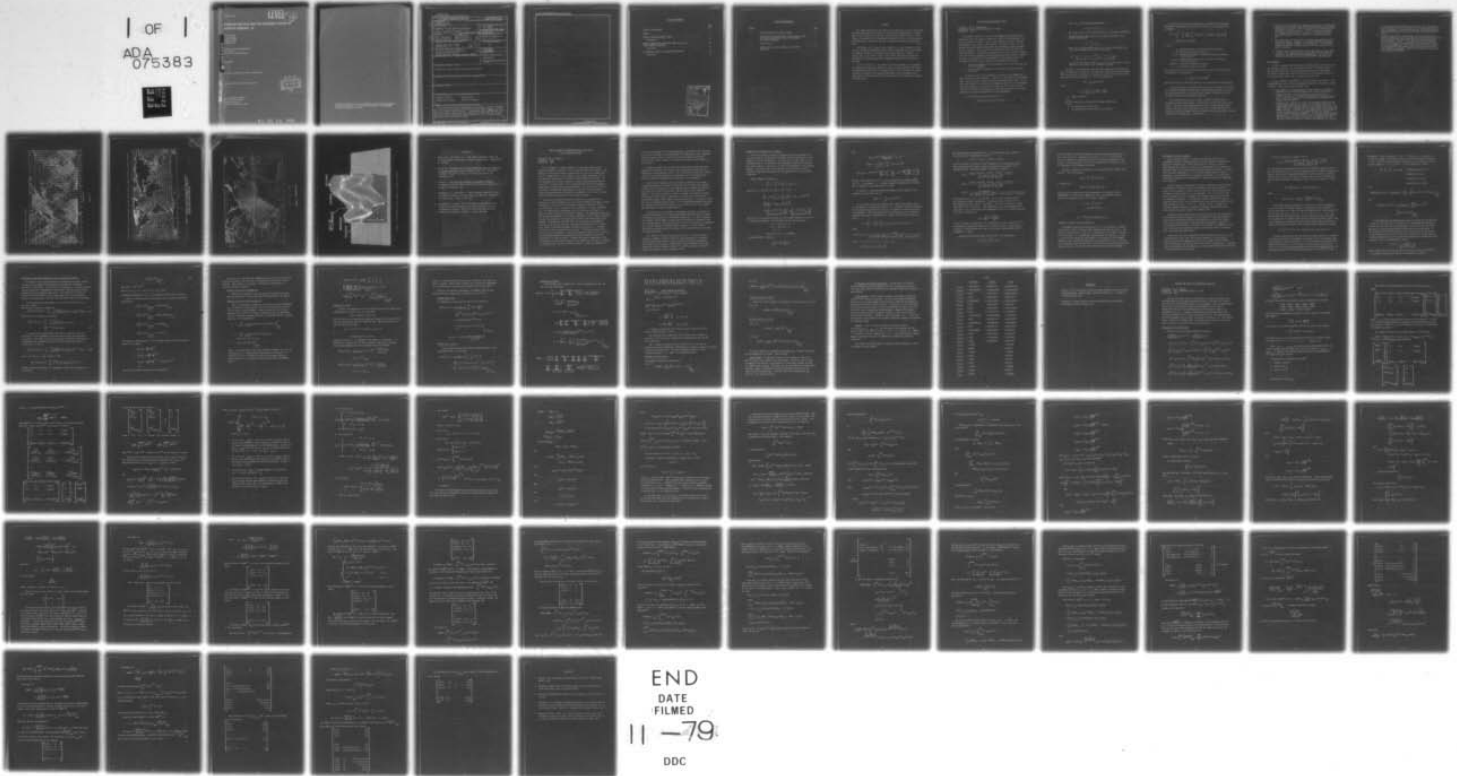


AD-A075 383

BOSTON COLL CHESTNUT HILL MASS SPACE DATA ANALYSIS LAB F/G 9/2
NUMERICAL AND DATA ANALYSIS TECHNIQUES APPLIED TO SCIENTIFIC RE--ETC(U)
MAY 79 N J GROSSBARD , J E MARTINE F19628-76-C-0147
UNCLASSIFIED BC-SDAL-79-1 AFGL-TR-79-0132 NL

| OF |
ADA
075383



END
DATE
FILMED
11 -79
DDC

AFGL-TR-79-0132

LEVEL #

12

**NUMERICAL AND DATA ANALYSIS TECHNIQUES APPLIED TO
SCIENTIFIC RESEARCH - III**

Neil J. Grossbard
Joseph E. Martine
Edward N. Richards
Theodore Persakis
Robert Boudreau

SPACE DATA ANALYSIS LABORATORY
Boston College
Chestnut Hill, Massachusetts 02167

31 May 1979

Final Report for period 1 January 1976 - 31 December 1978

Approved for public release; distribution unlimited

DDC
RECEIVED
OCT 23 1979
A

Air Force Geophysics Laboratory
Air Force Systems Command
United States Air Force
Hanscom AFB, Massachusetts 01731

DDC FILE COPY

AD A 075383

79 22 19 005

Qualified requestors may obtain additional copies from the Defense Documentation Center. All others should apply to the National Technical Information Service.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER 18 AFGL-TR-79-0132	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) 6 NUMERICAL AND DATA ANALYSIS TECHNIQUES APPLIED TO SCIENTIFIC RESEARCH • III, N. 14		5. TYPE OF REPORT & PERIOD COVERED 9 Final Report, 1 Jan 76 - 31 Dec 78.	
7. AUTHOR(s) 10 Neil J. Grossbard, Edward Richards, Joseph E. Martine, Theodore Persakis, Robert Boudreau		6. PERFORMING ORG. REPORT NUMBER 14 BC-SDAL-79-1	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Trustees of Boston College Chestnut Hill, MA Space Center 12 72		8. CONTRACT OR GRANT NUMBER(s) 15 F19628-76-C-0147	
11. CONTROLLING OFFICE NAME AND ADDRESS Analysis Lab. Air Force Geophysics Laboratory Hanscom AFB, Massachusetts Contract Monitor: Dr. Paul Tsipouras (SUWA)		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS N/A	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE 11 31 May 1979	
		13. NUMBER OF PAGES 72	
		15. SECURITY CLASS. (of this report) UNCLASSIFIED	
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited			
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)			
18. SUPPLEMENTARY NOTES			
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Numerical Analysis Data Analysis Mathematical Analysis Contour Integration			
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report describes the mathematical and numerical analysis techniques used in the solution of three of the major problem efforts engaged in under Air Force Contract F19628-76-C-0147. Although many other problems were solved under this contract, the analysis and techniques utilized were for the most part similar with variations to those described in the two previous reports (AFRL-TR-73-0433 and AFGL-TR-76-0091).			

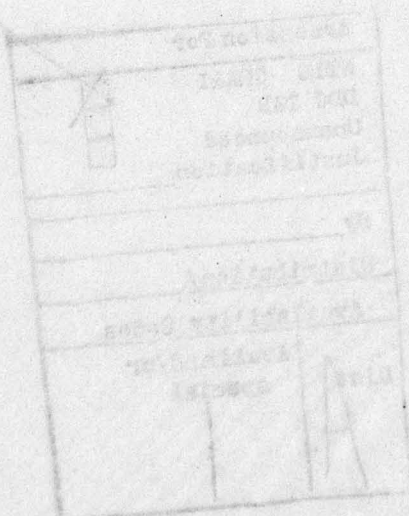
TABLE OF CONTENTS

	<u>Page</u>
LIST OF ILLUSTRATIONS	iv
PREFACE	v
EARTH'S GEOID AND GRAVITY FIELD	1
References	10
MATRIX ELEMENTS FOR BOUND/FREE THREE BODY STATES WITH COULOMB INTERACTION	11
References	30
ELECTRON AND PARTICLE DISTRIBUTION FUNCTIONS	31
References	67

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DDC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	<input type="checkbox"/>
By _____	
Distribution/ _____	
Availability Codes	
Dist	Availand/or special
A	

LIST OF ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	Ground Tracks of 1253 GEOS-3 Passes	6
2	Contours of Geoid Undulations (10m Interval) from the Global Adjustment of Satellite Altimetry and Gravity Anomalies	7
3	Contour Plots	8
4	North Atlantic Gravity Anomalies from GEOS-3 Altimetry	9



PREFACE

The effort described in this report is a sampling of the most interesting problems undertaken in this contract. Many of the problems were a continuation in kind of work done under Air Force Contract F19628-73-C-0136. That effort was summarized in a Final Report dated 31 March 1976, titled Numerical and Data Analysis Techniques Applied to Scientific Research-II, AFGL-TR-76-0091.

A034047

The authors wish to express their thanks to Dr. Paul Tsipouras of the Air Force Geophysics Laboratory (AFGL) for his assistance in the development of solutions to many of the problems considered under this contract. A similar debt of gratitude is owed to all Problem Initiators from AFGL for providing the detailed background information regarding their particular problems.

Mr. Leo F. Power, Jr., the Director of this laboratory, was invaluable in supplying the necessary resources to meet critical deadlines and finally, our appreciation goes out to the support staff of this laboratory for their continual assistance in the program preparation of these problems and in particular to Miss Mary Kelly for the excellent job of preparing this report for publication.

EARTH'S GEOID AND GRAVITY FIELD

Initiator: Mr. G. Hadgigeorge
Problem No: 4919 (formerly Problem No: 4850)
Project No: 7600

Of special interest to the Air Force Geophysics Laboratory (AFGL) through the years has been the reduction of satellite measurements to the various parameters measured. This is most certainly true for satellite measurements of the oceanic geoid surface. Of primary concern, is that approach which may remove the necessity for extremely accurate reference orbits in order to exploit satellite altimetry of 1 meter accuracy for geoid improvement. Conventional approaches utilize long-arc orbit integration that requires extensive tracking from ground based trackers. Whereas, the short-arc approach⁽¹⁾ in program SARRA has shown that accuracies of integration of better than 1 meter can be attained provided that

- a. spherical harmonics at least through $(n,m) = (4,4)$ are exercised in the integration
- b. all six orbital parameters $(x,y,z,\dot{x},\dot{y},\dot{z})$ at mid-arc are free to adjust.

At present, there are two options for selecting the mathematical model defining the geoid surface. The first is the spherical multiquadric that is based on least squares fitting of the observational data. If the area being studied is covered with an adequately dense set of measurements, then, this approach has no limit as to the size of the area to be processed or the detail to be obtained. There are two distinct but coupled mathematical formulations adhered to in the programming.

A. The basic observation equation⁽²⁾ is

$$f(H_{ij}, x_j, y_j, z_j, \dot{x}_j, \dot{y}_j, \dot{z}_j, r_o, c_n) = 0$$

where, H_{ij} is the altimetry measurement

$$(x_j, y_j, z_j, \dot{x}_j, \dot{y}_j, \dot{z}_j)$$

the state vector; r_0 the reference radius; c_n the geoid coefficient.

B. The geoidal model (here expressed in terms of spheroidal multi-quadric functions)⁽³⁾

$$r_i = \sum_{j=1}^n c_{ij} \psi_{ij}$$

where, r_i is the geocentric radius; c_{ij} is to be determined; ψ_{ij} is called the kernel function and of the form

$$\psi_{ij} = \frac{r_i^2}{a \left[(x_i - x_j + ka)^2 + (y_i - y_j + ka)^2 + (z_i - z_j + ka)^2 \right]^{1/2}}$$

x_i, y_i, z_i is the geoidal point; x_j, y_j, z_j is the node point; a is the semi-major axis; and k is the arbitrary fraction.

The other is the Covariance function whose advantage is realized when data is irregularly distributed, with some areas being sparse in measurements. This function $D(\psi)$ is obtained by averaging the product of undulations separated by the spherical distance ψ . It can be expressed as

$$D(\psi) = \sum_{n=2}^{\infty} d_n P_n(\cos\psi)$$

where,

$$d_n = \frac{R^2}{G^2 (n-1)^2} \sum_{n=0}^{\infty} (\overline{\Delta C_{nm}^2} + \overline{\Delta S_{nm}^2})$$

d_n degree variances

$\left. \begin{array}{l} \Delta C_{nm} \\ \Delta S_{nm} \end{array} \right\}$ corrections to the spherical harmonic coefficient

R the mean radius of the earth

G the mean gravity of the earth at the surface.

In an effort to decrease computational requirements the programmed mathematical formulation for the radial distance \bar{r}_i was replaced by the following:

$$\tilde{r}_i = r_c \left\{ 1 + \sum_{n=2}^N \left(\frac{a}{\tilde{r}_i} \right)^n \sum_{m=0}^n (C_{nm} \cos m \lambda_i + S_{nm} \sin m \lambda_i) P_{nm}(\sin \theta_i) \right\} + \frac{1}{2} w^2 r_o^2 \tilde{r}_i^3 \cos^2 \phi_i / (GM)$$

in which

a = equatorial radius of adopted reference spheroid

GM = gravitational constant of earth

θ, λ, r = geocentric latitude, longitude and radius on the mean sea level geoid

C_{nm}, S_{nm} = coefficients of the potential function expanded in spherical harmonics

$P_{nm}(\sin \theta)$ = associated Legendre polynomial

w = rotational velocity of earth.

The solution of r_i is to be obtained from an iterative process using the following for a starting value:

$$r_o = a / \left(1 + e^2 / (1 - e^2) \sin^2 \phi \right)^{1/2}$$

where a and ϕ are as defined previously, and e represents the eccentricity.

The observed data was obtained from in excess of 100 tapes of altimetry data received from NASA and NSWC (Naval Surface Weapons Center). This data was reduced and edited by the preprocessor computer program which builds the input data base for the SARRA reductions.

The preprocessing and editing (PREP) of GEOS-3 altimetry data is designed as a three level effort. The raw data is first edited for "gross" errors. The data is then smoothed and fitted to a polynomial and again edited. The third level of editing rejects data, either smoothed or unsmoothed, that have standard deviations greater than some specified criterion. The editing is accomplished in three steps,

- 1) "Gross" errors are detected by comparing the change in the measured altimetry (Δh), between succession data points, with the corresponding change in time (Δt). This is a measure of the altimetry rate (\dot{h}) which is a smooth function. If the altimetry rate exceeds the maximum value the point is rejected. The maximum for \dot{h} may be estimated empirically from several sets of GEOS-3 altimetry data.
- 2) The second level of editing uses a n^{th} order polynomial to smooth the altimetry data. Residuals are computed from the difference between the smoothed altimetry and the measured. If the residual exceeds 10 times the standard deviation of the measurements the point is rejected.
- 3) The third level of editing rejects data based upon an input sigma criterion. The criterion will be based upon realistic expected accuracy of the GEOS-3 altimeter measurements. This value will be adjusted as experience is gained with GEOS-3 altimetry data.

Plot Programs

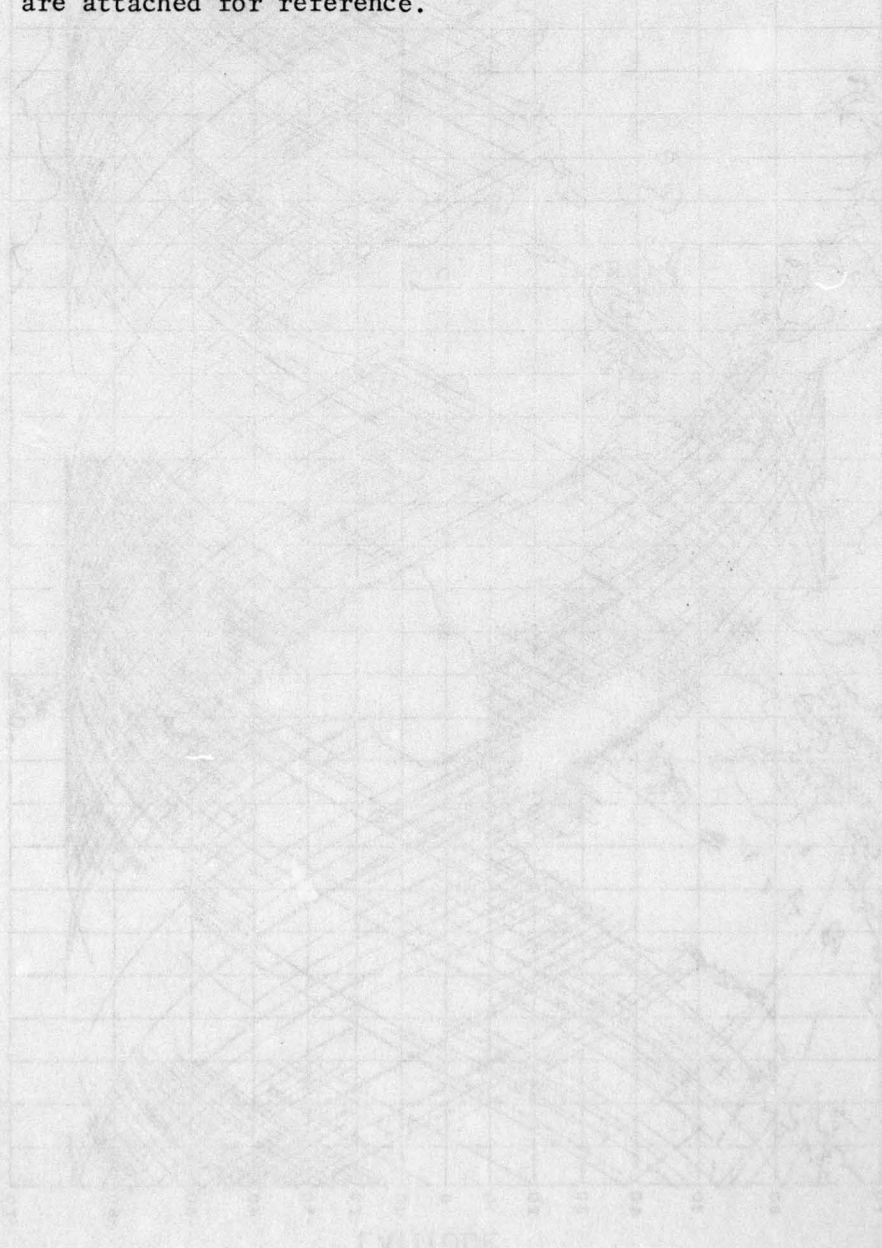
The plot programs perform on option satellite ground track plots, land outline plots, contours, and either geoid profiles or residual profiles. The land and island areas are plotted from digitized land coordinates. The normal procedure for formal presentation of the continent outlines is to plot the ground tracks over a reproduced accurate map.

The plot programs are merged into a single program called WMAP. This program WMAP consists of several independent functions that may be selected on option. These options are described below.

- a. Land Outlines - The land outlines of the earth may be plotted in combination with either the ground track or the contour plots.
- b. Ground Track Plot - This program plots the ground track of satellite passes over the globe from ephemeris latitude and longitude. Additional parallels and meridians may be selected with program input parameters.
- c. Contour Plots - Contours of either the standard deviations or the actual geoid heights may be plotted. The standard deviations are computed from the covariance matrix from the SARRA reductions. The geoid heights are computed from the surface coefficients saved from SARRA solution. In either case the computations are performed for a grid of surface points. The program includes the option to process unordered geoid measurements. The original program required measurements at locations equally spaced in longitude and latitude. This later grid, in essence, outlines a rectangular area. The option is of the form that the unordered measurements are searched

to provide points approximating those on the grid. For most of the grid these points are available and the corresponding contour plot is obtained. A three-dimensional plot of the geoid height over the North Atlantic region was produced (see plotting attachments).

- d. Residual Profile Plot - The altimeter observational residuals (measured minus computed height) obtained from the SARRA reductions are plotted for the purpose of visually reviewing measurement characteristics. The residual profile plot provides immediate display of altimeter measurement noise and such oceanic details as sea state variations. The resulting output from this effort was published^(4,5) and communicated by the Problem Initiator to other interested personnel. Several plots depicting this output are attached for reference.



GROUND TRACKS OF 1253 GEOS-3 PASSES

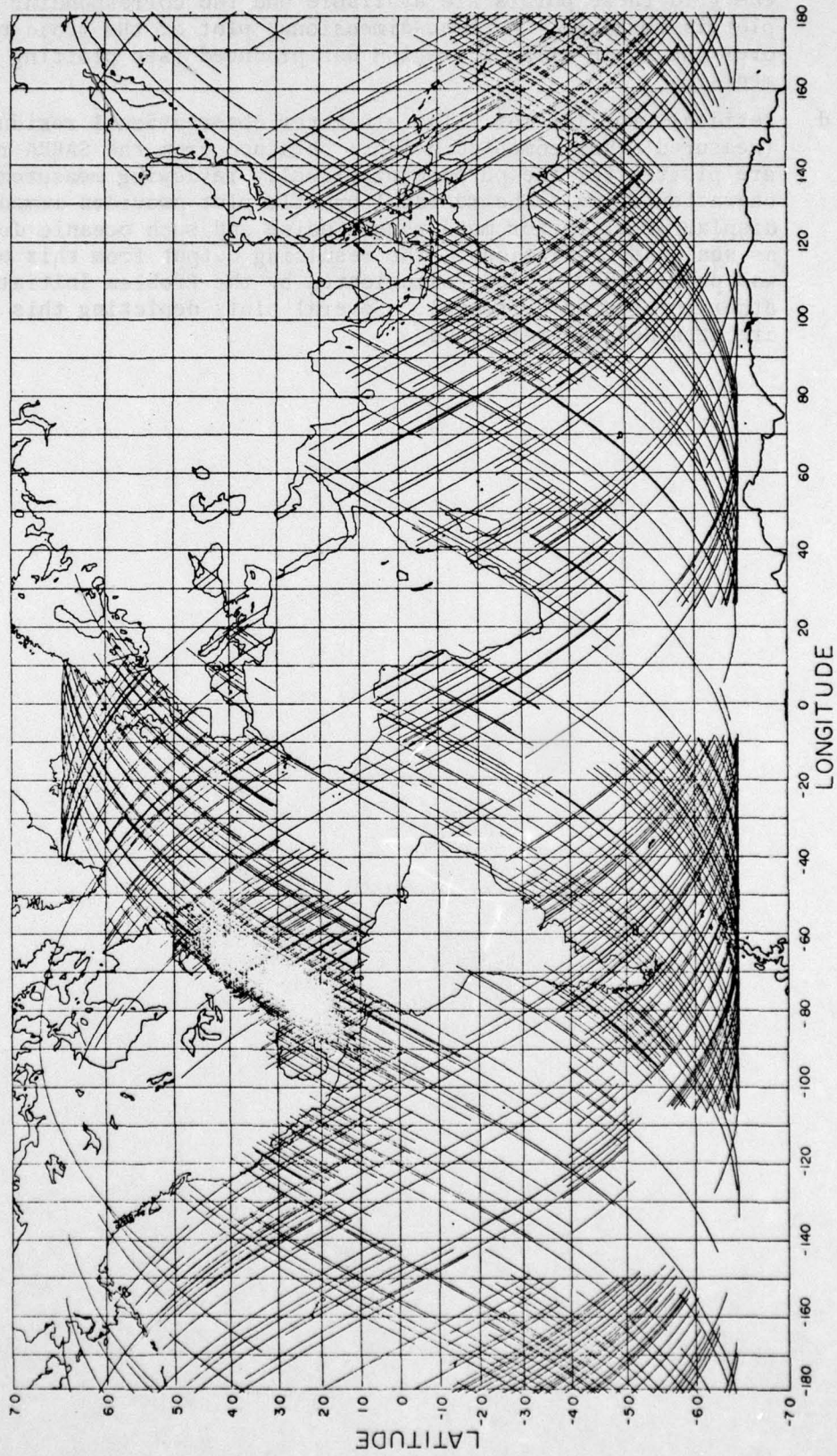


Figure 1.

CONTOURS OF GEOID UNDULATIONS (10m INTERVAL) FROM THE GLOBAL ADJUSTMENT OF SATELLITE ALTIMETRY AND GRAVITY ANOMALIES

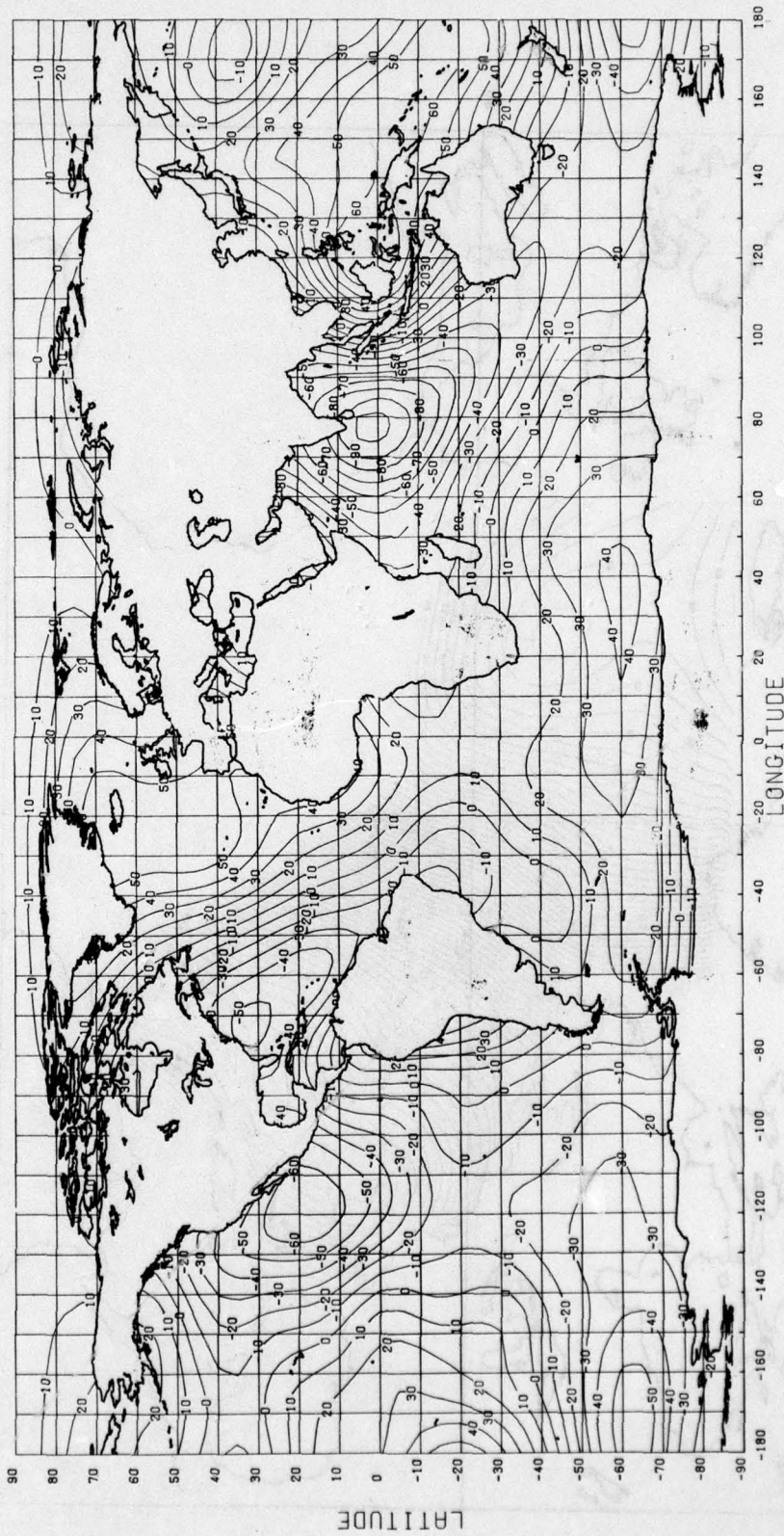


Figure 2. Contours of Geoid Undulations (10m Interval) from the Global Adjustments of Satellite Altimetry and Gravity Anomalies

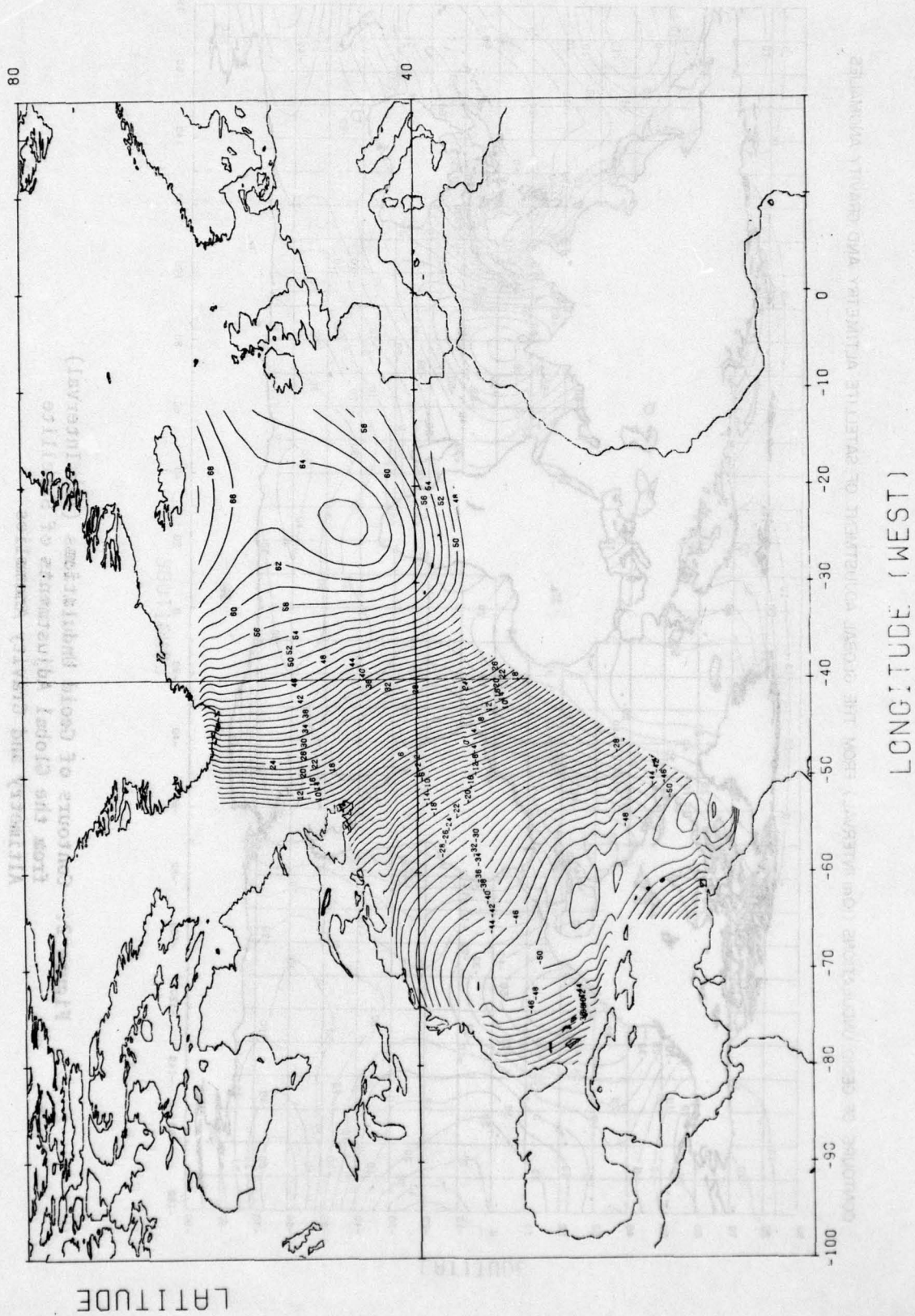


Figure 3. Contour Plots

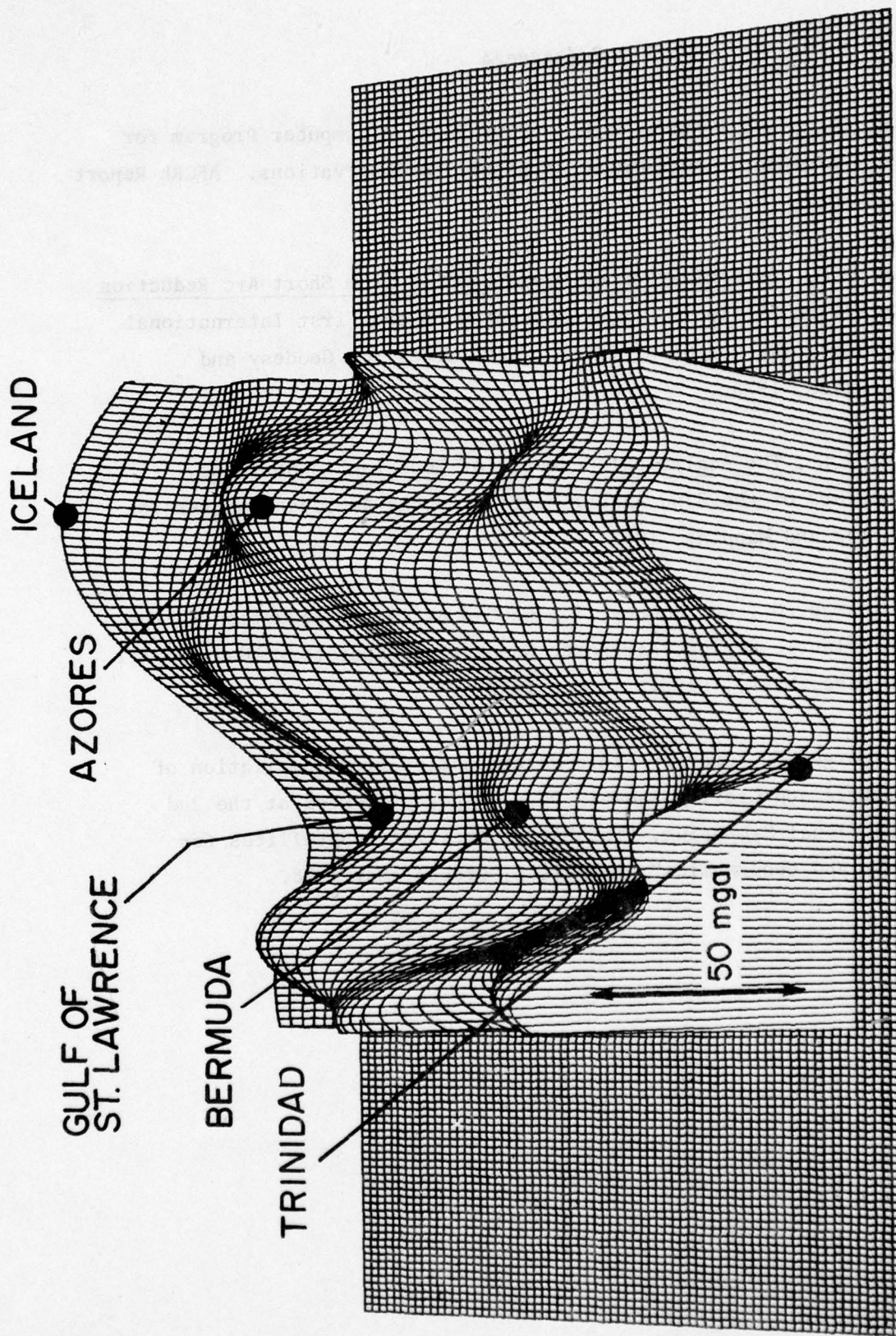


Figure 4. North Atlantic Gravity Anomalies from GEOS-3 Altimetry

References

1. Brown, D.C., and Trotter, J.E., 1969, SAGA, A Computer Program for Short Arc Geodetic Adjustment of Satellite Observations. AFCRL Report No. 69-0080
2. Brown, P.C., Determination of Oceanic Geoid from Short Arc Reduction of Satellite Altimetry, Paper presented to the First International Symposium: The Use of Artificial Satellites for Geodesy and Geodynamics, 1973.
3. Hardy, R.L., The Analytical Geometry of Topographic Surfaces, Proceedings of the 32nd Annual Meeting of the American Congress on Surveying and Mapping, 1972.
4. Hadgigeorge, G., and Rooney, T., GEOS-3 Altimetry Reduction in the North Atlantic Region, Air Force Geophysics Laboratory, Hanscom Air Force Base, Mass., 01731, 12 Oct 1977.
5. Hadgigeorge, G., Gravity Field Determination from Compilation of Altimetric and Gravity Anomaly Data, Paper presented at the 2nd International Symposium, "The Use of Artificial Satellites for Geodesy and Geodynamics", Lagonissi, Greece, May 1978.

MATRIX ELEMENTS FOR BOUND/FREE THREE BODY STATES
WITH COULOMB INTERACTION

Initiator: Dr. J. Jasperse
Problem No: 4932
Project No: 7663

The L integrals, or matrix elements, of bound/free three particle states are defined by Jasperse⁽¹⁾ in his study of three body problems. Computation of the required volume integrals for the case of a Coulomb interaction by quadrature has, for the general case, proven unfeasible due to the amount of time required to achieve a useful degree of precision. The work described here utilizes a method of contour integration developed by L. Calabi⁽²⁾, combined with a set of algorithms which allow the computer to handle explicit high-order derivative formulas. In this way the inherent limitations of quadrature approximation are avoided; leaving, however, truncation problems which may hopefully be adequately met by exploiting the longer wordlength capabilities of other computer systems.

The contour integration procedure begins with a breakdown of the integral for a matrix element into a sum of more elementary integrals, called T-integrals, each of which is evaluated by contour integration. One of the coordinates of the T-integrals, the z coordinate, is easily integrated out. The remaining two integrations lead to "residues of residues", or double residues, expressed as derivative formulas. The computations described here allowed the computer to use explicit functional forms of the derivatives, carrying all quantities to the precision allowed by the word-length of the computer. Nevertheless, although there are no numerical approximations involved, it was not obvious that some truncation would not occur, therefore, checks of both the individual residues (evaluations of T-integrals) and their sum (the L-integrals) were devised. Individual residues of all orders were also computed by a symbolic manipulation program, also utilizing full word-length precision, and agreed with those computed by the above algorithms to at least 12 places. However, the range of magnitudes of the residues was so large that for energy states higher than 2 there was virtually no guaranteed precision on the CDC-6600 computer. Since the computations

involve complex numbers, only single precision is available on the CDC-6600. It has been considered to run the program on an IBM machine with a FORTRAN-H extended compiler, which is capable of providing quadruple wordlength precision with complex numbers, or approximately double the wordlength of the CDC-6600.

All matrix elements for the first and second energy levels have been computed and verified to the precision of the quadrature method used for comparison, and the computer itself. This would seem to constitute a verification of the residual approach to the problem, and also the analytical details involved in generating the computations to be programmed.

In his study of applying contour integration methods to the volume integrals developed by Jasperse in connection with the three body problem, Calabi's aim was to devise a way of computing these integrals with greater precision, or at least in less time, than was possible with numerical methods. The result of Calabi's study was a number of derivative formulas which require either special algorithms to evaluate, or a basically new programming approach using symbolic manipulation methods. The present effort has led to a package designed around a set of FORTRAN programmable algorithms for handling the derivative formulas in a non-numerical way, thus avoiding the limitations of numerical approximation.

In addition the use of symbolic manipulation packages was explored as a possible alternative to one based upon FORTRAN algorithms and as a means of checking the precision of the FORTRAN computations. A symbolic manipulation package, called SYMBAL, designed for the CDC-6600 was eventually used for checking purposes, but was too primitive in the form used to be integrated into a useful package for handling the L-integrals. Another symbolic manipulation program designed by a member of the Laboratory was also useful for checking purposes.

This report is divided into two main sections. The first section presents a summary of Calabi's analysis, which expanded the L-integrals into functions which could be readily evaluated by contour integration. The second section includes a presentation of the algorithms used in the computations, and the manner in which the residues indicated in Calabi's report were adapted to the algorithms at hand.

Reduction of L-Integrals to T-Integrals

Calabi has derived in detail the expansion of the L-integral into a sum of T-integrals, defining the T-integrals and the coefficients of the expansion, called E coefficients. The discussion in this report will be confined mainly to the results of that derivation, omitting the development except where it could help to clarify the development of the software package. In several cases it was necessary to re-work the derivative formulas to make them compatible with the capabilities of the derivative algorithms.

The L-integral is defined as:

$$L = \int_{-1}^1 dz \int_{-\infty}^{\infty} dy \int_{-\infty}^{\infty} dz F(x,y,z) ,$$

where $F(x,y,z) = \frac{1}{4} x^2 y^2 [x^2 + 2exyz + e^2 y^2 + \alpha^2]$

$$\begin{aligned} & \cdot \int_{n''''\ell}^{\gamma} (y) \int_{n''\ell}^{\delta} (x) \int_{n'\ell}^{\alpha} \left([x^2 + 2exyz + e^2 y^2]^{1/2} \right) \\ & \cdot \int_{n\ell}^{\beta} \left([y^2 + 2fxyz + f^2 x^2]^{1/2} \right) \\ & \cdot P_{\ell'} \left(\frac{xz + ey}{[x^2 + 2exyz + e^2 y^2]^{1/2}} \right) \cdot P_{\ell} \left(\frac{fx + yz}{[y^2 + 2fxyz + f^2 x^2]^{1/2}} \right) \end{aligned}$$

Here the S's are Sturmian functions for the Coulomb potential and the P's are Legendre polynomials. The constants are all real and verify

$$\begin{aligned} 0 & \leq \ell < \min(n, n'') , \\ 0 & \leq \ell' < \min(n', n''''') , \\ ef & < 1 , \end{aligned}$$

$\ell, \ell', n, n', n'', n'''''$ all integers ,

Following Calabi's notation:

$$\int_{n\ell}^{\alpha} (x) = N_{n\ell}^{\alpha} Q_{n\ell}^{\alpha}(x) ,$$

where

$$N_{n\ell}^\alpha = Z^{2(\rho+1)} \left[\frac{\alpha(n-\ell-1)!}{\pi(n+\ell)!} \right]^{1/2} n\ell! \alpha^{\ell+2}$$

$$Q_{n\ell}^\alpha(x) = \frac{x^\ell}{(x^2+\alpha^2)^{n+1}} \sum_{p=0}^{n-\ell-1} C(n,\ell,\alpha,p) x^{2p}$$

$$C(n,\ell,\alpha,p) = \alpha^{2(n-\ell-1-p)} \sum_{p_1=0}^{n-\ell-1} C_{n\ell}^{p_1} \sum_{p_2=0}^{p_1} (-1)^{p_1-p_2} \binom{p_1}{p_2} \binom{n-\ell-1-p_1}{p-p_1}$$

and $C_{n\ell}^P$ is the coefficient of x^P in the ultraspherical polynomial $C_{n-\ell-1}^{(\ell+1)}$ [see e.g., M. Abramowitz, I. A. Stegun, Handbook of Mathematical Functions, Nat. B. Standards, Appl. Math Series 55, 1964].

In addition, a standard Legendre polynomial form is adapted to the arguments in the present case as:

$$P_\ell\left(\frac{u}{v}\right) = v^{-\ell} \sum_{p=0}^{\ell} C(\ell,p) u^p v^{\ell-p}$$

It is proved, without the condition $ef < 1$, that the L-integral exists for the domain of integration indicated. It is apparent from the definition of the Sturmian functions that $F(x,y,z)$ is a product of rational fractions of polynomials. When these functions are substituted, and the numerator expanded into powers of products of x , y and z the L-integral assumes the form:

$$L = N \int_{-\infty}^{\infty} dx \int_{-\infty}^{\infty} dy \int_{-1}^1 dz \sum_k \sum_r \sum_s E(k,r,s) T(x,y,z,k,r,s)$$

where

$$T(x,y,z,k,r,s) = \frac{x^s y^r z^k}{(x^2+\delta^2)^a (y^2+\delta^2)^b (x^2+2exyz+e^2y^2+\alpha^2)^c y^2+2fxyz+fx^2+\beta^2)^d}$$

Also: $a = n'''+1$, $b = n''+1$, $c = n'$, $d = n+1$

$$N = \frac{1}{4} N_n^\gamma N_n^\delta N_n^\alpha N_n^\beta$$

The function T may be integrated over z in closed form to give a function of x and y which is separated as follows:

$$T_2(x,y) = f(x,y) + \sum_{\lambda} f_{\lambda}(x,y) + g(x,y)$$

It will be useful to present the above functions in an abbreviated form to show certain essential features which determined the development of the software package. Departing somewhat from Calabi's notation:

$$f(x,y) = \frac{H_0(x,y) [\text{Ln}(y+A) + \text{Ln}(y-\bar{B}) - \text{Ln}(y-\bar{A}) - \text{Ln}(y+B)]}{(x^2+\delta^2)^a (y^2+\gamma^2)^b (y^2-\phi^2)^m}$$

$$g(x,y) = \frac{H_0(x,y) [\text{Ln}(y+\bar{A}) + \text{Ln}(y-B) - \text{Ln}(y-A) - \text{Ln}(y+B)]}{(x^2+\delta^2)^a (y^2+\gamma^2)^b (y^2-\phi^2)^m}$$

$$f_{\lambda}(x,y) = \frac{(-1)^{\lambda} H_{\mu}(x,y)}{(x^2+\delta^2)^a (y^2+\gamma^2)^b (y^2-\phi^2)^{m-r_{\mu}}} \frac{1}{[(y-(-1)^{\lambda}U_{\mu})(y-(-1)^{\lambda}\bar{U}_{\mu})]^{r_{\mu}}}$$

The functions $A(x) = x/e + i\alpha/e$ and $B(x) = fx + i\beta$ are complex with the bar indicating complex conjugation. The function $U_{\mu}(x)$ is either $A(x)$ or $B(x)$ depending on the index μ . The indices λ and μ are related to r_{μ} which runs from 0 to d, with $m=c+d-1$. The quantities c and d are exponents introduced above. The ϕ appearing in the denominator factor in all three functions is itself a function of x:

$$\phi(x) = \sqrt{\frac{f}{e} x^2 + \frac{f\alpha^2 - e\beta^2}{eg}}$$

This T_2 integral is understood to be evaluated in the limit as $xy(y^2-\phi^2) \rightarrow 0$ if necessary to guarantee existence at all points x and y. Finally $H_0(x,y)$ and $H_{\mu}(x,y)$ are polynomials in x and y, the orders of which depend on the exponents k, r and s.

Denoting the Cauchy principal value by CPV, it is proved that:

$$\text{CPV} \int_{-\infty}^{\infty} dx \int_{-\infty}^{\infty} dy T_2(x,y)$$

exists if $f\alpha^2 > e\beta^2$, and may be evaluated as the sum of the residues of f , g and $f\lambda$ as long as the condition $\max(n', n+1) \leq \ell + \ell' + 5$ is satisfied. The latter condition is the only constraint on the basic parameters of the problem. The relationship $f\alpha^2 < e\beta^2$ may be accommodated by changing the order of integration or interchanging variables.

The above integral is carried out as a succession of two contour integrations. First the quantity

$$T_3(x) = \text{CPV} \int_{-\infty}^{\infty} T_2(x, y) dy$$

is computed as:

$$T_3(x) = 2\pi i \sum_i \text{Res} [T_2(x, y_i)] .$$

Since $T_2(x, y)$ is broken into several components, with contours chosen independently for the several components, the y_i are whatever poles are necessary to include in all of the contours. Likewise,

$$\begin{aligned} T_4 &= \text{CPV} \int_{-\infty}^{\infty} T_3(x) dx \\ &= 2\pi i \sum_j \text{Res} [T_3(x_j)] , \end{aligned}$$

or

$$T_4 = -4\pi^2 \sum_j \sum_i \text{Res} [\text{Res} [T_2(x_j, y_i)]] ,$$

leading to a double residue computation.

The double residue in the expression for T_4 is equivalent to a double derivative in x and y , of each of the component functions of $T_2(x, y)$. The order of the derivative in each case is determined by the order of the pole in each variable. It is immediately apparent from the functions f , g and $f\lambda$, and from the potentially great number of T -integrals in an L -integral expansion, that computer evaluation of the residue derivatives is necessary. It also becomes clear from the order of the derivatives and from the precision required that any type of numerical differentiation would be inadequate.

Development of Residue Formulas

Calabi develops formulas to evaluate the required derivatives as programmable algebraic expressions which avoid the precision limitations of numerical differentiation. These expressions are, however, extremely cumbersome and were replaced by derivative algorithms which are basically a much more compact way of expressing the above algebraic evaluation. In most cases using the algorithms, it is a relatively simple matter to code an individual residue computation, although in some cases, to be seen below, considerable expansion of the derivatives is necessary before the algorithms can be applied.

Calabi's algebraic expressions also permit an analysis of the existence of individual T-integrals. While the existence of the Cauchy principal value of the L-integral is proved without restriction, the existence of all T-integrals without condition is not guaranteed. This is due to the fact that the expansion of the L-integrals can introduce special features into the component T-integrals. In particular, Calabi shows that the contour integral leading to one of the residues exists only under the condition noted above that $\max(n', n+1) \leq l + l' + 5$. (See pp. 32-33). Whether this condition could be removed or relaxed by a suitable manipulation of the $T_2(x,y)$ functions may be a matter for further study.

Although the individual poles in the $T_2(x,y)$ function are not difficult to discover and isolate, being the roots of explicit linear and quadratic forms, their distribution presents some difficulty in view of the branch points of the log functions. Calabi develops an integration theorem for the complex plane which gives a procedure for grouping the components of the $T_2(x,y)$ function into different regions of the complex plane in accordance with their poles and branch points.

Looking first at the y integration, the resolution of T_2 into f, g, and $f\lambda$ provides that, for x real, the contour integrals of f and $f\lambda$ may be taken in the positive imaginary part of the $y+iv$ plane, with the branch points restricted to the negative half of the plane (including $v=0$). Similarly, g may be integrated over the negative $y+iv$ plane, with branch points restricted to the positive half of the plane.

It may be observed, for example, that in

$$f(x,y) = \frac{H_0(x,y) [\ln(y+A) + \ln(y-\bar{B}) - \ln(y-\bar{A}) - \ln(y+B)]}{(x^2+\delta^2)^a (y^2+\gamma^2)^b (y^2-\phi^2)^m},$$

the log functions have branch points only in the negative imaginary part of the $y+i\gamma$ plane, for real x , since the imaginary parts of A and B are positive. Further, since by the integration theorem it is only necessary to include poles in the $v>0$ part of the plane, excluding the real axis, the only pole in the y plane that must be included is that at $y=i\gamma$, of order b . In this way:

$$\text{CPV} \int_{-\infty}^{\infty} f(x,y) dy = 2\pi i \text{Res} [f(x,i\gamma)] ,$$

where

$$\text{Res} [f(x,i\gamma)] = \frac{1}{(b-1)!} \frac{\partial^{b-1}(y+\gamma)^b}{\partial y^{b-1}} f(x,y) \Big|_{y=i\gamma} .$$

If the same integration theorem is invoked for a contour integration in the $x+iw$ plane, it becomes apparent on inspection of the arguments of the log terms that two of the log terms have branch points in the upper half of the $x+iw$ plane [i.e., $i\gamma-\bar{B} = i\gamma - fx + i\beta$, and $i\gamma - \bar{A} = i\gamma - \frac{x}{e} + i\frac{\alpha}{e}$], and the other two have branch points in the lower half of the $x+iw$ plane. If the function f is further broken down to isolate these branch points, with $f_1(x,y)$ and $f_2(x,y)$ having branch points in the lower half and upper half, respectively of the $x+iw$ plane, then the above residue computation becomes:

$$\text{CPV} \int_{-\infty}^{\infty} f(x,y) dy = 2\pi i \text{Res}[f_1(x,i\gamma)] + \text{Res} [f_2(x,i\gamma)] .$$

Avoiding those parts of the $x+iw$ plane with branch points it is apparent that $\text{Res}[f_1(x,i\gamma)]$ must be integrated in the upper half of the complex plane and $\text{Res}[f_2(x,y)]$ in the lower half. It is also clear that both $\text{Res}[f_1]$ and $\text{Res}[f_2]$ have poles at $x = \pm i\delta$ of order a . From the above definition of $\phi(x)$ it is also clear that upon substituting $y=i\gamma$ after differentiation, the denominator factor $(y^2-\phi^2)^{+m}$ produces the quantity $(-\gamma^2-\phi^2)^{m+b-2}$ in the

denominator. Then by factoring: $\phi^2(x) + \gamma^2 = \frac{f}{e}(x+iv)(x-iv)$, where v is a positive constant. Thus both f_1 and f_2 also have poles in the x plane at $x = \pm iv$. Thus the contour integrals of $\text{Res}[f_1(x, i\gamma)]$ and $\text{Res}[f_2(x, i\gamma)]$ must include residues at $x = i\delta, iv$ and $x = -i\delta, -iv$ respectively. Thus:

$$\text{CPV} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \, dx dy = -4\pi^2 \left\{ \begin{aligned} &\text{Res}[\text{Res}[f_1(i\delta, i\gamma)]] \\ &+ \text{Res}[\text{Res}[f_1(iv, i\gamma)]] \\ &+ \text{Res}[\text{Res}[f_2(-i\delta, i\gamma)]] \\ &+ \text{Res}[\text{Res}[f_2(-iv, i\gamma)]] \end{aligned} \right\} .$$

Here

$$\text{Res}[\text{Res}[f_1(i\delta, i\gamma)]] = \frac{1}{(b-1)!(a-1)!} \left. \frac{\partial^{a-1}}{\partial x^{a-1}} \frac{\partial^{b-1}}{\partial y^{b-1}} (x-i\delta)^a (y-i\gamma)^b f_1(x, y) \right|_{\substack{x=i\delta \\ y=i\gamma}}$$

and

$$\text{Res}[\text{Res}[f_1(iv, i\gamma)]] = \frac{1}{(b-1)!(m+b-2)!} \frac{\partial^{m+b-2}}{\partial x^{m+b-2}} (x-iv)^{m+b-1} \left. \frac{\partial^b}{\partial y^b} (y-i\gamma) f_1(x, y) \right|_{\substack{y=i\delta \\ x=iv}}$$

In the expression for $\text{Res}[\text{Res}[f_1(i\delta, i\gamma)]]$, where the poles in both the y plane and the x plane may be factored out of the denominator before any derivatives are taken, the algorithms make it possible to code the double derivative by simply calling general subroutines as described below. In the expression for $\text{Res}[\text{Res}[f_1(iv, i\gamma)]]$ the x pole may not be factored out of the denominator until the y derivative has been taken, so that some preliminary manipulation must take place before applying the double derivative algorithm. This is accomplished by writing

$$f_1(x, y) = \frac{P(x, y)}{(y^2 + \gamma^2)^b (y^2 - \phi^2)^m} ,$$

and by expressing the derivative and extracting the factors of $(y^2 - \phi^2)^{-m}$ in a general form before the derivative algorithms are applied.

Differential Algorithm Formulations for the Residue Evaluations

The values for the residues necessary to determine the atomic Coulomb integrals are determined by incorporating algorithms for evaluating derivatives of functions. This procedure is equivalent to a straight forward computer evaluation of mathematical formulations, thereby, having accuracy proportional to the computer word length. A procedure of this form is a necessity for this problem, in that, approximation by finite difference methods is too inaccurate. The two basic algorithms used are the following:

Let the derivatives with respect to x,y exist for functions $F(x,y)$ and $G(x,y)$, then

Derivative of a product of functions

$$D_{x,y}^{I,J} (F(x,y)G(x,y)) = \sum_{i=0}^I \sum_{j=0}^J \binom{I}{i} \binom{J}{j} D_{x,y}^{i,j} F(x,y) D_{x,y}^{I-i,J-j} G(x,y) \quad (A1)$$

Derivative of a reciprocal of a function

$$D_{x,y}^{I,J} (1/F(x,y)) = \sum_{i=0}^{I-1} \sum_{j=0}^{J-1} \binom{I}{i} \binom{J}{j} D_{x,y}^{i,j} (1/F(x,y)) \\ D_{x,y}^{I-i,J-j} (F(x,y))/F(x,y) \quad (A2)$$

In regards to the computational needs for this problem, the above algorithms are the fundamentals from which other needed algorithms were constructed. For example, the later algorithm is needed for constructing a formulation for evaluating derivatives of the Natural Logarithm of a function $F(x,y)$, that is,

$$D_{x,y}^{I,J} (\ln(F(x,y))) = \sum_{i=0}^I \sum_{j=0}^{J-1} \binom{I}{i} \binom{J}{j} D_{x,y}^{i,j} \frac{1}{F(x,y)} D_{x,y}^{I-i,J-j} F(x,y) \quad (A3)$$

Note: $J > 0$; for $J = 0$ $D_{x,y}^{0,0} = \ln(F(x,y))$ and

$$D_{x,y}^{I,0} (\ln(F(x,y))) = \sum_{i=0}^{I-1} \binom{I-1}{i} D_{x,y}^{i,0} \frac{1}{F(x,y)} D_{x,y}^{I-i,0} F(x,y)$$

Another formulation required, of a fundamental nature for this problem, is to evaluate

$$D_{x,g}^{I,J} p(x,y) \Big|_{\substack{y=\phi(x) \\ x=x_0}} \quad (A4)$$

where $\phi(x) = (Ax^2 + B)^{1/2}$

and $p(x,y) = p_0 + p_1 y + p_2 y^2 + p_3 xy + p_4 x + p_5 x^2$

a quadratic polynomial in x,y . It was decided to treat this as follows:

Take the y differentials and substitute $y=\phi(x)$, thereby requiring the evaluation of

$$D_{x,y}^{I,0} p(x,y) \Big|_{\substack{y=\phi(x) \\ x=x_0}} \rightarrow D_X^I p(x,\phi(x)) \Big|_{x=x_0}$$

$$D_{x,y}^{I,1} p(x,y) \Big|_{\substack{y=\phi(x) \\ x=x_0}} \rightarrow D_X^I (p_1 + 2p_2 \phi(x)) \Big|_{x=x_0}$$

$$D_{x,y}^{I,2} p(x,y) \Big|_{\substack{y=\phi(x) \\ x=x_0}} \rightarrow D_X^I (2p_2) \Big|_{x=x_0}$$

$$D_{x,y}^{I,J} p(x,y) \Big|_{\substack{y=\phi(x) \\ x=x_0}} = 0 \text{ for } J > 2$$

Then algebraic differentials of the irrational function $\phi(x)$ were obtained manually, for example:

$$D_X^0 \phi(x) \rightarrow (Ax^2 + B)^{1/2}$$

$$D_X^1 \phi(x) \rightarrow Ax(Ax^2 + B)^{-1/2}$$

$$D_X^2 \phi(x) \rightarrow A(Ax^2 + B)^{-1/2} - A^2 x^2 (Ax^2 + B)^{-3/2}$$

etc.

and thus the required formulation was accomplished.

The first (2) of each of these formulations is utilized in each residue calculation, whereas, the later (2) is required only for certain of these residues. What follows is in general, a differential formulation for specified residue evaluations.

Residue (x=iδ,y=iγ) (including fλ)

These are the simplest of all the required residue calculations. They are obtained by a direct application of equations A1, A2, and A3.

Residue (x=iv,y=iγ) (not including fλ)

These and subsequent residues are of a nature that upon performing the y differential corresponding to the order of that pole, the order of the x pole is thereby changed. Hence, in addition to utilizing equations A1,A2, and A3 the following formulation must be implemented into the calculation and upon taking the y differentiation the resulting equations to be programmed are:

$$n = 1 , \frac{\partial^m}{\partial x^m} \frac{1}{(x+iv)^{m+1}} [(x+iv) (x-iv) P_y^{-m(2y)} P] \Bigg|_{\substack{y=i\gamma \\ x=iv}}$$

$$n = 2 , \frac{\partial^m}{\partial x^m} \frac{1}{(x+iv)^{m+2}} [(x+iv)^2 (x-iv)^2 p_{yy} - 2m(x+iv) (x-iv) (2y P_y + P) + (m+1) (m) (2y)^2 P] \Bigg|_{\substack{y=i\gamma \\ x=iv}}$$

where P is the functional form of the "reduced integrand" with the subscripting P_y , etc., representing differentiation with respect to y. Currently, this is the formulation used in the program; however, a general form for n is available and has been separately programmed, which is:

$$\text{Residue } (i\gamma, iv) = \frac{1}{(b-1)!} \sum_{i=0}^{b-1} \sum_{\gamma} \sum_{q_1} \sum_{q_2}^r$$

$$\binom{b-1}{i} \binom{i}{\gamma} \binom{i-\gamma}{q_1} \binom{\gamma}{q_2} \frac{(m+\gamma-1)!(m+i-\gamma-1)!}{(m-1)!^2} (-1)^{i+q_2}$$

$$\left. \begin{aligned} & \frac{(-i\gamma)^{i-q_1-q_2}}{(m+i-1)!} D_x^{m+i-1} D_y^{b-1-i} \frac{\phi^{q_1+q_2} P(x,y)}{(-f/e(x+ispv))^{m+1}} \end{aligned} \right\} \begin{array}{l} x=ispv \\ y=i\gamma \end{array}$$

Residue (x=iδ, y=±φ)

In brief, the computation of the φ residues requires the simultaneous consideration of g(x,y) at y = ±φ in the form

$$G(x) = \text{Res}(x, \phi) + \text{Res}(x, -\phi)$$

Care must be taken in choosing the planes of integration to avoid conflict with the branch points from the logarithm term. Upon substituting y=φ, the Ln factor has the form

$$\text{Ln } u = \text{Ln } [\alpha^2 - y^2] = \text{Ln } [E(x+i\eta_+) (x+i\eta_-)]$$

or

$$\text{Ln } [E(x+i\eta_+)] + \text{Ln } [x+i\eta_-]$$

wherein the sign of η₊, η₋ determines the contour of integration. Expressing the integral g as the sum of integrals over G₁(η₊) and G₂(η₋) the following formulations were determined for these residues:

$$\text{Res}(G_1, i\delta \cdot sp) = \frac{1}{(m-1)!(a-1)!} D_x^{a-1} D_y^{m-1} \frac{\text{Ln } [E(x+i\eta_+)]}{(x+i\delta \cdot sp)^a}$$

$$P(x,y) \quad \begin{array}{l} y = \phi \\ x = i\delta \cdot sp \end{array}$$

$$\text{Res}(G_2, i\delta \cdot sm) = \frac{1}{(m-1)!(a-1)!} D_x^{a-1} D_y^{m-1} \frac{\text{Ln } [u]}{(x+i\delta \cdot sm)^a}$$

$$P(x,y) \quad \begin{array}{l} y = \phi \\ x = i\delta \cdot sm \end{array}$$

where u is loaded at D_y^0 by $[x+i\eta_-]$ and $sp = \text{sign}(\eta_+)$, $sm = \text{sign}(\eta_-)$, and $E = \text{constant}$. Hence, these residues were evaluated by programming this formulation in conjunction with equations A1, A2, A3, and A4.

Below, just the Residue Formulations are formally outlined with partial mention of referencing to equations A1, A2, A3, A4.

Residue $(x=iv, y=\pm\phi)$

The mathematical formulation derived for this residue is

$$\begin{aligned} \text{Res}(G_1, iv \text{ sp}) = & \frac{1}{(m-1)!(m+b-2)!} \sum_{k=0}^{m-1} \binom{m-1}{k} \binom{m+b-2}{m-k-1} \\ & \cdot \left(\frac{f}{e}\right)^{-b-k} (m-k-1)! D_x^{b+k-1} (x+iv \cdot sp)^{-b-k} \\ & \cdot \text{Ln} [E(x+i\eta_+)] Dy^{m-1-k} P(x,y) \\ & \cdot \left. \sum_{j=0}^k W(b,j,k) (\phi+i\gamma)^{k-j} (\phi-i\gamma)^j \right|_{\substack{y = \phi \\ x = iv \cdot sp}} \end{aligned}$$

$$W(b,j,k) = (-1)^k \binom{k}{j} \frac{(b+j-1)!(B+k-j-1)!}{[(b-1)!]^2} .$$

Residue $(x=-i\eta_+, y=\pm\phi)$

This residue, precipitating from the differentiation of the logarithmic term, is as follows:

$$\begin{aligned} \text{Res}(G_2, -i\eta_+) = & \frac{1}{(m-2)!(m-1)!} \sum_{\ell=1}^{m-1} \sum_{j=0}^{\ell} \binom{m-1}{\ell} \binom{\ell}{j} \\ & \binom{m-2}{m-1-\ell} (m-1-\ell)! (\ell-1)! (-1)^{\ell-1} (1+(-1)^{\ell-j}) E^{-\ell} \\ & D_x^{\ell-1} (x+i\eta_-)^{-\ell} \phi^j \alpha^{\ell-j} D_y^{m-\ell-1} H(x,y) \Big|_{\substack{y = \phi \\ x = -i\eta_+}} . \end{aligned}$$

Residue (x=in, y=phi=0)

This residue, upon considering an x pole corresponding to phi=0, has the following formulation:

$$\text{Res}(\phi=0) = \text{const} \left\{ \begin{aligned} & (sp, sm) \sum_{\ell=0}^{m-1} \sum_{j=0}^{\min(\ell, r-k-1)} \binom{m-1}{\ell} \binom{\ell}{j} \frac{(-1)^{\ell-j} (r-k-1)!}{(r-k-1-j)! (m-1)!} \\ & \frac{2^{-m-\ell+j} D_x^{IX-1}}{(IX-1)!} \frac{\text{Ln}(E(X+i\eta+))}{\left(\frac{p}{e}(x+iwsp)\right)^{IX}} \\ & \phi^{r-k-1-m-\ell} D_y^{m-1-\ell} p(x, y) \Big|_{\substack{y=\phi \\ x=iw, (\eta+, \eta-)}} \\ & + (sm) \sum_{\ell=1}^{m-1} \sum_{\eta=0}^{m-1-\ell} \sum_{i=0}^{\min(\eta, r-k-1)} \sum_{\epsilon=0}^{\ell} \binom{m-1}{\ell} \binom{m-1-\ell}{\eta} \binom{\eta}{i} \binom{\ell}{\epsilon} \\ & (-1) \frac{(r-k-1)! (m+n-i-1)! 2^{-m-n+1}}{(r-k-1-i)! (m-1)!} ((-1)^{\ell+\epsilon-1} - 1) \\ & (\ell-1)! \frac{D_x^{IX-1}}{(IX-1)!} \frac{\alpha_L^{\ell-\epsilon}}{(\alpha_L^2 - y^2)} \frac{\phi^{r-k-1-m-\eta+\epsilon}}{\left(\frac{p}{e}(x+iwsm)\right)^{IX}} D_y^{m-1-\ell-\eta} p_{q, y} \Big|_{\substack{y=\phi \\ x=iw(\eta)}} \end{aligned} \right.$$

$$D_y^t(Px, y) = \frac{X^{s-k-1}}{(x^2 + \delta^2)^a} \sum_{j=0}^k \sum_{s=j}^{\min(t, 2j)} \sum_{n=0}^{t-s} \sum_{y=0}^n \sum_{q_2=0}^v \sum_{r=0}^{\min(2(k-j), t-s-\mu)}$$

$$\sum_{p_2=0}^r \sum_{\substack{q_1=0, 2, 4, 6 \\ \text{or } 1, 3, 5, 7}}^{n-v} \sum_{\substack{p_1=0, 2, 4, 6 \\ \text{or } 1, 3, 5, 7}}^{t-s-n-r} \frac{(eg)^j (e^2)^{k-j}}{F^j} \binom{k}{j} \binom{m-1-j}{\phi-1}$$

$$\binom{t}{s} \binom{t-s}{u} \binom{s}{s-j} \binom{n}{v} \binom{u-v}{q_1} \binom{v}{q_2} \binom{v}{q_2} \binom{t-s-n}{r} \binom{t-s-n-r}{p_1}$$

$$\binom{r}{p_2} \frac{(j!)^2}{(2j-s)!} \frac{(b+v-1)!(b+n-v-1)!(k-j)!^2}{(b-1)!^2 (k-j-r) : (k-j-t+s+n+r)!}$$

$$2^{2j-s} (-1)^{(q_2+p_2 + (q_2+q_1+p_2+p_1)/2)}$$

$$\left(\gamma^2\right)^q \left(\eta^2\right)^p \left(\phi^2+\gamma^2\right)^{-b-n} \left(\phi^2+\eta^2\right)^{k-j-t+s+n}$$

$$\phi^{t+2j-2s-2p-2q}$$

$$w = \sqrt{\frac{eB^2}{Fg} - \frac{\alpha^2}{g}} \quad q_1 + q_2 = 2q$$

$$\eta^2 = (x^2 + \alpha^2)/e^2 \quad p_1 + p_2 = 2p$$

It should be noted that the residues described above are the so-called f and g residues or non- λ .

The following section describes the λ residues and it should be noted that of the four basic formulations described above only A1 and A2 are required.

The λ residues of form $(x=i\delta, y=i\lambda)$ and $(x=iv, y=i\gamma)$ will not be discussed since the differential formulations are similar to those of the same notation described earlier. The main difference being only in the "reduced integrand", i.e., what is to be differentiated.

Residue $(x-x_0(u, \bar{u}\lambda), y=i\gamma)$

The formulation for this residue being

case b=2

$$\text{Residue} = \frac{1}{(r_n)!} D_x^{r_n} ((x-x_0) P_y - r_n P) \Big|_{\substack{y=i\gamma \\ x=x_0}}$$

case b=3

$$\text{Residue} = \frac{1}{2(r_n+1)!} D_x^{r_n+1} ((x-x_0)^2 P_{yy} = 2r_n(x-x_0)P_y + r_n(r_n+1)P) \Big|_{\substack{y=i\gamma \\ x=x_0}}$$

$$\underline{\text{Residue } (x-x_0(y, \gamma, \lambda), y=u, \bar{u})}$$

Basically, these residues are evaluated as follows: for n=2, r_n=1,2.

case r_n=2

$$\text{Residue} = \frac{1}{b!} D_x^b (x-x_0) P_y - P \Big|_{\substack{y=u, \bar{u} \\ x=x_0}}$$

$$\underline{\text{Residue } (x=x_0(y, \phi, \lambda), y=u, \bar{u})}$$

case r_n=1:

$$\text{Residue} = \frac{1}{(m-2)!} D_x^{m-2} P \Big|_{\substack{y=n, \bar{u} \\ x=x_0}} ;$$

case r_n=2:

$$\text{Residue} = \frac{1}{(m-2)!} D_x^{m-2} ((y^2 - \phi^2) P - (m-2)(2y)P) \Big|_{\substack{y=n, \bar{u} \\ x-x_0}}$$

As can be expected in a problem of this magnitude, a number of analytical requirements were incurred. They included:

Time Estimate - In the early stages of this problem, an effort was made to forecast the computer time requirements. Basically, this was done by comparing the time for a calculation of a lower order integral and that of a higher order integral. This data was then correlated to the calculation procedure, namely, the number of residue calculations required and the order of each differentiation. From this some time estimating has been done but is not assumed accurate.

Verification of Residue Calculation - In addition to the bulk of checking mentioned in the introductory section, it should be noted that some lower order residue calculations were checked by hand and by numerical approximation,

Error Estimate - Also of primary concern is the number of digits of accuracy in an internal calculation. As noted, the residue evaluations (for $n=2$), are not the problem in themselves but, rather in the summing of them to form the value for the integral. In essence, the loss of accuracy is caused by combining terms of equivalent magnitude but opposite in sign. This latter case can apparently be determined by comparing the separate sum for both the positive or negative residues. A more general form is naturally more desirable. Presently, being considered is to calculate the same integral using different word lengths, which when compared then exhibits the truncation error.

Results - Table I contains a family of Coloumb integrals of the form $L_{n,n',n'',n''',l,l'}$ for two earlier investigations and the current effort for data $\alpha = 2.31$, $\beta = 1.14$, $r = \delta = 2.54555$, $E = .999864$, $F = .05$ where, in brief, Jasperse set $l = l' = 0$ and then parallels Calabi; whereas, Grossbard utilized a straightforward 3-dimensional numerical integration.

This effort for higher orders is presently being continued as a part of another Air Force project.

TABLE I

	<u>Grossbard</u>	<u>Jasperse</u>	<u>Calabi</u>
L ₁₁₁₁₀₀	3.86290148436	3.86265601596	3.862656015929
L ₁₁₂₁₀₀	-.493275047545	-.493340150938	-.4933401497509
L ₁₁₁₂₀₀	-3.9007	-3.90067764136	-3.900677740736
L ₁₁₂₂₀₀	3.3943	3.39404022841	3.3940402366
L ₁₂₁₁₀₀	-2.16382206187	-2.16297156832	-2.162971562578
L ₁₂₁₂₀₀	3.9085	3.90817439123	3.908174487
L ₁₂₂₁₀₀	4.0236	4.02337484402	4.023373012508
L ₁₂₂₂₀₀	-6.1341	-6.12737514331	-6.1273690616
L ₂₁₁₁₀₀	.0128271740761	.0128830519028	.01288304718
L ₂₁₁₂₀₀	2.6816	2.68125017719	2.68125013708
L ₂₁₂₁₀₀	.928635340444	.928580637072	.92858049261
L ₂₁₂₂₀₀	-2.2927	-2.29275562759	-2.29275790467
L ₂₂₁₁₀₀	3.58132361885	3.58092539223	3.5809311925
L ₂₂₁₂₀₀	-5.6336	-5.63342551987	-5.6334222233
L ₂₂₂₁₀₀	-1.4738	-1.47375365940	-1.4737602310
L ₂₂₂₀₀₀	5.6309	5.63090879091	5.630200201
L ₁₂₁₂₀₁	-.322285		-.32213486
L ₁₂₂₂₀₁	-4.06004		-4.0661064
L ₂₁₂₁₁₀	.767619		.76769742
L ₂₁₂₂₁₀	-3.58466		3.5845788
L ₂₂₁₂₀₁	4.14230		4.1416992
L ₂₂₂₁₁₀	2.8789		2.8706734
L ₂₂₂₂₀₁	-.679022		-.67904956
L ₂₂₂₂₁₀	-1.03949		-1.0390567
L ₂₂₂₂₁₁	-.0653925		-.065392053

References

1. Calabi, L., The Integration of Certain Volume Integrals Required in Three Body Calculations, AFCRL-69-0021, Scientific Report No. 1, Contract No. F19628-68-C-0061, 1968.
2. Jasperse, J., Three Families of Integrals Which Arise in Some Three Body Problems, AFCRL-67-0222, 1967.

ELECTRON AND PARTICLE DISTRIBUTION FUNCTIONS

Initiator: Dr. J. Jasperse
 Problem No: 4037 (formerly Problem No: 4916)
 Project No: 4643

Over the last three years the mathematical and numerical analyses has been developed and the computer programs written to calculate the electron and particle distribution functions in the ionosphere. This was accomplished using a non-linear difference equation approximation of an integral-differential equation which described the energy transport equation (see references 1,2,3,4,5). Computer programs have been developed which use various approximations to a larger more complete model. The complete model has been programmed and the results presented in references 4,5. The complete model is known as the BFPEDF code. This model and the numerical approximations developed by Boston College are described below.

Simulation of the BFPEDF Code

BFPEDF solves the following differential equation

$$\begin{aligned} & \sum_j \left(\frac{2m_e}{m_{nj} + m_e} \right) \gamma_{jn}(z, E) E^{3/2} + \sum_s \left(\frac{2m_e}{m_{is} + m_e} \right) \gamma_{eis}^1(z) \\ & + K_{ee} I_o(z, E) H(z, E) = + \int_E^{E_M} dE' \sum_j G_{pij}(z, E') + \int_E^{E_M} dE^1 \sum_j G_{enij}(z, E^1) \\ & - \int_E^{E_M} dE^1 \sum_s \Sigma(E^1)^{1/2} \gamma_{rs}(z, E^1), H(z, E^1) - \int_E^{E_M} dE^1 \sum_{tk} \Sigma(E^1)^{1/2} \gamma_{tk}(z, E^1) \\ & \theta(E^1 - E_{tk}) H(z, E^1) + \int_E^{E_M} dE^1 \sum_{tk} \Sigma(E^1 + E_{tk})^{1/2} \gamma_{tk}(z, E^1 + E_{tk}) H(z, E^1 + E_{tk}) \\ & - \int_E^{E_M} dE^1 \sum_{tk} \Sigma \exp \left[- \frac{E_{tk}}{(K_{Ttk}(z))} \right] (E^1 + E_{tk})^{1/2} \gamma_{tk}(z, E^1 + E_{tk}) H(z, E^1) \\ & + \int_E^{E_M} dE^1 \sum_{tk} \Sigma \exp \left[- \frac{E_{tk}}{(K_{Ttk}(z))} \right] (E^1)^{1/2} \gamma_{tk}(z, E^1) \theta(E^1 - E_{tk}) H(z, E^1 - E_{tk}) \end{aligned}$$

$$\begin{aligned}
& - \sum_j \left(\frac{2m_e}{m_{nj} + m_e} \right) \gamma_{mnj}(z, E) E^{3/2} K_{Tn}(z) \\
& + \sum_s \left(\frac{2m_e}{m_{is} + m_e} \right) \gamma_{eis}(z) K_{Ti}(z) + \frac{2 K_{ee}}{3} [I_2(z, E) + J(z, E)] \frac{\partial}{\partial E} H(z, E)
\end{aligned}$$

for $H(z, E)$ versus E for a particular z will be constructed as follows:

$$\begin{aligned}
& \text{TERM1} + \text{TERM2} + \text{TERM3} = \text{TERM4} + \text{TERM5} \\
& + \text{TERM6} + \text{TERM7} + \text{TERM8} + \text{TERM9} + \text{TERM10} \\
& + \text{TERM11} + \text{TERM12} + \text{TERM13} + \text{TERM14}
\end{aligned}$$

where the terms are numbered according to how they appear in the formula.

Thus TERM13 =

$$\frac{2 K_{ee}}{3} I_2(z, E) \frac{\partial H(z, E)}{\partial E}$$

In addition, the following boundary condition was used in the solution:

$$1 = \int_{E_1}^{E_M} dE (E^{1/2}) H(z, E) / n_e(z)$$

The method used to solve this differential equation was to guess a point distribution for E s.t. $0 = E_1 < E_2 < E_3 < \dots < E_{\text{NEVAL}} = E_M$

For $E = E_2, E_3, \dots, E_{\text{NEVAL}}$ the equation was then approximated by a non-linear difference equation with values of $H(z, E)$ defined at $E_1, E_2, E_3, \dots, E_{\text{NEVAL}}$. In addition, the boundary condition was also approximated by a non-linear difference equation.

These equations are numbered as follows:

1. boundary condition
2. Equation for E_2
3. Equation for E_3
- ⋮
- ⋮
- ⋮
- ⋮
- NEVAL Equation for E_{NEVAL}

The unknowns in these equations are $H(z, E)$ defined at $E_1, E_2, E_3, \dots, E_{NEVAL}$.

Thus we wish to solve a matrix equation which can be put in the form.

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & \dots & A_{1,NEVAL} \\ A_{2,1} & A_{2,2} & A_{2,3} & \dots & A_{2,NEVAL} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{NEVAL,1} & A_{NEVAL,2} & A_{NEVAL,3} & \dots & A_{NEVAL,NEVAL} \end{bmatrix} \begin{bmatrix} H(z, E_1) \\ H(z, E_2) \\ \vdots \\ H(z, E_{NEVAL}) \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_{NEVAL} \end{bmatrix}$$

where: $A_{i,j}$ is the value associated with $H(z, E_j)$ for equation i . $H(z, E_i)$ is the desired solution and B_i are constants. The difficulty in solving this equation is that the $A_{i,j}$'s are functions of the $H(z, E_1)$'s which makes the equation non-linear.

The method of solution for the matrix equation is as follows:

Guess a value for $H(z, E_1), H(z, E_2), \dots, H(z, E_{NEVAL})$ say $H^{old}(z, E_1), H^{old}(z, E_2), \dots, H^{old}(z, E_{NEVAL})$ then calculate

$$\begin{bmatrix} ERR_1^{old} \\ ERR_2^{old} \\ \vdots \\ ERR_{NEVAL}^{old} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,NEVAL} \\ A_{2,1} & A_{2,2} & & A_{2,NEVAL} \\ \vdots & \vdots & & \vdots \\ A_{NEVAL,1} & A_{NEVAL,2} & \dots & A_{NEVAL,NEVAL} \end{bmatrix} \begin{bmatrix} H^{old}(z, E_1) \\ H^{old}(z, E_2) \\ \vdots \\ H^{old}(z, E_{NEVAL}) \end{bmatrix} - \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_{NEVAL} \end{bmatrix}$$

where $A_{i,j}$ is calculated using the values of H^{old} . If

$$\frac{1}{NEVAL} \sqrt{\sum_{i=1}^{NEVAL} (ERR_i^{old})^2} < RMSMAX$$

where RMSMAX is a constant, then the H^{old} vector is the desired solution, otherwise, a new H vector H^{new} is calculated as follows:

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & \dots & C_{1,NEVAL} \\ C_{2,1} & C_{2,2} & C_{2,3} & \dots & C_{2,NEVAL} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{NEVAL,1} & C_{NEVAL,2} & C_{NEVAL,3} & \dots & C_{NEVAL,NEVAL} \end{bmatrix} =$$

$$\begin{bmatrix} \frac{\partial ERR_1}{\partial H^{old}(z, E_1)} & \frac{\partial ERR_1}{\partial H^{old}(z, E_2)} & \dots & \frac{\partial ERR_1}{\partial H^{old}(z, E_{NEVAL})} \\ \frac{\partial ERR_2}{\partial H^{old}(z, E_1)} & \frac{\partial ERR_2}{\partial H^{old}(z, E_2)} & \dots & \frac{\partial ERR_2}{\partial H^{old}(z, E_{NEVAL})} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial ERR_{NEVAL}}{\partial H^{old}(z, E_1)} & \frac{\partial ERR_{NEVAL}}{\partial H^{old}(z, E_2)} & \dots & \frac{\partial ERR_{NEVAL}}{\partial H^{old}(z, E_{NEVAL})} \end{bmatrix}$$

Then solve the following matrix equation

$$\begin{bmatrix} C_{1,1} & C_{1,2} & \dots & C_{1,NEVAL} \\ C_{2,1} & C_{2,2} & \dots & C_{2,NEVAL} \\ \vdots & \vdots & \ddots & \vdots \\ C_{NEVAL,1} & C_{NEVAL,2} & \dots & C_{NEVAL,NEVAL} \end{bmatrix} \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \vdots \\ \Delta_{NEVAL} \end{bmatrix} = \begin{bmatrix} ERR_1^{old} \\ ERR_2^{old} \\ \vdots \\ ERR_{NEVAL}^{old} \end{bmatrix}$$

and the new guess for the H vector is

$$\begin{bmatrix} H^{new}(z, E_1) \\ H^{new}(z, E_2) \\ \vdots \\ H^{new}(z, E_{NEVAL}) \end{bmatrix} = \begin{bmatrix} H^{old}(z, E_1) \\ H^{old}(z, E_2) \\ \vdots \\ H^{old}(z, E_{NEVAL}) \end{bmatrix} + \frac{1}{2^\ell} \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \vdots \\ \Delta_{NEVAL} \end{bmatrix}$$

where $\ell = 0$ or 1 , or $2 \dots$ is defined as the smallest integer s.t.

$$\frac{1}{NEVAL} \sqrt{\sum_{i=1}^{NEVAL} (ERR_i^{new})^2} < \frac{1}{NEVAL} \sqrt{\sum_{i=1}^{NEVAL} (ERR_i^{old})^2}$$

After H^{new} is found H^{old} is redefined as H^{new} and the method is iterated.

To start the iterative process we need an initial guess for $H(z, E_i)$. This initial guess is usually provided by guessing $T_e(z)$ the n_i 's and approximating the $H(z, E_i)$ by $H_{max}(z, E_i) + H_{CSD}(z, E_i)$, where

$$H_{max}(z, E_i) = 2\pi n_e(z) \left[\frac{1}{\pi K_{T_e}(z)} \right]^{3/2} \exp[-E_i/K_{T_e}(z)]$$

and

$$\begin{aligned} H_{CSD}(z, E_i) = & \left(\frac{1}{E_i} \right)^{3/2} \sum_{tk} \left[1 - \exp\left(-\frac{E_{tk}}{K_{T_{tk}}(z)}\right) \right] \left(\frac{E_{tk}}{E_i} \right) \gamma_{tk}(z, E_i) \\ & \theta(E_i - E_{tk}) + K_{ev} E_i^{1/2} \sum_{j\ell} \left(\frac{E_{ijk}}{E_i} \right) n_{nj}(z) \theta(E_i - E_{ij\ell}) Q_{ij\ell}(E_i) \\ & + \sum_j \left(\frac{2 m_e}{m_{nj} + m_e} \right) \gamma_{mnj}(z, E_i) + \sum_s \left(\frac{1}{E_i} \right)^{3/2} \left(\frac{2 m_e}{m_{is} + m_e} \right) \gamma_{eis}(z) \\ & + \left(\frac{1}{E_s} \right)^{3/2} \gamma_{ee}^1(z) \times \int_{E_i}^{E_m} dE^1 \sum_j G_{pij}(z, E^1) \end{aligned}$$

Where $\gamma_{tk}(z, E_i) = n_{nt}(z) \text{Kev}(E_i)^{1/2} Q_{tk}(E_i)$ and for $t=1, k=1, 7$:

a.

$$Q_{tk}(E_i) = \begin{cases} 0 & \text{for } E_1 \leq E < E_{tk} \\ \frac{q_0 A_{tk}}{E_{tk}} \left(\frac{E_{tk}}{E_i}\right)^{\phi_{tk}} \left[1 - \left(\frac{E_{tk}}{E_i}\right)^{B_{tk}}\right]^{N_{tk}} & \text{for } E_{tk} \leq E \leq E_m \end{cases}$$

with $T_{ik}(z) = T_{el}(z)$

- b. for $t=2, k=1-8$: $Q_{2k}(E)$ is read in as a table of $Q_{2k}(E)$ versus E values, where additional values of $Q_{2k}(E)$, not directly found in the table, are approximated by linear interpolation. In this case $T_{2k}(z) = T_{VIB}(z)$ for $t=2, k=9-23$: $Q_{tk}(E_i)$ is found similarly to the case $t=1, k=1, 7$. In this case $T_{2k}(z) = T_{el}(z)$.
- c. for $t=3, k=1-8$: $Q_{3k}(E)$ is read in as a table of $Q_{3k}(E)$ versus E values, where additional values of $Q_{3k}(E)$, not directly found in the table, are approximated by linear interpolation. In this case $T_{3k}(z) = T_{VIB}(z)$.
- d. for $t=3, k=9-18$: $Q_{tk}(E_i)$ is found similarly to the case $t=1, k=1, 7$. In this case $T_{3k}(z) = T_{el}$.
- e. for $t=4, k=1, 2$ and $t=5, k=1$: $Q_{tk}(E)$ is read in as a table of $Q_{tk}(E)$ versus E values where additional values of $Q_{tk}(E)$, not directly found in the table, are approximated by linear interpolation. In this case $T_{tk}(z) = T_{FS}(z)$.

f. for $t=6-35$, $k=1$:

$$Q_{t1}(E) = \begin{cases} 0 & \text{for } E_1 \leq E < E_{t1} \\ 4.7 \cdot 10^{-17} (q_1)^2 \frac{(t-5)(t-4)}{(2t-11)(2t-9)} \left[\frac{E-E_{t1}}{E} \right]^{1/2} & \text{for } E_{t1} \leq E < E_{c1} \\ 0 & \text{for } E_{c1} < E \leq E_m \end{cases}$$

In this case $T_{t1}(z) = T_{rot}(z)$

g. for $t=36-55$ $k=1$:

$$Q_{t1}(E) = \begin{cases} 0 & \text{for } E_1 \leq E \leq E_{t1} \\ 4.7 \cdot 10^{-17} (q_2)^2 \frac{(t-35)(t-34)}{(2t-71)(2t-69)} \frac{E-E_{t1}}{E}^{1/2} & \text{for } E_{t1} \leq E < E_{c2} \\ 0 & \text{for } E_{c2} < E \leq E_m \end{cases}$$

$$\text{Further: } n_{n4}(z) = n_{n1}(z) \frac{5}{5+3 \exp\left[-\frac{E_{41}}{K_T T_{41}(z)}\right] + \exp\left[-\frac{E_{42}}{K_T T_{42}(z)}\right]}$$

$$n_{n5}(z) = n_{n1}(z) \frac{3 \exp\left[-\frac{E_{41}}{K_T T_{41}(z)}\right]}{5+3 \exp\left[-\frac{E_{41}}{K_T T_{41}(z)}\right] + \exp\left[-\frac{E_{42}}{K_T T_{42}(z)}\right]}$$

h. for $t=6-35$:

$$n_{nt}(z) = n_{n2}(z) \frac{(2t-11) \exp\left[-\frac{\chi_t}{K_T T_{t1}(z)}\right]}{35 \sum_{t=6} (2t-11) \exp\left[-\frac{\chi_t}{K_T T_{t1}(z)}\right]}$$

with $\chi_t = B_1(t-6)(t-5)$

i. for t=36-55:

$$n_{nt}(z) = n_{n3}(z) \frac{(2t-71) \exp[-Y_t/K_T T_{t1}(z)]}{55 \sum_{t=36} (2t-71) \exp\left[-\frac{Y_t}{K_T T_{t1}(z)}\right]}$$

with $Y_t = B_2(t-36)(t-35)$

j. for t=6-35:

$$E_{t1} = B_1 [(t-4)(t-3) - (t-6)(t-5)]$$

k. for t=36-55

$$E_{t1} = B_2 [(t-34)(t-33) - (t-36)(t-35)]$$

$$\text{where } \theta(x-x^1) = \begin{cases} 0 & \text{for } x < x^1 \\ 1 & \text{for } x \geq x^1 \end{cases}$$

$$\text{also } Q_{ijl}(E) = \int_{E_1}^{E-E_{ijl}} dE^1 D_{ijl}(E, E^1)$$

$$\text{with } D_{ijl}(E, E^1) = \frac{q_0 A_{ijl}}{2(E^1/2 + E_{ijl})^2} \left(\frac{E_{ijl}}{E^1/2 + E_{ijl}}\right)^{P_{ijl}} \left(\frac{E/2 + E_{ijl}}{E}\right)^{O_{ij}}$$

$$\left[1 - \left(\frac{E/2 + E_{ijl}}{E}\right)^{B_{ijl}}\right]^{N_{ijl}} \text{ for } E_{ijl} \leq E \leq E_m \text{ and } E_1 \leq E^1 \leq E - E_{ijl}$$

=0 otherwise.

The integral was approximated by the following trapezoidal type logic. This logic was also used wherever the speed of calculation was judged to be more important than accuracy.

Define $EMID_1 = E_1$

$$EMID_2 = \frac{(E_1 + E_2)}{2}$$

$$EMID_3 = \frac{(E_2 + E_3)}{2}$$

$$\vdots$$

$$EMID_{NEVAL} = \frac{(E_{NEVAL-1} + E_{NEVAL})}{2}$$

$$EMID_{NEVAL+1} = E_{NEVAL}$$

Now find $EMID_{N+1}$ s.t

$$EMID_N \leq E - E_{ijl} < EMID_{N+1}$$

Then

$$Q_{ijl}(E) = \sum_{s=1}^{N-1} (EMID_{s+1} - EMID_s) D_{ijl}(E, E_s) + (E - E_{ijl} - EMID_N) D_{ijl}(E, E_N)$$

also

$$\gamma_{mnj}(z, E) = n_{nj}(z) K_{ev}(E)^{1/2} Q_{mj}(E)$$

and

$$\gamma_{eis}^1(z) = K_{ei} n_{is}(z)$$

and

$$\gamma_{ee}^1(z, E) = K_{ee} n_e(z)$$

and

$$n_e(z) = \sum_s n_{is}(z)$$

also

$$G_{pij}(z, E) = \sum_m G_{pijm}(z, E)$$

with

$$G_{pijm}(z,E) = n_{nj}(z) Q_{pijm}(E+E_{pijm})^* I(z,E+E_{pijm})$$

$$\text{and } I(z,E) = I_b(E) \exp\left(-10^5 \secant \chi \sum_j Q_{paj}(E) \int_z^{z_m} dz^1 n_{nj}(z^1)\right)$$

In the expression for $G_{pij}(z,E)$, $Q_{pij}(E)$, $Q_{pijm}(E)$, and $I(E)$ are bar graphs. Thus, if we can evaluate $\int_z^{z_m} dz^1 n_{nj}(z^1)$ we can analytically

evaluate $\int_{E_i}^{E_m} \sum_j G_{pij}(z,E)$ since $G_{pij}(z,E)$ is piecewise constant. $F_j(z) = \int_z^{z_m} dz^1 n_{nj}(z^1)$ is approximated as follows:

The logic reads in a set of z_i values s.t. $z_1 < z_2 < \dots < z_m$.

A solution is found at one of these z values, say z_i . Then:

$$F_j(z_m) = 0$$

for all other z_i 's

$$F_j(z_i) = \sum_{t=i}^m w_{t,i} n_{nj}(z_i)$$

where $w_{j,i}$ are the weights found by dividing the integral into sub-integrals usually of ORDINT points. Thus an ORDINT point integration rule is used for the points $m, m-1, \dots, m-ORDINT+1$ and then points $m-ORDINT+1, m-ORDINT, \dots, m-2ORDINT+2$, etc., until $m-k(ORDINT-1) \leq i + ORDINT$ for $K = \text{INTEGER}$ including 0 at this point the remaining sub-integral is evaluated using the appropriate integration rule for the remaining values.

In the above logic $n_{is}(z)$ and $T_e(z)$ are guessed values and all values not otherwise described are input parameters or are described by a table of values versus E or z whichever is applicable.

After making an initial guess the iteration procedure is begun. For each iteration the values of ERR_i^{old} and the C matrix are calculated. The procedure first calculates ERR_1 and the first row of the C matrix. This corresponds to the values due to the approximation to the equation

$$SCALE1 \left(\int_{E_1}^{E_M} dE(E)^{1/2} H(z,E) \right) / n_e(z) = SCALE1$$

where SCALE1 is an input parameter adjusted so that ERR_1 is the same order of magnitude as the other ERR_i 's. The integral

$$\int_{E_1}^{E_n} dE(E)^{1/2} H(z,E)$$

is approximated as

$$\sum_{i=1}^M (E_i)^{1/2} (EMID_{i+1} - EMID_i) H(z, E_i)$$

and therefore:

$$ERR_1 = SCALE1 * \left(\sum_{i=1}^M (E_i)^{1/2} (EMID_{i+1} - EMID_i) H(z, E_i) \right) / n_e(z) - SCALE1$$

The C_{1i} values = $\frac{dERR_1}{\partial H(z, E_i)} = SCALE1 * \left(\sum_{i=1}^M (E_i)^{1/2} (EMID_{i+1} - EMID_i) / n_e - (E_i)^{1/2} (EMID_{i+1} - EMID_i) H(z, E_i) / n_e^2 \frac{\partial n_e}{\partial H(z, E_i)} \right)$ where we calculate

$n_e = \sum n_{is}(z)$ and $\frac{\partial n_e}{\partial H(z, E_i)} = \sum \frac{\partial n_{is}(z)}{\partial H(z, E_i)}$ as follows:

$$n_{i2} = (q_{pI2}^1 + q_{eni2}) \text{kev} \int_{E_1}^{E_m} dE EQr_2(E) H(z,E) + [K_{211}(z)$$

$$K_{241}(z)] n_{n1} + [K_{233}(z) + K_{243}(z)] n_{n3} + K_{244} \gamma n_{n3}$$

-1

where we approximate

$$\int_{E_1}^{E_m} dE E Q_{rs}(E) H(z, E)$$

as

$$\frac{Q_{rs}}{2^{-n_{rs}}} \sum_{i=1}^M (EMID_{i+1} - EMID_i) (E_i)^{2-n_{rs}} H(z, E_i)$$

with Q_{rs} and n_{rs} input parameters s.t. $Q_{rs}(E) = Q_{rs}(E)^{-n_{rs}}$ and

$$q_{pI2}^1 = q_{pI2}(z) + \int_{E_1}^{E_m} dE G_{pi12}(z, E)$$

with

$$q_{pij}(z) = \int_{E_1}^{E_m} dE G_{pij}(z, E)$$

where $\int_{E_1}^{E_m} dE G_{pi12}(z, E)$ and $\int_{E_1}^{E_m} dE G_{pij}(z, E)$ are approximated as described

in the explanation of the initial guess and

$$q_{enij} = \int_{E_1}^{E_m} dE G_{enij}(z, E)$$

with

$$G_{enij}(z, E) = G_{enipj}(z, E) - G_{enimj}(z, E)$$

where

$$G_{enipj}(z, E) = Kev n_{nj}(z) \sum_{\ell} \int_{E+E_{ij\ell}}^{E_m} dE^1 E^1 L_{ij\ell}(E^1, E) H(z, E^1)$$

$$G_{enimj}(z, E) = Kev n_{nj}(z) EH(z, E) \sum_{\ell} \theta(E-E_{ij\ell}) Q_{ij\ell}(E)$$

Further

$$L_{ij\ell}(E^1, E) = Q_{ij\ell}(E^1) \frac{1}{2 I_{ij\ell}} \tan^{-1} [(E-E_{ij\ell})/I_{ij\ell}]$$

$$\frac{1}{1+(E/I_{ij\ell})^2} + \frac{1}{1+[(E^1-E_{ij\ell}-E)/I_{ij\ell}]^2}$$

for $E_{ij\ell} < E^1 < E_m$ and $E_1 < E < E^1 - E_{ij\ell}$

$$L_{ij\ell}(E^1, E) = 0 \text{ otherwise .}$$

Here $Q_{ij\ell}(E)$ is approximated as described in the explanation of the initial guess.

$$\int_{E+E_{ij\ell}}^{E_m} dE E^1 L_{ij\ell}(E^1, E) H(z, E^1)$$

is approximated as follows:

$$\text{Let } EMID_t \leq E + E_{ij\ell} < EMID_{t+1}$$

Then

$$\begin{aligned} & \int_{E+E_{ij\ell}}^{E_m} dE^1 E^1 L_{ij\ell}(E^1, E) H(z, E^1) \\ & \approx \sum_{s=t+1}^M (EMID_{s+1} - EMID_s) E_s L_{ij\ell}(E_s, E) H(z, E_s) \\ & \quad + (EMID_{t+1} - E - E_{ij\ell}) E_t L_{ij\ell}(E_t, E) H(z, E_t) \end{aligned}$$

and

$$\int_{E_1}^{E_m} dE G_{enij}(z, E)$$

is approximated by

$$\sum_{t=1}^M (EMID_{t+1} - EMID_t) G_{enij}(z, E_t)$$

This leads to a form

$$q_{enij} = \sum_{t=1}^M U_{tj} H(z, E_t)$$

where the U_{tj} 's are constants. We also have

$$K_{211}(z) = CK_{211} \left(\frac{1000}{T_n(z)} \right)^{n_{211}}$$

$$K_{241}(z) = CK_{241} \left(\frac{1000}{T_n(z)} \right)^{n_{241}} - K_{211}(z)$$

$$K_{233}(z) = CK_{233} \left(\frac{1000}{T_n(z)} \right)^{n_{233}}$$

$$K_{243}(z) = CK_{243} \left(\frac{1000}{T_n(z)} \right)^{n_{243}}$$

$$K_{244}(z) = CK_{244} \left(\frac{1000}{T_n(z)} \right)^{n_{244}}$$

with $n_{n3}(z)$, γ , $n_{n1}(z)$, K_{ev} , E_{ijl} , I_{ijl} , Q_{rs} , n_{rs} , CK_{211} , N_{211} , CK_{241} , CK_{233} , n_{233} , CK_{243} , n_{243} , CK_{244} , n_{244} input parameters.

This implies $n_{i2}(z)$ is of the form

$$n_{i2}(z) \left(CON_{12} + \sum_{t=1}^M v_{t2} H(z, E_t) \right) \left(\sum_{t=1}^M w_{t2} H(z, E_t) + CON_{22} \right)^{-1}$$

where CON_{12} , v_{t2} , w_{t2} , CON_{22} are constants and $\frac{\partial n_{i2}}{\partial H(z, E_i)} = v_{i2} \left(\sum_{t=1}^M w_{t2} H(z, E_t) + CON_{22} \right)^{-1} - w_{i2} \left(CON_{12} + \sum_{t=1}^M v_{t2} H(z, E_t) \right) \left(\sum_{t=1}^M w_{t2} H(z, E_t) + CON_{22} \right)^{-2}$

Next

$$n_{i1}(z) = (q_{pi1}^1(z) + q_{eni1} + K_{211}(z) n_{n1} n_{i2}) \left(K_{ev} \int_{E_1}^{E_m} dE E Q_{r1}(E) H(z, E) + K_{142}(z) n_{n2} + K_{133}(z) n_{n3} + K_{144}(z) \gamma n_{n3} \right)^{-1}$$

with

$$K_{142}(z) = CK_{144} \left(\frac{1000}{T_n(z)} \right)^{n_{144}}$$

$$K_{133}(z) = CK_{133} \left(\frac{1000}{T_n(z)} \right)^{n_{133}}$$

$$K_{142}(z) = CK_{142*} \begin{cases} \left(\frac{750}{T_n(z)} \right)^{n_{142a}} & \text{for } T_n(z) < 750 \\ \left(\frac{750}{T_n(z)} \right)^{n_{142b}} & \text{for } T_n(z) \geq 750 \end{cases}$$

where CK_{144} , n_{144} , CK_{133} , n_{133} , CK_{142} , n_{142a} , n_{142b} are input parameters.

Further

$$q_{pi1}^1(z) = q_{pi1} - \int_{E_1}^{E_m} dE G_{pi12}(z, E)$$

which is approximated exactly as $q_{pi2}^1(z)$.

The approximation used for

$$\int_{E_1}^{E_m} dE E Q_{r1}(E) H(z, E)$$

has been previously introduced. With these approximations n_{i1} is of the form

$$n_{i1}(z) = (CON_{11} + \sum_{t=1}^M v_{t1} H(z, E_t) + CON_{21} n_{i2}(z))$$

$$\left(\sum_{t=1}^M w_{t1} H(z, E_t) + CON_{31} \right)^{-1}$$

where CON_{11} , v_{t1} , CON_{21} , w_{t1} , CON_{31} are constants and

$$\frac{\partial n_{i1}(z)}{\partial H(z, E_i)} = (u_{i1} + CON_{21} \frac{\partial n_{i2}(z)}{\partial H(z, E_i)}) \left(\sum_{t=1}^M w_{t1} H(z, E_t) \right)$$

$$+ \text{CON}_{31})^{-1} - w_{i1} (\text{CON}_{11} + \sum_{t=1}^M v_{t1} H(z, E_t) + \text{CON}_{21} n_{i2}(z))$$

$$\left(\sum_{t=1}^M w_{t1} H(z, E_t) + \text{CON}_{31} \right)^{-2}$$

Next

$$n_{i3}(z) = (q_{pi3} + q_{eni3} + K_{133}(z) n_{n3} n_{i1}(z) + K_{233}(z) n_{n3} n_{i2}(z) \left(\text{Kev} \int_{E_1}^{E_m} dE E Q_{r3}(E) H(z, E) + K_{342}(z) n_{n2} + K_{344}(z) \gamma n_{n3} \right)^{-1}$$

with

$$K_{342}(z) = \text{CK}_{344} \left(\frac{1000}{T_n(z)} \right)^{n_{342}}$$

$$K_{344}(z) = \text{CK}_{344} \left(\frac{1000}{T_n(z)} \right)^{n_{344}}$$

where CK_{342} , n_{342} , CK_{344} , n_{344} are input parameters. Using approximations similar to those used for $n_{i1}(z)$ and $n_{i2}(z)$ this equation has the form.

$$n_{i3}(z) = (\text{CON}_{13} + \sum_{t=1}^M v_{t3} H(z, E_t) + \text{CON}_{23} n_{i1}(z) + \text{CON}_{33} n_{i2}(z)) \left(\sum_{t=1}^M w_{t3} H(z, E_t) + \text{CON}_{43} \right)^{-1}$$

where CON_{13} , v_{t3} , CON_{23} , CON_{33} , w_{t3} , CON_{43} are constants and

$$\frac{\partial n_{i3}}{\partial H(z, E_i)} = (v_{i3} + \text{CON}_{23} \frac{\partial n_{i1}(z)}{\partial H(z, E_i)} + \text{CON}_{33} \frac{\partial n_{i2}(z)}{\partial H(z, E_i)})$$

$$\left(\sum_{t=1}^M w_{t3} H(z, E_t) + \text{CON}_{43} \right)^{-1} - w_{i3} (\text{CON}_{13}$$

$$+ \sum_{t=1}^M U_{t3} H(z, E_t) + \text{CON}_{23} n_{i1}(z) + \text{CON}_{33} n_{i2}(z))$$

$$\left(\sum_{t=1}^M w_{t3} H(z, E_t) + \text{CON}_{43} \right)^{-2}$$

Next

$$n_{i4}(z) = \left([K_{142}(z) n_{n2} + K_{144}(z) \gamma n_{n3}] n_{i1} \right.$$

$$+ [K_{241}(z) n_{n1} + K_{243}(z) n_{n3} + K_{244}(z) \gamma n_{n3}] n_{i2}$$

$$+ [K_{342}(z) n_{n2} + K_{344}(z) \gamma n_{n3}] n_{i3} \left. \right) \left(\text{Kev} \int_{E_1}^{E_m} dE E Q_{r4}(E) \right.$$

$$\left. H(z, E) \right)^{-1}$$

Using the approximation

$$\int_{E_1}^{E_m} dE E Q_{r4}(E)$$

this equation has the form

$$n_{i4}(z) = (\text{CON}_{14} n_{i1}(z) + \text{CON}_{24} n_{i2}(z) + \text{CON}_{34} n_{i3}(z))$$

$$\left(\sum_{t=1}^M w_{t4} H(z, E_t) \right)^{-1}$$

where CON_{14} , CON_{24} , CON_{34} , w_{t4} are constants and

$$\begin{aligned} \frac{\partial n_{i4}}{\partial H(z, E_i)} &= (\text{CON}_{14} \frac{\partial n_{i1}(z)}{\partial H(z, E_i)} + \text{CON}_{24} \frac{\partial n_{i2}(z)}{\partial H(z, E_i)} \\ &+ \text{CON}_{34} \frac{\partial n_{i3}(z)}{\partial H(z, E_i)}) \left(\sum_{t=1}^M w_{t4} H(z, E_t) \right)^{-1} - w_{i4} \\ &(\text{CON}_{14} n_{i1}(z) + \text{CON}_{24} n_{i2}(z) + \text{CON}_{34} n_{i3}(z)) \\ &\left(\sum_{t=1}^M w_{t4} H(z, E_t) \right)^{-2} \end{aligned}$$

and since

$$n_e(z) = \sum_{s=1}^4 n_{is}(z) ; \frac{\partial n_e(z)}{\partial H(z, E_t)} = \sum_{s=1}^4 \frac{\partial n_{is}(z)}{\partial H(z, E_t)}$$

we can calculate

$$\frac{\partial \text{ERR}_1}{\partial H(z, E_1)}$$

which, in general is unequal to 0 except for $E_1=0$.

So the first row of the C matrix is filled except for the first element (at this point)

$$C = \begin{pmatrix} 0 & X & X & X & \dots & X & X & X \\ & & & & & & & 0 \end{pmatrix}$$

After the first row of the C matrix and $\text{ERR}_1^{\text{old}}$ have been calculated, the logic calculates the second row of the C matrix and $\text{ERR}_2^{\text{old}}$. This will be followed by the third row of the C matrix and $\text{ERR}_3^{\text{old}}$, etc. The j^{th} row of the C matrix and $\text{ERR}_j^{\text{old}}$ uses the differential equation approximated by a difference equation for $E=E_j$. Thus equations for $E_2, E_3, \dots, E_{\text{NEVAL}}$ are considered. To calculate each row of C and its associated ERR^{old} value, the effects of TERM1, TERM2, TERM3, TERM4, TERM5, TERM6, TERM11, TERM12, TERM13 and TERM14 are analyzed separately. The effects TERM7 with TERM8 and TERM9 with TERM10 on each row of C and its associated ERR^{old} value are considered in pairs.

The effect of

$$\text{TERM1} = \sum_j \left(\frac{2 m_e}{m_{nj} + m_e} \right) \gamma_{mn}(z, E) E^{3/2} H(z, E)$$

on the C matrix and ERR^{old} is easily calculated. Here $\gamma_{mn}(z, E) = n_{nj}(z) \text{Kev}(E)^{1/2} Q_{mn}(E)$ and $m_e, m_{nj}, n_{nj}(z)$, and Kev are input parameters. $Q_{mn}(E)$ is a table of Q_{mn} values versus E which is linearly interpolated for any missing value. The effect on $\text{ERR}_i^{\text{old}}, 2 \leq i \leq \text{NEVAL}$, is to add the factor

$$\sum_j \left(\frac{2 m_e}{m_{nj} + m_e} \right) \gamma_{mn}(z, E_i) (E_i)^{3/2} H(z, E_i)$$

and the effect on the i^{th} row of C is

$$\sum_j \left(\frac{2 m_e}{m_{nj} + m_e} \right) \gamma_{mn}(z, E_i) (E_i)^{3/2} C_{i\ell} \delta(i-\ell)$$

Thus: TERM1 adds to the following elements of the C matrix.

$$\begin{bmatrix} 0, 0, \dots, 0, \dots, 0 \\ 0, X, 0, \dots, 0, \dots, 0 \\ 0, 0, X, 0, \dots, 0, \dots, 0 \\ 0, 0, 0, X, 0, \dots, 0, \dots, 0 \\ 0, 0, \dots, 0, \dots, 0, X \end{bmatrix}$$

The effect of $\text{TERM2} = \sum_s \frac{2 m_e}{m_{is} + m_e} \gamma_{eis}^1(z) H(z, E)$ on the C matrix and ERR^{old} is also calculated. Thus with $\gamma_{eis}^1(z) = K_{ei} n_{is}(z)$ and m_e, m_{is} , and K_{ei} as input parameters, the effect of TERM2 on ERR_j with $2 \leq j < \text{NEVAL}$, is to add the factor $\sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) \gamma_{eis}^1(z) H(z, E)$. The effect of TERM2 on the j^{th} row of C is found as follows:

$$\begin{aligned}
\text{add to } C_{j\ell} &= C_{j\ell}^* = \frac{\partial(\text{TERM2 for } E=E_p)}{\partial H(z, E_\ell)} \\
&= \sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) \gamma_{eis}^1(z) \delta(j-\ell) + \sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) \\
&\quad K_{ei} \frac{\partial(m_{is}(z))}{\partial H(z, E_\ell)} H(z, E_j) = \text{TERM2}^{(1)} + \text{TERM2}^{(2)} ;
\end{aligned}$$

here the effect of TERM2⁽¹⁾ is to add to the following elements of the C matrix

$$\begin{bmatrix}
0, 0, 0, \dots, 0, \dots, 0 \\
0, X, 0, \dots, 0, \dots, 0 \\
0, 0, X, 0, \dots, 0, \dots, 0 \\
\vdots \\
\vdots \\
\vdots \\
0, 0, 0, \dots, 0, \dots, 0, X
\end{bmatrix}$$

The effects of TERM2⁽²⁾ is to add the C matrix wherever any of the terms $\partial n_{is} / \partial H(z, E_\ell)$ is unequal to zero. As shown in the explanation of the first row of the C matrix these terms are generally unequal to zero. Thus the effect of TERM2⁽²⁾ is to add to the following elements of the C matrix.

$$\begin{bmatrix}
0, 0, 0, \dots, 0, \dots, 0 \\
X, X, X, \dots, X, \dots, X, X \\
X, X, X, \dots, X, \dots, X, X \\
\vdots \\
\vdots \\
\vdots \\
X, X, X, \dots, X, \dots, X, X
\end{bmatrix}$$

The effect of TERM3 = $K_{ee} I_0(z, E) H(z, E)$ on the C matrix and ERR^{old} is also calculated.

Thus with $I_0(z, E) = \int_{E_1}^{E_\ell} dE^1 (E^1)^{1/2} H(z, E^1)$ which is approximated as:

$$\sum_{t=1}^{\ell=1} (\text{EMID}_{t+1} - \text{EMID}_t) (E_t)^{1/2} H(z, E_t) + (E_\ell - \text{EMID}_\ell) (E_\ell)^{1/2} H(z, E_\ell)$$

and with the knowledge that K_{ee} is an input parameter, the effect of TERM3 on ERR_j , with $2 \leq j < \text{NEVAL}$, is to add the factor $K_{ee} I_0(z, E) H(z, E_j)$. The effect of TERM3 on the j^{th} row of C is found as follows:

$$\begin{aligned} \text{Add to } C_{j\ell} &= C_{j\ell}^* = \frac{\partial(\text{TERM3 for } E=E_j)}{\partial H(z, E_\ell)} \\ &= \left\{ \begin{array}{l} K_{ee} I_0(z, E_j) \delta(j-\ell) \\ K_{ee} (\text{EMID}_{\ell+1} - \text{EMID}_\ell) (E_\ell)^{1/2} H(z, E_\ell) \text{ for } \ell < j \\ + K_{ee} (E_j - \text{EMID}_j) (E_j)^{1/2} H(z, E_j) \text{ for } \ell = j \\ 0 \text{ for } \ell > j \end{array} \right. \\ &= \text{TERM3}^{(1)} + \text{TERM3}^{(2)} \end{aligned}$$

here the effect of $\text{TERM3}^{(1)}$ is to add to the following elements of the C matrix

$$\begin{bmatrix} 0, 0, 0, \dots, 0, \dots, 0 \\ 0, X, 0, \dots, 0, \dots, 0 \\ 0, 0, X, 0, \dots, 0, \dots, 0 \\ 0, 0, 0, X, 0, \dots, 0, \dots \\ \vdots \\ \vdots \\ 0, 0, 0, \dots, \dots, 0, \dots, 0, X \end{bmatrix}$$

The effects of $\text{TERM3}^{(2)}$ is to add to the C matrix wherever the terms $C_{j\ell}^* = \frac{\partial I_0(z, E_j)}{\partial H(z, E_\ell)}$ are unequal to zero this occurs on the lower diagonal except for the first row and the first column. Thus the effects of $\text{TERM3}^{(2)}$ is to add to the following elements of the C matrix

$$\begin{bmatrix} 0,0,0,\dots,0,\dots,0 \\ 0,X,0,\dots,0,\dots,0 \\ 0,X,X,0,\dots,0,\dots,0 \\ 0,X,X,X,0,\dots,0,\dots,0 \\ \vdots \\ 0,X,X,X,\dots,X,\dots,X,X \end{bmatrix}$$

The effect of +TERM4 = $+\int_E^{E_M} dE^1 \sum G_{pij}(z,E)$ is to add a constant to the values of ERR_j^{old} , for $2 \leq j \leq NEVAL$. This constant is approximated as described in the method for determining an initial guess for $H(z,E)$.

The effects of -TERMS = $-\int_E^{E_M} dE^1 \sum_j G_{enij}(z,E^1)$ is described in finding $n_{is}(z)$ for the first row of the C matrix. The addition to ERR_j^{old} , for $2 \leq j \leq NEVAL$, is equal to the approximation for $-\int_{E_j}^{E_M} dE^1 \sum_j G_{enij}(z,E^1)$.

The addition to the C matrix does not include the first row, first column and last row. In addition, for the usual input data some of the first and last columns have no contribution from this term. In general, however, the addition to the C matrix due to TERMS has the following form:

$$\begin{bmatrix} 0,0,0,\dots,0,\dots,0 \\ 0,X,X,X,\dots,X,\dots,X \\ 0,X,X,X,\dots,X,\dots,X \\ \vdots \\ 0,X,X,X,\dots,X,\dots,X \\ 0,0,0,\dots,0,\dots,0 \end{bmatrix}$$

The effects of

$$\begin{aligned} TERM6 &= \int_E^{E_M} dE^1 \sum_s (E^1)^{1/2} \gamma_{rs}(z,E^1) H(z,E^1) \\ &= + \int_E^{E_M} (E^1) \sum_{is} n_{is}(z) K_{rs} Q_{rs} (E)^{-n_{rs}} \end{aligned}$$

are approximated similarly to the integral described in the first row of the C matrix. So

$$\begin{aligned}
 & + \int_{E_\ell}^{E_M} (E^1) \sum n_{is} (z) \text{Kev} Q_{rs} (E)^{-n_{rs}} H(z, E^1) \\
 & \sim + \text{Kev} \sum_s \left(\frac{q_{rs} n_{is}}{2^{-n_{rs}}} \right) \left(\sum_{j=\ell+1}^M (\text{EMID}_{j+1} - \text{EMID}_j) (E_j^{2-n_{rs}}) H(z, E_j) \right. \\
 & \quad \left. + (\text{EMID}_{\ell+1} - E_\ell) (E_\ell^{2-n_{rs}}) H(z, E_\ell) \right)
 \end{aligned}$$

which is added to $\text{ERR}_\ell^{\text{old}}$, for $2 \leq \ell < \text{NEVAL}$. Analyzing $\partial \text{ERR}_\ell^{\text{old}} / \partial H(z, E_j)$ we see that TERM6 adds terms to the upper triangular portion of the matrix C, excluding the first and last row.

Thus we add the following type of terms, form B, to the C matrix.

$$\begin{bmatrix}
 0, 0, 0, \dots, 0, \dots, 0 \\
 0, X, X, X, \dots, \quad , X \\
 0, 0, X, X, \dots, \quad , X \\
 \cdot \\
 \cdot \\
 \cdot \\
 0, 0, 0, \dots, 0, \dots, 0, X, X \\
 0, 0, 0, \dots, 0, \dots, 0, 0, 0
 \end{bmatrix}$$

To find the effects of TERM7 and TERM8 we note

$$\begin{aligned}
 -\text{TERM7} - \text{TERM8} &= - \int_E^{E_M} dE^1 \sum_t \sum_k (E^1 + E_{tk})^{1/2} \gamma_{tk}(z, E^1 + E_{tk}) \\
 & \quad H(z, E^1 + E_{tk}) + \int_E^{E_M} dE^1 \sum_t \sum_k (E^1)^{1/2} \gamma_{tk}(z, E^1) H(z, E^1) \\
 &= - \sum_t \sum_k \left[\int_E^{E_M} dE^1 f_{tk}(E^1 + E_{tk}) - \int_E^{E_M} dE^1 f_{tk}(E^1) \right]
 \end{aligned}$$

where $f_{tk}(E^1) = (E^1)^{1/2} \gamma_{tk}(z, E^1) H(z, E^1) = E^1 * n_{nt}(z) * \text{Kev} * Q_{tk}(E^1) * H(z, E^1)$

with $n_{nt}(z)$ and Kev input parameters and $Q_{tk}(E^1)$ defined in the description of the initial guess. We approximate -TERM7-TERM8 as follows: Use an input parameter EREAD and rewrite -TERM7-TERM8 as

$$-\theta(EREAD-E) \sum_t \sum_k \left[\int_E^{EREAD} dE^1 f_{tk}(E^1 + E_{tk}) - \int_E^{EREAD} dE^1 f_{tk}(E^1) \right]$$

$$- \sum_t \sum_k \left[\int_{\max(EREAD, E)}^{\infty} dE^1 f_{tk}(E^1 + E_{tk}) - \int_{\max(EREAD, E)}^{\infty} dE^1 f_{tk}(E^1) \right] \text{ where we}$$

assume $EREAD - E_{tk} > 0$ and we let $E_M \rightarrow \infty$.

Now approximate $f_{tk}^1(E^1)$ by

$$\frac{f_{tk}(E^1 + E_{tk}) - f_{tk}(E^1)}{E_{tk}}$$

and cancel terms in the first two integrals then this equation can be rewritten as

$$-\theta(EREAD-E) \sum_t \sum_k \left[\int_{EREAD}^{EREAD + E_{tk}} dE f_{tk}(E^1) - \int_E^{E + E_{tk}} dE^1 f_{tk}(E^1) \right]$$

$$- \sum_t \sum_k E_{tk} (f_{tk}(\infty) - f_{tk}(\max(EREAD, E))) \text{ but } f_{tk}(\infty) = 0$$

We wish to evaluate this expression at $E_\ell = E_\ell$, for $2 \leq \ell \leq NEVAL$, and with $EREAD = E_n$ for $2 \leq n < NEVAL$ since EREAD is so set by the logic then we can approximate

$$-\theta(EREAD-E_\ell) \sum_t \sum_k \left[\int_{E_m}^{E_n + E_{tk}} dE^1 f_{tk}(E^1) \right] \text{ by}$$

$$-\theta(E_n - E_\ell) \sum_t \sum_k \left[(\min(E_n + E_{tk}, EMID_{n+1}) - E_n) f_{tk}(E_n) \right]$$

$$+ \sum_{u=n+1}^{NEVAL} \theta(EMID_u - E_n - E_{tk}) (\min(E_n + E_{tk}, EMID_{u+1}) - EMID_u) f_{tk}(E_u)]$$

Since $f_{tk}(E_u)$ is linearly related to $H(z, E_u)$ then derivatives of this approximation with $H(z, E_t)$, for $1 \leq t \leq \text{NEVAL}$, leads to additions to the C matrix for row 2 through row n-1 on the n^{th} column and possibly for a few additional terms to the right of the n^{th} column. Similarly we approximate

$$\theta(E-E_n) \sum_t \sum_k \int_{E_\ell}^{E_\ell + E_{tk}} dE^1 f_{tk}(E^1) \text{ by}$$

$$\begin{aligned} & \theta(E_n - E_\ell) \sum_t \sum_k [(\min(E_\ell + E_{tk}, \text{EMID}_{\ell+1}) - E_\ell) f_{tk}(E_\ell) \\ & + \sum_{u=\ell+1}^{\text{NEVAL}} \theta(\text{EMID}_u - E_\ell - E_{tk}) (\min(E_u + E_{tk}, \text{EMID}_{u+1}) - \text{EMID}_u) f_{tk}(E_u)] \end{aligned}$$

Since $f(E_u)$ is linearly related to $H(z, E_u)$ then derivatives of this approximation with $H(z, E_t)$, for $2 \leq t \leq \text{NEVAL}$, leads to additions to the C matrix for row 2 through n-1 on the main diagonal and possibly a few columns to the right of the main diagonal. Thus we add to $\text{ERR}_j^{\text{old}}$, for $2 \leq j \leq \text{NEVAL}$. The value

$$\begin{aligned} & -\theta(E_n - E_j) \sum_t \sum_k [(\min(E_n + E_{tk}, \text{EMID}_{n+1}) - E_n) f_{tk}(E_n) \\ & + \sum_{u=n+1}^{\text{NEVAL}} \theta(\text{EMID}_u - E_n - E_{tk}) (\min(E_n + E_{tk}, \text{EMID}_{u+1}) - \text{EMID}_u) f_{tk}(E_u)] \\ & + \theta(E_n - E_j) \sum_t \sum_k [(\min(E_j + E_{tk}, \text{EMID}_{j+1}) - E_j) f_{tk}(E_j) \\ & + \sum_{u=j+1}^{\text{NEVAL}} \theta(E_j + E_{tk} - \text{EMID}_u) (\min(E_u + E_{tk}, \text{EMID}_{u+1}) - \text{EMID}_u) f_{tk}(E_u)] \\ & + \sum_t \sum_k E_{tk} f_{tk}(\max(E_n, E_j)) \end{aligned}$$

where $f_{tk}(E^1) = E^1 n_{nt}(z) \text{Kev } Q_{tk}(E^1) H(z, E^1)$ and we add to the C matrix a matrix of the form

nth
equation

$$\begin{bmatrix}
 0,0,0,\dots & ,0 & E_n & , & \dots & ,0 \\
 0,X, \text{ a few values},0,\dots,0 & X, \text{ a few values},\dots,0 & & & & \\
 0,0,X, \text{ a few values},\dots,0 & X, \text{ a few values},\dots,0 & & & & \\
 \vdots & & & & & \\
 \vdots & & & & & \\
 X,0,0,\dots & ,0 & & & & \\
 0,X,0,\dots & ,0 & & & & \\
 0,0,X,\dots & ,0 & & & & \\
 \vdots & & & & & \\
 \vdots & & & & & \\
 0,0,0,\dots & ,0,X & & & &
 \end{bmatrix}$$

To find the effects of TERM9 and TERM10 we note

$$\begin{aligned}
 -\text{TERM9}-\text{TERM10} &= \int_E^{E_M} dE^1 \sum_t \sum_k \exp \left[-\frac{E_{tk}}{K_T T_{tk}(z)} \right] (E^1 + E_{tk})^{1/2} \\
 &\quad \gamma_{tk}(z, E^1 + E_{tk}) H(z, E^1) - \int_E^{E_M} dE^1 \sum_t \sum_k \\
 &\quad \exp \left[\frac{-E_{tk}}{K_T T_{tk}(z)} \right] (E^1)^{1/2} \\
 &\quad \gamma_{tk}(z, E^1) \theta(E^1 - E_{tk}) H(z, E^1 - E_{tk}) \\
 &= \sum_t \sum_k \left[\int_E^{E_M} dE^1 g_{tk}(E^1) - \int_E^{E_M} dE^1 g_{tk}(E^1 - E_{tk}) \right] \\
 &\quad \theta(E^1 - E_{tk})
 \end{aligned}$$

where

$$\begin{aligned}
 g_{tk}(E^1) &= (E^1 + E_{tk})^{1/2} \gamma_{tk}(z, E^1 + E_{tk}) e^{-\left(\frac{E_{tk}}{K_T T_{tk}(z)}\right)} H(z, E^1) \\
 &= e^{-\left(\frac{E_{tk}}{K_T T_{tk}(z)}\right)} (E^1 + E_{tk}) n_{nt}(z) \text{Kev} * Q_{tk}(E^1 + E_{tk}) * H(z, E^1)
 \end{aligned}$$

with K_T , $T_{tk}(z)$, $n_{nt}(z)$ and Kev input parameters and $Q_{tk}(E^1)$ defined in the description of the initial guess. We approximate -TERM9-TERM10 as follows: Use an input parameter EREAD and rewrite -TERM9-TERM10 as

$$\begin{aligned} & \theta(\text{EREAD}-E) \sum_t \sum_k \left[\left(\int_E^{\text{EREAD}} dE^1 g_{tk}(E^1) \right) \right. \\ & - \int_E^{\text{EREAD}} dE^1 g_{tk}(E^1 - E_{tk}) \theta(E^1 - E_{tk}) \\ & \left. + \sum_t \sum_k \left[\int_{\max(\text{EREAD}, E)}^{\infty} dE^1 g_{tk}(E^1) - \int_{\max(\text{EREAD}, E)}^{\infty} dE^1 g_{tk}(E^1 - E_{tk}) \right] \right] \end{aligned}$$

where we assume $\text{EREAD} - E_{tk} > 0$ and we let $E_M \rightarrow \infty$. Now approximate $g_{tk}^1(E^1)$ by

$$\frac{g_{tk}(E^1) - g_{tk}(E^1 - E_{tk})}{E_{tk}}$$

and cancel terms in the first two integrals. Now this equation can be rewritten as

$$\theta(\text{EREAD}-E) \sum_t \sum_k \left[\int_{\text{EREAD}-E_{tk}}^{\text{EREAD}} dE^1 g_{tk}(E^1) - \int_{E-E_{tk}}^E dE^1 g(E^1) \right]$$

$$\theta(E^1)] + \sum_t \sum_k E_{tk} (g_{tk}(\infty) - g_{tk}(\max(\text{EREAD}, E)))$$

but $g_{tk}(\infty) = 0$

We wish to evaluate this expression at $E=E_\ell$, for $2 \leq \ell \leq \text{NEVAL}$, and with $\text{EREAD}=E_n$ for $2 \leq n < \text{NEVAL}$ (which is how EREAD is set by the logic), then we approximate

$$\theta(E_n - E_r) \sum_t \sum_k \left[\int_{E_r - E_{tk}}^{E_n} dE^1 g_{tk}(E^1) \right] \text{ by}$$

$$+ \sum_{u=1}^{n-1} \theta(\text{EMID}_{u+1} - E_n + E_{tk}) (\text{EMID}_{u+1} - \max(\text{EMID}_u, E_n - E_{tk})) g_{tk}(E_u)]$$

Since $g_{tk}(E_u)$ is linearly related to $H(z, E_u)$ then derivatives of this approximation with $H(z, E_t)$, for $1 \leq t \leq \text{NEVAL}$, leads to additions to the C matrix for row 2 through row n-1 on the n^{th} column and possibly for a few additional terms to the left of the n^{th} column.

Similarly, we approximate

$$\begin{aligned}
 & -\theta(E_r - E_\ell) \sum_t \sum_k \int_{E_\ell - E_{tk}}^{E_\ell} dE^1 g(E^1) \theta(E^1) \quad \text{by} \\
 & -\theta(E_n - E_\ell) \sum_t \sum_k [(E_\ell - \max(\text{EMID}_u, E_\ell - E_{tk})) g_{tk}(E_\ell)] \\
 & + \sum_{u=1}^{\ell-1} \theta(\text{EMID}_{u+1} - E_\ell + E_{tk}) (\text{EMID}_{u+1} - \max(\text{EMID}_u, E_\ell - E_{tk})) g_{tk}(E_\ell) \quad .
 \end{aligned}$$

Since $g_{tk}(E_u)$ is linearly related to $H(z, E_u)$ then derivatives of this approximation with $H(z, E_t)$, for $1 \leq t \leq \text{NEVAL}$, leads to additions to the C matrix for row 2 through row n-1 on the main diagonal and possibly a few columns to the left of the main diagonal.

Thus we add to $\text{ERR}_j^{\text{old}}$, for $2 \leq j \leq \text{NEVAL}$. The value

$$\begin{aligned}
 & \theta(E_n - E_j) \sum_t \sum_k [(E_n - \max(E_n - E_{tk}, \text{EMID}_n)) g_{tk}(E_n)] \\
 & + \sum_{u=1}^{n+1} \theta(\text{EMID}_{u+1} - E_n + E_{tk}) (\text{EMID}_{u+1} - \max(\text{EMID}_u, E_n - E_{tk})) g_{tk}(E_u)] \\
 & -\theta(E_n - E_j) \sum_t \sum_k [(E_j - \max(\text{EMID}_u, E_j - E_{tk})) g_{tk}(E_j)] \\
 & + \sum_{u=1}^{j-1} \theta(\text{EMID}_{u+1} - E_j - E_{tk}) (\text{EMID}_{u+1} - \max(\text{EMID}_u, E_j - E_{tk})) g_{tk}(E_j)] \\
 & + \sum_t \sum_k E_{tk} g_{tk}(\max(E_n, E_j))
 \end{aligned}$$

where

$$g_{tk}(E^1) = (E^1 + E_{tk}) e^{-\left(\frac{E_{tk}}{K_T^* T_{tk}(z)}\right)} n_{nt}(z) \text{Kev}^* Q_{tk}(E^1 + E_{tk})^* H(z, E^1)$$

and we add to the C matrix a matrix of the form.

$$\begin{array}{ccccccc}
 0,0,0, & & \dots, & 0, & \dots & & ,0 \\
 X,X,0,\dots, & & & \text{a few values}, & X,0, & \dots & ,0 \\
 \text{a few values}, & X,0,\dots, & \text{a few values}, & X,0, & \dots & & ,0 \\
 \text{a few values}, & X,X,\dots, & \text{a few values}, & X,0, & \dots & & ,0 \\
 \\
 0,0,0,\dots & & & ,X,0,0,\dots & & & ,0 \\
 0,0,0,\dots & & & ,0,X,0,\dots, & & & ,0 \\
 0,0,0,\dots & & & ,0,0,X,\dots, & & & ,0 \\
 0,0,0,\dots & & & 0,0,0,\dots, & \dots, & 0,X & \\
 \end{array} \quad n^{\text{th}} \text{ equation}$$

The effects of

$$\begin{aligned}
 -\text{TERM11} &= \sum_j \frac{2 m_e}{m_{nj} + m_e} \gamma_{mnj}(z, E) E^{3/2} K_{Tn}(z) \frac{\partial H(z, E)}{\partial E} \\
 &= \sum_j \frac{2 m_e}{m_{nj} + m_e} n_{nj}(z) \text{Kev}(E)^2 Q_{mj}(E) K_{Tn}(z) \frac{\partial H(z, E)}{\partial E}
 \end{aligned}$$

are approximated by approximating $\frac{\partial H(z, E)}{\partial E}$ at $E_\ell = E$ for $J_0 \leq \ell < \text{NEVAL}$. J_0 is an input parameter read in as an E value $E_{j_0-1} < E_{J_0} \leq E_{j_0}$, if $1 < E_{j_0} \leq \text{NEVAL}$ and if $E_{J_0} = 0$ $J_0 = 1$.

$$\left(\frac{\partial H(z, E_\ell)}{\partial E} \right)_{\text{app.}} = \sum_{u=-LL}^{LL} E_{u, \ell} H(z, E_{\ell+u})$$

where $LL = \min(\frac{\text{ORDINT}}{2}, \ell-1, \text{NEVAL}-\ell)$ with ORDINT an input parameter and the $E_{u, \ell}$ set to be the constants which set the derivative to be the derivative at E_ℓ of the lowest order polynomial hitting the values of $H(z, E_{\ell+u})$ for u between $-LL$ and LL . For the special case

$$\ell = \text{NEVAL}, \left(\frac{\partial H(z, E_{\text{NEVAL}})}{\partial E} \right)_{\text{app.}} = \sum_{u=-1}^0 E_{u, \text{NEVAL}} H(z, E_{\text{NEVAL}+u})$$

If $\ell < J_0$ it is assumed that $H(z, E)$ can be approximated in the region around

E_ℓ by $K e^{-\frac{E}{T_e * K_T}}$ with K a constant and where

$$T_e = \left(\frac{2}{3}\right) (11605) \int_{E_1}^{E_M} dE E^{3/2} H(z, E) / n_e$$

$$\sim \left(\frac{2}{3}\right) (11605) \frac{1}{n_e} \sum_{u=1}^{NEVAL} E^{3/2} (EMID_{u+1} - EMID_u) H(z, E_u)$$

In this case we approximate $\frac{\partial H(z, E_\ell)}{\partial E}$.

$$\left(\frac{\partial H(z, E_\ell)}{\partial E}\right)_{app} = -\frac{H(z, E_\ell)}{T_e * K_T} + e^{-\left(\frac{E_\ell}{K_T * T_e}\right)} \sum_{u=-\ell\ell}^{\ell\ell} \left(\frac{E_{u,\ell} H(z, E_{\ell+u})}{E_{\ell+u}} - \frac{1}{K_T * T_e}\right)$$

Thus we add to ERR_j^{old} for $2 \leq j \leq NEVAL$ $\sum_k \left(\frac{2 m_e}{m_{nk} + m_e}\right) n_{nk}(z) Kev(E^2) Q_{mk}$

(E) $K_{T_n} T_n(z) \left(\frac{\partial H(z, E_j)}{\partial E}\right)_{app}$. If $ORDINT=7$ and $EJO > 0$ the terms

$$\frac{\left(\frac{\partial H(z, E_j)}{\partial E}\right)_{app}}{\partial H(z, E_s)}$$

lead to the following type matrix to be added to the C matrix,

$$J_0 = \begin{bmatrix} 0,0,0,\dots & ,0,\dots & ,0 \\ X,X,X, & ,X,\dots & ,X \\ X,X,X, & ,X,\dots & ,X \\ \cdot & & \\ \cdot & & \\ X,X,X, & ,X,\dots & ,X \\ 0,0,0,\dots,0,X,X,X,X,X,X,X,0,\dots & ,0 \\ 0,0,0,\dots,0,X,X,X,X,X,X,X,0,\dots & ,0 \\ 0,0,0,\dots,0,0,X,X,X,X,X,X,X,\dots & ,0 \\ \\ 0,0,0,\dots & 0,X,X,X,X,X,X,X \\ 0,0,0,\dots & ,0,X,X,X,X,X \\ 0,0,0,\dots & ,0,X,X,X \end{bmatrix}$$

where we note

$$\begin{aligned} \left(\frac{\partial H(z, E_\ell)}{\partial E} \right)_{app} & \text{ for } \ell < J_0 = \\ & = - \frac{1}{T_e * K_T} \delta(E_\ell - E_s) + \frac{\partial T_e}{\partial H(z, E_s)} \frac{H(z, E_\ell)}{T_e^2 * K_T} \\ & + \sum_{u=-\ell\ell}^{\ell\ell} \left[\frac{(E_\ell - E_{\ell+u})}{K_T T_e^2} E_{u,\ell} H(z, E_{\ell+u}) \frac{\partial T_e}{\partial H(z, E_s)} \right. \\ & \left. + e^{-\left(\frac{E_\ell + E_{\ell+u}}{K_T * T_e} \right)} E(z, E_{\ell+u}) E_{u,\ell} \delta(E_{\ell+u} - E_s) \right] \end{aligned}$$

and we note

$$\frac{\partial T_e}{\partial H(z, E_s)} \sim \left(\frac{2}{3} \right) (11605) \frac{1}{n_e} E_s^{3/2} (EMID_{s+1} - EMID_s)$$

$$- \left(\frac{2}{3}\right) (11605) \frac{1}{n_e} \sum_{u=1}^{NEVAL} (E^{3/2} (EMID_{u+1} - EMID_u) H(z, E_u)) \frac{\partial n_e}{\partial H(z, E_s)}$$

where $\partial n_e / \partial H(z, E_s)$ has been described in the description for ERR^{old} and the first row of the C matrix.

The effects of

$$\begin{aligned} -TERM12 &= \sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) \gamma_{eis}^1(z) K_{T_i}(z) \frac{\partial H(z, E)}{\partial z} \\ &= \sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) K_{ei} n_{is}(z) K_{T_i}(z) \frac{\partial H(z, E)}{\partial E} \end{aligned}$$

are found by using the approximation for $\partial H(z, E) / \partial E$ and where the approximation for $n_{is}(z)$ has been described in setting up ERR_1^{old} and the first row of the C matrix. Using these approximations we add to ERR_j^{old} for

$$2 \leq j \leq NEVAL, \sum_s \left(\frac{2 m_e}{m_{is} + m_e} \right) K_{ei}(n_{is}(z)) \underset{app}{K_{T_i}(z)} \left(\frac{\partial H(z, E_j)}{\partial E} \right) \underset{app}$$

where app indicates approximation.

The value of $\frac{-TERM12 \text{ at } E_j}{\partial H(z, E_u)}$ for $2 \leq j \leq NEVAL$ and $1 \leq u \leq NEVAL$ are found by chain rule differentiation. The derivatives of $\left(\frac{\partial H(z, E_j)}{\partial E} \right) \underset{app}$ adds a matrix of the form A type to the C matrix. The derivatives of $(n_{is}(z)) \underset{app}$ adds a matrix of the following form to the C matrix,

$$\begin{bmatrix} 0, 0, 0, \dots, 0, \dots, 0 \\ X, X, X, X, \dots, X, \dots, X \\ X, X, X, X, \dots, X, \dots, X \\ \cdot \\ \cdot \\ X, X, X, X, \dots, X, \dots, X \end{bmatrix}$$

The effects of

$$-\text{TERM13} = \frac{2 K_{ee}}{3} I_2(z, E) \frac{\partial H(z, E)}{\partial E} = \frac{2 K_{ee}}{3} \left(\int_{E_1}^E dE^1 (E^1)^{3/2} H(z, E^1) \right) \frac{\partial H(z, E)}{\partial E}$$

are found by approximating $\int_{E_1}^E dE^1 (E^1)^{3/2} H(z, E^1)$

where $E = E_\ell$ for $2 \leq \ell < \text{NEVAL}$ as $(I_2(z, E_\ell))_{\text{app}} \sum_{u=1}^{\ell} w_{u, \ell} (E_u)^{3/2} H(z, E_u)$ where $w_{u, \ell}$ are found using logic similar to the logic used in finding w_t $1 \leq t \leq m$ for approximating

$$F_j(z_i) \int_{z_i}^{z_m} dz^1 n_{nj}(z).$$

except that the polynomials start at E_1 rather than z_m .

Using these approximations we add to $\text{ERR}_j^{\text{old}}$ for

$$2 \leq j \leq \text{NEVAL}, \frac{2 K_{ee}}{3} (I_2(z, E_j))_{\text{app}} \left(\frac{\partial H(z, E_j)}{\partial E} \right)_{\text{app}}$$

The value of $\frac{-\text{TERM13 at } E_j}{\partial H(z, E_u)}$ for $2 \leq j \leq \text{NEVAL}$ and $1 \leq u \leq \text{NEVAL}$ are found by chain rule differentiation. For $\text{ORDINT}=7$ the derivatives of $\frac{\partial H(z, E_j)}{\partial E}$ app add a matrix of the following type to the C matrix.

0,0,0,	,0,	,0,0
X,X,X,		X,X
X,X,X,		,X,X
.		
.		
X,X,X,		,X,X
0,0,...,0,X,X,X,X,X,X,X,0		,0,0
0,0,...,0,0,X,X,X,X,X,X,X,0		,0,0
0,0,	0,0,X,X,X,X,X,X,X	
.		
.		
0,0,0,...	0,X,X,X,X,X,X,X	
0,0,0,...	0,X,X,X,X,X,X	
0,0,0,...	0,X,X,X	
0,0,0,...	,0,X,X	

The derivatives for $(I_2(z, E_j))$ adds a matrix of the following type to the C matrix.

0,0,0,	,0,0
X,X,0,	,0,0
X,X,X,0,	,0,0
.	
.	
X,X,X,...,X,X,0, X,X,0,	,0,0
.	
.	
X,X,X,...,X,... ,X	,X

The derivatives for $(J(z, E_j))$ adds a matrix of the following type to the C matrix

$$\begin{bmatrix}
 0, 0, 0, \dots, 0 & \dots & 0, 0, 0 \\
 0, X, X, & X & \dots & X, X \\
 0, 0, X, X, \dots, X & \dots & X, X \\
 \vdots & & \\
 \vdots & & \\
 0, 0, 0, \dots, 0, & & 0, X, X \\
 0, 0, 0, & 0, & 0, 0, 0
 \end{bmatrix}$$

$$\begin{bmatrix}
 0, 0, 0, \dots, 0, 0, 0 \\
 X, X, X, & X & \dots & X, X \\
 X, X, X, & X & \dots & X, X \\
 \vdots & & & \\
 \vdots & & & \\
 0, 0, 0, \dots, 0, 0, 0, & & 0, X, X \\
 0, 0, 0, & 0, & 0, 0, 0
 \end{bmatrix}$$

References

1. Jasperse, J.R., Bottomside Electron Profiles for G.N.C. (Problem 4916, Project 7663).
2. Theoretical Predictions of the Electron Density Distribution Function in the Ionosphere, page 40, AFGL-TR-76-0091.
3. Electron-Ion Mathematical Modeling in the Ionosphere, page 80, AFGL-TR-76-0091.
4. Jasperse, J. R., Electron Distribution Function and Ion Concentrations in the Earth's Lower Ionosphere from Boltzmann-Fokker-Plank Theory, Vol. 25, 743 to 756 of Planet. Space Sci., Pergamon Press, Northern Ireland, 1977.
5. Jasperse, John R., Smith, Ed R., The Photoelectron Flux in the Earth's Ionosphere at Energies in the Vicinity of Photoionization Peaks, Vol. 5, No. 10, pp. 843-846 of Geophysical Research Letters, October, 1978.