictions. Here we have recovered about 3.5 mgal anomaly information with respect to the GEM 9 surface. The small magnitude of the recovered anomaly masks the efficiency of the algorithm. To achieve a considerably higher magnitude we considered block 614 ($\varphi_N = 15°$, $\varphi_s = 10°$, $\lambda_E = 139°$, $\lambda_W = 134°$), through which the 12th arc passes. Unfortunately we recovered only 5.2 mgals with respect to the GEM 9 surface, so it did not help us much to check the efficiency of the algorithm.

4.5 Windowed Frequency Domain Collocation with Sine-cosine Coefficients

In section 4.4.2 we have seen that the frequency domain covariance matrix C_{xx} of windowed observations has a bnad–diagonal shape with some elements in the upper–right and lower-left corners. In addition, the elements of C_{xx} are complex valued. In order to use the standard banded Cholesky algorithm for the solution of the system of linear equations given by eqn. (4.76), we have to eliminate the upper–right and lower-left corner elements of C_{xx} in addition to real-valued band-diagonal elements. If we use a sine-cosine transform (Fuller, 1976, pp. 135-137) instead of a Fourier transform, then the resulting new covariance matrix C_{xx} is real-valued and band-diagonal (Heller et. al., 1977, pp. 38-39).

The sine-cosine transform enables us to delete redundant components from the real-valued data at negative frequencies.

Consider a circular symmetric matrix, T_s, defined as follows:

$$T_s = \begin{bmatrix} c(0) & c(1)\ldots\ldots\ldots c(2) & c(1) \\ c(1) & c(0)\ldots\ldots\ldots c(3) & c(2) \\ \vdots & \\ \vdots & \\ c(1) & c(2)\ldots\ldots\ldots c(1) & c(0) \end{bmatrix} \tag{4.97}$$

The circulant matrix T_s given above can be derived from T_c defined by eqn. (4.44) by substituting $C(N-1) = C(1)$, $C(N-2) = C(2),\ldots$. The characteristic roots of T_s are (Fuller, 1976, pp. 135-137)

$$\lambda_j = \begin{cases} \displaystyle\sum_{n=-(N-1)/2}^{(N-1)/2} c(n)\cos 2\pi nj/N & \text{for } N \text{ odd} \\ & \quad\quad j = 0, 1,\ldots,(N-1) \\ \displaystyle\sum_{n=-\frac{N}{2}+1}^{N/2} c(n)\cos 2\pi nj/N & \text{for } N \text{ even} \end{cases} \tag{4.98}$$

The equation above has a root for $j = 0$ and $(N-1)/2$ repeated roots for $j = 1, 2,\ldots,(N-1)/2$, and finally another root for $j = N/2$ when N is even. We can find two orthogonal vectors for each of these repeated roots. These vectors are

$$a_0 = \frac{1}{\sqrt{N}} [1, 1, \ldots, 1] \qquad \text{for } j = 0$$

$$a_j = \frac{\sqrt{2}}{\sqrt{N}} \left[1, \cos 2\pi \frac{j}{N}, \cos 2\pi \frac{2j}{N}, \ldots, \cos 2\pi \frac{(N-1)j}{N} \right], \ j = 1, 2, \ldots, N_M \ (4.99.\text{a})$$

$$a_j = \frac{1}{\sqrt{N}} [1, -1, 1, -1, \ldots, -1] \qquad j = N/2 \text{ and } N \text{ even}$$

$$b_j = \frac{\sqrt{2}}{\sqrt{N}} \left[0, \sin 2\pi \frac{j}{N}, \sin 2\pi \frac{2j}{N}, \ldots, \sin 2\pi \frac{(N-1)j}{N} \right], \ j = 1, 2, \ldots, N_M \ (4.99.\text{b})$$

where $\qquad N_M = \begin{cases} (N-1)/2 & \text{for } N \text{ odd} \\ N/2 - 1 & \text{for } N \text{ even} \end{cases}$

Hence we can form an orthogonal matrix Q as given below

$$Q = \frac{\sqrt{2}}{\sqrt{N}} \begin{bmatrix} A \\ B \end{bmatrix} \tag{4.100}$$

where

$$A = \begin{cases} \begin{bmatrix} a_0/\sqrt{2} \\ a_1 \\ \vdots \\ a_{N_M} \end{bmatrix} & \text{if } N \text{ odd} \\[2em] \begin{bmatrix} a_0/\sqrt{2} \\ a_1 \\ \vdots \\ a_{N/2} \end{bmatrix} & \text{if } N \text{ even} \end{cases}$$

and

$$B = \begin{bmatrix} b_1 \\ b \\ \vdots \\ b_{N_M} \end{bmatrix}$$

Since Q^T is composed of the N characteristic vectors given by eqns. (4.99.a) and (4.99.b) we can write

$$Q \ T_S \ Q^T = \text{Diag} (\lambda_0, \lambda_1, \ldots, \lambda_{N-1}) \tag{4.101}$$

where λ_j is defined by eqn. (4.98).

There exists a linear relation between the characteristic vectors of Q and the characteristic vectors of the F matrix defined previously. The F-matrix given by eqn. (4.30) can be written as follows

$$F = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{bmatrix}$$

where $\quad f_j = \frac{1}{\sqrt{N}} [1, e^{-i2\pi j/N}, e^{-i2\pi 2j/N}, \ldots, e^{-i2\pi(N-1)j/N}]$

-120-

From eqns. (4.99.a-b) and (4.101) it is possible to write

$$a_j = \frac{1}{\sqrt{2}} \, [f_j + f_{N-j}]$$

$$j = 1, 2, \ldots, N_H \qquad (4.102)$$

$$b_j = \frac{1}{\sqrt{2}} \, [f_j - f_{N-j}]$$

Using the relation given by eqn. (4.102), the Q matrix is written

$$Q = H \, F \qquad (4.103)$$

where

$$N = \text{odd} \qquad (4.104.a)$$

$$N = \text{even} \qquad (4.104.b)$$

Now we can write down the frequency domain equations (windowed) when the sine-cosine transform (Q) instead of the Fourier transform (F) is used for the computation of $y = C_{xx}^{-1} x$. The x-data vector is transformed into the frequency domain by

$$X = A_3 \, x \qquad (4.105)$$

and the y-solution vector (unknown) by

$$Y = (A_3^{\dagger})^{-1} \, y \qquad (4.106)$$

where A_3 is the transform matrix defined as

$$A_3 = H \, F \, w \qquad (4.107)$$

For the covariance matrix of X given by eqn. (4.105) we have (analogous to the derivation of eqn. (4.35))

$$C_{XX} = A_3 \, C_{xx} \, A_3^{\dagger}$$

$$= H \, F \, w \, C_{xx} \, w \, F^{\dagger} H^{\dagger} \qquad (4.108)$$

or equivalently

$$C_{XX} = 2 \, K \, H_{2N} \, F_{2N} \, \hat{w} \, \hat{C}_{XX} \, \hat{w} \, F_{2N}^\dagger \, H_{2N}^\dagger \, K^T \tag{4.109.a}$$

where K, F_{2N}, $\hat{w}$, and $\hat{C}_{XX}$ are defined by eqns. (4.77.a-d) respectively, and H_{2N} is defined by eqn. (4.104.b) after letting $N \to 2N$. Eqn. (4.109.a) can also be written as

$$\begin{aligned} C_{XX} &= 2 \, K \, H_{2N} \, F_{2N} \, \hat{w} \, F_{2N}^\dagger \, F_{2N} \, \hat{C}_{XX} \, F_{2N}^\dagger \, F_{2N} \, \hat{w} \, F_{2N}^\dagger \, F_{2N}^\dagger \, K^T \\ &= 2 \, K \, H_{2N} \, \hat{W} \, \hat{C}_{XX} \, \hat{W} \, H_{2N}^\dagger \, K^T \end{aligned} \tag{4.109.b}$$

where $\hat{W} = F_{2N} \, \hat{w} \, F_{2N}^\dagger, \quad \hat{C}_{XX} = F_{2N} \, \hat{C}_{XX} \, F_{2N}^\dagger$ (4.109.c)

If we denote

$$W_H = H_{2N} \, \hat{W} \tag{4.110}$$

then we can write

$$C_{XX} = 2 \, K \, W_H \, \hat{C}_{XX} \, W_H^\dagger \, K^T \tag{4.111.a}$$

Eqn. (4.111.a) is similar to eqn. (4.78), the only difference being the replacement of $\hat{W}$ of eqn. (4.78) by $\hat{W}_H$ in eqn. (4.111.a). Although $\hat{W}$ of eqn. (4.78) is a circular matrix, $\hat{W}_H$ of eqn. (4.111.a) is not. Therefore, we will use eqn. (4.109.b) to determine the elements of C_{XX}. Through the substitution of the definitions for K, H_N, and H_{2N} it is easy to see that

$$K \, H_{2N} = H_N \, K$$

Thus we can write eqn. (4.109.b) as follows

$$C_{XX} = 2 \, H_N \, K \, \hat{W} \, \hat{C}_{XX} \, \hat{W} \, K^T \, H_N^\dagger \tag{4.112}$$

Now substitute eqns. (4.104.b) and (4.77.a) for H_N and K respectively and eqn. (4.109.c) for $\hat{W}$ and $\hat{C}_{XX}$ in eqn. (4.112). After some simplifications, we obtain

$$[C_{XX}]_{j,k} = \begin{cases} \sum\limits_{\ell=0}^{2N-1} (\Omega_{\ell-2j} + \Omega_{\ell-2N+2j})(\overline{\Omega}_{\ell-2k} + \overline{\Omega}_{\ell-2N+2k}) \, \omega_\ell & \text{for } 0 \leq j,k \leq N/2 \\ \sum\limits_{\ell=0}^{2N-1} i(\Omega_{\ell-2j} + \Omega_{\ell-2N+2j})(\overline{\Omega}_{\ell+2k-2N} - \overline{\Omega}_{\ell+N-2k})\omega_\ell & \text{for } \begin{cases} 0 \leq j \leq N/2 \\ N/2 < k \leq (N-1) \end{cases} \\ \sum\limits_{\ell=0}^{2N-1} i(\Omega_{\ell+N-2j} - \Omega_{\ell+2j-2N})(\overline{\Omega}_{\ell-2k} + \overline{\Omega}_{\ell-2N+2k})\omega_\ell & \text{for } \begin{cases} 0 \leq k \leq N/2 \\ N/2 < j \leq (N-1) \end{cases} \\ \sum\limits_{\ell=0}^{2N-1} (\Omega_{\ell+N-2j} - \Omega_{\ell+2j-2N})(\overline{\Omega}_{\ell+N-2k} - \overline{\Omega}_{\ell+2k-2N})\omega_\ell & \text{for } N/2 < j,k \leq (N-1) \end{cases} \tag{4.113}$$

where Ω_n and ω_ℓ are given by eqns. (4.81.b) and (4.82.b) respectively and $\Omega_{-n} = \Omega_{2kN-n}$, $k = 0,1,2,\ldots$. Since C_{XX} given by eqn. (4.113) is approximately band-diagonal, as shown in Figure 4.7 (Heller et. al., 1977, p. 41), we only compute the elements of the band-diagonal, so we can write

-122-

$$[C_{xx}]_{j,j+m} = \begin{cases} \sum_{\ell=0}^{2N-1}(\Omega_{\ell-2j} + \Omega_{\ell-2N+2j})(\overline{\Omega}_{\ell-2(j+m)} + \overline{\Omega}_{\ell-2N+2(j+m)})\,\omega_\ell \\ \qquad \text{for } 0 \le j \le N/2,\ 0 \le m \le N_B \text{ such that } m+j \le N/2 \\[6pt] \sum_{\ell=0}^{2N-1}(\Omega_{\ell+N-2j} - \Omega_{\ell+2j-2N})(\overline{\Omega}_{\ell+N-2(j+m)} - \overline{\Omega}_{\ell+2(j+m)-2N})\,\omega_\ell \\ \qquad \text{for } N/2 < j \le (N-1),\ 0 \le m \le N_B \text{ such that } m+j \le (N-1) \end{cases} \tag{4.114}$$

where N_B is the width of the band.
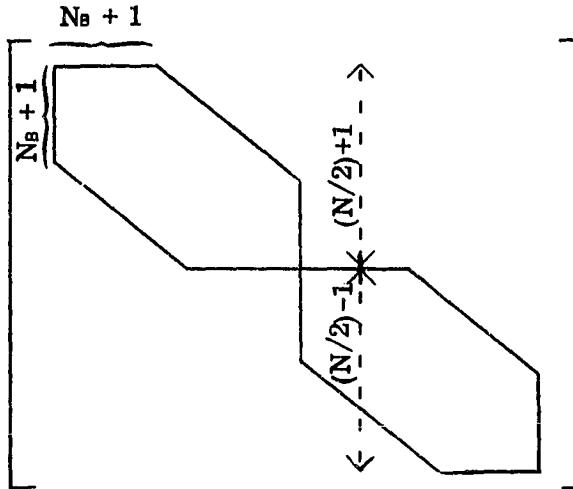
Figure 4.7. Structure of C_{xx} under sine–cosine transformation (N = even).

Having computed C_{xx} we can go back to the solution of the system of linear equations

$$C_{xx}\ Y = X \tag{4.115}$$

where X, Y, and C_{xx} are given in eqns. (4.105), (4.106), and (4.114), respectively. The equation above can be solved for Y by using the standard band-diagonal Cholesky decomposition which numerical computations are proportional to $(N_B^2 N)$. Finally, we can compute the space domain solution vector (y) from eqn. (4.106) to obtain

$$y = C_{xx}^{-1}\ x = w\ F^\dagger\ H^\dagger\ Y \tag{4.116}$$

Thus, the windowed frequency domain solution using sine–cosine coefficients, in order to compute the s-signal vector, is completed.

Since we use a highly tapered window, such as the Kaiser window, in order to minimize the width of the band, C_{xx} will have very small eigenvalues. In other words, C_{xx} is almost ill-conditioned (Heller et. al., 1977, p. 42). Consequently, C_{xx} has to be modified before solving eqn. (4.115). This modification

can be done by adding a small amount, say δ, to the diagonal elements of C_{xx} to obtain

$$\overline{C}_{xx} = C_{xx} + \delta I \tag{4.117}$$

Thus, C_{xx} cannot have any eigenvalues smaller than δ, therefore, C_{xx} is well-conditioned.

The addition of (δI) to C_{xx} actually corresponds to the addition of

$$D = \delta \cdot \text{Diag}\,(w_0^{-2},\ w_1^{-2}, \ldots,\ w_{N-1}^{-2}) \tag{4.118}$$

where w_j are the window coefficients, to the space domain covariance matrix, C_{xx}, (Ibid., p. 43). A detailed study on the introduction of δ and its consequences can be found in Heller et. al. (1977, pp. 42-47).

4.6 Frequency Domain Collocation on a Two-Dimensional Region

4.6.1 Large N and Two-Dimensional Wiener Filtering

In previous sections, we have considered equally spaced observations along a <u>single</u> profile exclusively for the determination of the s-signal vector. In this section, we will consider observations given along more than one profile leading us to the two-dimensional frequency domain collocation. In order to obtain covariance matrices (for observations) of block-Toeplitz form, we will select the profiles in such a way that observations form a grid as well as they are equally spaced along every individual profile.

Let us denote that:

M   is the number of parallel profiles with N observations along each
X_D is the data vector of (M N) observations
$C_{x_D x_D}$ is the covariance matrix of observations (it has block-Toeplitz form)
S_D is the signal vector desired
$C_{s_D x_D}$ is the cross-covariance matrix between signals and observations.

We can transform x_D into the frequency domain as follows

$$X_D = F_D\ x_D \tag{4.119}$$

where

$$F_D = \begin{bmatrix} F & 0 \ldots \ldots\ 0 \\ 0 & F \ldots \ldots\ 0 \\ \vdots & \\ 0 & 0 \ldots \ldots\ F \end{bmatrix}_{[(MN)\ \times\ (MN)]} \tag{4.120}$$

here F is $(N \times N)$ Fourier matrix given by eqn. (4.30).

Hence the covariance matrix of X_D can be expressed as (analogous to the derivation of eqn. (4.35))

$$C_{x_D x_D} = F_D \ C_{x_D x_D} \ F_D^\dagger \tag{4.121}$$

Since $C_{x_D x_D}$ has a block-Toeplitz form, we can write

$$C_{x_D x_D} = \begin{bmatrix} T_{11} & T_{12} & T_{13} \dots \dots T_{1M} \\ T_{21} & T_{22} & T_{23} \dots \dots T_{2M} \\ \vdots \\ T_{M1} & T_{M2} & T_{M3} \dots \dots T_{MM} \end{bmatrix} \tag{4.122}$$

where T_{ij} are $(N \times N)$ square blocks of Toeplitz form so that eqn. (4.121) becomes

$$C_{x_D x_D} = \begin{bmatrix} F T_{11} F^\dagger & F T_{12} F^\dagger \dots & F T_{1M} F^\dagger \\ F T_{21} F^\dagger & F T_{22} F^\dagger \dots & F T_{2M} F^\dagger \\ \vdots \\ F T_{M1} F^\dagger & F T_{M2} F^\dagger \dots & F T_{MM} F^\dagger \end{bmatrix} \tag{4.123}$$

We demonstrated before that the elements of $(F \ T_{11} \ F^\dagger)$ converge to the elements of the diagonal matrix

$$D_{11} = 2\pi \ \text{Diag} \ [\ t_{11}(\omega_0), t_{11}(\omega_1), \dots, t_{11}(\omega_{N-1})] \tag{4.124}$$

where $t_{11}(\omega_n)$ is the spectral density of T_{11}, and it is given by eqn. (4.47) evaluated at $n = 0, 1, \dots, N-1$.

Fuller (1974, pp. 308-310) shows that the elements of $(F \ T_{jk} \ F^\dagger)$, $j \neq k$, also converge to the elements of the diagonal matrix

$$D_{jk} = 2\pi \ \text{Diag} \ [\ t_{jk}(\omega_0), t_{jk}(\omega_1), \dots, t_{jk}(\omega_{N-1})], \quad j \neq k \tag{4.125}$$

where $t_{jk}(\omega_n)$ is the spectral density of T_{jk}, and is given by eqn. (4.47) evaluated at $n = 0, 1, \dots, N-1$. Thus, we can write eqn. (4.120) approximately as

$$C_{x_D x_D} = \begin{bmatrix} D_{11} & D_{12} \dots \dots D_{1M} \\ D_{21} & D_{22} \dots \dots D_{2M} \\ \vdots \\ D_{M1} & D_{M2} \dots \dots D_{MM} \end{bmatrix} \tag{4.126}$$

where D_{jk}, $j, k = 1, 2, \dots, M$, which are square sub-matrices of $C_{x_D x_D}$, are diagonal.

The solution of the s_D-signal vector from two dimensional data can be written (in the space domain) from eqn. (4.22)

$$s_D = C_{s_D x_D} \ C_{x_D x_D}^{-1} \ x_D \tag{4.127}$$

-125-

In the determination of S_D above, the costly and time consuming part is the inversion of $C_{x_D x_D}^{-1}$ or equivalently the computation of $(C_{x_D x_D}^{-1} x_D)$. Let us denote

$$y_D = C_{x_D x_D}^{-1} x_D \tag{4.128}$$

and define the frequency domain representation of y_D (c.f. to eqn. (4.119)) as

$$Y_D = F_D \, y_D \tag{4.129}$$

then we can write

$$Y_D = C_{x_D x_D}^{-1} X_D \tag{4.130}$$

Since $C_{x_D x_D}$ defined by eqn. (4.126) is formed by $(N \times N)$ diagonal sub-matrices, we can easily and cheaply invert and compute the product $(C_{x_D x_D}^{-1} X_D)$. Then, by an inverse Fourier transform, we obtain

$$y_D = F_D^{\dagger} \, Y_D \tag{4.131}$$

Finally, we compute the S_D-signal vector

$$S_D = C_{s_D x_D} \, y_D \tag{4.132}$$

The algorithm described above is equivalent to the "two-dimensional Wiener filtering".

4.6.2 Small N and Two-Dimensional Windowed Frequency Domain Collocation with Sine-Cosine Coefficients

If N, the number of observations along a single profile, is small then the $(F \, T_{jk} \, F^{\dagger})$ submatrices of eqn. (4.123) do not converge to a diagonal matrix. Therefore, in the case of small N, we will use the windowed frequency domain collocation in a two dimensional region as explained below. In order to obtain real-valued and band-diagonal submatrices for $C_{x_D x_D}$ the orthogonal matrix Q of sine-cosine coefficients given by eqn. (4.103) will be used instead of the imaginary matrix F given by eqn. (4.30) in the transformation from the space domain into the frequency domain.

Consider the definitions M, X_D, $C_{x_D x_D}$, S_D, $C_{s_D x_D}$, as given in the preceeding section. Here the transformations are applied as follows. The x_D-data vector is transformed into the frequency domain by

$$X_D = Q_D \, W_D \, x_D \tag{4.133}$$

where

$$Q_D \;=\; H_D \; F_D \;=\; \begin{bmatrix} H\,F & 0 & \ldots\ldots & 0 \\ 0 & H\,F & \ldots\ldots & 0 \\ \vdots & & & \\ 0 & 0 & \ldots\ldots & H\,F \end{bmatrix}_{[(MN)\,x\,(MN)]} \tag{4.134.a}$$

$$w_D \;=\; \begin{bmatrix} w & 0 & \ldots\ldots & 0 \\ 0 & w & \ldots\ldots & 0 \\ \vdots & & & \\ 0 & 0 & \ldots\ldots & w \end{bmatrix}_{[(MN)\,x\,(MN)]} \tag{4.134.b}$$

here w is the window matrix (diagonal with diagonal elements $(w_0, w_1, \ldots, w_{N-1})$ defined by the window function).

and y_D-solution vector is transformed by

$$Y_D \;=\; [\,(Q_D \; w_D)^{\dagger}\,]^{-1} \; y_D \tag{4.135}$$

and finally, the covariance matrix $C_{x_D x_D}$ by

$$C_{x_D x_D} \;=\; Q_D \; w_D \; C_{x_D x_D} \; w_D^{\intercal} \; Q_D^{\dagger} \tag{4.136}$$

Substitute eqn. (4.134.a) for Q_D, eqn. (4.134.b) for w_D, and eqn. (4.122) for $C_{x_D x_D}$ to obtain

$$C_{x_D x_D} \;=\; \begin{bmatrix} HFwT_{11}wF^{\dagger}H^{\dagger} & HFwT_{12}wF^{\dagger}H^{\dagger} & \ldots\ldots\ldots & HFwT_{1M}wF^{\dagger}H^{\dagger} \\ HFwT_{21}wF^{\dagger}H^{\dagger} & HFwT_{22}wF^{\dagger}H^{\dagger} & \ldots\ldots\ldots & HFwT_{2M}wF^{\dagger}H^{\dagger} \\ \vdots & & & \\ HFwT_{M1}wF^{\dagger}H^{\dagger} & HFwT_{M2}wF^{\dagger}H^{\dagger} & \ldots\ldots\ldots & HFwT_{MM}wF^{\dagger}H^{\dagger} \end{bmatrix} \tag{4.137}$$

We can write for each square $(N \times N)$ sub-matrix of $C_{x_D x_D}$

$$B_{jk} \;=\; H\,F\,w\,T_{jk}\,w\,F^{\dagger}H^{\dagger} \;=\; 2K\,H_{2N}\,F_{2N}\,\hat{w}\,\hat{T}_{jk}\,\hat{w}\,F_{2N}^{\dagger}H_{2N}^{\dagger}K^{\intercal}, \quad 1 \le j,k \le M \tag{4.138}$$

where K, H_{2N}, F_{2N}, $\hat{w}$ are defined as in section 4.4, and
$\hat{T}_{jk}$ is a circular matrix $(2N \times 2N)$ extended from T_{jk} as explained in section 4.4.2.

Eqn. (4.138) is band-diagonal as shown in section 4.5. The elements of the $(N \times N)$ band-diagonal matrix B_{jk} can be computed as in eqn. (4.116).

As before, we are here interested in solving the system of linear equations

$$C_{x_D x_D} \; Y_D \;=\; X_D \tag{4.139}$$

Since $C_{x_D x_D}$ consists of sub-matrices which are band-diagonal, eqn. (4.139) can be solved by a modified Cholesky's decomposition which requires $(M^3 N_B^2 11)$

operations compared to $(M^3 N^3)$ in the space domain solution for y_0. Following the computation of Y_0 from eqn. (4.139) we can find the y_0-solution vector from eqn. (4.135)

$$y_0 = w_0 \; Q_0^{\dagger} \; Y_0 \tag{4.140}$$

Finally, we can wrtie for the S_0-signal vector

$$S_0 = C_{s_0 \, x_0} \; y_0 \tag{4.141}$$

Thus, the windowed frequency domain collocation in two-dimensional region is completed. Due to extensive programming needed for this technique we did not apply this algorithm.

4.7. Summary

In this chapter we have covered one- and two-dimensional frequency domain least-squares collocation. For a fast solution we have introduced an approximation and neglected off-band diagonal (off-diagonal in the case of Wiener filtering) terms of the frequency domain representation C_{xx} of the covariance matrix, c_{xx}, of observations. These neglected terms cause an error in the prediction of signals. In order to minimize this error, and yet preserve the speed of the algorithm, effective wi   are used so that the off-band diagonal terms of C_{xx}, the frequency domain cc ι   natrix of the windowed data, are as small as possible. Designing an optimum   or a particular purpose and the error analysis due to the approximation ne,   oove are open for future studies.

5. Conclusions

As we stated before, the modern instrumentation enabled us to collect a large amount of geodetic data. In order to process and interpret this data efficiently we introduced the frequency methods and the fast inversion of Teoplitz matrices.

In Chapter 2 we reduced the complicated function of geoid heights along a single profile to a series of simple trigonometric functions in order to investigate the resolution (or cut-off frequency) of the GEOS-3 altimeter data as a function of wavelengths or frequencies. The minimum full wavelength recoverable from GEOS-3 altimeter data has been found to be about 100 km with the assumption of 27 mgals standard deviation on predicted point gravity anomalies. We also computed the total power in the sea surface topography at and above 20 cycles/ revolution, which is equivalent to the total power with respect to the GEM-9 surface. This power is about 2.1 meters from some 9 profiles described in Table 2.3.

In Chapter 3 we introduced a rigorous fast inversion algorithm for matrices of simple or block Toeplitz forms. In least-squares collocation solutions we have to invert the covariance matrix of observations (denoted by $\bar{C} = C_{xx}$ throughout the paper). This covariance matrix has a dimension, say N, equal to the number of observations. In the case of equally weighted and spaced observations along a single profile and an isotropic and global (stationary) covariance function used in the computation of $\bar{C}$, the resulting covariance matrix, $\bar{C}$, is of simple Toeplitz form. The inversion of such a covariance matrix requires a numerical operation proportional to N^2 compared to N^3 in case of conventional inversions. If we have observations forming a Cartesian grid with the properties explained above, then $\bar{C}$ is of block-Toeplitz form. The inversion of such a block Toeplitz matrix requires roughly $(N^3 P^2)$ numerical operations, where N in equal to the number of observations along a single profile and P is equal to the number of profiles, compared to $(N^3 P^3)$ in the case of classical inversions. Another advantage of Toeplitz inversion algorithms over classical inversion algorithms are that we only have to store one row in the case of simple Toeplitz matrices and one-block row in the case of block Toeplitz matrices. The only disadvantages in the application of Toeplitz algorithms are the following:

a) The requirement of gridded data consisting of equally weighted and spaced observations along, at least, a certain direction.

b) The requirement of an isotropic and global (stationary) covariance function in the computation of $\bar{C}$. This is also assumed in regular collocation.

In our computations of gravity anomalies from geoid heights we used the theoretical covariance function given by the subroutine COVA of Tscherning and Rapp (1974). This function satisfies the second requirement. In order to satisfy the first requirement we created gridded data from GEOS-3 altimeter data using the

subroutine PREDICT of the Geodetic Science Plotting Package (GSPP) written by Sünkel (1979). By using the subroutine PREDICT, it took roughly 1 second CPU time on Amdahl 470 of OSU for the prediction of 1000 points. The mean variance of original observations was assigned to each of the predicted data. These are the only approximations introduced when we compare Toeplitz solutions and rigorous least-squares solutions. So it is up to the user to decide between the gain in computer time, storage and approximations. In our cases, as we showed in Chapter 3, the approximations are negligible.

The simple and block Toeplitz algorithms are much more efficient than the classical inversion algorithms, but for very large N, say greater than 1000, they are time consuming and become inefficient as well. Yet another faster method is the frequency domain collocation discussed in Chapter 4. Here every quantity in the space domain is transformed into the frequency domain and the solution in performed there. For a fast solution we take advantage of the simple structure of the frequency domain covariance and cross-covariance matrices. In the limit as N → ∞ these matrices become diagonal and the method reduces to the generalized Wiener filtering. By the use of the frequency domain least-squares collocation, a considerable gain in computer time and storage is obtained in comparison with conventional least-squares collocation or least-squares collocation with Toeplitz inversion. However this method has disadvantages as well, such as the aforementioned disadvantages in the application of Toeplitz inversion algorithms and the errors due to edge effects.

The methods presented in Chapters 3 and 4 show that they are effective tools for a fast calculation of desired signals from large amounts of gravimetric data. In previous chapters these methods have been demonstrated for obtaining gravity anomalies from geoid heights.

References

Bath, Markus, "Spectral Analysis in Geophysics," Elsevier Scientific Publishing Company, New York, 1974.

Blackman, R. B., and J. W. Tukey, "The Measurement of Power Spectra," New York, Dover Publishing, 1959.

Brigham, E. O., "The Fast Fourier Transform," Prentice-Hall, Inc., New Jersey, 252 pages, 1974.

Byron, W. Frederick, Jr., and Robert W. Fuller, "Mathematics of Classical and Quantum Physics," Volume One, Addison-Wesley Publishing Company, 1969.

Cooley, J. W. and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," Math. Comput., 19: 297-301, 1965.

Cooley, J. W. and P.A.W. Lweis and P. D. Welch, "The Fast Fourier Transform and its Applications," IEEE Transactions on Education, Vol 12, No. 1, 27-34, March 1969.

Faddeyev, D. K, and Faddeyeva, V M., "Computational Methods of Linear Algebra," San Francisco, 1963.

Findley, F. D., "Applied Time Series Analysis," Academic Press, New York, 1978.

Forsythe, G. E. and C. B. Moler, "Computed Solution of Linear Algebraic Systems," Prentice-Hall, Englewood Cliffs, New Jersey, 1967.

Gray, R. M., "Toeplitz and Circulant Matrices II," Information Systems Laboratory, Technical Report No. 6504-1, Stanford Electronics Laboratories, Stanford University, Stanford, Californiz, 1977.

Harris, F. J., "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform," IEEE Proceedings, Vol. 66, pp. 51-83, 1978.

Heller, W. G., K. S. Tait, S. W. Thomas, "GEOFAST - A Fast Cravimetric Estimation Algorithm," The Analytic Sciences Corporation, 25 August 1977.

Jordan, Stanley K., "Fourier Physical Geodesy," Scientific Report No. 1, The Analytic Sceinces Corporation, Six Jacob Way, Reading Massachusetts, 01867, 1 March 1978.

Levinson, Norman,"The Wiener RMS Error Criterion in Filter Design and Prediction," J. Math. and Phys. 25, 4 261-278, Jan. 1947.

Mayhan, R. J., "Analysis of Discrete-Time and Continuous-Time Linear Systems, Part II," The Ohio State University Department of Electrical Engineering.

Otnes, R. K. and L. Enochson, "Digital Time Series Analysis," John Wiley & Sons, New York, 1972.

Papoulis, A., "The Fourier Integral and Its Applications," McGraw-Hill Book Company, Inc., New York, 318 pp., 1962.

Papoulis, A., "Probability, Random Variables and Stochastic Processes," McGraw-Hill Book Company, Inc., New York, 1965.

Papoulis, A., "Signal Analysis," McGraw-Hill Book Company, Inc., New York, 431 pp.,1977.

Parzen, Emanuel, "Time Series Analysis Papers," Holden Day Series, San Francisco, 1967.

Rabiner, L. R., and R. Gold, "Theory and Application of Digital Signal Processing," Prentice-Hall, Inc., New York, 1975.

Rapp, R. H., "Mean Gravity Anomalies and Sea Surface Heights Derived from GEOS-3 Altimeter Data," Department of Geodetic Science Report No. 168, The Ohio State University, Columbus, 1977.

Rapp, R. H., "Global Anomaly and Undulation Recovery Using GEOS-3 Altimeter Data," Department of Geodetic Science Report No. 284, The Ohio State University, Columbus, May 1979.

Rayner, John, "An Introduction to Spectral Analysis," Academic Press, Inc., 1971.

Rummel, R., L. Sjöberg, and R. H. Rapp, "The Determination of Gravity Anomalies from Geoid Heights," Department of Geodetic Science Report No. 269, The Ohio State University, Columbus, December 1977.

Runge, C., and H. König, "Die Grundlehren der Mathematischen Wissenschaften," Band XI, Vorlesungen Uber Numerisches Rechnen, Berlin: Springer, 1924.

Sünkel, Hans, "A General Surface Representation Module Designed for Geodesy," Department of Geodetic Science Report No. 292, The Ohio State University, Columbus, 1979.

Thomas, S. W., and W. G. Heller, "Efficient Estima on Techniques for Integrated Gravity Data Processing," The Analytic Sciences Corporation, Report No. AFGL-TR-76-0232, September 1976.

Trench, W. F., "An Algorithm for the Inversion of Finite Toeplitz Matrices," J. Siam 12, 3, 515-522, September 1964.

Tscherning, C. C., and R. H. Rapp, "Closed Covariance Expressions for Gravity Anomalies, Geoid Undulations, and Deflections of the Vertical Implied by Anomaly Degree Variance Models," Department of Geodetic Science Report No. 208, The Ohio State University, Columbus, 1974.

Tukey, J. W., "An Introduction to the Calculation of Numerical Spectrum Analysis," Spectral Analysis of Time Series, Edited by Bernard Harris, pp. 25-47, 1967.

Uotila, U. A., "Introduction to Adjustment Computations with Matrices," Unpublished Lecture Notes, Department of Geodetic Science, The Ohio State University, Columbus, 1967.

Wagner, C. A., "The Spectrum of the Geoid from Altimeter Data," Goddard Space Flight Center, Greenbelt, Maryland, 1977.

Zohar, S., "Toeplitz Matrix Inversion: The Algorithm of W. F. Trench," Journal of the Association for Computing Machinery, Vol. 16, No. 4, 592-601, October 1969.

Appendix 1.A

Fourier Series Representation of Periodic Functions and Parseval's Theorem

A periodic function $x(t)$ with period T, i.e. $x(t) = x(t + T)$ can be arbitrarily closely approximated by a harmonic polynomial of degree N and period T such that

$$x_N(t) = \frac{a_0}{2} + \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \qquad (1.A.1)$$

The closeness of $x_N(t)$ to $x(t)$ is usually computed by least-squares approximation over the interval $(0, T)$ defined by

$$v^2(t) = \int_T [x(t) - x_N(t)]^2 \, dt \qquad (1.A.2)$$

$v^2(t)$, whose parameters are $a_0, a_1, b_1, \ldots, a_N, b_N$, is to be minimized. So let us substitute eqn. (1.A.1) in (1.A.2) to obtain

$$v^2(t) = \int_T [x(t) - \frac{a_0}{2} - \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)]^2 \, dt \qquad (1.A.3)$$

Now let us take the derivatives of eqn. (1.A.3) with respect to the parameters:

$$\frac{\partial v^2(t)}{\partial a_m} = 2 \int_T [x(t) - \frac{a_0}{2} - \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)] \cos m\omega_0 t \, dt \qquad (1.A.4.a)$$

$$\frac{\partial v^2(t)}{\partial b_m} = 2 \int_T [x(t) - \frac{a_0}{2} - \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)] \sin m\omega_0 t \, dt \qquad (1.A.4.b)$$

The solutions to $\partial v^2(t)/\partial a_m = \partial v^2(t)/\partial b_m = 0$ give the desired parameters.

Recall the orthogonality relations defined by eqns. (1.2.a-b) in order to solve eqns. (1.A.4.a-b) for the parameters

$$a_n = \frac{2}{T} \int_T x(t) \cos n\omega_0 t \, dt, \qquad n = 0, 1, \ldots, N \qquad (1.A.5.a)$$

$$b_n = \frac{2}{T} \int_T x(t) \sin n\omega_0 t \, dt \qquad n = 1, 2, \ldots, N \qquad (1.A.5.b)$$

How close does eqn. (1.A.1) approximate the true function $x(t)$? In order to answer this question open the parenthesis of eqn. (1.A.2) to get

$$v^2(t) = \int_T x^2(t) \, dt + \int_T x_N^2(t) \, dt - 2\int_T x(t) \, x_N(t) \, dt \qquad (1.A.6)$$

Using the orthogonality relations mentioned above we can easily derive that

$$\frac{1}{T} \int_T x_N^2(t) \, dt = \frac{1}{T} \int_T [\frac{a_0}{2} + \sum_{n=1}^{N} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)]^2 \, dt = (\frac{a_0}{2})^2 + \frac{1}{2} \sum_{n=1}^{N} (a_n^2 + b_n^2) \qquad (1.A.7)$$

(Eqn. (1.A.7) is called <u>Parseval's Theorem</u>) and

$$\int_T x(t) \, x_N(t) \, dt \ = \ T\left(\frac{a_0}{2}\right)^2 + \frac{T}{2} \sum_{n=1}^{N} (a_n^2 + b_n^2) \tag{1.A.8}$$

Therefore eqn. (1.A.6) can be written as

$$v^2(t) \ = \ \int_T x^2(t) \, dt \ - \ [T\left(\frac{a_0}{2}\right)^2 + \frac{T}{2} \sum_{n=1}^{N} (a_n^2 + b_n^2)] \ \geq \ 0 \tag{1.A.9}$$

or

$$\infty \ < \ \frac{2}{T} \int_T x^2(t) \, dt \ \geq \ \left[\left(\frac{a_0}{2}\right)^2 + \sum_{n=1}^{N} (a_n^2 + b_n^2) \right] \tag{1.A.10}$$

Eqn. (1.A.10) is known as Bessel's inequality. From this inequality we can read that

$$\lim_{n \to \infty} a_n \ = \ \lim_{n \to \infty} b_n \ = \ 0$$

Now let $y(t)$ be equal to $x_N(t)$ in the limit as $N \to \infty$, i.e.

$$y(t) \ = \ \lim_{N \to \infty} x_N(t) \ = \ \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \tag{1.A.11}$$

where a_n and b_n are Fourier coefficients of square integrable function $x(t)$. By Bessel's inequality the partial sum of squares of the Fourier coefficients converges for any square integrable function. Therefore the function $y(t)$ is square integrable on $(0, T)$. Now consider

$$d(t) \ = \ x(t) - y(t) \tag{1.A.12}$$

then, $d(t)$ is also square integrable on $(0, T)$. By Shwarz' inequality we can write write

$$0 \ \leq \ \int_T |d(t)|^2 \, dt \ \leq \ \int_T |d(t)| \, dt \tag{1.A.13}$$

This expression is equal to zero by the "theorem of uniqueness of integrable functions". That is to say

$$d(t) = 0 \, , \qquad x(t) = y(t) \tag{1.A.14}$$

Thus all the Fourier coefficients in the Fourier expansion of $d(t)$ are zero according to the "theorem of uniqueness of continuous functions". Hence we can conclude that

$$x(t) \ = \ \lim_{N \to \infty} x_N(t) \ = \ \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \tag{1.A.15}$$

Appendix 1. B

Fejer Kernel and the Derivation

Substitute eqns. (1.3.a-b) in (1.44) to obtain

$$\bar{x}_N(t) = \frac{2}{T}\int_T\left\{\sum_{n=1}^{N}\left(1-\frac{n}{N+1}\right)x(\lambda)[\cos n\omega_0\lambda\cos n\omega_0 t + \sin n\omega_0\lambda\sin n\omega_0 t]\right\}d\lambda + \frac{1}{T}\int_T x(\lambda)\,d\lambda$$

$$= \frac{2}{T}\int_T\left\{\sum_{n=1}^{N}\left(1-\frac{n}{N+1}\right)x(\lambda)\cos[n\omega_0(\lambda-t)]\right\}d\lambda + \frac{1}{T}\int_T x(\lambda)\,d\lambda \qquad (1.B.1)$$

Now let $z = \omega_0(\lambda - t)$ to have

$$\bar{x}_N(t) = \frac{2}{T}\int_T\left[\sum_{n=1}^{N}\left(1-\frac{n}{N+1}\right)x(\lambda)\cos nz\right]d\lambda + \frac{1}{T}\int_T x(\lambda)\,d\lambda \qquad (1.B.2)$$

$$\bar{x}_N(t) = \frac{1}{T}\int_T F_{N+1}(z)\,x(\lambda)\,d\lambda \qquad (1.B.3)$$

where $F_{N+1}(z) = 1 + 2\sum_{n=1}^{N}(1-n/(N+1))\cdot\cos nz$ is called the Fejer kernel. The Fejer kernel consists of cosine functions as we can see from the definition above. Thus it is possible to express the Fejer kernel as a sum of cosine functions as follows:

$$F_{N+1}(z) = 1 + 2\left(1-\frac{n}{N+1}\right)\cos z + 2\left(1-\frac{2}{N+1}\right)\cos 2z + \ldots + 2\left(1-\frac{N}{N+1}\right)\cos Nz \qquad (1.B.4)$$

Multiplying eqn. (1.B.4) by $\sin z/2$ from both sides we obtain,

$$\sin\frac{z}{2}\,F_{N+1}(z) = \sin\frac{z}{2} + 2\left(1-\frac{1}{N+1}\right)\sin\frac{z}{2}\cos z + \ldots + 2\left(1-\frac{N}{N+1}\right)\sin\frac{z}{2}\cos Nz \qquad (1.B.5)$$

Using the trigonometric identity

$$2\sin x\cos y = \sin(y+x) - \sin(y-x) \qquad (1.B.6)$$

we can easily prove that

$$\sin\frac{z}{2}\,F_{N+1}(z) = \sin\frac{z}{2} + \left(1-\frac{1}{N+1}\right)\left(\sin\frac{3}{2}z - \sin\frac{1}{2}z\right) + \left(1-\frac{2}{N+1}\right)\left(\sin\frac{5}{2}z - \sin\frac{3}{2}z\right) + \ldots$$

$$+ \left(1-\frac{N}{N+1}\right)\left[\sin(N+\tfrac{1}{2})z - \sin(N-\tfrac{1}{2})z\right]$$

$$= \frac{1}{N+1}\sin\frac{z}{2} + \frac{1}{N+1}\sin\frac{3}{2}z + \ldots + \frac{N}{N+1}\sin(N+\tfrac{1}{2})z$$

$$= \frac{1}{N+1}\sum_{n=0}^{N}\sin(n+\tfrac{1}{2})z \qquad (1.B.7)$$

Thus the Fejer kernel can be expressed as

$$F_{N+1}(z) = \frac{1}{N+1}\sum_{n=0}^{N}\frac{\sin(n+\frac{1}{2}z)}{\sin\frac{1}{2}z} \qquad (1.B.8)$$

The expression for the Fejer kernel is not used in its present form. It can be simplified further, first by writing

$$F_{N+1}(z) = \frac{1}{N+1} \frac{1}{\sin^2 \frac{1}{2}z} \sum_{n=0}^{N} \sin \frac{z}{2} \sin(n + \frac{1}{2})z \qquad (1.B.9)$$

then, opening the sum of the expression above to get

$$F_{N+1}(z) = \frac{\sin^{-2}\frac{1}{2}z}{2(N+1)}[2\sin\tfrac{1}{2}z \sin\tfrac{1}{2}z + 2\sin\tfrac{1}{2}z \sin\tfrac{3}{2}z + \ldots + 2\sin\tfrac{1}{2}z \sin(N+\tfrac{1}{2})z] \qquad (1.B.10)$$

finally using the trigonometric identity

$$2 \sin x \sin y = \cos(x-y) - \cos(x+y) \qquad (1.B.11)$$

we obtain

$$F_{N+1}(z) = \frac{\sin^{-2}\frac{1}{2}z}{2(N+1)}[2 - 2\cos^2\tfrac{1}{2}z + \cos z - \cos 2z + \cos 2z + \ldots + \cos(N-1)z - \cos(N+1)z]$$

$$= \frac{\sin^{-2}\frac{1}{2}z}{2(N+1)}[2 - 2\cos^2\frac{z}{2} + \cos z - \cos(N+1)z]$$

$$= \frac{\sin^{-2}\frac{1}{2}z}{2(N+1)}[1 - \cos^2(N+1)\frac{z}{2} + \sin^2(N+1)\frac{z}{2}] \qquad (1.B.12)$$

The equation above is equivalent to

$$F_{N+1}(z) = \frac{1}{N+1}\left[\frac{\sin(N+1)(z/2)}{\sin(z/2)}\right]^2 \qquad (1.B.13)$$

Variances of Fourier Coefficients

Let us suppose that the function $x(k \cdot \Delta t)$ is independent and normally distributed $N(\mu, \sigma^2)$, and has the Fourier coefficients defined by eqn. (1.93), i.e.

$$\begin{Bmatrix} a_n \\ b_n \end{Bmatrix} = \frac{2}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) \begin{Bmatrix} \cos 2\pi k n/N \\ \sin 2\pi k n/N \end{Bmatrix}, \quad n = 0, 1, \ldots, N_H \qquad (1.C.1)$$

where $N_H = \begin{cases} (N/2) & \text{if } N \text{ even} \\ (N-1)/2 & \text{if } N \text{ odd} \end{cases}$

So if we denote

$$Y = [a_0, a_1, \ldots, a_{N_H}]^T, \quad X = [x(0), x(\Delta t), \ldots, x((N-1)\Delta t)]^T$$

$$G = \frac{2}{N} \begin{bmatrix} 1 & 1 & 1 \ldots \ldots \ldots 1 \\ 1 & \cos \dfrac{2\pi 1}{N} & \cos \dfrac{4\pi 1}{N} \ldots \ldots \cos \dfrac{2(N-1)\pi 1}{N} \\ \vdots & \vdots & \vdots \qquad\qquad \vdots \\ 1 & \cos \dfrac{2\pi N_H}{N} & \cos \dfrac{4\pi N_H}{N} \ldots \ldots \cos \dfrac{2(N-1)\pi N_H}{N} \end{bmatrix} \qquad (1.C.2)$$

then we can write

$$Y = G X \qquad (1.C.3)$$

The variance-covariance matrix of the function Y can be expressed as follows (Uotila, 1967):

$$\Sigma_Y = G \, \Sigma_X \, G^T \qquad (1.C.4)$$

where Σ_Y is the variance-covariance matrix of the Fourier coefficients (a_0, $a_1, \ldots, a_{N_H}$) and Σ_X is the variance-covariance matrix of $x(k \cdot \Delta t)$, $k = 0$, $1, \ldots, (N-1)$, which is diagonal with diagonal elements equal to σ^2 by assumption, i.e.

$$\Sigma_X = \text{diag}(\sigma^2, \sigma^2, \ldots, \sigma^2) \qquad (1.C.5)$$

Therefore, we have,

$$\Sigma_Y = \begin{bmatrix} \sigma_{a_0}^2 & 0 & \ldots \ldots & 0 \\ 0 & \sigma_{a_1}^2 & \ldots \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots \ldots & \sigma_{a_{N_H}}^2 \end{bmatrix} = \sigma^2 \, G \, G^T \qquad (1.C.6)$$

We know already that G and G^T are orthogonal such that

$$G \; G^T = \frac{4}{N^2}\begin{bmatrix} N & 0 & \cdots\cdots\cdots & 0 \\ 0 & (N/2) & \cdots\cdots\cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots\cdots\cdots & (N/2) \end{bmatrix} = \frac{2}{N}\begin{bmatrix} 2 & 0 & \cdots\cdots & 0 \\ 0 & 1 & \cdots\cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots\cdots & 1 \end{bmatrix} \qquad (1.C.7)$$

Thus eqn. (1.C.4) can be written as

$$\sigma_{a_n}^2 = \mathrm{var}(a_n) = \begin{cases} \dfrac{2}{N}\,\sigma^2 & \text{for } n = 1, 2, \ldots, (N_H - 1) \\[2mm] \dfrac{4}{N}\,\sigma^2 & \text{for } n = 0, \text{ or } n = N_H \text{ and } N \text{ even} \\[2mm] \dfrac{2}{N}\,\sigma^2 & \text{for } n = N_H \text{ and } N \text{ even} \end{cases} \qquad (1.C.8)$$

Similarly we can prove that

$$\sigma_{b_n}^2 = \mathrm{var}(b_n) = \begin{cases} \dfrac{2}{N}\,\sigma^2 & \text{for } n = 1, 2, \ldots, (N_H - 1) \\[2mm] 0 & \text{for } n = N_H \text{ and } N \text{ even} \\[2mm] \dfrac{2}{N}\,\sigma^2 & \text{for } n = N_H \text{ and } N \text{ odd} \end{cases} \qquad (1.C.9)$$

Appendix 1.D

The Fast Fourier Transform (FFT)

The number of operations required by the conventional Fourier transform method is proportional to N^2, N being the number of data points. On the other hand, the FFT algorithm requires a number of operations, which is proportional to $(N \log_2 N)$. The FFT is an algorithm by which the discrete Fourier transform (DFT) can be computed much more rapidly than by other available algorithms.

Let DFT be defined as in eqns. (1.65) and (1.66), namely,

$$X_n = \frac{1}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t)\, e^{-i 2 \pi k n / N} \tag{1.D.1}$$

$$x(k \cdot \Delta t) = \sum_{n=0}^{N-1} X_n\, e^{+i 2 \pi k n / N} \tag{1.D.2}$$

When N is a product, say $N = r \cdot s$, then the Fourier transform can be calculated in a two-stage process. That is to say, as if X_n and $x(k \cdot \Delta t)$ were defined on two dimensional (rxs) arrays with array indices (k_1, k_0) and (n_1, n_0) such that

$$k \longrightarrow (k_1, k_0) \tag{1.D.3}$$

$$k = k_1 \cdot r + k_0 \quad \text{for } k_0 = 0, 1, \ldots, (r-1) \text{ and } k_1 = 0, 1, \ldots, (s-1)$$

$$n \longrightarrow (n_1, n_0) \tag{1.D.4}$$

$$n = n_1 \cdot s + n_0 \quad \text{for } n_0 = 0, 1, \ldots, (s-1) \text{ and } n_1 = 0, 1, \ldots, (r-1)$$

If we define $\quad w_N = \exp(2 \pi i / N) \tag{1.D.5}$

and use eqns. (1.D.3) and (1.D.4), then we can write

$$w_N^{kn} = w_N^{k_1 n_1 r s} \cdot w_N^{k_0 n_1 s} \cdot w_N^{k_1 n_0 r} \cdot w_N^{k_0 n_0}$$

$$= \quad 1 \quad \cdot w_r^{k_0 n_1} \cdot w_s^{k_1 n_0} \cdot w_N^{k_0 n_0} \tag{1.D.6}$$

Thus eqn. (1.D.2) becomes

$$x(k \cdot \Delta t) = x[(k_1, k_0) \cdot \Delta t] = \sum_{n_0=0}^{s-1} \sum_{n_1=0}^{r-1} X_{(n_1, n_0)} \cdot w_r^{k_0 n_1} \cdot w_s^{k_1 n_0} \cdot w_N^{k_0 n_0} \tag{1.D.7}$$

If we denote

$$X_{(k_0, n_0)}^1 = w_N^{k_0 n_0} \sum_{n_1=0}^{r-1} X_{(n_1, n_0)} \cdot w_r^{k_0 n_1} \tag{1.D.8}$$

then,

$$x[(k_1, k_0) \cdot \Delta t] = \sum_{n_0=0}^{s-1} X_{(k_0, n_0)}^1 \cdot w_s^{k_1 n_0} \tag{1.D.9}$$

Notice that first we sum over n_1 in eqn. (1.D.8) to form an intermediate array $X^1_{(k_0, n_0)}$ and then sum over n_0. Hence $X^1_{(k_0, n_0)}$ is $w_N^{k_0 n_0}$ times a set of r-term Fourier transforms. Finally eqn. (1.D.9) is a set of s-term Fourier series with $X^1_{(k_0, n_0)}$ as coefficients. The total computation amounts to

$$r^2 \cdot s + s^2 \cdot r = N(r+s) \tag{1.D.10}$$

for this particular two-stage process.

Similarly, it can be shown that if $N = r_1 \times r_2 \times \ldots \times r_s$, then the number of operations is equal to

$$N \times (r_1 + r_2 + \ldots + r_s)$$

If we further assume that $r_1 = r_2 = \ldots = r_s$, then $N = r^s$ and the number of operations is equal to

$$N \cdot m \cdot r = N(\log_r N) r = N(\log_2 N) r / \log_2 r \tag{1.D.11}$$

when r is equal to three, we obtain the lowest number of calculations. This can be proven by minimizing the above equation, i.e.

$$y(r) = N(\log_2 N) r / \log_2 r \qquad \text{to be minimized} \tag{1.D.12}$$

The solution to $y'(r) = 0$ gives the optimum factor r.

$$y'(r) = N(\log_2 N) \left[\log_2 r - r \frac{\partial}{\partial r}(\log_2 r)\right]/(\log_2 r)^2 = 0 \tag{1.D.13}$$

This is equivalent to

$$\log_2 r - r \frac{\partial}{\partial r}(\log_2 r) = \log_2 r - \frac{1}{\ln 2} = 0$$

The solution of the above equation is.

$$r = 2^{1.442695} \approx 3 \qquad \text{(nearest integer)} \tag{1.D.14}$$

However for $r = 2, 4, 8, \ldots$ we can further reduce calculations by avoiding multiplications when the powers of w_N are simple numbers like $\pm 1, \pm i$. Therefore the most commonly used FFT algorithms use $r = 2$, i.e. $N = 2^s$. Generally a seperate derivation is given for this case. Let

$$x(k \cdot \Delta t) \longleftrightarrow X_n \qquad k, n = 0, 1, \ldots, (N-1) \tag{1.D.15}$$

$$\left.\begin{array}{l} x(2k' \cdot \Delta t) \longleftrightarrow X'_n \\ x[(2k'+1) \cdot \Delta t] \longleftrightarrow X''_{n'} \end{array}\right\} \qquad k'n' = 0, 1, \ldots, ((N/2)-1) \tag{1.D.16}$$

then the discrete Fourier transform (DFT) $X_{n'}$ can be written as

$$X_{n'} = \frac{1}{N} \sum_{k=0}^{N-1} x(k \cdot \Delta t) w_N^{-kn'} \tag{1.D.17}$$

using eqn. (1.D.16) we have

$$X'_n = \frac{1}{N}\left\{ \sum_{k'=0}^{(N/2)-1} [x(2k'\cdot\Delta t)\cdot w_N^{-2k'n'} + x((2k'+1)\cdot\Delta t)\cdot w_N^{-(2k'+1)n'}]\right\} \qquad (1.D.18)$$

But we know that $w_N^2 = w_{N/2}$, therefore

$$X'_n = \frac{1}{2}\left\{\frac{1}{(N/2)}\sum_{k'=0}^{(N/2)-1} x(2k'\cdot\Delta t)\cdot w_{N/2}^{-k'n'} + \frac{1}{(N/2)}\sum_{k'=0}^{(N/2)-1} x[(2k'+1)\cdot\Delta t]\cdot w_{N/2}^{-k'n'}\cdot w_N^{-n'}\right\} \qquad (1.D.19)$$

which can be written as

$$X'_n = \frac{1}{2}[X'_{n'} + X''_{n'}\, w_N^{-n'}] \qquad (1.D.20)$$

and

$$X'_{n'+(N/2)} = \frac{1}{2}[X'_{n'} - X''_{n'}\, w_N^{-n'}] \qquad (1.D.21)$$

Notice that $X'_{n'}$ and $X''_{n'}$ in eqn. (1.D.21) are also calculated through "doubling". So this successive doubling continues until it is no longer a multiple integer of 2.

When the FFT algorithm is used, we avoid computations by a factor proportional to α_N, where α_N is a constant times the ratio of the number of operations in the conventional method over that of FFT, i.e.

$$\alpha_n \cong c\, N^2/(N\log_2 N) = c\, N/\log_2 N \qquad (1.D.22)$$

where c is a constant greater than one. The computational reduction by a factor α_N (assuming $c = 1$) for various N are tabulated below:

N	α_N
2	2
16	4
256	32
1024	341
4096	341
8192	630
16384	1170

Since less operations are required in FFT computations, the truncation error is considerably smaller than that of conventional computations. Thus FFT gives more accurate results.

In order to demonstrate a simple application of the FFT algorithm, the discrete data given at N points were transformed by the "FFTR" subroutine of the IMSL library and by the conventional algorithm. The results are given below:

N	Conventional Method (sec.)	FFT Method (sec.)
256	2.52	0.06
512	9.92	0.13

The time required for computing Fourier transform by conventional algorithms and by the FFT algorithm is illustrated in Figure 1.D.1 below (extracted from Cooley, 1969).
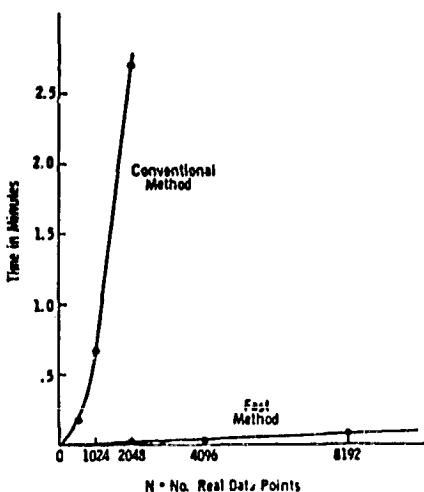
Figure 1. D. 1.   Time required for calculation of Fourier transform of real data on IBM 7094 using FORTRAN with conventional and fast methods.

We have tried to introduce the FFT algorithm as a computational tool here. For detailed discussions see Brigham (1974) and Rath (1974).

Appendix 3.A. A FORTRAN Program for the Inversion of Simple Toeplitz Matrices.

```
C     SUBROUTINE TOEPL(B,C,X,F,MAX)                                  A    10
C                                                                    A    20
C     -------------------------------------------------------------- A    30
C     FUNCTION              - TO COMPUTE THE TOEPLITZ INVERSE OF  'T' A    40
C                             AND MULTIPLY IT BY THE DATA VECTOR  'X' A    50
C                             TO GET  'F'                            A    60
C     PARAMETERS     B      - THE INTERMEDIATE VECTOR OF LENGTH -MAX- A    70
C                             TO GET THE INVERSE                     A    80
C                    C      - VECTOR OF LENGTH -MAX-.CONTAINS THE FIRST A  90
C                             'T' ON INPUT AND REPLACED BY  BY INVERSE A 100
C                             ROW OF  'T' ON INPUT AND REPLACED BY THE A 110
C                             INVERSE LATER                          A   120
C                    X      - DATA VECTOR OF LENGTH -MAX-            A   130
C                    F      - VECTOR OF LENGTH -MAX- AND IS PRODUCT OF A 140
C                             TINV * X                               A   150
C                    MAX    - THE DIMENSION OF THE TOEPLITZ MATRIX 'T' A  160
C     -------------------------------------------------------------- A   170
C                                                                    A   180
      SUBROUTINE TOEPL (B,C,X,F,MAX)                                 A   190
      IMPLICIT REAL*8(A-H,O-Z)                                       A   200
      DIMENSION B(1), C(1), X(1), F(1)                              A   210
C     ...                                                            A   220
      B(MAX)=1.D0                                                    A   230
C     ..   COMPUTE  'AA' AND 'E'  VARIABLES OF  THE ALGORITHM       A   240
      DO 13 K=1,MAX                                                 A   250
      E=0.D0                                                         A   260
      AA=0.D0                                                        A   270
      DO 11 L=1,K                                                    A   280
      NA=MAX-L+1                                                     A   290
      AA=AA+B(NA)*C(L)                                               A   300
      E=E+B(NA)*C(K-L+2)                                             A   310
   11 CONTINUE                                                       A   320
      AA=1.D0/AA                                                     A   330
      IF(K.EQ.MAX) GO TO 13                                         A   340
C     COMPUTE    'B'  VECTOR  USING RECURSIVE ALGORITHM             A   350
      EAA=-E*AA                                                      A   360
      NB=MAX-K                                                       A   370
      B(NB)=EAA                                                      A   380
      IF(K.EQ.1) GO TO 13                                           A   390
      KHALF=K/2                                                      A   400
      KR=MOD(K,2)                                                    A   410
      DO 12 LL=1,KHALF                                              A   420
      NB1=NB+LL                                                      A   430
      NB2=MAX-LL                                                     A   440
      T1=B(NB1)                                                      A   450
      B(NB1)=B(NB1)+EAA*B(NB2)                                      A   460
      IF(KR.EQ.0.AND.LL.EQ.KHALF) GO TO 12                         A   470
      B(NB2)=B(NB2)+EAA*T1                                          A   480
   12 CONTINUE                                                       A   490
   13 CONTINUE                                                       A   500
C     ...  COMPUTE THE LAST ROW OF THE INVERSE                     A   510
      DO 14 I=1,MAX                                                A   520
   14 C(I)=AA*B(I)                                                  A   530
C     ...  MULTIPLY THE COMPUTED ELEMENTS OF THE INVERSE BY THE CORR.PART A  540
C          OF   X-VECTOR                                           A   550
      NI=MAX                                                        A   560
      DO 15 NJ=1,MAX                                               A   570
   15 CALL TMULT (C,X,F,NI,NJ,MAX)                                 A   580
C     ...  COMPUTE THE REMAINING ELEMENTS OF THE INVERSE AND MULTIPLY BY A 590
C     THE CORRESPONDING ELEMENTS OF THE X-VECTOR                  A   600
      N=1                                                           A   610
      NN=MAX                                                        A   620
      M=MAX-1                                                       A   630
      MHALF=MAX/2+1                                                A   640
      MM=M+MHALF                                                    A   650
      DO 17 I=MHALF,M                                              A   660
      II=MM-I                                                       A   670
      N=N+1                                                         A   680
      NN=NN-1                                                       A   690
      NL=I-MHALF+1                                                 A   700
      T1=AA*B(NL)                                                   A   710
      T2=AA*B(II)                                                   A   720
      DO 16 J=N,NN                                                 A   730
      NB1=M-J+I                                                     A   740
      T3=B(NB1)*T1                                                  A   750
      T4=B(J)*T2                                                    A   760
      C(J)=C(J+1)-T3+T4                                            A   770
   16 CALL TMULT (C,X,F,II,J,MAX)                                 A   780
   17 CONTINUE                                                      A   790
      RETURN                                                        A   800
      END                                                           A   810
```

```
      SUBROUTINE TMULT (C,X,F,NI,NJ,MAX)                               B    10
C     ---------------------------------------------------------------  B    20
C        ... THIS SUBROUTINE CONSIDERS SYMMETRY,PERSYMMETRY AND COMPUTES B  30
C     THE PRODUCT                                                      B    40
C        ... C(NJ)   IS THE   (NI,NJ)TH,(NJ,NI)TH,(NII,NJJ)TH,(NJJ,NII)TH ELE B 50
C           OF THE INVERSE                                             B    60
C        ... NI = ROW NUMBER OF THE INVERSE                            B    70
C     NJ = COLUMN NUMBER OF THE INVERSE MATRIX                         B    80
C     ---------------------------------------------------------------  B    90
      IMPLICIT REAL*8(A-H,O-Z)                                         B   100
      DIMENSION C(1), X(1), F(1)                                       B   110
      NII=MAX-NJ+1                                                     B   120
      NJJ=MAX-NI+1                                                     B   130
      F(NI)=F(NI)+C(NJ)*X(NJ)                                          B   140
      IF(NI.EQ.NJ.AND.NI.EQ.NII) RETURN                               B   150
      IF(NI.EQ.NJ) GO TO 1                                            B   160
      F(NJ)=F(NJ)+C(NJ)*X(NI)                                          B   170
      IF(NI.EQ.NII) RETURN                                            B   180
      F(NJJ)=F(NJJ)+C(NJ)*X(NII)                                       B   190
    1 F(NII)=F(NII)+C(NJ)*X(NJJ)                                       B   200
      RETURN                                                          B   210
      END                                                             B   220
```

Appendix 3.B. A FORTRAN Program for the Inversion of Block Toeplitz Matrices.

```
C      SUBROUTINE BTOEPL(B,DUMMY,C,E,AA,EAA,T1,TEMP,T2,BG,T3,T4,AINV,IR,    A    10
C      IC)                                                                 A    20
C                                                                          A    30
C      ---------------------------------------------------------------     A    40
C      FUNCTION                     - TO COMPUTE THE INVERSE OF BLOCK TOEPLITZ  A    50
C                                     'T' AND MULTIPLY IT BY THE DATA VECTOR    A    60
C                                     'X'  TO GET  'F'                      A    70
C      PARAMETERS            T       - THE BLOCK TOEPLITZ MATRIX TO BE INVERTED A    80
C                            B       - THE INTERMEDIATE MATRIX OF DIMEN.(IR,IC) A    90
C                                      TO OBTAIN THE INVERSE                A   100
C                            C       - FIRST BLOCK ROW MATRIX OF 'T' MATRIX ON  A   110
C                                      LATER REPLACED BY THE BLOCK INVERSE ROWS A   120
C                                      INPUT,LATER REPLACED BY BLOCK INVERSE    A   130
C                                      ROWS                                 A   140
C                            X       - DATA VECTOR OF LENGTH -MAX-          A   150
C                            F       - VECTOR OF LENGTH -MAX- AND IS PRODUCT OF A   160
C                                      TINV * X                            A   170
C                            IR      - NO.OF MAXIMUM ROWS POSSIBLE          A   180
C                            IC      - NO.OF MAXIMUM COLUMNS POSSIBLE       A   190
C                            IMAX    - DIMENSION OF THE SQUARE SUBBLOCKS OF 'T' A   200
C                            JMAX    - DIMENSION OF  'T'  MATRIX           A   210
C                            MAX     - RATIO OF JMAX OVER  AX, MAX=JMAX/IMAX A   220
C      ( B & DUMMY ) CAN SHARE THE SAME LOCATIONS ,   CALLING PROGRAM IS    A   230
C      CALL BTOEPL(B,B,C,E,AA,EAA,T1,TEMP,T2,BG,T3,T4,AINV,IR,IC)          A   240
C      ---------------------------------------------------------------     A   250
C                                                                          A   260
       SUBROUTINE BTOEPL (B,DUMMY,C,E,AA,EAA,T1,TEMP,T2,BG,T3,T4,AINV,IR,  A   270
      1IC)                                                                 A   280
       IMPLICIT REAL*8(A-H,O-Z)                                           A   290
       DIMENSION B(IR,IC),  DUMMY(IR,IC),  C(IR,IC)                       A   300
       DIMENSION E(IR,IR),  AA(IR,IR),  EAA(IR,IR),  T1(IR,IR),  T2(IR,IR) A   310
       DIMENSION T3(IR,IR),  T4(IR,IR),  BG(IR,IR),  TEMP(IR,IR),  AINV(1) A   320
       COMMON /DIMEN/ IMAX,JMAX,MAX                                       A   330
       LOGICAL PICK                                                       A   340
C      ... INITIALIZE  -B- MATRIX TO ZERO                                 A   350
       CALL DZERO (B,IR*IC)                                               A   360
       INS=IR*IR                                                          A   370
       NBM=(MAX-1)*IMAX                                                   A   380
       DO 11 I=1,IMAX                                                     A   390
       JJ=NBM+I                                                           A   400
   11  B(I,JJ)=1.D0                                                       A   410
C      ...                                                                A   420
C      .. COMPUTE  'AA' AND 'E'  MATRICES OF  THE ALGORITHM               A   430
       DO 16 K=1,MAX                                                      A   440
       DO 12 I=1,IMAX                                                     A   450
       DO 12 J=1,IMAX                                                     A   460
       E(I,J)=0.D0                                                        A   470
   12  AA(I,J)=0.D0                                                       A   480
       DO 13 L=1,K                                                        A   490
       NBC=(L-1)*IMAX                                                     A   500
       NBA=(MAX-L)*IMAX                                                   A   510
       ID=2                                                               A   520
       CALL MULT (B,C,TEMP,TEMP,BG,IR,IC,NBA,NBC,ID)                      A   530
       CALL ABS (AA,BG,AA,IR,IC)                                          A   540
       IF(K.EQ.MAX) GO TO 13                                             A   550
       NBC=(K-L+1)*IMAX                                                   A   560
       CALL MULT (B,C,TEMP,TEMP,BG,IR,IC,NBA,NBC,ID)                      A   570
       CALL ABS (E,BG,E,IR,IC)                                            A   580
   13  CONTINUE                                                           A   590
C      .. INVERT  THE SQUARE MATRIX  'AA'                                 A   600
       CALL AAINV (AA,AINV,IR)                                           A   610
       IF(K.EQ.MAX) GO TO 16                                             A   620
C      ... COMPUTE   'B'  MATRIX  USING RECURSIVE ALGORITHM              A   630
C      ...  'T1'    IS THE THE SQUARE MATRIX  TO BE ADDED TO  THE PRODUCT A   640
C      'T2'    TIMES     'EAA'     ,WHERE  EAA=E* AA                      A   650
       ID=1                                                               A   660
       CALL MULT (DUMMY,DUMMY,E,AA,EAA,IR,IC,NX,NX,ID)                    A   670
       PICK=.FALSE.                                                       A   680
       DO 14 I=1,IMAX                                                     A   690
       DO 14 J=1,IMAX                                                     A   700
   14  EAA(I,J)=-EAA(I,J)                                                 A   710
       NB=(MAX-K-1)*IMAX                                                  A   720
       CALL PUTIN (B,EAA,IR,IC,NB,PICK)                                   A   730
       IF(K.EQ.1) GO TO 16                                              A   740
       KHALF=K/2                                                          A   750
       PICK=.TRUE.                                                        A   760
       KR=MOD(K,2)                                                        A   770
       DO 15 LL=1,KHALF                                                   A   780
       NB1=NB+LL*IMAX                                                     A   790
       CALL PUTIN (B,T1,IR,IC,NB1,PICK)                                   A   800
       NB2=NBM-LL*IMAX                                                    A   810
       CALL PUTIN (B,T2,IR,IC,NB2,PICK)                                   A   820
```

```
C           ...                                                              A   830
            ID=1                                                             A   840
            CALL  MULT  (DUMMY,DUMMY,EAA,T2,T3,IR,IC,NX,NX,ID)               A   850
            CALL  AABS  (B,T1,T3,IR,IC,NP1)                                  A   863
            IF(KR.EQ.0.AND.LL.EQ.KHALF)  GO TO 15                           A   870
            CALL  MULT  (DUMMY,DUMMY,EAA,T1,T3,IR,IC,NX,NX,ID)              A   880
            CALL  AABS  (B,T2,T3,IR,IC,NB2)                                  A   890
     15     CONTINUE                                                         A   900
     16     CONTINUE                                                         A   910
C           ...  COMPUTE  THE  LAST  BLOCK  ROW  OF  THE  INVERSE            A   920
            ID=3                                                             A   930
            DO  17  I=1,MAX                                                  A   940
            NL=(I-1)*IMAX                                                    A   950
            CALL  MULT  (B,C,AA,TEMP,TEMP,IR,IC,NL,NL,ID)                   A   960
     17     CONTINUE                                                         A   970
            PICK=.TRUE.                                                      A   980
            DO  18  J=1,MAX                                                  A   990
            NI=(MAX-1)*IMAX                                                  A  1000
            NJ=(J-1)*IMAX                                                    A  1010
            CALL  PUTIN  (C,T1,IR,IC,NJ,PICK)                               A  1020
     18     CALL  BMULT  (T1,X,F,NI,NJ,IR,IC)                              A  1030
C           ...                                                              A  1040
            N=1                                                              A  1050
            NN=MAX                                                           A  1060
            M=MAX-1                                                          A  1070
            MHALF=MAX/2+1                                                    A  1080
            MP=M+MHALF                                                       A  1090
            DO  20  I=MHALF,M                                                A  1100
            II=MP-I                                                          A  1110
            N=N+1                                                            A  1120
            NN=NN-1                                                          A  1130
C           ...  COMPUTE      ' T1 = B(N,N-S)TRANSP * AA     '              A  1140
            ID=3                                                             A  1150
            NL=(M-II)*IMAX                                                   A  1160
            CALL  MULT  (B,DUMMY,AA,TEMP,T1,IR,IC,NL,NL,ID)               A  1170
C           ..   COMPUTE      ' T2 = B(N,S-1)TRANSPOSE  * AA     '          A  1180
            NL=(II-1)*IMAX                                                   A  1190
            CALL  MULT  (B,DUMMY,AA,TEMP,T2,IR,IC,NL,NL,ID)               A  1200
C           ...  COMPUTE  THE  REMAINING  SUBBLOCKS  OF  THE  INVERSE  AND  MULTIPLY  BY   A  1210
C                X-VECTOR                                                    A  1220
            ID=4                                                             A  1230
            DO  19  J=1,NN                                                   A  1240
            NB1=(N-J)*IMAX                                                   A  1250
            NB2=(J-1)*IMAX                                                   A  1260
            CALL  MULT  (B,DUMMY,T1,TEMP,T3,IR,IC,NB1,NB1,ID)             A  1270
            CALL  MULT  (B,DUMMY,T2,TEMP,T4,IR,IC,NB2,NB2,ID)             A  1280
            CALL  ZZTT  (C,T3,T4,IR,IC,NB2,TEMP)                          A  1290
     19     CALL  BMULT  (TEMP,X,F,NL,NB2,IR,IC)                          A  1300
     20     CONTINUE                                                         A  1310
            RETURN                                                           A  1320
            END                                                              A  1330
```

```fortran
      SUBROUTINE AAINV (AA,AINV,IR)                                     B  10
C     --------------------------------------------------               B  20
C     ... THIS SUBROUTINE COMPUTES THE INVERSE OF AA-MATRIX            B  30
C     ... AA IS THE MATRIX TO BE INVERTED ON INPUT AND INVERSE ON OUTPUT B 40
C                                                                      B  50
      IMPLICIT REAL*8(A-H,O-Z)                                         B  60
      DIMENSION AA(IR,IR), AINV(1)                                     B  70
      COMMON /DIMEN/ IMAX,JMAX,MAX                                     B  80
      IN(I)=(I*I-I)/2                                                  L  90
      N=0                                                              B 100
      DO 1 I=1,IMAX                                                    B 110
      DO 1 J=1,I                                                       B 120
      N=N+1                                                            B 130
    1 AINV(N)=AA(I,J)                                                  B 140
      CALL DSINV (AINV,IMAX,1.D-14,IER)                               B 150
      IF(IER.EQ.0) GO TO 2                                             B 160
      WRITE (6,7) IER                                                  B 170
      STOP 10                                                          B 180
    2 CONTINUE                                                         B 190
      DO 6 I=1,IMAX                                                    B 200
      DO 6 J=1,IMAX                                                    B 210
      IF(J-I) 3,4,4                                                    B 220
    3 N=IN(I)+J                                                        B 230
      GO TO 5                                                          B 240
    4 N=IN(J)+I                                                        B 250
    5 CONTINUE                                                         B 260
    6 AA(I,J)=AINV(N)                                                  B 270
      RETURN                                                           B 280
C                                                                      B 290
C                                                                      B 300
    7 FORMAT (10X,'INVERSION ERROR   IER =',I4)                        B 310
      END                                                              B 320
```

```fortran
      SUBROUTINE PUTIN (A,B,IR,IC,NA,PICK)                             C  10
C     --------------------------------------------------               C  20
C     ... -B-   IS THE MATRIX TO BE INSERTED FOLLOWING (NA)TH COLUMN OF C 30
C     A-MATRIX IF PICK=.FALSE.,OR ELSE TO PICK FROM 'A' IF PICK=.TRUE.  C 40
C                                                                      C  50
      IMPLICIT REAL*8(A-H,O-Z)                                         C  60
      DIMENSION A(IR,IC), B(IR,IR)                                     C  70
      COMMON /DIMEN/ N,M                                               C  80
      LOGICAL PICK                                                     C  90
      IF(PICK) GO TO 2                                                 C 100
      DO 1 I=1,N                                                       C 110
      DO 1 J=1,N                                                       C 120
      NN=NA+J                                                          C 130
    1 A(I,NN)=B(I,J)                                                   C 140
      RETURN                                                           C 150
    2 DO 3 I=1,N                                                       C 160
      DO 3 J=1,N                                                       C 170
      NN=NA+J                                                          C 180
    3 B(I,J)=A(I,NN)                                                   C 190
      RETURN                                                           C 200
      END                                                              C 210
```

```fortran
      SUBROUTINE ABS (A,B,R,IR,IC)                                     D  10
C     --------------------------------------------------               D  20
C     .. SUMS UP  'A'   AND   'B'   MATRICES  TO GET   'R'   MATRIX    D  30
C                                                                      D  40
      IMPLICIT REAL*8(A-H,O-Z)                                         D  50
      DIMENSION A(IR,IR), B(IR,IR), R(IR,IR)                           D  60
      COMMON /DIMEN/ N,M                                               D  70
      DO 1 I=1,N                                                       D  80
      DO 1 J=1,N                                                       D  90
    1 R(I,J)=A(I,J)+B(I,J)                                             D 100
      RETURN                                                           D 110
      END                                                              D 120
```

```
      SUBROUTINE AABS (A,T,TT,IR,IC,NB)                              E   10
C     --------------------------------                              E   20
C     ... SUMS UP T AND TT-MATRICES AND INSERTS IN A-MATRIX FOLLOWING E  30
C     (NB) TH   COLUMN                                              E   40
C                                                                   E   50
      IMPLICIT REAL*8(A-H,O-Z)                                      E   60
      DIMENSION T(IR,IR), A(IR,IC), TT(IR,IR)                       E   70
      COMMON /DIMEN/ N,M,MAX                                        E   80
C     . . . .                                                       E   90
      DO 1 I=1,N                                                    E  100
      DO 1 J=1,N                                                    E  110
      NN=NB+J                                                       E  120
    1 A(I,NN)=T(I,J)+TT(I,J)                                        E  130
      RETURN                                                        E  140
      END                                                           E  150


      SUBROUTINE MULT (A,C,D,E,R,IR,IC,NJ1,NJ2,ID)                  F   10
      IMPLICIT REAL*8(A-H,O-Z)                                      F   20
      DIMENSION A(IR,IC), C(IR,IC), D(IR,IR), E(IR,IR), R(IR,IR)    F   30
      COMMON /DIMEN/ N,M                                            F   40
C     . . . . . .                                                   F   50
      GO TO (1,3,5,7,9), ID                                         F   60
    1 CONTINUE                                                      F   70
      DO 2 I=1,N                                                    F   80
      DO 2 J=1,N                                                    F   90
      R(I,J)=0.D0                                                   F  100
      DO 2 K=1,N                                                    F  110
    2 R(I,J)=R(I,J)+D(I,K)*E(K,J)                                   F  120
      RETURN                                                        F  130
    3 DO 4 I=1,N                                                    F  140
      DO 4 J=1,N                                                    F  150
      R(I,J)=0.D0                                                   F  160
      NC=NJ2+J                                                      F  170
      DO 4 K=1,N                                                    F  180
      NA=NJ1+K                                                      F  190
    4 R(I,J)=R(I,J)+A(I,NA)*C(K,NC)                                 F  200
      RETURN                                                        F  210
    5 DO 6 I=1,N                                                    F  220
      DO 6 J=1,N                                                    F  230
      JJ=NJ1+J                                                      F  240
      C(I,JJ)=0.D0                                                  F  250
      DO 6 K=1,N                                                    F  260
    6 C(I,JJ)=C(I,JJ)+D(I,K)*A(K,JJ)                                F  270
      RETURN                                                        F  280
    7 DO 8 I=1,N                                                    F  290
      DO 8 J=1,N                                                    F  300
      JJ=NJ1+J                                                      F  310
      R(I,J)=0.D0                                                   F  320
      DO 8 K=1,N                                                    F  330
    8 R(I,J)=R(I,J)+D(I,K)*A(K,JJ)                                  F  340
      RETURN                                                        F  350
    9 DO 10 I=1,N                                                   F  360
      JJ=NJ1+I                                                      F  370
      DO 10 J=1,N                                                   F  380
      R(I,J)=0.D0                                                   F  390
      DO 10 K=1,N                                                   F  400
   10 R(I,J)=R(I,J)+A(K,JJ)*D(K,J)                                  F  410
      RETURN                                                        F  420
      END                                                           F  430


      SUBROUTINE ZZTT (Z,T3,T4,IR,IC,NB,T5)                         G   10
      IMPLICIT REAL*8(A-H,O-Z)                                      G   20
      DIMENSION Z(IR,IC), T3(IR,IR), T4(IR,IR), T5(IR,IR)           G   30
      COMMON /DIMEN/ N,M                                            G   40
C     . . .                                                         G   50
      M1=NB+N                                                       G   60
      DO 1 I=1,N                                                    G   70
      DO 1 J=1,N                                                    G   80
      MM=M1+J                                                       G   90
      NN=NB+J                                                       G  100
      Z(I,NN)=Z(I,MM)-T3(I,J)+T4(I,J)                               G  110
    1 T5(I,J)=Z(I,NN)                                               G  120
      RETURN                                                        G  130
      END                                                           G  140
```

```
      SUBROUTINE BMULT (C,X,F,NI,NJ,IR,IC)                              H   10
C     -------------------------------                                   H   20
C     ..   COMPUTE PRODUCT OF   'T'   AND PROPER PART OF   'X'  TO GET  'F'   H   30
      IMPLICIT REAL*8(A-H,O-Z)                                          H   40
      DIMENSION X(1), F(1), C(IR,IR)                                    H   50
      COMMON /DIMEN/ IMAX,JMAX,MAX                                      H   60
      II=NI/IMAX+1                                                      H   70
      JJ=NJ/IMAX+1                                                      H   80
      MM=MAX-JJ+1                                                       H   90
      NII=(MAX-1)*IMAX-NJ                                               H  100
      NJJ=(MAX-1)*IMAX-NI                                               H  110
      CALL AB (C,X,F,NI,NJ,IR,IC)                                       H  120
      IF(II.EQ.JJ.AND.II.EQ.MM) RETURN                                 H  130
      IF(II.EQ.JJ) GO TO 1                                             H  140
      CALL ABTR (C,X,F,NJ,NI,IR,IC)                                    H  150
      IF(II.EQ.MM) RETURN                                              H  160
      CALL AB (C,X,F,NJJ,NII,IR,IC)                                    H  170
    1 CALL ABTR (C,X,F,NII,NJJ,IR,IC)                                  H  180
      RETURN                                                           H  190
      END                                                              H  200



      SUBROUTINE AB (T,X,F,NI,NJ,IR,IC)                                 I   10
      IMPLICIT REAL*8(A-H,O-Z)                                         I   20
      DIMENSION T(IR,IR), X(1), F(1)                                   I   30
      COMMON /DIMEN/ IMAX,JMAX,MAX                                     I   40
C     ........                                                         I   50
C     ... COMPUTES THE PRODUCT OF   'T'   AND PROPRE PART OF   'X'   AND   I   60
C         ADDS TO 'F'                                                  I   70
      DO 1 I=1,IMAX                                                    I   80
      II=NI+I                                                          I   90
      DO 1 J=1,IMAX                                                    I  100
      JJ=NJ+J                                                          I  110
    1 F(II)=F(II)+T(I,J)*X(JJ)                                         I  120
      RETURN                                                           I  130
      END                                                             I  140



      SUBROUTINE ABTR (T,X,F,NI,NJ,IR,IC)                               J   10
      IMPLICIT REAL*8(A-H,O-Z)                                         J   20
      DIMENSION T(IR,IR), X(1), F(1)                                   J   30
      COMMON /DIMEN/ IMAX,JMAX,MAX                                     J   40
C     ..............                                                   J   50
      DO 1 I=1,IMAX                                                    J   60
      II=NI+I                                                          J   70
      DO 1 J=1,IMAX                                                    J   80
      JJ=NJ+J                                                          J   90
    1 F(II)=F(II)+T(J,I)*X(JJ)                                         J  100
      RETURN                                                           J  110
      END                                                             J  120
```