Reports of the Department of Geodetic Science

Report No. 310

NUMERICAL METHODS FOR HARMONIC ANALYSIS ON THE SPHERE

by

Oscar L. Colombo

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

March 1981

Foreword

This report was prepared by Dr. Oscar L. Colombo, Post Doctoral Researcher, Department of Geodetic Science, The Ohio State University, under Air Force Contract No. F19628-79-C-0027, The Ohio State University Research Foundation Project No. 711664, Project Supervisor Richard H. Rapp. The contract covering this research is administered by the Air Force Geophysics Laboratory, Hanscom Air Force Base, Massachusetts, with Mr. Bela Szabo, Contract Monitor.

Foreword

This report was prepared by Dr. Oscar L. Colombo, Post Doctoral Researcher, Department of Geodetic Science, The Ohio State University, under Air Force Contract No. F19628-79-C-0027, The Ohio State University Research Foundation Project No. 711664, Project Supervisor Richard H. Rapp. The contract covering this research is administered by the Air Force Geophysics Laboratory, Hanscom Air Force Base, Massachusetts, with Mr. Bela Szabo, Contract Monitor.

The reproduction and distribution of this report was carried out through funds supplied by the Department of Geodetic Science. This report was also distributed by the Air Force Geophysics Laboratory as document AFGL-TR-80-0294, Scientific Report No. 7 under Contract No. F19628-79-C-0027.

Acknowledgements

I am indebted to Dr. R. H. Rapp for his support during the long months when this report was gestating; to K. Katsambalos and C. Jekeli for comments, particularly on Section 1; to Pamela Hager for typing the beginning of section 1, and section 3; and to Susan L. Carroll for typing all the rest, a hard task that she has done with care and good cheer.

The ideas presented in the pages that follow began to develop while I was working for the late Dr. R. S. Mather at the University of New South Wales, Australia. They continued to grow, out of discussions with him, his students, and many others, first in Australia and then in the USA. Those discussions made clear to me that there was (and still is, of course) real need to find efficient ways for handling the large amounts of data that are becoming available to Earth scientists on a global basis, chiefly through the use of space technology. To all of them, my sincere thanks for their beneficial influence.

Table of Contents

Foreword iii

Acknowledgements iv

1. Introduction 1

1.1 Spherical Harmonic Analysis and Synthesis: Definitions 2

1.2 Relationships between Spherical Harmonics and 2-D Fourier Series 6

1.3 Sampling Errors 9

(a) Number of equations and number of unknowns 11

(b) 100% aliasing 12

(c) Orthogonality 12

1.4 Number of Operations in Analysis and in Synthesis 13

1.5 Algorithm for the Analysis of Point Values 15

1.6 Algorithm for the Synthesis of Point Values 17

1.7 Algorithms for the Analysis and Synthesis of Area Means 18

1.8 Duality between Analysis and Synthesis 18

1.9 Usefulness of the Fast Fourier Transform Method 19

1.10 Functions Harmonic in Space and their Gradients 21

2. Error Measure and Optimal Quadrature Formulas 23

2.1 The Isotropic Covariance 23

2.2 Some Additional Notation 26

2.3 Estimation Errors, Sampling Errors, and Propagated Noise . 29

2.4 The Quadratic Error Measure. 30

(a) Propagated noise measure 30

(b) Sampling Error Measure 31

(c) Total Error Measure 31

2.5 The Meaning of the Error Measure 32

2.6 Simple Formulas for Area Means 33

2.7 Optimum de-Smoothing Factors 35

2.8 Least Squares Collocation 36

2.9 The Best Quadrature Formula for non-Uniform, Uncorrelated Noise 38

2.10 The Structure of the Covariance Matrix and its Consequences . 40

2.11 Setting up and Inverting the Covariance Matrix 43

2.12 Optimal Formulas for non-Uniform, Correlated Noise 46

2.13 Least Squares Adjustment and Least Squares Collocation 47

(a) Band-limited signal 47

(b) Infinite bandwith 50

2.14 Ridge Regression and Least Squares Collocation 51

2.15 Structure of the Normal Matrix 53

2.16 Global Adjustment and Collocation with Scattered Data 56

(a) Full region bound by lines of latitude and longitude 56

(b) Arbitrarily scattered data 57

2.17 The Error Matrix in the Band Limited Case 63

2.18 The Use of a Priori Information of the Coefficients 63

(a) Simple weighted averages 64

(b) A priori values included as data 65

(c) The method of Kaula and Rapp 65

2.19 Optimal Estimation over a Band of Spatial Frequencies .. 66

3. Numerical Examples 68

3.1 Generation and Analysis of Simulated Data 68

3.2 Agreement between the Actual and the Theoretical Measures of the Sampling Errors 71

3.3 Accuracies of Various Quadratures Formulas 71

3.4 The Analysis of a Global Data Set of $5^\circ \times 5^\circ$ Mean Anomalies . 80

4. Covariances between Area Means 82

4.1 Derivation of an Approximate Formula for the Covariance .. 82

4.2 Choosing N_{max} 84

4.3 Numerical Examples 85

4.4 Covariances between Mean Values and Point Values 90

5. Conclusions 92

References 94

Appendix A: Term by Term Integration of Formula (4.5) (Proof of) .. 97

Appendix B: Computer Programs (Descriptions and Listings) 100

B.1 General Programming Considerations 100

B.2 Subroutine SSYNTH 101

B.3 Subroutine HARMIN 106

B.4 Subroutine NORMAL 112

B.5 Subroutine NORMAX 124

B.6 Subroutine LEGFDN 131

B.7 Subroutine NVAR 134

B.8 Subroutine COVBLK 138

1. Introduction

Spherical harmonics are closely associated with the basic theory of gravitational and magnetic fields, such as those of the Earth and planets; for this reason they are important both in geodesy and in Earth and planetary physics.

The present work considers the numerical aspects of the reduction of global data sets to spherical harmonic coefficients, so the emphasis has been laid on the algorithms for this purpose. The procedures for harmonic analysis (and synthesis) given here are general enough to be used in the study of magnetic or electric fields, but most conclusions regarding their accuracy are restricted to the gravity field of our planet and to fields with the same power spectrum. The accuracy of these methods cannot be separated from the type of signal being used.

Modern instrumentation has provided scientists and engineers with vast amounts of information, and modern computers have made the processing of it possible, and even routine, thanks to constant improvements in both hardware and software. In the mid sixties, those branches of applied mathematics, physics, and engineering concerned with the sifting of data, or with the study of very large regular structures, were greatly affected by the advent of the Fast Fourier Transform (FFT). In spite of the fact that spherical harmonics are members of the family of Fourier transforms, closely related to two dimensional Fourier series, geodesy has lagged behind in the development of techniques similar to the FFT, partly because of the rather wicked nature of the sphere on which data are usually given, partly because there have not been enough data to make the development of powerful techniques a general concern. The topological differences between the Euclidean plane and the surface of the sphere may very well prevent the finding of algorithms as efficient as the FFT for the latter (certainly none seems to have been reported to date) but such algorithms should be regarded, nonetheless, as the desideratum for all those who wish to put their time and work into developing good numerical methods for spherical harmonic analysis.

The increasing use of artificial satellites for surveying the gravity field, particularly by radar altimetry and by the projected tracking of one satellite by another, are making the use of very efficient and accurate techniques for handling the resulting data imprescindible: even a casual review of the literature of the last few years will show that serious efforts to provide such techniques are getting under way. The days of scarce, scattered, unreliable data are just about over.

The remainder of this section defines the basic problem and the associated notation, shows the relationships between spherical harmonics and 2D-Fourier series, presents some of the similitudes and differences between both, and explains some efficient algorithms for harmonic analysis and synthesis that are common to a number of different problems. Section 2 begins by defining a quadratic measure for the accuracy of the estimated harmonic coefficients based on the covariance functions of the signal and the noise; a discussion on the optimization of this measure follows, leading to the application of least squares collocation to harmonic analysis. The use and implementation of least squares adjustment follows, and then a discussion of the connection between least squares and least squares collocation, shown as alternative and efficient techniques for solving the same problem. The section closes with an algorithm for the case when data are irregularly distributed. Section 3 illustrates with several numerical examples some of the methods presented earlier. Section 4 introduces an efficient formula for computing the covariances between block averages when analyzing mean values by collocation. This formula is much more efficient than others based on the numerical quadratures of the "point" covariance function.

1.1. Spherical Harmonic Analysis and Synthesis: Definitions

A square integrable, analytical function $f(\theta,\lambda)$ defined on the unit sphere $0 \leq \theta \leq \pi$ and $0 \leq \lambda \leq 2\pi$ can be expanded in a series of surface spherical harmonics

$$f(\theta,\lambda) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\,[\,\overline{C}_{nm}\cos m\lambda + \overline{S}_{nm}\sin m\lambda\,] \tag{1.1}$$

where: $\overline{P}_{nm}$ are the associated Legendre functions of the first kind, fully normalized so $\frac{1}{4\pi}\int_\sigma \overline{P}_{nm}(\cos\theta)^2 \begin{pmatrix}\cos\\ \sin\end{pmatrix}^2 m\lambda \, d\sigma = 1$ (here $\int_\sigma d\sigma$ indicates integration on the unit sphere);

$\overline{C}_{nm}, \overline{S}_{nm}$ are the fully normalized spherical harmonic coefficients.

For the sake of brevity, the following alternative notation shall be used when possible:

$$\overline{Y}_{nm}^{\alpha}(\theta,\lambda) = \begin{cases} \overline{P}_{nm}(\cos\theta)\cos m\lambda & \text{if } \alpha = 0 \\ \overline{P}_{nm}(\cos\theta)\sin m\lambda & \text{if } \alpha = 1 \end{cases}$$

and

$$\overline{C}_{nm}^{\alpha} = \begin{cases} \overline{C}_{nm} & \text{if } \alpha = 0 \\ \overline{S}_{nm} & \text{if } \alpha = 1 \end{cases}$$

The purpose of spherical harmonic analysis is to estimate the coefficients $\overline{C}_{nm}^{\alpha}$ from measurements of the signal $f(\theta,\lambda)$. These measurements, which may be corrupted by some noise or error signal "n", and which are assumed in what follows to be finite in number, constitute the data. The individual samples are called z_{ij}, so $z_{ij} = f(\theta_i, \lambda_j) + n_{ij}$. The subscripts i and j are used only to designate the position of the sample in a two-dimensional array, or grid, covering in some more or less regular way the sphere: i corresponds always to (co)latitude, and j to longitude. While, as in the last paragraph of section 3, some places in the grid may be empty, the grid itself is defined by a set of complete parallels and meridians. Unless otherwise specified, the separation between the lines of latitude can be variable, but that between meridians is always constant and equal to $\Delta\lambda = \pi/N$, where N is an integer. The grid most often considered in this work is the equal angular grid, also called the regular grid; in this case $\Delta\theta$ is also constant and equal to $\Delta\lambda$.

For equal angular grids i and j take values $0 \le i \le N-1$ and $0 \le j \le 2N-1$, i increasing from North to South, and j from West to East. Formulas where the i,j subscripts appear are in the form appropriate to the regular grid, though most of them can be extended in a very simple way to other partitions.

Data may consist of values determined at the intersections of the grid, in which case they are referred to as "point data", or they may be averages over the blocks defined by the lines of the grid, and then they are called "area means" or "block means". In the equal angular grid the northernmost and southernmost blocks reach to the respective poles; there are N rows of blocks (i.e. blocks between the same parallels), there are $2N$ blocks per row, and i,j identify blocks according to the row and column they are in. Grids for equal angular point data may be "center point" grids, data being measured at the center of each block, so $i = 0$ corresponds to $\theta = \Delta\theta/2 = \Delta\lambda/2$.

No blocks stride the equator, so regular grids are symmetrical with respect to it; in other words: N is always even (extension to N odd is trivial).

Area means are identified here by overbars; they can be expanded in series simply by integrating (1.1) term by term, which can be done because the spherical harmonic series is always uniformly convergent for $0 \le \theta \le \pi$ and $0 \le \lambda \le 2\pi$:

$$\bar{f}_{ij} = \frac{1}{\Delta_{ij}} \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha} \int_{\sigma_{ij}} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)\, d\sigma$$

$$= \frac{1}{\Delta_{ij}} \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \int_{\theta_i}^{\theta_i+\Delta\theta} \bar{P}_{nm}(\cos\theta)\sin\theta\, d\theta \left[\bar{C}_{nm} \int_{\lambda_j}^{\lambda_j+\Delta\lambda} \cos m\lambda \right.$$

$$\left. + \bar{S}_{nm} \int_{\lambda_j}^{\lambda_j+\Delta\lambda} \sin m\lambda\, d\lambda \right] \tag{1.2}$$

Here $\bar{f}_{ij}$ is the area mean of $f(\theta,\lambda)$ on the block σ_{ij} whose area is $\Delta_{ij} = \Delta\lambda(\cos\theta_i - \cos(\theta_i + \Delta\theta))$.

A basic property of spherical harmonics is their orthogonality:

$$\frac{1}{4\pi} \int_{\sigma} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)\, \bar{Y}_{kp}^{\alpha}(\theta,\lambda)\, d\sigma = \begin{cases} 1 & \text{if } \alpha = \beta,\ m = p \text{ and } n = k \\ 0 & \text{otherwise} \end{cases} \tag{1.3}$$

as a consequence of which

$$\bar{C}_{nm}^{\alpha} = \frac{1}{4\pi} \int_{\sigma} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)\, f(\theta,\lambda)\, d\sigma \tag{1.4}$$

Expression (1.4) is the inverse of (1.1); both constitute the basis of spherical harmonic analysis. In general, (1.4) cannot be calculated analytically, because what is available is not the function $f(\theta,\lambda)$, but a finite set of noisy measurements in the form of point values z_{ij} or their area means $\overline{z}_{ij}$. Discretizing (1.4) on an equal angular grid results, for instance, in the following numerical quadratures formula:

$$\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{Y}^{\alpha}_{nm}(\theta_i,\lambda_j)\, f(\theta_i,\lambda_j)\, \Delta_{ij} \tag{1.5}$$

where $\hat{C}^{\alpha}_{nm}$ indicates the estimate of $\overline{C}^{\alpha}_{nm}$, as this type of formula is usually only an approximation. Formulas resembling (1.5) can be called "point values-type quadrature formulas" and can be handled by algorithms that are all identical from a structural point of view and whose prototype is that of paragraph 1.5.

If the data are block averages, several simple approximations have been proposed that take the form

$$\hat{C}^{\alpha}_{nm} = \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{f}_{ij} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma$$

where the μ_n are scale factors. This kind of formula shall be studied further in section 2. Writing the above expression in full

$$\hat{C}^{\alpha}_{nm} = \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{f}_{ij} \int_{\theta_i}^{\theta_i+\Delta\theta} \overline{P}_{nm}(\cos\theta)\sin\theta\, d\theta \int_{\lambda_j}^{\lambda_j+\Delta\lambda} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m\lambda\, d\lambda \tag{1.6}$$

This belongs to the general type

$$\hat{C}^{\alpha}_{nm} = K \sum_{i=0}^{N-1} \chi^{nm}_{i} \sum_{j=0}^{2N-1} \overline{f}_{ij} \left[\begin{Bmatrix} A(m)\\ -B(m)\end{Bmatrix} \cos mj\Delta\lambda + \begin{Bmatrix} B(m)\\ A(m)\end{Bmatrix} \sin mj\Delta\lambda \right] \tag{1.7}$$

where $B(m) = \begin{cases}(\cos m\,\Delta\lambda - 1)/m & \text{if } m \neq 0\\ 0 & \text{if } m = 0\end{cases}$; $A(m) = \begin{cases}(\sin m\,\Delta\lambda/m & \text{if } m \neq 0\\ \Delta\lambda & \text{if } m = 0\end{cases}$

Formulas resembling (1.6) or (1.7) appear several times in this work, and are called here area means-type quadrature formulas.

If all $\overline{C}^{\alpha}_{nm}$ with $0 \leq n \leq N_{max}$ are known, they can be used to compute the $(N_{max} + 1)^2$ terms in

$$f(\theta,\lambda)_{N_{max}} = \sum_{n=0}^{N_{max}} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}^{\alpha}_{nm}\, \overline{Y}^{\alpha}_{nm}(\theta,\lambda) \tag{1.8}$$

which can be regarded as an approximation to $f(\theta,\lambda)$ at the point (θ,λ). Expression (1.8), together with the truncation at N_{max} of (1.2) (area means), defines the object of spherical harmonic synthesis: given the coefficients, estimate the function. As shown later, analysis and synthesis are related by a simple duality, so they require about the same number of operations when performed on a certain grid and with a given number of coefficients.

The set of all degree variances

$$\sigma_n^2 = \sum_{m=0}^{n} \overline{C}_{nm}^{\,2} + \overline{S}_{nm}^{\,2} \tag{1.9}$$

constitutes the power spectrum of $f(\theta,\lambda)$. If all coefficients of degree $n > N_{max}$ are zero, so are the corresponding σ_n ; in such case the function $f(\theta,\lambda)$ is said to be band limited.

Closely associated with the power spectrum is the isotropic covariance function

$$\mathrm{cov}(f(P), f(Q)) = \sum_{n=0}^{\infty} \sigma_n^2 \, P_n(\cos\psi_{PQ}) \tag{1.10}$$

Here P and Q are two points on the sphere, $\psi_{PQ} = \cos^{-1}[\cos\theta_P \cos\theta_Q + \sin\theta_P \times \sin\theta_Q \cos(\lambda_P - \lambda_Q)]$ is the geocentric angle or spherical distance between both, and $P_n(\cos\psi)$ is the unnormalized Legendre polynomial of degree n. The power spectrum and the covariance of a function on the sphere, like those of functions defined on the real line, on the plane, and on Euclidean spaces of higher dimensions, can be used to obtain estimators related to discrete Wiener filters and predictors, i.e., satisfying a minimum variance condition. The spherical version of such technique is known as least squares collocation (Moritz, 1972). The inverse of expression (1.10) is

$$\sigma_n^2 = (2n+1) \int_0^{\pi} P_n(\cos\psi_{PQ}) \, \mathrm{cov}(f(P), f(Q)) \sin\psi_{PQ} \, d\psi_{PQ} \tag{1.11}$$

Equations (1.10) and (1.11) show that, as usual, power spectrum and covariance function are linear transforms of each other. A formula such as (1.11) is sometimes called a Legendre transform.

In addition to the autocovariance (1.10), one can define, more generally,

$$\mathrm{cov}(u(P), v(Q)) = \sum_{n=0}^{\infty} c_n^{u,v} \, P_n(\cos\psi_{PQ}) \tag{1.10*}$$

as the covariance of two functions $u(\theta_P, \lambda_P)$ and $v(\theta_Q, \lambda_Q)$ on the sphere, where the

$$c_n^{u,v} = \sum_{m=0}^{n} \overline{C}_{nm}^{\,u} \, \overline{C}_{nm}^{\,v} + \overline{S}_{nm}^{\,u} \, \overline{S}_{nm}^{\,v} , \qquad n = 0, 1, \ldots \tag{1.9*}$$

constitute the "power crosspectrum".

A relationship similar to (1.11) applies to the $c_n^{u,v}$ and to $\mathrm{cov}(u(P), v(Q))$.

A very important property of spherical harmonics is Parseval's theorem:

$$\frac{1}{4\pi} \int_\sigma f(\theta,\lambda)^2 \, d\sigma = \sum_{n=0}^{\infty} \sigma_n^2 \tag{1.12}$$

The orthogonality of spherical harmonics, and the fact that they form a complete orthogonal set of functions on the sphere, are among the reasons why they are used so widely in theory and in practice, but there is more to them than orthogonality.

The solid spherical harmonics

$$\frac{1}{r^{n+1}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda) \qquad \text{and} \qquad r^{n}\, \overline{Y}^{\alpha}_{nm}(\theta,\lambda)$$

where r is the distance to the origin of coordinates, are all solutions of Laplace's equation, which in Cartesian coordinates is

$$\nabla^2 W = \frac{\partial^2 W}{\partial x^2} + \frac{\partial^2 W}{\partial y^2} + \frac{\partial^2 W}{\partial z^2} = 0 \tag{1.13}$$

This makes them appropriate for the study of harmonic functions, such as the gravitational potential, in spherical coordinates. Another property they have, unique among functions that are orthogonal on the sphere, is the relationship

$$\frac{1}{4\pi}\int_{\sigma} f(P) \sum_{n=0}^{\infty} (2n+1)\, k_n\, P_n(\psi_{PQ})\, d\sigma = \sum_{n=0}^{\infty}\sum_{m=0}^{n}\sum_{\alpha=0}^{1} k_n\, \overline{C}^{\alpha}_{nm}\, \overline{Y}^{\alpha}_{nm}(Q) \tag{1.14}$$

which corresponds to the Convolution Theorem for ordinary Fourier series, and is the basis of such fundamental formulas as Stokes' in gravimetric geodesy.

1.2. Relationship Between Spherical Harmonics and 2-D Fourier Series

As indicated in the previous paragraph, spherical harmonics share important properties with ordinary (trigonometric) Fourier series in one or more dimensions. There is also a very immediate relationship with ordinary two-dimensional (2-D) series that will be explained here. From Hobson (1931, Ch. III, formula (7)) we know that

$$P_{nm}(\cos\theta) = \frac{(-1)^n\,(2n)!}{2^n\, n!\,(n-m)!}\sin^m\theta \left\{\cos^{n-m}\theta - \frac{(n-m)(n-m-1)}{2(2n-1)}\cos^{n-m-2}\theta + \right.$$

$$\left. + \frac{(n-m)\ldots(n-m-3)}{2\cdot 4\cdot(2n-1)(2n-3)}\cos^{n-m-4}\theta - \ldots\right\} \tag{1.15}$$

so the normalized Legendre function is

$$\overline{P}_{nm}(\cos\theta) = \frac{(-1)\;(2n!)}{2^n\, n!\,(n-m)!}\sqrt{\frac{2(2n+1)(n-m)!}{(m+n)!}}\,\sin^m\theta \sum_{k=0}^{L(n,m)} a_k(n,m)\cos^{n-m-2k}\theta$$

where

$$L(n,m) = \begin{cases} (n-m)/2 & \text{if } n-m \text{ is } \underline{\text{even}} \\ (n-m-1)/2 & \text{if } n-m \text{ is } \underline{\text{odd}} \end{cases}$$

while $a_k(n,m) = (-1)^k\,(n-m)(n-m-1)\ldots(n-m-2k+1)[2\cdot 4\ldots 2k\,(2n-1)\ldots(2n-2k+1)]^{-1}$
for $k > 0$

In the interval $-\pi \le \theta \le \pi$ $\sin^m \theta$ is <u>even</u> when m is even, and <u>odd</u> when m is odd. In the interval $-\pi \le \theta \le \pi$ the sum $\sum_{k=0}^{L(n,m)} a_k(n,m) \cos^{n-m-2k} \theta$ <u>is</u> always <u>even</u> (it is a sum of even powers of $\cos\theta$), so <u>the parity of $P_n(\cos\theta)$ is the same as that of m</u> if $-\pi \le \theta \le \pi$. An <u>even</u> function can be expanded into a sum of <u>cosines</u>, and an <u>odd</u> function into a sum of <u>sines</u>. The highest frequency term will correspond to the highest frequency in the expansion of $\sin^m \theta \cos^{n-m} \theta$, so this term will be of the form $a_n \cos n\theta$ or $b_n \sin n\theta$. Therefore, the Legendre function satisfies one of the following equations in $-\pi \le \theta \le \pi$:

$$\bar{P}_{nm}(\cos\theta) = \sum_{t=0}^{n} C_t^{mn} \cos t\theta \qquad \text{(a) if } m \text{ is } \underline{\text{even}};$$

$$\bar{P}_{nm}(\cos\theta) = \sum_{t=0}^{n} S_t^{mn} \sin t\theta \qquad \text{(b) if } m \text{ is } \underline{\text{odd}}$$

A spherical harmonic $\bar{Y}_n^{\alpha}(\theta,\lambda)$ can have one of four possible forms:

for m <u>even</u>:

$$\bar{Y}_{nm}^{\alpha}(\theta,\lambda) = \begin{cases} \sum_{t=0}^{n} C_t^{mn} \cos t\theta \cos m\lambda & \text{for } \alpha = 0 \\ \sum_{t=0}^{n} C_t^{mn} \cos t\theta \sin m\lambda & \text{for } \alpha = 1 \end{cases} \tag{1.16a}$$

for m <u>odd</u>:

$$\bar{Y}_{nm}^{\alpha}(\theta,\lambda) = \begin{cases} \sum_{t=0}^{n} C_t^{mn} \sin t\theta \cos m\lambda & \text{for } \alpha = 0 \\ \sum_{t=0}^{n} C_t^{mn} \sin t\theta \sin m\lambda & \text{for } \alpha = 1 \end{cases} \tag{1.16b}$$

A sum of spherical harmonics such as (1.8) is equivalent to a sum of terms of the form $C_t^{mn} \sin t\theta \cos m\lambda$, $C_t^{mn} \sin t\theta \sin m\lambda$, $C_t^{mn} \cos t\theta \cos m\lambda$ and $C_t^{mn} \cos t\theta \sin m\lambda$, which are also the basic functions of 2-D Fourier series. The highest m and n in the spherical harmonic expansion

$$f(\theta,\lambda) = \sum_{n=0}^{N_{max}} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)$$

are equal to N_{max}, so the highest degree and order (or spatial frequencies) in the Fourier series are also equal to N_{max}. In conclusion: every surface spherical harmonic expansion where the highest degree is N_{max} <u>is identical</u> to a 2-D Fourier series (where the highest m and n are also N_{max}), in the domain $-\pi \le \theta \le \pi$, $0 \le \lambda \le 2\pi$. The converse is not true, because continuous functions on a sphere, such as the $\bar{Y}_{nm}^{\alpha}$, must satisfy certain conditions at the poles that ordinary functions on the $-\pi \le \theta \le \pi$, $0 \le \lambda \le 2\pi$ domain do not have to. Spherical harmonics correspond to a subclass (linear subspace) of 2-D Fourier series.

For example: (Heiskanen and Moritz, Chapter I, 1967)

$$\bar{P}_{11}(\cos\theta) = \sqrt{3}\,\sin\theta$$

$$\bar{P}_{21}(\cos\theta) = \sqrt{30}\,\sin\theta\,\cos\theta = \tfrac{1}{2}\sqrt{30}\,\sin 2\theta$$

So

$$\bar{Y}^{o}_{11} = \sqrt{3}\,\sin\theta\,\cos\lambda,\quad \bar{Y}^{1}_{11} = \sqrt{3}\,\sin\theta\,\sin\lambda$$

$$\bar{Y}^{o}_{21} = \tfrac{1}{2}\sqrt{30}\,\sin 2\theta\,\cos 2\lambda,\quad \bar{Y}^{1}_{21} = \tfrac{1}{2}\sqrt{30}\,\sin 2\theta\,\sin 2\lambda$$

Calling the 2-D Fourier coefficients "a^{β}_{pm}", where

a^{o}_{pm} correspond to terms of the form $\cos p\theta \cos m\lambda$

a^{1}_{pm} " " " " " " $\cos p\theta \sin m\lambda$

a^{2}_{pm} " " " " " " $\sin p\theta \cos m\lambda$

a^{3}_{pm} " " " " " " $\sin p\theta \sin m\lambda$

these can be related to the respective $\bar{C}^{\alpha}_{nm}$ by expressions of the form

$$\bar{C}^{\alpha}_{pm} = \frac{-}{4}\sqrt{\frac{2(2n+1)\,(n-m)\,!}{(n+m)\,!}}\sum_{p=0}^{N-1} I^{m}_{n,p}\, a^{\beta}_{pm} \qquad n = m,\ m+1,\ m+2,\ \ldots \tag{1.17}$$

where $\beta = \alpha$ if m is even, and $\beta = \alpha + 2$ if m is odd. The $I^{m}_{n,p}$ are defined as

$$I^{m}_{n,p} = \int_{0}^{\pi} \cos p\theta\, P_{nm}(\cos\theta)\,\sin\theta\, d\theta \qquad \text{if } m \text{ is even}$$

$$I^{m}_{n,p} = \int_{0}^{\pi} \sin p\theta\, P_{nm}(\cos\theta)\,\sin\theta\, d\theta \qquad \text{if } m \text{ is odd.}$$

and can be computed recursively using the formula

$$I^{m}_{n,p} = \frac{2n-1}{2(n-m)}\left\{ I^{m}_{n-1,\,p+1} + I^{m}_{n-1,\,p-1} \right\} - \frac{n+m-1}{n-m}\, I^{m}_{n-2,\,p} \tag{1.18a}$$

with the following starting values

$$I^{m}_{m,p} = \begin{cases} 0 \quad \text{if } (m+p) \text{ is odd} \\ \dfrac{2(m+1)\,(2m)\,!}{2^{m}\left[(m+1)^2 - p^2\right]\left[(m-1)^2 - p^2\right]\ldots\left[3^2 - p^2\right]\left[1^2 - p^2\right]} & \text{if } m \text{ is even, } p \text{ even,} \\ \dfrac{2(m+1)\,(2m)\,!}{2^{m}\left[(m+1)^2 - p^2\right]\left[(m-1)^2 - p^2\right]\ldots\left[2^2 - p^2\right]\left[-p\right]} & \text{if } m \text{ is odd, } p \text{ odd.} \end{cases} \tag{1.18b}$$

The $I^{m}_{m,p}$ are zero for alternate values of both p and m. These equations were reported by Ricardi and Burrows (1972), and show how to obtain the

$\bar{C}^{\alpha}_{nm}$ once the a^{β}_{pm} have been computed from the data by means of the 2-D discrete Fourier transform. Normally this would be impossible, because the data only exists in the upper half of the $-\pi \leq \theta \leq \pi$, $0 \leq \lambda \leq 2\pi$ interval, and the 2-D algorithm requires information on all of it. However, the fact that the $\bar{P}_{nm}(\cos\theta)$ are sums of sines only or cosines only makes the calculation possible.

While the maximum degree and order in a 2-D Fourier series do not reach the Nyquist frequency[1] ($m, n < N = \frac{\pi}{\Delta\lambda}$) in an equal angular grid, all the coefficients can be recovered exactly by solving $(2N)^2$ equations such as

$$f(\theta_i, \lambda_i) = \sum_{n=0}^{N} \sum_{m=0}^{N} \sum_{\beta=0}^{3} a^{\beta}_{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} n\,\theta_i \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m\lambda_i \tag{1.19}$$

When n or m exceed N_{max}, the matrix of the system of equations becomes singular, and the discrete Fourier transform consists of coefficients that "fit" the data, but differ from the true coefficients. The estimated coefficients are said to have been aliased with those that exceed the Nyquist frequency. In the case of spherical harmonics, which are a special case of 2-D Fourier series, a similar situation must arise: the harmonics in the data with $n \geq N$ are going to be aliased with those of lower degree, so the information available is not enough to recover all coefficients because the sampling is too coarse.

The aliasing of spherical harmonics sampled on regular grids is a consequence of the aliasing of the respective 2-D Fourier series, so it makes sense to talk of a "Nyquist frequency" in the case of those functions. Having established the connection between aliasing in both types of series, it is time to point out also some important differences.

1.3 Sampling Errors

Expressions (1.16a-b) shows that spherical harmonics are finite sums of 2-D Fourier harmonics, which is not the same as being each a Fourier harmonic. From this simple fact follow some important distinctions.

To understand them better, let us begin by stating some basic properties of Fourier series in one dimension, which carry over to higher dimensions but are easier to explain in one dimension.

If sampled at a constant interval $\Delta\lambda = \frac{\pi}{N}$, the following is always true of sines and of cosines:

---

[1]Named after the Nyquist Theorem: the Fourier coefficients of a function of period $2N\Delta\lambda$ can be recovered only if $N_{max} < N$.

$$\left.\begin{aligned}
&\sum_{k=0}^{2N-1} \cos mk\Delta\lambda \ \cos pk\Delta\lambda = 0 && \text{if} \quad m \neq p < N \\
&\sum_{k=0}^{2N-1} \sin mk\Delta\lambda \ \sin pk\Delta\lambda = 0 && \text{if} \quad m \neq p < N \\
&\sum_{k=0}^{2N-1} \cos mk\Delta\lambda \ \sin pk\Delta\lambda = 0 && \text{for all} \quad m \quad \text{and all} \quad p.
\end{aligned}\right\} \quad (1.20\text{-a})$$

These expressions are discrete counterparts of

$$\left.\begin{aligned}
&\int_0^{2\pi} \cos m\lambda \ \cos p\lambda \, d\lambda = \int_0^{2\pi} \sin m\lambda \ \sin p\lambda \, d\lambda = 0 \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{when} \quad m \neq p \\
&\int_0^{2\pi} \cos m\lambda \ \sin p\lambda \, d\lambda = 0 \qquad\qquad \text{for all} \quad m \quad \text{and} \quad p.
\end{aligned}\right\} \quad (1.20\text{-b})$$

and show that the orthogonal properties of sines and cosines are maintained when these are sampled regularly, provided the Nyquist frequency is not exceeded. From this follows that

$$a_m^{\alpha} = H \sum_{k=0}^{2N-1} \binom{\cos}{\sin} mk\,\Delta\lambda f(k\Delta\lambda) = h\int_0^{2\pi} \binom{\cos}{\sin} m\lambda f(\lambda)\, d\lambda \qquad (1.21)$$

where $H = \begin{cases} \frac{1}{2N} & \text{if} \quad m = 0 \\ \frac{1}{N} & \text{otherwise} \end{cases}$ and $h = \begin{cases} \frac{1}{2\pi} & \text{if} \quad m = 0 \\ \frac{1}{\pi} & \text{otherwise} \end{cases}$, so the "Fourier" counterpart of (1.5), (i.e. (1.21)) is an exact "numerical quadratures" formula for Fourier series. When the Nyquist frequency is exceeded, the trigonometric relationships

$$\cos mk\Delta\lambda = \cos(2N \pm m)\, k\Delta\lambda$$

$$\pm \sin mk\Delta\lambda = \sin(2N \pm m)\, k\Delta\lambda$$

imply that (1.21) will give, not the true coefficients, but the aliased estimates

$$\left.\begin{aligned}
\hat{a}_m^0 &= a_m^0 + \sum_{h=0}^{K} a^0_{2hN+m} + \sum_{h=1}^{K} a^0_{2hN-m} \\
\hat{a}_m^1 &= a^1_m + \sum_{h=0}^{K} a^1_{2hN+m} - \sum_{h=1}^{K} a^1_{2hN-m}
\end{aligned}\right. \qquad (1.22)$$

with K such that 2KN is below, and 2(K+1)N is above the highest frequency present in $f(\lambda)$. Expression (1.20 a-b), (1.21), and (1.22) are the foundations of discrete Fourier analysis (also known as the computation of the discrete Fourier transform, or D.F.T.), and so well known that they are almost second nature to many engineers and scientists. Unfortunately, none of these discrete formulas has exact counterparts in spherical harmonic analysis, and this fact has been the cause of considerable confusion. The most common misunderstandings seem to occur in the following areas:

(a) Number of equations and number of unknowns:
At each point in the grid it is possible to write an equation such as

$$f(\theta_i \lambda_j) = \sum_{n=0}^{N_{max}} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}_{nm}^{\alpha} \; \overline{Y}_{nm}^{\alpha} \; (\theta_i, \lambda_j) \tag{1.23}$$

similar to (1.19). The maximum number of coefficients that can be recovered without exceeding the Nyquist frequency (i.e., those with $n < N$) is N^2. Coefficients of degree or order equal or higher than N shall not be, in general, free from aliasing. Since there are $2N^2$ points in an equiangular grid, it follows that the maximum number of fully recoverable coefficients is also half the number of points (equations) in the grid. By contrast, in 2-D Fourier analysis the number of coefficients equals the number of points in the grid (in the interval $-\pi \leq \theta \leq \pi$, $0 \leq \lambda \leq 2\pi$). One could have a grid with only N^2 points, as proposed by Giacaglia and Lundquist (1972), but such a grid would not be equal angular on the sphere.

The author had discussed this problem elsewhere (Colombo, 1979a, paragraph (5.2)), showing that the system of equations (1.23) becomes singular when all coefficients of degree and order $0 \leq (n, m) \leq M$ are included among the unknowns, and $M \geq N$. In other words: it is not possible to solve for a complete set of coefficients to degree and order $M \geq N$. The relevant part of that argument can be summarized as follows: the columns of A, the matrix of the system of equations (1.23), consist in successions of values $\overline{Y}_{nm}^{\alpha}(\theta_i, \lambda_j)$ of the harmonics corresponding to the unknowns $\overline{C}_{nm}^{\alpha}$ at the points (θ_i, λ_j) in the grid. The scalar product of two such columns is,

$$\sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{Y}_{nm}^{\alpha} \; \overline{Y}_{pq}^{\beta} = \sum_{i=0}^{N} \overline{P}_{nm}(\cos\theta_i) \; \overline{P}_{pq}(\cos\theta_i)$$

$$\times \sum_{j=0}^{2N-1} \left\{ \begin{matrix} \cos \\ \sin \end{matrix} \right\} mj\Delta\lambda \left\{ \begin{matrix} \cos \\ \sin \end{matrix} \right\} qj\Delta\lambda = \begin{cases} 0 \text{ if } \alpha \neq \beta \\ 0 \text{ if } m \neq q \\ \neq 0 \text{ otherwise} \end{cases}$$

according to (1.20-a). Therefore, if two columns correspond to unknowns of different orders m and q, they must be orthogonal and, thus, independent. For the whole matrix not to be singular, all columns of the same order m must form sub-matrices $A(m)$ that have full rank. Otherwise, there will be columns in those $A(m)$, and consequently in A, that are linearly dependent, so A cannot be inverted. Consider $A(0)$, corresponding to all unknowns of order 0. This is a $2N^2 x (M + 1)$ matrix, and the elements of the columns of $A(0)$ have the same values as $\overline{P}_{no}(\cos\theta_i)$, $0 \leq n \leq M$, $0 \leq i \leq N-1$. The $\overline{P}_{no}$ are functions of θ_i only, and there are N parallels in the grid, so there are nc more than N independent rows in $A(0)$. Because the $\overline{P}_{no}(\cos\theta)$ are polynomials of degree n in $\cos\theta$, these rows form a sub-matrix $S(0)$ of $A(0)$ that has $M + 1$ independent columns, as long as $M + 1 \leq N$ or $M < N$. If $M \geq N$ the extra columns will turn S(0) into a rectangular matrix with more columns than rows; in other words: there will be no square submatrix in A(0) of rank $M + 1$ if $M \geq N$, so A(0) shall be rank defficient, and from this follows that A must be singular.

Emphasis must be placed on the word complete when referring to the set of "solvable" coefficients: it is possible, by removing some coefficients with $n < N$ from the unknowns, to introduce others in their stead with $n \geq N$, but then the solution will not be a complete set of coefficients.

(b) 100% aliasing:

If a term in a Fourier series has a frequency $n > N$, then it will be aliased with lower frequency terms and become impossible to discriminate from them. For most functions of practical interest, the higher the frequency, the smaller the term, so the coefficient estimated using the sum in (1.21) will be dominated by the lower frequency terms: the estimation error is thus likely to exeeed 100%. Estimates above the Nyquist frequency are usually regarded as meaningless and the closer a term is to that frequency with increasing n, the less reliance is placed on its estimate.

In the case of spherical harmonics, expressions (1.16a-b) show clearly that the harmonic $\overline{Y}^{\alpha}_{nm}$ consists of several Fourier terms of frequencies ranging from 0 to n. When $n > N$, only that part of the Fourier expansion of $\overline{Y}^{\alpha}_{nm}$ above the Nyquist frequency will become scrambled beyond recovery; part of the harmonic is left intact: the low frequency "tail", which means that the effect of aliasing on the recovered coefficients does not necessarily reach 100% (or even 70%) at the Nyquist frequency, as shown in the examples of section 3.

(c) Orthogonality

From (1.20a) follows that the matrix of equations (1.19) for the Fourier series is orthogonal, so the coefficients estimated according to (1.21) are independent from each other.

In the case of the $\overline{Y}^{\alpha}_{nm}$, orthogonality does not carry over to all the sampled harmonics, unless special "quadratures' weights" are introduced in (1.5) or (1.6). This lack of orthogonality affects, for instance, the formulas for mean values discussed in section 2. The method of Gaussian quadratures is an example of "quadratures with weights" that gives exact coefficients when the Nyquist frequency is not exceeded, though it requires a special grid where the parallels are situated at the same latitudes as the zeroes of $P_{N+1}(\cos\theta)$. The use of this method is possible because the product $[\overline{P}_{nm}(\cos\theta)\,\overline{P}_{pm}(\cos\theta)]$ is a polynomial in $\cos\theta$ of degree $n + p \leq 2N$; the grid, however, is an unusual one. Details of the application of Gaussian quadratures to spherical harmonic analysis are given in a report by Payne

(1971). Other examples of methods that recover the coefficients exactly when the data is noisless and $N_{max} < N$ are: least squares collocation, least squares adjustment, and the algorithm developed by Rice and Burrows from expressions (1.17) and (1.18 a-b).

In general, not all the discretised harmonics retain the orthogonality property, and the estimated coefficients are affected by the values of many of the other $\bar{C}^{\alpha}_{nm}$, in addition to those whose degrees exceed the Nyquist frequency. More about this will be said in section 2, when discussing least squares collocation and adjustment.

Summarising: there are enough differences between the aliasing of Fourier series and that of spherical harmonics, in spite of their being so closely related, to require a great deal of caution before using the intuition gained from one type of analysis when attempting the other. For this reason, the expression "aliasing error" will be replaced by the just as appropriate "sampling error", which is perhaps less charged with misleading connotations because it has not been applied almost exclusively to Fourier series.

It should be noticed here that Gaposchkin (1980) has published formulas for the sampling error on a type of equal area grid (i.e., all blocks have the same area as the equatorial ones). His formulas are the equivalent, for such a grid, of expression (1.22) for Fourier analysis, but much more complicated; they are made tractable numerically by the use of certain recursive expressions that he provides, thus showing an interesting new approach to the study of the problem.

1.4 Number of Operations in Analysis and in Synthesis

Expressions (1.17) and (1.18 a-b) can be used to calculate the $\hat{C}^{\alpha}_{nm}$ once the 2-D Fourier coefficients of $f(\theta_i, \lambda_j)$, or a^{β}_{pm}, have been estimated by means of the 2-D Discrete Fourier Transform (DFT). Using the Fast Fourier Transform (FFT) algorithm (Cooley and Tuckey, 1965), the number of operations required[1] is proportional to the number of data points, which is the order of N^2, or $O(N^2)$ for short. The FFT is discussed further in paragraph 1.9.

Having obtained the a^{β}_{pm}, calculation of (1.17) requires N operations per $\hat{C}^{\alpha}_{nm}$, or $N \times N^2 = N^3$ for all of them. Finding the coefficients

---

[1] Most of these calculations have the form of scalar products
$p = \sum_{k=0}^{N-1} a_k b_k$, so the basic operation of finding $p^k = a_k b_k + p^{k-1}$ $(p^0 = 0,\ p^{N-1} = p)$
consists of one sum and one product.

$I^m_{n,p}$ by means of the recursives (1.18 a-b) adds another $O(N^3)$ operations that can be obviated by computing the $I^m_{n,p}$ once, and then storing them on magnetic tape or disk. One way or the other, $O(N^2) + O(N^3)$ operations are needed altogether, or $O(N^3)$ when N is large (say, $N \geq 180$).

As already mentioned, this prodedure was first described by Ricardi and Burrows in 1972; more recently (1977) Goldstein developed a very similar idea and formulated a similar algorithm for synthesis. Goldstein's method uses recursive formulas for the $I^m_{n,p}$ different from (1.18 a-b). The synthesis algorithm also requires $O(N^3)$ operations.

As explained in paragraphs 1.5 through 1.7, the procedures presented there also require $O(N^3)$ operations for analysis, and as many for synthesis, though they are formally different from Ricardi and Burrows'. In paragraph 1.8 it will be shown that synthesis requires as many operations as analysis, because one is the dual of the other.

The fact that two rather different approaches (Ricardi and Burrows' and the one described in this report) require essentially the same number of operations suggests that "$O(N^3)$" might be a property of all analysis and synthesis algorithms on regular spherical grids due, somehow, to the nature of the sphere itself. This is speculation, of course, but if not, are there other ways of partitioning the sphere for which faster methods exist? The author has discussed this possibility before (Colombo, 1979, paragraph 4.6). It is interesting to notice that in all these procedures the $O(N^3)$ operations are those associated with θ_i, or "column operations"; "row operations" are only $O(N^2)$. In the case of the Euclidean plane, the 2-D Fourier transform requires the same number of operations per row than per column, $O(N)$, thus the total is only $O(N^2)$, or $O(N)$ times faster than its spherical "counterpart."

While not as efficient as the 2-D FFT, the algorithms for the sphere considered here can be much faster than the straightforward implementation of expressions (1.5), (1.6), (1.2), or (1.8). The latter has been the approach of many scientists who have developed their own software, but whose main interest has generally been far removed from the study of numerical techniques. In 1976, while working at the University of New South Wales (Australia), the author developed the two algorithms of paragraphs 1.5 and 1.6, and C. Rizos programmed them. Subsequently they were used at Goddard Space Flight Center, in Maryland. To everybody's surprise, Rizo's programs turned out to be more than 100 times faster than those in use at the time, when run under the same conditions. More recently, this author has written the subroutines HARMIN and SSYNTH described in appendix B. SSYNTH has been used, after the fashion of the numerical experiments described in Section 2, to generate 64000 $1^\circ \times 1^\circ$ mean values (simulated averaged gravity anomalies), each the sum of the 90000 terms of an expansion complete to degree and order 300. This took less than 50 central processor unit seconds in the AMDHAL 470 V/6-II owned by The Ohio State University (OSU). All calculations were in double precision (32 bits words), using FORTRAN H EXTENDED. Precomputed values of the Legendre functions (or their integrals) were read from tape, and all operations involving trigonometric functions were carried out by a Fast Fourier Transform subroutine. These two characteristics, plus a generally tighter coding, are the reasons for the greater speed of this program, compared to the older versions mentioned before.

In all these methods all operations along a given row or parallel (constant θ_i) are independent from those for any other row, so a parallel processing computer with N processors (arithmetic and control units) could analyse or synthesize a full grid of N rows as fast as an ordinary computer with one central processor can do a single row. This N-fold increase in speed can be obtained with the same type of basic hardware (gates, registers) that is currently used in conventional "general purpose" main-frame machines. The full power of the algorithms presented in this work will be realized when computers of parallel structure become more widely available for scientific applications than they are today.

1.5 Algorithm for the Analysis of Point Values

Expression (1.5) written in full becomes

$$\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \bar{P}_{nm}(\cos\theta_i) \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \;\; f(\theta_i, \lambda_j) \, \Delta_{ij}$$

which corresponds to the general type

$$\hat{C}^{\alpha}_{nm} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \chi_i^{nm} \sum_{j=0}^{2N-1} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \;\; f(\theta_i, \lambda_j) \, \Delta_{ij} \tag{1.24}$$

where χ_i^{nm} could be $\frac{1}{4\pi}$ $\bar{P}_{nm}(\cos\theta_i)\,\Delta_{ij}$ as above, or $\bar{P}_{nm}(\cos\theta_i)\,\omega_i$ in the case of quadrature with weights ω_i, etc.

To simplify the discussion, the grid is supposed to be equal angular and $N_{max}=N-1$, This and the algorithms that follow can be easily adapted for the equal area grids current today. Subroutines HARMIN AND SSYNTH (Appendix B) can handle the cases $N_{max}<N-1$ and $N_{max}>N-1$ as well as $N_{max}= N-1$.

The equal angular grid is symmetrical with respect to the Equator, and assuming that (the same as $\bar{P}_{nm}(\cos\theta_i)$ or $\bar{P}_{nm}(\cos\theta_i)\sin\theta_i$) $\chi_i^{nm} = \chi_{N-1-i}^{nm}$ if n-m is even, and $\chi_i^{nm} = -\chi_{N-1-i}$ if n-m is odd, one can write

$$\hat{C}^{\alpha}_{nm} = \sum_{i=0}^{\frac{1}{2}N-1} \left(\chi_i^{nm} \left[\sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \ f(\theta_i, \lambda_j) \right] + (-1)^{n-m} \ \chi_i^{nm} \left[\sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \, f(\theta_{N-1-i}, \lambda_j) \right] \right) \tag{1.25}$$

This formula suggests the following procedure in two nested loops:

<u>START</u>: set $i = -1$, $\hat{C}^{\alpha(o)}_{nm} = 0$ for all $0 \le n, \ m \le N$;

<u>Outer loop</u>:

(a) increment i by 1 unless $i = \frac{1}{2}N - 1$, in which case <u>STOP</u>

<u>Inner loop</u>:

(b) compute all

$$\begin{Bmatrix} a^i_m \\ b^i_m \end{Bmatrix} = \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \ f(\theta_i, \lambda_j)$$

$$\begin{Bmatrix} a^{N-1-i}_m \\ b^{N-1\ i}_m \end{Bmatrix} = \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \ f(\theta_{N-1-i}, \lambda_j)$$

(c) find all

$$\hat{C}^{\alpha(i)}_{nm} = \hat{C}^{\alpha(i-1)}_{nm} + K\left[\begin{Bmatrix} a^i_m \\ b^i_m \end{Bmatrix} + (-1)^{n-m} \begin{Bmatrix} a^{N-1-i}_m \\ b^{N-1-i}_m \end{Bmatrix} \right] \chi_i^{nm} \tag{1.26}$$

for $0 \le n, \ m \le N$ (where "$(-1)^{n-m} \{ \}$" merely indicates that "$\{ \}$" is to be added or substracted according to the parity of (n-m)); <u>GO BACK TO (a)</u>.

At the end of the outer loop, $\hat{C}^{\alpha(\frac{1}{2}N-1)}_{nm} = \hat{C}^{\alpha}_{nm}$.

The a^i_m and b^i_m in (1.26) can be computed by taking the Fourier transform along row i and row $N-1-i$ of the values of $f(\theta, \lambda)$. This involves O(N) operations. There are half as many χ_i^{nm} as $C^{\alpha(i)}_{nm}$, therefore it takes $2(N+1)^2$ products, and just as many sums, to form the $\hat{C}^{\alpha(i)}_{nm}$ for the pair of rows i and $N-1-i$. Consequently, there are O(N) + O(N^2) operations per pair of rows, or O(N^2) + O(N^3) for the grid as a whole. This is the same as with the Ricardi and Burrows' algorithm, quickly approaching O(N^3) as N increases.

Subroutine HARMIN (Appendix B) implements this technique.

1.6 Algorithms for the Synthesis of Point Values

Expression (1.8) is of the general form

$$f(\theta_i, \lambda_j) = \sum_{n=0}^{N-1} \sum_{m=0}^{n} \chi_i^{nm} [\bar{C}_{nm} \cos m\lambda_j + \bar{S}_{nm} \sin m\lambda_j] \tag{1.27}$$

where χ_i^{nm} can be, for instance, $K\,\bar{P}_{nm}(\cos\theta_i)$, etc., with K being a proportionality constant. Rearranging terms and considering the parity of χ_i^{nm} leads to

$$\begin{Bmatrix} f(\theta_i, \lambda_j) \\ f(\theta_{N-1-i}, \lambda_j) \end{Bmatrix} = \sum_{m=0}^{N-1} \left(\left[\sum_{n=m}^{N-1} \begin{Bmatrix} \chi_i^{nm} \\ (-1)^{n-m} \chi_i^{nm} \end{Bmatrix} \bar{C}_{nm} \right] \cos mj\Delta\lambda + \left[\sum_{n=m}^{N-1} \begin{Bmatrix} \chi_i^{nm} \\ (-1)^{n-m} \chi_i^{nm} \end{Bmatrix} \bar{S}_{nm} \right] \sin mj\Delta\lambda \right) \tag{1.28}$$

which suggests a procedure in two nested loops:

START: set $i = -1$

Outer loop:

(a) increment i by 1, unless $i = \frac{1}{2}N - 1$, in which case STOP

Inner loop:

(b) compute all

$$\begin{Bmatrix} \alpha_m^i \\ \beta_m^i \end{Bmatrix} = \sum_{n=m}^{N-1} \chi_i^{nm} \begin{Bmatrix} \bar{C}_{nm} \\ \bar{S}_{nm} \end{Bmatrix}$$

$$\begin{Bmatrix} \alpha_m^{N-1-i} \\ \beta_m^{N-1-i} \end{Bmatrix} = \sum_{n=m}^{N-1} \chi_i^{nm} (-1)^{n-m} \begin{Bmatrix} \bar{C}_{nm} \\ \bar{S}_{nm} \end{Bmatrix}$$

for $0 \le m \le N$;

(c) find all

$$\begin{Bmatrix} f(\theta_i, \lambda_j) \\ f(\theta_{N-1-i}, \lambda_j) \end{Bmatrix} = K \sum_{m=0}^{N-1} \left(\begin{Bmatrix} \alpha_m^i \\ \alpha_m^{N-1-i} \end{Bmatrix} \cos mj\Delta\lambda + \begin{Bmatrix} \beta_m^i \\ \beta_m^{N-1-i} \end{Bmatrix} \sin mj\Delta\lambda \right) \tag{1.29}$$

for $0 \le j \le 2N - 1$ (where $(-1)^{n-m}\{---\}$ means the same as in the previous paragraph); GO BACK TO (a) .

At the end of the outer loop all $f(\theta_i, \lambda_j)$ in the grid are known.

Expression (1.29) is computed by applying the FFT to the 2N α_m^i, β_m^i (per row) and the 2N α_m^{N-1-i}, β_m^{N-1-i}, taking O(N) operations. The first part of the inner loop (forming the α_m^i, β_m^i and α_m^{N-1-i}, β_m^{N-1-i}) involves $O(N^2)$ operations; for all $\frac{1}{2}N$ pairs of rows the total is $O(N^2) + O(N^3)$, and this tends to $O(N^3)$ as N increases.

Subroutine SSYNTH (Appendix B) implements this technique.

1.7 Algorithms for the Analysis and Synthesis of Area Means

(I) Analysis

Rearranging (1.7)

$$\hat{C}^{\alpha}_{nm} = \sum_{i=0}^{\frac{1}{2}N+1} \chi_i^{nm} \left[\begin{Bmatrix} A(m) \\ -B(m) \end{Bmatrix} \left(a_m^i + (-1)^{n-m}\, a_m^{N-1-i}\right) + \begin{Bmatrix} B(m) \\ A(m) \end{Bmatrix} \left(b_m^i + (-1)^{n-m}\, b_m^{N-1-i}\right) \right] \tag{1.30}$$

were a_m^i, b_m^i are the same as in paragraph (1.5). A procedure similar to that for the analysis of point values can be obtained directly by replacing the bracket in (1.26) with that in (1.30), and then proceeding as in the algorithm for point values. The total number of operations is, once more, $O(N^2) + O(N^3)$, or $O(N^3)$ for large N. Subroutine HARMIN also implements this algorithm.

(II) Synthesis

Truncating the series in equation (1.2), replacing $\int_{\theta_i}^{\theta_i+\Delta\theta} \overline{P}_{nm}(\cos\theta)\sin\theta d\theta$ with χ_i^{nm}, rearranging terms,considering the parity of χ_i^{nm}, and using α_m^i, β_m^i as defined in paragraph (1.6), leads to the expression

$$\begin{Bmatrix} \overline{f}_{ij} \\ \overline{f}_{N-1-i\,j} \end{Bmatrix} = \sum_{m=0}^{N-1} \left[\begin{Bmatrix} \alpha_m^i \\ \alpha_m^{N-1-i} \end{Bmatrix} A(m) - \begin{Bmatrix} \beta_m^i \\ \beta_m^{N-1-i} \end{Bmatrix} B(m) \right] \cos mj\Delta\lambda + \left[\begin{Bmatrix} \alpha_m^i \\ \alpha_m^{N-1-i} \end{Bmatrix} B(m) + \begin{Bmatrix} \beta_m^i \\ \beta_m^{N-1-i} \end{Bmatrix} A(m) \right] \sin mj\Delta\lambda \tag{1.31}$$

The algorithm for the synthesis of the $\overline{f}_{ij}$ is a direct extension of that for point values. The number of operations, once more, is $O(N^2) + O(N^3)$ for large N . Subroutine SSYNTH implements this algorithm as well.

1.8 Duality between Analysis and Synthesis

Pairs of direct and inverse linear transforms, such as Fourier transforms, possess dual characteristics: certain words and mathematical expressions can be arranged in pairs (a, b) such that, if every "a" is replaced by its "b" in any statement or equation valid for a function f , then the modified statement is valid for the transform F of f , and viceversa.

Analysis and synthesis of spherical harmonics are reciprocal linear transformations of data into coefficients and of coefficients into "data" closely akin to Fourier transforms, so they can be expected to exhibit some dual properties. Comparing the formulas and algorithms in paragraphs (1.5), (1.6) and (1.7) shows many similarities, among them the number of operations. This can be understood as being a consequency of duality. To make this point clear, consider the following pairs:

$$\left(\begin{Bmatrix} f(\theta_i , \lambda_j) \\ f(\theta_{N-1-i}\ \lambda_j) \end{Bmatrix}, \begin{Bmatrix} \overline{C}_{nm} \\ \overline{S}_{nm} \end{Bmatrix}\right); \left(\begin{Bmatrix} \hat{C}_{nm} \\ \hat{S}_{nm} \end{Bmatrix} , \begin{Bmatrix} f(\theta_i, \lambda_j) \\ f(\theta_{N-1-i}, \lambda_j) \end{Bmatrix}\right) \text{(and similarly with } \begin{Bmatrix} \overline{f}_{ij} \\ \overline{f}_{N-1-i} \end{Bmatrix});$$

$$\left(\chi_i^{nm} (-1)^{n-m} , \ \sin mj\Delta\lambda\right) ; \ (i, m) ; \ (j, n); \ (\chi_i^{nm} , \ \cos mj\Delta\lambda)$$

$$\left(\sum_{i=0}^{\frac{1}{2}N-1} , \sum_{m=0}^{N-1}\right); \left(\sum_{j=0}^{2N-1} , \sum_{\alpha=0}^{1} \sum_{m=n}^{N-1}\right)$$

From these we can derive the following pairs:

$$\left(\begin{Bmatrix} a_m^i \\ a_m^{N-1-i} \end{Bmatrix} , \begin{Bmatrix} \alpha_m^i \\ \alpha_m^{N-1-i} \end{Bmatrix}\right); \left(\begin{Bmatrix} b_m^i \\ b_m^{N-1-i} \end{Bmatrix} , \begin{Bmatrix} \beta_m^i \\ \beta_m^{N-1-i} \end{Bmatrix}\right); \left(\begin{Bmatrix} A(m) \\ -B(m) \end{Bmatrix} , \begin{Bmatrix} \alpha_m^i \\ \alpha_m^{N-1-i} \end{Bmatrix}\right); \left(\begin{Bmatrix} B(m) \\ A(m) \end{Bmatrix} , \begin{Bmatrix} \beta_m^i \\ \beta_m^{N-I-i} \end{Bmatrix}\right)$$

$$\left(a_m^i , \ A(m)\right); \ \left(b_m^i , \ B(m)\right) ; \ \left(a_m^{N-1-i} , -B(m)\right) ; \left(b_m^{N-1-i} , \ A(m)\right); \text{ etc.},$$

and, in conclusion:

("ANALYSIS", "SYNTHESIS")

Each one of the analysis algorithms becomes its synthesis counterpart by a simple replacement of terms. Once an algorithm for analysis (synthesis) is defined, the corresponding algorithm for synthesis (analysis) follows. For instance, one can easily apply the principle of duality to the Ricardi and Burrows' method of paragraph (1.4) to obtain a synthesis technique.

1.9 Usefulness of the Fast Fourier Transform Method

The excellent book by Brigham (1974) gives a thorough presentation of the 1-D discrete Fourier transform and its applications, and explains in detail the method known as FFT for computing such transform. The Fourier transform in 2 and higher dimensions can be found simply as follows:

first, get the 1-D transform of each row, then that of each column of the modified array . . . , etc., until all dimensions have been exhausted in this way. Understanding the workings of the 1-D transform is enough to understand those of the N-dimensional transform as well. The FFT requires O(number of points) operations for each row along the nth dimension, so the number for all points in a regular, euclidean array is always of the order of the total number of points in that array.

Before the mid-sixties _ when the FFT came along _ the best techniques available for the analysis of data on regular arrays required O(number of points)^2 operations. The increase in speed of O(number of points) brought about a true revolution in data processing: work that had been long regarded as impossible became feasible overnight, the field of industrial and scientific applications for numerical Fourier transforms expanded tremendously; the impact in areas as diverse as cristalography and communications engineering was remarkable.

Having mentioned the positive side of the FFT, which is used in the algorithms described so far (at least in principle) and in the programs HARMIN and SSYNTH, it is only proper to say something about its alternatives. The FFT calculates <u>all</u> 2N Fourier coefficients a_m, b_m very efficiently, but takes just about as many operations to get only <u>a few</u> coefficients as it takes to get all: for N_{max} small compared to N there may be a real disadvantage in using the FFT. The FFT is most efficient when the grid is such that <u>N is an integer power of 2.</u> The grids used in geodesy are usually based on the division of the circle in 360°, and many on the sexagesimal division of 1° as well. In all of these N contains factors other than 2, so a less efficient version of the FFT, known as the <u>mix-radix FFT</u> (Singleton's algorithm) must be used.

Finally, the mix-radix algorithm is rather convoluted, so it is best to take ready available subroutines from software libraries (as it is done in HARMIN and SSYNTH) rather than to incorporate the FFT "on line" in the program one is writing. This means that the program is going to be less self-contained.

The "pre-FFT" methods can be more efficient than the FFT when $N_{max} \ll N$; they are also very easy to program. For the sake of completeness, the outline of a method this author has used quite often will be given here.

Consider the trigonometric relationships

$$\cos(\alpha\beta) = 2\cos\beta\,\cos(\alpha-1)\,\beta - \cos(\alpha-2)\,\beta \tag{1.32}$$

$$\sin(\alpha\beta) = 2\cos\beta\,\sin(\alpha-1)\,\beta - \sin(\alpha-2)\,\beta \tag{1.33}$$

with α, β real. If all the values of the trigonometric functions could be obtained with one operation each[1] the number of operations involved in finding $\begin{Bmatrix} a_m \\ b_m \end{Bmatrix} = \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \, (\lambda_j) \, f(\theta_i, \lambda_j)$ would be 2(2N), or $4N^2$ for all $0 \le m \le N$. In other words: $O(N^2)$, as expected. This is precisely what can be done using (1.32), (1.33) as recursive expressions for $\cos m(j\Delta\lambda)$ and $\sin m(j\Delta\lambda)$ with m integer and $0 \le j \le 2N-1$. The values of $\cos m(-\Delta\lambda) = \cos m\Delta\lambda$ and $\sin m(-\Delta\lambda) = -\sin m\Delta\lambda$, are needed to start the recursion; they can be calculated with standard trigonometric subroutines. The use of such subroutines increases the number of operations slightly over $4N^2$, but if N is large enough, this is negligible.

With all calculations carried out in double precision, this method gives values of cosine and sine that coincide to better than 5 significant figures with those provided by the standard FORTRAN functions, when N is as large as 1800 ($0.1° \times 0.1°$ grid). By taking advantage of half-wave symmetries in the sine and cosine, and by ingenuous programming, the number of operations can be reduced by a factor of 4 or more rather easily.

1.10 Functions Harmonic in Space and their Gradients

If $f(\theta, \lambda, r)$ satisfies Laplace's equation $\nabla^2 f = 0$ in the space outside a sphere of radius a, then it can be represented, in that space, by the solid spherical harmonic expansion

$$f(\theta,\lambda,r) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \frac{a^n}{r^{n+1}} \, \overline{C}_{nm}^{\alpha} \, \overline{Y}_{nm}^{\alpha} \, (\theta, \lambda) \tag{1.34}$$

If we consider, at a point $P \equiv (\theta, \lambda, r)$ in space, the local triad $\vec{r}, \vec{h}, \vec{t}$ oriented downwards to the origin, West to East along the tangent to the local parallel, and North-South along the local meridian, the components of the gradient of $f(\theta, \lambda, r)$ along this three axes are

$$\frac{\partial f}{\partial r} (\theta, \lambda, r) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \frac{a^n}{r^{n+2}} (n+1) \, \overline{C}_{nm}^{\alpha} \, \overline{Y}_{nm}^{\alpha} \, (\theta, \lambda) \tag{1.35}$$

$$\frac{\partial f}{\partial h} (\theta, \lambda, r) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \frac{a^n}{r^{n+2}} \overline{P}_{nm} (\cos\theta) \, \mathrm{cosec}\,(\theta) \, [m\overline{S}_{nm} \cos m\lambda - m\,\overline{C}_{nm} \sin m\lambda] \tag{1.36}$$

$$\frac{\partial f}{\partial t} (\theta, \lambda, r) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \frac{a^n}{r^{n+2}} \frac{d\overline{P}_{nm}}{d\theta} (\cos\theta) \, [\overline{C}_{nm} \cos m\lambda + \overline{S}_{nm} \sin m\lambda] \tag{1.37}$$

---

[1]Here one "operation", as mentioned in paragraph (1.4), consists of one sum and one product.

Expressions (1.34), (1.35), (1.36), and (1.37) appear often in the discussion of geodetic problems, and their calculation from a set of coefficients is an important problem. If all the values of any of them(with r constant,and on a regular grid) are required, the methods discussed so far can be used after a few minor additions.

The synthesis algorithms can be thought of, for the purpose of this discussion, as "black boxes" with the coefficients $\overline{C}^{\alpha}_{nm}$, the Legendre functions, N_{max}, and N as inputs, and all the $2N^2$ values of $f(\theta, \lambda)$ on the corresponding regular grid as outputs. To compute the expressions given above, only the part of the input consisting in the coefficients and/or the Legendre functions has to be modified before they enter the "box", which remains untouched. For instance: to compute (1.34) one should replace "$\overline{C}^{\alpha}_{nm}$" with "$\frac{a^n}{r^{n+2}} \overline{C}^{\alpha}_{nm}$" in the "input"; the others are equally obvious and will not be explained further. The following recursive formulas can be used to obtain the derivatives of the Legendre functions:

$$\frac{d\overline{P}_{nm}}{d\theta} = (\sin\theta)^{-1} \left\{ n \; \overline{P}_{nm}(\cos\theta) \; \cos\theta - \left[\frac{(n^2-m^2)(2n+1)}{(2n-1)}\right]^{-\frac{1}{2}} P_{n-1\,m}(\cos\theta)\right\} \qquad (1.38a)$$

$$\frac{d\overline{P}_{nn}}{d\theta} = \left[\frac{(2n+1)}{(2n)}\right]^{\frac{1}{2}} \left\{\sin\theta \frac{d\overline{P}_{n-1\;n-1}}{d\theta} + \cos\theta \; P_{n-1\;n-1}\right\} \qquad (1.38b)$$

with the starting value

$$\frac{d\overline{P}_{oo}}{d\theta} = 0$$

These recursives follow from the unnormalized formula

$$(\cos^2\theta - 1) \; \frac{d P_{nm}(\cos\theta)}{d(\cos\theta)} = n \; \cos\theta \; P_{nm}(\cos\theta) - (n+m) \; P_{n-1\,m}(\cos\theta)$$

(N.N. Lebedev, "Special Functions", Dover, 1972, Ch. 7, equation 7.12.16), and from

$$\overline{P}_{nn} = \left[\frac{(2n+1)}{(2n-1)}\right]^{\frac{1}{2}} \sin\theta \; \overline{P}_{n-1\;n-1} \qquad \text{(see paragraph (4.4))}$$

and

$$\overline{P}_{nn} = \left[\frac{(n+m)!}{2(2n+1)(n-m)!}\right]^{-\frac{1}{2}} P_{nn} \; ; \; \overline{P}_{no} = \sqrt{2n+1} \; P_{no}$$

The complete recursive expressions for the $\overline{P}_{nm}$ are given in paragraph (4.4).

The expansion for the area means defined by (1.2) can be differentiated term by term because it converges uniformly. The expressions for area means gradients, equivalent to those given here for point values, are immediate. They can be computed after simple modifications to the $\overline{C}^{\alpha}_{nm}$ and/or the χ^{nm}_{1}, and using the same programs for computing the area means.

Subroutine "LEGFDN", listed on Appendix B, can compute both the normalized $\overline{P}_{nm}(\cos\theta)$ and their derivatives $\frac{d\overline{P}_{nm}}{d\theta}(\cos\theta)$.

2. Error Measure and Optimal Quadrature Formulas

This section introduces a criterion for quantifying the errors of numerical quadratures formulas that is based on the statistical properties of the data. Three qualities are highly desirable in an error measure: (a) it should be easy to determine; (b) it should be mathematically tractable; (c) it should provide a good idea of the likely size of the actual errors. Point (a) is taken into account by choosing a quadratic measure, because the numerical formulas are linear estimators of the $\overline{C}^{\alpha}_{nm}$, and the linear, quadratic estimation problem is fairly simple, with its mathematical side very well understood and developed today, which takes care also of (b). Regarding (c), the reader will have to wait till section three, where certain evidence, obtained from numerical experiments, supports the assertion that, though statistical in nature, the error measure adopted represents the actual errors very closely.

Having defined the error measure, the notion of optimal or best formula according to such measure is investigated, leading to the application of least squares collocation and least squares adjustment to spherical harmonic analysis.

2.1 The Isotropic Covariance

The isotropic covariance (expression (1.10)) between two functions $u(\theta, \lambda)$, $v(\theta, \lambda)$ on the unit sphere, both expandible in spherical harmonic series

$$u(\theta, \lambda) = K_u \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}^{u\,\alpha}_{nm} \; \overline{Y}^{\alpha}_{nm}(\theta, \lambda) \; ;$$

$$v(\theta, \lambda) = K_v \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}^{v\,\alpha}_{nm} \; \overline{Y}^{\alpha}_{nm}(\theta, \lambda)$$

can be formally defined as follows

$$\operatorname{cov}(u(P), v(Q)) = M\{u(P) \; v(Q)\} \tag{2.1}$$

where $M\{\;\}$ is the isotropic averaging operator and P and Q are two points on the sphere separated by the spherical distance ψ_{PQ}. The operator $M\{\;\}$ symbolizes the average of its argument (in the present case the product $u(P)\,v(Q)$) over all rotations of the sphere. This can be visualized if one thinks of the points P and Q as given in a fixed system of coordintes, while the sphere, on which u and v are defined, rotates in all possible ways. After all the (infinitely many) possible rotations, the average product $u(P)\ v(Q)$ will be identical to $\operatorname{cov}(u(P), v(Q))$. This kind of covariance, though purely geometrical, resembles closely that of stochastic processes such as time series.

The importance of the isotropic operator and the isotropic covariance function in spherical harmonic analysis stems from the fact that the latter can be described as the estimation of certain parameters of a function $f(\theta, \lambda)$, the $\overline{C}^{\alpha}_{nm}$, from data sampled on a sphere. From paragraph (2.7) on, this report deals with optimal estimators for the $\overline{C}^{\alpha}_{nm}$ based on the theory of least squares collocation. Such optimal estimators minimize a quadratic measure of the error that is defined in terms of the operator $M\{\ \}$, this measure being introduced in paragraph (2.4).

The idea of least squares collocation is related to the basic principles of such linear, minimum variance estimators for time series as the Wiener and Kalman filters, which have found wide application in the physical sciences and in engineering over the last thirty years, and have been generalized to deal with both continuous and discrete time processes, and also "processes" in more than one dimension, such as are found in pattern recognition and in digital image enhancement. Two-dimensional Wiener filtering, of which the reader can find several fine descriptions in the special issue of the Proceedings of the IEEE, Vol 65, No. 6, 1977, is also applicable to "flat-Earth" geodetic calculations; least squares collocation can be regarded as the extension of this type of filtering to calculations on the sphere. Isotropic average operators are not the only ones that could be used in the "statistical" approach, though they are probably the easiest to work with and, perhaps, the best for the sort of application considered here. For a description of other likely operators, the reader is referred to the paper by Rummel and Schwarz (1978). Probably the most didatic introduction to the method of collocation remains Heiskanen and Moritz, (Ch. 7, 1967).

Reasoning as in Heiskanen and Moritz (ibid), one can show that

$$\operatorname{cov}(u(P), v(Q)) = \sum_{n=0}^{\infty} c_n^{u,v} \; P_n(\cos\psi_{PQ})$$

which is, in fact, expression (1.10*) the definition of the isotropic covariance given in section 1 without any reference to $M\{\ \}$. Similarly,

$$\operatorname{cov}(u(P), u(Q)) = \sum_{n=0}^{\infty} \sigma_n^2 \; P_n(\cos\psi_{PQ})$$

usually known as "the covariance of u" (expression 1.10), while (1.10*) represents the "covariance between u and v", or "the crosscovariance of u and v". The one-to-one relationship between covariance and power spectrum (or crosscovariance and crosspectrum) should be clear from these expressions.

To apply the notions introduced above to the $\overline{C}^{\alpha}_{nm}$, it is necessary to think of them as functions rather than fixed values. This is possible if one considers changes in the coordinates θ, λ brought about by rotation. Each such change results in different coefficients, though the function they describe is the same, only rotated with respect to the old system. The new system can be related to the first by three angles: the coordinates θ, λ of the shifted north pole, and the azimuth A of the zero meridian. Therefore, the $\overline{C}^{\alpha}_{nm}$ are functions of θ, λ, A and this is enough to define the average over all rotations of the product of each $\overline{C}^{\alpha}_{nm}$ by itself or by another function, in a meaningful way. Two important properties of spherical harmonics are:

(A)
$$M\left\{\sum_{m=0}^{n} \overline{C}_{nm}^{2} + \overline{S}_{nm}^{2}\right\} \equiv M\left\{\sigma_n^2\right\} = \sigma_n^2 \tag{2.2}$$

i.e., the power spectrum is invariant with respect to rotations. This follows from the plain fact that the integral $\int_{\sigma} f^2(\theta, \lambda)\, d\sigma$ is invariant over rotations, and from Parseval's identity (1.12); (2.2) implies that the isotropic covariance function (1.10) is likewise invariant.

(B)
$$M\left\{\overline{Y}^{\alpha}_{nm}(P)\ \overline{Y}^{\beta}_{pq}(Q)\right\} = 0 \ \text{ if } \begin{cases} \alpha \neq \beta \\ n \neq p \\ m \neq q \end{cases} \quad \text{for all P and Q} \tag{2.3}$$

i.e., the orthogonality properties of spherical harmonics with respect to integration on the sphere are also true with respect to averaging over rotations.

As a consequence of (2.2) and (2.3) above, the following relationships are also true:

$$M\left\{\overline{C}_{nm}^{\,2}\right\} = M\left\{\overline{S}_{nm}^{\,2}\right\} = \frac{\sigma_n^2}{2n+1} \tag{2.4-a}$$

$$M\left\{\overline{C}^{\alpha}_{nm}\ \overline{C}^{\beta}_{pq}\right\} = 0 \ \text{ if } \begin{cases} \alpha \neq \beta \\ n \neq p \\ m \neq q \end{cases} \tag{2.4-b}$$

$$M\left\{\overline{C}^{\alpha}_{nm}\right\} = 0 \ \text{ if } \ n \neq 0 \tag{2.5}$$

$$M\left\{\overline{C}^{\alpha}_{nm}\ f(\theta, \lambda)\right\} = \frac{\sigma_n^2}{2n+1}\ \overline{Y}^{\alpha}_{nm}(\theta, \lambda) \tag{2.6}$$

$$M\left\{\overline{C}^{\alpha}_{nm}\ \overline{f}_{ij}\right\} = \frac{\sigma_n^2}{(2n+1)\Delta_{ij}} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta, \lambda)\, d\sigma \tag{2.7}$$

For the derivation of (2.6) the reader can see (Rummel, 1976), and (Sjoberg, 1978) for (2.7). As for (2.4a-b) and (2.5), the proof is given now.

According to (1.4):

$$16\pi^2 \; M\left\{(\bar{C}^{\alpha}_{nm})^2\right\} = M\left\{\int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\, f(\theta,\lambda)\, d\sigma \int_{\sigma'} \bar{Y}^{\alpha}_{nm}(\theta',\lambda')\, f(\theta',\lambda')\, d\sigma'\right\} =$$

$$= M\left\{\int_\sigma \int_{\sigma'} \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\, f(\theta,\lambda) \,.\; f(\theta',\lambda')\, \bar{Y}^{\alpha}_{nm}(\theta',\lambda')\, d\sigma\, d\sigma'\right\} =$$

$$\int_\sigma \int_{\sigma'} \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; Y^{\alpha}_{nm}(\theta',\lambda')\; M\left\{f(\theta,\lambda)\, f(\theta',\lambda')\right\}\, d\sigma d\sigma' =$$

$$\int_\sigma \int_{\sigma'} \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; \bar{Y}^{\alpha}_{nm}(\theta',\lambda')\; \mathrm{cov}\,(f(P), f(Q))\, d\sigma d\sigma'$$

where $P = (\theta, \lambda)$ and $Q = (\theta', \lambda')$. According to (1.10):

$$16\pi^2 \; M\left\{(\bar{C}^{\alpha}_{nm})^2\right\} = \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma \int_{\sigma'} \bar{Y}^{\alpha}_{nm}(\theta',\lambda') \sum_{n=0}^{\infty} \sigma_n^2\; P_n(\cos\psi_{pq})\, d\sigma' =$$

$$= 4\pi \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; \frac{\sigma_n^2}{2n+1}\; d\sigma = \frac{\sigma_n^2}{2n+1}\; 16\pi^2 \quad \text{because of (1.14)}$$

Similarly,

$$M\left\{\bar{C}^{\alpha}_{nm}\, \bar{C}^{\beta}_{pq}\right\} = \frac{\sigma_n^2}{2n+1} \cdot \frac{1}{4\pi} \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; \bar{Y}^{\beta}_{pq}(\theta,\lambda)\, d\sigma = 0 \text{ if } \begin{cases} \alpha \neq \beta \\ n \neq p \\ m \neq q \end{cases}$$

Finally, recalling that $Y^{o}_{oo}(\theta,\lambda) = P_{oo}(\cos\theta) = 1$ for all $-\pi \le \theta \le \pi$ and all λ,

$$M\left\{\bar{C}^{\alpha}_{nm}\right\} = \frac{1}{4\pi} \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; M\left\{f(\theta,\lambda)\right\} d\sigma = \frac{M}{4\pi}\left\{f(\theta,\lambda)\right\} \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma$$

$$= \frac{M}{4\pi}\left\{f(\theta,\lambda)\right\} \int_\sigma \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\; Y^{o}_{oo}(\theta,\lambda)\, d\sigma = 0 \qquad \text{if } n \neq 0,$$

which completes the proof.

2.2 Some Additional Notation

So far, data points on the sphere have been identified by the subscripts i and j. Alternatively, they could be arranged according to a single subscript $k = 2Ni + j$ (where N is the number of parallels in a grid and 2N the number of meridians), so the points in the "0" row, ordered by increasing j, are followed by those in the "1" row, in the same order, etc., the last element in the "N - 1" row closing the sequence. Based on this convention, the set of all values of $f(\theta_i, \lambda_j)$ (or $\overline{f}_{ij}$) can be arranged in N_p - vector form according to k:

$$\underline{z} = [z_0 \, z_1 \ldots z_k \ldots z_{N_p-1}]^T \tag{2.8}$$

where $z_k = f(\theta_i, \lambda_j)$ or $z_k = \overline{f}_{ij}$ with $k = 2Ni + j$

and where N_p is the number of data points, or $2N^2$ for equal angular grids. In a similar way, the coefficients $\overline{C}^{\alpha}_{nm}$ can be ordered according to a single subscript $p = n^2 + \alpha n + m + 1$ (with the understanding that the meaningless $\overline{S}_{n0}$ are not included) defining the following N_c - vector:

$$\underline{c} = [c_0 \, c_1 \ldots c_p \ldots c_{N_c-1}]^T \tag{2.9}$$

$$c_p = \overline{C}^{\alpha}_{nm} \qquad p = n^2 + \alpha n + m + 1$$

where $N_c = (N_{max} + 1)^2$. Using this notation, expression (1.27) for point data quadratures can be written

$$\hat{C}^{\alpha}_{nm} = \underline{f}^{\alpha\,T}_{nm} \, \underline{z} \tag{2.10}$$

where the estimator vector $\underline{f}^{\alpha}_{nm}$, of dimension N_p, has elements of the form

$$f^{\alpha}_{nm} = \chi^{nm}_i \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda$$

under the convention given above relating k, i and j..

Grouping all the estimates $\hat{C}^{\alpha}_{nm}$ in a vector $\hat{\underline{c}}$ ordered in the same way as $\underline{c}$, the relationship between the $\hat{C}^{\alpha}_{nm}$ and $\underline{z}$ can be written, in matrix form,

$$\hat{\underline{c}} = F \underline{z} \tag{2.11}$$

where F is the estimator matrix implied by(2.10). It is a $N_p \times N_c$ matrix (where N_p is the number of data points in the grid), each row being formed by the coefficients of the quadrature formula for the corresponding $\overline{C}^{\alpha}_{nm}$. Such row is also the transpose of the estimator vector of this $\overline{C}^{\alpha}_{nm}$, designated $\underline{f}^{\alpha}_{nm}$ in (2.10).

In the same way as the covariance function between scalars, the cov-iance between vector functions can be defined in terms of $M\{\,\}$:

$$M\left\{\underline{z}\,\underline{z}^T\right\} = C_{zz} \tag{2.12}$$

where C_{zz} is the covariance matrix of $\underline{z}$, of dimension $N_p \times N_p$. This matrix if a function of the relative positions of the points in the grid on which $\underline{z}$ has been determined, in the same way as the scalar covariance depends only on the distance between two points. The elements of C_{zz} are

$$c_{zz}^{rs} = M\left\{z_r\, z_s\right\} = M\left\{f(P_r)\, f(P_s)\right\},$$

i.e., the values of the scalar covariance corresponding to pairs of points in the grid.

In the same way

$$M\left\{\underline{c}\,\underline{c}^T\right\} = C \tag{2.13}$$

is a $N_c \times N_c$ diagonal matrix according to (2.4a-b). C is the covariance matrix of the coefficients. Similarly, the covariance between $\underline{c}$ and $\underline{z}$ is

$$M\left\{\underline{c}\,\underline{z}^T\right\} = C_{cz} = [M\left\{\underline{z}\,\underline{c}^T\right\}]^T = C_{zc}^T \tag{2.14}$$

where C_{cz} is a $N_c \times N_p$ matrix, the elements of which are

$$c_{cz}^{pk} = M\left\{c_p\, z_k\right\} = M\left\{\bar{C}_{nm}^{\alpha}\, f(\theta_i, \lambda_j)\right\}$$

where the right hand side is given by (2.6).

Finally, when estimating the $\bar{C}_{nm}^{\alpha}$, not from samples of $f(\theta, \lambda)$, but from measurements corrupted by noise

$$m(\theta_i, \lambda_j) = f(\theta_i, \lambda_j) + n_{ij} \tag{2.15}$$

the measurement errors can be grouped in a N_p - vector $\underline{n}$ with the same ordering as $\underline{z}$, and the sum of both will be, then, the N_p - <u>vector of observed values</u>

$$\underline{m} = \underline{z} + \underline{n} \tag{2.16}$$

The measurement errors are values that occur in time, as successive observations are carried out: they constitute a time series. The average operator appropriate to them is the usual <u>statistical expectation operator</u> $E\{\ \}$. The measurements are supposed to be <u>unbiased</u>, so $E\{n_k\} = 0$ for all k. The covariance implied by this operator is the usual statistical covariance: $E\{n_k^2\} = \sigma_{kk}^2$, and $E\{n_k\, n_r\} = \sigma_{kr}^2$. This can be generalized for the <u>noise vector</u> $\underline{n}$:

$$E\left\{\underline{n}\,\underline{n}^T\right\} = D \tag{2.17}$$

where D is a $N_p \times N_p$ matrix of elements

$$d_{kr} = E\left\{n_k\, n_r\right\} = \sigma_{kr}^2 \tag{2.18}$$

Both C_{zz} and D have in common a very important property:

$$\left.\begin{array}{l}\underline{x}^T C_{zz}\,\underline{x} \geq 0 \\ \underline{x}^T D\,\underline{x} \geq 0\end{array}\right\} \quad \text{if } \underline{x}^T\underline{x} \neq 0\,,\ \underline{x} \text{ any } N_p \text{ vector,}$$

i.e., they are always <u>positive</u> matrices, moreover, in all the cases considered here, at least D, is <u>positive definite</u>:

$$\underline{x}^T D\,\underline{x} > 0 \quad \text{for all } \underline{x}$$

Positiveness can be inferred readily from the definitions of C_{zz} and D:

i.e., $$\underline{x}^T D\,\underline{x} = \underline{x}^T E\left\{\underline{n}\,\underline{n}^T\right\}\underline{x} = E\left\{\underline{x}^T\underline{n}\,\underline{n}^T\underline{x}\right\} = E\left\{h^2\right\} \geq 0$$

(where $h = \underline{x}^T\underline{n}$), and similarly for $\underline{x}^T C_{zz}\,\underline{x}\left(\text{with } M\{\ \}\right)$.

2.3 Estimation Errors, Sampling Errors, and Propagated Noise

A linear estimator is of the general form

$$\underline{s} = F\,\underline{m}$$

where $\underline{m}$ is the vector of measurements defined by (2.16), and $\underline{s}$ is the vector of estimates, made up in our case of the $\hat{C}^{\alpha}_{nm}$. According to (2.11) and (2.16)

$$\underline{s} \equiv \underline{\hat{c}} = F\,(\underline{z} + \underline{n})$$

In general, the estimates will not be exactly equal to that which is estimated, the difference being the estimation error. In <u>matrix</u> notation

$$\underline{e} = \underline{c} - \underline{\hat{c}} = (\underline{c} - F\,\underline{z}) - (F\,\underline{n}) \tag{2.19}$$

$\underline{e}$ being the estimation error vector. The two terms in the expression above can be defined as the components of this error:

$$\underline{e}_s = \underline{c} - F\,\underline{z}$$

which is the estimation error in the case of noisless (perfect) data; and

$$\underline{e}_\eta = F\,\underline{n}$$

which is the error due to the noise, or <u>propagated noise</u>.

The error e_{sp} may be due to a number of reasons.[1] If it is zero for

[1]Using the relationship $p = n^2 + \alpha n + m + 1$ of paragraph (2.2), e_{sp} stands for the sampling error in $\hat{C}^{\alpha}_{nm}$, and $e_{\eta p}$ for the propagated noise.

some estimator, then its presence in other estimators could be blamed on them being somewhat inadequate. For instance, if the estimator was chosen by taking the elements of F from a set of random numbers, then the estimation error is likely to be always high, as the estimator has nothing to do with the actual problem. In particular, the addition of extra measurements to the vector $\underline{m}$ is not going to bring any general improvement on the estimates. On the other hand, if attention is paid to the nature of the problem when selecting F, one would expect the error to decrease as more data is introduced. If, as the number N_p of samples in $\underline{m}$ tends to infinity, e_{sp} tends to zero, one could say that the error is due to the incomplete sampling of the signal $f(\theta, \lambda)$, and call it the sampling error. This is precisely the case with any of the quadrature formulas to be studied here, all of which can be written formally as linear estimators according to (2.11), and for all of which the error e_{sp} vanishes as the number of samples tends to infinity, because the sums become identical with the integrals defined by (1.4). In this sense it is quite suitable to call e_{sp} the sampling error, as in paragraph (1.3).

2.4 The Quadratic Error Measure

The overall error measure will be defined here as the sum of two quadratic terms: onefor the propagated noise, the other for the sampling error.

(a) Propagated Noise Measure

This measure is the same as in least squares adjustment, i.e., the variance of the error defined in terms of the usual statistical expectation operator

$$\sigma^{2\alpha}_{\eta nm} = E\left\{ (\underline{f}^{\alpha T}_{nm} \, \underline{n})^2 \right\} \tag{2.20}$$

according to (2.10). This variance represents the scatter in the value of $\hat{C}^{\alpha}_{nm}$ due to the uncertainty in the values of the data. In matrix form

$$E_\eta = E\left\{ \underline{e}_\eta \underline{e}^T_\eta \right\} = E\left\{ F \underline{n}\,\underline{n}^T F^T \right\} = F\, E\left\{ \underline{n}\,\underline{n}^T \right\} F^T = F D F^T \tag{2.21}$$

where E_η is a $N_e \times N_e$ matrix, while D was already presented in paragraph (2.2).

In the special case where the measurement errors are uncorrelated, D is diagonal, and (2.20) becomes

$$\sigma^2_{\eta p} = E\left\{ \underline{f}^{\alpha T}_{nm} \underline{n}\,\underline{n}^T \underline{f}^{\alpha}_{nm} \right\} = \underline{f}^{\alpha T}_{nm} D \underline{f}^{\alpha}_{nm} = \sum_{i=0}^{N-1} (X^{nm}_i)^2 \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos^2 \\ \sin^2 \end{Bmatrix} mj\Delta\lambda \, E\left\{ n^2_{ij} \right\} \tag{2.22}$$

which is the usual formula for propagating the covariance of the noise.

(b) Sampling Error Measure

This measure is defined in terms of the isotropic averaging operator of paragraph (2.1)

$$\sigma_{s\,nm}^{2\,\alpha} = M\left\{(c_P - \underline{f}_{nm}^{\alpha}\ \underline{z})^2\right\} \equiv M\left\{(\bar{C}_{nm}^{\alpha} - \underline{f}_{nm}^{\alpha} z)^2\right\} \tag{2.23}$$

or, in matrix form

$$\begin{aligned} E_s &= M\left\{\underline{e}_s\,\underline{e}_s^{T}\right\} = M\left\{(\underline{c} - F\,\underline{z})(\underline{c} - F\,\underline{z})^{T}\right\} \\ &= C - 2\,C_{cz}\,F^{T} + F\,C_{zz}\,F^{T} \end{aligned} \tag{2.24}$$

where E_s is a $N_e \times N_e$ matrix, and C, C_{cz}, C_{zz} were introduced in paragraph (2.2).

(c) Total Error Measure

The total measure is the sum of (a) and (b)

$$\sigma_{nm}^{2\,\alpha} = \sigma_{\eta\,nm}^{2\,\alpha} + \sigma_{s\,nm}^{2\,\alpha} \tag{2.25}$$

or, in matrix form,

$$\begin{aligned} E_T &= E_S + E_{\eta} = C - 2C_{cz}\,F^{T} + FC_{zz}\,F^{T} + FD\,F^{T} = \\ &C - 2C_{cz}\,F^{T} + F(C_{zz} + D)\,F^{T} \end{aligned} \tag{2.26}$$

where E_T is the $N_p \times N_p$ error matrix associated with F and with the covariances that define C, C_{cz} and $(C_{zz} + D)$. Expression (2.26) is a special case of the formula for "E_{ss}" in least squares collocation (for instance, Moritz (1978), Ch. 3, eqn, (3.20)); moreover, it belongs to a family of formulas also found in the minimum variance estimation and filtering of time series and of processes sampled on the euclidean plane.

The total measure has been chosen simply as the sum of $\sigma_{s\,nm}^{2\,\alpha} + \sigma_{\eta\,nm}^{2\,\alpha}$ by making the basic assumption that the sampling error and the propagated noise are due to completely independent causes. The first depends on the values $f(\theta_i, \lambda_j)$, while the second depends on the measurement errors of instruments that, at least ideally, operate with accuracies unaffected by the quantities measured, or in such way that any interactions can be eliminated by simple corrections.

The columns of F are defined by the quadrature formula used, and such formulas either satisfy, or tend to satisfy, orthogonality conditions (paragraph (1.3 (c)). For this reason, provided that C_{zz} and D belong to the type to be described in paragraph (2.9), matrices E_{η}, E_S, and, thus, E_T, are either diagonal or diagonal dominant, and in the latter case tend to become diagonal as the sampling intervals decrease, or $N_p \to \infty$. For this reason the correlations among the errors for individual coefficients are, or "tend to be", very small.

The diagonal elements of the error matrices E_s, E_η and E_T are the variances of the errors in the respective coefficients, as defined by (2.20), (2.23), and (2.25), respectively.

2.5 The Meaning of the Error Measure

The treatment of the propagated noise is the same as in least squares adjustment, so this part of the error measure should be easily understood. The sampling error measure, on the other hand, is a geometrical measure: $M\{\ \}$ belongs, as a concept, in the field of integral geometry, or the study of "geometric probabilities". This is a branch of mathematics closely related to integration and to measure theory, and also to statistical mechanics. In geodesy, this type of idea is relatively new (Kaula, 1959), Moritz (1965), but it has been used already extensively enough to show its considerable worth.

From expressions (1.10) and (1.11), the covariance and the power spectrum are functions of each other. Since either of them, and the sampling grid, define matrices C, C_{gz} and C_{zz} in expression (2.24), it follows that a statement on $\sigma^2_{s\,nm}$ is, somehow, also a statement on the performance of F for all the functions that have the same power spectrum that determines the diagonal elements of C. To put this more precisely, consider a function $f_1(\theta, \lambda)$ having the given power spectrum. If $\bar{C}^{\alpha}_{nm}$ were estimated for f_1 and also, at least ideally, for all its rotations, then the mean square of the sampling error e_{sp} in $\hat{C}^{\alpha}_{nm}$ for all this functions would be, by definition of $M\{\ \}$, the measure $\sigma^{2\,\alpha}_{s\,nm}$. If a second function f_2 (perhaps not a rotation of f_1) and all its rotations were then analysed in the same way, the average of e^2_{sp} for all these functions would be, once more, $\sigma^{2\,\alpha}_{s\,nm}$, as long as f_2 has the same spectrum as f_1. Moreover, the average of e^2_{sp} for f_1, f_2, and their rotations put together, would also be $\sigma^2_{s\,nm}$. In fact, if we had a finite set of functions $f_1, f_2, \ldots f_n$, with arbitrary n, all with the same power spectrum (or covariance), then e^2_{sp} would average $\sigma^{2\,\alpha}_{s\,nm}$ for all the f_i and their rotations.

It appears, from the preceeding discussion, that one could take a simple step and say "$\sigma^{2\,\alpha}_{s\,nm}$ is the mean of the sampling error squared of the estimator $\hat{C}^{\alpha}_{nm} = \underline{f}^{\alpha}_{nm}\,\underline{z}$, over all possible functions with the given power spectrum." Unfortunately, as mentioned in the Introduction, the sphere is a rather wicked surface. There is a theorem by Lauritzen (1973) that states the impossibility of having the same average $\sigma^2_{s\,nm}$ for all functions as for every function, when the distribution of the ensemble happens to be gaussian. Moritz (1978) has endeavoured to show that this is no problem if the ensemble of functions is not gaussian, but using his conclusions here would force the introduction of a rather strange requirement of "non-gaussness" on the ensemble of the signals analysed that is best left out, if possible.

Perhaps there is a way out in going back to the idea of a finite set of functions f_i, where the problem does not exist, by saying:

"the error measure $\sigma_{s\,nm}^{2\,\alpha}$, for a certain estimator and a certain type of signal power spectrum, is the average of the square of the sampling error in $\hat{C}_{nm}^{\alpha}$ for all functions with the given power spectrum EVER TO BE ANALYSED with that estimator, and for all their rotations".

After all, accuracy is what geodesists are always interested in, not perfection.

2.6 Simple Formulas for Area Means

The numerical studies of section 3 concentrate in area mean type formulas, because area means are preferred for collating information, particularly on a global basis, at present. The formulas to be studied here and in that section can be divided into "simple" and "optimal". The name "simple" is given here to expressions of the type

$$\hat{C}_{nm}^{\alpha} = \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \bar{f}_{ij} \int_{\sigma_{ij}} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)\, d\sigma \tag{2.27}$$

where μ_n in a scale factor affecting the nth harmonic as a whole. Expressions of this type have been developed more or less intutitively, along the lines of the following reasoning:

If the signal were constant on each block, it will equal its mean value there, and the coefficients of such a fucntion would be precisely

$$\bar{C}_{nm}^{\alpha} = \frac{1}{4\pi} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \bar{f}_{ij} \int_{\sigma_{ij}} \bar{Y}_{nm}^{\alpha}(\theta,\lambda)\, d\sigma \tag{2.28}$$

according to (1.4). In general, most signals are not equal to their mean value over whole blocks, so the expression would not be exact. In most cases, the signal would have fluctations in each block, and it would be less smooth than a function that is constant over each block, so using the formula above with $\bar{f}_{ij}$ as data may result in the $\tilde{\bar{C}}_{nm}^{\alpha}$ of a smoothed function. As a refinement, one could try to de-smooth the $\tilde{\bar{C}}_{nm}^{\alpha}$. If the blocks were circular, the relationship between "true" and "smooth" $\bar{C}_{nm}^{\alpha}$ would be

$$\overline{C}^{\alpha}_{nm} = \frac{1}{\beta_n} \tilde{\tilde{C}}^{\alpha}_{nm} \tag{2.29}$$

where β_n is known as the Pellinen smoothing factor of degree n The relationship between β_n and the radius of the circular blocks is given in paragraph (4.3). For small blocks, experience shows that there is little difference between the area means of geodetic data on circular or on square blocks, so the error is small if one assumes that they are the same; in such case the modified expression

$$\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi\beta_n} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{f}_{ij} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma = \overline{C}^{\alpha}_{nm} \tag{2.30}$$

could be used; in practice, this is only an approximation, though a good one, as showed by Katsambalos (1979), who tested this expression extensively.

In addition to (2.28) and (2.30), Lowes (1978) has proposed using

$$\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi\beta_n^2} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{f}_{ij} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma \tag{2.31}$$

to estimate the harmonic coefficients. All these expressions have the property that, because $\frac{1}{\beta_n} \to 1$, and $\sum_i \sum_j \overline{f}_{ij} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma \to \int_{\sigma_{ij}} f(\theta,\lambda)\, \overline{Y}^{\alpha}(\theta,\lambda)\, d\sigma$ as $\Delta_{ij} \to 0$ (or $N_p \to \infty$), it is true that the error $e_{sp} = \overline{C}^{\alpha}_{nm} - \hat{C}^{\alpha}_{nm} \to 0$ with $N_p \to \infty$; in other words: e_{sp} is properly called a sampling error in the sense given to this term in paragraph (2.3).

Comparing (2.28), (2.30), and (2.31) it is easy to see that they all belong to a class of expressions of the form (2.27), with $\mu_n = \frac{1}{4\pi}$, $\mu_n = \frac{1}{4\pi\beta_n}$, and $\mu_n = \frac{1}{4\pi\beta_n^2}$, respectively.

The scaling factor μ_n can also be regarded as a de-smoothing factor, if one wishes to retain the intuitive meaning of these formulas. In the notation of paragraph (2.2), these expressions can be written, according to (2.27), as

$$\hat{C}^{\alpha}_{nm} = \mu_n (\underline{h}^{\alpha}_{nm})^T \underline{z} \tag{2.32}$$

with $\mu_n \underline{h}^{\alpha}_{nm} = \underline{f}^{\alpha}_{nm}$.

Replacing (2.32) in the definition of the sampling error measure, (2.23), and adding with respect to m and α to obtain the total error in the nth harmonic:

$$\sum_{\alpha=0}^{1} \sum_{m=0}^{n} \tilde{\sigma}^2_{s\,nm} = \sigma_n^2 - \left[2 \sum_{\alpha=0}^{1} \sum_{m=0}^{n} \underline{c}^T_{nm\alpha,z}\ \underline{h}^{\alpha}_{nm} \right] \mu_n +$$
$$\left[\sum_{\alpha=0}^{1} \sum_{m=0}^{n} (\underline{h}^{\alpha}_{nm})^T (C_{zz})\, \underline{h}^{\alpha}_{nm} \right] \mu_n^2 \tag{2.33}$$

(where $\underline{c}^{T}_{nm\alpha,z}$ is a row of $C_{\alpha z}$). This is the sum of certain diagonal elements of E_s, according to (2.24), when the estimator has the form (2.27). Clearly, (2.33) is a quadratic function of the scalar μ_n, and as such it can have either a maximum or a minimum. If C_{zz} is positive definite, it must be a a minimum. Finding the corresponding value of μ_n is the same as finding the formula of type (2.27) that has the smallest sampling error per harmonic for signals with the covariance (power spectrum) specified by C_{zz}. In addition to the sampling error, the measure of the propagated noise can be added to obtain

$$\sum_{\alpha=0}^{1} \sum_{m=0}^{n} \sigma^{2\alpha}_{nm} = \sigma^2_n - \left[\sum_{\alpha=0}^{1} \sum_{m=0}^{n} 2\, \underline{c}^{T}_{nm\alpha,z}\, \underline{h}^{\alpha}_{nm}\right] \mu_n + \left[\sum_{\alpha=0}^{1} \sum_{m=0}^{n} (\underline{h}^{\alpha}_{nm})^{T} (C_{zz} + D)\, \underline{h}^{\alpha}_{nm}\right] \mu^2_n \tag{2.34}$$

This is also quadratic and has a minimum, and finding the optimum μ_n is the subject of the next paragraph.

2.7 Optimum de-Smoothing Factors

The coefficients of μ^2_n and μ_n in (2.33) and (2.34) are both real scalars, and so is the independent term σ^2_n. The expressions represent parabolas, and because both C_{zz} and D are positive, if the further (and likely) assumption is made that they are also definite, then $\frac{\partial^2}{\partial\mu^2_n} \sum_{\alpha} \sum_{m} \sigma^{2\alpha}_{s,nm} > 0$,

and the parabola has a minimum where μ_n satisfies the condition

$$\frac{1}{2} \frac{\partial}{\partial\mu_n} \sum_{\alpha} \sum_{m} \sigma^{2\alpha}_{s\,nm} = -\sum_{\alpha} \sum_{m} \underline{c}^{T}_{nm\alpha,z}\, \underline{h}^{\alpha}_{nm} + \left[\sum_{\alpha} \sum_{m} (\underline{h}^{\alpha}_{nm})^{T} C_{zz}\, \underline{h}^{\alpha}_{nm}\right] \mu_n \qquad \text{for the sampling error (2.33)}$$

i.e., at

$$\hat{\mu}_n = \frac{\sum_{\alpha} \sum_{m} \underline{c}^{T}_{nm\alpha,z}\, \underline{h}^{\alpha}_{nm}}{\sum_{\alpha} \sum_{m} (\underline{h}^{\alpha}_{nm})^{T} C_{zz}\, \underline{h}^{\alpha}_{nm}} \tag{2.35}$$

or at

$$\overset{*}{\mu}_n = \frac{\sum_{\alpha} \sum_{m} \underline{c}^{T}_{nm\alpha,z}\, \underline{h}^{\alpha}_{nm}}{\sum_{\alpha} \sum_{m} (\underline{h}^{\alpha}_{nm})^{T} (C_{zz} + D)\, \underline{h}^{\alpha}_{nm}} \tag{2.36}$$

for the total error (2.34).

Expressions (2.28), (2.30), (2.31), and (2.36) will be studied further, by means of computed examples, in section 3.

2.8 Least Squares Collocation

In the notation of Paragraph (2.2), optimizing μ_n is the same as obtaining the optimal vectors $\underline{f}^{\alpha}_{nm}$ of the form

$$\underline{f}^{\alpha}_{nm} = \mu_n \underline{h}^{\alpha}_{nm}$$

for the estimator

$$\hat{\underline{c}} = F\underline{m}$$

where $\underline{m} = \underline{z} + \underline{n}$, and the $(\underline{f}^{\alpha}_{nm})^T$ are the rows of F . If no restriction is placed on the form the rows of F can take, then a reasoning similar to that in the preceeding paragraph leads to the best possible linear estimator for the $\hat{C}^{\alpha}_{nm}$.

Considering the total measure of error for $0 \leq n \leq N$:

$$\sigma^2_{\epsilon} = \sum_{\alpha=0}^{1} \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sigma^{2\alpha}_{nm} = \sum_{n=0}^{N-1} \sigma^2_n - 2 \sum_{\alpha=0}^{1} \sum_{n=0}^{N-1} \sum_{m=0}^{n} \underline{c}^T_{nm\alpha,z} \underline{f}^{\alpha}_{nm} + \sum_{\alpha=0}^{1} \sum_{n=0}^{N-1} \sum_{m=0}^{n} (\underline{f}^{\alpha}_{nm})^T (C_{zz} + D) \underline{f}^{\alpha}_{nm} \tag{2.37}$$

it is not difficult to see that, because all σ^2_{nm} are non-negative, finding the F that minimizes their sum is the same as finding the F that minimizes them individually. The sum of the mean squared errors of all coefficients is the trace of the error matrix E_T of (2.26) :

$$\sigma^2_{\epsilon} = \sum_{\alpha=0}^{1} \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sigma^{2\alpha}_{nm} = tr[E_T]$$

To obtain the condition for a minimum, one must differentiate (2.37) so, according to (2.26),

$$\frac{1}{2} \frac{\partial tr}{\partial F} [E_T] = -C^T_{cz} + (C_{zz} + D) F^T = 0 \tag{2.38}$$

as found using well-known matrix analysis formulas.
From this follows that

$$F = C_{cz} (C_{zz} + D)^{-1} \tag{2.39}$$

is the F that minimizes (2.37), provided that $(C_{zz} + D)$ is positive definite. As already explained, both matrices are always positive and their sum is usually definite. The expression for the optimal estimator for $\bar{C}^{\alpha}_{nm}$ is

$$\hat{C}^{\alpha}_{nm} = (\overset{*\alpha}{\underline{f}}_{nm})^T \underline{m} = \underline{c}^T_{nm}{}_{\alpha,z} (C_{zz} + D)^{-1} \underline{m} \tag{2.40}$$

where $(\overset{*}{\underline{f}}{}^{\alpha}_{nm})^T$ is the row of the optimal estimator matrix F corresponding to $\hat{C}^{\alpha}_{nm}$.

The use of expression (2.40) is, in brief, least squares collocation applied to spherical harmonic analysis.

When the optimal F is used, the error matrix becomes, according to (2.26) and (2.39),

$$E_T = C - C_{sz}(C_{zz} + D)^{-1} C^T_{sz} \tag{2.41}$$

and the total error measure is the trace of this matrix: by definition, the smallest for all possible F.

Clearly, whether one is interested in estimating coefficients or in determining the likely accuracy of such estimates, using expressions (2.40) or (2.41) require a knowledge of either $(C_{zz} + D)^{-1}$ (inversion) or, at least, of $C_{sz}(C_{zz} + D)^{-1}$ (solution). Because the matrix $(C_{zz} + D)$ has dimension $N_p \times N_p$, obtaining either requires, by usual linear algebra methods, $O(N_p^3)$ (or $O(N^6)$), operations. In the case of a $1^\circ \times 1^\circ$ grid, $N_p = 64800$, so, at some 200000 products and sums (double precision) per second, a modern computer like the one at OSU would need about one century to obtain all $\hat{C}^{\alpha}_{nm}$ to degree and order 180 from data on such a grid. Fortunately, as explained in paragraph (2.9), if the covariance functions of signal and noise both satisfy certain conditions, and if $\Delta\lambda$ is constant for the whole grid, then both C_{zz} and D (consequently their sum) can be inverted in much fewer operations than by conventional methods, because they possess a particularly strong structure. Moreover, the optimal estimator $\hat{C}^{\alpha}_{nm} = (\overset{*}{\underline{f}}{}^{\alpha}_{nm})^T \underline{m}$ turns out to be of the form (1.24) or (1.7), depending on the kind of data $\underline{m}$, so, under rather general conditions, the optimal estimator of $\overline{C}^{\alpha}_{nm}$ is also the best quadratures type formula for point data or for area means, as the case may be.

The conditions mentioned above are satisfied, for instance, when both the geometrical covariance $\mathrm{cov}(f(P), f(Q))$ and the stochastic covariance $E\{n_{ij} n_{kz}\}$ $(P \equiv (\theta_i, \lambda_j)$, $Q \equiv (\theta_k, \lambda_z))$ are isotropic, i.e., functions only of the separation between the points P and Q. By definition of $M\{\}$, the geometrical covariance obtained using this operator is isotropic, so C_{zz} has the desired structure. A common assumption regarding good instruments is that the n_{ij} are uncorrelated, so D is diagonal. If the errors are stationary, so their variances are constant, or at least constant along parallels, then matrix D has the required structure, and inverting $C_{zz} + D$ can be greatly expedited. In practice, however, this is not likely to be the case, as the number and quality of measurements will vary from region to region, resulting in different σ^2_{ij} both globally and along parallels. As a result, the best linear estimator in terms of the chosen error measure will not have the quadratures form, and it will be very difficult to compute when the number of data values is very large. Nevertheless, as shown in section 3, quadrature formulas can give reasonable estimates of the $\overline{C}^{\alpha}_{nm}$ with noisy data sets where the noise is uneven, so it would be interesting to get the best quadrature formula for a particular combination of signal and noise, provided that such a formula can be obtained without undue effort.

2.9 The Best Quadrature Formula for non-Uniform, Uncorrelated Noise

If the variance of the noise fluctuates along parallels, matrix D, though diagonal, is such that the minimization of the error measure (2.37)

$$\sigma^2_{\epsilon} = \mathrm{tr}\ [C - 2C_{cz} F^T + F(C_{zz} + D) F^T] = \Phi(D) \tag{2.42}$$

(see also (2.26))
can be very difficult with large N_p, and the optimal estimator is not of the quadrature type. Introducing a "modified noise matrix" L, also diagonal and where the diagonal elements are

$$\ell_{kk} = \frac{1}{2N} \sum_{j=0}^{2N-1} \sigma^2_{ij} \quad (\text{with } k = 2N\, i + j) \tag{2.43}$$

the following modified error measure $\Phi(L)$ can be defined:

$$\Phi(L) = \mathrm{tr}\ [C - 2C_{cz} F^T + F(C_{zz} + L) F^T] \tag{2.44}$$

The optimal estimator for this measure is easy to obtain, and is of the quadratures type.

The parts of $\Phi(D)$ and $\Phi(L)$ that measure the sampling errors are identical, so any difference between the overall measures must come from the "noise propagation" parts $\mathrm{tr}\ [F D F^T]$ and $\mathrm{tr}\ [F L F^T]$. If the estimator (not necessarily optimal) happens to be of the quadratures type, i.e., for point data:

$$\hat{C}^{\alpha}_{nm} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} x_i^{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m j \Delta\lambda\ [f(\theta_i, \lambda_j) + n_i] \tag{2.45}$$

then the propagated noise is, assumming the n_{ij} to be uncorrelated,

$$\sigma^{2\alpha}_{\eta nm} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} (x_i^{nm})^2 \begin{Bmatrix}\cos^2\\ \sin^2\end{Bmatrix} m j \Delta\lambda\ E\{n^2_{ij}\}$$

so

$$\Phi_{\eta}(D) = \mathrm{tr}\ [F D F^T] = \sum_{\alpha=0}^{1} \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sigma^{2\alpha}_{\eta nm} = \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} (x_i^{nm})^2 \sum_{j=0}^{2N-1} \sigma^2_{ij} \tag{2.46}$$

The "modified noise" on the other hand, is, according to (2.43) and (2.44):

$$\Phi_{\eta}(L) = \mathrm{tr}\,[F\,L\,F^{T}] = \sum_{n=0}^{N-1}\sum_{m=0}^{n}\sum_{i=0}^{N-1}(X_i^{nm})^2\sum_{j=0}^{2N-1}(\cos^2 mj\Delta\lambda + \sin^2 mj\Delta\lambda)\cdot$$

$$\cdot\frac{1}{2N}\sum_{j=0}^{2N-1}\sigma_{ij}^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{n}\sum_{i=0}^{N-1}(X_i^{nm})^2\sum_{j=0}^{2N-1}\sigma_{ij}^2 \tag{2.47}$$

Comparing the expressions for $[F\,L\,F^{T}]$ and for $[F\,D\,F^{T}]$, it follows that they are identical, and since the "sampling" parts are also identical in (2.42) and (2.44), then

$$\mathrm{tr}\,[C - 2C_{cz}F^{T} + F(C_{zz} + L)\,F^{T}] = \mathrm{tr}[C - 2C_{cz}F^{T} + F\;(C_{zz} + D)\,F^{T}] \tag{2.48}$$

This means that <u>the actual and the modified error measures must coincide if the estimator is of the quadratures type.</u>

Replacing D with L in equation (2.38) and solving for the estimator matrix, one gets

$$F_L = C_{cz}(C_{zz} + L)^{-1} \tag{2.49}$$

where F_L is the estimator matrix that minimizes the modified error measure (2.44). Because of the way L has been defined, this estimator is of the quadratures type, so the modified and the actual error measures coincide, as just shown.

Assume that there is an estimator, different from $\underline{\hat{c}} = F_L\,\underline{m}$ but also of the quadratures type, the estimator matrix of which is $\tilde{F}$, and such that:

$$\mathrm{tr}\;[C - 2C_{cz}\,\tilde{F}^{T} + \tilde{F}(C_{zz} + D)\tilde{F}^{T}] < \mathrm{tr}\;[C - 2C_{cz}\,F_L^{T} + F_L(C_{zz} + D)\;F_L^{T}]$$

Then, according to (2.48),

$$\mathrm{tr}\;[C - 2C_{cz}\,\tilde{F}^{T} + F\;(C_{zz} + L)\;\tilde{F}^{T}] < \mathrm{tr}\;[C - 2C_{cz}\,F_L^{T} + F_L\,(C_{zz} + L)\;F_L^{T}] \tag{2.50}$$

which contradicts the fact that F_L minimizes the modified error measure. Therefore, (2.50) cannot be true, and F_L <u>must be the matrix of the optimal quadratures type estimator</u> that minimizes the <u>actual</u> error measure σ_ϵ^2.

The optimal quadratures type estimator, as the name indicates, is the best of a certain kind, not the absolute best. The best estimator, when no conditions as to its form are imposed, will not be (in general) of the quadratures type, unless D happens to have the "right form" specified before, i.e., unless D = L.

When D = L, the optimal estimator and the best quadrature formula coincide. Regardless of this, the quadrature formula obtained from (2.49) is the best, so its error measure is a lower bound for those of all other quadratures formulas with the given signal and noise.

When $D \neq L$, while minimizing the sum of all error variances $\sigma^{2\alpha}_{nm}$, i.e., tr (E_T), the optimal quadrature formula does not minimize each individual variance $\sigma^{2\alpha}_{nm}$. To show this, consider the propagated noise measure for $\overline{C}^{\alpha}_{nm}$ when the n_{ij} are uncorrelated (to simplify the argument):

$$\sigma^{2\alpha}_{\eta_{nm}} = \sum_{i=0}^{N-1} (\chi_i^{nm})^2 \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ \sin \end{Bmatrix}^2 mj\,\Delta\lambda\ \sigma^2_{ij} = \sum_{i=0}^{N-1} (\chi_i^{nm})^2 \left[\frac{1}{2} \sum_{j=0}^{2N-1} \sigma^2_{ij} + \frac{1}{2} \sum_{j=}^{2N-1} \begin{Bmatrix} \cos \\ -\sin \end{Bmatrix} 2mj\,\Delta\lambda\ \sigma^2_{ij} \right]$$

The modified error measure, minimized by the formula, is

$$\sum_{i=0}^{N-1} (\chi_i^{nm})^2 \sum_{j=0}^{2N-1} \begin{Bmatrix} \cos^2 \\ \sin^2 \end{Bmatrix} mj\,\Delta\lambda\ \frac{1}{2N} \sum_{j=0}^{2N-1} \sigma^2_{ij} = \sum_{i=0}^{N-1} (\chi_i^{nm})^2\ \frac{1}{2} \sum_{j=0}^{2N-1} \sigma^2_{ij}$$

Clearly, both are not the same, unless

$$\sum_{j=0}^{2N-1} \begin{Bmatrix} \cos \\ -\sin \end{Bmatrix} 2mj\,\Delta\lambda\ \sigma^2_{ij} = 0$$

which is not likely to be fulfilled for arbitrary σ^2_{ij}. However, looking at the reasoning which leads to (2.47), one can see that the sums of the modified and the actual error measures for pairs $(\overline{C}_{nm}, \overline{S}_{nm})$, and also for the individual $\overline{C}_{no}$, are already identical. From this follows that the variance of the error per degree

$$\sum_{\alpha=0}^{1} \sum_{m=0}^{n} \sigma^{2\alpha}_{nm} = \sigma^2_{\epsilon n} \tag{2.51}$$

and per average coefficient per degree:

$$\delta^2_{\epsilon_n} = \frac{\sigma^2_{\epsilon n}}{2n+1} \tag{2.52}$$

are also identical to the modified measure. So, while nothing can be predicated of individual coefficients, the error for each harmonic as a whole and that for the "average coefficient" in it are going to be minimum. By Parseval's identity (1.12), if the coefficients were used to calculate, say, geoidal undulations, the mean squared error of the computed geoid, globally, would be the same as the sum of the error squared of the normalized coefficients, so individual coefficient variances are of little interest in this and similar applications, while the $\sigma^2_{\epsilon n}$ are very important. This shows that the optimal quadratures formula when $D \neq L$ can be just as useful as when $D = L$.

The discussion in this paragraph has been centered on point value type formulas; the conclusions apply equally well to area mean type formulas, the extension of the reasoning being quite straightforward.

2.10 The Structure of the Covariance Matrix and its Consequences

The following discussion summarizes some results presented by this author in a previous report (Colombo, 1979a). In order to be able to calculate the variance of the error $\sigma^{2\alpha}_{n,m}$ with expressions such as (2.26), and also to be able to obtain the optimal estimator according to collocation theory, it is necessary to create and invert the $N_p \times N_p$ matrix $(C_{zz} + D)$, which can be very large if the number of data N_p is large. In the case of regularly sampled data this two problems can be greatly simplified if the covariances and the grid has certain symmetries. The most important of these are: (a) the sampling in longitude must be at constant intervals and along parallels (or parallel bands, i.e., rows of blocks); (b) for given i and p the covariances $\mathrm{cov}(u(\theta_i, \lambda_j), v(\theta_p, \lambda_Q))$ (or $\mathrm{cov}(\bar{u}_{ij}, \bar{v}_{PQ})$) and $E\{n_{ij}, n_{pq}\}$, must depend only on $|j-q|$. It is also very advantageous, though not essential, that the grid be symmetrical with respect to the Equator.

In what follows N_r is the number of parallels and Nl the number of meridians ($N_r = N$, $Nl = 2N$ when the grid is equal angular).

Under this set of conditions, if the data vector $\underline{m}$ is ordered according to (2.8) and is subdivided into partitions $\underline{m}_i$, where

$$\underline{m}_i = [m_{i0}\, m_{i1} \ldots m_{iN1-1}]^T$$

includes all data values in the same parallel or row of blocks, then the matrix $(C_{zz} + D)$ can be partitioned into N_r^2 blocks C^{ip}, each of dimension $Nl \times Nl$, containing the covariances between the data along rows i and p.

Each block C^{ip} has a Toeplitz circulant structure, because its elements satisfy the relationships

$$c^{ip}_{jq} = c^{ip}_{j+1\, q+1} \; ; \; c^{ip}_{j0} = c^{ip}_{j-1\; N1-1} \qquad \text{when } j > 0$$

which follow from the fact that parallels are circular, and that the covariance between points in parallels i and p is a function of $|j-q|$. Moreover, the elements in the first row or column (the C^{ip} are symmetrical) also satisfy

$$c^{ip}_{0q} = c^{ip}_{0\,N1-q} \qquad \text{when } q > 0$$

Therefore, the first row can be represented exactly as a sum of $\frac{1}{2}Nl + 1$ cosines:

$$c^{ip}_{0q} = \sum_{m=0}^{\frac{1}{2}N1} a^{ip}_m \cos m \frac{2\pi}{N1} q \tag{2.53}$$

The a^{ip}_m form the discrete Fourier transform of the sequence

$$c^{ip}_{00}, c^{ip}_{02}, \ldots c^{ip}_{0\,N1-1}$$

If

$$r_m^{ip} = H\, a_m^{ip} \tag{2.54}$$

where $H = \begin{cases} N1 & \text{if } m = 0 \\ \frac{N1}{2} & \text{if } m \neq 0 \end{cases}$

If $R(m)$ is the matrix where each "ip" element equals r_m^{ip}, then (as explained by Colombo, (op. cit.)) inverting $(C_{zz} + D)$ is equivalent to inverting the $Nr \times Nr$ matrices $R(m)$ for $m = 0, 1, \ldots \frac{1}{2}N1$
Isotropic covariances satisfy the "$|j - q|$ condition" mentioned above, so, for fixed $\Delta\lambda$, the covariance matrix always has this regular structure.

Let

$$\underline{c}_m^{\alpha} = [\begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m\,0\,\Delta\lambda, \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m\Delta\lambda, \ldots \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} m(N1-1)\,\Delta\lambda]^T$$

A vector of the type

$$\underline{v}^{\alpha} = [v_0\, \underline{c}_m^{\alpha T},\ v_1\, \underline{c}_m^{\alpha T}, \ldots v_{Nr-1}\, \underline{c}_m^{\alpha T}]^T \tag{2.55}$$

shall be called, for convenience, a vector "<u>of frequency m</u>".

Under the conditions described before <u>all the eigenvectors of $(C_{zz} + D)$ are vectors of frequency m</u>, with $m = 0, 1, \ldots \frac{1}{2}N1$. Moreover, if λ_{tm} $(t = 1, 2, \ldots Nr)$ is one of the Nr eigenvalues of $R(m)$, and if

$$\underline{\xi}_{tm} = [s_0^{tm} \ldots s_{Nr-1}^{tm}]^T$$

is the corresponding eigenvector of $R(m)$, then λ_{tm} is also an eigenvalue of $(C_{zz} + D)$, and the pair

$$\underline{s}_{tm}^{\alpha} = [s_0^{tm}\, \underline{c}_m^{\alpha T}, \ldots s_{Nr-1}^{tm}\, \underline{c}_m^{\alpha T}]^T$$

the <u>two</u> corresponding eigenvectors of $(C_{zz} + D)$. Therefore, to decompose the large covariance matrix in eigenvectors and eigenvalues is equivalent to decomposing the $\frac{1}{2}$ Nl + 1 matrices $R(m)$, and this is why the latter are relevant to the inversion of $(C_{zz} + D)$: the eigenvalues of the inverse are the reciprocal of the λ_{tm}, while its eigenvectors are the same as the $\underline{s}_{tm}$. Further, this implies that $(C_{zz} + D)^{-1}$ has the same structure as the covariance matrix, i.e., it consists of Toeplitz-circulant blocks.

Since $(C_{zz} + D)^{-1}$ has eigenvectors of frequency m, then, if $\underline{h}$ is a linear combination of vectors of a given frequency, $\underline{z} = (C_{zz} + D)^{-1}\,\underline{h}$ is also a linear combination of vectors of that frequency. In the case of point data, from expression (2.6) follows that the cross-covariances vector in (2.40) is

$$\underline{c}^T_{nm\alpha,z} = \frac{\sigma_n^2}{2n+1}\,[\overline{P}_{nm}(\cos\theta_0)\,\underline{c}_m^{\alpha T}\ \ldots\ \overline{P}_{nm}(\cos\theta_{Nr-1})\,\underline{c}_m^{\alpha T}]$$
$$= [k_0^{nm}\,\underline{c}_m^{\alpha T}\ \ldots\ k_{Nr-1}^{nm}\,\underline{c}_m^{\alpha T}] \tag{2.56}$$

Define

$$\underline{k}^{nm} = [k_0^{nm}\ \ldots\ k_{Nr-1}^{nm}]^T$$

and

$$\underline{\chi}^{nm} = [\chi_0^{nm}\ \ldots\ \chi_{Nr-1}^{nm}]^T$$

Then $\overset{*}{\underline{f}}{}^{\alpha}_{nm} = (C_{zz} + D)^{-1}\underline{c}_{nm\alpha,z}$ must be of the form

$$\overset{*}{\underline{f}}{}^{\alpha}_{nm} = [\chi_0^{nm}\,\underline{c}_m^{\alpha T}\ \ldots\ \chi_{Nr-1}^{nm}\,\underline{c}_m^{\alpha T}]^T \tag{2.57}$$

where, according to Colombo (ibid),

$$\underline{\chi}^{nm} = R(m)^{-1}\,\underline{k}^{nm} \tag{2.58}$$

Similarly, for area means,

$$\underline{c}_{nm\alpha,\bar{z}} = \frac{\sigma_n^2}{(2n+1)\Delta_{ij}}\ [\ldots \int_{\theta_i}^{\theta_i+\Delta\theta} \overline{P}_{nm}(\cos\theta)\,\sin\theta\, d\theta \int_{\lambda_i}^{\lambda_j+\Delta\lambda} \underline{c}_m^{\alpha T}\, d\lambda \ldots]^T$$
$$= \frac{\sigma_n^2}{(2n+1)\Delta_{ij}}\ [\ldots \int_{\theta_i}^{\theta_i+\Delta\theta} \overline{P}_{nm}(\cos\theta)\,\sin\theta\, d\theta \left(\begin{Bmatrix}A(m)\\-B(m)\end{Bmatrix}\underline{c}_m^0 + \begin{Bmatrix}B(m)\\A(m)\end{Bmatrix}\underline{c}_m^1\right)^T \ldots]^T \tag{2.59}$$

(where Δ_{ij} is supposed to be independent of j) according to expressions (1.7) and (2.7), so

$$\overset{*}{\underline{f}}{}^{\alpha}_{nm} = [\ldots\ \chi_i^{nm}\left(\begin{Bmatrix}A(m)\\-B(m)\end{Bmatrix}\underline{c}_m^0 + \begin{Bmatrix}B(m)\\A(m)\end{Bmatrix}\underline{c}_m^1\right)^T \ldots]^T$$

In conclusion, the optimal estimator for point values has the form

$$\hat{\underline{c}}{}^{\alpha}_{nm} = (\overset{*}{\underline{f}}{}^{\alpha}_{nm})^T\,\underline{m} = \sum_{i=0}^{Nr-1}\sum_{j=0}^{Nl-1} \chi_i^{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda\, m_{ij} \tag{2.60}$$

while that for area means is of the type

$$\hat{\underline{c}}{}^{\alpha}_{nm} = (\overset{*}{\underline{f}}{}^{\alpha}_{nm})^T\,\underline{m} = \sum_{i=0}^{Nr-1}\sum_{j=0}^{Nl-1} \chi_i^{nm}\ [\begin{Bmatrix}A(m)\\B(m)\end{Bmatrix}\cos mj\Delta\lambda + \begin{Bmatrix}B(m)\\A(m)\end{Bmatrix}\sin mj\Delta\lambda]\, m_{ij} \tag{2.61}$$

so they are both <u>of the quadratures kind</u>, as anticipated in the preceeding paragraph.

2.11 <u>Setting up and Inverting the Covariance Matrix</u>

Each block C^{ip} of $(C_{zz} + D)$ is wholly determined by the 1st $\frac{1}{2}Nl + 1$ elements in its first row; if the number of operations required to compute any element of C^{ip} is k, then only $(\frac{1}{2}Nl + 1)\,k$ operations are needed per block, instead of Nl^2k, as would be the case if C^{ip} did not have the Toeplitz structure described previously. This is a reduction of the number of operations by a factor of 2Nl, and clearly applies not only to C^{ip} but to the whole covariance matrix as well. So $(C_{zz} + D)$ can be set up about 2Nl times faster than an ordinary matrix of the same size.

The total number of elements to be computed is $\frac{1}{2}Nl \times Nr^2$, or N^3 in the case of equal angular grids. If the grid is a fine one, this can still be a very large number of covariances. This is particularly serious in the case of <u>area means</u>, because the area mean covariances are given by expressions of the form

$$\operatorname{cov}(\bar{u}_{ij}, \bar{u}_{pq}) = M\left\{\int_{\sigma_{ij}} u\, d\sigma \int_{\sigma_{pq}} u\, d\sigma\right\} = \int_{\sigma_{ij}} \int_{\sigma_{pq}} M\left\{u(\theta,\lambda)\, u(\theta',\lambda')\right\} d\sigma\, d\sigma'$$

$$= \int_{\sigma_{ij}} \int_{\sigma_{pq}} \operatorname{cov}(u(\theta,\lambda),\, u(\theta',\lambda'))\; d\sigma\, d\sigma' \qquad (2.62)$$

involving double area integrals of the covariance function. Numerical quadratures methods, such as the one described in paragraph (4.3), have been used in the past to obtain $\operatorname{cov}(\bar{u}_{ij}, \bar{u}_{pq})$ (see, for example, Rapp (1977)). These methods take so much time in the case of fine equal angular grids for instance, that it may be practically impossible to use them to set up the covariance matrix of a global data set, in spite of the reduction by 2Nl in the number of operations. Fortunately, the coefficients a_m^{ip} in the Fourier expansion of the elements

$$c_{jq}^{ip} = \operatorname{cov}(\bar{u}_{ij}, \bar{u}_{pq}) + E\left\{n_{ij}\, n_{pq}\right\}$$

(expression (2.53)) can be obtained by means of a series expansion (truncated to a conveniently high degree N_{max}) according to expression (4.14) in paragraph (4.1). These coefficients are

$$a_m^{ip} = \left[\sum_{h=0}^{K} \sum_{n=m}^{N_{max}} I_{n,2Nh+m,i}\; I_{n,2Nh+m,p} + \sum_{h=1}^{K} \sum_{n=m}^{N_{max}} I_{n,2Nh-m,i}\; I_{n,2Nh-m,p}\right] F(m)$$

where

$$I_{n,z,i} = \int_{\sigma_i}^{\sigma_i+\Delta\theta} \bar{P}_{nz}(\cos\theta)\, \sin\theta\, d\theta\, \frac{\sigma_n}{\sqrt{2n+1}}\; \frac{1}{\Delta\lambda(\cos\theta_i - \cos(\theta_i+\Delta\theta))} \qquad (2.63)$$

$$F(m) = \begin{cases} \Delta\lambda^2 & \text{if } m = 0 \\ (2m^{-2})\;(1 - \cos m\Delta\lambda) & \end{cases} \qquad \text{and } (2h+1)\, N \leq N_{max}$$

A similar reasoning to that for area means leads to an analogous formula for point values:

$$a_m^{ip} = \left[\sum_{h=0}^{K} \sum_{n=m}^{N_{max}} (\bar{P}_{n\,2Nh+m}(\cos\theta_i)\, \bar{P}_{n\,2Nh+m}(\cos\theta_p)) + \sum_{h=0}^{K} \sum_{n=m}^{N_{max}} (\bar{P}_{n\,2Nh-m}(\cos\theta_i)\,\cdot \right.$$

$$\left. \cdot\, \bar{P}_{n\,2Nh-m}(\cos\theta_p))\right] \frac{\sigma_n^2}{2n+1} \qquad (2.64)$$

The importance of (2.63) and (2.64) is that, if the signal and noise are such that the number of terms in the summations is not too large (N_{max} is a "manageable" number), they allow the direct determination of the elements of the R(m) matrices according to (2.54). In this way, the R(m) can be created without first having to set up the whole covariance matrix and then to obtain the discrete Fourier transform of the first row of each C^{ip}. This advantage further increases in the case when the grid is symmetrical with respect to the equator, a situation that applies to all equal angular grids. Then each R(m) is persymmetrical, i.e., symmetrical with respect both to the main diagonal and the main antidiagonal, provided D is also persymmetrical (for instance, uniform noise). This means that only approximately $\frac{1}{4} N_r^2$ elements in each R(m) are different and have to be calculated individually.

Having set up the R(m) without first creating the covariance matrix, the inverse of $(C_{zz} + D)$ can be found by the equivalent operation of obtaining all $R(m)^{-1}$. The number of operations in a matrix inversion is usually O(dimension^3), or $O(N^6)$ for a covariance matrix of an equal angular data set. The number of operations per R(m) is $O(N_r^3)$, or $O(N^3)$ for the equal angular grid. In fact, as explained in (Colombo, 1979a), the inversion of a persymmetrical R(m) is equivalent to that of two matrices of half its dimension, one related to vectors of frequency m of the cosine type, and the other to vectors of the same frequency of the sine type. This further reduces calculation by a factor of $\frac{1}{4}$. With O(N) R(m) matrices to be inverted, the total comes to $O(N^4)$ operations, or $O(N^2)$ times less than for the inversion of $(C_{zz} + D)$ by ordinary techniques (Choleskii factorization, Gauss-Jordan elimination, etc.). $O(N^2)$ is also the order of the number of data points in the grid, so in the case of a $1^\circ \times 1^\circ$ equal angular grid with 64800 elements the reduction in computing time is O(64800).

The numerical examples in section 3 all involve $5^\circ \times 5^\circ$ data sets with 2592 elements, so $(C_{zz} + D)$ is of dimension 2592. Setting up and inverting such a matrix is a large exercise, even with a modern digital computer such as the AMDHAL 470 at Ohio State, unless the matrix has a strong structure that can be exploited to simplify the work. As such is indeed the case here, the subroutine NORMAL described in Appendix B has been able to do the whole setting up and inversion in only 20 seconds.

The inversion of $(C_{zz} + D)$ requires O(dimension^2) operations $(O(N^4))$ instead of O(dimension^3) because of the Toeplitz circulant structure of the C^{ip} blocks. This "O(dimension^2)" property is common to other algorithms for inverting Toeplitz-type matrices, such as the famous Trench algorithm (Trench, 1965), and the Justice algorithm (Justice, 1977), the first for data sampled on the real line and the second for data sampled on the plane, So, in spite of its "rather wicked" nature, the sphere allows this very convenient property of regular grids to apply also on its surface. In fact, not only on the sphere, but also on any body of revolution (cone, oblate and prolate spheroid, hyperboloids and paraboloids of revolution, etc.,) regular sampling and covariances that satisfy the "$|j-q|$ condition" will result in covariance matrices of the type described here, and this is also true of other matrices based on symmetrical kernels, such as the normal matrices of point mass models, when the points belong to a regular grid, etc. Finally, the optimal estimator $\hat{C}^{\alpha}_{nm} = \underline{\overset{*}{f}}^{\alpha}_{nm}\ \underline{m}$ for this type of covariance matrix is, as shown in the previous paragraph, of the quadratures type, so the optimal $\hat{C}^{\alpha}_{nm}$ can be obtained using the same efficient algorithms described in paragraphs (1.5)-(1.7). Altogether, the powerful structure of the covariance matrix for regular global data sets is most remarkable. One of its many advantageous features is that, because the creation and invertion of each R(m) can be done quite independently from those of the others, the algorithms developed for this type of matrices are eminently suited for implementation in parallel processing computers.

The separation of the algorithm according to orders also means that, although setting up and inverting all the R(m) may require a large number of operations, only a fraction of those actually correspond to the recovery of the $\overline{C}^{\alpha}_{nm}$ of any given m, so the numerical errors due to rounding or truncation are not likely to accumulate to any great extent in the results.

2.12 Optimal Formulas for non-Uniform, Correlated Noise

Irregular noise, already discussed in paragraph (2.9), may be due not only to the varying quality of the measurements, but also to the way the data is "grided", i.e., the way the value attributed to a node (or block) ij is obtained by interpolation from actual measurements nearby, as usually data is not sampled regularly on a global basis. As the number, disposition, and quality of the measurements used will vary from point to point in the grid, so will the accuracy of the interpolated values. Furthermore, even if the measurements themselves are not correlated, the grided values may be correlated because some of the data may be used for more than one interpolated value. This brings about the question of what can be done when D is neither diagonal, nor are the D^{ir} blocks in D, corresponding to the C^{ir} blocks in C_{zz}, all Toeplitz circulant. The answer is a simple extension of the results already obtained for the uncorrelated case.

When the noise is both non-stationary and correlated, replacing the covariances $E\{n_{ij}\, n_{rs}\}$ with

$$\overline{\sigma}^{ir\,|h|} = \frac{1}{4N}\left[\sum_{j=0}^{2N-1} \left(E\left\{n_{ij}\, n_{r\,j+h}\right\} + E\left\{n_{ij}\, n_{r\,j-h}\right\}\right)\right], \text{ where } h = j - s,$$

will result in a modified "noise matrix" L where the L^{ir} (corresponding to the D^{ir} and the C^{ir}) will be all Toeplitz circulant, because the "covariance" $\sigma^{ir\,|h|}$ satisfies the condition that, for a given i and r, it is a function of $|j-s|$ alone. The optimal estimator for the modified measure

$$\Phi(L) = tr[C - 2C_{sz}\, F^T + F(C_{zz} + L)\, F^T]$$

must be of the quadratures type, because of the structure of L . To show that is is also the best estimator of this kind in terms of the original norm

$$\Phi(D) = \mathrm{tr}[\, C - 2C_{cz}F^T + F(C_{zz} + D)\, F^T] ,$$

the proof will proceed much as in the case of paragraph (2.9).

The propagated error measure for $\hat{C}^{\alpha}_{nm}$ is

$$\sigma^{2\alpha}_{\eta,nm} = E\left\{ \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \chi_i^{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda\, n_{ij} \sum_{r=0}^{N-1} \sum_{s=0}^{2N-1} \chi_r^{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} ms\Delta\lambda\, n_{rs} \right\}$$
$$= \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \sum_{r=0}^{N-1} \sum_{s=0}^{2N-1} \chi_i^{nm}\chi_r^{nm} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} ms\Delta\lambda \;\; E\left\{n_{ij}\, n_{rs}\right\}$$

so, for $\hat{C}_{nm}$ and $\hat{S}_{nm}$ combined,

$$\sum_{\alpha=0}^{1} \sigma^{2\alpha}_{\eta\, nm} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \sum_{r=0}^{N-1} \sum_{s=0}^{2N-1} \chi_i^{nm}\chi_r^{nm} \cos m(j-s)\Delta\lambda \;\, E\left\{n_{ij}\, n_{rs}\right\}$$

thus

$$\Phi_{\eta}(D) = \mathrm{tr}\,[F D F^T] = \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} \sum_{r=0}^{N-1} \chi_i^{nm}\chi_r^{nm} \sum_{j=0}^{2N-1} \sum_{s=0}^{2N-1} \cos m(j-s)\Delta\lambda \;\, E\{n_{ij}\, n_{rs}\}$$
$$= \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} \sum_{r=0}^{N-1} \chi_i^{nm}\chi_r^{nm} \sum_{h=0}^{N-1} \cos mh\Delta\lambda \sum_{j=0}^{2N-1} \left(E\left\{n_{ij}\, n_{r\, j+h}\right\} + E\left\{n_{ij}\, n_{r\, j-h}\right\}\right)$$

Replacing both $E\left\{n_{ij}\, n_{r\, j+h}\right\}$ and $E\left\{n_{ij}\, n_{r\, j-h}\right\}$ in the last expression with $\bar{\sigma}^{ir|h|}$ is the same as replacing D with L , so

$$\Phi_{\eta}(L) = \mathrm{tr}\,[F L F^T] = \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} \sum_{r=0}^{N-1} \chi_i^{nm}\chi_r^{nm} \sum_{h=0}^{N-1} \cos mh\Delta\lambda \sum_{j=0}^{2N-1} 2\bar{\sigma}^{ir|h|}$$
$$= \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} \sum_{r=0}^{N-1} \chi_i^{nm}\chi_r^{nm} \sum_{h=0}^{N-1} \cos mh\Delta\lambda \;\; 4N\bar{\sigma}^{ir|h|}$$
$$= \sum_{n=0}^{N-1} \sum_{m=0}^{n} \sum_{i=0}^{N-1} \sum_{r=0}^{N-1} \chi_i^{nm}\chi_r^{nm} \sum_{h=0}^{N-1} \cos mh\Delta\lambda \sum_{j=0}^{2N-1} \left(E\left\{n_{ij}\, n_{r\, j+h}\right\} + E\left\{n_{ij}\, n_{r\, j-h}\right\}\right)$$

because of the definition of $\bar{\sigma}^{ir|h|}$. Comparing the expressions for $\Phi_{\eta}(L)$ and for $\Phi_{\eta}(D)$ it is clear that they are identical. From this follows that the modified error measure $\Phi(L)$ coincides with $\Phi(D)$ when the estimator is the optimal estimator of the quadratures type for $\Phi(L)$, and that this must be the optimal estimator of the quadratures type for $\Phi(D)$ as well. The other conclusions arrived at in paragraph (2.9) for the uncorrelated case apply equally well here.

2.13 Least Squares Adjustment, and Least Squares Collocation

(a) Band-Limited Signal

If there is a degree N_{max} above which the degree variances σ_n^2 are all negligible or zero, then the signal can be said to be band limited, and the data will satisfy equations of the type

$$m_{ij} = \sum_{n=0}^{N_{max}} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha} \bar{Y}_{nm}^{\alpha} (\theta_i, \lambda_j) + n_{ij} \qquad (2.65)$$

(the treatment here is for point values; the extension to area means is trivial) With one equation such as (2.65) per point in the grid, the result is a system of equations

$$\underline{m} + \underline{v} = A\underline{c} \qquad (\underline{v} = -\underline{n}) \qquad (2.66)$$

where A is a $N_p \times N_c$ matrix (N_p is the number of points in the grid, N_c the number of coefficients). The columns of A consist of elements of the type

$$a_{ij}^{nm} = \bar{Y}_{nm}(\theta_i, \lambda_j) \qquad (2.67)$$

According to the discussion in paragraph (1.3), if the grid is equal angular, A has full rank when $N_{max} < N$. $N_c = N^2$ and the upper limit in the summations is $N - 1$ in what follows.

Least squares adjustment is a method for solving for the $\bar{C}_{nm}^{\alpha}$ while minimizing the propagated noise defined in paragraph (2.4). The least squares solution is

$$\begin{aligned} \hat{\underline{c}} &= (A^T D^{-1} A)^{-1} A^T D^{-1} \underline{m} \\ &= G^{-1} A^T D^{-1} \underline{m} \end{aligned} \qquad (2.68)$$

where

$$G = A^T D^{-1} A \qquad (2.69)$$

is the $Nc \times Nc$ normal matrix, while $D = E\{\underline{n}\, \underline{n}^T\}$ is the same noise matrix considered before. Clearly, the least squares estimator matrix is

$$F_{\ell s} = (A^T D^{-1} A)\, A^T D^{-1}$$

When the noise has zero mean ($E\{\underline{n}\} = \underline{o}$), the estimator of (2.68) is the best linear unbiased estimator, because it minimizes

$$\mathrm{tr}\,[E\{F\,\underline{n}\,\underline{n}^T F^T\}] = \mathrm{tr}\,[F D F^T]$$

and $E\{F(\underline{z} + \underline{n})\} = \underline{c}$. If, in addition to all this, the probability distribution of the noise is Gaussian, then (2.68) corresponds to the maximum likelihood estimator as well. In many scientific applications the noise has approximately zero mean and near-Gaussian distribution, while D is known reasonably well;

for this reason, methods based on expression (2.68) are used quite often. The linearity of the resulting estimators is helpful, because this avoids the use of methods based on non-linear formulas that are usually difficult both from a theoretical and from a practical point of view. The variances of the estimates are given by the corresponding diagonal elements of the a posteriori variance-covariance matrix

$$E_{\ell_s} = (A^T D^{-1} A)^{-1} = G^{-1}$$

Therefore, to obtain both the estimates and their variances it is necessary to know G^{-1} . Sometimes, because of the nature of A and D , G can be seriously ill-conditioned, the inversion suffering from strong numerical instabilities. To reduce this problem, a simple device known as regularization is often used (see, for instance, Tikhonov and Arsenin, 1977). Generally speaking, regularization is the introduction of a slight change in a problem, so the solution virtually remains the same, but the modified problem has better numerical properties. In least squares methods regularization usually implies adding a small positive definite matrix K(diagonal, as a rule) to G before attempting to invert it. The regularized optimal estimator would be

$$\hat{\underline{c}} = (A^T D^{-1} A + K) A^T D^{-1} \underline{m} \tag{2.70}$$

The inverse of the covariance matrix of the harmonic coefficients C is a positive, diagonal matrix which could be used to regularize the normal matrix:

$$\hat{\underline{c}} = (A^T D^{-1} A + C^{-1}) A^T D^{-1} \underline{m} \tag{2.71}$$

It is easy to see that this expression minimizes the quadratic form

$$Q = \underline{c}^T C^{-1} \underline{c} + \underline{v}^T D^{-1} \underline{v} \tag{2.72}$$

subject to the constraint

$$\underline{m} = A \underline{c} + \underline{v} \tag{2.73}$$

Moreover, (2.72) is the equivalent of the least squares collocation error measure when then signal is band-limited (Moritz, (1980)). This idea has been used, among others, by Schwarz (1975) for the determination of low degree zonal coefficients of the geopotential, and by Lerch et al. (1979), who employed it to estabilize the adjustment of the GEM-9 gravity field model with remarkable success. The equivalence of (2.71) to the collocation estimator is true only for band-limited signals; in the "real world" the gravity field has infinite bandwith, so (2.71) is no more than an approximation. The band-limited assumption is a reasonable one, however, as the σ_n eventually become negligible for large n.. This is particularly true at satellite altitudes; in any case, geodesy is a science of wise approximations. Moritz (1980) has provided a very clear and concise explanation of the use of collocation in general, and expression (2.71) in particular, in spherical harmonic analysis.

An alternative derivation of (2.71) follows from the matrix equation

$$C A^T (A C A^T + D)^{-1} = (A^T D^{-1} A + C^{-1})^{-1} A^T D^{-1} \tag{2.74}$$

(see, for instance, Uotila (1978), equation (29)), which is valid for symmetrical matrices, provided the inverses or pseudoinverses of D , C , and $(A C A^T + D)$ do exist. According to the definitions in paragraph (2.2):

$$C_{cz} = M\{\underline{c}\,\underline{z}^T\} = M\{\underline{c}\,\underline{c}^T A^T\} = M\{\underline{c}\,\underline{c}^T\}A^T = C A^T \tag{2.75}$$

$$C_{zz} = M\{\underline{z}\,\underline{z}^T\} = M\{A\underline{c}\,\underline{c}^T A^T\} = A M\{\underline{c}\,\underline{c}^T\} A^T = A C A^T \tag{2.76}$$

Replacing C_{cz} and C_{zz} in the expression of the collocation estimator matrix (2.39) with their equivalents given by (2.75) and (2.76):

$$\begin{aligned} F = C_{cz} (C_{zz} + D)^{-1} &= C A^T (A C A^T + D)^{-1} \\ &= (A^T D^{-1} A + C^{-1})^{-1} A^T D^{-1} \end{aligned} \tag{2.77}$$

according to equation (2.74). This shows that the "regularized" estimator matrix $(A^T D^{-1} A + C^{-1})^{-1} A^T D^{-1}$ is indeed the same as the collocation estimator matrix, so (2.71) represents an alternative form of collocation when the data is band-limited.

(b) Infinite Bandwith

In this case the "observation equations" are

$$m_{ij} = \sum_{n=0}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha}\, \bar{Y}_{nm}^{\alpha} (\theta_i , \lambda_j) + n_{ij}$$

Calling

$$w_{ij} = \sum_{n=N+1}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha}\, \bar{Y}_{nm}^{\alpha} (\theta_i , \lambda_j) \tag{2.78a}$$

and

$$\underline{w} = [w_1 \ldots w_k \ldots w_{Np}]^T \quad (k = 2N i + j) \tag{2.78b}$$

then

$$m_{ij} = \sum_{n=0}^{N} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \bar{C}_{nm}^{\alpha}\, \bar{Y}_{nm}^{\alpha} (\theta_i , \lambda_j) + w_{ij} + n_{ij} \tag{2.79}$$

and, regarding this expression as a modified observation equation, and replacing D with $D + M\{\underline{w}\,\underline{w}^T\}$ in (2.71), the linear estimator that minimizes the quadratic form

$$\tilde{Q} = \underline{c}^T C^{-1} \underline{c} + \underline{v}^T (D + M\{\underline{w}\,\underline{w}^T\})^{-1} \underline{v} \tag{2.80}$$

is

$$\hat{\underline{c}} = (A^T (D + M\{\underline{w}\,\underline{w}^T\})^{-1} A + C^{-1})^{-1} A^T (D + M\{\underline{w}\,\underline{w}^T\})^{-1} \underline{m} \tag{2.81}$$

It is easy to show, either following the lines of Moritz (1972), or going back once more to the matrix identity (2.74), that expression (2.81) is identical with the estimator of least squares collocation. Rigorously speaking, expression (2.81) should be used whenever $\sigma_n^2 \neq 0$ for $n > N$, even if $\sigma_n^2 = 0$ for $n > N_{max}$, for some finite $N_{max} > N$.

2.14 Ridge Regression and Least Squares Collocation

Consider once more the estimator

$$\underline{\hat{c}} = F_{\ell s}\, \underline{m} = (A^T D^{-1} A)^{-1} A^T D^{-1} \underline{m}$$

If there is no noise, so $\underline{m} = \underline{z} = A\,\underline{c}$, and if $n < N$, then

$$\underline{\hat{c}} = (A^T D^{-1} A)^{-1} A^T D^{-1} A\, \underline{c} = \underline{c}$$

According to the definition given in paragraph (2.4), the sampling error of this estimator is zero, so the measure of this error must be also zero. If the noise has zero mean, it follows that

$$E\left\{\underline{\hat{c}}\right\} = E\left\{F_{\ell s}\, \underline{m}\right\} = F_{\ell s}\, E\left\{A\,\underline{c} + \underline{n}\right\} = F_{\ell s}\, A\,\underline{c} + E\left\{\underline{n}\right\} = \underline{c}$$

or, as it is usually said, the estimator is unbiased. Moreover, by a simple extension of the Gauss-Markov theorem to the case of a general symmetrical positive matrix D (see, for instance, Bibby and Toutenberg, 1977), $F_{\ell s}\,\underline{m}$, of all linear unbiased estimators, has the least propagated error measure $\mathrm{tr}\left\{F_{\ell s} D F_{\ell s}^T\right\} = \mathrm{tr}\left\{(A^T D^{-1} A)\right\}$, as mentioned previously.

The estimator of expression (2.77) does not, in general, give perfect estimates of $\underline{c}$ in the absence of noise: it is a biased estimator, and the measure of the bias is $\mathrm{tr}\left\{C - 2C_{cz}F^T + FC_{zz}F^T\right\}$ (this term[1] can no longer be regarded as the measure of the sampling error, as it is the presence of C^{-1} inside the parenthesis in (2.71) and not the sampling that brings about this error). According to (2.39), F is the estimator that minimizes the total error measure, so

$$\begin{aligned}
\sigma_\epsilon^2 &= \mathrm{tr}\left\{C - 2\,C_{cz}\,F^T + F(C_{zz} + D)\,F^T\right\} \\
&\leq \mathrm{tr}\left\{C - 2C_{cz}\,F_{\ell s}^T + F_{\ell s}(C_{zz} + D)\,F_{\ell s}^T\right\} \\
&= \mathrm{tr}\left\{F_{\ell s}\,D\,F_{\ell s}^T\right\}
\end{aligned}$$

If the covariance matrix is positive definite (which only requires that all $\sigma_n^2 \neq 0$ for $0 \leq n < N$) then the last expression applies with strict inequality

$$\begin{aligned}
\mathrm{tr}\left\{C - 2C_{cz}\,F^T + F(C_{zz} + D)\,F^T\right\} &< \mathrm{tr}\left\{F_{\ell s}\,D\,F_{\ell s}^T\right\} \\
&= \mathrm{tr}\left\{(A^T D^{-1} A)^{-1}\right\}
\end{aligned}$$

1 In the band-limited case.

Some may find this result rather surprising: the best estimator with zero bias is in fact worse than the biased estimator of (2.71)! The difficulty is only apparent: $F_{\ell s}$ is the best estimator with no bias; once the "no bias" condition is removed, the expression above merely indicates that there is an estimator in the larger class of the estimators that have a bias (including those with zero bias) such that the sum of the bias and the propagated noise is smaller. From this, it is clear that

$$\operatorname{tr}\left\{F D F^{T}\right\} < \operatorname{tr}\left\{(A^{T} D^{-1} A)\right\} \quad (\text{as } \operatorname{tr}\left\{C - 2C_{cz}F^{T} + F C_{zz} F^{T}\right\} \geq 0\ )$$

which is indeed possible when the condition $C - 2\,C_{cz}F^{T} + F\ C_{zz}F^{T} = 0$ is removed. In fact, there is nothing very new about all this: the use of biased estimators to obtain estimates with small variances is a reasonably well-established practice in applied statistics. In particular, the technique known as ridge regression consists in using the biased estimator

$$\hat{\underline{c}} = (\chi^{T}\chi + K)^{-1}\chi^{T}\underline{m}$$

with a suitable choice of K (Bibby, 1972). Clearly, this expression is the same as (2.71) when $\chi = A$, $D = I$, and $K = C^{-1}$. With some obvious modifications suggested by (2.81), this argument can be extended to the estimation of $\underline{c}$ when $n \geq N$, so it can be said that within the scope of spherical harmonic analysis least squares collocation is a form of ridge regression.

This brings up the question of just how realistic the error measure is; after all the best of all possible estimators in terms of a given norm could be a very bad one for some specific problem where that norm is not suitable. To answer this question, one must start by defining the meaning of "realistic". If one is interested in minimizing the actual error variance of the coefficients per degree, i.e., the expression

$$\delta_n^2 = \sum_{\alpha=0}^{1}\sum_{m=0}^{n}(\hat{C}_{nm}^{\alpha} - \overline{C}_{nm}^{\alpha})^2\,(2n+1)^{-1}$$

which may be of interest because this corresponds to, say, the global mean square of the error of representing the continuous function $(\theta,\lambda) = \sum_{m=0}^{n}\sum_{\alpha=0}\overline{C}_{nm}^{\alpha}\ \overline{Y}_{nm}(\theta,\lambda)$ with $f(\theta,\lambda) = \sum_{m=0}^{n}\sum_{\alpha=0}\hat{C}_{nm}^{\alpha}\,\overline{Y}_{nm}(\theta,\lambda)$, according to Parseval's identity (exp. (1.12)), then one could say that a realistic measure is one that gives close estimates of the actual error variances. The "actual error" measure δ_n^2 corresponds to one of infinitely many "events" over which the collocation measure is an average. The proof of a pudding being in the eating, the reader can judge just how realistic the collocation measure is by looking at the numerical results in section 3, where the value of the measure turns out to differ only by a small percentage from the actual variance in each one of a number of simulated "events", i.e., the recovery of the $\bar{C}^{\alpha}_{nm}$ from "simulated data", where the $\bar{C}^{\alpha}_{nm}$ are known random numbers scaled to have the desired power spectrum.

2.15 Structure of the Normal Matrix

The elements of the normal matrix G (in the case of point data) are of the form

$$g^{\alpha,\beta}_{nm,pq} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \bar{Y}^{\alpha}_{nm}(\theta_i, \lambda_j)\, \bar{Y}^{\beta}_{pq}(\theta_i,\lambda_j)\, \sigma_i^{-2}$$
$$= \sum_{i=0}^{N-1} \bar{P}_{nm}(\cos\theta_i)\, \bar{P}_{pq}(\cos\theta_i) \sum_{j=0}^{2N-1} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} qj\Delta\lambda\, \sigma_i^{-2} \tag{2.82}$$

where $\sigma_i^2 = \sigma_{ij}^2$ for all $0 \le j \le 2N$ (i.e., "regular" noise as defined in paragraph 2.8). If $n, m < N$, then the following equations apply:

$$\sum_{j=0}^{2N-1} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} qj\Delta\lambda = 0 \quad \text{if} \begin{cases}\alpha \neq \beta\\ \text{or}\\ m \neq q\end{cases} \tag{2.83}$$

Moreover, if the grid is symmetrical with respect to the Equator (as equal angular grids are), and if $\sigma_i^2 = \sigma^2_{N-1-i}$ (for instance, if σ_i^2 is independent also of i) the relationships

$$\sigma_i^{-2}\, \bar{P}_{nm}(\cos\theta_i) = \bar{P}_{nm}(\cos\theta_i')\,(-1)^{n-m}\,\sigma^{-2}_{N-1-i}$$

$$\sigma_i^{-2}\, \bar{P}_{pm}(\cos\theta_i) = \bar{P}_{pm}(\cos\theta_i')\,(-1)^{p-q}\,\sigma^{-2}_{N-1-i}$$

$$\theta_i' = \pi - \theta_i$$

must apply, according to par. (1.2), and from these follows

$$\sum_{i=0}^{N-1} \sigma_i^{-2}\, \bar{P}_{nm}(\cos\theta_i)\, \bar{P}_{pm}(\cos\theta_i) = 0 \quad \text{if } n-p \text{ is odd.} \tag{2.84}$$

In brief: if $n, m < N$, and the grid is regularly spaced in longitude, and symmetrical with respect to the equator, then

$$g^{\alpha,\beta}_{nm,pq} = 0 \quad \text{if} \begin{cases}\alpha \neq \beta,\ m \neq q\\ \text{or}\\ n-p \text{ is odd}\end{cases} \tag{2.85}$$

If neither of the conditions listed above apply, then $g^{\alpha,\beta}_{nm,pq}$ may or may not be zero. If the coefficients $\bar{C}^{\alpha}_{nm}$ are ordered in $\underline{c}$ so that all those of the same m are grouped together, and for a given m all $\bar{C}^{1}_{nm}$ are separated from all $\bar{C}^{0}_{nm}$, and furthermore all $\bar{C}^{\alpha}_{nm}$ with n - m even are separated from all those with n - m odd, then the normal matrix becomes arranged in such a way that all potentially non-zero elements (i.e., not satisfying (2.85)) are also grouped together forming a series of diagonal blocks $G^{\alpha,\delta}_{m}$, where δ signals the parity of n-p. Each one of these diagonal blocks is made exclusively of one of the following types of elements

$$\begin{matrix} g^{0,0}_{nm,pm} \\ g^{1,1}_{nm,pm} \end{matrix} \left\{ \begin{matrix} n-p \ \text{even} \ (\delta=0) \\ n-p \ \text{odd} \ (\delta=1) \end{matrix} \right.$$

If the grid is not symmetrical with respect to the equator, groups of type $\binom{\alpha=0}{\delta=0}$ and $\binom{\alpha=0}{\delta=1}$, or $\binom{\alpha=1}{\delta=0}$ and $\binom{\alpha=1}{\delta=1}$, become included into larger non-zero blocks, as there are more non-zero elements in that case. However, here the discussion will cover only the symmetrical case.

The largest blocks are $G^{0,\delta}_{0}$, and their dimension is N^2; the smallest blocks have dimension 1 x 1 , for example $G^{\alpha,\delta}_{N-1}$.

The inverse of any block-diagonal matrix such as G is another block-diagonal matrix made up of the inverse of the blocks of G. There are 4 N - 2 diagonal blocks $G^{\alpha,\delta}_{n}$ in G, and as many in G^{-1}.

The eigenvalues of G are those of the diagonal blocks, and the eigenvectors of G, those of the same blocks "expanded" with zeroes at both ends, so as to reach the dimension N_p of G.

The estimates' vector

$$\hat{\underline{c}} = G^{-1} A^{T} D^{-1} \underline{m}$$

can be partitioned in the same way as $\underline{c}$, each partition $\hat{\underline{c}}^{\alpha,\delta}_{m}$ including all coefficients' estimates of the same m, α, and parity δ of n - m :

$$\hat{\underline{c}}^{\alpha,\delta}_{m} = (G^{\alpha,\delta}_{m})^{-1} A^{\alpha,\delta}_{m} D^{-1} \underline{m} \tag{2.86}$$

where $A^{\alpha,\delta}_{m}$ is a (N - m) x N_p matrix with rows that are N_p - vectors of the type

$$\underline{a}^{\alpha,\delta}_{m,n} = [\ldots \bar{P}_{nm}(\cos\theta_i) \left\{ \begin{matrix} \cos \\ \sin \end{matrix} \right\} mj\Delta\lambda \ldots] \tag{2.87}$$

(n - m even if $\delta = 0$, odd if $\delta = 1$) So the rows of $(G^{\alpha,\delta}_{m})^{-1} A^{\alpha,\delta}_{m}$ are linear combinations of vectors of the same frequency m and, therefore, also vectors of the same frequency:

$$\underline{h}^{\alpha}_{nm} = [\ldots \chi^{nm}_{i} \left\{ \begin{matrix} \cos \\ \sin \end{matrix} \right\} mj\Delta\lambda \ldots] \tag{2.88}$$

Consequently, the estimate of a given $\bar{C}^{\alpha}_{nm}$ is

$$\hat{C}^{\alpha}_{nm} = \underline{h}^{\alpha}_{nm} D^{-1} \underline{m} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \chi^{nm}_{i} \left\{ \begin{matrix} \cos \\ \sin \end{matrix} \right\} mj\Delta\lambda\, \sigma_i^{-2} m_{ij} \tag{2.89}$$

and this is a quadratures type estimator.

Setting-up and inverting the $G_m^{\alpha,\delta}$ blocks is tantamount to setting up and inverting the whole matrix G . Since all operations related to one of the blocks are independent from those for the others, the inversion of G is ideally suited for parallel-processing computing. On average, each $G_m^{\alpha,\delta}$ requires $O(N^3)$ operations to invert, or $O(N^4)$ altogether. Inverting G by ordinary techniques would involve $O(((N)^2)^3) = O(N^6)$, so there is an increase in efficiency of $O(N^2)$.

Finally, it is quite simple to show that these properties carry over both to the case of area means, and to problems where the surface being studied is not a sphere, but a surface of revolution symmetrical about a plane perpendicular to the axis of rotation, provided that the longitude increments be constant and the grid symmetrical with respect to the "equator". In this latter case, the expansion of the signal in solid spherical harmonics is

$$z_{ij} = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \sum_{\alpha=0}^{1} \frac{a^n}{r_{ij}^{n+1}} \bar{Y}_{nm}^{\alpha} (\theta_i, \lambda_j)$$

and the factors $\frac{a^n}{r_{ij}^{n+1}}$ are symmetrical about the equator, from which all the properties already mentioned for G follow.

Clearly, the structure of G possesses many properties similar or identical to those of $(C_{zz} + D)$ when the data is regularly sampled on a surface of revolution. These similitudes underline the intimate relationship between least squares adjustment and least squares collocation shown in the preceeding paragraph. In fact, as least squares, regularized least squares, and collocation differ only in the diagonal matrix (K , or C^{-1}) being added to G in expressions (2.68), (2.70), and (2.77), all the properties mentioned here for G apply to the normal matrices in each of the three methods equally well. The one important consideration, in the case of collocation, is that the data be band-limited. Otherwise, expression (2.81) indicates that $(C + M\{\underline{w}\,\underline{w}^T\})^{-1}$ and not C^{-1} must be added to G . Matrix $C + M\{\underline{w}\,\underline{w}^T\}$ has the same Toeplitz-type structure of $(C_{zz} + D)$ discussed in paragraph (2.10). Therefore, creating and inverting the normal matrix requires: (a) creating and inverting $C + M\{\underline{w}\,\underline{w}^T\}$, and (b) creating and inverting $A^T D^{-1} A + (M\{\underline{w}\,\underline{w}^T\} + C)^{-1}$, which can be shown to have the same block diagonal structure discussed here. This is twice the work needed to set up and invert $(C_{zz} + D)$ using the approach of paragraph (2.11), so, in the case of infinite bandwith, that approach is more economical in computing and, therefore, more practical.

There may be one important point in favor of using formula (2.77) or (2.81) rather than formula (2.39) for obtaining the optimal estimator matrix F at least in the band-limited case: as the density of the grid increases, matrix $(C_{zz} + D)$ becomes increasingly more ill-conditioned, because the closer distance between data points results in covariances that have much the same values in consecutive rows or columns. On the other hand, the non-zero diagonal blocks in G are likely to become more and more diagonal-dominant as $\Delta\theta, \Delta\lambda \to 0$. This will depend on D : for instance, if the variances of the noise were of the form

$$\sigma_{ij}^2 = \sin\theta_i$$

then

$$\lim_{\substack{\Delta\theta,\Delta\lambda \\ \to 0\ \to 0}} \frac{\Delta\theta\,\Delta\lambda}{4\pi}\, g^{\alpha,\delta}_{nm,pm} = \lim_{\substack{\Delta\theta,\Delta\lambda \\ \to 0\ \to 0}} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{Y}^{\alpha}_{nm}(\theta_i,\lambda_j)\, \overline{Y}^{\alpha}_{pm}(\theta_i,\lambda_j)\, \sin\theta_i\, \frac{\Delta\theta\,\Delta\lambda}{4\pi}$$

$$= \frac{1}{4\pi}\int_{\sigma} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\; \overline{Y}^{\alpha}_{pm}(\theta,\lambda)\, d\sigma = \begin{cases} 0 & \text{if } n \neq p \\ 1 & \text{if } n = p \end{cases}$$

because of the orthogonality relationships. In general, the variances of the noise are not going to follow a sinusoidal law, but one may reasonably expect (at least with more or less homogeneous noise) that the stability of the normal equations will not deteriorate with $\Delta\theta, \Delta\lambda \to 0$.

2.16 Global Adjustment and Collocation with Scattered Data

The efficient set up and inversion of the covariance matrix $(C_{zz} + D)$, or of the normal matrix G, depend on the regular nature of the grid. If not all nodes or blocks in the grid have data associated with them, the data is said to be scattered. The blanks or "holes" in the grid destroy the orderly structure of the matrices, making the application of the techniques previously discussed impossible. Yet so strong is this structure that, even in fragments, still it can be dealt with more efficiently than in the case of ordinary matrices of the same size.

(a) Full Region Bound by Lines of Latitude and Longitude

In the case when there is data at every point or block inside a "square" region limited by parallels and meridians, the partitioning of the data vector along the arcs or parallels inside the zone reveals a strong structure in the $(C_{zz} + D)$ matrix, if all the other assumptions made in paragraph (2.10) still apply.

If N_r is the number of rows and N_c the number of meridians that cross the region, then the covariance matrix will consist of N_r^2 blocks C^{ip} of dimension N_c, both persymmetrical and Toeplitz, though not circulant (i.e., the relationship $c^{ip}_{jq} = c^{ip}_{j+1\,q+1}$ is fulfilled, but not $c^{ip}_{0q} = c^{ip}_{N_c\,q-1}$); moreover the first row in each block does not have the property that $c^{ip}_{0q} = c^{ir}_{0\,N_c-q}$. Clearly, though weaker than in the case of a global grid, there is a definite structure here that can be exploited to make both setting up and inverting the matrix more efficient.

Because each block C^{ip} is Toeplitz, only the N_c elements in its first row have to be computed, or about $\frac{1}{2}N_c\,N_r^2$ for the matrix as a whole, instead of $\frac{1}{2}N_c^2 N_r^2$; this amounts to a reduction in operations by a factor of N_c.

The solution of the equation $\overset{*}{\underline{f}}{}^{\alpha}_{nm} = (C_{zz} + D)^{-1}\, \underline{c}^{T}_{nm\alpha,z}$, for the optimal estimator vector $\overset{*}{\underline{f}}{}^{\alpha}_{nm}$ for $\bar{C}^{\alpha}_{nm}$, can be obtained by a technique such as conjugate gradients, or similar, in which a finite number K of matrix-vector multiplications $(C_{zz} + D)\,\underline{v}_t$ (where $\underline{v}_0, \underline{v}_1, \ldots \underline{v}_t, \ldots \underline{v}_k$ are K intermediate N_p - vectors created during the solution) constitute the bulk of the computing effort. There is no need to go into the details of any specific technique, as the reader will find excellent descriptions in the literature (Householder, 1964, Luenberger, 1969). A discussion of the matrix-vector operation is sufficient here.

Let $\underline{m}^{i}$ be the N_c - vector partition of the $N_c\,N_r$ data vector $\underline{m}$, containing the measurements along the ith parallel in the region, and let $\underline{v}_t^{i}$ be the corresponding partition in any of the $\underline{v}_t$ vectors. The product $(C_{zz} + D)\underline{v}_t = \underline{p}_t$ is, under such partition,

$$\underline{p}_t = [\underline{p}_t^{0} \ldots \underline{p}_t^{N_r-1}]^{T}$$

with

$$\underline{p}_t^{i} = \sum_{p=0}^{N_r-1} C^{ip}\, \underline{v}_t^{p} \tag{2.90}$$

so the whole matrix-vector multiplication can be broken up into N_r^2 products $C^{ip}\,\underline{v}_t^{p} = \underline{h}_t^{ip}$. Because C^{ip} is a Toeplitz matrix, the N_c components of $\underline{h}_t^{ip}$ can be obtained by "weighted running averages" or <u>discrete convolution</u> of the elements of $\underline{v}_t^{ip}$ with those in the first row of C^{ip}. Such convolution can be calculated efficiently using the Fast Fourier Transform algorithm (see, for instance, Brigham, 1974, Ch. 13). Therefore, all N_r^2 products involve $O(N_c\, N_r^2)$ operations, and since there are K matrix-vector multiplications in the whole procedure, the total number of operations needed to obtain $\overset{*}{\underline{f}}{}^{\alpha}_{nm}$ amounts to $O(KN_c\, N_r^2)$. For conjugate gradients, K does not exceed (in theory) $N_p = N_r\, N_c$, so there should be $O(N_c^2\, N_r^3)$ operations altogether. If $(C_{zz} + D)$ were handled by conventional techniques, disregarding its well defined structure, the number would be $O(N_c^3\, N_r^3)$, so the increase in efficiency is $O(N_c)$, the same as for the setting up.

(b) Arbitrarily Scattered Data

It is common in geodesy and in geophysics to have a set of measurements scattered throughout the globe, without the data being on the nodes of a regular grid or without all σ_{ij}^2 being equal along parallels (nonhomogenous noise). If the set is dense enough, however, it is possible to interpolate the data quite reliably on the closest nodes of a conveniently chosen grid. Assuming that this is done, and that the accuracies of the interpolated values are known well enough, then the problem can be dealt with by conventional least squares or by collocation. In general, there will be blanks or "holes" irregularly distributed over the sphere, the actual data points falling among them in no precise pattern. This problem will be considered here as a least squares adjustment problem. If the data has little or no power above the Nyquist frequency of the grid on which it has been interpolated, then the extension of the ideas that follow to collocation is quite simple, according to paragraph (2.13).

Consider the element $g^{\alpha\beta}_{nmpq}$ of the matrix $G = A^T D^{-1} A$

$$g^{\alpha,\beta}_{nm,pq} = \sum_{i=0}^{N-1} \overline{P}_{nm}(\cos\theta_i)\, \overline{P}_{pq}(\cos\theta_i) \sum_{j=0}^{2N-1} \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} mj\Delta\lambda \begin{Bmatrix}\cos\\ \sin\end{Bmatrix} pj\Delta\lambda W_{ij}\, \sigma^{-2}_{ij} \tag{2.91}$$

where $W_{ij} = 0$ if the point ij is blank, otherwise $W_{ij} = 1$; σ^2_{ij} is the variance of the noise n_{ij} . In general σ^2_{ij} is a function both of i and of j . Clearly, matrix D is taken to be diagonal (i.e., uncorrelated noise).

From the relationships

$$\begin{aligned}
\cos mj\Delta\lambda\ \cos pj\Delta\lambda &= \tfrac{1}{2}[\cos(m+p)j\Delta\lambda + \cos(m-p)j\Delta\lambda] \\
\cos mj\Delta\lambda\ \sin pj\Delta\lambda &= \tfrac{1}{2}[\sin(m+p)j\Delta\lambda - \sin(m-p)j\Delta\lambda] \\
\sin mj\Delta\lambda\ \sin pj\Delta\lambda &= -\tfrac{1}{2}[\cos(m+p)j\Delta\lambda - \cos(m-p)j\Delta\lambda]
\end{aligned}$$

and calling

$$C_r^{i\alpha} \equiv \begin{cases} C_r^{i0} = \frac{1}{2}\sum_{j=0}^{2N-1} \cos rj\Delta\lambda\, \sigma^{-2}_{ij}\, W_{ij} \\ C_r^{i1} = \frac{1}{2}\sum_{j=0}^{2N-1} \sin rj\Delta\lambda\, \sigma^{-2}_{ij}\, W_{ij} \end{cases} \tag{2.92}$$

where $-N < r < 2N$, follows

$$\sum_{j=0}^{2N-1} \sigma^{-2}_{ij} W_{ij} \underset{(\alpha)}{\begin{Bmatrix}\cos\\ \sin\end{Bmatrix}} mj\Delta\lambda \underset{(\beta)}{\begin{Bmatrix}\cos\\ \sin\end{Bmatrix}} pj\Delta\lambda = [C^{i\,|\alpha-\beta|}_{(m-q)} + (-1)^{2|\alpha+\beta|} C^{i\,|\alpha-\beta|}_{(m-q)}] \tag{2.93}$$

Moreover, because

$$\begin{aligned}
\cos(-r)j\Delta\lambda &= \cos rj\Delta\lambda \\
\sin(-r)j\Delta\lambda &= -\sin rj\Delta\lambda \\
\cos rj\Delta\lambda &= \cos(2N-r)j\Delta\lambda \\
\sin rj\Delta\lambda &= -\sin(2N-r)j\Delta\lambda
\end{aligned} \qquad (2N\Delta\lambda = 2\pi)$$

follows

$$C^{i\alpha}_{-r} = C^{i\alpha}_r\,(-1)^\alpha$$

$$C^{i\alpha}_{2N-r} = C^{i\alpha}_r\,(-1)^\alpha$$

where the last one is of special interest when $r \geq N$. So only the $C_r^{i\alpha}$ with $0 < r < N$ are needed. Finally, calling

$$g^{i\alpha,\beta}_{nm,pq} = \bar{P}_{nm}(\cos\theta_i)\,\bar{P}_{pq}(\cos\theta_i)\,[C^{i\,|\alpha-\beta|}_{m-q} + (-1)^{2\alpha+\beta}\,C^{i\,|\alpha-\beta|}_{(m+q)}] \quad (2.94)$$

(where $|\alpha-\beta|$ is the absolute value of $\alpha-\beta$)

it is

$$g^{\alpha,\beta}_{nm,pq} = \sum_{i=0}^{N-1} g^{i\alpha,\beta}_{nm,pq} \quad (2.95)$$

Assume that out of N rows the grid has only I with any data in them, so at least one $W_{ij} \neq 0$ in each. Once the corresponding $C_r^{i\alpha}$ have been obtained by computing the discrete Fourier transform of $\sigma_{ij}^{-2} W_{ij}$ along each row with data ($O(N^2)$ operations for the whole grid), what remains is to get the I non-zero terms $g^{i\alpha,\beta}_{nm,pq}$ that form each element $g^{\alpha,\beta}_{nm,pq}$ of G. As there are about $\frac{1}{2}N^4$ such elements that are different, and I is $O(N)$ (except for very sparse data sets), the total number of operations needed to create G is $O(N^2) + O(N^5)$, or virtually $O(N^5)$. If the $g^{\alpha,\beta}_{nm,pq}$ were computed according to (2.91) as it is written, instead of according to its reduced version. (2.95), the number of operations would be $O(2N^2\,(N)^4)$, or $O(N^6)$, so the gain in efficiency allowed by this approach is $O(N)$, which is the same as in the case studied in the first part of this paragraph, where the data completely filled a "square" sector of the sphere. This count does not include the time needed to obtain the $\bar{P}_{nm}(\cos\theta_i)$, as these can be pre-computed once and kept on disk or tape for repeated use.

The number of operations can be reduced further, almost by half, by taking advantage of the fact that $\bar{P}_{nm}(\cos\theta_i)$ is a common factor in all $g^{i\alpha,\beta}_{nm,pq}$ with the same n and the same m. Furthermore, if there are pairs of rows i and $N-i-1$ (i.e., symmetrical with respect to the Equator) where both rows contain some data, then

$$g^{i\,\alpha,\beta}_{nm,pq} + g^{N-i-1\,\alpha,\beta}_{nm,pq} = \bar{P}_{nm}(\cos\theta_i)\,\bar{P}_{pq}(\cos\theta_i)$$
$$[(C^{i\,|\alpha-\beta|}_{m-q} + (-1)^{n-p}C^{N-i-1\,|\alpha-\beta|}_{m-q}) + (-1)^{2\alpha+\beta}(C^{N-i-1\,|\alpha-\beta|}_{m+q} + (-1)^{n-p}\,C^{i\,|\alpha-\beta|}_{m+q})] \quad (2.96)$$

which leads to further savings in computing at the cost of additional programming complexity. These economies are important, but they will not bring the number of operations much below $O(N^5)$ unless the grid is so sparse that I is much smaller than N.

Notice that the normal matrix $G = A^T D^{-1} A$ is created here without actually forming the observation equations matrix A. This means considerable savings in computing and in storage requirements. As for the right hand side of the normals $A^T D^{-1}\,\underline{m} = \underline{b}$, the elements of $\underline{b}$ are given by the formula

$$b_k = \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \bar{P}_{nm} (\cos \theta_i) \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda \, \sigma_{ij}^{-2} \, W_{ij} \, m_{ij} \tag{2.97}$$

where $k = n^2 + \alpha n + m + 1$ and $W_{ij} = 0$ if there is no data at the point ij, as before. Such expression is of the quadratures type, and can be computed efficiently by the corresponding algorithm of section 1, also without first creating A. Finally, the residuals vector $\underline{v} = \underline{m} - A\hat{\underline{c}}$, usually of interest, can be obtained as the difference between the data $\underline{m}$ and the values of

$$\hat{z}(\theta_i, \lambda_j) = \sum_{\alpha=0}^{1} \sum_{n=0}^{1} \sum_{m=0}^{n} \hat{C}_{nm}^{\alpha} \, \bar{P}_{nm} (\cos \theta_i) \begin{Bmatrix} \cos \\ \sin \end{Bmatrix} mj\Delta\lambda$$

computed by means of the appropriate synthesis procedure given in section 1. Thus $\underline{v}$ can be found without knowing A explicitly.

The normal equations can be solved by means of conjugate gradients or a similar method involving M matrix-vector products $\underline{x}_t = G\underline{h}_t$, where $\underline{h}_t$ is part of a sequence of intermediate vectors $\underline{h}_0, \underline{h}_1, \ldots \underline{h}_t, \ldots \underline{h}_M$.

Introducing the notation

$$\underline{h}_{nmt}^{\alpha} = \underline{h}_{kt}$$

where $k = n^2 + \alpha n + m + 1$, and calling G^i to the matrix of all $g_{nm,pq}^{i\,\alpha,\beta}$,

so $$G = \sum_{i=0}^{N-1} G^i \text{ and } \underline{x}_t^t = \sum_{i=0}^{N-1} G^i \underline{h}_t = \sum_{i=0}^{N-1} \underline{x}_t^i \tag{2.98}$$

then each element $x_{nmt}^{\alpha\, i}$ of the product vector $\underline{x}_t^i$ is of the form

$$x_{nmt}^{\alpha i} = \sum_{p=0}^{N-1} \sum_{q=0}^{p} \sum_{\beta=0}^{1} g_{nm,pq}^{i\alpha\beta} \, h_{pqt}^{\beta} = \bar{P}_{nm}(\cos\theta_i) \sum_{pq\beta} [C_{m+q}^{i|\alpha-\beta|} + (-1)^{2\alpha+\beta} C_{m-q}^{\alpha-\beta|}] \cdot \bar{P}_{pq}(\cos\theta_i) \, h_{pqt}^{\beta} \tag{2.99}$$

Every h_{pqt}^{β} above multiplies always the same $\bar{P}_{pq}(\cos\theta_i)$; calling

$$d_{pq}^{i\beta} = \bar{P}_{pq}(\cos\theta_i) \, h_{pqt}^{\beta}$$

expression (2.99) becomes

$$x_{nmt}^{\alpha i} = \bar{P}_{nm}(\cos\theta_i) \sum_{pq\beta} [C_{m+q}^{i|\alpha-\beta|} + (-1)^{2\alpha+\beta} C_{m-q}^{|\alpha-\beta|}] \, d_{pq}^{i\beta}$$

All quantities inside the square brackets are constant if α, β, m, and q are the same. Grouping equal factors together

$$\begin{aligned} x_{nmt}^{\alpha i} &= \bar{P}_{nm}(\cos\theta_i) \sum_{q=n}^{N-1} \sum_{\beta=0}^{1} [C_{m+q}^{i|\alpha-\beta|} + (-1)^{2\alpha+\beta} C_{m-q}^{i|\alpha-\beta|}] \sum_{p=0}^{N-1} d_{pq}^{i\beta} \\ &= \bar{P}_{nm}(\cos\theta_i) \sum_{q=n}^{N-1} \sum_{\beta=0}^{1} [C_{m+q}^{i|\alpha-\beta|} + (-1)^{2\alpha+\beta} C_{m-q}^{i|\alpha-\beta|} D_q^{i\beta} \end{aligned} \tag{2.100}$$

where $D_q^{i\beta} = \sum_{p=0}^{N-1} d_{pq}^{i\beta}$

There are (N^2) products needed to form all $d_{pq}^{i\beta}$; (N^2) sums to compute the $2N$ $D_q^{i\beta}$, and $(N)^2$ further multiplications by the $P_{nm}(\cos\theta_i)$ to form all $x_{n m t}^{\alpha i\beta}$. As there are I non-zero G^i, there are, per matrix-vector product, $I x O(N^2)$ operations, or $O(N^3)$ if the data is not too sparse. M, the number of matrix-vector products in the solution, is $O(N^2)$, so the total comes to $O(N^5)$. Inverting G by the usual methods and without having regard for its structure involves $O(N^6)$ operations. The gain in efficiency is, again, $O(N)$.

Besides providing a convenient way of demonstrating how the properties of G can be exploited to make its inversion more efficient (or the solution of the normal equations, both approaches are equivalent), conjugate gradients is interesting on its own right. Sparse data sets with a poor distribution will result in ill-conditioned normals, so the inversion of G may be numerically impossible. So-called iterative methods, such as conjugate gradients, usually improve the <u>initial guess</u> (represented by $\underline{h}_0$) of the correct values of the unknowns, at least for the first few iterations. Improvement here means a reduction in the quadratic form being minimized, such as the mean square value of the residuals. If the initial guess is a good one, and present day spherical harmonic coefficients of the gravity field are reasonable good for degrees up to 30 or so, then a few iterations are likely to improve this guess, and to produce reasonable estimates of those coefficients that, being wholly unknown, are taken to be zero at the start. If a "few" iterations are much less than the maximum N^2, then a reduction in computing time of $O(N^3)$ takes place. This might allow scientists to "extend" existing models to much higher degree and order than at present, simply by obtaining approximate solutions of this type.

Clearly, all that has been said here regarding G and least squares adjustment applies equally well to $(G + C^{-1})$ and least squares collocation. Though the formulas have been developed on the basis of a point values' formulation, their extension to area means is not difficult.

While the descriptions of the methods for estimating the $\bar{C}^{\alpha}_{nm}$ discussed here have been confined to the case where all the data are of one kind (i.e., gravity anomalies only, or magnetic anomalies only, or geopotential numbers, etc.) their extension to mixed data sets is immediate, provided that all values are globally distributed according to grids of the same $\Delta\lambda$, although the latitudes need not be the same as well.

Note on the Accuracies of the $\hat{C}^{\alpha}_{nm}$ when Using Conjugate Gradients:

Besides the estimates of the coefficients, one usually wants to know the accuracies of those estimates. When a few iterations of conjugate gradients are used to update some initial estimates in the efficient way described above, what are the accuracies of the improved coefficients? In the course of the conjugate gradients procedure (see references), the conjugate directions $\underline{v}_k$ of $G = (A^T D^{-1} A + C^{-1} + Q_s^{-1})$ are generated (Q_s^{-1} is the variance covariance matrix of the initial estimates, as explained in paragraph (2.18) part (b)), together with the scalars $\underline{v}_k^T G \underline{v}_k = \alpha_k$. The conjugate directions have the property

$$\underline{v}_k^T G \underline{v}_p = 0$$

for $k \neq p$. The estimator implied by R iterations of this procedure is

$$\hat{\underline{c}} = \tilde{G}(A^T D^{-1} \underline{m} + Q_s^{-1} \underline{c}_s) = \tilde{G} A^T D^{-1} (\underline{z} + \underline{n}) + \tilde{G} Q_s^{-1} (\underline{c} + \Delta\underline{c}_s)$$

where $\underline{c}_s$ is the vector of initial estimates, $\Delta\underline{c}_s$ the errors in this vector, and

$$\tilde{G} = \sum_{k=1}^{R} \alpha_k^{-1} \underline{v}_k \underline{v}_k^T$$

The variance-covariance matrix of the updated errors, for ordinary least squares, is (assuming that $E\{\underline{n}\, \Delta\underline{c}_s^T\} = 0$)

$$E\left\{(\tilde{G} A^T D^{-1} \underline{n})(\tilde{G} A^T D^{-1} \underline{n})^T\right\} + E\left\{(\tilde{G} Q_s^{-1} \Delta\underline{c}_s)(\tilde{G} Q_s^{-1} \Delta\underline{c}_s)^T\right\} = \tilde{G} A^T D^{-1} D D^{-1} A \tilde{G}$$

$$+ \tilde{G} Q_s^{-1} Q_s Q_s^{-1} \tilde{G} = \tilde{G}(A^T D^{-1} A + Q_s^{-1}) \tilde{G} = \tilde{G} G \tilde{G}$$

$$= \sum_{k=1}^{R} \alpha_k^{-1} \underline{v}_k \alpha_k \underline{v}_k^T \alpha_k^{-1} = \sum_{k=1}^{R} \alpha_k^{-1} \underline{v}_k \underline{v}_k^T = \tilde{G}$$

Therefore, the variances of the errors in the $\hat{C}^{\alpha}_{nm}$ are equal to the corresponding diagonal elements of $\tilde{G}$:

$$\sigma^{2\alpha}_{\eta\, nm} = \sum_{k=1}^{R} \alpha_k^{-1} v^2_{k,nm\alpha}$$

where $v_{k,nm\alpha}$ is the element of $\underline{v}$ corresponding to $\hat{C}^{\alpha}_{nm}$ in $\hat{\underline{c}}$ ($\underline{v}$ and $\hat{\underline{c}}$ have the same dimension). The same result applies to least squares collocation. All that is needed to obtain the $\sigma^{2\alpha}_{\eta\, nm}$, according to the formula above, is knowledge of the α_k and of the $\underline{v}_k$, which can be saved as they are created, during the kth iteration of the procedure. For the efficient algorithm to be applicable, Q_s^{-1} must have a suitable structure. One case is when Q_s^{-1} is diagonal, or can be satisfactorily approximated by a diagonal matrix.

2.17 The Error Matrix in the Band-Limited Case

From (2.41) and (2.77) results

$$E_T = C - C_{gz}(C_{zz} + D)^{-1} C_{gz}^T = C - F C_{gz}^T = C - (A^T D^{-1} A + C^{-1})^{-1} A^T D^{-1} A C$$

$$= [I - (A^T D^{-1} A + C^{-1})^{-1} A^T D^{-1} A] C = (A^T D^{-1} A + C^{-1})^{-1} [(A^T D^{-1} A + C^{-1}) - A^T D^{-1} A] C$$

$$= (A^T D^{-1} A + C^{-1})^{-1} \tag{2.101}$$

In the case of the best unbiased estimator C^{-1} is not present in the normal matrix, so (2.101) becomes

$$E_T = (A^T D^{-1} A)^{-1} = G^{-1} \tag{2.102}$$

which is the well known expression of the error matrix for ordinary least squares.

Because of the block-diagonal structure of the variance-covariance matrix,[1] the estimates of $\bar{C}_{nm}^{\alpha}$ of different orders are uncorrelated. The diagonal elements of the variance-covariance matrix are the variances of the estimated coefficients total errors (i.e., sampling plus propagated noise), in (2.101).

Obtaining the variances of the $\hat{C}_{nm}^{\alpha}$ in band-limited collocation is formally identical to getting the variances of the estimates in ordinary least squares, according to (2.101) and to (2.102)

2.18 The Use of a priori Information on the Coefficients

Assume that all coefficients up to some degree and order M are approximately known, and that Q_s is the variance-covariance matrix of their errors. This could be the case where a model of the gravity field has been obtained from data gathered using artificial satellites, complete to degree M, and terrestrial data is to be used to improve the existing coefficients and obtain new ones beyond degree M.

Three possible approaches to this question will be discussed here, using a "point values" formulation in the first two cases for simplicity.

---

[1] When the data set is not sparse.

(a) Simple weighted averages:

The terrestrial data can be used separately to obtain a model to degree and order $M_T > M$, together with the variance-covariance matrix Q_T of the coefficients' errors. To combine the satellite and the terrestrial coefficients one can set up the following observation equations

$$\begin{bmatrix} \underline{c}_s \\ \underline{c}_T \end{bmatrix} + \begin{bmatrix} \underline{s} \\ \underline{t} \end{bmatrix} = \begin{bmatrix} I_s \\ I_T \end{bmatrix} \underline{c} \tag{2.103}$$

where $\underline{c}$ is the vector potential coefficients and I_s is the $(M + 1)^2 \times (M + 1)^2$ unit matrix augmented with zeroes on the right, and I_T is the $(M_T + 1)^2 \times (M_T + 1)^2$ unit matrix, while $\underline{c}_s$ is the vector of "satellite" coefficients and $\underline{c}_T$ the vector of "terrestrial" coefficients; $\underline{s}$ and $\underline{t}$ are the corresponding vectors of residuals. The best linear unbiased estimator for the combined system of observations is

$$\begin{aligned} \hat{\underline{c}} &= \left([I_s^T \; I_T^T] \begin{bmatrix} Q_s^{-1} & O \\ O & Q_T^{-1} \end{bmatrix} \begin{bmatrix} I_s \\ I_T \end{bmatrix} \right)^{-1} [I_s^T \; I_T^T] \begin{bmatrix} Q_s^{-1} & O \\ O & Q_T^{-1} \end{bmatrix} \begin{bmatrix} \underline{c}_s \\ \underline{c}_T \end{bmatrix} \\ &= (Q_s^{-1} + Q_T^{-1})^{-1} (Q_s^{-1} \underline{c}_s + Q_T^{-1} \underline{c}_T) \end{aligned} \tag{2.104}$$

If Q_s and Q_T are the inverses of ordinary least squares normal matrices G_s and G_T, then the error matrix of the combined solution

$$E = (Q_s^{-1} + Q_T^{-1})^{-1} = (G_s + G_T)^{-1}$$

corresponds to the propagated noise only. If they are "collocation" matrices of the type $(A^T D^{-1} A + C^{-1})^{-1}$ (see expression (2.77)), then the error matrix includes the effect of the sampling error as well. Most satellite models are, to date, "least squares-type" and it would be incongrous to combine them with "collocation-type" models, terrestrial or otherwise. The problem need not be a serious one, because geodetic spacecrafts so far have orbited at altitudes of 800 km or more, where the field is much smoother than at the surface, so the sampling errors are bound to be small compared to the propagated data errors reflected by Q_s.

In the case where the terrestrial model has been derived from a regularly sampled data set using the "band-limited approach" of previous paragraphs, Q_T is block-diagonal, and those blocks corresponding to orders $m > M$ are identical to the corresponding blocks in $(Q_s^{-1} + Q_T^{-1})^{-1}$. From this it is not difficult to conclude that the coefficients in the combined model up to order M will be somewhat different (and presumably better) than those in either the satellite or the terrestrial sets, while those above M will be identical to the corresponding terrestrial coefficients.

(b) A priori values included as data in the adjustment:

Consider the system of observation equations

$$\begin{bmatrix} \underline{m} \\ \\ \underline{c}_s \end{bmatrix} + \begin{bmatrix} \underline{v} \\ \\ \underline{s} \end{bmatrix} = \begin{bmatrix} A \\ \\ I_s \end{bmatrix} \underline{c} \tag{2.105}$$

which is the system (2.66) augmented with equations of the type

$$\hat{C}^{\alpha}_{nm(s)} - e^{\alpha}_{nm(s)} = \bar{C}^{\alpha}_{nm}$$

where $e^{\alpha}_{nm(s)}$ is the error in the "satellite model" coefficient $\hat{C}^{\alpha}_{nm(s)}$. The normal matrix of collocation is

$$E_T^{-1} = ([A^T I_s^T] \begin{bmatrix} D^{-1} & O \\ O & Q_s^{-1} \end{bmatrix} \begin{bmatrix} A \\ I_s \end{bmatrix} + C^{-1}) \tag{2.106}$$

and the optimal estimator of the band-limited type is

$$\hat{\underline{c}} = (A^T D^{-1} A + Q_s^{-1} + C^{-1})^{-1} (A^T D^{-1} \underline{m} + Q_s^{-1} \underline{c}_s) \tag{2.107}$$

Once more, if the data in $\underline{m}$ has been sampled regularly on the sphere, the estimated coefficients will be affected by the existence of a priori values only if their order is no higher than M. Naturally, this is a desirable situation. Also it is important that the error measures corresponding to E_T and Q_s be congruous, though this is probably not very important in the case of satellite models obtained with high-orbiting spacecraft. Notice that (a) and (b) are equivalent when $Q_T = (A^T D^{-1} A)^{-1}$ and when C^{-1} is excluded from (2.106)-(2.107). In other words: these first two approaches are equivalent for ordinary least squares.

(c) The method of Kaula and Rapp:

W. Kaula (1966) proposed a technique for simultaneously filtering errors out of a terrestrial data set and improving the coefficients of a satellite model. This method was later developed by R. Rapp (1968), who more recently (1978) used it to improve a global data set of mean $1^\circ \times 1^\circ$ anomalies by combining it with the potential coefficients of the GEM-9 model. This adjusted data set was used by the author of this report to create the $5^\circ \times 5^\circ$ mean anomalies analysed in one of the numerical experiments of section 3.

The idea is to satisfy condition equations of the type

$$\hat{C}^{\alpha}_{nm(s)} + d^{\alpha}_{nm} - (4\pi\gamma(n-1)\beta_n S^{n+2})^{-1} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \int_{\sigma_{ij}} \bar{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma (\bar{\Delta g}_{ij} + v_{ij}) = 0 \tag{2.108}$$

while minimizing the quadratic form

$$g(\underline{d}, \underline{v}) = \underline{d}^T Q_s^{-1} \underline{d} + \underline{v}^T D^{-1} \underline{v} \tag{2.109}$$

where $\underline{v}$ is the vector of corrections v_{ij} to the mean gravity anomalies $\overline{\Delta g}_{ij}$, and $\underline{d}$ is the vector of corrections d_{nm}^{α} to the satellite potential coefficients $\hat{C}_{nm(s)}^{\alpha}$; β_n is the nth degree Pellinen factor discussed in paragraph (4.3), γ is the mean value of equatorial gravity, and S is the ratio between the radius of the Earth's largest inner geocentric sphere and the mean Earth radius. In matrix form, the condition equations (2.108) are

$$\underline{c}_s + \underline{d} - A^T \underline{\overline{\Delta g}} - A^T \underline{v} = 0 \tag{2.110}$$

where A is a $2N^2 \times (M+1)^2$ matrix having columns $\underline{a}_{nm}^{\alpha}$ of the form

$$\underline{a}_{nm}^{\alpha} = (4\pi(n-1)\gamma S^{n+2} \beta_n)^{-1} \left[\int_{\sigma_{0\,0}} \overline{Y}_{nm}^{\alpha}(\theta,\lambda)\, d\sigma \ldots \int_{\sigma_{N-1\;2N-1}} \overline{Y}_{nm}^{\alpha}(\theta,\lambda)\, d\sigma\right]^T$$

so, except for the factor $(4\pi(n-1)\gamma S^{n+2} \beta_n)^{-1}$, A is the "area means version" of the matrix of system (2.66), and has the same properties as the "A" matrices considered so far.

The optimal estimates are given by the expressions

$$\underline{\hat{c}} = \underline{c}_s + \underline{d} \tag{2.111-a}$$

$$\underline{\hat{\Delta g}} = \underline{\overline{\Delta g}} + \underline{v} \tag{2.111-b}$$

where

$$\underline{d} = -((A^T D A)^{-1} + Q_s^{-1})^{-1} (A^T D A)^{-1} (\underline{c}_s - A^T \underline{\overline{\Delta g}}) \tag{2.112}$$

and

$$\underline{v} = D A Q_s \underline{d} \tag{2.113}$$

If the data set is both complete and of uniform quality, the matrix $A^T D A$ has the block structure first discussed in paragraph (2.15). The presence of D instead of D^{-1} makes no difference to the calculations needed to set up and invert the matrix: the procedures are those already explained. Terrestrial data, however, is usually both scattered and of varying quality (i.e., different noise variances). For this type of data, therefore, the methods for scattered measurements given in paragraph (2.16) could be used.

2.19 Optimal Estimation over a Band of Spatial Frequencies

Assume that the signal is of the type

$$\underline{m} = A\,\underline{c} + \underline{n}$$

where A and $\underline{c}$ may now be infinite (i.e., all degrees from 0 to ∞ may be present). If $\underline{c}'$ is a sub-vector of $\underline{c}$ comprising, say, the first N^2 coefficients, and if $\hat{\underline{s}}$ is a vector of estimates of a function s at a given set of points on the sphere, such that the values of s depend only on those of $\underline{c}'$ according to the relationship

$$\underline{s} = B\,\underline{c}' \tag{2.114}$$

where B is some matrix of appropriate size, not necessarily of the same type as A, then the optimal estimator for $\hat{\underline{s}}$ is, according to (2.39), (2.75),

$$\begin{aligned}
\hat{\underline{s}} &= C_{sz}\,(C_{zz} + D)^{-1}\,\underline{m} \\
&= M\{\underline{s}\,\underline{z}^T\}\,(C_{zz} + D)^{-1}\,\underline{m} = M\{B\,\underline{c}'\,\underline{c}'^T A^T\,(C_{zz} + D)^{-1}\} \\
&= B\,C'\,A^T\,(C_{zz} + D)^{-1}\,\underline{m} \qquad (\text{where } C' = M\{\underline{c}'\,\underline{c}'^T\}) \\
&= B\,C_{c'z}\,(C_{zz} + D)^{-1}\,\underline{m}
\end{aligned}$$

or

$$\hat{\underline{s}} = B\,\hat{\underline{c}}' \tag{2.115}$$

according to (2.40).

Expression (2.115) indicates that the optimal estimates of a band-limited function s from data $\underline{m}$ are identical to the values of s obtained from the optimal estimates of the coefficients $\underline{c}'$ by means of the relationship $\hat{\underline{s}} = B\,\hat{\underline{c}}'$.

3. Numerical Examples

This section presents several computed examples to illustrate some of the ideas and methods discussed earlier on. The question posed here is, basically, that of the accuracy of the various procedures, and is answered by means of error analysis carried out with the formulas for error variance developed in section 2, and also by analysing simulated data and comparing the recovered coefficients to the original ones in order to find the actual errors. A comparison of the rms of these errors with the theoretical rms (i.e., the square root of the variance) provides both a check on each set of results and, more important, shows just how adequate an error measure the theoretical rms can be. Besides error analysis and simulations, this section shows, in the last paragraph, the results of the harmonic analysis of a real data set: a 5° x 5° equal angular set of mean gravity anomalies covering the whole Earth, from which the coefficients of the disturbing potential have been recovered to degree and order 36 by means of least squares collocation, using the "Toeplitz matrix" approach of paragraphs (2.10) and (2.11).

3.1 Generation and Analysis of Simulated Data

As explained in the preceeding section, the variance of the error in the estimate of $\bar{C}_{nm}^{\alpha}$ depends on the power spectrum (or covariance function) of the signal, and on the variance-covariance matrix of the noise. The propagation of the noise is quite straightforward, and anybody who has had any practical experience with adjustments of geodetic networks and the like already has enough "feeling" for this part of the error measure, and is capable of understanding its significance when its value is given to him. The part corresponding to the sampling error is somewhat different; it involves a rather unusual geometric average over rotations, and this type of error measure, while not exactly new (collocation, based on this measure, has been around since the mid-sixties) is not so familiar to geodesists yet, and its use, in harmonic analysis in particular, far from common practice. For this reason, it is probably fair to the readers to provide some illustration of how "close" this part of the error measure is to the actual sampling error that occurs when data of the assumed power spectrum is analyzed in any of the ways discussed so far to recover spherical harmonic coefficients. By "close" one means that the actual numbers measuring the theoretical and the actual variances (or rms) should differ from each other by a small percentage, or some equally clear-cut criterion.

The theoretical variance considered here is the variance of the estimation errors per degree $\sigma_{\epsilon n}^2$, defined in terms of the error measure of section 2 as follows (see paragraph (2.8))

$$\sigma^2_{\epsilon n} = \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \sigma^2_{nm} = \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \left(\frac{\sigma_n^2}{(2n+1)} - 2\underline{c}^T_{nm}\alpha,_z \underline{f}^{\alpha}_{nm} + (\underline{f}^{\alpha}_{nm})^T (C_{zz} + D) \underline{f}^{\alpha}_{nm} \right) \tag{3.1}$$

The rms of the error is the square root of this variance, and the ratio of this rms to the rms per coefficient:

$$\frac{\sigma_{\epsilon n}}{\sigma_n} = \left(1 - \sigma_n^{-2} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} \left(2\, \underline{c}^T_{nm}\alpha,_z \underline{f}^{\alpha}_{nm} - (\underline{f}^{\alpha}_{nm})^T (C_{zz} + D)\, \underline{f}^{\alpha}_{nm} \right)\right)^{\frac{1}{2}} \tag{3.2}$$

multiplied by 100 , or the percentage rms error per degree, is the theoretical quantity to be compared to the "actual" percentage rms error per degree derived from the analysis of simulated data with the same statistical characteristics (i.e., σ_n^2's , C_{zz} , and D) in formula (3.2).
The $\sigma^2_{\epsilon n}$ were computed with subroutine NORMAX (Appendix B).

To obtain the actual percentage rms error per degree, sets of simulated data were created on full regular grids as follows: the artificial data consisted of area means computed globally, using the algorithm outlined in paragraph (1.7) and subroutine SSYNTH (Appendix B), on the basis of expression (1.2). The $\overline{C}^{\alpha}_{nm}$, complete to degree and order $N_{max} \gg N = \frac{2\Pi}{\Delta\lambda}$, came from sequences of random numbers. The random numbers, obtained using the IMSL subroutine "GGNOR" with generating "seeds" of the order of 10^4 , were scaled to give them the desired degree variances σ_n^2 . For each simulation, a sequence of $(N_{max} + 1)^2$ numbers was obtained, the first corresponding to $\overline{C}_{00}$, the second, third, and fourth to $\overline{C}_{10}$, $\overline{C}_{11}$, and $\overline{S}_{11}$, respectively, and so forth. If r_{n0} , r^{α}_{n1} , . . . r^{α}_{nn} were the (2n + 1) numbers corresponding to degree n , then the scaling that resulted in the corresponding $\overline{C}^{\alpha}_{nm}$ was

$$\overline{C}^{\alpha}_{nm} = r^{\alpha}_{nm} \left[\frac{\sigma_n}{\sqrt{\sum_{m=0}^{n} \sum_{\alpha=0}^{1} (r^{\alpha}_{nm})^2}} \right] \tag{3.3}$$

The harmonic coefficients $\overline{C}^{\alpha}_{nm}$ obtained in this way were the "actual" coefficients to which the $\hat{C}^{\alpha}_{nm}$, recovered by some of the procedures described in section 2, were then compared to obtain the actual percentage rms errors per degree

$$\zeta_n = \left[\sum_{m=0}^{n} \sum_{\alpha=0}^{1} (\hat{C}^{\alpha}_{nm} - \overline{C}^{\alpha}_{nm})^2 \right]^{\frac{1}{2}} \sigma_n^{-1} \times 100 \tag{3.4}$$

The analysis of the simulated data was done with subroutine HARMIN (appendix B). This type of numerical experiment was carried out three or more times in each case, varying only the seed used to generate the random numbers, much as a Montecarlo-type of analysis is conducted. The seeds were chosen widely apart, to ensure that the correlation between "trials" would be virtually nil.

The maximum degree and order in the set of artificial coefficients, N_{max}, was chosen so that the power in the mean values above degree N_{max}

$$P_{N\,max} = \sum_{n=N\,max+1}^{2000} \beta_n^2\ \sigma_n^2 \tag{3.5}$$

were less than 1% of the power between degrees 0 and 2000. The β_n are the Pellinen coefficients, discussed in paragraph (4.3), corresponding to 5° x 5° area means. The values used for the degree variances σ_n^2 had been empirically obtained from terrestrial data in the manner described below. The data were supposed to be noise-free, as only the sampling part of the error was studied in this way, for the reasons given at the beginning of this paragraph. The estimators being linear, the propagated noise and the sampling error merely add arithmetically to each other, and can therefore be studied separately, if so desired. Summing up, it can be said that the simulated data consisted in global data sets of artificial gravity anomalies, averaged over equal angular grids.

The empirical degree variances were obtained as follows: up to degree 100 they were those implied by a set of coefficients, complete to degree 180, obtained by R. Rapp and associates at O.S.U. from a global data set of 1° x 1° mean anomalies. Above degree 100, the $\sigma_n^2(\Delta g)$ were obtained from a model of the form

$$\sigma_n^2(\Delta g) = (n-1)\left(\frac{\alpha_1}{(n+A)}\,S_1^{n+2} + \frac{\alpha_2}{(n+B)}\,\frac{S_2^{n+2}}{(n-2)}\right)\ [mgal^2] \tag{3.6}$$

(Moritz, 1976), where the parameters α_1, α_2, S_1, S_2, A and B have been adjusted to fit existing gravimetric data, satellite altimetry, satellite field models, and other geophysical data. The parameters used in most examples were

$\alpha_1 = 3.4050$	$S_1 = 0.998006$	$A = 1.$
$\alpha_2 = 140.03$	$S_2 = 0.914232$	$B = 2.$

corresponding to the best model of this type given in a report by R. Rapp (1979) who, in the same work, discusses also the empirical degree variances obtained from his 180, 180 field model. The degree variances implied by (3-6) with the parameters listed above are also very similar to those obtained by quite different means by Wagner and Colombo (1979), who analyzed the (Fourier) power spectrum of short arcs of GEOS-3 altimetry, and converted their average to a spherical harmonics spectrum using formulas that follow from the relationship between spherical harmonics and Fourier series. The empirical variances for $n \leq 100$ are included in the listing of subroutine NVAR, in Appendix B.

In order to understand how critical the choise of empirical degree variances is to the theoretical and actual errors, a different two-term model obtained by C. Jekeli (1978) was used as well. This model has the following parameter values

$\alpha_1 = 18.3906$ $\qquad S_1 = 0.9943667$ $\qquad A = 140.$

$\alpha_2 = 658.6132$ $\qquad S_2 = 0.9048949$ $\qquad B = 10.$

All the examples considered in this section refer to complete equal angular sets of mean values with the same statistical properties of terrestrial gravity anomalies (as far as such properties are known). The analysis of actual mean gravity anomalies is shown in the last paragraph. Besides being important in geodetic studies, gravity anomalies constitute a type of geophysical data with reasonably well known statistical properties, and their study here is meant to give the reader some idea of how effective are the ideas presented earlier when it comes to handling "real data" (or something resembling it).

3.2 Agreement between the Actual and the Theoretical Measures of the Sampling Errors

Table (3.1) lists side by side the theoretical precentage rms per degree of the sampling error according to (3.2) and the actual value of this percentage for two different sets of coefficients (i.e., from random sequences with different seeds). The coefficients were recovered using the quadratures formula

$$\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi\beta_n} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{\Delta g}_{ij} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma$$

This type of formulas has been discussed in paragraph (2.6). The simulated data consisted in full sets of $5^\circ \times 5^\circ$ mean anomalies obtained from harmonic coefficients complete to degree and order $N_{max} = 140$. The power above degree 140 in $5^\circ \times 5^\circ$ anomalies is negligible, according to the empirical power spectrum model that was used. The results shown here are fairly typical of similar tests conducted with other quadrature formulas, so the conclusions that can be drawn are likely to be valid for the analysis of area means by numerical quadratures in general. There is clear agreement between the theoretical and the actual rms of the errors, and not just the average rms of actual errors, but the actual rms of each trial as well. The agreement is close, and the reader will probably agree that to use a theoretical error measure that can predict the actual error so well is a meaningful way of quantifying the error.

3.3 Accuracies of Various Quadratures Formulas

Five quadratures formulas for area means have been studied: the first four of the type

$$\hat{C}^{\alpha}_{nm} = \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma \; \overline{\Delta g}_{ij}$$

Table 3.1

Comparison of actual versus theoretical percentage rms error per degree.
$5^\circ \times 5^\circ$ mean anomalies, $N_{max} = 140$, 0. mgal rms noise.

n	Actual, No 1 (seed=53218)	Actual, No 2 (seed=31765)	Average 1 and 2	Theoretical (expression (3.2) x 100)
2	0.50	0.62	0.56	0.51
5	1.06	1.10	1.08	1.23
10	4.99	5.81	5.40	4.89
15	13.06	11.03	12.05	12.75
20	23.48	25.97	24.72	24.77
25	23.04	30.82	26.93	30.03
30	45.84	46.26	46.05	43.28
* 36 (N)	55.33	61.10	58.22	60.35
40	70.61	75.82	73.22	75.14
45	85.91	86.41	86.16	87.03
50	101.51	101.43	101.47	101.47

differ only in μ_n:
(a) $\mu_n = \overset{*}{\mu}_n$ the optimal de-smoothing factor given by expression (2.36) in paragraph (2.7):

(b) $\mu_n = \dfrac{1}{4\pi\beta_n^2}$

(c) $\mu_n = \dfrac{1}{4\pi\beta_n}$

(d) $\mu_n = \dfrac{1}{4\pi}$

(e) The optimal quadratures - type formula, in the least squares collocation sense (i.e., minimum combined error measure) for the given grid; signal and noise (paragraph (2.8)). The grid was equal angular in all five cases. Table (3.2) compares the percentage rms of the errors per degree for a $30^\circ \times 30^\circ$ grid; table (3.3) corresponds to a $10^\circ \times 10^\circ$ grid; and table (3.4) to a $5^\circ \times 5^\circ$ grid. N_{max} was 100 for the first two tables, and 140 for the last. All these values are theoretical, computed in accordance to the formulas in paragraph (3.1). In all three cases noise is not present, so these errors are purely sampling errors.

Table (3.5) corresponds to a $5^\circ \times 5^\circ$ grid and an uniform noise of 5 mgal rms. Here the effect of the noise has been included in the results. Table (3.6) compare the degree correlation coefficients for the various methods, for a noise of 0 mgal. The correlation coefficient for the nth degree is defined as

* (N): "Nyquist frequency" of the grid.

$$\rho_n = [\sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}_{nm}^{\alpha} \hat{C}_{nm}^{\alpha} (\sum_{m=0}^{n} \sum_{\alpha=0}^{1} \overline{C}_{nm}^{\alpha 2})^{-1} (\sum_{m=0}^{n} \sum_{\alpha=0}^{1} \hat{C}_{nm}^{\alpha 2})^{-1}]^{\frac{1}{2}} \tag{3.7}$$

and it is also equal to

$$\rho_n = \left[\frac{\int_\sigma \Delta g_n \, \Delta\hat{g}_n \, d\sigma}{\int_\sigma \Delta g_n^2 \, d\sigma \int_\sigma \Delta\hat{g}_n^2 \, d\sigma} \right]^{\frac{1}{2}} \tag{3.8}$$

This coefficient can be regarded either as a measure of the agreement between the actual and the recovered coefficients of the nth harmonic, or of between the nth harmonic Δg_n in the signal and the harmonic $\Delta\hat{g}_n$ that can be computed from the recovered coefficients. If the $\overline{C}_{nm}^{\alpha}$ could be seen as random variables with gaussian distribution, the interpretation of ρ_n would move along well-worn paths; however, as it was mentioned in paragraph (2.5), there are some unexpected problems when extending the idea of a gaussian random process to the sphere, so is better to chose another approach. One could regard the coefficients of the nth harmonic as the coordinates of a vector in (2n + 1)-dimensional euclidean space. The actual coefficients will define thus one vector, and the recovered coefficients another. Expression (3.7) then merely defines ρ_n as the scalar product of this two vectors. Likewise, expression (3.8) is that of the scalar product of two elements of a function space. The angle formed by these vectors is $0°$ when correlation is (maximum), and $90°$ for 0 correlation; a minimum correlation of -1 corresponds to the case when the vectors are equal but of opposite sense. The scalar product is independent of any scale factors that may multiply the vectors: it depends only on their mutual orientation. For this reason, the correlation coefficients are the same for the four quadratures formulas of the type

$$\hat{C}_{nm}^{\alpha} = \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \overline{\Delta g}_{ij} \int_\sigma \overline{Y}_{nm}^{\alpha} (\theta, \lambda) \, d\sigma$$

because the difference between the errors for the same harmonic, predicted with two different formulas of this kind, consists in a scale factor $\mu_n^{(1)}/\mu_n^{(2)}$. Clearly, as the rms of the error increases with n, ρ_n decreases from almost 1 where the error is smallest (very low degrees), to below 0.5 where the error exceeds 90% (highest degrees analyzed).

Observing the percentual rms of the errors in the first four tables, it is easy to see that they by no means reach 100% as soon as the Nyquist frequency is reached (n = N), but that they remain substantially smaller than 100% even at degrees considerably higher than N ; this is in line with the conclusions in paragraph (1.3). The optimal estimator itself cannot have an error larger than 100% , be it due to sampling, noise, or both. Otherwise, a null estimator (one that predicts only zeroes) would be better than the optimal, which is not possible.

Table (3.7), compares the theoretical errors with zero noise (i.e., the sampling errors) of the collocation estimator obtained, first according to

σ_n^2 implied by R. Rapp's model (used in all the other tables), and then according to the "2L" model of C. Jeleky, both described in paragraph (3.1). This table is included here to give the reader an idea of how sensitive the theoretical error variances are to the empirical degree variances used to compute them. The "2L" model has considerably more power than Rapp's at high degrees, and this may be reflected in the somewhat larger errors in the corresponding column of the table.

Table 3.2

Theoretical percentage rms error per degree.

$30° \times 30°$ mean anomalies, $N_{max} = 100$, 0. mgal rms noise.

n	Optimal Estimator	$\overset{*}{\mu}_n$	$\frac{1}{4\pi\beta_n}$	$\frac{1}{4\pi}$	$\frac{1}{4\pi\beta_n^2}$
2	12.51	13.17	13.95	15.86	13.57
4	25.09	25.60	26.91	36.12	28.72
6 (N)	58.77	65.13	68.03	67.18	101.41
10	92.82	96.01	194.08	97.13	882.98
12	98.44	99.27	915.49	105.44	17835.58

Table 3.3

Theoretical percentage rms error per degree.

$10° \times 10°$ mean anomalies, $N_{max} = 100$, 0. mgal rms noise.

n	Optimal Estim.	$\overset{*}{\mu}_n$	$\frac{1}{4\pi\beta_n}$	$\frac{1}{4\pi}$	$\frac{1}{4\pi\beta_n^2}$
2	1.52	1.63	1.71	2.04	1.65
4	2.24	2.46	3.00	4.76	2.56
6	4.27	4.68	5.83	9.51	4.94
8	8.74	9.60	10.99	16.39	10.16
10	13.19	14.02	15.89	23.68	15.19
12	31.60	33.03	33.10	37.84	37.43
14	40.43	41.41	41.41	46.45	49.91
16	41.41	42.18	42.36	51.41	53.12
18 (N)	56.81	59.94	61.26	63.41	88.17
20	76.50	78.99	94.69	79.07	158.26
22	76.02	78.54	89.75	79.05	167.49
24	84.96	87.40	116.57	87.59	258.09
26	90.40	94.03	164.50	97.85	443.76
28	92.21	95.98	193.17	100.35	646.88
30	93.31	97.30	232.24	101.88	1010.13
32	95.80	98.30	303.89	103.49	1828.98
34	94.95	97.40	281.73	97.77	2619.98
36	96.87	98.08	485.72	98.45	8694.11

Table 3.4
Theoretical percentage rms error per degree
$5^\circ \times 5^\circ$ mean anomalies, $N_{max} = 140$, 0. mgal rms noise.

n	Optimal Estim.	$\overset{*}{\mu}_n$	$\frac{1}{4\pi\beta_n}$	$\frac{1}{4\pi}$	$\frac{1}{4\pi\beta_n^2}$
2	0.47	0.49	0.51	0.59	0.50
6	1.36	1.44	1.71	2.61	1.48
12	9.74	10.11	10.40	12.27	10.37
18	15.14	15.55	16.35	21.20	16.46
24	30.58	31.06	31.21	36.30	34.89
30	42.88	43.28	43.28	49.04	53.32
36 (N)	57.62	59.15	60.34	62.67	85.90
42	72.16	73.51	80.25	74.72	137.19
48	82.15	83.62	101.19	83.82	216.98

Table 3.5
Theoretical percentage rms per degree
$5^\circ \times 5^\circ$ mean anomalies, $N_{max} = 140$, 5. mgal rms noise.

n	Optimal Estim.	$\overset{*}{\mu}_n$	$\frac{1}{4\pi\beta_n}$	$\frac{1}{4\pi}$	$\frac{1}{4\pi\beta_n^2}$
2	8.74	8.81	8.84	8.83	8.85
6	9.33	9.34	9.34	9.43	9.41
12	32.80	32.88	33.79	33.02	35.28
18	35.94	36.04	36.54	36.25	39.86
24	53.32	53.47	56.73	53.49	66.99
30	61.97	62.14	67.15	62.15	87.22
36 (N)	71.98	72.62	82.81	72.63	122.67
42	81.55	82.17	102.91	82.54	181.19
48	88.00	88.72	124.09	89.54	271.87

Table 3.6
Correlation factor per degree.
$5^\circ \times 5^\circ$ mean anomalies, N_{max}=140, 0. mgal noise.

n	Optimal Estim.	Simple quadratures
2	1.00	1.00
6	1.00	1.00
12	1.00	1.00
18	0.99	0.99
24	0.96	0.96
30	0.90	0.89
36 (N)	0.81	0.81
42	0.72	0.71
48	0.59	0.57

Table 3.7
Comparison between the theoretical errors (optimal estimator) when different covariance functions are assumed. $N_{max} = 140$, $5^\circ \times 5^\circ$ mean anomalies, 0. mgal noise.

n	R. Rapp's variances	"2L" model
2	0.47	0.64
6	1.36	1.79
12	9.74	7.84
18	15.14	18.10
24	30.58	31.41
30	42.88	46.28
36 (N)	57.62	59.84
42	72.16	72.19
48	82.15	81.80

The results in the tables show that the optimal estimator errors are the smallest, as expected, and that the quadratures formula with the optimal de-smoothing factors is the best of the four simple quadratures formulas compared here, though it is not quite as good as the optimal estimator, also as expected. In the case of zero noise each of the three non-optimal quadratures formulas has errors that, in some region of the spectrum, are smaller than those for the other two; concretely, the de-smoothing factor $\mu_n = \frac{1}{4\pi\beta_n^2}$ works better for low degree harmonics (or $n \leq \frac{1}{3}N$), i.e., the percentage rms of the error is less than for the other two formulas there, the factor $\mu_n = \frac{1}{4\pi\beta_n}$ is best for middle harmonics (approximately $1/3N < n \leq N$), and $\mu_n = \frac{1}{4\pi}$ is best above the Nyquist frequency N. It is possible, therefore, to obtain a simple "composite" quadratures formula that combines the good properties of all three formulas, by defining its de-smoothing factor as follows:

$$\mu_n = \frac{1}{4\pi\eta_n} \quad \text{where} \quad \eta_n = \begin{cases} \beta_n^2 & \text{if } 0 \leq n \leq 1/3N \\ \beta_n & \text{if } 1/3N < n \leq N \\ 1 & \text{if } n > N \end{cases} \tag{3.9}$$

This composite formula has been implemented in the version of subroutine HARMIN listed in Appendix B, through the subroutine can be easily changed to compute other quadrature formulas.

It is clear from the tables that, while better than all the others, the optimal formula is only marginally so: from a practical point of view, the simple "composite" formula (3.9) above is virtually as good, but it is much easier to implement and compute. Therefore, when analysing data of the type considered here (resembling mean anomalies with uniform noise on an equal angular grid), the quadratures formulas discussed the composite in particular, are about as good as any linear technique for estimating the coefficients, and also very easy to program and very efficient.

There may be cases, however, when this is not true. If the data set were very noisy and/or incomplete, or if the coefficients to be recovered were not those of the signal, but those of some complicated transformation of it (as in the case of satellite-to-satellite tracking data, where the signal depends on a combination of differences of radial and horizontal derivatives of the gravitational potential) so no simple integral formula like (1.4) exists that can be readily discretized into summations like (1.5) or (1.6), then simple quadrature formulas like those in section 1 would be of no real use, and the optimal estimator could provide the only practical way of obtaining the coefficients. Both collocation and least squares adjustment are very similar, as shown in section 2, and both can be implemented with reasonable efficiency if nothing simpler is available. More important, the optimal formulas provide a theoretical background on which one can build a coherent and comprehensive understanding of the other linear techniques for spherical harmonic analysis. It is, after all, because of the theory developed in the previous section that it has been possible to obtain the results shown in the preceeding tables, results that constitute the factual basis for these considerations.

The techniques for setting up and inverting the variance-covariance matrix of the data are of interest in a number of estimation and filtering problems, besides harmonic analysis. The author hopes that the examples provided in this section will encourage the wider use of least squares collocation for the processing of large, global sets of data, both point values and area means. Creating the simulated mean anomalies with subroutine SSYNTH, obtaining the "weights" χ_i^{nm} of the optimal quadratures formula with NORMAL (Appendix B), and recovering the $\hat{C}_{nm}^{\alpha}$ for comparison with the original $\overline{C}_{nm}^{\alpha}$ up to degree and order 72 (same 5000 coefficients) on a 5° x 5° degree grid (about 2600 "data" values) took less than 20 seconds, central processor unit time, in the AMDHAL computer at Ohio State. To recover the harmonic coefficients up to degree and order 180 from a complete equal angular set of 1° x 1° mean values would require less than two hours, using the same machine. Most of this time would be dedicated to creating and inverting $(C_{zz} + D)$. But in the optimal estimation and filtering of geoidal undulations, deflections of the vertical, and any other function of the gravity field estimable from (say) gravity anomalies, the fact that $(C_{zz} + D)$ can be set up and inverted efficiently transcends harmonic analysis. A major implication is that such extremely large global adjustments require a computational effort that is already within the reach of most researchers.

As mentioned already in section 2, matrix $(C_{zz} + D)$ may become poorer conditioned, i.e., its numerical inversion less stable, as the data distribution becomes denser. This tendency towards instability was noticed: when there was no noise (D = 0) the R(m) matrices had to be regularized by adding a small positive constant k to each diagonal term (Colombo, 1979, par. (4.5)) before they could be inverted successfully. This constant, which in most cases was much smaller than the diagonal elements it was being added to, was 10^{-6} for 30° x 30° and 10° x 10°, but had to be increased to 10^{-3} in the case of a 5° x 5° grid. When noise was present, the nonzero diagonal elements in D were sufficient to provide stability, and no regularization was needed. Because of this tendency to instability, the "band-limited" approach of paragraphs (2.13) to (2.17) may be preferable, whenever it can be properly applied.

As shown in Table (3.1), the theoretical and the actual sampling errors are almost the same in most cases. The propagated noise measure is very easy to compute in the case of uncorrelated noise, and can be added to the actual sampling error (variance) to get an estimate of the total error, both actual and theoretical. This estimate, where the sampling part is the result of a Montecarlo-like approach, is much easier to obtain than the theoretical one that involves setting up matrix $(C_{zz} + D)$, or at least the $R(m)$ matrices. In the case of equal angular data sets like those considered here, this empirical estimate is likely to be just as accurate, when it comes to judge the performance of any given type of harmonic analysis. Such estimate has been used to evaluate the likely errors in the potential coefficients obtained from $1^\circ \times 1^\circ$ mean anomalies using the quadratures formula $\hat{C}^{\alpha}_{nm} = \frac{1}{4\pi\beta_n} \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \int_{\sigma_{ij}} \bar{Y}^{\alpha}_{nm}(\theta,\lambda) d\sigma \, \overline{\Delta g}_{ij}$. The Montecarlo method described in paragraph (3.1) was implemented with the help of subroutines SSYNTH and HARMIN, the error variance being equated to its average over three "trials" (three sets of coefficients created from different random sequences), C. Jeleli, also at O.S.U., who undertook this work as part of his own research,fitted a quartic to the percentage rms per degree thus obtained. When this was expressed as a function of the "normalized degree" n/N , the quartic fitted equally well the theoretical results for $30^\circ \times 30^\circ$, $10^\circ \times 10^\circ$, and $5^\circ \times 5^\circ$ presented in this section. Jekeli's quartic expression for the truncation error is (private communication):

$$\frac{\sigma\epsilon_n}{\sigma_n} \times 100 = [(-16.19570 (\frac{n}{N}) + 30.34506)(\frac{n}{N}) + 40.29588](\frac{n}{N})^2 \qquad (3.10)$$

It is quite remarkable that such a complex phenomenon can be described satisfactorily by such a simple law.

The expansion of the simulated $1^\circ \times 1^\circ$ mean anomalies was complete up to degree and order $N_{max} = 300$. Creating (or analysing) area mean values up to degree and order $N_{max} = 300$ required about 50 seconds c.p.u. time using double precision arithmetic in the AMDHAL computer at O.S.U.

Table 3.8 shows the actual percentage rms sampling error per degree as computed in one of the trials, and the percentage rms propagated noise (theoretical) corresponding to a 1 mgal rms noise in the data. Clearly, the errors are much smaller than for any of the cases considered previously: this improvement is due to the finer sampling (the sampling error tends to zero as the area of the blocks tends to zero). The data, however, tends to be noisier when averaged on smaller blocks, so the propagated noise may increase. For a given rms error in the data, multiply the number in the "propagated noise" column by this rms (in mgals) to obtain the corresponding percentage. These numbers are only valid for the estimator where $\mu_n = \frac{1}{4\pi\beta_n}$. Repeated trials with different random coefficients resulted in much the same percentages for the sampling errors, so these values are probably fairly typical.

Table 3.8 Percentage rms errors per degree for a 1° x 1° equal angular grid. $\mu_n = \frac{1}{4\pi \beta_n}$, $N_{max} = 300$, $\epsilon \, \Delta g = 1$ mgal		
n	actual sampling	propagated error (theor.)
2	0.01	0.47
10	0.10	0.72
20	0.46	1.39
30	0.98	1.72
40	1.55	1.93
50	2.68	2.07
60	4.71	2.20
70	6.94	2.34
80	9.43	2.47
90	13.51	2.62
100	14.79	2.77
110	17.98	2.96
120	25.06	3.08
130	25.33	3.26
140	30.87	3.43
150	37.02	3.62
160	44.17	3.81
170	44.43	4.03
180 (N)	53.07	4.24
190	61.87	4.47
200	65.90	4.73

A possible way of bringing the sampling error down is to use <u>weighted area means</u> of the form

$$\Delta \bar{g}_{ij} = \sum_{k=1}^{N_{ij}} \Delta g_{ij}^{(k)} W_k$$

where $\Delta g_{ij}^{(k)}$ is the kth measurement inside the block σ_{ij}, and where the W_k are functions of the distance of $\Delta g_{ij}^{(k)}$ to the center of the block. If the W_k decay gently towards the border of the area element, the resulting weighted means will be smoother than the ordinary area means considered so far (all $W_k = N_{ij}^{-1}$) and their harmonic content above the Nyquist frequency will be atenuated. Consequently, the harmonics below N can be recovered with less sampling error. This idea certainly deserves further study. Obviously, it is applicable only in those cases where the original measurements $\Delta g_{ij}^{(k)}$ are available.

3.4 The Analysis of a Global Data Set of $5^\circ \times 5^\circ$ Mean Anomalies

As a final demonstration of the use of optimal estimators, this paragraph presents some results obtained by analysing a real data set, consisting of $5^\circ \times 5^\circ$ mean gravity anomalies. These anomalies were obtained from a global set of $1^\circ \times 1^\circ$ mean values created by R. Rapp and associates (Rapp, 1978) from the combination of land and gravity measurements, satellite altimetry over the oceans, and the GEM-9 satellite model (Lerch et al., 1979).

The $5^\circ \times 5^\circ$ values, $\overline{\Delta g}_{(5^\circ)}$, were obtained from the $1^\circ \times 1^\circ$ values, $\overline{\Delta g}_{(1^\circ)}$, using the formula

$$\overline{\Delta g}_{(5^\circ)} = \frac{1}{25} \sum_{ij} \overline{\Delta g}_{(1^\circ)ij}$$

where $\sum_{ij}$ represents summation over all $1^\circ \times 1^\circ$ blocks inside a $5^\circ \times 5^\circ$ block. The variances of the $\overline{\Delta g}_{(5^\circ)}$ were obtained from the formula

$$\sigma^2 \overline{\Delta g}_{(5^\circ)} = \frac{1}{(25)^2} \sum_{ij} \sigma^2 \overline{\Delta g}_{(1^\circ)ij}$$

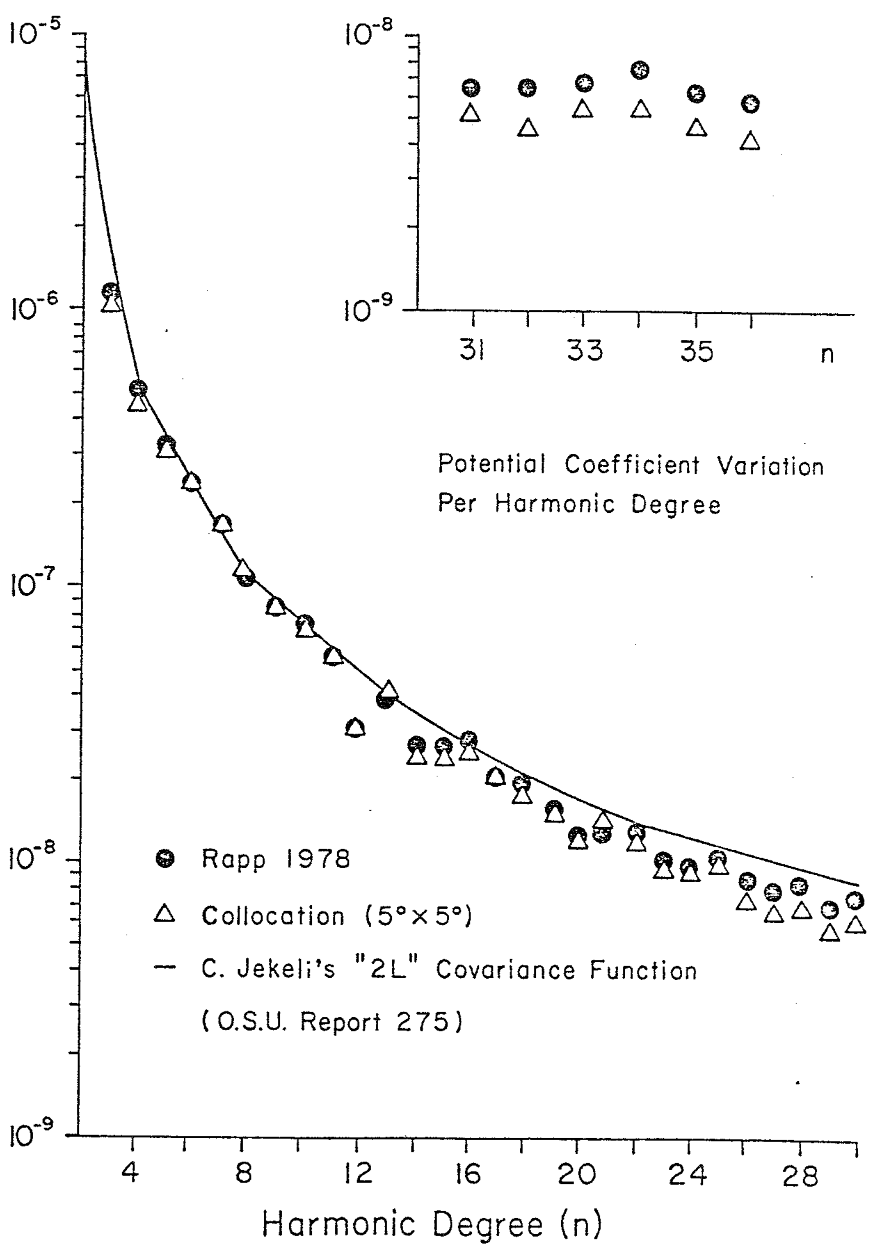
where the assumption has been made that the errors in the $1^\circ \times 1^\circ$ values were uncorrelated, which is not quite true, as the values used were the product of an adjustment. The variances $\sigma^2 \Delta g_{(5^\circ)}$ were different from 5° block to 5° block, but they were "homogeneized" as described in paragraph (2.9), in order to obtain a $(C_{zz} + D)$ matrix that was easily invertible and resulted in a quadratures-type estimator inexpensive to implement.

Figure (3.1) shows a comparison between the collocation solution, complete to degree and order 36, and the same coefficients obtained by R. Rapp by numerical quadratures from the original $1^\circ \times 1^\circ$ values. The figure shows the rms per degree ($\sqrt{\sigma_n^2/(2n+1)}$) for potential coefficients ($\sigma_n^2(T) = \sigma_n^2(\overline{\Delta g})\, \gamma^{-2} (n-1)^{-2}$, $\gamma = 979800$ mgal). The circles correspond to Rapp's results, and the triangles to those obtained using collocation as already explained. Because the grid used by Rapp is much finer, the corresponding results are likely to be less affected by sampling errors, at least up to degree 36, than those obtained from the $5^\circ \times 5^\circ$ anomalies; for this reason Rapp's rms values are regarded here as the "true" ones. The solid line corresponds to C. Jeleki's "2L" model for the σ_n^2, used here to obtain the optimal estimator. It is interesting to notice that the triangles follow the circles (or "true" values) rather than the line. A common concern among those using this type of estimators is to what extent the "a priori" power spectrum or covariance function used to set up the estimator may "bias" the results by forcing the spectrum of the output to resemble the "a priori" spectrum. Here there is little evidence of such a "bias".

In addition to Rapp's coefficients, those of GEM-10B (Lerch, 1980) were also used for comparison. The "collocation" values follow them very closely too (the three sets of results agree, in fact, very well with each other). The

GEM values are not shown in the figure, to have a cleaner picture. In any event, the "collocation" values compare very well with the other two, and in the higher degrees (31 to 36) where the divergence between Rapp's model and collocation is largest, GEM-10B fits right in between them. The results shown here were presented in a previous paper by the author (Colombo, 1979-b).

Figure 3.1



4. Covariances Between Area Means

In what follows, the expression "point" covariance refers to

$$\operatorname{cov}(u(P), v(P')) = \sum_{n=0}^{\infty} c_n^{u,v} P_n(\psi_{PP'}) = M\{u(P), v(P')\} \tag{4.1}$$

The covariance between the area means of two functions u and v is related to the "point" covariance function by the following integral relationship (Sjöberg, 1978):

$$\operatorname{cov}(\bar{u}_{ij}, \bar{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \int_{\sigma_{ij}} d\sigma \int_{\sigma_{kl}} \operatorname{cov}(u(P), v(P'))\, d\sigma' \tag{4.2}$$

where P belongs to the area block σ_{ij}, and P' to σ_{kl}, while

$$\bar{u}_{ij} = \frac{1}{\Delta_{ij}} \int_{\sigma_{ij}} u(P)\; d\sigma \tag{4.3}$$

is the average of u over the block σ_{ij} of area

$$\Delta_{ij} = \Delta\lambda\,(\cos(\theta_i) - \cos(\theta_i + \Delta\theta_i)) \tag{4.4}$$

$\Delta\theta_i$, $\Delta\lambda$ are the colatitude and longitude spans of the blocks in the "row" between colatitudes θ_i and $\theta_i + \Delta\theta_i$. To simplify the discussion, $\Delta\theta_i = \Delta\theta$ is assumed constant; extension to the more general case where $\Delta\theta_i$ varies from row to row is straightforward.

4.1. Derivation of an Approximate Formula for the Covariance[1]

Expression (4.2) can be computed by numerical quadratures (Rapp, 1977). Because of the double integral, this is a very laborious process, and it is not practical if many covariance values have to be found, as in the case of large data sets. A more efficient alternative is needed.

Replacing the covariance function in the integrand of (4.2) with its Legendre expansion

$$\operatorname{cov}(\bar{u}_{ij}, \bar{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \int_{\sigma_{ij}} d\sigma \int_{\sigma_{kl}} \sum_{n=0}^{\infty} c_n^{u,v} P_n(\psi_{PP'}) \tag{4.5}$$

Here $\psi_{PP'} = \cos^{-1}(\cos\theta \cos\theta' + \sin\theta \sin\theta' \cos(\lambda - \lambda'))$ is the spherical distance between points $P \equiv (\theta, \lambda)$, and $P' \equiv (\theta', \lambda')$ in the unit sphere, while

$$c_n^{u,v} = \sum_{m=0}^{n} \bar{C}_{nm}^{u}\, \bar{C}_{nm}^{v} + \bar{S}_{nm}^{u}\, \bar{S}_{nm}^{v}$$

is the nth degree variance of the cross-spectrum of u and v. As shown in the Appendix, the order of summation and integration can be reversed.

[1] In this section, expression $P_n(\psi_{PP'})$ is shorthand notation for $P_n(\cos\psi_{PP'})$.

$$\mathrm{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \sum_{n=0}^{\infty} c_n^{u,v} \int_{\sigma_{ij}} d\sigma \int_{\sigma_{kl}} P_n(\psi_{PP'})\, d\sigma' \tag{4. 6}$$

Let $P_{nm}(\cos\theta)$ be the fully normalized associate Legendre function of the first kind of degree n and order m. Applying the Summation Theorem for such functions, the $P_n(\psi_{PP'})$ can be replaced in the integrand as shown below.

$$\mathrm{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \sum_{n=0}^{\infty} \frac{c_n^{u,v}}{2n+1} \int_{\sigma_{ij}} d\sigma \int_{\sigma_{kl}} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda-\lambda')\, d\sigma' \tag{4. 7}$$

The sum in the general term of the series above has a finite number of terms (n+1), so it is valid to exchange summation and integration once more:

$$\mathrm{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \sum_{n=0}^{\infty} \frac{c_n^{u,v}}{2n+1} \sum_{m=0}^{n} \int_{\sigma_{ij}} d\sigma \int_{\sigma_{kl}} P_{nm}(\cos\theta)\, P_{nm}(\cos\theta') \cos m(\lambda-\lambda')\, d\sigma' \tag{4. 8}$$

Writing out the area integrals as colatitude and longitude integrals:

$$\mathrm{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \sum_{n=0}^{\infty} \frac{c_n^{u,v}}{2n+1} \sum_{m=0}^{n} \int_{\theta_i}^{\theta_i+\Delta\theta_i} \overline{P}_{nm}(\cos\theta)\, \sin\theta\, d\theta \;\cdot \tag{4. 9}$$

$$\cdot \int_{\theta_k}^{\theta_k+\Delta\theta_k} \overline{P}_{nm}(\cos\theta')\, \sin\theta'\, d\theta' \int_{\lambda_j}^{\lambda_j+\Delta\lambda} d\lambda \int_{\lambda_l}^{\lambda_l+\Delta\lambda} \cos m(\lambda-\lambda')\, d\lambda'$$

Calling $\Delta_i \equiv \Delta_{ij}$, as all blocks in the same "row" have the same area, we introduce two functions,

$$F(m) = \begin{cases} \Delta\lambda^2 & \text{if } m=0 \\ (2/m^2)(1-\cos m\Delta\lambda) & \text{otherwise,} \end{cases} \tag{4. 10-a}$$

and

$$I_{nm,i} = \left(\frac{c_n^{u,v}}{2n+1}\right)^{\frac{1}{2}} \frac{1}{\Delta_i} \int_{\theta_i}^{\theta_i+\Delta\theta_i} \overline{P}_{nm}(\cos\theta)\, \sin\theta\, d\theta \tag{4. 10-b}$$

Then expression (4.9) can be written, after integrating and reordering terms, as follows:

$$\mathrm{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \sum_{m=0}^{\infty} F(m) \sum_{n=m}^{\infty} I_{nm,i}\, I_{nm,k} \cos m(\lambda_l-\lambda_j) \tag{4. 11-a}$$

This regrouping of the series is valid because, as shown in the Appendix, the series is absolutely convergent. The integrals in the definition of $I_{nm,i}$ (expression (4.10-b)) can be calculated very accurately and efficiently with recursive formulae obtained by M. K. Paul (1979). Regarding j as being fixed, the last expression is also that of the Fourier series of cosines of $\mathrm{cov}(u_{ij}, v_{kl})$, with amplitudes

$$a_m^{i,k} = F(m) \sum_{n=m}^{\infty} I_{nm,i}\, I_{nm,k} \tag{4. 11-b}$$

and phases $\varphi_m = -m\lambda_j$.

While (4.11-a, b) are valid for all values of λ_j in the interval $0 \leq \lambda_j \leq 2\pi$, this expression has to be calculated only for $\lambda_j = 0, \Delta\lambda, 2\Delta\lambda, \ldots, j\Delta\lambda$. According to the sampling theorem for ordinary Fourier Series, at such regularly spaced λ_j expression (4.11-a) takes precisely the same values as the finite sum

$$\mathrm{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = \sum_{m=0}^{N} \hat{a}_m^{ik} \cos m(\lambda_j - \lambda_l) \tag{4.12}$$

where $N = \pi/\Delta\lambda$

$$\hat{a}_m^{ik} = \sum_{h=0}^{\infty} a_{2Nh+m}^{ik} + \sum_{h=1}^{\infty} a_{2Nh-m}^{ik} \tag{4.13}$$

To calculate the covariances, (4.11-a, b) and (4.13) have to be truncated, excluding all harmonics above some degree N_{max}; (4.12) becomes:

$$\mathrm{cov}(\overline{u}_{ij}, \overline{v}_{kl}) \simeq \mathrm{cov}(\overline{u}_{ij}, \overline{v}_{kl})_{N_{max}} = \sum_{m=0}^{N} \Bigg[\sum_{h=0}^{K} \sum_{n=m}^{N_{max}} (I_{n,2Nh+m,i})(I_{n,2Nh+m,k})$$

$$+ \sum_{h=1}^{K} \sum_{n=m}^{N_{max}} (I_{n,2Nh-m,i})(I_{n,2Nh-m,k}) \Bigg] F(m) \cos m(\lambda_j - \lambda_l) \tag{4.14}$$

where $(2K+1)N \leq N_{max}$.

4.2. Choosing N_{max}

To calculate the values of covariances between equal angular mean anomalies, $\overline{\Delta g}$, the value of N_{max} could be chosen so that the percentage error

$$v = \left| \frac{\mathrm{cov}(\overline{\Delta g}_{ij}, \overline{\Delta g}_{kl})_Q - \mathrm{cov}(\overline{\Delta g}_{ij}, \overline{\Delta g}_{kl})_{N_{max}}}{\mathrm{cov}(\overline{\Delta g}_{ij}, \overline{\Delta g}_{kl})_Q} \right| \times 100$$

($\mathrm{cov}(\overline{\Delta g}_{ij}, \overline{\Delta g}_{kl})_Q$ being computed by numerical quadratures) does not exceed a prescribed upper bound ϵ. The smallest N_{max} that meets this condition increases towards the poles, because the decrease in area of equal angular blocks with latitude means that the averages have a high frequency content that increases accordingly. On the other hand, the absolute values of the integrals of the $\overline{P}_{nm}$, and therefore the Fourier coefficients a_m in (4.12), decrease quite fast with increasing order m near the poles, so their contribution soon becomes insignificant. This is fortunate, because the need for lengthier calculations for each Fourier coefficient nearer to the poles can be offset by the existence of fewer coefficients there. In fact, because of the finite arithmetic of digital computers, all $\hat{a}_m$ for blocks less than 30° from the poles are rounded off to zero for m considerably less than N in the cases presented here. Because of this, calculations near the poles can be less laborious than close to the equator, in spite of the larger N_{max}. To take advantage of this in the programming of (4.14), the $\hat{a}_m$ coefficients were not computed above the first m for which the following condition was met:

$$\frac{\hat{a}^2_{m-1}}{\sum_{\ell=0}^{n-1} \hat{a}^2_{\ell}} \leq \delta \quad \text{and} \quad \frac{\hat{a}^2_{m}}{\sum_{\ell=0}^{m} \hat{a}^2_{\ell}} \leq \delta ,$$

where $\delta \leq 10^{-12}$. This ensured no change in the first six significant figures of the result when compared to the case where no coefficients were ignored, but there were great savings in computing.

4.3. Numerical Examples

To verify the accuracy of the truncated series in expression (4.14), mean gravity anomaly covariances were computed both with this formula and by numerical quadratures. All calculations were carried out in double precision (32 bytes), some of those for expression (4.14) being repeated in extended precision (64 bytes). The agreement between both sets of results was better than 6 significant figures, suggesting that both expression (4.14) and Paul's recursives (used to obtain the $I_{nm,i}$) are quite stable numerically at all latitudes.

The covariances computed were between a mean anomaly in a fixed block and all the anomalies in the same row of blocks (i.e., all blocks bounded by the same parallels), calculations being done for several rows, at close intervals from equator to pole. Results for only a few of those will be shown here, because they are typical of the rest. Both a 5° and a 1° grid (equal angular) were studied. The results were compared to those obtained by numerical quadratures, and by the approximate formula

$$\text{cov}(\overline{\Delta g}_{ij}, \overline{\Delta g}_{kl}) \simeq \sum_{n=0}^{N_{max}} \beta_{n,i}\, \beta_{n,k}\, c_n^{\Delta g, \Delta g}\, P_n(\psi_{YY'}) \tag{4.15}$$

where Y and Y' are the center points of σ_{ij} and σ_{kl}, while

$$\beta_{n,i} = \frac{1}{1 - \cos\psi_0} \frac{1}{2n+1} [P_{n-1}(\cos\psi_0) - P_{n+1}(\cos\psi_0)]$$

is the nth degree Pellinen smoothing factor (the formula is Meissl's), and

$$\psi_0 = \cos^{-1}\left[\frac{\Delta\lambda}{2\pi}(\cos(\theta_i + 1) - \cos\theta_i) + 1\right]$$

This formula gives the covariances of averages on circular blocks of the same area as that of the equal angular blocks in the row between colatitudes θ_i and θ_{i+1}. These covariances are used sometimes as approximations to the equal angular covariances between mean values.

The numerical quadratures technique consists in the following: (a) subdividing each block with a grid of k equally spaced latitude and k equally spaced longitude lines; (b) computing the covariances between point gravity anomalies at the nodes of each subdivision (there are k^4 different pairs of nodes to be considered); (c) obtaining the approximate value of the covariance between mean anomalies as

$$\mathrm{cov}(\overline{\Delta g}_{ij},\overline{\Delta g}_{kl}) = \sum_{m=0}^{k-1}\sum_{n=0}^{k-1}\sum_{r=0}^{k-1}\sum_{s=0}^{k-1} \mathrm{cov}(\Delta g_{ij}^{mn},\Delta g_{kl}^{rs}) \tag{4.16}$$

where m, n, r, and s are indexes that identify the elements of each pair of nodes in the subdivisions of σ_{ij} and σ_{kl}. The covariances between point anomalies such as Δg_{ij}^{mn} and Δg_{kl}^{rs} were obtained using a two term model for the degree variances of the anomalies:

$$c_n^{\Delta g,\Delta g} = 18.3906\frac{(n-1)}{(n+100)}\ 0.9943667^{n+2} + 658.6132\frac{(n-1)}{(n+20)(n-2)}\ 0.9048949^{n+2}$$

This is the "2L" model of C. Jekeli (1978), and has the advantage that the value of the point covariance function can be computed using finite recursion formulas (Moritz, 1977). The same degree variances were used in (4.14) and (4.15). The number of point covariances being very large (k^4), a table with entries spaced at $\Delta\psi = 0.05°$ intervals was created first, the required values being obtained by linear interpolation from this table. Numerical tests showed that $k = 10$ was large enough for both 5° and 1° blocks, because doubling this number resulted in a change of less than 0.2% in the values given by (4.16). Reducing the interval $\Delta\psi$ from 0.05° to 0.005° had a negligible effect also, therefore the values obtained with (4.16) are probably accurate enough to test those given by (4.14). The only exception was the "polar" row, where the equal angular blocks are, in fact, triangles with a common vertex at the pole. Both with 5° and 1° blocks the discrepancies between (4.14) and (4.16) were large (more than 30%), regardless of how large a k, how small a $\Delta\psi$, or how big a N_{max} were chosen. The probable explanation is that the pole is given undue weight in (4.16) because it is treated as a whole row, instead of as a single point. For this reason, the numerical examples presented here stop at the row immediately below the pole.

The first two tables show the covariances between 5° mean anomalies in the row between latitudes 0° and 5° (just above the equator) and in the row between latitudes 80° and 85° (one below the pole). The error is at most 8%, though much less in most cases, and $N_{max} = 180$ in each table. Under "Pellinen" one finds the values obtained using (4.15) with due regard for the change in block areas with latitude. While there is very good agreement near the equator, there is no resemblance at all close to the pole to the other values listed.

Table 4.1. Comparison between covariances of 5° mean anomalies $\overline{\Delta g}$ computed with expressions (4.14), (4.16) and (4.15), respectively. $N_{max} = 180$. Row between 0° and 5°.

Expression (4.14)	Numerical	Pellinen	Block No.
250.53	253.77	251.78	1
148.80	149.46	151.03	2
193.93	93.95	93.90	3
57.16	57.15	57.10	4
31.80	31.76	31.67	5
13.94	13.93	13.86	6
-18.09	-18.09	-18.08	12
9.12	9.12	9.11	24
-13.68	-13.68	-13.65	36

Table 4.2. Comparison between covariances of 5° mean anomalies $\overline{\Delta g}$ computed with expressions (4.14), (4.16) and (4.15), respectively. $N_{max} = 180$. Row between 80° and 85°.

Expression (4.14)	Numerical	Pellinen	Block No.
418.60	437.23	835.53	1
329.03	318.27	800.09	2
220.91	229.17	709.03	3
191.75	196.40	592.19	4
179.79	179.70		5
165.58	168.46		6
120.60	123.73		12
69.89	73.27		24
54.66	58.01	149.40	36

The next three tables show results for 1° blocks. Here the numerical method was conducted with the same $\Delta\psi$ and k as in the case of the 5° grid. Results for rows between latitudes 0° and 1°, 45° and 46°, and 88° and 89° are shown. In tables 4.3 and 4.4 the discrepancies between (4.14) and (4.16) stay below 1%; this increases to about 5% near the pole (table 4.5). N_{max} is 300 for the equatorial row, and rises to 400 from 45° on. As with 5°-blocks, the "Pellinen" values are quite close to these of (4.14) and (4.16) near the equator, but become very different near the pole.

Table 4.3. Comparison between covariances of 1° mean gravity anomalies computed with (4.14) and (4.16). $N_{max} = 300$. Row between 0° and 1°.

Expression (4.14)	Numerical	Block No.
849.68	855.16	1
411.32	410.30	2
220.35	219.63	3
181.84	181.50	4
163.33	163.21	5
149.11	149.15	6
1.07	1.08	31
-16.79	-16.79	61
1.99	1.99	91
8.54	8.53	121
-4.79	-4.79	151
-14.39	-14.38	181

Table 4.4. Comparison between covariances of 1° mean gravity anomalies computed with (4.14) and (4.16). $N_{max} = 400$. Row between 45° and 46°.

Expression (4.14)	Numerical	Block No.
952.25	959.17	1
531.15	531.97	2
275.55	274.65	3
210.81	210.46	4
185.36	185.57	5
170.82	171.08	6
27.79	27.79	31
-14.37	-14.37	61
-18.01	-17.01	91
-8.27	-8.28	121
-1.01	-2.92	151
1.39	1.39	181

Table 4.5. Comparison between covariances of 1° mean gravity anomalies computed with (4.14), (4.16), and (4.15), respectively. $N_{max} = 400$. Row between 88° and 89°.

Expression (4.14)	Numerical	Pellinen	Block No.
1137.97	1187.94	1795.88	1
1135.79	1184.15	1795.02	2
1129.28	1177.99	1792.45	3
1118.55	1155.08	1788.18	4
1103.76	1131.34	1782.25	5
1085.15	1102.85	1774.68	6
408.59	443.12		31
248.27	257.88		61
207.19	210.64		91
189.65	192.55		121
181.93	184.53		151
179.65	179.86	368.60	181

Calculations were carried out in the Ohio State University's AMDHAL 470 V/6-II computer, using the FORTRAN H EXTENDED compiler, and double precision. The computing times for obtaining all 37 different covariances between the elements of a row of 5° mean anomalies, and all 181 covariances in a row of 1° mean anomalies, using (4.14) and (4.16) are listed in table 4.6 for comparison. The integrals of the Legendre functions required by (4.14), and the table of point anomalies covariances needed in (4.16), were precomputed and stored in core memory arrays, so the times given here do not include the determination of those auxiliary values. In most ordinary applications of these formulas, those quantities can be read from disk or tape whenever needed, because they are the same for a whole variety of problems. Clearly, using (4.14) can be orders of magnitude more efficient than using (4.16), while the accuracy is much the same. In fact, accuracy is probably better at the polar rows with (4.14) than with (4.16), because the latter seems to have problems handling triangular blocks. Finally, not all of the time-saving properties of (4.14) were exploited in the computer program used to calculate it, so there is scope for some improvement in efficiency beyond that shown in the table. Notice the time saved in the $88^\circ - 89^\circ$ row thanks to the neglect of terms in (4.14) that become too small near the poles, as explained in paragraph (4.2).

Table 4.6. Comparative efficiencies of algorithms based on expressions (4.14) and (4.16).

Block Size	Row	N_{max}	Expression (4.14)	Numerical Quadratures
5°	0° - 5°	180	0.1 sec.	23.3 sec.
5°	80°-85°	180	0.1 sec.	23.3 sec.
1°	0° - 1°	300	0.35 sec.	113.3 sec.
1°	45°-46°	400	0.43 sec.	113.3 sec.
1°	88°-89°	400	0.06 sec.	113.3 sec.

4.4. Covariances Between Mean Values and Point Values

The prediction by least squares collocation of mean values from point values, or vice versa, requires finding the corresponding "mixed" covariances. In such case, formula (4.2) becomes

$$\operatorname{cov}(\bar{u}_{ij}, v(\theta,\lambda)) = \frac{1}{\Delta_{ij}} \int_{\sigma_{ij}} \operatorname{cov}(u(P'), v(P))\, d\sigma \tag{4.17}$$

where $P' \equiv (\theta',\lambda') \in \sigma_{ij}$, $P \equiv (\theta,\lambda)$, $v(P)$ is the "point" value of v, and $\bar{u}_{ij}$ is the average over the i,j block of the grid, as before. Following a similar reasoning, one arrives at a formula that corresponds to (4.14), except that only one area integral has to be considered. The new expression is:

$$\operatorname{cov}(\bar{u}_{ij}, v(\theta,\lambda)) \simeq \frac{1}{\Delta_{ij}} \sum_{m=0}^{N} \Bigg[\sum_{h=0}^{K} \sum_{n=m}^{N_{max}} \left(\frac{c_n^{u,v}}{2n+1}\right)^{\frac{1}{2}} (I_{n,2Nh+m,i}\ \bar{P}_{n,2Nh+m}(\cos\theta))$$
$$+ \sum_{h=1}^{K} \sum_{n=m}^{N_{max}} \left(\frac{c_n^{u,v}}{2n+1}\right)^{\frac{1}{2}} (I_{n,2Nh-m,i}\ \bar{P}_{n,2Nh-m}(\cos\theta)\Bigg] \Big[(A(m)\cos m(\lambda_j - \lambda)$$
$$+ B(m)\sin m(\lambda_j - \lambda)\Big] \tag{4.18}$$

where

$$A(m) = \begin{cases} \Delta\lambda & \text{if } m = 0 \\ (\cos m\Delta\lambda - 1)/m & \text{otherwise} \end{cases} \qquad B(m) = \begin{cases} 0 & \text{if } m = 0 \\ (\sin m\Delta\lambda)/m & \text{otherwise} \end{cases}$$

and the $I_{n,m,i}$, the $c_n^{u,v}$, and the N, K, and N_{max} are as in (4.14). Expression (4.18) assumes that the area means belong to a grid with constant $\Delta\lambda$. If the point values are also on a grid, and if this grid is congruent with that of the mean anomalies, implementation of (4.18) is quite efficient. In fact, the speed of a good algorithm for doing this should be much the same as that of one for implementing (4.14). On the other hand, computing the same covariances by numerical quadratures is k^2 times faster in this "mixed" case than it was in the previous one, because there is only one area integration involved. Assuming $k = 10$, as in the previous examples, then expression (4.18) should be only 2-3 times faster than numerical quadratures for 5° mean values, and from 4 to 15 times faster in the case of 1° averages.

Both the normalized Legendre function and their integrals are needed for (4.18). They can be precomputed and stored on disk or tape until needed. In this study the following recursive relationships were used to generate their values:

$$\overline{P}_{n-1,m-1}(\cos\theta) = \left[\frac{(2n-1)(2n-3)}{(n-m)(n+m-2)}\right]^{\frac{1}{2}}(\cos\theta\,\overline{P}_{n-2,m-1}(\cos\theta)$$

$$-\left[\frac{(2n-1)(n+m-3)(n-m-1)}{(2n-5)(n+m-2)(n-m)}\right]^{\frac{1}{2}}\overline{P}_{n-3,m-1}(\cos\theta), \quad m \neq n \qquad (4.19a)$$

$$\overline{P}_{n-1,n-1}(\cos\theta) = \left[\frac{(2n-1)}{(2n-2)}\right]^{\frac{1}{2}}\sin\theta\,\overline{P}_{n-2,n-2}(\cos\theta) \qquad (4.19b)$$

$$\int_{\theta_1}^{\theta_2}\overline{P}_{n-1,m-1}(\cos\theta)\sin\theta\,d\theta = -\frac{1}{n}\left[\frac{(2n-1)(2n-3)}{(n-m)(n+m-2)}\right]^{\frac{1}{2}}\sin^2\theta\,\overline{P}_{n-2,m-1}(\cos\theta)\Big|_{\theta_1}^{\theta_2} +$$

$$+\frac{(n-3)}{n}\left[\frac{(2n-1)(n+m-3)(n-m-1)}{(2n-5)(n+m-2)(n-m)}\right]^{\frac{1}{2}}\int_{\theta_1}^{\theta_2}\overline{P}_{n-3,m-1}(\cos\theta)\sin\theta\,d\theta \qquad (4.20a)$$

$$\int_{\theta_1}^{\theta_2}\overline{P}_{n-1,n-1}(\cos\theta)\sin\theta\,d\theta = \frac{1}{2n}\left[\frac{(2n-1)}{(n-1)(n-2)}\right]^{\frac{1}{2}}\sin^2\theta\;\overline{P}_{n-2,n-3}(\cos\theta)\Big|_{\theta_1}^{\theta_2} +$$

$$+\frac{1}{2n}\left[\frac{(n-1)(2n-1)(2n-3)}{(n-2)}\right]^{\frac{1}{2}}\int_{\theta_1}^{\theta_2}\overline{P}_{n-3,n-3}(\cos\theta)\sin\theta\,d\theta \qquad (4.20b)$$

if θ is not very small, or

$$\int_{\theta_1}^{\theta_2}\overline{P}_{n-1,n-1}(\cos\theta)\sin\theta\,d\theta = -\left[\frac{(2n-1)(2n-3)\ldots3}{(2n-2)(2n-4)\ldots4}\right]^{\frac{1}{2}}\sin^{n+1}\theta\;\cdot$$

$$\cdot\left[\frac{1}{n+1}+\frac{1}{2}\frac{\sin^2\theta}{n+3}+\frac{1\cdot3}{2\cdot4}\frac{\sin^4\theta}{n+5}+\frac{1\cdot3\cdot5}{2\cdot4\cdot6}\frac{\sin^6\theta}{n+7}+\ldots\right]\Big|_{\theta_1}^{\theta_2} \qquad (4.20c)$$

if θ is very small. Here $y(\theta)\Big|_{\theta_1}^{\theta_2} \equiv y(\theta_2) - y(\theta_1)$. The recursive formulas for the integrals of the Legendre functions were derived by M.K. Paul (1978); the author has been fortunate enough to have available a FORTRAN subroutine programmed by Paul, and kindly sent by him to Professor R.H. Rapp of the Department of Geodetic Science at O.S.U. The results reported here have been made possible by, and bear witness to, the great numerical stability of Paul's formulas.

5. Conclusions

The relationships between spherical harmonic series and Fourier series, coupled to the symmetries of spherical grids, permit the development of efficient algorithms for numerical analysis of data regularly sampled on the sphere.

The algorithms presented in section 1 for implementing numerical quadratures are efficient enough to allow the analysis of 64800 $1^\circ \times 1^\circ$ mean values through degree 180 in less then 20 seconds, and the summation of the 90000 terms of a harmonic series complete through degree 300, on a full $1^\circ \times 1^\circ$ equal angular grid, in less than 1 minute. The analysis of large global data sets to a very high resolution is a relatively trivial operation with modern digital computers.

The principles and ideas behind the optimal estimators of section 2 provide a rational basis for the study of linear techniques for spherical harmonic analysis, both optimal and non-optimal. The error measure introduced in this section is shown to be a very reasonable way of evaluating the estimation error, as illustrated by the results listed in table (3.1).

The optimal estimators themselves are reasonably easy to compute and use, particularly when they are of the quadratures type, which happens under the fairly general conditions discussed in section 2. Even when such conditions are not present, as in the case of scattered data, the problem still has a structure strong enough to allow efficient algorithms for creating and inverting the normal matrix.

The separation of the problem of estimating the coefficients (by least squares collocation or by least squares adjustment) according to the order m of the coefficients, allows both for efficiency and for numerical stability. Even if the total number of unknown coefficients is very large, the largest matrix to be inverted is of dimension N, instead of $O(N^2)$, as it would be if the problem could not be separated in this way.

All the algorithms presented here, when the grid is complete and regular, are well suited to parallel processing.

In the case of full grids of mean values with even noise, the results of section 3 suggest that the optimal "collocation" estimator can be approximated very closely by a much simpler quadratures-type formula, the "composite formula" (3.9). The search for simple, near-optimal estimators is just as important, from a practical point of view, as the search for efficient algorithms for obtaining the optimal estimators themselves. This is a topic that certainly deserves further research.

The methods for creating and inverting the normal matrix, that make possible to find optimal estimators for large data sets, have application outside spherical harmonic analysis, in all areas of estimation, filtering and prediction on the sphere. This has been the subject of a previous report (Colombo, 1979). It must be added that the principles presented here can be generalized to bodies of revolution other than the sphere; to the case where the data are not homogeneous (i.e., a mixture of, say, gravity anomalies, satellite altimetry, etc.); to the case where the coefficients to be estimated are not those of the signal as given, but of some more or less complex linear transformation of this signal, satellite-satellite tracking data being a good example. In fact, the author is at present considering the error analysis of the determination, by least squares collocation, of the potential coefficients from satellite-satellite tracking, following some of the principles of section 2. This will be the subject of another report.

The method developed in section 4 to calculate the covariance between two area means without employing cumbersome numerical integrations is of interest, not only in spherical harmonic analysis, but more generally in filtering, prediction and estimation from mean values on the sphere.

The computer programs described and listed in Appendix B should help the interested reader to implement some of the techniques discussed in this report. The author sincerely hopes that this will be done by workers concerned with improving and further developing such methods.

Above all, the author hopes that he has conveyed to those who had read this far, the idea that the detailed analysis of very large sets of global, regularly sampled data can be done within the computing resources available today to most scientists who work at universities and research institutions everywhere. The processing, be it by numerical quadratures, or by simultaneous adjustment, of "all the data in the world" is not a fanciful thought, but a practical possibility.

References

Bibby, J., and H. Toutenberg, Prediction and Improved Estimation in Linear Models, John Wiley and Sons, New York, 1977.

Brigham, E. O., The Fast Fourier Transform, Prentice-Hall, New Jersey, 1974.

Carslaw, H.S., An Introduction ot the Theory of Fourier's Series and Integrals, 3rd. ed., Dover, New York, 1950.

Colombo, O.L., Optimal Estimation from Data Regularly Sampled on a Sphere with Applications in Geodesy, Department of Geodetic Science Report No. 291, The Ohio State University, Columbus, Ohio, 1979 (a).

Colombo, O.L., Algorithms for Spherical Harmonic Analysis, paper presented at the Fall meeting of the A.G.U., San Francisco, December 1979 (b).

Cooley, J.W., and J.W. Tuckey, An Algorithm for Machine Calculation of Complex Fourier Series, Math. Computation, Vol. 19, pp. 297-301, April 1965.

Gaposchkin, E.M., Averaging on the Surface of a Sphere, Journal of Geophysical Research, Vol. 86, No. B6., June 10, 1980.

Giacaglia G.E.O. and A. Lundquist, Smithonian Astrophysical Observatory Special Report No. 344, July 1972.

Goldstein, J.D., paper presented at the Fall Meeting of the A.G.U., San Francisco, December 1978.

Hauseholder, A.S., The Theory of Matrices in Numerical Analysis, Dover, New York, 1964.

Heiskanen, W.A., and H. Moritz, Physical Geodesy, W.H. Freeman, U.S.A., 1967.

Hobson, E.W., The Theory of Spherical and Ellipsoidal Harmonics, Chelsea, New York,second reprint, 1965.

Jekeli, C., An Investigation of Two Models for the Degree Variances of Global Covariance Functions, Dept. of Geodetic Science Report No. 275, The Ohio State University, Columbus, Ohio, 1978.

Justice, J.H., A Levinson-Type Algorithm for Two-Dimensional Wiener Filtering UsingBivariate Szegö Polynomials, Proc. IEEE, Vol. 65, No. 6, pp. 882-886, New York, 1977.

Katsambalos, K., The Effect of the Smoothing Operator on Potential Coefficients Determination, Dept. of Geodetic Science Report No. 287, The Ohio State University, Columbus, 1977.

Kaula, W.M., Statistical and Harmonic Analysis of Gravity, J. Geophysical Research, V. 64, pp. 2401-2421, U.S.A., 1959.

Kaula, W.M., Tests and Combinations of Satellite Determinations of the Gravity Field with Gravimetry, Journal of Geophysical Research, vol. 71, 11. 5303-5314, 1966.

Lauritzen, S.L., The Probabilistic Background of Some Statistical Methods in Physical Geodesy, Publication No. 48, Danish Geodetic Institute, Copenhagen.

Lerch, F.J., S.M. Klosko, R.E., Laubscher, C.A. Wagner, Gravity Model Improvement Using Geos-3 (GEM 9 & 10), Journal of Geophysical Research, Vol. 84, No. B8, July 30, 1979.

Lerch, F.J., Gravity Model Development, Earth Survey Applications Division Research Report, 1979, Ed. Lloyd Carpenter, NASA Technical Memo. 80642, Goddard SFC, Greenbelt, Maryland, Jan. 1980.

Lowes, F.J., Comments on 'The Relationship Between Mean Anomaly Block Sizes and Spherical Harmonic Representations' by R.H. Rapp, Dept. of Geophysics and Planetary Physics, University of Newcastle Upon Tyne, U.K., 1978.

Luenberger, D.G., Optimization by Vector Space Methods, Wiley & Son's N.Y., 1969.

Moritz, H., Schwerevorhersage und Ausgleichungsrechnung. Zeitschrift für Vermessungswesen, vol. 90, pp. 181-184, 1965.

Moritz, H., Advanced Least Squares Methods, Dept. of Geodetic Science, Report No. 175, The Ohio State University, Columbus, 1972.

Moritz, H., Covariance Functions in Least Squares Collocation, Dept. of Geodetic Science Report No. 240, The Ohio State University, Columbus, 1976.

Moritz, H., Statistical Foundations of Collocation, Dept. of Geodetic Science Report No. 272, The Ohio State University, Columbus, 1978.

Moritz, H., Advanced Physical Geodesy, Abacus Press, Tunbridge Wells, Kent U.K., and Herbert Wichmann Verlag, Karlsruhe, W. Germany, 1980.

Payne, M.H., Truncation Effects in Geopotential Modelling, Analytical Mechanics Associates, Seabrook, Maryland, 1971.

Paul, M.K., Recurrence Relationships for the Integrals of the Associate Legendre Functions, Bull. Geódésique, 52, 1978.

Rapp, R.H., Comparison of Two Methods for the Combination of Satellite and Gravimetric Data, Dept. of Geodetic Science Report 113, The Ohio State University, Columbus, 1968.

Rapp, R.H., Mean Gravity Anomalies and Sea Surface Heights Derived from Geos-3 Altimeter Data, Dept. of Geodetic Science Report No. 268, The Ohio State University, Columbus, 1977.

Rapp, R.H., A Global 1° x 1° Anomaly Field Combining Satellite, Geos-3 Altimeter And Terrestrial Anomaly Data, Dept. of Geodetic Science Report No. 278, The Ohio State University, Columbus, 1978.

Rapp, R.H., Potential Coefficient and Anomaly Degree Variance Modelling Revisited, Depart. of Geodetic Science Report No. 293, The Ohio State University, Columbus, 1979.

Ricardi, L.J., and J.L. Burrows, A Recurrence Technique for Expanding a Function in Spherical Harmonics, IEEE Transactions on Computers, June 1972.

Rummel, R., A Model Comparison in Least Squares Collocation, Dept. of Geodetic Science Report No. 238, The Ohio State University, Columbus, Ohio, 1976.

Rummel, R. and K.P. Schwarz, On the Non-Homogeneity of the Global Covariance Function, Bull. Géodésique, Vol. 51, No. 2, pp. 93-103, Paris, 1977.

Schwarz, K.P., Application of Collocation: Spherical Harmonics from Satellite Observations, Brosowski and Martensen, vol. 14, pp. 111-132, 1975.

Sjöberg, L., Potential Coefficient Determinations from 10° Terrestrial Gravity Data by Means of Collocation, Dept. of Geodetic Science Report No. 274, The Ohio State University, Columbus, 1978.

Tikhonov, A.N. and V.Y. Arsenin, Solutions of Ill-Posed Problems, John Wiley and Sons, N.Y., 1977.

Trench, W., An Algorithm for the Inversion of Finite Hankel Matrices, S.I.A.M. Journal of Applied Mathematics, Vol. 13, pp. 1102-1107, 1965.

Uotila, U.A., Useful Matrix Equivalences, Lecture Notes, Dept. of Geodetic Science, The Ohio State University, Columbus, 1978.

Wagner, C.A. and O.L. Colombo, Gravitational Spectra from Direct Measurements, Journal of Geophysical Research, Vol. 84, No. B9, August 10, 1979.

Appendix A. Term by Term Integration of Formula (4.5), and Rearrangement of Terms in Formula (4.9) to Arrive at Formulas (4.11-a) and (4.14)

The area integrals in expression (4.5) can be split into integrals in θ and λ, and the summation theorem for fully normalized spherical harmonics can be used to replace $P_n(\psi_{PP'})$ with an equivalent expression in the general term of the series:

$$\operatorname{cov}(\overline{u}_{ij},\overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}}\int_{\theta_i}^{\theta_i+\Delta\theta}\sin\theta\, d\theta\int_{\theta_k}^{\theta_k+\Delta\theta}\sin\theta'\, d\theta'\int_{\lambda_j}^{\lambda_j+\Delta\lambda}d\lambda\int_{\lambda_l}^{\lambda_l+\Delta\lambda}d\lambda'\sum_{n=0}^{\infty}c_n^{u,v}\cdot$$
$$\cdot\sum_{m=0}^{n}P_{nm}(\cos\theta)\, P_{nm}(\cos\theta')\cos m(\lambda-\lambda')\,(2n+1)^{-1} \qquad (A.1)$$

Starting with (A.1), the proof proceeds in four steps, each justifying in turn the taking of one of the four integrals inside the summation symbols. Each time, three theorems are invoked:

The first is the "M-Test" theorem, due to Weiertrass (see, for instance, Carslaw, (1950)):

> The series $f(x) = \sum_{n=0}^{\infty}u(x)_n$ will converge uniformly in $a\le x\le b$ if there is a convergent series of positive constants $M_0, M_1, \ldots, M_n, \ldots$ such that, for all x in $a\le x\le b$, $|u(x)_n|\le M_n$ for every positive integer n.

The second theorem is (also according to Carslaw):

> If the general term $u(x)_n$ of the series $\sum_{n=0}^{\infty}u(x)_n$ is continuous, and if the series converges uniformly to some function $f(x)$ in the interval $a\le x\le b$, then
> $$\int_a^b f(x)\,dx = \int_a^b\sum_{n=0}^{\infty}u(x)_n dx = \sum_{n=0}^{\infty}\int_a^b u(x)_n\, dx$$

This is a sufficient condition for term by term integration. The third theorem is the mean value theorem for integrals

> If $f(x)$ is analytic in $a\le x\le b$ then $\int_a^b f(x)\,dx = (b-a)f(c)$ for some c such that $a\le c\le b$.

Proof: The series in the left hand side of (A.1), if all variables but λ' are kept fixed, is uniformly convergent in the interval $\lambda_l\le\lambda'\le\lambda_l+\Delta\lambda$ because

$$(2n+1)^{-1}c_n^{u,v}\sum_{m=0}^{n}\overline{P}_{nm}(\cos\theta)\,\overline{P}_{nm}(\cos\theta')\cos m(\lambda-\lambda') = c_n^{u,v}P_n(\psi_{PP'})$$

and $|c_n^{u,v}P_n(\psi_{PP'})| \le c_n^{u,v}$ because $\max\limits_{0\le\psi\le\pi}|P_n(\psi)| = 1$ (the argument of P_n is real) for all n, while $\sum_{n=0}^{\infty} c_n^{u,v} = \operatorname{cov}(u(P), v(P))$ is always finite (equals the value of the "point" covariance when $\psi_{PP'} = 0$). The "M-Test" condition is satisfied so the series converges uniformly; therefore, term by term integration with respect to λ' is valid:

$$\operatorname{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \int_{\theta_i}^{\theta_i+\Delta\theta} \sin\theta \, d\theta \int_{\theta_k}^{\theta_k+\Delta\theta} \sin\theta' \, d\theta' \int_{\lambda_j}^{\lambda_j+\Delta\lambda} d\lambda \; \cdot$$

$$\cdot \sum_{n=0}^{\infty} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \int_{\lambda_l}^{\lambda_l+\Delta\lambda} \cos m(\lambda - \lambda')\, d\lambda \,(2n+1)^{-1}$$

Applying the mean value theorem to the last expression:

$$\operatorname{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = \frac{1}{\Delta_{ij}\Delta_{kl}} \int_{\theta_i}^{\theta_i+\Delta\theta} \sin\theta \, d\theta \int_{\theta_k}^{\theta_k+\Delta\theta} \sin\theta' \, d\theta' \int_{\lambda_j}^{\lambda_j+\Delta\lambda} d\lambda \; \cdot$$

$$\cdot \sum_{n=0}^{\infty} c_n^{u,v} \sum_{m=0}^{n} \Delta\lambda \; \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda - \lambda_Q)\, (2n+1)^{-1}$$

where $\lambda_l \le \lambda_Q \le \lambda_l + \Delta\lambda$. Removing the common factor $\Delta\lambda$ from the summation:

$$\operatorname{cov}(\overline{u}_{ij} \overline{v}_{kl}) = \frac{\Delta\lambda}{\Delta_{ij}\Delta_{kl}} \int_{\theta_i}^{\theta_i+\Delta\theta} \sin\theta \, d\theta \int_{\theta_k}^{\theta_k+\Delta\theta} \sin\theta' \, d\theta' \int_{\lambda_j}^{\lambda_j+\Delta\lambda} d\lambda \; \cdot$$

$$\cdot \sum_{n=0}^{\infty} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda - \lambda_Q)(2n+1)^{-1} \qquad \text{(A.2)}$$

The general term in the partially integrated series is

$$(2n+1)^{-1} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda - \lambda_Q) = c_n^{u,v} P_n(\psi_{PQ})$$

where $Q \equiv (\theta', \lambda_Q)$; now $|c_n^{u,v} P_n(\psi_{PQ})| \le c_n^{u,v}$ for all n, and for all λ in the interval $\lambda_j \le \lambda \le \lambda_j + \Delta\lambda$, so the "M-Test" is satisfied again and the series is uniformly convergent, and thus integrable, with respect to λ. Therefore

$$\operatorname{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = \frac{\Delta\lambda^2}{\Delta_{ij}\Delta_{kl}} \int_{\theta_i}^{\theta_i+\Delta\theta} \sin\theta \, d\theta \int_{\theta_k}^{\theta_k+\Delta\theta} \sin\theta' \, d\theta \sum_{n=0}^{\infty} c_n^{u,v} \; \cdot$$

$$\cdot \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda_R - \lambda_Q)\, (2n+1)^{-1} \qquad \text{(A.3)}$$

where $\lambda_j \le \lambda_R \le \lambda_j + \Delta\lambda$. Once more the general term of the twice integrated series satisfies the "M-Test", because

$$\left|(2n+1)^{-1} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta') \cos m(\lambda_R - \lambda_Q)\right| = \left|c_n^{u,v} P_n(\psi_{RQ})\right| \le c_n^{u,v}$$

(where $R \equiv (\theta, \lambda_R)$) for all n and, in particular, for θ' in the interval

$\theta_k \le \theta' \le \theta_k + \Delta\theta$. In consequence

$$\operatorname{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = \frac{\Delta\lambda^2 \sin\theta_S \,\Delta\theta}{\Delta_{ij}\,\Delta_{kl}} \int_{\theta_i}^{\theta_i+\Delta\theta} \sin\theta \, d\theta \sum_{n=0}^{\infty} c_n^{u,v} \,\cdot$$

$$\cdot \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta_S) \cos m(\lambda_R - \lambda_Q)\,(2n+1)^{-1} \quad \text{(A.4)}$$

Finally,

$$\left|(2n+1)^{-1} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta)\, \overline{P}_{nm}(\cos\theta_S) \cos m(\lambda_R - \lambda_Q)\right| \le \left|c_n^{u,v} P_n(\psi_{RX})\right| \le c_n^{u,v}$$

(where $X \equiv (\theta_S, \lambda_Q)$) for all n and, in particular, for θ in the interval $\theta_i \le \theta \le \theta_i + \Delta\theta$ so the last integral can be put inside of the summations, and the proof of the term by term integration of (4.5) is complete:

$$\operatorname{cov}(\overline{u}_{ij}, \overline{v}_{kl}) = h \sum_{n=0}^{\infty} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta_T)\, \overline{P}_{nm}(\cos\theta_S) \cos m(\lambda_R - \lambda_Q) \quad \text{(A.5)}$$

where $\theta_i \le \theta_T \le \theta_i + \Delta\theta$ and $h = \Delta\lambda^2 \sin\theta_S \,\Delta\theta \sin\theta_T \,\Delta\theta / \Delta_{ij}\Delta_{kl}$

The general term in (A.5) satisfies the "M-Test":

$$|z_n| = \left|(2n+1)^{-1} c_n^{u,v} \sum_{m=0}^{n} \overline{P}_{nm}(\cos\theta_T)\, \overline{P}_{nm}(\cos\theta_S) \cos m(\lambda_R - \lambda_Q)\right| \le c_n^{u,v}$$

Since the series $\sum_{n=0}^{\infty} c_n^{u,v}$ converges, any series of positive terms $|z_n|$ satisfying $|z_n| \le c_n$ must converge also: the "M-Test" condition implies the absolute convergence of $\sum_{n=0}^{\infty} z_n$. Absolutely convergent series can have the order of their terms changed arbitrarily, without changing the value of their limit sums. This justifies the reordering of (4.9) that leads to expression (4.11-a).

Appendix B: Computer Programs (Descriptions and Listings)

This Appendix contains the description and listing of each of the major subroutines developed in the course of this research, together with their own auxiliary routines. The listings of the main subroutines contain explanatory comments that the author hopes will be of help to those who may use them. The description accompanying each main program defines arguments, gives the dimensions of the memory arrays required, mentions the relevant formulas from preceeding sections, and gives a brief explanation of the various segments of the program.

B.1 General Programming Considerations

Only one-dimensional arrays are used in the software described here. The language used for all of them is FORTRAN IV ; some subroutines from the International Mathematical and Statistical Libraries Inc. (IMSL) are called by the main subroutines. Implicit DO loops in READ or WRITE statements have been avoided as much as possible, because their execution may be rather slow, depending on the compiler; instead, subroutines FREAD, FWRITE and REWIND are used for all imput/output operations involving large files on tape or disk, in some of the subroutines. All operations involving real arithmetic have been coded in double precision (8 bytes, or 32 bits), which is equivalent to retaining the first 7 significant figures in all arithmetic operations.

The arrays containing the associated Legendre functions or their integrals, as the case may be, are arranged first by degree, and then by order : 00, 10, 11, 20, 21, 22, . . .(N_{max} , N_{max}).The $\overline{C}^{\alpha}_{nm}$ are arranged accordingly, always in two separated arrays: one for the $\overline{C}^{0}_{nm}$ and another for the $\overline{C}^{1}_{nm}$. In order to get the value of the element 'nm' from one of this arrays, the following formula is used:

$$k = \tfrac{1}{2} n (n + 1) + m + 1$$

where k is the position of this element in the one-dimensional array. When the elements are recovered sequentially from the beginning (00), the following type of DO loop is used:

```
KOUNT = 1
Do XX        N1 = 1, Nmax
DO XX        M1 = N1, Nmax

LEGEND (KOUNT) = ARRAY (KOUNT) **2

XX KOUNT = KOUNT + 1
```

where, in this particular example, the nm (n=N1-1, m=M1-1) element in array L LEGEND is equated to the square of the nm element in ARRAY. Avoidance of two-dimensional memory arrays results in considerable improvements in efficiency.

B.2 Subroutine SSYNTH

This subroutine computes the sum of a spherical harmonic series complete to degree and order NMAX at each one of the $2N^2$ points or blocks in an equal angular grid. The subroutine can calculate point values (IFLAG = 0) or area means (IFLAG = 1) . The number of rows or parallels N (Nyquist frequency) must be even. Subroutines FFTP from the IMSL Double Precision Library is used to calculate the sum of the series along rows by means of the Mix Radix Fast Fourier Transform algorithm.

The procedure used is that described in paragraphs (1.6) and (1.7). The symmetry of the grid with respect to the equator, and the corresponding even-odd symmetry of the values of the Legendre functions or their integrals, (i.e., the χ_i^{nm} of section 1) are exploited. The values of those functions, or of their integrals, are read from mass storage (disk or tape) into array ROW , in the order described in the previous paragraph. All the values for one latitude, or "row," are read at once, so the dimension of ROW is $\frac{1}{2}(N_{max}+1)(N_{max}+2)$. All the coefficients C^{α}_{nm} are also stored in core, the corresponding RCNM and RSNM arrays (for $\bar{C}_{nm}$ and $\bar{S}_{nm}$ respectively) have the same dimension (the $\bar{S}_{no's}$ are included, though they are all zero). The output consists of $2N^2$ values in array DATA . This array is organized in rows, from North Pole to South Pole. The rows, of 2N points or blocks each, have their values written consecutively. The following is the list of arrays, and their dimensions:

	NAME	DIMENSION	TYPE
	ROW	$RD=\frac{1}{2}(N_{max}+1)(N_{max}+2)$	REAL *8
	RCNM	RD	"
	RSNM	RD	"
	X	RD	INTEGER *2
	DATA	$2N^2$ (N = 180/BLOCK)	REAL *8
	CR1	N + 1	"
	CR2	N + 1	"
	SR1	N + 1	"
	SR2	N + 1	"
	AM	N + 1	"
	BM	N + 1	"
F	AUX1	4N	"
F	AUX2	4N	"
F	IWK	see IMSL Handbook	INTEGER *4
F	LL	" " "	LOGICAL *4
F	A	" " "	REAL *8
	IV	N + 1	"

("F" designates those arrays required by the IMSL subroutine FFTP). In addition, the size (in degrees) of the blocks is defined by BLOCK; $NPP = \frac{1}{2}(N_{max}+1)(N_{max}+2)$; IU is the number of the unit (disk, tape) from which the Legendre functions or their integrals are to be read.

Array X contains information on whether a given χ_i^{nm} is even or odd;

```
0001          SUBROUTINE SSYNTH(BLOCK,DATA,NMAX,RCNM,RSNM,AM,BM,X,ROW,CR1,SR1,
             2 CR2,SR2,AUX1,AUX2,IWK,IV,IU,NPP,A,IFLAG,LL)
       C
       C              THIS SUBROUTINE CREATES CENTER VALUES (IFLAG = 0) OR
       C              AREA MEANS (IFLAG = 1)  ON A
       C              (BLOCK (DEGREES) X BLOCK (DEGREES))  EQUAL AREA GRID, BY
       C              ADDING UP A SPHERICAL HARMONIC EXPANSION (WITH COEFFS.
       C              CNM, SNM  IN ARRAYS  RCNM  AND  RSNM  , RESPECTIVELY)
       C              TO DEGREE AND ORDER NMAX  (LIMITED ONLY BY STORAGE CAPACITY
       C              AND AVAILABLE COMPUTER TIME).  THIS SUBROUTINE USES
       C              A FILE IN UNIT  IU  CONTAINING THE LEGENDRE FUNCTIONS
       C              OR THEIR INTEGRALS, AS THE  CASE MAY BE, FOR ALL DISCRETE
       C              LATITUDE BANDS ("ROWS"), FROM  N.  POLE TO THE EQUATOR.
       C              IT USES THE FAST FOURIER SUBROUTINE (NON BINARY NUMBER
       C              OF DATA POINTS, OR "MIXED RADIX")  "FFTP"  FROM THE
       C              IMSL  LIBRARY. THE VALUES OF THE HARMONIC SUM ARE STORED
       C              IN THE  NC X NC/2  ARRAY  "DATA" ,  TO FORM AN EQUAL
       C              ANGULAR ARRAY OF SYNTHETIC DATA. TWO LATITUDE "ROWS" ARE
       C              CREATED TOGETHER, TAKING ADVANTAGE OF THE FACT THAT ALL
       C              LEGENDRE FUNCTIONS (AND THEIR INTEGRALS) ARE EITHER EVEN
       C              OR ODD  WITH RESPECT TO THE EQUATOR. THE SIZE OF THE
       C              BLOCK IS SUPPOSED TO BE AN EXACT DIVIDER OF  90  DEGREES,
       C              SO THERE IS NO "EQUATORIAL ROW" (NUMBER OF ROWS IS EVEN).
       C              AND EQUAL TO  180/BLOCK = NC/2
       C
       C              PROGRAMMED BY OSCAR L. COLOMBO, GEODETIC SCIENCE,
       C              OHIO STATE UNIVERSITY, COLUMBUS, FEBRUARY  1980  .
       C
0002          IMPLICIT REAL*8(A-H,O-Z)
0003          INTEGER*2 X
0004          LOGICAL*4 LL
0005          DIMENSION DATA(1),RCNM(1),RSNM(1),AM(1),BM(1),X(1),ROW(NPP),
             2 CR1(1),SR1(1),CR2(1),SR2(1),AUX1(1),AUX2(1),IWK(1),LL(1),IV(1),
             3 A(1)
0006          REWIND IU
0007          PI = 3.14159265358979300
0008          DRCONV = PI/180.D0
0009          BLK = BLOCK*DRCONV
0010          BL2 = BLK*0.5D0
0011          NP1 = NMAX+1
0012          NC = 360.D0/BLOCK+0.00000001D0
0013          NC2 = NC/2
0014          NC2P = NC2+1
0015          NROW2 = NC/4
0016          NCP = NC+1
0017          NUC = 1
0018          DO 2     N1 = 1,NP1
0019          DO 2     M1 = 1,N1
0020          X(NUC) = 1
0021          IF(((N1-M1)/2)*2.NE.(N1-M1)) X(NUC) = -1
0022        2 NUC = NUC+1
0023          IF(IFLAG.NE.1) GO TO 5
0024          DO 4     M1 =2,NP1
0025          M = M1-1
0026          AM(M1) = DSIN(M*BLK)/M
0027        4 BM(M1) =-(1.D0-DCOS(M*BLK))/M
0028          AM(1) = BLK
```

```
      BM(1) = 0.D0
      GO TO 8
    5 DO 6          M1 = 1,NP1
      M = M1-1
      BL2M = BL2*M
      AM(M1) = DCOS(BL2M)
    6 BM(M1) = -DSIN(BL2M)
    8 NDT = NC*NC2
      DO 9          I = 1,NP1
    9 IV(I) = I*(I-1)/2
      ISH = -NC
      ISHF = NDT
      PHI1 = 90.D0*DRCONV
      PHI2 = PHI1-BLK
C
C             MAIN LOOP, WHERE "DATA" IS CREATED, TWO EQUATORIALLY
C             SYMMETRICAL ROWS AT A TIME.
C             (OUTER LOOP)
C
      DO 40         NR = 1,NROW2
      FROWIN = 0.5D0
      IF(IFLAG.NE.1) GO TO 12
C
C             COMPUTING THE INVERSE OF THE AREA OF A BLOCK IN
C             ROW  NR  .
C
      FROWIN = 0.5D0/((DSIN(PHI1)-DSIN(PHI2))*BLK)
      PHI1 = PHI1-BLK
      PHI2 = PHI2-BLK
   12 ISH = ISH+NC
      ISHF = ISHF-NC
C
C             FORM THE COS AND SIN FOURIER COEFFICIENTS OF EACH "ROW"
C             UP TO FREQ.  M = NMAX , IN ARRAYS  CR1,SR1  AND
C             CR2, SR2, RESPECTIVELY.
C             (INNER LOOP)
C
C             READ THE (INTEGRALS OF) NORMALIZED LEGENDRE FUNCTIONS
C             CORRESPONDING TO THE NORTHERN "ROW", FROM UNIT "IU"
C             (UNFORMATTED RECORDS) .
C
      READ(IU) ROW
      DO 20        M1 = 1,NP1
      C1 = 0.D0
      C2 = 0.D0
      S1 = 0.D0
      S2 = 0.D0
C
C
C             THIS IS THE BEGINNING OF THE INNERMOST DO LOOP .
C
      DO 15       N1 = M1,NP1
      NUC = IV(N1)+M1
      R = ROW(NUC)
      S = R*RSNM(NUC)
      C = R*RCNM(NUC)
      IF(X(NUC).EQ.-1) GO TO 14
      C1 = C1+C
```

```
            C2 = C2+C
            S1 = S1+S
            S2 = S2+S
            GO TO 15
         14 C1 = C1+C
            C2 = C2-C
            S1 = S1+S
            S2 = S2-S
         15 CONTINUE
      C
      C            END OF INNERMOST LOOP .
      C
      C
            CR1(M1)  =  C1*AM(M1)-S1* BM(M1)
            CR2(M1)  =  C2*AM(M1)-S2*BM(M1)
            SR1(M1)  =  S1*AM(M1)+C1*BM(M1)
            SR2(M1)  =  S2*AM(M1)+C2*BM(M1)
         20 CONTINUE
            CR1(1)  =  2.D0*CR1(1)
            CR2(1)  =  CR2(1)*2.D0
      C
      C            "ALIASE" THE COEFFICIENTS .
      C
            DO 25         M1 = 1,NC2P
            MA = M1
            MB = -M1+2
            DO 25      K = 1,NP1
            MA = MA+NC
            MB = MB+NC
            IF(MB.GT.NP1)  GO TO 25
            IF(MA.GT.NP1)  GO TO 21
            CR1(M1)  =  CR1(M1)+CR1(MA)+CR1(MB)
            CR2(M1)  =  CR2(M1)+CR2(MA)+CR2(MB)
            SR1(M1)  =  SR1(M1)+SR1(MA)-SR1(MB)
            SR2(M1)  =  SR2(M1)+SR2(MA)-SR2(MB)
            GO TO 25
         21 CR1(M1)  =  CR1(M1)+CR1(MB)
            CR2(M1)  =  CR2(M1)+CR2(MB)
            SR1(M1)  =  SR1(M1)-SR1(MB)
            SR2(M1)  =  SR2(M1)-SR2(MB)
         25 CONTINUE
      C
      C            FORM THE COMPLEX FOURIER COEFFICIENTS .
      C
            K1 = -1
            K2 = 0
            DO 30      M1 = 1,NC
            K1 = K1+2
            K2 = K2+2
            IF(M1.GT.NC2P)  GO TO 26
            AUX1(K1)  =  CR1(M1)
            AUX2(K1)  =  CR2(M1)
            AUX1(K2)  =  -SR1(M1)
            AUX2(K2)  =  -SR2(M1)
            GO TO 30
         26 MX = NCP-M1+1
            AUX1(K1)  =  CR1(MX)
            AUX2(K1)  =  CR2(MX)
```

```
0112                  AUX1(K2) = SR1(MX)
0113                  AUX2(K2) = SR2(MX)
0114               30 CONTINUE
               C
               C             OBTAIN THE CONTENTS OF EACH ROW IN "DATA" AS FOURIER
               C             TRANSFORMS OF  AUX1  AND  AUX2 .
               C
0115                  CALL FFTP(AUX1,NC,IWK,A,LL)
0116                  CALL FFTP(AUX2,NC,IWK,A,LL)
0117                  K1 = -1
               C
               C             END OF OUTER AND INNER LOOPS .
               C
0118                  DO 40     I = 1,NC
0119                  K1 = K1+2
0120                  DATA(ISH+1) = AUX1(K1)*FROWIN
0121                  DATA(ISHF+1) = AUX2(K1)*FROWIN
0122               40 CONTINUE
0123                  REWIND IU
0124                  RETURN
0125                  END
```

arrays CR1 , SR1 , and CR2 , SR2 , respectively, contain the Fourier coefficients α_m^i (a_m^i) and $\beta_m^i(b_m^i)$ of rows i and N - i - 1 ; arrays AM and BM contain the values of A(m) and B(m) , as defined in expression (1.7); the auxiliary array IV contains the numbers $\frac{1}{2}n(n + 1)$ needed to locate individual elements within arrays ROW , RCNM , and RSNM , when they are not addressed sequentially.

The comments in the listing are probably enough to understand most of it on close inspection; one point however may be worth explaining further: the "aliasing" of the Fourier coefficients has been incorporated to take care of the case when $N_{max} > N - 1$. In such situation the α_m^i (a_m^i) and $\beta_m^i(b_m^i)$ become aliased, as Fourier coefficients must, and it is their aliased values that the FFT subroutine requires to compute the values of the spherical harmonic expansion along parallels. The formulas for the aliased coefficients are

$$\hat{\alpha}_m^i = \alpha_m^i + \sum_{k=1}^{KM} (\alpha_{m+KN}^i + \alpha_{KN-m}^i) \quad \text{(B 1-a)}$$

$$\beta_m^i = \beta_m^i + \sum_{k=1}^{KM} (\beta_{m+KN}^i - \beta_{KN-m}^i) \quad \text{(B 2-b)}$$

where KM is a large enough integer. A similar expression applies to a_m and to b_m .

The arrangement of the output in latitude corresponds, in the case of area means, to the intervals on which the $\overline{P}_{nm}$ are integrated; for point values, it is defined by the latitudes θ_i at which the $\overline{P}_{nm}(\cos\theta_i)$ have been precomputed. As regards longitude, the grid starts from the zero meridian used for defining the coefficients. In the case of point values it is usual to compute all values at the center of each block. To do this, the $\overline{P}_{nm}$ must be precomputed at the latitudes of the center points, while the longitudes are taken care by modifying the coefficients as follows

$$\overline{C}'_{nm} = \overline{C}_{nm} \cos m\frac{\Delta\lambda}{2} + \overline{S}_{nm} \sin m\frac{\Delta\lambda}{2} \quad \text{(B 2-a)}$$

$$\overline{S}'_{nm} = \overline{S}_{nm} \cos m\frac{\Delta\lambda}{2} - \overline{C}_{nm} \sin m\frac{\Delta\lambda}{2} \quad \text{(B 2-b)}$$

This is equivalent to rotating the grid eastwards from the zero meridian by $\frac{\Delta\lambda}{2}$.

B.3 Subroutine HARMIN

This subroutine implements either the algorithm of paragraph (1.5) for the harmonic analysis of area means, or that of paragraph (1.7) for the analysis of point values.

The subroutine calls IMSL's FFCSIN to calculate the a_m^i , b_m^i or the α_m^i , β_m^i by means of the Fast Fourier Transform (Mix Radix) algorithm. It also calls subroutine QUADFS , that returns in array A the de-smoothing

```
0001                SUBROUTINE HARMIN(BLOCK,DATA,NN,RCNM,RSNM,AM,BM,X,ROW,CR1,
                   2 SR1,CR2,SR2,AUX1,AUX2,IWK,IV,IU,NDD,A,IFLAG)
              C                                                                00007300
              C           THIS SUBROUTINE USES THE INTEGRALS OF LEGENDRE FUNCTIONS
              C           STORED IN UNIT  "IU" (CREATED USING PAUL'S RECURSIVES),
              C           OR, ALTERNATIVELY, POINT VALUES OF LEGENDRE FUNCTIONS, TO
              C           ANALYZE MEAN VALUES ON A COMPLETE, EQUAL ANGULAR GRID
              C           OR POINT DATA (IFLAG = 0) ON THE SAME TYPE OF GRID .
              C           THE INPUT, OR DATA, OCCUPIES ARRAY  "DATA"  , WHILE OUTPUT
              C           APPEARS IN ARRAYS "RCNM" AND "RSNM". MAXIMUM DEGREE AND
              C           ORDER CONSIDERED IS  NN  .
              C           THE NUMBER OF BLOCKS PER ROW (NC) MUST BE EVEN.         00008200
              C           IT USES THE FAST FOURIER SUBROUTINE
              C           "FFCSIN" FROM THE  IMSL(DOUBLE) LIBRARY.                00007900
              C           USER'S SUBROUTINE  "QUADFS"  PROVIDES THE "WEIGHTS" FOR THE
              C           QUADRATURES' FORMULA OF INTEREST.
              C
              C           FOR ADDITIONAL INFORMATION ON THE VARIOUS ARGUMENTS, SEE
              C           SUBROUTINE  "SSYNTHS"  .
              C
              C           PROGRAMED BY OSCAR L. COLOMBO, GEODETIC SCIENCE,
              C           OHIO STATE UNIVERSITY, COLUMBUS, FEBRUARY 1980 .
              C
0002                IMPLICIT REAL*8(A-H,O-Z)                                   00008600
0003                INTEGER*2 X
0004                DIMENSION DATA(1)
0005                DIMENSION RCNM(1),RSNM(1),AM(1),BM(1),ROW(NDD),CR1(1),SR1(1),
                   2 CR2(1),SR2(1),AUX1(1),AUX2(1),IWK(1),IV(1),X(1),A(1)
0006                REWIND IU
0007                PI = 3.14159265358979D0                                    00009100
0008                FT = 1.D0/(8.D0*PI)
0009                DRCONV = PI/180.D0                                         00009200
0010                BLK = BLOCK*DRCONV                                         00009300
0011                BL2 = BLK*0.5D0
0012                IF(IFLAG.NE.1) FT = FT*BLK**2
0013                NN1 = NN+1                                                 00009400
0014                NC = 360.D0/BLOCK                                          00009600
0015                NC2 = NC/2                                                 00009700
0016                NROWS = NC2
0017                NROW2 = NROWS/2
0018                NUC = 1
0019                DO 2      N1 = 1,NN1
0020                DO 2      M1 = 1,N1
0021                X(NUC) = 1
0022                IF(((N1-M1)/2)*2.NE.(N1-M1)) X(NUC) = -1
0023              2 NUC = NUC+1
0024                IF(IFLAG.NE.1) GO TO 4
0025                DO 5      M1 = 2,NN1                                       00010100
0026                M = M1-1                                                   00010200
0027                AM(M1) = DSIN(M*BLK)/M                                     00010300
0028              5 BM(M1) =-(1.D0-DCOS(M*BLK))/M                              00010400
0029                AM(1) = BLK                                                00010500
0030                BM(1) = 0.D0                                               00010600
0031                GO TO  7
0032              4 DO 6      M1 = 1,NN1
0033                M = M1-1
0034                BL2M = BL2*M
0035                AM(M1) = DCOS(BL2M)
```

```
    6 BM(M1) = -DSIN(BL2M)
    7 CALL QUADFS(A,NN1,BLOCK,NC)
      DO 8        I = 1,NDD                                          00010700
      RCNM(I) = 0.D0                                                 00010800
    8 RSNM(I) = 0.D0                                                 00010900
      DO 9        I = 1,NN1
      IF(IFLAG.EQ.1) A(I) = A(I)*FT
    9 IV(I) = I*(I-1)/2
C
C            BEGINNING OF MAIN (OUTER) LOOP, WHERE DATA "ROWS" (I.E.,
C            CONSTANT LONGITUDE BLOCKS) ARE ANALYZED TWO AT  THE TIME
C            TO  EXPLOIT EQUATORIAL SYMMETRY.
C
      ISH = -NC
      ISHF = NC*NROWS
      DO 20        NR = 1,NROW2
      FINN = FT*DSIN(BLK*NR-BL2)
      ISH = ISH+NC
      ISHF = ISHF-NC
      DO 10        I = 1,NC
      AUX1(I) = DATA(ISH+I)
   10 AUX2(I) = DATA(ISHF+I)
      CALL FFCSIN(AUX1,NC,CR1,SR1,IWK)
      CALL FFCSIN(AUX2,NC,CR2,SR2,IWK)
C            READING THE (INTEGRALS OF) LEGENDRE FUNCTIONS FROM
C            UNIT "IU" (UNFORMATTED RECORDS)
      READ(IU) ROW
C
C            BEGINNING OF INNER LOOP .
C
C
      DO 20        M1 = 1,NN1
      MM1 = M1
      M = M1-1
      IF(M.LE.NC2) GO TO 12
      MM1 = NC-M+1
      SR1(MM1) = -SR1(MM1)
      SR2(MM1) = -SR2(MM1)
   12 CONTINUE
      C1 = CR1(MM1)*AM(M1)+SR1(MM1)*BM(M1)
      C2 = CR2(MM1)*AM(M1)+SR2(MM1)*BM(M1)
      S1 = SR1(MM1)*AM(M1)-CR1(MM1)*BM(M1)
      S2 = SR2(MM1)*AM(M1)-CR2(MM1)*BM(M1)
      CP = C1+C2
      CM = C1-C2
      SP = S1+S2
      SM = S1-S2
      IF(IFLAG.EQ.1) GO TO 17
      CP = CP*FINN
      CM = CM*FINN
      SP = SP*FINN
      SM = SM*FINN
C            INNERMOST DO LOOP .
C
   17 DO 20       N1 = M1,NN1
      NUC = IV(N1)+M1
      R = ROW(NUC)
      IF(X(NUC).EQ.-1) GO TO 18
```

```
0081              RCNM(NUC)  =  RCNM(NUC)+R*CP
0082              RSNM(NUC)  =  RSNM(NUC)+R*SP
0083              GO TO 20
0084           18 RCNM(NUC)  =  RCNM(NUC)+R*CM
0085              RSNM(NUC)  =  RSNM(NUC)+R*SM
0086           20 CONTINUE
            C      END OF OUTER AND INNER LOOPS.
            C
0087              IF(IFLAG.EQ.0) GO TO 999
            C
            C            IN THE CASE OF AREA MEANS, MULTIPLY COEFFICIENTS BY
            C            DE-SMOOTHING FACTORS (IN ARRAY A ).
0088              NAC = 1
0089              DO 25       N1 = 1,NN1
0090              DO 25       M1 = 1,N1
0091              RCNM(NAC)  =  RCNM(NAC)*A(N1)
0092              RSNM(NAC)  =  RSNM(NAC)*A(N1)
0093           25 NAC = NAC+1
0094          999 CONTINUE
0095              REWIND IU
0096              RETURN
0097              END
```

```
0001          SUBROUTINE QUADFS(A,NN1,BLOCK,NC)
      C
      C              THIS SUBROUTINE COMPUTES THE VECTOR OF INTEGRATION WEIGHTS.
      C
0002          IMPLICIT REAL*8(A-H,O-Z)                                           00023400
0003          DIMENSION A(1),POLS(301)
0004          COMMON /AN/ ANM1(301),A2NM1(301)
0005          NN = NN1-1                                                         00023700
0006          PI = 3.14159265358979 3D0
0007          DRCONV = PI/180.D0
0008          NC2 = NC/2
0009          NC3 = NC2/3
0010          BLK = BLOCK*DRCONV
0011          BLICK = DSQRT(DSIN(BLK)*BLK/PI)/DRCONV
0012          F = 1.D0/(1.D0-DCOS(BLICK*DRCONV))
0013          DO 5      N = 2,NN1                                                00024000
0014          ANM1(N) = -((N-1)*1.D0)/N                                          00024100
0015        5 A2NM1(N) = (2.D0*N-1.D0)/N                                         00024200
0016          CALL LEGPOL(POLS,BLICK,NN1)                                        00024300
0017          A(1) = 0.D0                                                        00024400
0018          A(2) = 0.D0                                                        00024500
0019          DO 10     N = 2,NN                                                 00024600
0020          A(N+1) = 1.D0/(F/(2*N+1)*(POLS(N-1)-POLS(N+1)))
0021          IF(N.LE.NC3) A(N+1) = A(N+1)**2
0022          IF(N.GT.NC2) A(N+1) = 1.D0
0023       10 CONTINUE                                                           00025000
0024          WRITE(6,170) (I,A(I),I=1,NN1)
0025      170 FORMAT(1X,4(2X,I3,2X,G20.10))
0026          RETURN                                                             00025900
0027          END                                                                00026000
```

```
FORTRAN IV G1  RELEASE 2.0          LEGPOL            DATE = 80242        12/29/34            PAGE 0001

                SUBROUTINE LEGPOL(POLS,PSI,NMAX)                                00026100
                IMPLICIT REAL*8(A-H,O-Z)                                        00026200
                COMMON /AN/ ANM1(301),A2NM1(301)
                  DIMENSION POLS(1)                                             00026400
                PI = 3.141592653589793D0
                DRCONV = PI/180.D0
             10 T = DCOS(PSI*DRCONV)                                            00026600
                PNM1 = 1.D0                                                     00026700
                PN = T                                                          00026800
                POLS(1) = T                                                     00026900
                DO 20     N = 2,NMAX                                            00027000
                POLS(N) = ANM1(N)*PNM1+A2NM1(N)*PN*T                            00027100
                PNM1 = PN                                                       00027200
                PN = POLS(N)                                                    00027300
             20 CONTINUE                                                        00027400
                RETURN                                                          00027500
                END                                                             00027600
```

factors μ_n. QUADFS calls LEGPOL, a subroutine that computes the Legendre polynomials up to degree NN + 1 needed for the β_n^{-1} in μ_n.

The data is arranged as in SSYNTH, in array DATA, before the subroutine is called. Afterwards, the contents of DATA are destroyed, as the a_m^i, b_m^i or the α_m^i, β_m^i are formed in place of them, row by row, by FFCSIN. The resulting coefficients' estimates are put into arrays RCNM and RSNM, in the same order as for HARMIN. The other arrays, with the exception of A, are as in SSYNTH. The same is true of the scalars, with the exception of NN. NN is the highest degree and order to be estimated. NDD is the total number of Legendre functions, or their integrals, to be read from unit IU, per θ_i. This number is (NN+1) + (NN+2)/2. A is a REAL *8 array of dimension NN + 1. The dimension in QUADFS allows for a maximum NN = 300; for larger solutions, the dimensions there and in LEGPOL must be increased accordingly.

In the case of point data, the estimated coefficients are computed using a center point formula that assumes that the data are situated at the centers of the blocks; the resulting coefficients are referred, nonetheless, to a grid starting at the zero meridian (the "rotation" of the coefficients takes place between statements 0071 and 0073, when IFLAG = 0). When IFLAG=1, the area means formula (1.30) is computed; the $\chi_i^{nm} = \mu_n \int_{\sigma} \bar{P}_{nm}(\theta) \sin\theta \, d\theta$, and the μ_n are those produced by QUADFS, as already mentioned. The integrals of the Legendre functions are read from unit "IU", as in SSYNTH (same format), and the size of the blocks is specified by BLOCK (in degrees). The version of QUADFS listed here implements the "composite" estimator of paragraph (3.3). If another is desired, this can be achieved simply by replacing lines 0021 through 0024 in QUADFS.

B.4 Subroutine NORMAL

This subroutine creates the optimal estimators for the $\bar{C}_{nm}^{\alpha}$ based on the formation and inversion of the R(m) matrices described in paragraph (2.10) The algorithm exploits the fact that $(C_{zz} + D)$ is a block matrix of Toeplitz circulant sub-matrices. This subroutine is meant only for mean values.

The grid is as in SSYNTH and HARMIN. The symmetry with respect to the the equator is only partly exploited: matrix D may not be persymmetric, so the total matrix $(C_{zz} + D)$ may not be so either. C_{zz} however, is always persymmetric, and this is taken into account to save computing and storage. A general diagonal matrix D corresponds to a rather broad class of actual problems, such as the analysis of the $5^\circ \times 5^\circ$ real gravity anomalies described in paragraph (3.4).

This subroutine requires four input/output units: 8 (read only) contains the values of the integrals of the Legendre functions, row by row, arranged as in SSYNTH or HARMIN; 10 contains the right hand sides of the "reduced normals" $\underline{k}_{nm} = R(m)\ \underline{\chi}^{nm}$ (expression (2.58)); 15 contains the R(m) matrices, ordered by

```
                    C
                    C           READ THE (INTEGRALS OF) NORMALIZED LEGENDRE FUNCTIONS
                    C           FOR NORTHERN ROW .
                    C
0001                      SUBROUTINE NORMAL(NMAX,NN,DGRID,REGUL,IGEO,NRUN,ROWP,ROWQ,RHS,S,A,00000100
                         2 UL,NC2,W,DVAR,FC,PN,SS)
                    C                                                                        00000300
                    C           THIS SUBROUTINE IMPLEMENTS THE FAST COLLOCATION
                    C           ALGORITHM. IT CREATES THE OPTIMAL FILTER FOR ESTIMATING
                    C           SPHERICAL HARMONIC COEFFICIENTS OF AREA MEANS (EQUAL
                    C           ANGULAR) ON A SPHERE. A CHOICE CAN BE MADE OF EITHER
                    C           GEOCENTRIC OR GEODETIC COORDINATES.
                    C
                    C           THE GRID MUST BE BOTH COMPLETE AND SYMMETRICAL RESPECT
                    C           TO THE EQUATOR
                    C           BECAUSE OF THE GRID'S SYMMETRY, THE "SIGNAL" PART OF EACH
                    C           R(M)  MATRIX IS PERSYMMETRIC, AND FULL ADVANTAGE IS TAKEN
                    C           OF THIS FACT WHEN SETTING UP THE  R(M)'S . THE "NOISE" PART,
                    C           ON THE OTHER HAND, CAN BE ANY DIAGONAL MATRIX (ASYMMETRIC).
                    C
                    C                                                                        00000800
                    C           INTEGRALS  INMP  ARE READ IN FROM UNIT  8, STORED, AFTER    00000900
                    C           MULTIPLYING THEM BY CERTAIN FACTORS, IN UNIT  10 .
                    C           THE  R(M)'S  MATRICES ARE STORED IN UNIT  15   ,             00001100
                    C           OPTIMAL "QUADRATURES' WEIGHTS" IN UNIT 30.                   00001200
                    C                                                                        00001300
                    C           THIS PROGRAM USES SUBROUTINES  "LUDECP"  AND  "LUELMP" FROM
                    C           THE  IMSL.DOUBLE  LIBRARY. IN TURN, THESE USE OTHER
                    C           SUBROUTINES FROM THE  IMSL.SINGLE  LIBRARY. THIS PROGRAM
                    C           ALSO CALLS SUBROUTINE  "FUR"  TO COMPUTE THE FOURIER
                    C           COEFFICIENTS OF THE BLOCK COVARIANCES, AND SUBS. "FREAD",
                    C           "FWRITE", AND "REWIND" FOR FAST INPUT-OUTPUT. THE USER
                    C           CAN REPLACE THESE WITH HIS OWN, OR WITH THE STANDARD "READ",
                    C           "WRITE" AND "REWIND" COMMANDS. NPP8,NR8, ETC ARE THE
                    C           LENGTHS*8 (I.E., IN BYTES) OF THE ARRAYS TO BE READ
                    C           OR WRITEN.
                    C           IF THE  R(M)  MATRIX OF A PARTICULAR ORDER CANNOT BE
                    C           INVERTED, NULL QUADRATURES'"WEIGHTS" FOR THAT ORDER ARE
                    C           STORED IN UNIT  30 .
                    C
                    C PROGRAMMED BY OSCAR L. COLOMBO, GEOD. SC., OHIO STATE U., SEPT. 1979. 00001400
                    C                                                                        00001500
0002                      IMPLICIT REAL*8(A-H,O-Z)                                           00001600
0003                      DIMENSION ROWP(1),ROWQ(1),RHS(1),S(1),A(1),UL(1)                   00001700
0004                      DIMENSION ANMPQ(400),SS(1),B(200),X(200),FF(400),
                         2 IDD(400),X0(200),BT(200),FC(1),PN(1),W(1),PHI(200),DVAR(1),FINMP
                         3 (400)
0005                      COMMON/MM/ MT(310)
                    C                                                                        00002300
                    C           BASIC CONSTANTS                                              00002400
                    C                                                                        00002500
0006                      PI = 3.14159265358979 3D0                                          00002600
0007                      DRCONV = PI/180.D0                                                 00002700
0008                      F = 1.D0/298.257D0                                                 00002800
                    C                                                                        00002900
                    C           PROBLEM DEFINITION                                           00003000
                    C                                                                        00003100
```

```
              C              DGRID:    BLOCK SIZE IN DEGREES. MINIMUM:  1  DEGREE .
              C              ICEO      = 2  IF  GEODETIC  LATITUDES ARE TO BE CONVERTE00003300
              C                        TO GEOCENTRIC.                              00003400
              C              NN        MAX. DEG. AND ORDER TO BE ANALYZED (LESS THAN N00003500
              C              NC        NO. OF "COLUMNS", ALWAYS EVEN,              00003600
              C              NR        NO. OF ROWS. THERE IS NO EQUATORIAL ROW.    00003700
              C              NMAX      MAX. DEGREE IN TRUNCATED EXPANSION OF BLOCK CO-
              C                        VARIANCE FUNCTIONS. NOT GREATER THAN  NC .
              C              NRUN      = 0   IF  R(M)'S ARE CREATED AND STORED IN   00003900
              C                        UNIT 15.                                    00004000
              C                        = 999 IF R(M)'S ARE READ FROM UNIT  15  INSTEAD00004100
              C                        OF BEING CREATED.                           00004200
              C            REGUL       REGULARIZATION PARAMETER. ADDED TO ALL DIAGONAL00004300
              C                        TERMS OF THE  R(M)  MATRICES.               00004400
              C                                                                    00004500
0009                DGR = DGRID*DRCONV                                             00004700
0010                NC = 360.D0/DGRID+0.000001D0                                   00004800
0011                NC4 = NC*0.25+0.000001D0                                       00004900
0012                NC2 = NC4*2                                                    00005000
0013                NR = NC2                                                       00005100
0014                NN1 = NN+1                                                     00005200
0015                NP = 1                                                         00005300
0016                NP1 = NMAX+1                                                   00005400
0017                NPP = (NP1*(NP1+1))/2                                          00005500
0018                NLL = (NN1*(NN1+1))/2                                          00005600
0019                NPP8 = 8*NPP                                                   00005700
0020                NLL8 = 8*NLL                                                   00005800
0021                NC21 = NC2+1                                                   00005900
0022                NC28 = 8*NC2                                                   00006000
0023                FF(1) = DGR**2                                                 00006100
0024                DVAR0 = 0.D0                                                   00006200
0025                DO 5   I = 1,NC                                                00006300
0026                FC(I) = 0.D0                                                   00006400
0027                PN(I) = 0.D0                                                   00006500
0028              5 X0(I) = 0.D0                                                   00006600
              C                                                                    00006700
              C            FORM ARRAY OF GEOCENTRIC OR GEODETIC GRID'S GEOCENTRIC LATS00006800
              C                                                                    00006900
0029                EN = F/(2.D0-F)                                                00007000
0030                EM = 2.D0*EN/(1.D0+EN**2)                                      00007100
0031                NC2P = NC2+1                                                   00007200
0032                ANG = 90.D0                                                    00007300
0033                DO 6       I = 1,NC2P                                          00007400
0034                PHI(I) = ANG*DRCONV                                            00007500
0035                IF(ICEO.EQ.2) PHI(I) = PHI(I)-EM*DSIN(2.D0*PHI(I))            00007600
0036              6 ANG = ANG-DGRID                                                00007700
0037                DO 15     II = 2,NP1                                           00007800
0038                I = II-1                                                       00007900
0039                FINMP(II) = DSQRT(DVAR(I)/(2.D0*I+1))                          00008000
0040             15 CONTINUE                                                       00008100
0041                FINMP(1) = DSQRT(DVAR0)                                        00008200
              C                                                                    00008300
              C            READ TAPE OF INTEGRALS OF LEGENDRE FUNCTIONS FROM UNIT  8 ,00008400
              C            MULTIPLY THEM BY THE "ROW FACTORS" (INVERSE OF BLOCK AREA, 00008500
              C            AND BY SQUARE ROOT OF DEGREE VARIANCE (PER COEFF.), STORING00008600
              C            THESE IN UNIT  10  .
              C                                                                    00009000
0042                IRS = -NN1                                                     00009100
```

```
      DO 25       NT = 1,NC2
      IRS = IRS+NN1
      FROWIN = 1.D0/((DSIN(PHI(NT))-DSIN(PHI(NT+1)))*DCR)
      CALL FREAD(ROWP(1),8,NPP8,8999,8999)
      NIS = 1
      DO 20      I = 1,NP1
      FX = FINMP(I)*FROWIN
      DO 20    IL = 1,I
      ROWP(NIS) = ROWP(NIS)*FX
   20 NIS = NIS+1
      CALL FWRITE(ROWP(1),10,NPP8,8999,8999)
   25 CONTINUE
      ND = NR*NN1
      NR8 = NR*8
      WRITE(6,555) DGRID, NMAX, NN, NR, NRUN
  555 FORMAT(//' BLOCK SIZE ',F7.2,'  MAX. DEGREE IN COVARIANCE ',I5,'
     2 MAX. DEGREE TO BE ANALYZED ',I5,'  NUMBER OF ROWS ',I5,' NRUN ',
     3 I5//)
      WRITE(6,556) NC
  556 FORMAT('    NUMBER OF COLUMNS ',I5//)
      IF(ICEO.EQ.1) WRITE(6,557)
      IF(ICEO.EQ.2) WRITE(6,558)
  557 FORMAT('    DATA ON A SPHERE'//)
  558 FORMAT('    DATA ON AN ELLIPSOID'//)
      DO 99         M1 = 1,NP1
   99 IDD(M1) = ((M1-1)*M1)/2
      DO 100        M1 = 2,NP1
  100 FF(M1) = 2.D0*(1.D0-DCOS((M1-1)*DCR))/(M1-1)**2
C
C            READ THE R.H.S.'S OF THE REDUCED "NORMALS" FROM UNIT  10  .
C            INTO VECTOR  RHS   .
C
      CALL REWIND(10)
      CALL REWIND(30)
      ISH = -NLL
      DO 210          I = 1,NC4
      CALL FREAD(ROWP(1),10,NLL8,8999,8999)
      ISH = ISH+NLL
      DO 210          J = 1,NLL
  210 RHS(ISH+J) = ROWP(J)
      IF(NRUN.EQ.999) GO TO 1500
C
C      MAIN LOOP FOR THE CREATION OF THE  R(M)'S  MATRICES.
C
      CALL REWIND(10)
 1000 CONTINUE
      DO 101          I = 1,ND
  101 S(I) = 0.D0
C
C            POSITION  THE  INMP'S TAPE TO CURRENT ROW  NP .
C
      DO 102        NX = 1,NP
  102 CALL FREAD(ROWP(1),10,NPP8,8999,8999)
      IS = (NP-2)*NN1
      NCV = NC2-NP+1
      DO 120        NQ = NP,NCV
      IF(NQ.NE.NP) CALL FREAD(ROWQ(1),10,NPP8,8999,8999)  .
      IF(NQ.NE.NP) GO TO 105
```

```
0089                DO 103        I = 1,NPP                                            00016400
0090            103 ROWQ(I) = ROWP(I)                                                  00016500
              C                                                                        00017700
              C             OBTAIN THE FOURIER COEFFICIENTS OF THE BLOCK COVARIANCES
              C             CORRESPONDING TO ROWS  P  AND  Q   IN  CDD.                00017900
              C             AND ALIASE THEM TO FORM THE ELEMNTS OF THE  R(M)  MATRICES.
              C                                                                        00018000
0091            105 CONTINUE
0092                DO 1     I = 1,310
0093                MT(I) = 0
0094              1 CONTINUE
0095                DO 111       M1 = 1,NN1                                            00018100
0096                M = M1-1                                                           00018200
0097                H = NC2                                                            00018300
0098                IF(M.EQ.0.OR.M.EQ.NC2) H = NC                                      00018400
0099                CALL FUR(AN,M1,ROWP,ROWQ,IDD,FF,NP1)
0100                ANMPQ(M1) = AN
0101                N1 = NC-M+1                                                        00018600
0102                N2 = NC+M+1                                                        00018700
0103            112 CONTINUE                                                           00018800
0104                IF(N1.GT.NP1) GO TO 111
0105                CALL FUR(AN1,N1,ROWP,ROWQ,IDD,FF,NP1)
0106                ANMPQ(M1) = ANMPQ(M1)+AN1
0107                IF(N2.GT.NP1) GO TO 108
0108                CALL FUR(AN2,N2,ROWP,ROWQ,IDD,FF,NP1)
0109                ANMPQ(M1) = ANMPQ(M1)+AN2
0110            108 N1 = N1+NC
0111                N2 = N2+NC                                                         00019200
0112                GO TO 112
0113            111 ANMPQ(M1) = ANMPQ(M1)*H
0114                IS = IS+NN1                                                        00019600
0115                DO 114          NL = 1,NN1                                         00019700
0116                S(IS+NL) = ANMPQ(NL)                                               00019800
0117            114 CONTINUE                                                           00019900
0118            120 CONTINUE                                                           00020000
              C                                                                        00020100
              C             STORE THE LOWER DIAGONAL PART OF  R(M)  COLUMN BY COLUMN   00020200
              C             ON DISK FILE (UNIT  15  ).                                 00020300
              C                                                                        00020400
0119                DO 116       M2 = 1,NN1                                            00020500
0120                IS = -NN1                                                          00020600
0121                DO 115         I = 1,NC2                                           00020700
0122                IS = IS+NN1                                                        00020800
0123            115 SS(I) = S(IS+M2)                                                   00020900
0124                CALL FWRITE(SS(1),15,NC28,8999,8999)                               00021000
0125            116 CONTINUE                                                           00021100
0126                CALL REWIND(10)                                                    00021200
0127                NP = NP+1                                                          00021300
0128                IF(NP.LE.NC4) GO TO 1000                                           00021400
0129           1500 CONTINUE                                                           00021500
              C                                                                        00021600
              C             ADD "REGUL" (THE REGULARIZATION CONSTANT) TO DATA "ROW     00021700
              C             VARIANCES" .                                               00021800
              C                                                                        00021900
0130                DO 217        I = 1,NC2                                            00022000
0131                  W(I) = W(I)+REGUL                                                00022100
0132            217 CONTINUE                                                           00022200
              C                                                                        00022300
```

```
                C            BEGIN MAIN LOOP WHERE ERRORS ARE ESTIMATED.                 00022400
                C                                                                        00022500
0133                  DO 300        M1 = 1,NN1                                           00022600
0134                  CALL REWIND(15)                                                    00022700
0135                  FFF = FF(M1)*NC                                                    00022800
0136                  M = M1-1                                                           00022900
0137                  IF(M.EQ.0) GO TO 214                                               00023000
0138                  DO 213        I = 1,M                                              00023100
0139              213 CALL FREAD(SS(1),15,NC28,8999,8999)                                00023200
                C                                                                        00023300
                C      READ IN MATRIX  R(M)  FROM UNIT  15 , COLUMN BY COLUMN,           00023400
                C            AND PUT IT IN SYMMETRIC STORAGE FORM IN VECTOR  A .         00023500
                C                                                                        00023600
0140              214 DO 220        J = 1,NC4                                            00023700
0141                  NC2J1 = NC2-J+1                                                    00023800
0142                  IF(J.EQ.1) GO TO 222                                               00023900
0143                  DO 215        I = 1,NN                                             00024000
0144              215 CALL FREAD(SS(1),15,NC28,8999,8999)                                00024100
0145              222 CALL FREAD(SS(1),15,NC28,8999,8999)                                00024200
0146                  DO 220        I = J,NC2                                            00024300
0147                  IF(I.GT.NC2J1) GO TO 220
0148                  K = IDD(I)+J                                                       00024400
0149                  NC2I1 = NC21-I                                                     00024500
0150                  KP = IDD(NC2J1)+NC2I1                                              00024600
0151                  A(K) = SS(I)                                                       00024700
0152                  A(KP) = SS(I)
0153                  IF(I.NE.J) GO TO 220                                               00024900
0154                  A(K) = A(K)+W(J)                                                   00025000
0155                  A(KP) = A(KP)+W(NC2J1)                                             00025100
0156              220 CONTINUE                                                           00025200
                C
                C            FORM THE CHOLESKY FACTOR  UL .
                C
0157                  CALL LUDECP(A,UL,NC2,D1,D2,IER)
0158                  IF(IER.EQ.129) WRITE(6,221) M
0159              221 FORMAT(//' ***  MATRIX  R(M)+W  CORRESPONDING TO ORDER ',I5,' IS
                     2SINGULAR ***'//)
0160                  IF(IER.EQ.129) GO TO 298
                C
                C            SOLVE THE REDUCED EQUATIONS  R(M) X(M,N) = B(M,N)  FOR ALL
                C            N  AND ALL  M  , AND SAVE THE OPTIMAL "QUADRATURE WEIGHTS",
                C            OR SOLUTIONS, IN UNIT  30  .
                C
                C            FIRST FORM  B(M,N)
                C
0161                  DO 240        N1 = M1,NN1
0162                  SNM = 1.D0
0163                  IF(((N1-M1)/2)*2.NE.(N1-M1)) SNM = -1.D0
0164                  ISH = -NLL
0165                  KT = IDD(N1)+M1
0166                  DO 230        I = 1,NC4
0167                  ISH = ISH+NLL
0168                  B(I) = RHS(ISH+KT)*FINMP(N1)
0169                  NC21I = NC21-I
0170                  B(NC21I) = B(I)
0171              230 IF(SNM.EQ.-1) B(NC21I) = -B(I)
                C
                C            SOLVE THE (N,M)  NORMAL EQUATION.
```

```
C            SAVE RESULT IN UNIT  30 .
C
       CALL LUELMP(UL,B,NC2,X)
C
C            COMPUTE THE A POSTERIORI STANDARD ERRORS OF THE COEFFICIENTS
C            PER DEGREE.
C
       PNOISE = 0.D0
       FCSD = 0.D0
       DO 232        I = 1,NR
       FCSD = FCSD+B(I)*X(I)
   232 PNOISE = PNOISE+X(I)**2*(W(I)-REGUL)
       FCSD = FCSD*FFF
       PNOISE = PNOISE*FFF
       FC(N1) = FC(N1)+FCSD
       PN(N1) = PN(N1)+PNOISE
       CALL FWRITE(X(1),30,NR8,8999,8999)
   240 CONTINUE
C
C            CHECK FOR AGREEMENT BETWEEN ORIGINAL AND RECONSTRUCTED R.H.S.'S
C            BY TESTING THE (N,N) CASES.
C
       SO = 0.D0
       SAV = 0.D0
       DO 245        I = 1,NC2
       BT(I) = 0.D0
       DO 244        J = 1,NC2
       KT = IDD(I)+J
       IF(J.GT.I) KT = IDD(J)+I
   244 BT(I) = BT(I)+X(J)*A(KT)
       SAV = SAV+B(I)**2
   245 SO = SO+(B(I)-BT(I))**2
       SO = DSQRT(SO/NC2)
       IF(SAV.NE.0) SO = DSQRT(SO/SAV)
       WRITE(6,247) M,SO
   247 FORMAT(//'  ACCURACY OF INVERSION OF  R(',I3,') IS',G20.10)
       GO TO 300
   298 DO 299        I = M1,NN1
       CALL FWRITE(X0(1),30,NR8)
   299 CONTINUE
   300 CONTINUE
       CALL REWIND (30)
C
C            FINISH COMPUTING THE A POSTERIORI COEFFICIENTS'
C            ERRORS PER DEGREE.
C
       DO 310        I = 2,NN1
       N2IM1 = 2*I-1
       ECOEFN = DVAR(I-1)-FC(I)
       ALIAS = ECOEFN-PN(I)
       FC(I) = ECOEFN
       SS(I) = ALIAS
   310 CONTINUE
   999 CONTINUE
       RETURN
       END
```

```
0001              SUBROUTINE FUR(AN,M1,RP,RQ,IDD,FF,NP1)
            C
            C            THIS SUBROUTINE COMPUTES THE FOURIER COEFFICIENTS
            C            OF THE BLOCK COVARIANCES.
            C
0002              IMPLICIT REAL*8 (A-H,O-Z)
0003              DIMENSION RP(1),RQ(1),IDD(1),FF(1)
0004              DIMENSION ANP(310)
0005              COMMON/MM/ M(310)
0006              IF(M(M1).EQ.1) GO TO 15
0007              AN = 0.D0
0008              DO 10      N1 = M1,NP1
0009              IX = IDD(N1)+M1
0010           10 AN = AN+RP(IX)*RQ(IX)
0011              AN = AN*FF(M1)
0012              ANP(M1) = AN
0013              M(M1) = 1
0014              RETURN
0015           15 AN = ANP(M1)
0016              RETURN
0017              END
```

```
0001                SUBROUTINE ANALYS(X,BLOCK,DATA,NN,RCNM,RSNM,AM,BM,NROWS,        00056100
                   2 ROW,CR,SR,IWK,CAA,CBB,SAA,SBB)
            C                                                                       00056300
            C             THIS SUBROUTINE USES THE OUTPUT OF SUBR. "NORMAL",        00056400
            C             READ FROM UNIT  10  , TO COMPUTE THE SPHERICAL HARMONIC   00056500
            C             COEFFICIENTS OF THE DATA STORED IN ARRAY "DATA" AS        00056600
            C             (BLOCK) DEGREES X (BLOCK) DEGREES  AREA MEANS, USING      00056700
            C             AN ALGORITHM FOR FAST COLLOCATION. THIS SUBR. CALLS SUBR. 00056800
            C             "FFCSIN" FROM THE  IMSL(DOUBLE) LIBRARY.                  00056900
            C             THE ORIGINAL CONTENTS OF ARRAY  "DATA"  ARE DESTROYED.    00057000
            C             THE GRID IS SUPPOSED TO BE EQUAL ANGULAR.                 00057100
            C             THE NUMBER OF BLOCKS PER ROW (NC) MUST BE EVEN.           00057200
            C                                                                       00057300
            C      PROGRAMMED BY OSCAR L. COLOMBO, DEPT. OF GEODET. SC., OHIO S.U., 00057400
            C                                                                       00057500
0002                IMPLICIT REAL*8(A-H,O-Z)                                        00057600
0003                DIMENSION RCNM(1),RSNM(1),DATA(1),AM(1),BM(1),X(1),FR(1),CR(1)  00057700
                   2, SR(1),IWK(1),CAA(1),CBB(1),SAA(1),SBB(1)
0004                DIMENSION ROW(1)                                                00057900
0005                DIMENSION D(4000)                                               00058000
0006                PI = 3.141592653589793D0                                        00058100
0007                DRCONV = PI/180.D0                                              00058200
0008                BLK = BLOCK*DRCONV                                              00058300
0009                NN1 = NN+1                                                      00058400
0010                NDD = NN1*(NN1+1)/2                                             00058500
0011                NC = 360.D0/BLOCK                                               00058600
0012                NC2 = NC/2                                                      00058700
0013                NCP = NC+2                                                      00058800
0014                NC2P = NC2+1                                                    00058900
0015                NRB = NROWS*8                                                   00059000
0016                DO 5      M1 = 2,NN1                                            00059100
0017                M = M1-1                                                        00059200
0018                AM(M1) = DSIN(M*BLK)/M                                          00059300
0019              5 BM(M1) = (1.D0-DCOS(M*BLK))/M                                   00059400
0020                AM(1) = BLK                                                     00059500
0021                BM(1) = 0.D0                                                    00059600
0022                DO 8        I = 1,NDD                                           00059700
0023                RCNM(I) = 0.D0                                                  00059800
0024              8 RSNM(I) = 0.D0                                                  00059900
0025                ISH = -NC                                                       00060000
0026                ISHP = -NCP                                                     00060100
0027                DO 15        NR = 1,NROWS                                       00060200
0028                ISH = ISH+NC                                                    00060300
0029                ISHP = ISHP+NCP                                                 00060400
0030                DO 10        I = 1,NC                                           00060500
0031             10 ROW(I) = DATA(ISH+I)                                            00060600
0032                CALL FFCSIN(ROW,NC,CR,SR,IWK)                                   00060700
0033                ISH2P = ISHP+NC2P                                               00060800
0034                DO 15        I = 1,NC2P                                         00060900
0035                D(ISH2P+I) = SR(I)*0.5D0                                        00061000
0036             15 D(ISHP+I) = CR(I)*0.5D0                                         00061100
0037                DO 20      M1 = 1,NN1                                           00061200
0038                MM1 = M1                                                        00061300
0039                M = M1-1                                                        00061400
0040                IF(M.GT.NC2) MM1 = NC-M+1                                       00061500
0041                NOFFC = -NCP+MM1                                                00062000
0042                NOFFS = NOFFC+NC2P                                              00062100
0043                DO 166       I = 1,NROWS
```

```
      NOFFC = NOFFC+NCP                                 00062300
      NOFFS = NOFFS+NCP                                 00062400
      CAA(I) = D(NOFFC)*AM(M1)
      CBB(I) = D(NOFFC)*BM(M1)
      SAA(I) = D(NOFFS)*AM(M1)
      SBB(I) = D(NOFFS)*BM(M1)
  166 CONTINUE
      DO 20      N1 = M1,NN1                            00061600
      CALL FREAD(X(1),30,NRB,8999,8999)                 00061700
      NUC = (N1-1)*N1/2+M1                              00061900
      DO 20      I = 1,NROWS
      CA = CAA(I)
      CB = CBB(I)
      SA = SAA(I)
      SB = SBB(I)
      NR = I
      IF(M.LE.NC2) GO TO 16
      SA = -SA
      SB = -SB
   16 RCNM(NUC) = RCNM(NUC)+X(NR)*(CA+SB)
      RSNM(NUC) = RSNM(NUC)+X(NR)*(SA-CB)
   20 CONTINUE                                          00062800
  999 CONTINUE                                          00062900
      RETURN                                            00063000
      END                                               00063100
```

```
FORTRAN IV G1   RELEASE 2.0          FWRITE          DATE = 80242          12/29/34          PAGE 0001

                      SUBROUTINE FWRITE(ROW,IU,N,*,*)
                      LOGICAL*1 ROW
                      DIMENSION ROW(N)
                      WRITE(IU) ROW
                      RETURN
                      END


FORTRAN IV G1   RELEASE 2.0          FREAD           DATE = 80242          12/29/34          PAGE 0001

                      SUBROUTINE FREAD(ROW,IU,N,*,*)
                      LOGICAL*1 ROW
                      DIMENSION ROW(N)
                      READ(IU) ROW
                      RETURN
                      END


FORTRAN IV G1   RELEASE 2.0          REWIND          DATE = 80242          12/29/34          PAGE 0001

                      SUBROUTINE REWIND(IU)
                      REWIND IU
                      RETURN
                      END
```

increasing m , stored in vector array form column by column; 30 stores the χ_i^{nm} of the optimal quadratures-type estimator. The χ_i^{nm} are stored from N. Pole to S. Pole, and according to nm , as the Legendre integrals and the coefficients. In some circumstances the grid may be geocentric rather than geodetic and a change of coordinates might be desirable: this can be achieved by setting the parameter IGEO to 2 . The flattening assumed for this transformation is F = 1/298.257.

After the R(m) matrices have been created, they are inverted by IMSL subroutine LUDECP , that performs a Choleskii factorization. IMSL subroutine LUELMP solves the equations resulting in the χ_i^{nm} ; if during the inversion LUDECP detects an ill-conditioned (or a singular) matrix, the solution part is avoided, and a set of null χ_i^{nm} is stored for that particular m . As an additional check for the stability of the solution, the relative residuals.

$$r = \frac{\sum_{i=0}^{N-1} v_i^2}{\sum_{i=0}^{N-1} (k_i^{nm})^2} \qquad \text{where } \underline{v} = [v_0 , v_1 , \ldots v_{N-1}]^T \text{ is} \quad \underline{v} = \underline{k}^{nm} - R(m)\, \underline{\chi}^{nm} \text{ (computed)} \tag{B 3}$$

are computed and printed. In all the cases studied here these residuals indicated an agreement of at least 9 significant figures. To improve the stability of the solution, a regularizing constant REGUL is added to the diagonal elements of the R(m) (Paragraph (3.3)).

Arrays PN , SS , and FC contain the propagated noise, sampling, and total error measure (variance) per degree. W contains the averaged row variances (expression (2.43)) arranged from North to South.

The scalar arguments, NMAX , NN , DGRID , IGEO , REGUL , NRUN , and NC 2 , are described in the comments inserted between statements 0006 and 0010. The arrays are as follows:

NAME	DIMENSION	TYPE
ROWP	$\frac{1}{2}$(NMAX + 1)(NMAX + 2)	REAL * 8
ROWQ	"	"
RHS	$\frac{1}{4}$NN (NN + 1) N (N = Nyquist freq.)	"
S	(NN + 1) N	"
A	$\frac{1}{2}$N(N - 1)+N	"
UL	"	"
W	N	"
DVAR	NMAX + 1	"
FC	NN + 1	"
PN	"	"
SS	"	"

Arrays ANMPQ , FF, XO , B , X , BT , FINMP (all REAL * 8), and IDD (REAL * 4), all dimensioned 200 or 400 in the subroutine itself, are large enough for problems where $N < 200$. For finer grids, the size of these arrays should be increased in the same proportion as that of N .

The R(m) matrices are formed according to expression (2.63) . Subroutine FUR computes the "aliased" Fourier coefficients of the covariance functions that are, in fact, the elements of the R(m) , scaled by N or 2N , depending on m . Common MM and array MT are part of a logic set up to ensure that the Fourier coefficients are not computed more than once each.

Subroutine ANALYS uses the χ_i^{nm} stored in unit 30 to analyze the data in array DATA . The α_m^i, β_m^i are formed in place, as in HARMIN , so the original values in DATA are destroyed. Arrays CR , SR , CAA , CBB , SAA , and SBB have all the same description as CR1 , SR1 , etc., in HARMIN . IMSL subroutine FFCSIN (double precision) is used to obtain the α_m^i , β_m^i. The reason why ANALYS is used instead of HARMIN , is the arrangement of the χ_i^{nm} "columnwise", or by increasing latitudes, rather than "rowise" (i.e., all the χ_i^{nm} for the same i stored together) as HARMIN would require.

The listings of FUR , ANALYS , and those of the fast input/output subroutines FREAD , FWRITE , REWIND , are given after that of NORMAL . The input/output subroutines have dummy arguments, because originally NORMAL was written to work with certain subroutines available at O.S.U. that may not be in the software libraries of other institutions.

B.5 Subroutine NORMAX

A modified version of NORMAL , this subroutine was created to compute the variance of the estimation errors in ordinary quadratures formulas according to the theory in section 2 . Essentially, it computes

$$\sigma^2_{\epsilon n} = \sigma^2_n - 2\sum_{m=0}^{n}\sum_{\alpha=0}^{1} \underline{c}^{T}_{nm\alpha, z}\,\underline{f}^{\alpha}_{nm} + \sum_{m=0}^{n}\sum_{\alpha=0}^{1} (\underline{f}^{\alpha}_{nm})^{T}\,(C_{zz} + D)\,\underline{f}^{\alpha}_{nm}$$

by forming and using the R(m) . Since no inversion or solution of the normal equations is required, the corresponding segments have been removed from NORMAL , and a new final segment added for the computation of the various accuracies.

The theory behind the calculation of the $\sigma^2_{\epsilon n}$ using the R(m) matrices is as follows:

In the case of ordinary formulas of the type (see expression (1.7))

```
0001             SUBROUTINE NORMAX(NMAX,NN,DGRID,ICEO,NRUN,ROWP,ROWQ,RHS,S,A,
                2 UL,NC2,W,DVAR,QUADS,XHS,FC,PN,SS)
           C                                                                   00015100
           C             SUBROUTINE FOR COMPARING NUMERICAL QUADRATURES FORMULAS USING
           C             COLLOCATION THEORY AND "FAST COLLOCATION"
           C             ALGORITHM FOR DATA AVERAGED OVER EQUAL ANGULAR BLOCKS     00015300
           C             ON A SURFACE OF REVOLUTION.                               00015400
           C                                                                       00015500
           C             INTEGRALS  INMP  ARE READ IN FROM UNIT  8, STORED, AFTER  00015600
           C             MULTIPLYING THEM BY SUITABLE FACTORS, IN UNITS 10 AND 12.
           C             THE  R(M)'S  MATRICES ARE STORED IN UNIT  15   ,          00015800
           C             OPTIMAL "QUADRATURES' WEIGHTS" IN UNIT 30.                00015900
           C                                                                       00018400
           C PROGRAMMED BY OSCAR L. COLOMBO, GEOD. SC., OHIO STATE U., SEPT. 1979. 00018500
           C                                                                       00018600
0002             IMPLICIT REAL*8(A-H,O-Z)                                          00018700
0003             DIMENSION ROWP(1),ROWQ(1),RHS(1),S(1),A(1),UL(1)                  00018800
0004             DIMENSION ANMPQ(200),SS(1),B(200),X(200),FF(200),
                2 IDD(200),BT(200),FC(1),PN(1),W(1),
                3 PHI(200),FINMP(200),FRADN(200),DVAR(1)
0005             DIMENSION QUADS(1)
0006             DIMENSION XHS(1)
0007             COMMON/MM/ MT(310)
           C                                                                       00019200
           C             BASIC CONSTANTS                                           00019300
           C                                                                       00019400
0008             PI = 3.141592653589793D0
0009             DRCONV = PI/180.D0
0010             F = 1.D0/298.257D0                                                00019700
           C                                                                       00019800
           C             PROBLEM DEFINITION                                        00019900
           C                                                                       00020000
           C                DGRID:     BLOCK SIZE IN DEGREES. MINIMUM:  5   DEGREES .
           C                ICEO       = 2  IF  GEODETIC  LATITUDES ARE TO BE CONVERTE00020200
           C                           TO GEOCENTRIC.                              00020300
           C                NN         MAX. DEG. AND ORDER TO BE ANALYZED (LESS THAN N00020400
           C                NC         NO. OF "COLUMNS", ALWAYS EVEN,              00020500
           C                NR         NO. OF ROWS. THERE IS NO EQUATORIAL ROW.    00020600
           C                NMAX       MAX. DEGREE IN BAND LIMITED COVARIANCE FUNCTION00020700
           C                NRUN       = 0   IF  R(M)'S ARE CREATED AND STORED IN
           C                           UNIT 15.
           C                           = 999 IF R(M)'S ARE READ FROM UNIT  15  INSTEAD00021100
           C                           OF BEING CREATED.                           00021200
           C             REGUL         REGULARIZATION PARAMETER. ADDED TO ALL DIAGONAL00021300
           C                           TERMS OF THE  R(M)  MATRICES.               00021400
           C                                                                       00021500
0011             DCR = DGRID*DRCONV                                                00021700
0012             NC = 360.D0/DGRID+0.000001D0                                      00022000
0013             NC4 = NC*0.25+0.000001D0                                          00022100
0014             NC2 = NC4*2                                                       00022200
0015             NR = NC2
0016             NN1 = NN+1                                                        00022300
0017             NP = 1                                                            00022400
0018             NP1 = NMAX+1                                                      00022500
0019             NPP = (NP1*(NP1+1))/2                                             00022600
0020             NLL = (NN1*(NN1+1))/2                                             00022700
0021             NPP8 = 8*NPP                                                      00022800
0022             NLL8 = 8*NLL                                                      00022900
```

```
            NC21 = NC2+1                                                    00023000
            NC28 = 8*NC2                                                    00023100
            FF(I) = DCR**2                                                  00023200
            NR2 = NR/2
            BLK = DGRID*DRCONV
            BLICK = DSQRT(DSIN(BLK)*BLK/PI)/DRCONV
            CALL QUADFS(QUADS,NN1,BLICK,NR2)
            DVAR0 = 0.D0                                                    00023300
            DO 5    I = 1,NC
            FC(I) = 0.D0                                                    00023500
            PN(I) = 0.D0
          5 CONTINUE
C                                                                           00023800
C                   FORM ARRAY OF GEOCENTRIC OR GEODETIC GRID'S GEOCENTRIC LATS00023900
C                                                                           00024000
            EN = F/(2.D0-F)                                                 00024100
            EM = 2.D0*EN/(1.D0+EN**2)                                       00024200
            NC2P = NC2+1                                                    00024300
            ANG = 90.D0                                                     00024400
            DO 6        I = 1,NC2P                                          00024500
            PHI(I) = ANG*DRCONV                                             00024600
            IF(ICEO.EQ.2) PHI(I) = PHI(I)-EM*DSIN(2.D0*PHI(I))              00024700
          6 ANG = ANG-DGRID                                                 00024800
            DO 15     II = 2,NP1                                            00025900
            I = II-1                                                        00026000
            FINMP(II) = DSQRT(DVAR(I)/(2.D0*I+1))                           00026100
         15 CONTINUE                                                        00026200
            FINMP(I) = DSQRT(DVAR0)                                         00026300
C                                                                           00026400
C                   READ TAPE OF INTEGRALS OF LEGENDRE FUNCTIONS FROM UNIT  8 ,00026500
C                   AND MULTIPLY THEM BY THE SQUARE ROOT OF THE DEGREE VARIANCE
C                   PER COEFFICIENT, TIMES THE INVERSE OF THE BLOCK AREA
C                   (FROWIN), STORING THEM IN UNIT  10  .
C                                                                           00027200
            IRS = -NN1
            DO 25         NT = 1,NC2                                        00027500
            IRS = IRS+NN1
            FROWIN = 1.D0/((DSIN(PHI(NT))-DSIN(PHI(NT+1)))*DCR)             00027700
            CALL FREAD(ROWP(1),8,NPP8,8999,8999)
            IF(NT.GT.NC4) GO TO 19
            NUS = 1
            DO 18       NI = 1,NN1
            DO 18      MI = 1,NI
            ROWQ(NUS) = ROWP(NUS)*QUADS(IRS+NI)
         18 NUS = NUS+1
            CALL FWRITE(ROWQ(1),30,NLL8,8999,8999)
         19 CONTINUE
            NIS = 1                                                         00028700
            DO 20       I = 1,NP1                                           00028800
            FX = FINMP(I)*FROWIN                                            00028900
            DO 20     IL = 1,I                                              00029000
            ROWP(NIS) = ROWP(NIS)*FX                                        00029100
         20 NIS = NIS+1                                                     00029200
            CALL FWRITE(ROWP(1),10,NPP8,8999,8999)                          00029300
         25 CONTINUE                                                        00030200
            ND = NR*NN1                                                     00030400
            NR8 = NR*8                                                      00030500
            WRITE(6,555) DGRID, NMAX, NN, NR, NRUN                          00030600
```

```
0072          555 FORMAT(//' BLOCK SIZE ',F7.2,'  MAX. DEGREE IN COVARIANCE ',I5,'  00030700
                 2 MAX. DEGREE TO BE ANALYZED ',I5,'  NUMBER OF ROWS ',I5,' NRUN ', 00030800
                 3 I5//)                                                           00030900
0073              WRITE(6,556) NC                                                  00031000
0074          556 FORMAT('    NUMBER OF COLUMNS ',I5//)                            00031100
0075              IF(ICEO.EQ.1) WRITE(6,557)                                       00031200
0076              IF(ICEO.EQ.2) WRITE(6,558)                                       00031300
0077          557 FORMAT('    DATA ON A SPHERE'//)                                 00031400
0078          558 FORMAT('    DATA ON AN ELLIPSOID'//)                             00031500
0079              DO 99        M1 = 1,NP1                                          00031600
0080           99 IDD(M1) = ((M1-1)*M1)/2                                          00031700
0081              DO 100        M1 = 2,NP1                                         00031800
0082          100 FF(M1) = 2.D0*(1.D0-DCOS((M1-1)*DGR))/(M1-1)**2                  00031900
              C                                                                    00032000
              C            READ THE R.H.S.'S OF THE REDUCED "NORMALS" FROM UNIT  10  .00032100
              C            INTO VECTOR  RHS  .                                     00032200
              C                                                                    00032300
0083              CALL REWIND(10)                                                  00032400
0084              CALL REWIND(30)
0085              ISH = -NLL                                                       00032500
0086              DO 210        I = 1,NC4                                          00032600
0087              CALL FREAD(ROWP(1),10,NLL8,8999,8999)
0088              CALL FREAD(ROWQ(1),30,NLL8,8999,8999)
0089              ISH = ISH+NLL                                                    00032800
0090              DO 210        J = 1,NLL                                          00032900
0091              XHS(ISH+J) = ROWQ(J)
0092          210 RHS(ISH+J) = ROWP(J)                                             00033000
0093              IF(NRUN.EQ.999) GO TO 1500                                       00033100
              C                                                                    00033200
              C       MAIN LOOP FOR THE CREATION OF THE  R(M)'S  MATRICES.         00033300
              C                                                                    00033400
0094              CALL REWIND(10)                                                  00033500
0095              CALL REWIND(30)
0096         1000 CONTINUE                                                         00033600
0097              DO 101        I = 1,ND                                           00033700
0098          101 S(I) = 0.D0                                                      00033800
              C                                                                    00033900
              C            POSITION  THE  INMP'S TAPE TO CURRENT ROW  NP .         00034000
              C                                                                    00034100
0099              DO 102        NX = 1,NP                                          00034200
0100          102 CALL FREAD(ROWP(1),10,NPP8,8999,8999)                            00034300
0101              IS = (NP-2)*NN1                                                  00034400
0102              NCV = NC2-NP+1
0103              DO 120        NQ = NP,NCV
0104              IF(NQ.NE.NP) CALL FREAD(ROWQ(1),10,NPP8,8999,8999)               00034600
0105              IF(NQ.NE.NP) GO TO 105                                           00034700
0106              DO 103        I = 1,NPP                                          00034800
0107          103 ROWQ(I) = ROWP(I)                                                00034900
              C                                                                    00016500
              C            OBTAIN THE FOURIER COEFFICIENTS OF THE BLOCK COVARIANCES  00016600
              C            CORRESPONDING TO ROWS  P  AND  Q   IN  CDD.             00016700
              C            AND ALIASE THEM TO FORM THE ELEMNTS OF THE  R(M)  MATRICES,00016800
              C                                                                    00016900
0108          105 CONTINUE                                                         00017000
0109              DO 1      I = 1,310                                              00017100
0110              UL(I) = 0.D0
0111              MT(I) = 0                                                        00017200
0112            1 CONTINUE                                                         00017300
```

```
0113                DO 111       MI = 1,NN1                                          00017400
0114                M = MI-1                                                         00017500
0115                H = NC2                                                          00017600
0116                IF(M.EQ.0.OR.M.EQ.NC2) H = NC                                    00017700
0117                CALL FUR(AN,MI,ROWP,ROWQ,IDD,FF,NP1)                             00017800
0118                ANMPQ(MI) = AN                                                   00017900
0119                N1 = NC-M+1                                                      00018000
0120                N2 = NC+M+1                                                      00018100
0121            112 CONTINUE                                                         00018200
0122                IF(N1.GT.NP1) GO TO 111                                          00018300
0123                CALL FUR(AN1,N1,ROWP,ROWQ,IDD,FF,NP1)                            00018400
0124                ANMPQ(MI) = ANMPQ(MI)+AN1                                        00018500
0125                IF(N2.GT.NP1) GO TO 108                                          00018600
0126                CALL FUR(AN2,N2,ROWP,ROWQ,IDD,FF,NP1)                            00018700
0127                ANMPQ(MI) = ANMPQ(MI)+AN2                                        00018800
0128            108 N1 = N1+NC                                                       00018900
0129                N2 = N2+NC                                                       00019000
0130                GO TO 112                                                        00019100
0131            111 ANMPQ(MI) = ANMPQ(MI)*H                                          00019200
0132                IS = IS+NN1                                                      00019300
0133                DO 114          NL = 1,NN1                                       00019400
0134                S(IS+NL) = ANMPQ(NL)                                             00019500
0135            114 CONTINUE                                                         00019600
0136            120 CONTINUE                                                         00019700
              C                                                                      00019800
              C             STORE THE LOWER DIAGONAL PART OF  R(M)  COLUMN BY COLUMN  00019900
              C             ON DISK FILE (UNIT  15  ).                               00020000
              C                                                                      00020100
0137                DO 116       M2 = 1,NN1                                          00020200
0138                IS = -NN1                                                        00020300
0139                DO 115          I = 1,NC2                                        00020400
0140                IS = IS+NN1                                                      00020500
0141            115 SS(I) = S(IS+M2)                                                 00020600
0142                CALL FWRITE(SS(1),15,NC28,8999,8999)                             00020700
0143            116 CONTINUE                                                         00020800
0144                CALL REWIND(10)                                                  00020900
0145                NP = NP+1                                                        00021000
0146                IF(NP.LE.NC4) GO TO 1000                                         00021100
0147           1500 CONTINUE                                                         00021200
              C                                                                      00022000
              C             BEGIN MAIN LOOP WHERE ERRORS ARE ESTIMATED.              00022100
              C                                                                      00022200
0148                DO 300         MI = 1,NN1                                        00022300
0149                CALL REWIND(15)                                                  00022400
0150                FFF = FF(MI)*NC                                                  00022500
0151                M = MI-1                                                         00022600
0152                IF(M.EQ.0) GO TO 214                                             00022700
0153                DO 213          I = 1,M                                          00022800
0154            213 CALL FREAD(SS(1),15,NC28,8999,8999)                              00022900
              C                                                                      00023000
              C        READ IN MATRIX  R(M)  FROM UNIT  15 , COLUMN BY COLUMN,       00023100
              C             AND PUT IT IN SYMMETRIC STORAGE FORM IN VECTOR  A .      00023200
              C                                                                      00023300
0155            214 DO 220        J = 1,NC4                                          00023400
0156                NC2J1 = NC2-J+1                                                  00023500
0157                IF(J.EQ.1) GO TO 222                                             00023600
0158                DO 215          I = 1,NN                                         00023700
0159            215 CALL FREAD(SS(1),15,NC28,8999,8999)                              00023800
```

```
  222 CALL FREAD(SS(I),15,NC28,8999,8999)                            00023900
      DO 220      I = J,NC2                                          00024000
      IF(I.GT.NC2J1) GO TO 220                                       00024100
      K = IDD(I)+J                                                   00024200
      NC21I = NC21-I                                                 00024300
      KP = IDD(NC2J1)+NC21I                                          00024400
      A(K) = SS(I)                                                   00024500
      A(KP) = SS(I)                                                  00024600
      IF(I.NE.J) GO TO 220                                           00024700
      A(K) = A(K)+W(J)                                               00024800
      A(KP) = A(KP)+W(NC2J1)                                         00024900
  220 CONTINUE                                                       00025000
C                                                                    00045000
C            FORM  B(N,M), X(N,M), AND COMPUTE THE THEORETICAL
C            ERRORS PER DEGREE .
C                                                                    00045200
      DO 240        N1 = M1,NN1                                      00045300
      SNM = 1.D0                                                     00045400
      IF(((N1-M1)/2)*2.NE.(N1-M1)) SNM = -1.D0                       00045500
      ISH = -NLL                                                     00045600
      KT = IDD(N1)+M1                                                00045700
      DO 230        I = 1,NC4                                        00045800
      ISH = ISH+NLL                                                  00045900
      B(I) = RHS(ISH+KT)*FINMP(N1)                                   00046000
      X(I) = XHS(ISH+KT)
      NC21I = NC21-I                                                 00046100
      B(NC21I) = B(I)                                                00046200
      X(NC21I) = X(I)
      IF(SNM.EQ.-1.D0)B(NC21I) = -B(I)
      IF(SNM.EQ.-1.D0) X(NC21I) = -X(NC21I)
  230 CONTINUE
      PNOISE = 0.D0                                                  00047300
      FCSD = 0.D0                                                    00047400
      DO 232        I = 1,NR                                         00047500
      FCSD = FCSD+X(I)*B(I)
  232 PNOISE = PNOISE+X(I)**2*W(I)
      SO = 0.D0                                                      00049600
      DO 245           I = 1,NC2                                     00049800
      BT(I) = 0.D0                                                   00049900
      DO 244        J = 1,NC2                                        00050000
      KT = IDD(I)+J                                                  00050100
      IF(J.GT.I) KT = IDD(J)+I                                       00050200
  244 BT(I) = BT(I)+X(J)*A(KT)                                       00050300
  245 SO = SO+BT(I)*X(I)
      FC(N1) = FC(N1)+FFF*(2.D0*FCSD-SO)
      PN(N1) = PN(N1)+FFF*PNOISE
      UL(N1) = UL(N1)+SO*FFF
      CALL FWRITE(X(I),30,NR8,8999,8999)
  240 CONTINUE                                                       00049100
  300 CONTINUE                                                       00051400
      DO 310        I = 3,NN1
      ECOEFN = DVAR(I-1)-FC(I)                                       00052600
      ALIAS = ECOEFN-PN(I)                                           00052700
      FC(I) = ECOEFN
      SS(I) = ALIAS
  310 CONTINUE                                                       00053400
      CALL REWIND(10)
      CALL REWIND(15)
```

```
0214                CALL REWIND (30)                                00051500
0215            999 CONTINUE                                        00053500
0216                RETURN                                          00053600
0217                END                                             00053700
```

$$\hat{C}^{\alpha}_{nm} = (\underline{f}^{\alpha}_{nm})^{T}\ \underline{m}$$

$$= \mu_n \sum_{i=0}^{N-1} \sum_{j=0}^{2N-1} \int_{\sigma_{ij}} \overline{Y}^{\alpha}_{nm}(\theta,\lambda)\, d\sigma\ m_{ij}$$

$$= \sum_{i=0}^{N-1} \left[\chi_i^{nm} \sum_{j=0}^{2N-1} \begin{Bmatrix} A(m) \\ B(m) \end{Bmatrix} \cos mj\Delta\lambda + \begin{Bmatrix} B(m) \\ A(m) \end{Bmatrix} \sin mj\Delta\lambda \right] m_{ij}$$

the estimator vector is a combination of a "sine" and a "cosine" vector of "frequency" m (terminology introduced in paragraph (2.10)). The product of such vector by $(C_{zz} + D)$ is, because of the structure of this matrix, another combination of a "sine" and a "cosine" vector of the same "frequency". From the properties of the normal matrix follows that

$$(C_{zz} + D)\,\underline{f}^{\alpha}_{nm} = \begin{Bmatrix} 2N & \text{if } m = 0 \\ & \text{or } m = N \\ N & \text{if } m \neq 0 \end{Bmatrix} \left[\ldots\ \nu_i^{nm} \begin{Bmatrix} A(m) \\ B(m) \end{Bmatrix} \cos mj\Delta\lambda + \begin{Bmatrix} B(m) \\ A(m) \end{Bmatrix} \sin mj\Delta\lambda\ \ldots \right]^{T}$$

where, calling

$$\underline{\nu}^{nm} = [\nu_0^{nm}, \nu_1^{nm}, \ldots \nu_{N-1}^{nm}]^{T} = R(m)\,[\chi_0^{nm}, \chi_1^{nm}, \ldots \chi_{N-1}^{nm}]^{T}$$

and

$$F(m) = \begin{cases} 2N\,\Delta\lambda^2 & \text{if } m = 0 \text{ or } m = N \\ 4N\,\dfrac{(1-\cos m\Delta\lambda)}{m^2} & \text{if } m \neq 0,\ m \neq N \end{cases}$$

is

$$(\underline{f}^{\alpha}_{nm})^{T}\,(C_{zz} + D)\ \underline{f}^{\alpha}_{nm} = F(m) \sum_{i=0}^{N-1} \chi_i^{nm}\,\nu_i^{nm} \qquad \text{(B4-a)}$$

Regarding the scalar product $2\underline{c}^{T}_{nm\alpha,z}\,\underline{f}^{\alpha}_{nm}$, it is easy to show that

$$2\underline{c}^{T}_{nm\alpha,z}\ \underline{f}^{\alpha}_{nm} = 2\,\frac{\sigma_n^2}{2n+1} \sum_{i=0}^{N-1} (\chi_i^{nm})^2\ F(m) \qquad \text{(B4-b)}$$

Expressions (B 4-a) and (B 4-b) are implemented by NORMAX to obtain the error variances. This subroutine also uses FUR . Subroutine QUADRS is also called, to obtain the de-smoothing factors. In the case listed here, this factor is $\mu_n = \frac{1}{4\pi\beta_n}$. Array QUADS has been added to the list of arguments, and it contains the μ_n after the call to QUADFS .

B.6 Subroutine LEGFDN

This subroutine computes the normalized Legendre functions and, if so desired, their derivatives at a given θ_i . All values correspond to the same order M ; if more than one order at a time is needed, a DO loop, where the subroutine is called once for each order, can be set up. The subroutine is based on formulas (4.19 a-b) and (1.38 a-b). The use of this subroutine is explained by the comments inserted in the listing. The stability of the recursive formulas was tested by computing $\overline{P}_{nm}(\cos\theta)$ and $(d\overline{P}_{nm}/d\theta)(\cos\theta)$ for m = 350 and $350 \leq n \leq 400$, $2.5^{\circ} \leq \theta \leq 90^{\circ}$, at 5° intervals. Calculations were done first in double precision (8 byte words) and then repeated in extended precision

```
      SUBROUTINE LEGFDN(M,THETA,RLEG,DLEG,NMX,IR,RLNN,IFLAG)
C
C           THIS SUBROUTINE COMPUTES  ALL NORMALIZED LEGENDRE FUNCTIONS
C           IN "RLEG" AND THEIR DERIVATIVES IN "DLEG". ORDER IS ALWAYS
C           M , AND COLATITUDE IS ALWAYS THETA  (RADIANS). MAXIMUM DEGREE
C           IS  NMX  . ALL CALCULATIONS IN DOUBLE PRECISION.
C           IR  MUST BE SET TO ZERO BEFORE THE FIRST CALL TO THIS SUB.
C           THE DIMENSIONS OF ARRAYS  RLEG, DLEG, AND RLNN  MUST BE
C           AT LEAST EQUAL TO  NMX+1  .
C           IF THIS SUBROUTINE IS TO BE USED TO COMPUTE FUNCTIONS
C           AND THEIR DERIVATIVES FOR MORE THAN ONE ORDER M, THEN
C           THE HIGHEST ORDER SHOULD BE COMPUTED IN THE FIRST CALL.
C
C           THIS PROGRAM DOES NOT COMPUTE DERIVATIVES AT THE POLES .
C
C           IF    IFLAG = 1  , ONLY THE LEGENDRE FUNCTIONS ARE
C           COMPUTED.
C
C      PROGRAMMER :  OSCAR L. COLOMBO, DEPT. OF GEODETIC SCIENCE,
C      THE OHIO STATE UNIVERSITY, AUGUST 1980 . ***************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION RLEG(1),DLEG(1),RLNN(1)
     2, DRTS(800),DIRT(800)
      NMX1 = NMX+1
      NMX2P = 2*NMX+1
      M1 = M+1
      M2 = M+2
      M3 = M+3
      IF(IR.EQ.1) GO TO 10
      IR = 1
      DO 5     N = 1,NMX2P
      DRTS(N) = DSQRT(N*1.D0)
    5 DIRT(N) = 1.D0/DRTS(N)
   10 COTHET = DCOS(THETA)
      SITHET = DSIN(THETA)
      SITHI = 1.D0/SITHET
C
C           COMPUTE THE LEGENDRE FUNCTIONS .
C
      RLNN(1) = 1.D0
      RLNN(2) = SITHET*DRTS(3)
      DO 15     N1 = 3,M1
      N = N1-1
      N2 = 2*N
   15 RLNN(N1) = DRTS(N2+1)*DIRT(N2)*SITHET*RLNN(N1-1)
      IF(M.GT.1) GO TO 20
      IF(M.EQ.0) GO TO 16
      RLEG(2) = RLNN(2)
      RLEG(3) = DRTS(5)*COTHET*RLEG(2)
      GO TO 20
   16 RLEG(1) = 1.D0
      RLEG(2) = COTHET*DRTS(3)
   20 CONTINUE
      RLEG(M1) = RLNN(M1)
      RLEG(M2) = DRTS(M1*2+1)*COTHET*RLEG(M1)
      DO 30     N1 = M3,NMX1
      N = N1-1
```

```
0035             IF(M.EQ.0.AND.N.LT.2.OR.M.EQ.1.AND.N.LT.3) GO TO 30
0036             N2 = 2*N
0037             RLEG(N1) = DRTS(N2+1)*DIRT(N+M)*DIRT(N-M)*(DRTS(N2-1)*COTHET*
               2 RLEG(N1-1)-DRTS(N+M-1)*DRTS(N-M-1)*DIRT(N2-3)*RLEG(N1-2))
0038             GO TO 30
0039          30 CONTINUE
0040             IF(IFLAG.EQ.1) RETURN
0041             IF(SITHET.EQ.0.D0) WRITE(6,99)
0042          99 FORMAT(//' *** LEGFDN  DOES NOT COMPUTE DERIVATIVES AT THE POLES
               2 ****************'//)
0043             IF(SITHET.EQ.0.D0) RETURN
        C
        C               COMPUTE ALL THE DERIVATIVES OF THE LEGENDRE FUNCTIONS.
        C
0044             RLNN(1) = 0.D0
0045             RLN = RLNN(2)
0046             RLNN(2) = DRTS(3)*COTHET
0047             DO 40       N1 = 3, M1
0048             N = N1-1
0049             N2 = 2*N
0050             RLN1 = RLNN(N1)
0051             RLNN(N1) = DRTS(N2+1)*DIRT(N2)*(SITHET*RLNN(N)+COTHET*RLN)
0052             RLN = RLN1
0053          40 CONTINUE
0054             DLEG(M1) = RLNN(M1)
0055             DO 60       N1 = M2,NMX1
0056             N = N1-1
0057             N2 = N*2
0058             DLEG(N1) = SITHI*( N    *RLEG(N1)*COTHET-DRTS(N-M)*DRTS(N+M)*
               2 DRTS(N2+1)*DIRT(N2-1)*RLEG(N))
0059          60 CONTINUE
0060             RETURN
0061             END
```

(16 byte words). Both sets of results agreed with each other to better than 6 significant figures.

B. 7 Subroutine NVAR

This subroutine computes the degree variances of the gravity anomalies (point values), up to degree 100, according to the coefficients obtained by R. Rapp from a complete, equal angular set of mean $1^\circ \times 1^\circ$ anomalies, as mentioned in par. (3.1). Above $n = 100$, the subroutine uses a two-term model to calculate the $\sigma_n^2(\Delta g)$. The resulting degree variances are stored in array DVAR. The first element in DVAR is $\sigma_1^2(\Delta g)$.

```
0001                SUBROUTINE NVAR(DVAR,NMAX,DVAR0)
              C
              C             THIS SUBROUTINE COMPUTES ALL DEGREE VARIANCES UP TO
              C             DEGREE  "NMAX"  , USING SOME GIVEN MODEL .
              C
0002                IMPLICIT REAL*8 (A-H,O-Z)
0003                DIMENSION DVAR(1)
0004                DVAR0 = 0.D0
0005                DVAR(  1) =     .0
0006                DVAR(  2) =    7.6396518
0007                DVAR(  3) =    33.902065
0008                DVAR(  4) =    19.743429
0009                DVAR(  5) =    19.696103
0010                DVAR(  6) =    18.390314
0011                DVAR(  7) =    18.455112
0012                DVAR(  8) =    10.399362
0013                DVAR(  9) =    8.6398739
0014                DVAR( 10) =    9.1038027
0015                DVAR( 11) =    6.3690710
0016                DVAR( 12) =    2.8993199
0017                DVAR( 13) =    6.7286206
0018                DVAR( 14) =    3.2033050
0019                DVAR( 15) =    3.8198187
0020                DVAR( 16) =    4.9444726
0021                DVAR( 17) =    3.7168750
0022                DVAR( 18) =    4.1988269
0023                DVAR( 19) =    3.4360349
0024                DVAR( 20) =    2.2931356
0025                DVAR( 21) =    3.0053072
0026                DVAR( 22) =    3.5082009
0027                DVAR( 23) =    2.6197612
0028                DVAR( 24) =    2.6880078
0029                DVAR( 25) =    3.4761637
0030                DVAR( 26) =    2.2575367
0031                DVAR( 27) =    2.1490026
0032                DVAR( 28) =    2.5805299
0033                DVAR( 29) =    2.0568812
0034                DVAR( 30) =    2.9207785
0035                DVAR( 31) =    2.4778694
0036                DVAR( 32) =    2.4971826
0037                DVAR( 33) =    3.2040108
0038                DVAR( 34) =    3.9115457
0039                DVAR( 35) =    3.1070939
0040                DVAR( 36) =    2.9680243
0041                DVAR( 37) =    3.0126138
0042                DVAR( 38) =    2.1799143
0043                DVAR( 39) =    2.8229174
0044                DVAR( 40) =    2.6698616
0045                DVAR( 41) =    3.0049545
0046                DVAR( 42) =    2.8244301
0047                DVAR( 43) =    2.2442772
0048                DVAR( 44) =    3.0705790
0049                DVAR( 45) =    2.9169446
0050                DVAR( 46) =    2.7285179
0051                DVAR( 47) =    3.4346806
0052                DVAR( 48) =    2.8180541
0053                DVAR( 49) =    2.2040232
0054                DVAR( 50) =    3.4272534
```

```
0055                DVAR( 51) =     2.8176026                                       00057600
0056                DVAR( 52) =     2.6570577                                       00057700
0057                DVAR( 53) =     3.4527293                                       00057800
0058                DVAR( 54) =     3.0583738                                       00057900
0059                DVAR( 55) =     3.1636091                                       00058000
0060                DVAR( 56) =     2.8039749                                       00058100
0061                DVAR( 57) =     3.2159365                                       00058200
0062                DVAR( 58) =     2.8370499                                       00058300
0063                DVAR( 59) =     2.9285564                                       00058400
0064                DVAR( 60) =     3.2459338                                       00058500
0065                DVAR( 61) =     2.6989317                                       00058600
0066                DVAR( 62) =     3.3195559                                       00058700
0067                DVAR( 63) =     2.8978374                                       00058800
0068                DVAR( 64) =     2.7474396                                       00058900
0069                DVAR( 65) =     2.2005284                                       00059000
0070                DVAR( 66) =     3.4103538                                       00059100
0071                DVAR( 67) =     2.6096258                                       00059200
0072                DVAR( 68) =     2.5628852                                       00059300
0073                DVAR( 69) =     3.1792570                                       00059400
0074                DVAR( 70) =     2.4727662                                       00059500
0075                DVAR( 71) =     2.3711426                                       00059600
0076                DVAR( 72) =     2.7305264                                       00059700
0077                DVAR( 73) =     2.8756958                                       00059800
0078                DVAR( 74) =     2.9566934                                       00059900
0079                DVAR( 75) =     2.5071508                                       00060000
0080                DVAR( 76) =     2.5442190                                       00060100
0081                DVAR( 77) =     2.3198284                                       00060200
0082                DVAR( 78) =     2.6555035                                       00060300
0083                DVAR( 79) =     2.5151849                                       00060400
0084                DVAR( 80) =     2.8009285                                       00060500
0085                DVAR( 81) =     2.8064117                                       00060600
0086                DVAR( 82) =     3.1910590                                       00060700
0087                DVAR( 83) =     2.7446152                                       00060800
0088                DVAR( 84) =     3.0200313                                       00060900
0089                DVAR( 85) =     2.3415316                                       00061000
0090                DVAR( 86) =     2.6750626                                       00061100
0091                DVAR( 87) =     2.6775625                                       00061200
0092                DVAR( 88) =     2.5177941                                       00061300
0093                DVAR( 89) =     2.4932026                                       00061400
0094                DVAR( 90) =     2.4443577                                       00061500
0095                DVAR( 91) =     2.7512543                                       00061600
0096                DVAR( 92) =     2.3148969                                       00061700
0097                DVAR( 93) =     2.5410976                                       00061800
0098                DVAR( 94) =     2.5721650                                       00061900
0099                DVAR( 95) =     2.2715440                                       00062000
0100                DVAR( 96) =     2.5384265                                       00062100
0101                DVAR( 97) =     2.3560790                                       00062200
0102                DVAR( 98) =     2.6830127                                       00062300
0103                DVAR( 99) =     2.2631529                                       00062400
0104                DVAR(100) =     2.3381125                                       00062500
0105                IF(NMAX.LE.100) GO TO 20                                        00062600
0106                A = 1.D0                                                        00062700
0107                B = 2.D0                                                        00062800
0108                A1 = 3.4050D0                                                   00062900
0109                A2 = 140.03D0                                                   00063000
0110                S1 = 0.998006D0                                                 00063100
0111                S2 = 0.914232D0                                                 00063200
0112                DO 10      N = 101,NMAX                                         00063300
```

```
FORTRAN IV G1  RELEASE 2.0            NVAR              DATE = 80242          12/29/34              PAGE 0003

 0113                DVAR(N) = (N-1)*(S1**(N+2)*A1/(N+A)+S2**(N+2)*A2/((N+B)*(N-2)))   00063400
 0114             10 CONTINUE                                                        00063500
 0115             20 CONTINUE                                                        00063600
 0116                RETURN                                                          00063700
 0117                END                                                             00063800
```

B. 8 SUBROUTINE COVBLK

Subroutine COVBLK calculates covariances between area means according to expression (4.14). This subroutine is listed in the following pages. The arguments are explained in the comments at the beginning of the listing. In addition, the following things have to be born in mind: the dimension of the array DVAR is N_{max}; the dimension of both RINS and RINN is $\frac{1}{2}(N_{max}+1)(N_{max}+2)$; the dimension of COVS is 360/BLOCK. The values returned in DVAR are the original degree variances $c_n^{u,v}$, each divided by $2n+1$. If LB is less than 360/BLOCK, the LB+1, LB+2, ... ,(360/BLOCK - LB - 1) elements in COVS are returned as zeroes, the remainder contain the first (and last) LB covariances. The dimension of F is 180/BLOCK (Nyquist frequency). To use this subroutine with $N_{max} > 400$, the arrays FF and IDD (whose dimension should be no less than N_{max}), should be redimensioned.

The subroutine does not take advantage of the "aliasing" of Fourier coefficients built into expression (4.14). Implementing this aspect should lead to some additional improvement in efficiency. The Fourier series is computed, once the coefficients have been determined, by multiplying each coefficient by the corresponding cosine of $m\lambda_j$ and adding the products together. The values of $\cos m\lambda_j$ are computed using the following recursive formula:

$$\cos m\lambda_j = 2\cos\lambda_j \cos(m-1)\lambda_j - \cos(m-2)\lambda_j$$

which avoids repeated calculation of the FORTRAN COS function (only $\cos\lambda_j$ is required to start the recursion). Actual calculation of the Fourier series requires about 0.04 seconds in the most time consuming case: the grid of 1° blocks. The greater part of the time taken by this subroutine goes into finding the Fourier coefficients of the mean value covariances. For this reason, there is not much difference between computing all covariances in a certain row, or just a few of them, using this procedure.

```
FORTRAN IV G1  RELEASE 2.0          COVBLK          DATE = 80242       13/02/38                PAGE 0001

0001                SUBROUTINE COVBLK(DVAR,RINS,RINN,NMX,BLOCK,DLATS,DLATN,COVS,F,K,
                   2 LB)
              C
              C           THIS SUBROUTINE COMPUTES COVARIANCES BETWEEN AN AREA MEAN
              C           IN ROW "N"  AND  LB  CONTIGUOUS AREA MEANS IN ROW  "S",
              C           BEGINNING AT THE BLOCK WITH THE SAME LONGITUDE LIMITS AS
              C           THE ONE IN ROW  "N"  .
              C
              C           DVAR :      VECTOR OF DEGREE VARIANCES(PRECOMPUTED)
              C           RINS :      VECTOR OF INTEGRAL OF LEGENDRE FUNCTIONS, S. ROW
              C                       (PRECOMPUTED)
              C           RINN :      VECTOR OF INTEGRALS OF LEGENDRE FUNCTIONS, N. ROW
              C                       (PRECOMPUTED)
              C           NMX :       MAXIMUM DEGREE CONSIDERED IN THE EXPANSION
              C                       OF THE COVARIANCE FUNCTION.
              C           BLOCK :     SIZE OF  BLOCKS IN THE GRID (DEGREES).
              C           LB :        NUMBER OF CONTIGUOUS BLOCKS IN ROW  "S" .
              C           DLATS :     SOUTHMOST LAT. IN ROW "S" (SOUTH)(DEGREES)
              C           DLATN :     SOUTHMOST LAT. IN ROW "N" (NORTH) (DEGREES)
              C           COVS :      VECTOR OF COMPUTED COVARIANCES BETWEEN BLOCK MEANS
              C           F :         VECTOR OF FOURIER COSINE COEFFICIENTS OF THE
              C                       TRIGONOMETRIC EXPANSION OF THE BLOCK MEANS'
              C                       COVARIANCE FUNCTION
              C           K :         INTEGER*4 CONSTANT SET TO 0 BEFORE FIRST CALL TO
              C                       THIS SUBROUTINE.
              C
              C           PROGRAMMED BY OSCAR L. COLOMBO, DEPT. OF GEODETIC SC.,
              C           OHIO STATE UNIVERSITY, COLUMBUS, DECEMBER 1979 . **********
              C
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                DIMENSION DVAR(1),RINS(1),RINN(1),COVS(1),F(1),FF(400),IDD(400),C
                   2 (400)
0004                IF(K.NE.0) GO TO 10
0005                NP1 = NMX+1
0006                PI = 3.14159265358979300
0007                DRCONV = PI/180.D0
0008                NB = 360.D0/BLOCK +0.0000001D0
0009                NBH = NB/2
0010                NBHP = NBH+1
0011                BLK = BLOCK*DRCONV
0012                DO 1       I = 2,NP1
0013              1 FF(I) = 2.D0*(1.D0-DCOS((I-1)*BLK))/(I-1)**2
0014                DO 2       I = 1,NP1
0015              2 DVAR(I) = DVAR(I)/(2.D0*I-1.D0)
0016                FF(1) = BLK**2
0017                DO 3       M1 = 1,NP1
0018              3 IDD(M1) = ((M1-1)*M1)/2
0019                DO 4       I = 1,NB
0020              4 C(I) = DCOS(BLK*(I-1))
0021                K = 1
0022             10 RL1S = DLATS*DRCONV
0023                RL2S = DLATN*DRCONV
0024                RL1N = RL1S+BLK
0025                RL2N = RL2S+BLK
0026                FR = 1.D0/(((DSIN(RL1N)-DSIN(RL1S)) *BLK)*((DSIN(RL2N)-DSIN(RL2S)
                   2 )*BLK))
              C
              C           COMPUTE DE COSINE FOURIER COEFFICIENTS OF THE BLOCK
```

```
C               COVARIANCE FUNCTION .
C
          RA = 0.D0
          DO 20       M1 = 1,NP1
          F(M1) = 0.D0
          NXX = M1
          DO 18       N1 = M1,NP1
          IX = IDD(N1)+M1
       18 F(M1) = F(M1)+RINS(IX)*RINN(IX)*DVAR(N1)
          F(M1) = F(M1)*FF(M1)*FR
          FZ = F(M1)**2
          RA = RA+FZ
          RAG = RA*1.D-12
          IF(M1.GT.1.AND.F(M1-1)**2.LT.RAG.AND.FZ.LT.RAG) GO TO 21
       20 CONTINUE
C
C               FIND THE COVARIANCES BETWEEN BLOCKS BY FOURIER SYNTHESIS.
C
       21 DO 40        I = 1,NBHP
          COVS(I) = 0.D0
          IF(I.GT.LB) GO TO 40
          CNM1 = C(I)
          C2 = 2.D0*C(I)
          CN = 1.D0
          COVS(I) = F(1)
          DO 30       N1 = 2,NXX
          CNP1 = C2*CN-CNM1
          CNM1 = CN
          CN = CNP1
       30 COVS(I) = COVS(I)+CNP1*F(N1)
       40 CONTINUE
          DO 45         I = 1,NBH
          COVS(NB-I+1) = COVS(I+1)
       45 CONTINUE
          RETURN
          END
```