

September 1983

A Guide to Transform Software for Nonlinear Normal-Mode Initialization of the NCAR Community Forecast Model

Ronald Errico

ATMOSPHERIC ANALYSIS AND PREDICTION DIVISION
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH
BOULDER, COLORADO

TABLE OF CONTENTS

	<u>Page</u>
LIST OF FIGURES	v
PREFACE	vi
ACKNOWLEDGMENTS	vii
1. INTRODUCTION	1
2. THE PROBLEM	4
2.1 Forward transform	6
2.2 Backward transform	8
2.3 Modification of coefficients	9
2.4 Determination of surface pressure	11
3. DESCRIPTION OF STORAGE ARRAYS	13
3.1 Description of some parameters	13
3.2 Storage of some transform arrays	15
3.3 Storage of latitudinally dependent arrays	20
3.3.1 Storage of data	20
3.3.2 Storage of eigenvectors	22
3.4 Storage of coefficients	22
4. DESCRIPTION OF THE NMS	24
4.1 JCL instructions	24
4.2 The main program	26
4.2.1 Set constants	26
4.2.2 Read data for forward transform	28
4.2.3 Modify coefficients	28
4.2.4 Read data for backward transform	28
4.3 The forward transform	29
4.4 The backward transform	32
4.5 The accessing and saving of data files	34
4.5.1 User specification of file names and units	35
4.5.2 The accessing of files	40
4.5.3 The saving of files	41
4.6 The buffering of data	42
5. THE MODIFICATION OF COEFFICIENTS	43
5.1 Initialization control parameters	43
5.2 The routine INIT00	44
5.3 The routine INIT01	45
5.4 The routine INIT02	45
5.5 The routine COMODE0	48
5.6 The routine DEGREE	49

TABLE OF CONTENTS (continued)

	<u>Page</u>
6. MODIFICATIONS TO THE MODEL	50
7. A USER'S GUIDE TO THE NMS	51
7.1 JCL statements	51
7.2 UPDATE of the model for nonlinear initialization	52
7.3 UPDATE of the normal-mode software	54
7.4 Data input files	56
7.4.1 Input for starting iterate	57
7.4.2 Input to the model	58
7.4.3 Input for higher iterates	60
8. A DEMONSTRATION OF THE NMS	61
8.1 Convergence of an iterated solution	61
8.2 Comparison of time series of forecasts	62
8.3 Verification forecast	62
APPENDICES	
A. List of subroutines	64
B. List of variable names	67
C. The FFT routine	70
D. The ordering of modes	71
REFERENCES	73

LIST OF FIGURES

- Figure 1a CRAY JCL structure for a nonlinear initialization
- Figure 1b CRAY UPDATE of the model for nonlinear initialization
- Figure 1c CRAY UPDATE of the normal-mode software
- Figure 1d Data input for nonlinear initialization
- Figure 2a Values of VAR for the sample case
- Figure 2b Values of BAL for the sample case
- Figure 3 Time series of surface pressure for the sample case
- Figure 4a Initialized 500 mb height for the sample case
- Figure 4b Analyzed 500 mb height for the sample case
- Figure 5a Forecast 500 mb height for the sample case
- Figure 5b Verifying 500 mb height for the sample case
- Figure 6a The ordering index J for T3 truncation
- Figure 6b The ordering index J for R3 truncation

PREFACE

This report represents the culmination of an effort by Ronald Errico and James Vickroy of the Large-Scale Dynamics Section of NCAR's Atmospheric Analysis and Prediction Division. The purpose of this effort was to provide both NCAR and the university community with software for both initializing and examining atmospheric data. This software has been designed for use with both the NCAR Community Climate Model (CCM) and Community Forecast Model (CFM). It enlarges the range of studies that can be performed with those two models. It was designed especially for those investigators who want software computationally efficient, simple to use, and easy to modify. The contribution by James Vickroy to this effort is described in a separate report.

Ron Errico
September, 1983

ACKNOWLEDGMENTS

The author wishes to thank the several people who helped make this work possible. Dave Williamson helped locate errors in earlier versions of the NMS. Both Dave and Gloria Williamson provided the modifications of the CFM for interaction with the NMS. Thomas Mayer prepared the uninitialized datasets with which to test the NMS. Dave Baumhefner and Thomas Bettge analyzed the model forecasts for the final check of the NMS (various 5-day forecasts were compared). James Vickroy prepared the routines for creating the eigenvectors and authored a report related to this one. The author also had fruitful discussions with Roger Daley, Akira Kasahara, Bennert Machenhauer, Joseph Tribbia, and Yosuke Shigehisa. Mary Niemczewski typed both the preliminary and final draft version of this report. This work was partially supported by the National Oceanic and Atmospheric Administration under P.O. No. NA82AAG00927.

1. Introduction

This is the second part of a two-part technical note. In the first part (Vickroy, 1983), the normal modes of the NCAR Community Forecast Model (CFM) were described along with a description of the computer software designed to generate them. In this part, software designed to transform data from physical space into modal coefficient space are described along with software for nonlinear normal-mode initialization. The normal-mode software is hereafter referred to as the NMS. This note is especially designed for those who wish to initialize data, investigate dynamic balance, or investigate the modal structure of the CFM and its solutions.

The CFM is described in Williamson (1983) and in Sato et al. (1983). It is a global atmospheric model which includes physics and topography. It is spectral in the horizontal, with (user specified) pentagonal truncation. The vertical coordinate is sigma, with data defined at discrete levels. The number of levels is adjustable.

The normal modes, described in Vickroy (1983), are those of the CFM linearized about a state of rest with a statically-stable temperature field varying in the vertical only. The linearization excludes physics, parameterized eddy diffusion processes, and water vapor. The resulting eigenvalue problem is separable into two parts: a vertical part which yields vertical structure functions as the eigenvectors with equivalent depths as eigenvalues, and a horizontal part which includes the equivalent depth as a parameter. For each equivalent depth, the normal modes can be separated into three types: eastward propagating gravity modes, westward propagating gravity modes, and westward propagating rotational modes.

Details of the modes which are relevant to this report appear in Sections 2 through 5. Other information, and references, appear in Vickroy (1983).

The NMS is written in FORTRAN for the CRAY-1. The bulk of the software is stored as a CRAY UPDATE (version 1.8) program library, and can easily be edited as such. The job control language instructions (JCL) to be described are from the CRAY operating system (COS) version 1.10. This version allows loops in the job control.

The NMS is written in modular form. That is, distinct tasks are performed in distinct subroutines. This provides for easy modification and debugging as well as the potential for easy output of intermediate results. The code itself is extensively documented with comment cards. Variable labels are either mnemonic or chosen to agree with identical variables in the CFM.

The NMS transforms data from a multi σ -level, Gaussian grid directly to normal mode coefficients. Spherical harmonic coefficients of the data fields are not calculated in an intermediate step. The NMS also includes an inverse transform. The vectors and matrices necessary for the transformations need to be created using separate software described in Vickroy (1983).

The NMS is also designed to perform either linear or nonlinear normal-mode initialization. The necessary modifications of the coefficients are computed in separate subroutines. These routines also print diagnostics regarding the normal-mode coefficients.

Since the nonlinear initialization procedure requires calculation of the time tendencies of the normal modes, and since the process is iterative, the JCL is written using a loop structure. This is described in Section 4.1. The time tendencies are calculated by inputting data into the CFM, forecasting ahead one or more time steps, outputting the forecast fields, and then constructing the time tendencies by a forward difference approximation. The first three steps of this procedure are done by the CFM code; only the last step is accomplished in the code described here. The CFM code and the code described here are kept separate. The two codes are alternately loaded and executed within the JCL loop. Datasets containing grid-point fields in history-tape format are passed between the two programs.

The structure of this technical report is as follows: After the introduction, mathematical descriptions of the various transforms and the nonlinear initialization procedure of Machenhauer (1977) are presented. The truncation, scaling, and storage of data, normal modes, and coefficients are described in Section 3 (further descriptions appear in Vickroy (1983)). In Section 4, the JCL and software for performing the transforms are described in detail. The software for balancing or modifying coefficients appears in Section 5. In Section 6, some recommended modifications to the CFM for performing a nonlinear initialization are described. Section 7 contains brief instructions for the casual user. A twenty-four hour forecast from data initialized using the NMS is presented in Section 8. Lists of FORTRAN symbolic names and other information appear in appendices at the conclusion of this report.

2. The problem

The NMS accomplishes three tasks, all of which are necessary in order to perform a non-linear initialization. All involve calculations with grid-point data (hereafter called data) and normal mode coefficients (hereafter called coefficients). These tasks are:

- a) A forward transform: from data to coefficients;
- b) A backward transform: from coefficients to data;
- c) A modification of coefficients: linear and nonlinear initialization.

These tasks are described in a mathematical context here. Software is described in later sections.

The data set is defined on a grid with coordinates $\lambda_i, \mu_j, \sigma_k$. The first refers to longitude $\lambda_i = 2\pi(i-1)/N_1$, $i = 1, \dots, N_1$. The second refers to the sine of the Gaussian latitude θ_j , $j=1, \dots, N_2$. The last refers to the k th (full) sigma level, $k = 1, \dots, N_3$. The relevant data input to the NMS are u (the eastward wind component), v (the northward wind component), and T (the absolute temperature), as well as p_* (the surface pressure) and ϕ_* (the surface geopotential). The latter two are only defined at the surface, on a horizontal grid. The N 's determine the resolution of the grid on which the data are defined.

A complete description of both the normal modes and the dynamic system whose solution they describe is presented in Vickroy (1983). A summary of that description appears below so that the transform methods can be described. The modes are most easily described as a combination of two parts.

The first part is a vertical structure function $z_{k, \ell}$. The index ℓ denotes the vertical mode ($\ell = 1, \dots, L$; where $L = N_3$), and k is the σ -level index described earlier. The z 's are determined as eigenvectors of a particular matrix (described in Vickroy (1983)). They form a matrix Z . Associated with each ℓ there is an eigenvalue Φ_ℓ . This is an equivalent geopotential (i.e., the product of an equivalent depth and the gravitational constant g).

The second part is a horizontal structure function which is an eigenfunction of a system of shallow-water equations for a specified equivalent depth Φ_ℓ/g . It has three components u' , v' , and ϕ' . The primes denote a rescaling of the fields as described below. These components have implied indices j, m, n, ℓ . As earlier, j and ℓ denote indices for the Gaussian latitude and vertical mode, respectively. At any latitude, the normal modes have a structure $\exp(m\lambda\sqrt{-1})$, where m is a zonal wavenumber. The index n completes the specification of the mode. It denotes the meridional structure (see Section 3.2). Hereafter, the notation will be simplified by introducing a vector index J whose three components are values of $m, n,$ and ℓ , respectively. Associated with each structure function is an eigenvalue ω_J .

Associated with each normal mode is a coefficient (i.e., amplitude). This is denoted y_J . These are the coefficients which multiply the structure functions when data are expanded in terms of the normal modes. Given a (spherical harmonic) spectral truncation and a horizontal grid large enough to prevent aliasing in the quadrature to be described, the data can be completely described in terms of the set of y_J . How this is accomplished is shown in the next subsection.

2.1 The forward transform

The first step in determining the y_j , given the data, is the calculation of ϕ , a pseudo-geopotential. It is that quantity whose Laplacian appears as a linear term in the divergence equation when the primitive equations are described in σ -coordinates. It is a function of the grid-point indices:

$$\phi_{i,j,k} = \sum_{k'} g_{k,k'} (T_{i,j,k'} - \bar{T}_{k'}) + R\bar{T}_k \ln p_{*i,j} + \phi_{*i,j}. \quad (1)$$

R is the gas constant. T is the temperature at the denoted grid point. \bar{T} is the reference-state σ -level temperature used to define the normal modes (see Vickroy (1983)). g is an element in the hydrostatic matrix (see Vickroy (1983)). The other variables have been described previously. ϕ is computed for each i , j , and k .

The next step is a transformation from the index k (i.e., σ -levels) to index ℓ (i.e., vertical modes). These transformed values will be denoted by a caret (^); e.g.,

$$\hat{u}_{i,j,\ell} = \sum_k z'_{\ell,k} u_{i,j,k}. \quad (2)$$

This is computed for each i , j , and ℓ . \hat{v} and $\hat{\phi}$ are computed similarly with the same matrix Z' , which is the inverse of the matrix Z of vertical eigenvectors.

The third step is a transformation from data defined in terms of longitude to data defined in terms of zonal wavenumbers m (i.e., a Fourier transform), $m = -M, \dots, 0, \dots, M$. These transformed values are denoted by carets and primes, e.g.,

$$\hat{u}'_{m,j,\ell} = \frac{1}{N_1} \sum_i \hat{u}_{i,j,\ell} \exp(-m\lambda_i \sqrt{-1}). \quad (3)$$

These transformed quantities are now in general complex numbers, except for those with $m = 0$ which are always real. Since the data are real, the transformed values satisfy

$$\hat{u}'_{m,j,\ell} = \hat{u}'_{-m,j,\ell}^* \quad (4)$$

for all m . The asterisk superscript denotes a complex conjugate. Therefore, only values for non-negative m need be explicitly calculated. This is done for each j and ℓ . \hat{v}' and $\hat{\phi}'$ are computed similarly.

The final transform utilizes an orthogonality property of the horizontal structure functions expressed as :

$$\delta_{n,n'} = \sum_j (u'_{j,m,n,\ell} u'_{j,m,n',\ell} + v'_{j,m,n,\ell} v'_{j,m,n',\ell} + \phi'_{j,m,n,\ell} \phi'_{j,m,n',\ell}) G_j \quad (5)$$

The δ 's are Kronecker deltas. G_j is the Gaussian weight (for quadrature) at the j th latitude. The eigenvector components, denoted by primes (without carets), are expressed here in dimensionless form: u' , v' , and ϕ' are each scaled by $U_* = 2\Omega a$, where Ω is the earth's angular rotation rate, a the earth's radius, and ϕ' has an additional scaling factor $\phi_\ell^{1/2}$. Dimensional coefficients are then obtained from

$$y_{m,n,\ell} = \sum_j (\hat{u}'_{m,j,\ell} u'_{j,m,n,\ell} + \sqrt{-1} \hat{v}'_{m,j,\ell} v'_{j,m,n,\ell} + \phi_\ell^{-1/2} \hat{\phi}'_{m,j,\ell} \phi'_{j,m,n,\ell}) G_j \quad (6)$$

if quantities determined from the data (i.e., those with carets) are dimensional. The y_j are, in general, complex. (See Section 5.5 for special conditions when $m = 0$).

The quantity that represents total (kinetic plus available potential) energy in a linearized, shallow-water model of equivalent depth ϕ_ℓ/g is

$$E'_\ell = \frac{\rho\pi a^2\phi_\ell}{g} \sum_j \sum_i [(u_{i,j,\ell}^2 + v_{i,j,\ell}^2) + \phi_\ell^{-1} \phi_{i,j,\ell}^2] G_j, \quad (7)$$

where ρ is a mass density, and $\sum_j G_j = 2$. E'_ℓ can also be expressed in terms of the coefficients as

$$E'_\ell = \frac{4\rho\pi a^2\phi_\ell}{g} E_\ell, \quad (8)$$

where

$$E_\ell = \sum_n \sum_{m \geq 0} d_m y_{m,n,\ell} y_{m,n,\ell}^*, \quad (9)$$

with $d_m = 1/4$ for $m = 0$ and $d_m = 1/2$ for $m > 0$ (a consequence of Eq.

4). The quantity $4\rho\pi a^2\phi_\ell/g$ is the total mass of the fluid of given depth and density. Thus, E_ℓ is the total energy per unit mass.

2.2 The Backward Transform

Given the coefficients, data may be recreated by transforming backwards. The steps follow.

First, \hat{u}' , \hat{v}' , and $\hat{\phi}'$ are determined from

$$\hat{u}'_{m,j,\ell} = \sum_n y_{m,n,\ell} u'_{j,m,n,\ell}, \quad (10)$$

$$\hat{v}'_{m,j,\ell} = -\sqrt{-1} \sum_n y_{m,n,\ell} v'_{j,m,n,\ell}, \quad (11)$$

$$\hat{\phi}'_{m,j,\ell} = \phi_\ell^{1/2} \sum_n y_{m,n,\ell} \phi'_{j,m,n,\ell}, \quad (12)$$

for each m and ℓ .

Second, \hat{u} , \hat{v} , and $\hat{\phi}$ are determined from the inverse Fourier transform; e.g.,

$$\hat{u}_{i,j,\ell} = \sum_m \hat{u}'_{m,j,\ell} \exp(m\lambda_i \sqrt{-1}). \quad (13)$$

Finally, the grid-point values for u , v , and ϕ are determined; e.g.,

$$u_{i,j,k} = \sum_{\ell} z_{k,\ell} \hat{u}_{i,j,\ell}. \quad (14)$$

The determination of T from ϕ is described in Section 2.4.

2.3 Modification of coefficients

A linear initialization simply sets those y_J corresponding to gravity modes equal to zero. If the CFM were linear, gravity waves then would be absent from the forecast fields. Since the model is nonlinear however, dy_J/dt may be nonzero although $y_J = 0$. If dy_J/dt is not small initially, y_J will not remain smooth in the forecast.

Machenhauer (1977) suggested setting $dy_J/dt = 0$ initially for all gravity modes. This can be justified for most gravity modes by using scaling arguments.

The equation for dy_J/dt is

$$\frac{dy_J}{dt} = -\sqrt{-1} \omega_J y_J + N_J(y,y), \quad (15)$$

where N is a nonlinear term which generally depends on modes other than only y_J itself. Thus N acts to couple the linear modes. Setting $dy_J/dt = 0$ initially requires setting

$$y_J = -\sqrt{-1} N_J(y,y)/\omega_J \quad (16)$$

initially.

For the CFM, (16) is generally a sufficiently complicated system of equations so that it must be solved iteratively. Let a superscript I denote an iteration number, and set

$$y_J^{I+1} = -\sqrt{-I} N_J(y^I, y^I)/\omega_J. \quad (17)$$

It is not clear that (17) should converge to some solution for y_J . Indeed, in some cases it does not converge. However, the cases where (17) does converge are general enough to warrant its use as an initialization procedure.

N is a very complicated expression to derive. Therefore it is easiest to iterate (16) in the following manner. Set

$$y_J^{I+1} = y_J^I + \Delta y_J^I, \quad (18)$$

$$\Delta y_J^I = -\sqrt{-I} (dy_J/dt)^I / \omega_J. \quad (19)$$

The I th iterate of the time tendency can be approximated by

$$(dy_J/dt)^I = \frac{y_J^I(t=\Delta t) - y_J^I(t=0)}{\Delta t}, \quad (20)$$

where $y_J^I(t=\Delta t)$ is a forecast value of y_J using $y_J^I(t=0)$ as the initial condition. Δt is a small forward time step. y_J for $I = 0$ are most easily set to those y_J given by either the uninitialized data or a linear initialization.

By iterating (16) in the form of (18) through (20) the NMS is greatly simplified. Expressions for N or dy_J/dt need not be determined. Instead, the CFM may be used to forecast ahead one time step Δt , and the NMS simply transforms from CFM data fields to coefficients or vice versa. The coding

for a nonlinear initialization is greatly simplified by treating the CFM as a black box.

To quantify the convergence of (18) through (20), the quantity

$$F_{\ell}^I = \sum_n \sum_{m>0} d_m (dy_{m,n,\ell}/dt)^I (dy_{m,n,\ell}/dt)^{*I}, \quad (21)$$

may be compared with E_{ℓ}^I . As in (9), $d_m = 1/4$ for $m = 0$ and $d_m = 1/2$ for $m > 0$.

If the scheme converges, then an F_{ℓ}^I calculated from initialized gravity modes should be smaller with each iteration until some limit is reached due to round-off error. If the modes are slowly varying in time, then $E_{\ell}^I > F_{\ell}^I$ for any large set of modes.

2.4 Determination of surface pressure

In the backward transform, as a final step it is necessary to determine both T and p_* from ϕ . This cannot be done uniquely since there is one more unknown than known quantities. The procedure described by Temperton and Williamson (1981) is that chosen here, although the notation is different. For a justification of this procedure, see their paper.

Denote the projection of all the Δy_J^I on the pseudo-geopotential field as $\Delta\phi^I$. Then

$$(\ln p_*^{I+1} - \ln p_*^I)_{i,j} = \sum_k c_k \Delta\phi_{i,j,k}^I \quad (22)$$

where

$$c_k = \sum_{\ell} (\Delta\sigma)_{\ell} c'_{\ell,k} \quad (23)$$

The $\Delta\sigma$'s are those describing the σ -level spacing and the c' are elements of the inverse of that matrix from which the vertical modes are determined. (The latter is equivalent to $Z D Z'$, where D is the diagonal matrix of values of Φ_ρ , appropriately ordered.)

Once p_*^{I+1} is determined, T^{I+1} is given by the inverse of (1), namely,

$$T_{i,j,k}^{I+1} = \bar{T}_k + \sum_{k'} g'_{k,k'} (\phi_{i,j,k'}^{I+1} - R\bar{T}_{k'} \ln p_{*i,j}^{I+1} - \phi_{*i,j}) \quad (24)$$

where the g' are elements of the inverse of the hydrostatic matrix.

3. Description of storage arrays

This section is divided into four parts. Parameters which define spectral truncations are described in the first part. In the second part, the storage of all transformation arrays (except horizontal eigenvectors) is described. In the third part, storage of those arrays which are functions of latitude (i.e., data and horizontal eigenvectors) are described along with some input/output (I/O) operations. The storage of coefficients is described in the last part. For details on the definition of the normal modes (their derivation, structure, etc.), the reader should consult Vickroy (1983).

The CFM is a spectral model with a (user defined) pentagonal truncation. Grid-point data, or nonlinear functions within the CFM are defined at specified σ -levels on a horizontal grid which is equally spaced in longitude and Gaussian-spaced in latitude. In what follows, m and n' shall denote, respectively, the order and degree of an associated Legendre polynomial in the truncated spectral description of the data. Equivalently, m will be called a wavenumber, or more precisely, a zonal wavenumber. The index n (without a prime) will be reserved for the ordering of the modes.

3.1 Description of some parameters

There are four sets of user-defined parameters in the NMS. They are all defined once, at the beginning of the main program. The first and last sets are described here. The second set, which defines how modes are initialized, is described along with the software that modifies coefficients (Section 5.1). The third set, which defines file names, is described along with the accessing and saving of files (Section 4.5). All

the user-defined parameters are specified using a CRAY UPDATE command, as described in Section 7.

The first set of user-defined parameters define the spectral and grid truncations:

NLEV: the number of full sigma levels.

MMAX: the maximum zonal wavenumber m .

NMAX: the maximum value of n' for $m = 0$. For pentagonal truncation, this is also the maximum value of $n'-m$ for $0 \leq m \leq$
MMAX.

KMAX: the maximum value of n' given any m . For pentagonal truncation, this is equivalent to the maximum n' associated with $m=MMAX$.

NLON: the number of grid points around any latitude circle.

NLAT: the number of Gaussian latitudes.

The last set of user-defined parameters serves several functions. This set includes:

ITYPES: the number of types of modes stored: equal to 2 for only gravity modes and equal to 3 for all modes.

ISYMT: a symmetry indicator, set equal to -1 if the normal modes have been determined using a resting reference atmosphere so that the modes have the symmetry properties described in Section 3.2.2 and Appendix D. ISYMT must be set equal to +1 if the modes do not have these properties.

IBUF: a parameter equal to 1 if data are read in, or 2 if data are buffered in (see Section 4.6).

LVIRT: a logical parameter that is TRUE if ϕ is calculated from virtual temperatures, rather than from T itself. LVIRT is FALSE otherwise.

The pentagonal truncation is completely specified by the values MMAX, NMAX, and KMAX. The total number of retained spherical harmonic functions is

$$NMODES = MMAXP1 * NMAXP1 - (MMAX + NMAX - KMAX + 1) * (MMAX + NMAX - KMAX) / 2$$

where $MMAXP1 = MMAX + 1$ and $NMAXP1 = NMAX + 1$. $MMAXP1$ is an augmentation of the dimension MMAX so that wavenumber 0 is associated with an index $M=1$, wavenumber 1 with $M=2$, ..., and wavenumber MMAX with $M=MMAXP1$.

Similarly, a value $n' - m = 0$ is associated with an index $N = 1$ (i.e., $N = n' - m + 1$), so that the index $N = 1, \dots, NMAXP1$. Thus, there are no zero indices. The number of real and imaginary parts associated with the zonal description of the modes is expressed as $MMAX2 = 2 * MMAXP1$.

3.2 Storage of transform arrays

All transform arrays, eigenvalues, and eigenvectors are first created by a routine described in Vickroy (1983) and stored on a file. The information on that file is ordered so as to facilitate CRAY vector processing and I/O within the NMS. That ordering is described both in this section and in Appendix D. The ordered dataset is read by the NMS from an acquired file which is assigned to unit number IUNITA.

The first four records on that file are read in the routine RPARMA ("Read PARAMeters on unit A"). These records contain all the arrays

necessary to transform the data, except for the horizontal eigenvectors which are read in the main program. RPARMA is called by the main program.

The first record read contains respective values for NLEV, MMAX, NMAX, KMAX, NLAT, NLON, ISYMT, NMODES, and ITYPES. These values are compared with those set by the user. If any corresponding values disagree, a message that describes the differences is printed. Such an error indicates that the wrong file of transform arrays was specified by the user. Execution of the NMS is terminated in this case. A 20-word alphanumeric array FLABEL is also read in the first record. It contains a dataset label.

The second record contains all the arrays which describe the vertical modes. These include:

SIGMA(L): the value of σ at level $L/2$; for $L=1, \dots, 2*NLEV+1$.

TBAR(L): a (specified) σ -surface mean temperature ($^{\circ}K$) at (full) level $L-1$; $L=1, \dots, NLEV+2$. These temperatures define the reference atmosphere about which the model equations are linearized. TBAR(NLEV+2) is a surface value, and TBAR(1)=TBAR(2). The first and last values of TBAR do not affect the modes; they are present to facilitate coding only.

RTBAR(L): the product of the gas constant R and a corresponding element of TBAR.

GMATRX: a matrix which transforms temperatures to geopotential using the hydrostatic relationship.

RGMTRX: the inverse of GMATRX.

- ZMATRX: a matrix whose columns are the vertical structure functions (i.e., vertical eigenvectors) of the normal modes.
- RZMTRX: the inverse of ZMATRX.
- EDEPTH: a vector (of length NLEV) containing the equivalent depths of the vertical modes, in units of m.
- SEDPH: a vector (of length NLEV) containing the square roots of corresponding elements of the product of EDEPTH and the gravitational constant $g = 9.8 \text{ ms}^{-2}$.
- CPMTRX: a vector (of length NLEV) for calculating changes in surface pressure (see Section 4.4).

The four matrices are size NLEV×NLEV. The vertical modes are ordered from largest equivalent depth to smallest. The eigenvectors are normalized such that

$$1 = \sum_{I=1}^{\text{NLEV}} (\text{SIGMA}(2*I+1) - \text{SIGMA}(2*I-1)) * \text{ZMATRX}(I,L)**2$$

for each mode L. These modes are not necessarily orthogonal. The components of ZMATRX and RZMTRX are dimensionless. The components of GMATRX have units of $\text{m}^2 \text{ s}^{-2} \text{ K}^{-1}$, and are ordered such that the first component describes the contribution of the temperature at SIGMA(2) to the geopotential at that level.

The third record contains the following:

- GLAT: a vector containing the sines of the Gaussian latitudes.
(These values are not used by the NMS.)
- GAUSWT: a vector of Gaussian weights for Gaussian quadrature in the meridional direction.
- NTRUNC: a vector (of length MMAXP1) containing values of $\max(n'-m+1)$ given index $M = m + 1$.

MTRUNC: a vector (of length $NMAXP1$) containing values of $\max(m+1)$ given index $N = n' - m + 1$.

INDEX: a two-dimensional array (of size $MMAXP1, NMAXP1$) which describes the ordering of the normal modes as a function of indices $M = m + 1$ and $N = n$, where n is defined below.

GLAT and GAUSWT each have length NLAT. The latitudes are ordered to agree with those on the data file; i.e., alternating northern and southern latitudes, starting from those most poleward and ending at those most equatorward. The values of NTRUNC, MTRUNC, and INDEX are completely determined by those of MMAX, NMAX, and KMAX, as described in Appendix D.

The last record read in RPARMA is the three-dimensional array, ROMEGA, which contains the frequencies (i.e., eigenvalues) of the linear modes. The last index for ROMEGA is the vertical mode index, with values $L = 1, \dots, NLEV$. The smaller the value of L , the larger the equivalent depth of a mode. (Thus, $L = 1$ indicates the external mode.) The second index, $ITYPE=1, \dots, ITYPES$, denotes the type of mode. Values of 1, 2, and 3 denote westward gravity waves, eastward gravity waves, and rotational modes, respectively (an exception occurs when $J = 1$ (see the end of this section); also, for zonally-symmetric modes (i.e., those with $m = 0$), the designations "westward" and "eastward" refer only to, respectively, minus and plus signs for the values of ROMEGA, and not to motions of those modes). The first index of ROMEGA specifies the mode given its type and vertical mode index. This index has values $J=1, \dots, NMODES$.

The ordering of the modes is described most easily in terms of an ordering by zonal-wavenumber index M and meridional scale $n=1, \dots, NTRUNC(M)$. A smaller value of n indicates a larger meridional scale. The

ordering using n is identical to one using the modes' frequencies; for a given M , smaller n indicate lower frequencies in the case of gravity modes, and higher frequencies in the case of rotational modes (whose frequencies scale as Rossby waves). The modes are ordered such that all of those with smaller values of n precede those with larger values. For those modes with identical values of n , the modes are ordered in the order of their zonal wavenumbers, smallest to largest. Thus, the first mode has the largest meridional scale for $M = 1$ (zonal wavenumber $m = 0$), the second has the largest meridional scale for $M = 2$ (zonal wavenumber $m = 1$), ..., the $M_{MAX}+1$ mode has the largest meridional scale for the largest zonal wavenumber, the $M_{MAX}+2$ mode has the second largest meridional scale for $M = 1$, and the $M_{MAX}+3$ mode has the second largest meridional scale for $M = 2$, etc.

This ordering of the modes is equivalent to that of the Legendre polynomials in the CFM since the ordering and truncation of n are equivalent to those of n' . As such, the ordering is described by the array $INDEX(M,N)$ as presented in Appendix D, if N is interpreted as either n or $n'-m+1$. Thus, a mode of any particular scale can be referenced by setting its index $J=INDEX(M,N)$. As in the CFM, this somewhat complicated ordering is necessary in order to vectorize the Gaussian quadrature in the horizontal transforms. Examples of the ordering for simple rhomboidal and triangular truncations appear in Appendix D.

The modes denoted by $J=1$ are treated specially. These modes have no horizontal variation. They also have zero frequency. For that reason, all $J=1$ modes are treated as rotational modes. The $J=1$ modes associated with the index $ITYPE=1$ are the projections of the horizontal-mean pseudo-

geopotential fields onto the vertical modes. The $J=1$ modes associated with the index $ITYPE=2$ are derived from a constant horizontal velocity potential field. They are therefore modes whose eigenvector components in terms of velocity and geopotential are zeros. Those denoted by $ITYPE=3$ are derived from a constant streamfunction field. They also are modes whose eigenvector components are zeros.

3.3 Storage of latitudinally dependent arrays

3.3.1 Storage of data

The NMS is designed to read data which are stored using the format for CFM history tapes. For data at any particular time, the first record contains a header. This header contains a description of the dataset as well as values for pointers which describe the positioning of data. This header is read in the subroutine RPARMB (which stands for Read PARaMeters on unit B). There, the truncation parameters set in the NMS (MMAX, NMAX, KMAX, NLEV, NLAT, NLON) are compared with those on file. Comparisons of values of SIGMA are made as well. Pointers for the various fields described below are located. A check that the data are in a format appropriate for the NMS is made. Record lengths are also read from the header and tested to ensure that arrays are dimensioned sufficiently large. If any problem is detected by these tests, a message which describes it is printed. Execution then is terminated by a call to ABORT.

The header is followed by NLAT records of data. Each record contains all data at one latitude. The first is the northernmost one. The next is the southernmost one. The latitudes continue to alternate northern then southern, with each successive pair being closer to the equator.

Within each record, the data appears in a string. While the entire string must be read if the length of the records are to be maintained in an initialization procedure, only data on the first portion of each record is actually referenced by the NMS. The entire record of length NRSIZE is read into an array B having the two dimensions, NBDATA and IBUF (NBDATA>NRSIZE).

The position of relevant data within the array B is described by pointers. By means of these pointers, starting addresses for various data arrays are passed as subroutine arguments. Within the subroutines, these fields are separately described as

PHIS(N), the surface geopotential ($\text{m}^2 \text{s}^{-2}$), with pointer NPHIS.

PS(N), the surface pressure (Pascals), with pointer NPS.

T(N,L), the temperature (degrees Kelvin), with pointer NT.

U(N,L), the eastward velocity (m s^{-1}), with pointer NU.

V(N,L), the northward velocity (m s^{-1}), with pointer NV.

Q(N,L), the specific humidity (g kg^{-1}), with pointer NQ.

N varies from 1 to NLONP2=NLON+2. For $N < \text{NLON}$, the value N corresponds to a longitude $((N-1)/\text{NLON}) * 360^\circ$. The remaining two values of N denote words containing redundant or no information. L is the (full) σ -level index. Thus there are NUVT=NLONP2*NLEV words for each of T, U, V, and Q. Values of NT, NU, NV must be ordered consecutively, separated by NUVT, for the proper execution of routine ZONALT (see Section 4.3). Within some subroutines, portions of this data are overwritten as detailed in Sections 4.3 and 4.4.

3.3.2 Storage of eigenvectors

The u' , v' , and ϕ' components of the eigenvectors (normal modes) are read from a file on unit number IUNITA in the main program. Each record read contains all components at one latitude. If ISYMT=1 is specified by the user (Section 3.1), then the vectors are read as the data are read: records for alternating northern and southern latitudes, starting at the northernmost one and working equatorward.

For the case where the reference atmosphere is one of rest, then ISYMT=-1 should be set by the user (Section 3.1). In this case, only northern hemisphere latitudes need be read (and file space is reduced) since the NMS can use the symmetry properties of the vectors. The symmetry of each vector is determined by its meridional index n (Section 3.2). For gravity modes, those denoted by odd values of n have u and ϕ symmetric, whereas those denoted by even values of n have v symmetric. For rotational modes, the situation is reversed (e.g., odd values of n indicate antisymmetric u and ϕ). Where u is symmetric, v is antisymmetric, and vice versa.

Each record is read into an array VECTR. This array has the five dimensions NMODES, ITYPES, NLEV, 3, and IBUF. Values for the fourth index denote components for u' , v' , and ϕ' , respectively. For each value of that index, the vectors are stored as ROMEGA is stored (Section 3.2). The components of VECTR have been scaled as described in Section 2.1.

3.4 Storage of coefficients

The coefficients are stored in a complex array COEF. Components of COEF are stored as those of ROMEGA are stored. The coefficients are scaled such that the squared modulus of any particular coefficient equals four

times the contribution to the kinetic plus available potential energy per unit mass (units of J/kg) by that mode. This scaling has been described more completely in Section 2.1. When two sets of coefficients need to be stored (e.g., in order to calculate time tendencies), an old set is stored on either unit IUNITC or in the array VECTR if the latter can be overwritten.

4. Description of the NMS

This section begins with a subsection describing details of the JCL instructions in the NMS. This is followed by a description of the main program in the NMS. The next two subsections are descriptions of the subroutines for performing the forward and backward transforms, respectively. The final subsection is a description of the accessing and saving of the datasets that exchange information with the CFM in a nonlinear-initialization procedure. The subroutines for modifying the coefficients are described in Section 5.

4.1 JCL instructions

A sample set of JCL instructions for the NMS is presented in Fig. 1a. Note the imbedded comments within this structure. Figure 1a is also described briefly in Section 7.1. The listed instructions are those following the JOB and DISPOSE cards.

The user must set an integer variable J1 in the first instruction. This variable is used as a CRAY JCL loop counter, as described below. Its value must be at least as great as the number of nonlinear iterations in a nonlinear initialization procedure (this latter number is specified by ITMAX: cf. Section 5.1). It must be at least one larger than this number if balance diagnostics are to be computed for the output (i.e., for the initialized dataset.)

The next 9 instructions acquire and update files pertaining to the model. The files acquired include: a radiation program library, labeled here as CCMRDA; a model program library, labeled here as CCMZBA; and a file of radiation data labeled here as qqqqq2. The latter file must have

truncation compatible with the user-specified model and NMS resolutions. The two program libraries are updated as described in Sections 6 and 7.2. Then they are compiled. The resulting binary load module is assigned the local dataset name MODEL. For further explanation of these JCL instructions, the user should consult Sato et al. (1983).

The next four instructions concern that program library described in this report. First, that library is acquired with the name INITIAL. Next, a user-determined version is created using a CRAY UPDATE command. It reads UPDATE instructions and FORTRAN statements which appear in the third file on \$IN (not counting the JCL file). This UPDATE file is necessary to specify, for example, truncation parameters, array dimensions, and file names. The UPDATE command creates a compilation dataset named CINITL. Next, a CFT command creates a binary load module BINITL. This load module is then executed using an LDR command. This execution creates a starting iterate for nonlinear normal mode initialization. The identical load module is used to execute later iterations. A namelist input variable is used to control which type of iteration is executed (see Section 4.2.1). If only this iterate is desired (e.g., as a linear initialization), execution should be terminated at this point. In this case, the statements that reference model files may be deleted as well.

The remaining instructions are executed J1 times in a CRAY JCL loop. First, the acquired version of the CFM, denoted by the local dataset name MODEL, is executed. Then the UPDATED nonlinear initialization routine, denoted by the name BINITL, is executed. Prior to each execution, the input file \$IN is repositioned so that the next CRAY file to be read contains the data input (i.e., Namelist variables, etc.) for the

appropriate routine (CFM or INITAL). At the end of each iteration, J1 is reduced by one, until the loop is exited when $J1 = 0$.

4.2 The main program

Fortran statements in the NMS can be identified by their CRAY-UPDATE sequence identifier. For the main program, these identifiers all begin with the label INITAL, followed by a period and an integer, starting at 2 and incrementing by 1. That is, INITAL is the name of the program library deck containing the main program.

The main program is also called INITAL. It can be divided into four parts. Most constants are set or read in the first part. In the second part, data and horizontal eigenvectors are read, and normal-mode coefficients are determined. The next part calls a subroutine that modifies the coefficients to satisfy some condition (e.g., a balance condition). The last part also reads data and eigenvectors, and changes the data to agree with the modified coefficients using an inverse transform. These parts of the main program are described in detail below. The actual transform calculations are done in subroutines. They are described in later subsections.

4.2.1 Set constants

The first part of INITAL can be divided into four subsections. In the first, parameters are set and dimensions are declared. This is followed by a description of the variables. The setting of parameters is described in Sections 3.1, 4.5, and 5.1. A list of all variables appears in Appendix B. An iteration counter ITERAT and the normal mode coefficients COEF are set to zero in DATA statements. Default values of some

NAMelist variables (described below, and in Section 4.5.1) also are set in DATA statements.

A namelist, labeled as LIST1, is read in the second subsection. The NAMelist variable INLTYP indicates the type of initialization to be executed:

INLTYP=0 indicates that coefficients are to be determined from data.

No modifications of data are made.

INLTYP=1 indicates that a linear initialization is to be executed.

INLTYP=2 indicates that an iterate (excluding a starting iterate) of Machenhauer's initialization procedure is to be executed.

As a starting iterate for Machenhauer's scheme, INLTYP = either 0 or 1 must be chosen. The effects of INLTYP on data files and coefficients are described in detail in Sections 4.5 and 5.2, respectively. The remaining NAMelist variables specify files, and are described in Section 4.5.1.

In the next subsection, the unit numbers IUNITD and IUNITE are set to perform the functions specified by the control variable INLTYP. These functions are described in Section 4.5.

In the fourth subsection, constants that are used in the Fast Fourier Transform (FFT) are calculated. They are stored in the common block labeled COMFFT. The FFT itself is described in Appendix C.

The subroutines RPARMA and RPARMB are called in the last subsection. These routines have been described in Sections 3.2 and 3.3.1, respectively.

4.2.2 Read data for forward transform

In the second part of INITIAL, a subroutine GRD2NM ("GRiD TO Normal Mode") is called once for each latitude of data which are read from unit IUNITB. That subroutine in turn calls the various routines that transform the data. They are described in Section 4.3. The main program here functions primarily to read data and pass it to GRD2NM with the appropriate arguments.

The contributions to the coefficients by each latitude of data are accumulated in the array COEF. Only at the conclusion of the loop over latitudes does COEF contain the actual (i.e., totaled) value of the coefficients. They are stored as described in Section 3.4.

If ISYMT = -1 (i.e., if the horizontal structure functions have the appropriate symmetry properties, as described in Section 3.3.2), then the variable JSIGN denotes either a northern latitude (+1) or a southern latitude (-1). Also, if ISYMT = -1, only northern hemisphere components of the eigenvectors are read, as described in Section 3.2.2. If ISYMT = 1, then components for both hemispheres are read, and JSIGN = +1 for all latitudes.

4.2.3 Modify coefficients

Actual modifications of coefficients are done within subroutines described in Section 5. The arrays VECTR, WORK, and DZ are used for work space within those routines.

4.2.4 Read data for backward transform

The last part of INITIAL is structured almost identically to the second part. There are four significant differences.

In the beginning of this part, unit IUNITA is rewound and those records that have previously been read in RPARMA are skipped. At this point unit IUNITA is positioned so that the next record contains the northernmost latitude of eigenvector components.

A second difference is that data are read from the file on unit IUNITE. IUNITE equals IUNITB in a linear initialization and is unique in a nonlinear iteration (see Section 4.5 for further explanation). The header record on IUNITE is copied to the output file on unit IUNITD. This copy also acts to position IUNITE so that the next record contains the northernmost latitude of data.

A subroutine NM2GRD ("Normal Mode TO GRiD") is then called. It, in turn, calls the other subroutines that calculate data from coefficients. The coefficients are input to this routine (see Section 4.4). After each call to NM2GRD, the latitude of data modified within that routine is written to the file on unit IUNITD.

When the final initialized dataset is to be created, the name of that file, as well as the date, time, and sequence number of its creation, are inserted into the header. This insertion is done in a routine WRHEAD.

4.3 The forward transform subroutines

The passing of data and vectors to the subroutine GRD2NM has been described in Section 4.2.2. GRD2NM calls each routine that carries out the successive operations which are needed to calculate the contribution to the modal amplitudes by one latitude of data. The loop in ECMWFB that calls GRD2NM is equivalent to the quadrature described in Eqs. 1, 2, 3, and 6. GRD2NM and all the routines it calls are in the program library deck also named GRD2NM.

GRD2NM first calls the routine LOGPS. Input to this routine is the vector PS of surface pressure values. On output, this vector is overwritten, with each value replaced by the natural logarithm of that value. Although PS has dimension NLON+2, only the logarithms of the first NLON values are calculated since the last two values contain extraneous information. (They are set equal to zero in this case; cf. 3.3.1.)

If LVIRT = .TRUE. has been set by the user in INITAL, then a routine VIRT is called. In this routine, the virtual temperature T_v is calculated from the temperature T and specific humidity Q at each data point using the relation

$$T_v = T(.622 + Q)/(.622 (1 + Q)).$$

The array containing T is actually over-written by the values of T_v , so that ϕ will be calculated (see below) from T_v rather than T. If LVIRT = .FALSE., then the array T is unchanged at this point, and the array Q is not referenced explicitly by the NMS.

The third routine called is HYDROS ("HYDROStatic equation"). In this routine, the σ -surface "pseudo-geopotential" ϕ is calculated as described by Eq. 1. The temperature, logarithm of the surface pressure, surface geopotential, σ -surface mean temperature of the reference state, and hydrostatic matrix are input along with array dimensions. On output, the temperature array T is overwritten by the corresponding array of ϕ . HYDROS uses a work array in order to vectorize the routine.

The fourth routine, ZONALT, calls a vectorized FFT routine. This routine is described in Appendix C. It computes an FFT of the T (now ϕ), u, and v fields. The latter two arguments need not be listed since those

fields are stored immediately following T. If their storage is changed for some reason, the call to this routine must be changed as well.

The fifth routine is called SIGMAT. It computes the transformation from σ -levels to vertical mode coefficients for the U, V, and T arrays. It overwrites the input arrays, with values for vertical mode L replacing those for σ -level L. A work array is used in order to vectorize the transformation. SIGMAT is called after ZONALT so that the transformation need be done on MMAX2 spectral coefficients rather than NLON longitudinal data points (a 30% computational savings). Before output, the T (i.e., ϕ) field is divided by the square root (SEDPATH) of an equivalent geopotential depth in preparation for computations in the next routine.

The final routine, HOUGHT ("HOUGH Transform") computes the terms for one latitude in the Gaussian quadrature described by Eq. 6, Section 2.1. There are four loops in this routine. The first is a loop over the vertical mode index. The second is a loop over the types of modes. The third loop is over the number of diagonals (i.e., values of $N=n$, as described in Section 3.2). The inner loop is essentially a loop over zonal wavenumber m . These last two loops are designed to vectorize the routine.

The arrays U, V, Z, and COEF are treated as complex here. If ISYMT=-1 is set by the user (Section 3.1), then the value of JSIGN input to HOUGHT changes the sign of asymmetric northern-hemisphere vector components when multiplying southern-hemisphere data. Data are scaled in this routine so that the coefficients have the dimensional form described in Section 3.4. A minus sign is present before the imaginary components of V since the V-components of the horizontal eigenvectors have an implied factor of $\sqrt{-1}$, as described in Vickroy (1983).

4.4 The backward transform subroutines

The passing of data and eigenvectors to the subroutine NM2GRD has been described in Section 4.2.3. NM2GRD calls each routine that carries out the successive calculations which modify the data from modifications of coefficients and unmodified data using Eqs. 10 through 14 and 22 through 24 in Section 2. The term modification is used here in the sense of ΔU , ΔV , ΔT , ΔCOEF , etc.; Δ defined as in Eq. 18. In particular, ΔPS is computed using Eq. 22. The structure of NM2GRD is similar to that of GRD2NM. NM2GRD and all routines it calls, except VIRTR, ZEROA, and ZONALT, are in the program library deck named NM2GRD.

First, all elements of the work-space arrays WORK and Z are set to zero. They contain values of Δu , Δv , and $\Delta \phi$ at various steps of the transform.

Next, the routine HOUGHR is called. It computes the projection of Δy onto fields of Δu , Δv , and $\Delta \phi$ (as functions of zonal wavenumber $M-1$ and vertical mode L) using Eqs. 10-12. The values are stored in the array WORK. They are scaled here so that the final results are fields with units of m/s, m/s, and m^2/s^2 , respectively. In other respects, HOUGHR is similar to HOUGHT.

The call to HOUGHR is followed by one to ZONALT. The third argument +1 indicates an inverse Fourier transform of the first argument, WORK. In the process, WORK is overwritten: output values are those of Δu , Δv , $\Delta \phi$ as a function of (discrete) longitude and vertical mode number. The FFT is described in Appendix C.

SIGMAR is the next routine called. It transforms from fields in terms of vertical-mode amplitudes to σ -level data. In this sense, it is

the reciprocal of the routine SIGMAT. Rather than overwriting the input array WORK, the projections are added to the arrays U, V, and Z. Thus if, on input, U and V are old fields (i.e., unmodified, in the sense of an old iterate) and WORK contains modification fields, then U and V are the new fields (i.e., modified, in the sense of new iterate). In this case, Z contains values of $\Delta\phi$. On the other hand, if U and V are zeroed by the user before this point (i.e., with an UPDATE instruction) and if COEF is an array of coefficients rather than modifications, then the output U, V, and Z simply contain the projections of the coefficients onto the data grid. In this latter case, some method for determining a surface pressure is needed (see Section 2.4).

A modification of the surface pressure is computed in the routine SURFPR using the method described in Section 2.4. Input to this routine are values of $\Delta\phi$ (the array Z) and the transformation vector c, (the vector CPMTRX). Values of $\Delta\ln p_*$ at discrete longitudes are overwritten onto the array WORK (denoted by DPS in this subroutine). Values of the surface pressure, obtained from the file on unite IUNITE, are also input. They are overwritten by their respective natural logarithms.

If LVIRT = .FALSE., then the temperature is calculated next in routine HYDRSR using the reciprocal hydrostatic relation, Eq. 23. If, on input, T is nonzero and Z contains values of $\Delta\phi$ rather than ϕ itself, then on output, T is a new (i.e., modified) temperature data array. This calculation also uses input values of $\Delta\ln p_*$. However, it does not use values of PHIS (i.e., ϕ_*) as long as the topography is held fixed (i.e., $\Delta\phi_* = 0$). Instructions for calculating a T from ϕ , $\ln p_*$, and ϕ_* are described

by comment cards within the routine. If, on input to NM2GRD, COEF contains coefficients themselves rather than modifications of coefficients, then the user must change the routine as described.

If LVIRT = .TRUE., then values of zero are input to HYDRSR as temperatures. Therefore, on output, the temporary array WORK contains values of ΔT_v rather than T_v (the subscript v denotes these as virtual temperatures). The routine VIRTR, which is called next, calculates ΔT from ΔT_v , and then replaces the old (input) values of T by the new $T + \Delta T$.

Finally, a modified surface pressure is calculated in the routine LOGPSR. It is calculated from exponentiated values of $\ln p_* + \Delta \ln p_*$.

4.5 The Accessing and Saving of Data Files

There are several files to which the program INITAL must be attached. Excluding those referenced in the CRAY JCL file (as described in Section 4.1), these files are:

1. a file of matrices and vectors for the normal-mode transforms;
2. a file of data from which coefficients are to be determined (this is usually an analysis dataset to be initialized, or a forecast dataset from which time tendencies will be calculated);
3. a file of coefficients (the coefficients usually are those corresponding to an iterate of the initialized data);
4. A file of data containing modified fields (INITAL writes the newest iterate of the initialized data to this file);
5. a file of data whose fields are to be modified (this is usually either an analysis dataset or the previous iterate in an initialization procedure);
6. a file which contains temporary copies of datasets;

7. a file to which diagnostic output is written (usually this is \$OUT).

These files are on units specified by the values of IUNITA, IUNITB, IUNITC, IUNITD, IUNITE, IUNITF, and IUNITO, respectively.

All files read or written to by the NMS are manipulated (e.g., saved, copied, disposed, etc.) using FORTRAN-callable CRAY JCL. Names of permanent datasets and unit numbers are specified by the user as described in Section 4.5.1. The accessing of those files is described in Section 4.5.2, and file manipulations upon termination (of each iteration) of the program are described in Section 4.5.3.

4.5.1 User specification of file names and units

The files used by the NMS may be separated into two sets. One set includes all permanent files input to, and output from, the job. That is, they exist prior to, and/or after, the (entire) job execution. This set includes those files accessed in the CRAY JCL file (e.g., the model and NMS program libraries) as well as files specified in a PARAMETER statement at the beginning of the main program (INITAL). The system files \$IN and \$OUT are included in this set.

The second set of files includes those local to the job and those permanent datasets which have been created during job execution, but subsequently are deleted from the disk during execution. This set includes the files produced by UPDATE or CFT commands in the JCL file, which already have been described in Section 4.1. It also includes files used to pass data between the model and initialization software and/or between different iterates of the initialization software. The control and naming

of these latter files is specified using NAMELIST input. In this way, different iterates of the same program may be instructed to do different tasks.

Those file names specified as parameters include:

FLVECT = 6Hqqqqq3, where qqqqq3 is the appropriate file of transform arrays (i.e., eigenvectors, as described in Sections 3.2 and 3.3.2).

FLDATA = 6Hqqqqq4, where qqqqq4 is the appropriate data file, in CFM history-tape format, containing fields to be initialized.

FLOUT = 6Hqqqqq5, where qqqqq5 is the name of a dataset which will contain the final (desired) iterate of the initialized fields.

The files specified by FLVECT and FLDATA must agree with both the model and NMS regarding the spectral truncation.

All NAMELIST variables are read in the single list LIST1. This list includes the variable INLTYP which controls both the type of initialization (Section 4.2.1) and some file control operations (described below). LIST1 is read at the start of execution of the program INITAL. An example of its use to execute Machenhauer's scheme appears in Section 7.4. The NAMELIST variables are described in greater detail in this Section.

ISAVE is a variable which indicates whether datasets used solely for communication between the initialization software and the model are local (=0) or permanent (=1). For example, if the input dataset to the model is accessed using a (FORTRAN) CALL ACQUIRE, and output using a CALL SAVE, then the initialization software must treat this dataset as permanent, although before completion of the job that dataset may cease to exist. However, if the model treats these datasets as local (rewinding, rather than

saving them), then the initialization software must also treat them as local. The choice of values of ISAVE thus primarily depends on the choice of model configuration. ISAVE=1 by default. The value of ISAVE does not affect the saving of FLOUT or the accessing of FLVECT and FLDATA.

Local datasets are to be preferred. They cannot accidentally interact with those of other jobs. On the other hand, permanent datasets may cross job boundaries. Since these permanent datasets are actually temporary, in the sense described above, the user may attempt to run two different jobs using the same names and passwords for these files, with possible unnoticed job exchange occurring. As of the date of this writing, however, only the use of permanent datasets (ISAVE=1) is possible. The modification of the model for using local datasets is planned, and the NMS anticipates this change.

PASWRD=6Hwwwww, where wwwwww is a user specified write password for any permanent files created by the NMS. If ISAVE=1 has been specified by the user (or by default), then the password name defined in the model should be set identically. This is done by setting the NAMELIST variable NSWRPS in the model input list EXPDEF. The default value for both PASWRD and NSWRPS is 6HWRITEb ('b' is a blank). If ISAVE=0, then PASWRD applies only to the file FLOUT of (final) initialized data.

A dataset is referenced in the NMS by either an assigned unit number IUNITx (x = one of A, ..., F, or O) or a local dataset name (DN=) FTnn, where nn equals the value of IUNITx. If ISAVE=1 is specified, the unit numbers may be set to any allowed integer which does not conflict with any other unit numbers. However, if ISAVE=0, then certain unit numbers must agree with those of the model, as described below.

IUNITA is the unit number assigned to the dataset of vectors. This is the dataset whose name is specified by the PARAMETER FLVECT. The file is acquired only for a starting iterate (INLTYP=1 or 0). It then remains local to the job for any higher iterates. Unit IUNITA is rewound at the end of both forward and backward transforms.

IUNITB is the unit number assigned to the file of data from which coefficients are to be determined. If ISAVE=1, this dataset is accessed with the permanent dataset name given by the value of the namelist variable FTDIN. This latter variable has a default value equal to that of the parameter FLDATA. The default value for IUNITB is 26.

If ISAVE=0 has been specified, the value of IUNITB must be set appropriately. For example, for a nonzero iterate of Machenhauer's scheme (INLTYP=2), IUNITB must be the unit number assigned to the model's history-tape output. In this way, INITAL will be set to determine forecast coefficients (i.e., $y_J(t = \Delta t)$ in Eq. 20, Section 2.3) for calculation of time tendencies by a finite-difference approximation. Similarly, for a starting iterate (INLTYP=0 or 1), IUNITB should be the unit number assigned to the model input.

If ISAVE=1 has been specified (or chosen by default), then the specification of FTDIN must be done as carefully as for IUNITB when ISAVE=0. For the case INLTYP=2, the value of FTDIN must be identical to that of the model's NAMELIST variable LTAPES. In this case, that value must be different from all other file names. For the cases INLTYP=0 or 1, FTDIN must have its default value.

IUNITC is the unit number assigned to the file containing normal-mode coefficients, as described in sections 5.2 - 5.4. IUNITC is treated as a

local dataset and therefore may be assigned any value independent of other units. Its default value is 27.

IUNITD is the unit number assigned to a file of output from INITAL. It is in CFM history-tape format. It contains fields which have been modified using one of the initialization routines. If ISAVE=1, this file is saved with a permanent dataset name specified by the NAMELIST variable FTDOUT. The default value of the latter is the value of the parameter FLOUT. IUNITD may have any value which does not conflict with any other unit number. Its default value is 28.

For a nonzero iteration of Machenhauer's scheme (INLTYP=2), if ISAVE=1 has been specified (or chosen by default), the value of the model input NAMELIST variable LABDAT must be identical to that of FTDOUT. The value given by the parameter FLOUT is suitable. Thus, the model input will be the newest iterate of the initialized data.

IUNITE is the unit number assigned to the dataset containing fields to be modified. If both ISAVE=1 and IUNITE≠IUNITB, then IUNITE is associated with the permanent dataset name specified by the value of the NAMELIST variable FTDIN2. The default value of the latter is identical to that of the parameter FLOUT (appropriate for the case INLTYP=2).

For certain purposes, IUNITE (or FTDIN2) must have certain values. For a linear initialization, IUNITE=IUNITB is set within INITAL by the condition INLTYP=1. In that case, the name FTDIN2 is not referenced at all. For a nonzero iterate of Machenhauer's scheme, then FTDIN2=FTDOUT on condition that INLTYP=2. If ISAVE=0 has been specified, then IUNITE must be set by the user to equal the unit number for model input. For the case INLTYP=0, IUNITE is not referenced.

IUNITF is assigned to a unit used for copying of files; e.g., prior to a dispose. In most cases IUNITE may be used for these purposes, but it appears less confusing to separate the functions of this unit. IUNITF has the default value 39.

IUNITO is the unit number for some printed output. This output includes descriptions of the user-specified I/O files and truncation parameters. IUNITO=6 is the default value. For this value, this information is printed on \$OUT. If this information is not desired (e.g., to reduce the quantity of output), IUNITO should be set to some unassigned unit number in the NAMELIST input. The value of IUNITO applies only to the initialization output, and not to the model output. The diagnostics BAL and VAR, as well as possible error messages, are written directly to \$OUT, irrespective of the value of IUNITO.

4.5.2 The accessing of files

All files to be read by the NMS are accessed in the routine FILEIN. This routine is divided into several parts. Which part is executed depends on the function of the program (as specified by the input variable INLTYP; Section 4.2.1) and on the parameter ISAVE (Section 4.5.1). FILEIN is in a program library deck also named FILEIN.

If this is a first-time execution of the NMS for this job (e.g., a starting iterate) files of data and eigenvectors are acquired. That file of data is copied to a local dataset so that the non-initialized dataset remains intact.

If this is a nonzero iterate of Machenhauer's scheme, and if communication with the model is via permanent datasets (ISAVE=1), then the files

FTDIN and FTDIN2=FTDOUT are accessed. However, if communication is via local datasets (ISAVE=0), then both input and output data files to the model are simply rewound. All files are then properly positioned for reading by the NMS.

4.5.3 The saving of files

The last routine called by the main program is called FILOUT. This routine has several functions. Which functions are executed depends on the input parameters ISAVE (defined in Section 4.5.1), ITMAX, and ITERAT (defined in Section 5.1). FILOUT is in a program library deck also named FILOUT.

If ISAVE=0, specifying that communication with the model is via local datasets, then both input and output data files are rewound. The file of modified data on unit IUNITD is then copied to unit IUNITE. The latter is then rewound for ready access by the model as input data. Further, if ITMAX=ITERAT, which indicates that this iterate is to be saved as an initialized dataset, then this iterate is both saved and disposed (to the mass store device). It is given the file name specified by the parameter FLOUT with the write-protect password specified by PASWRD (both user-specified as described in 4.5.1).

If ISAVE=1, then the old iterate of the file on IUNITB is deleted and released. An exception occurs when IUNITB=IUNITD, which indicates that the file on unit IUNITB contains analyzed data to be used as a starting iterate (INLTYP=0). In that case, the file on unit IUNITB is not deleted, but instead is saved for input to the CFM. If IUNITE and IUNITB are distinct, as they are for nonzero iterates of nonlinear normal mode

initialization, then the old iterate of initialized data residing on unit IUNITE is deleted and released as well.

The new iterate of initialized data is saved with a file name specified by the variable FTDOU. The newly saved file is then released by the job, for later access by the model. Prior to this, if ITMAX=ITERAT, an initialized dataset is saved and disposed, as described above for the case ISAVE=0.

After FILOUT is executed, all files are properly positioned for another iterate of the model and NMS.

4.6 The buffering of data

In order to speed execution, an option is included in the NMS that allows for the buffering of data. It utilizes BUFFER IN statements. A record of data may thereby be buffered in concurrently as the previous record of data is transformed.

The BUFFER IN option is selected by the user specifying IBUF=2 in a parameter statement, as described in Section 3.1. The only drawback for this option is that the required dimensions for the arrays B and VECTR are then doubled. However, in most cases, the savings in I/O wait time with IBUF=2 will outweigh this drawback.

If IBUF=1 is user specified, then both data and eigenvectors are input using READ statements. Each record is read only after the transforms of data on the previous record are completed.

5. THE MODIFICATION OF COEFFICIENTS

The coefficients are modified in subroutines. The initialization procedures are controlled by a set of user-defined parameters and a NAMELIST variable INLTYP. The parameters are described in Section 5.1, and INLTYP has been described in Section 4.2.1. The specific subroutine chosen depends on the value of INLTYP. The routines called for values of INLTYP=0,1,2 are described in Sections 5.2, 5.3, and 5.4, respectively. These routines call two additional subroutines, CMODEO and DEGREE. They are described in Sections 5.5 and 5.6, respectively.

5.1 Initialization control parameters

A set of parameters which define how modes are initialized must be defined at the beginning of the main program. They are:

- ITMAX the number of iterations used to determine the initialized data. This count excludes the starting iteration. Note that $J1 > ITMAX$ must be specified in the job JCL instructions (Section 4.1).
- DTIME the time Δt , in seconds, used to calculate the finite difference approximation to the time derivatives. As such, it must equal the difference in time between the initial and forecast datasets for the model as run with the NMS software (Section 4.1). Thus, DTIME here must equal the product of the two model-input variables, NWTIME and DTIME.
- LFIX an integer, $1 \leq LFIX \leq NLEV$. Only gravity modes with vertical mode index $L \leq LFIX$ are initialized, although all those modes are not necessarily initialized (cf., CUTOFF, below). The vertical

modes are ordered from largest equivalent depth to smallest, so, for example, LFIX=2 means only the external and first internal modes are initialized. If LFIX=NLEV is specified by the user, the choice of modes which are initialized is independent of equivalent depth.

CUTOFF a period, in hours. Only modes with linear periods less than or equal to the value of CUTOFF are initialized. The linear periods are those determined from the corresponding eigenvalues of the modes. If CUTOFF $\geq 1.E10$ is specified, then no period cutoff is made. A cutoff frequency is determined from the value of CUTOFF using the function FREQNC called in the main program.

5.2 The routine INIT00

If INLTYP=0, then the routine INIT00 is called. The function of this routine is primarily to provide a starting iterate of non-modified coefficients (i.e., as determined by analyzed data) for an iterative, nonlinear initialization procedure. This routine also calls the routine CMODE0 that ensures the coefficients describe real fields (as described in Section 5.5), and calls the routine DEGREE which calculates the energy for particular sets of modes (as described in Section 5.6). INIT00 is in a program library deck of the same name.

The coefficients, as input to INIT00, are written to IUNITC. They are written on a single record using a binary WRITE. The coefficients are superseded on this record by the single integer ITERAT, which is an iteration counter. For this starting iterate, ITERAT=0 on input to INIT00. After this WRITE, IUNITC is immediately rewound, so as to be read by the next iterate of an initialization procedure.

The energy of the modes, as a function of modal type and vertical mode index, are written to \$OUT using a formatted PRINT.

After INIT00 is executed, execution of the main program proceeds with the rewinding and/or saving of particular files. No backward transform is executed in this case.

5.3 The routine INIT01

INIT01 is called if INLTYP=1. This routine performs a linear initialization, and may be used as a starting iterate for an iterative procedure. Its execution is therefore similar to that of INIT00 with the following important exceptions: those gravity modes to be initialized, as determined by the parameters CUTOFF and LFIX, are set to zero; the coefficients, as modified, are written to IUNITC; and the input coefficients are overwritten by an array of changes of those coefficients. Output to the main program are zero-valued coefficients for those modes not to be initialized, and values equal to the negative of those input to INIT01 for those to be initialized. Thus, the latter modes are subtracted from the data. INIT01 is in a program library deck of the same name.

5.4 The routine INIT02

The routine INIT02 is called if INLTYP=2. This routine calculates the changes to coefficients in an iteration of Machenhauer's scheme (excluding the starting iterate), as described by Eqs. 18-20 in Section 2.3. INIT02 is in a program library deck of the same name.

On input, COEF1 is a COMPLEX array of normal-mode coefficients. These are the coefficients determined from model forecasted fields at the

time `DTIME` ahead of the initial time. These values are used to calculate time tendencies, as described below and in Eq. 20, Section 2.3. On output, `COEF1` contains the modifications of the coefficients (i.e., values of Δy). The storage of the coefficients is described in more detail in Section 3.4.

On input to `DELTAY`, the arguments `COEF2`, `TEMP`, `BAL`, and `VAR` contain starting addresses for storage arrays. `ROMEGA` is the array of frequencies of the normal modes. `ITERATE` is an iteration counter which is zero valued on input. `LFIX` and `FREQNC` are described in Section 5.1. Other input arguments are dimension and truncation parameters.

At the start of execution of `INITO2`, the previous iterate of the initialized coefficients is read from unit `IUNITC`. These values are stored in a temporary array `COEF2` in a manner identical to that of `COEF1`. At this time, the iteration number associated with `COEF2` also is read and stored as the variable `ITERAT`. Unit `IUNITC` is immediately rewound. The value of `ITERAT` is printed, and then immediately incremented by 1.

All calculations occur within loops over the vertical mode index `L` and modal-type index `I`. At the beginning of each loop over `L`, the routine `CMODE0` is called. This routine insures that the coefficients describe real fields. It is described in Section 5.5. All the calculations within the `I` loop occur within loops over the mode index `J`. This index has been described in Section 3.2.

Next, the time tendencies of all the modes of type `I` and vertical-mode index `L` are calculated using a finite difference approximation (Eq. 20, Section 2.3). These values are stored in the temporary complex array `TEMP`.

Measures of balance are calculated next. These are the values VAR and BAL described by Eqs. 9 and 21, respectively, in Section 2. These values are accumulations over all modes of the given type and equivalent depth. Actual computations are performed in the routine DEGREE.

Next, either the modifications (Δy 's) of the modes are determined using Eq. 19, Section 2.3, or the modifications are set to zero (i.e., no initialization). Which action is taken depends on whether the particular mode is one to be initialized or not. A mode is initialized if three conditions are satisfied: it is a gravity mode ($I < 3$); its index $L < L_{FIX}$; and its frequency (i.e., value of ROMEQA) is not less than the cutoff value FREQNC. For those modes that have been modified, values of the new iterate of their coefficients are computed. These values overwrite the old values stored in COEF2.

The rotational mode coefficients remain unchanged in Machenhauer's nonlinear iteration scheme. Consequently, the only rotational mode calculations performed in DELTAY are those for BAL and VAR. These values are to be compared with those for the gravitational modes in determining a degree of balance.

Each time the routine DELTAY is entered, the line

```
NEW ITERATION      NUMBER iii
```

is printed. After calculations are performed for all modes of each vertical-mode index L, BAL is printed for both gravity modes (summed over westward and eastward gravity modes, and denoted in the output by BAL(G)) and for rotational modes (denoted by BAL(R)). Values for VAR are printed in the same format as BAL, on the same line.

These values are printed specifically so that the convergence of the nonlinear initialization scheme may be checked. If the scheme is converging to the desired solution, BAL(G) should decrease with each iteration for those gravity modes being initialized. At the same time VAR(G) of those modes should converge to some finite value. VAR(R) should remain fixed. BAL(R) should vary with each iteration since dy/dt for rotational modes is affected by gravity modes, which are changing. The variation in BAL(R) should be small since, for typical atmospheric values, the effect of gravity modes on rotational modes should be relatively small, at least for large equivalent depth modes. Values of these quantities, as determined for a software demonstration, are presented in Section 8.

After the loop over L, the new iterate of the coefficients, as stored in COEF2, are output to the file on unit IUNITC in a single record. On that record, COEF2 is preceded by the new value of the iteration counter ITERAT. Unit IUNITC is then rewound.

5.5 The routine CMODE0

The routine CMODE0 corrects the coefficients of zonally symmetric modes for possible round-off errors. CMODE0 is in a program library deck of the same name.

All $J=1$ modes are set to zero, except for the $I=1$ mode, whose imaginary part only is set to zero. These modes describe horizontal mean fields (described in Section 3.2), but only that for $I=1$ has any physical meaning.

The imaginary parts of all zonally-symmetric rotational modes are set to zero.

The zonally-symmetric gravity modes designated by ITYPE=1 are not independent of those designated by ITYPE=2. Rather, aside from round-off errors, for each value of $L=1, \dots, NLEV$, and $N=2, \dots, NMAXP1$, they should satisfy

$$Y_1 = Y_2^*,$$

where the subscript refers to the value of ITYPE, and indices L and N are implied. The asterisk denotes a complex conjugate. The zonally-symmetric gravity modes have been scaled such that the above condition is the one that insures reality. This has been described in Part I. CMODEO ensures this condition by the sequence of operations:

$$\text{Set } y_2 = 1/2(y_2 + y_1^*)$$

$$\text{Set } y_1 = y_2^*$$

for all values of $J = INDEX(1,N)$, $N=2, \dots, NMAXP1$. The value 1 as the first argument for INDEX indicates zonally symmetric modes ($m=0;M=1$).

5.6 The routine DEGREE

The routine DEGREE determines the sum (E) of the variances of all modes of a given equivalent depth and modal type. If the model were that linear model used to determine the normal modes (described in Vickroy (1983)), and if the vertical modes were orthogonal, then E would be the sum of kinetic plus available potential energy (per unit mass) contributed by these modes. E is calculated using Eq. 9 in Section 2.1. DEGREE is in a program library deck of the same name.

6. MODIFICATIONS TO THE MODEL

It is best to modify the model slightly when using it to initialize a dataset. Appropriate UPDATE modifications to the CFM, for the purposes of initialization, appear in Fig. 1b. These modifications only affect the disposition of data files at the completion of each iteration of the initialization scheme. In particular, they act to:

1. eliminate the dispose of the history tape to the mass-store device;
2. eliminate writing, copying, saving, and disposing of the save tape for restart or regeneration procedures;
3. eliminate writes by the model to \$OUT by setting the model variable NOUT to some otherwise unassigned unit number.

In this way unnecessary output and system time are not expended. Further discussion of Fig. 1b appears in Section 7.2.

7. A USER'S GUIDE TO THE NMS

The CRAY job instructions for executing a nonlinear initialization procedure are presented in four parts in Figs. 1a through 1d, respectively. These parts are: a CRAY JCL file; two CRAY UPDATE files for modifying model program libraries, an additional UPDATE file for modifying the normal-mode program library, and three files of data (primarily NAMELIST variables). Each part is described separately below, although all parts must be modified by the user for a particular job. Places where a user must supply values are noted in lower case. This section is written so that a user may read it apart from other sections and understand the software sufficiently to execute a nonlinear initialization. Execution of other tasks will require the reading of other sections of this report.

The following example shows how to structure a job that will execute a user-specified number of iterations of MACHENHAUER'S initialization scheme. Input are: a version of the CFM; a file of normal-mode structure functions; and a file of non-initialized data. None of these files is overwritten during execution. Output is a file of initialized data. The set of gravity modes initialized is defined by specifying both frequency and equivalent-depth cutoffs.

7.1. JCL statements

The CRAY JCL file for executing MACHENHAUER'S scheme is shown in Fig. 1a. The user must supply a CRAY job-card first (not shown). He may follow this with a DISPOSE, DN=\$OUT, DEFER, ... instruction for the disposing of printed output to the desired device. Following this, there is one file name, and one variable, that must be user-specified in the JCL file.

The JCL variable J1 must be set in the first statement. J1 is the number of iterations of the nonlinear initialization scheme, excluding the starting iteration. After the Nth iteration, diagnostics which describe the convergence of the initialization scheme at the (N-1)th iterate will be printed. Thus, convergence diagnostics of the final dataset will not be printed unless J1 equals the value of ITMAX+1 (ITMAX is described in Section 7.3). The printed diagnostics have been described in Sections 5.4 and 5.6.

A dataset name, written here as qqqqq2, must be specified in the next statement. This dataset contains radiation data input to the CFM. The particular dataset acquired must have truncation parameters compatible with the model truncation that the user will specify. It is possible that other such datasets may be required for some versions of the CFM and they should be acquired and assigned at this point.

There are three program libraries that are acquired in this JCL file. One is the normal-mode program library INITAL. The remaining two are program libraries for the model and radiation routines, specified in Fig. 1a as CCMZBA and CCMRDA, respectively. These may be replaced by other existing versions of the CFM routines if the UPDATE file in Fig. 1b is changed accordingly.

7.2 UPDATE of the model for nonlinear initialization

The next two CRAY files contain UPDATE modifications to the model's radiation program library, and the model program library, respectively. In the example presented in Fig. 1b, no modifications are made to the radiation library, and 3 sets of modifications are made to the model

library. The specific statement numbers referenced in Fig. 1b are those appropriate to the model program-library CCMZBA. The user should consult a listing if a different program library has been specified by the user in the JCL described in Section 7.1.

The first set of modifications are those labeled "Mods for running initialization of model." The purpose of these mods has been described in Section 6. This set of mods is not strictly necessary, but is highly recommended.

The second set of modifications are those which specify the model truncation. The truncation parameters in the NMS must agree with their corresponding values specified here, as described in Section 7.3. The model parameters are described in Sato et al. (1983). The values shown in Fig. 1b are those for a rhomboidal truncation at wavenumber $m = 15$.

The next modification redefines the output unit number so that several pages of output are not written to unit 6. In this way, no model output will be written to \$OUT, since this output is primarily irrelevant to the initialization procedure. This modification to the model is not strictly necessary, but is highly recommended.

Other modifications to the model, to make it agree with a desired version of the forecast model, may be included at this point. Those modifications which are irrelevant to the initialization procedure, or which are expected to produce insignificant changes in the initialization dataset, may be excluded from this initialization version of the model. For example, if the initialization is to be an adiabatic one, then modifications to the radiation program or diabatic portions of the model most likely would be entirely unnecessary in this version of the model.

7.3 UPDATE of the normal-mode software

The next CRAY file contains UPDATE directives for the normal-mode software (Fig. 1c). There are four sets of parameters to be defined by the user. Each set is described individually below.

Set 1 includes all the spectral and grid truncation parameters:

- NLEV the number of full sigma levels (equivalent to PLEV in the model).
 MMAX the maximum value of the zonal wavenumber m for the spectral truncation (equivalent to PTRM in the model).
 NMAX the maximum value of n for $m=0$, where n is the degree of an associated Legendre polynomial in the truncated spectral description of the data (equivalent to PTRN in the model).
 KMAX the maximum value of n for any m (equivalent to PTRK in the model).
 NLAT the number of Gaussian latitudes (Northern Hemisphere plus Southern Hemisphere; equivalent to POREC in the model).
 NLON the number of longitude points on any latitude circle (equivalent to PLON in the model).

These parameters are described in more detail in Section 3.1.

Set 2 includes all parameters which control the initialization procedure (additional NAMELIST variables are described below). These are:

- ITMAX the iteration number for the desired initialized dataset. $ITMAX < J1$ must be satisfied ($J1$ is described in Section 7.1).
 DTIME the time step (in units of seconds) used for the finite difference approximation of the time derivatives. DTIME must equal the product of the model NAMELIST variables NWTIME and DTIME. These latter two variables are described in Section 7.2.2.

LFIX the number of vertical modes to be initialized, ordered from largest equivalent depth to smallest. If the vertical mode index is not to be a criterion to determine which modes are initialized, set LFIX=NLEV.

CUTOFF Only gravity modes with periods (in hours) less than or equal to the value of CUTOFF will be initialized. If no such cutoff is desired, set CUTOFF=1.E10.

These parameters are also described in Section 5.1.

Set 3 includes the names of three permanent files:

FLVECT The name of the appropriate file of eigenvectors. This file, if not in existence, must be created using the software described in Vickroy (1983). This file must have truncation parameters identical to those defined here. The choice of a file also will be determined by the choice of basic state used to define the modes.

FLDATA The name of the input file containing data to be initialized. This file must be in CFM history-tape format, with the values of NLON, NLAT, and NLEV specified above.

FLOUT The name of the dataset that will contain the initialized data after execution by the initialization software. It will have the same format as the input data file.

Set 4 contains miscellaneous parameters:

ITYPES The number of types of modes stored on the file specified by FLVECT. Normally ITYPES=3, unless the truncation parameters are so large that only gravity-mode vectors will fit on the file, in which case ITYPES=2. If the user-specified value of ITYPES does

not agree with the value on the file FLVECT, execution will abort after the value on file has been printed.

ISYMT Equal to -1 if the modes are stored in a particular symmetric or antisymmetric pattern. For this case, the file FLVECT contains structure functions for the Northern Hemisphere only. In particular, this case occurs for modes defined using a linearization of the model about a basic state of rest. The alternative to ISYMT=-1 is ISYMT=1. If the user-specified value of ISYMT does not agree with the value on the file FLVECT, execution will abort after the value on file has been printed.

LVIRT A logical parameter equal to .TRUE. if the geopotential is to be calculated using virtual temperatures, and equal to .FALSE. if otherwise. It is best if this agrees with the model configuration.

IBUF A parameter equal to 2 if data eigenvectors are input using BUFFER IN statements, and equal to 1 if input using READ statements. IBUF=2 is recommended unless a storage or other problem is indicated.

These parameters are described further in Section 3.1.

7.4 Data input files

There are three files of data input. The first contains variables that control the starting iterate for the normal mode software. The next contains variables for input to the model. The last is similar to the first, except that it controls nonzero iterates.

7.4.1 Input for a starting iterate

The first data file contains the single NAMELIST, LIST1. This list is input to INITAL to control what type of initialization is executed, as well as which files (or units) are read or written. The LIST1 variables are described in detail in Sections 4.2.1 and 4.5.1.

For the example in Fig. 1d, it is assumed that the model accesses data input and saves history-tape output; i.e., that communication between the model and initialization programs is via permanent datasets. If default values of the NAMELIST variables are used (as denoted below), then only two variables need be specified. They are:

INLTYP = k, where k is an integer whose value is 0 or 1 in this case.

Both indicate this job step is a starting iterate suitable for Machenhauer initialization. A value of 0 indicates the initialization is to begin from the values of gravity-mode coefficients as determined from the analyzed data. A value of 1 indicates that the starting iterate is to be a linear initialization. In that case, all modes to be initialized are removed from the starting iteration.

PASWRD = 6Hwwwww, where wwwwww is a write protect password name. The same name applies to all permanent datasets. The value of PASWRD should be identical to that specified for the model output (Section 7.4.2).

All other variables may remain at their default values. These values are:

ISAVE = 1, which indicates that communication is via permanent datasets.

FTDIN = FLDATA, which indicates that coefficients are to be determined from the dataset of analyzed data with the name specified by the user-set parameter FLDATA (Section 7.2).

FTDOUT = FLOUT, which indicates that the dataset containing the starting iterate will be given the same name as that of the final (desired) output file. That name is specified by the user-set parameter FLOUT (Section 7.2).

IUNITO = 6, which indicates that information regarding the truncation and file names will appear in the output file.

FTDIN is not referenced in a starting iterate.

The remaining LIST1 variables are unit numbers. Their default values are set so as not to conflict with other unit numbers. However, upon execution of a starting iterate, IUNITE=IUNITB is specified, and if INLTYP=0 has been chosen, then IUNITD=IUNITB as well. The first relation states that only analyzed data may be modified in a starting iterate (no other data is yet in existence). The second relation states that the dataset for a starting iteration contains the unmodified, analyzed data.

7.4.2 Input to the model

For the version of the CFM used in this example, there are two sets of NAMELIST parameters and two sets of formatted data that serve as input to the model. The first list is named EXPDEF. The only variable in this list that need be changed from as it appears in Fig. 1d is NSWRPS which should equal the user specified value of PASWRD, as described in Sections 7.4.1 and 7.4.3. Default values of PASWRD in both the model and NMS are

identical. The first set of formatted data contains four labels that are written to Unit 6. In all other ways, they are irrelevant to the initialization. The second list is named NEWRUN. It is described below. The last set of formatted data may be left as is. For a detailed description of the model input, Sato et al. (1983) should be consulted.

For an adiabatic initialization, the list NEWRUN contains several variables that must be user-supplied. These are:

LTAPES=6Hqqqqq6 where qqqqq6 must be a user-specified name for the data file output by the CFM. Since ISAVE=1 has been specified (Section 7.4.1), this name must match that of the data file input to nonzero iterations of the initialization program (see below).

LABDAT=6Hqqqqq5 where qqqqq5 is the name of the data file input to the CFM. Since both ISAVE=1 has been specified and FTDOU=FLOUT by default (Section 7.4.1), this name must match that of the data file output by the initialization program as specified by the value of the parameter FLOUT (Section 7.3).

DTIME=tt., the time step for integration in the CCMOB, in units of seconds. A value of 60. is recommended.

NESTEP=i, where i is the number of integration steps of length DTIME. A value of 2 is recommended because it yields a more accurate balance (than i=1) by better accounting for the model centered differences.

NWTIME=i, where this i must be identical to that for NESTEP.

NLPHYS=.FALSE. turns off all physics (see Sato et al. (1983) for details).

For a diabatic initialization, in addition to `NLPHYS=.TRUE.`, values for other variables must be set as well. Sato et al. (1983) should be consulted in this case.

7.4.3 Input for higher iterates

The last file of data is similar to the first, except it specifies variables for execution of a nonzero iterate of Machenhauer's scheme (specified by `INLTYP=2`). The value of `PASWRD` is set identically to the previous value. Here, a unique name for the value of `FTDIN` must be specified in place of `qqqqq6`. This value must be identical to that specified for `LTAPES` (the model output file, described in Section 7.4.2). This file contains a forecast from which the time tendencies of the coefficients are determined using a finite difference approximation. The remaining variables have their default values.

8. A DEMONSTRATION OF THE NMS

A twenty-four hour forecast using a version of the CFM with NMS initialized data is presented as a demonstration of the NMS. The forecast model includes physical processes. In particular, radiation fields are calculated every 12 hours. The hydrostatic equation uses temperature, and not virtual temperature. The truncation is R15 with a global grid of 40 latitudes and 48 longitudes, and 9 levels in the vertical. The dataset is obtained from the ECMWF 3B FGGE analysis of 16 January 1979 at 0Z. The data are initialized using 1 linear followed by 4 nonlinear iterations of Machenhauer's scheme. All gravity modes with periods less than 24 hours are initialized. An adiabatic version of the model is used in that procedure.

The structures of the normal modes are described in Vickroy (1983), although for this case, the vertical modes of Temperton and Williamson (1981) have been used.

8.1 Convergence of an Iterated Solution

The diagnostic VAR, as described in Section 5.6, is presented in Fig. 2a. It is shown as a function of vertical mode index L and mode type for selected iterations. For each L , mode type R contains all rotational modes, excluding those which describes the mean temperature and pressure fields (the $J=1, I=1$ mode). The contribution to VAR by the latter is so dominant that its inclusion may render interpretations of the results difficult. For each L , mode type G contains all gravity modes, both eastward and westward. The labels X and G for this case represent the analysis (iteration 0) and initialized (iteration 4) values of VAR(G), respectively. VAR(R) is independent of iteration number.

The diagnostic BAL is presented in Fig. 2b in a format similar to that for VAR. The exception to the previous format is that values of BAL(G) evaluated after iterations 1 and 2 also are indicated in Fig. 2b, but only where they are substantially different from the final values.

The initialization substantially reduces BAL(G). By comparison, BAL(R) changes by less than 10% from its value at iteration 0 to its value after iteration 4. After the fourth iteration, values of BAL(G) are much less than corresponding values of BAL(R). These are the desired results. The fact that BAL(G) does not change significantly after the second iteration is a result of both the poor convergence properties of Machenhauer's scheme and the inclusion in BAL(G) of modes not being initialized (i.e., those with periods greater than 24 hours). For the present case, the latter effect is likely more significant. The present results suggest that three iterations of Machenhauer's scheme is sufficient when used with the 24-hour cutoff.

8.2 Comparison of Time Series of Forecasts

Two time series of forecast surface pressure for a randomly selected grid point in the United States are presented in Fig. 3. The specific point is that for 42.2N, 97.5W, in Nebraska. One time series is from an initialized forecast, and the other is from a forecast starting from the analysis (not initialized). The smoothing that results from initialization is apparent.

8.3 Verification of a Forecast

The initialized and analyzed height fields at 500mb for the Northern Hemisphere are presented in Figs. 4a and 4b, respectively. Detailed

analysis indicates that the mean difference in height is equal to 0.5 meters. The standard deviation of the difference is equal to 20.5 meters. Thus, the initialization procedure does not produce large changes in the 500mb geopotential field, as it should not if the forecast is to be accurate.

The one-day forecast and verifying height fields at 500mb for the Northern Hemisphere are presented in Figs. 5a and 5b, respectively. Detailed analysis of these fields indicates a mean forecast error of -15.2 meters with a standard deviation of 32.6 meters. These results, along with those of Section 8.2, suggest that the initialization procedure is performing satisfactorily.

APPENDIX A

List of Subroutines

Those subroutines explicitly appearing in the program library INITIAL are listed here. After a brief description of the routine, the name (underlined) of the program deck in which the routine may be found and the section in this report where the routine is described are noted.

- CMODE0: corrects roundoff errors in coefficients of zonally symmetric modes; CMODE0; 5.5
- DEGREE: calculates energy of normal modes of given type and equivalent depth; DEGREE; 5.6
- FILEIN: accesses input data files; FILEIN; 4.5.2
- FILOUT: saves and disposes output data files; FILOUT; 4.5.3
- FREQNC: computes cut-off frequency from period cut-off parameter; INITAL; 5
- FTNAME: a function that encodes local dataset names; INITAL; 4.5.1
- GRD2NM: sequences operations for transform from data to coefficients; GRD2NM; 4.3
- HOUGHR: the first step in the computation of data from coefficients; NM2GRD; 4.4
- HOUGHT: accumulates integrand for the transformation from data to coefficients (integration over latitude step); GRD2NM; 4.3
- HYDROS: computes "pseudo-geopotential" from temperature; GRD2NM; 4.3
- HYDRSR: computes temperature from "pseudo-geopotential;" NM2GRD; 4.4

IGTSEQ: function that returns sequence number of current job;
RPARMB.

INIT00: prepares coefficients for a starting iteration in which data
are not modified; INIT00; 5.2

INIT01: performs a linear initialization; INIT01; 5.3

INIT02: modifies coefficients for a non-zero iterate of Machen-
hauer's scheme; INIT02; 5.5

INITAL: main program; INITAL; 4.2

LOGPS: replaces values of surface pressure by their natural
logarithms; GRD2NM; 4.3

LOGPSR: exponentiates the natural logarithms of the surface pres-
sure; NM2GRD; 4.4

NM2GRD: sequences operations for transformation from coefficients to
data; NM2GRD; 4.4

RPARMA: reads some eigenvectors and transformation matrices; RPARMA;
3.2

RPARMB: reads header from data file; RPARMB; 3.3.1

RPOINT: reads pointers from header on CFM history tape; RPARMB;
3.3.1

SIGMAR: transforms from vertical-mode coefficients to sigma-level
data; NM2GRD; 4.4

SIGMAT: transforms from sigma-level data to vertical-mode coeffi-
cients; GRD2NM; 4.3

SURFPR: computes changes in surface pressure from changes in
"pseudo-geopotential;" NM2GRD; 4.4

VIRT: computes virtual temperature from temperature and specific
humidity; GRD2NM; 4.3

VIRTR: computes temperature from virtual temperature and specific humidity; GRD2NM; 4.4

WDATE: unpacks date and time which describe data; RPARMB.

WRHEAD: inserts information into header of data file; RPARMB; 4.2.4.

ZEROA: zeroes a vector; NM2GRD.

ZONALT: calls fast Fourier transform; GRD2NM; Appendix C

APPENDIX B

List of Variable Names

Almost all of the variables defined in the NMS code are listed here. Those not listed are either local to a particular routine and have an obvious function, or are equivalent to ones listed here and used as dummy arguments in subroutines. Most dummy arguments are listed here.

Following each name is a section number in the text where the variable is described.

B	3.3.1	GAUSWT	3.2
BAL	5.4	GLAT	3.2
COEF	3.4	GMATRX	3.2
COEF1	5.4	IBUF	4.6
COEF2	5.4	IFAX	Appendix C
CPMTRX	3.2,4.4	INC	Appendix C
CUTOFF	5.1	INDEX	3.2, Appendix D
DPS	4.4	ISYMT	3.1
DTIME	5.1	ITERAT	5.1
DZ	4.4	ITMAX	4.5
EDEPTH	3.2	ITYPES	3.1
FLDATA	4.5	IUNITA	4.5
FLOUT	4.5	IUNITB	4.5
FREQNC	5.1	IUNITC	4.5
FTDIN	4.5	IUNITD	4.5
FTDIN2	4.5	IUNITE	4.5
FTDOUT	4.5	IUNITF	4.5

IUNITO	4.5	NPS	3.3.1
JSIGN	4.2.2,4.3	NQ	3.3.1
JUMP	Appendix C	NSKIPR	4.2.4
KLAT	4.2.2	NT	3.3.1
KMAX	3.1	NTRUNC	3.1
LENHD	3.3.1	NU	3.3.1
LFIX	5.1	NU1	---
LIST1	5.1	NUVT	3.3.1
LVIRT	3.1	NV	3.3.1
MMAX	3.1	NV1	---
MMAX2	3.1	NVECTR	3.3.2
MMAXP1	3.1	NWORK	---
MTRUNC	3.1	NZ1	---
NBDATA	3.3.1	PASWRD	4.5
NBVECT	3.3.1	RGMTRX	3.2
NCOEF	3.4	ROMEGA	3.2
NLAT	3.1	RTBAR	3.2
NLEV	3.1	RZMTRX	3.2
NLEVP1	3.2	Q	3.3.1,4.3
NLEVP2	3.2	SEDPTH	3.1
NLON	3.1	SIGMA	3.2
NLONC	Appendix C	TRIGS	Appendix C
NLOND2	---	USCALE	3.2
NLONP2	3.3.1	VAR	5.4
NMAX	3.1	VECTR	3.3.2
NMAXP1	3.1	WORK	4.4, Appendix C

NMODES	3.1	Z	4.4
NPHIS	3.3.1	ZMATRX	3.2

APPENDIX C

The Fast Fourier Transform

The routine ZONALT within the NMS has only one function, that of calling the routine FFT991. The latter routine performs a number of simultaneous real/half-complex Fourier transforms (ITRANS = -1) or corresponding inverse transforms (ITRANS = +1). FFT991 is optimized for use on the CRAY-1. Previous to calling this routine, the two set-up arrays IFAX and TRIGS must be determined by calling the routine FFTFAX.

FFT991 uses CRAY vector processing to compute simultaneously up to 64 transforms of independent strings of data. The length of each transform is NLON, which must have a factor of 2, and may have factors of powers of only 2, 3, and 5. A value of INC = 1 indicates that grid-point data are stored consecutively. JUMP = NLONP2 is the increment between the first elements of successive sigma levels. The data for T, U, and V must be stored successively unless the call to FFT991 is modified.

APPENDIX D

The Ordering of Modes

The modes are ordered to exploit both their symmetry properties and vectorization of the code on the CRAY-1 computer.

If `ISYMT=-1` is specified by the user as a parameter in the main program, then the modes are assumed to have the symmetry properties described below. The file containing the eigenvectors also contains a value for `ISYMT`. If that value and the one specified by the user do not agree, an error message is printed and execution then terminates by a call to `ABORT`. If `ISYMT=1` is specified by both the user and the value on file, then eigenvector components for northern and southern latitudes must be read individually (alternating northern, southern; starting from the northernmost, and working equatorward).

If `ISYMT=-1`, then the eigenvectors specified by `ITYPE=1` or `2` are ordered so that, for each zonal wavenumber m , they alternate symmetric, antisymmetric, starting with symmetric. In this case, the ordering for `ITYPE=3` is similar, except starting with antisymmetric. The index `ITYPE` is described in Section 3.2.

The array `INDEX` describes the ordering of the modes ($J=1, \dots, NMODES$), given the two arguments M and N . M is the zonal wavenumber of the mode, augmented by 1 (i.e., $M = m+1$; $m = 0, \dots, MMAX$). N is the meridional scale ($N=1, \dots, NTRUNC(M)$). Smaller values of N denote larger scales. For gravity modes, $N-1$ denotes the number of zero crossings of the eastward velocity field between the poles. For rotational modes, this number is given by N (see Kasahara, 1976). This ordering of the modes is analogous to that of the Legendre polynomials in the CFM (Williamson, 1983).

The arrays MTRUNC and INDEX have been generated by using the following FORTRAN algorithm:

```

MLAST1 = 0
DO 2 N=1,NMAXP1
MLAST = MMAXP1
IF(N.GT.KMAX-NMAX+1)
*      MLAST = MMAXP1+KMAX-NMAX-N
MTRUNC(N) = MLAST
DO 1 M=1,MLAST
1 INDEX(M,N) = M+MLAST1
2 MLAST1 = MLAST1+MLAST

```

Examples of the ordering INDEX for MMAX=3 triangular truncation (T3) and rhomboidal truncation (R3) appear in Figs. 6a and 6b respectively. The abscissa is the zonal wavenumber $m = M-1$. The ordinate is the index $n' = N+M-2$ (if the modes were instead Legendre polynomials, m and n' would be their order and degree, respectively). The corresponding values of INDEX are indicated.

References

- Kasahara, A., 1976: Normal modes of ultralong waves in the atmosphere. Mon. Wea. Rev., 104, 669-690.
- Machenhauer, B., 1977: On the dynamics of gravity oscillations in a shallow water model, with applications to normal mode initialization. Contrib. Atmos. Phys., 50, 253-271.
- Sato, R. K., L. M. Bath, D. L. Williamson, and G. S. Williamson (1983): Users' guide to the NCAR CCMOB. NCAR Technical Note, NCAR/TN-211+IA, 133 pp.
- Temperton, C, and D. L. Williamson, 1981: Normal mode initialization for a multilevel grid-point model. Part I: linear aspects. Mon. Wea. Rev., 109, 729-743.
- Vickroy, J., 1982: Normal modes of the NCAR community forecast model. NCAR Technical Note (in preparation).
- Williamson, D., 1983: Description of NCAR community climate model (CCMOB). NCAR Technical Note, NCAR/TN-210+STR, 88 pp.

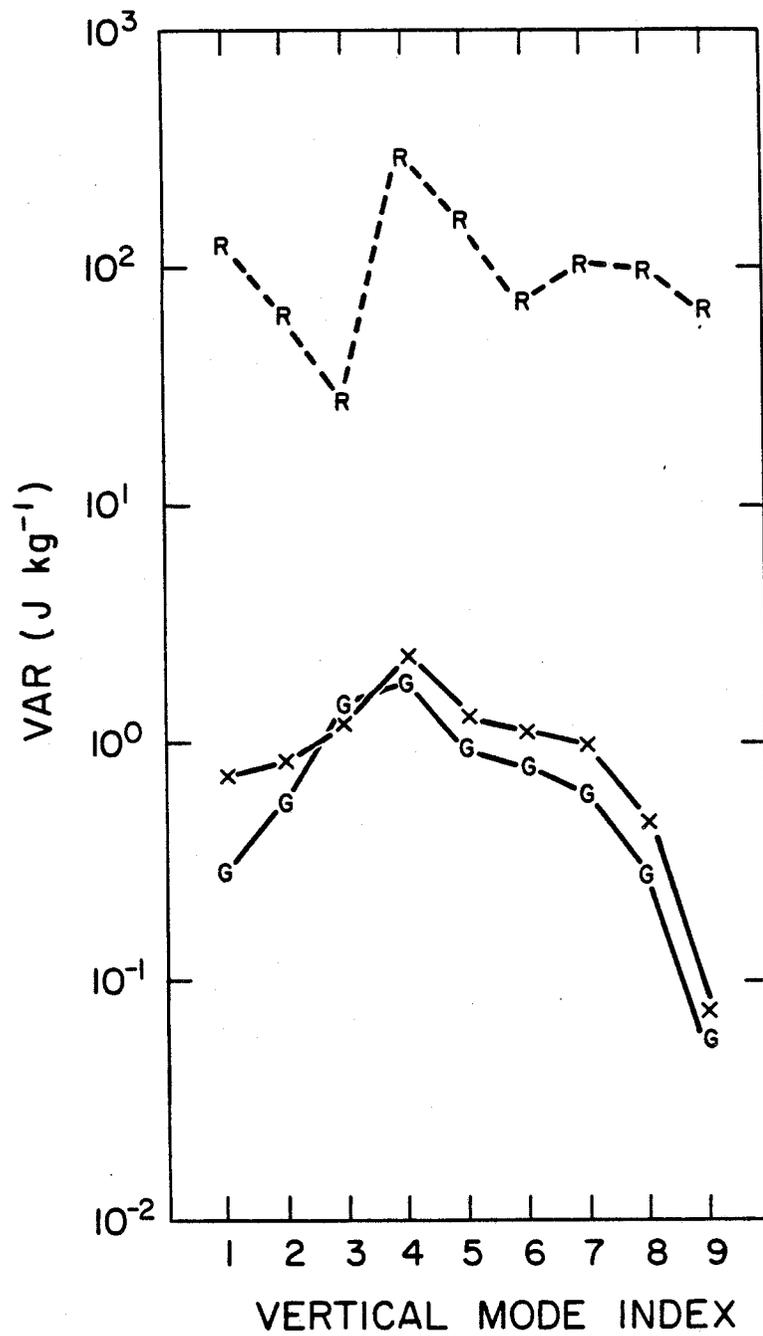


Fig. 2a The summed variances (VAR) of model coefficients. R indicates values for rotational modes. X indicates values summed for eastward and westward gravity modes, before initialization. G indicates the latter values after initialization.

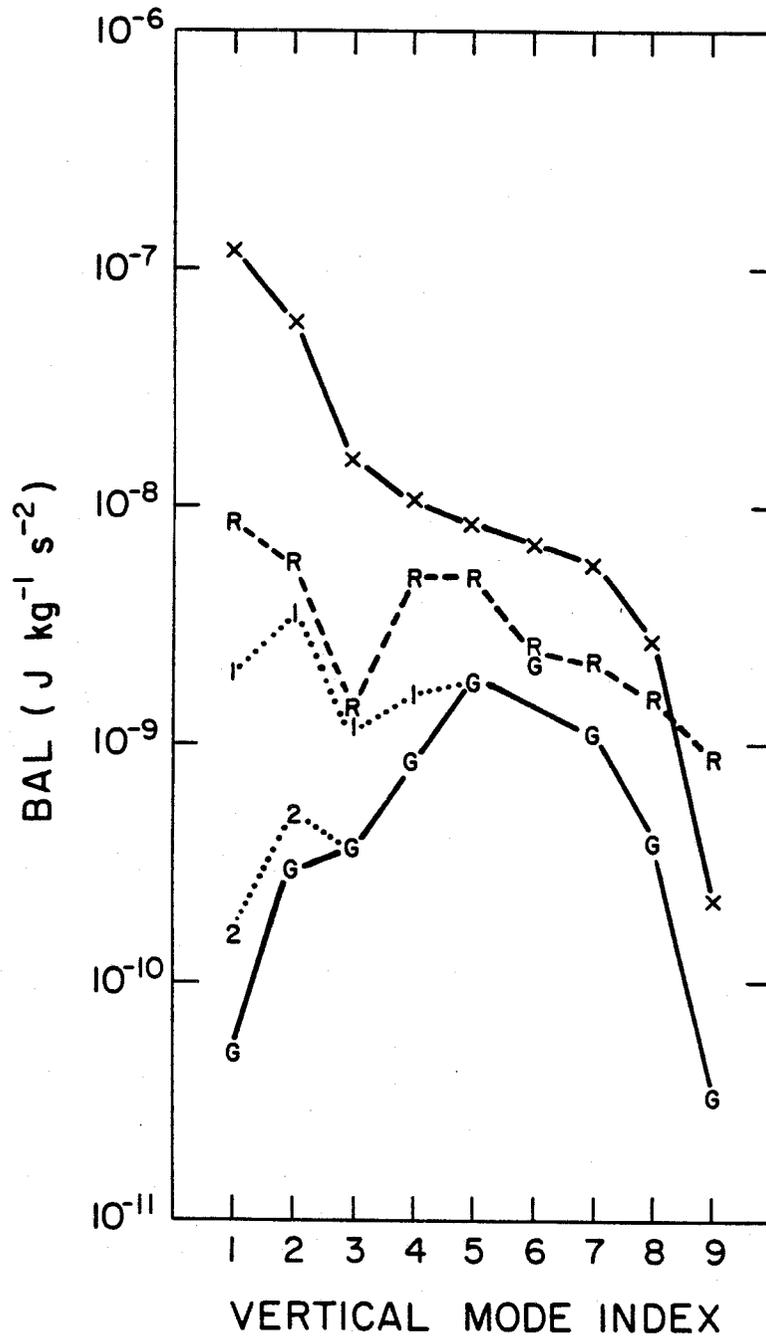


Fig. 2b Same as Fig. 2a, except for the variances (BAL) of the time tendencies of coefficients. The points labeled 1 and 2 indicate values of BAL for gravity modes after the 1st and 2nd iterations, respectively. If an iteration label is not indicated explicitly, then BAL for gravity modes or rotational modes is indicated by the label G or R, respectively.

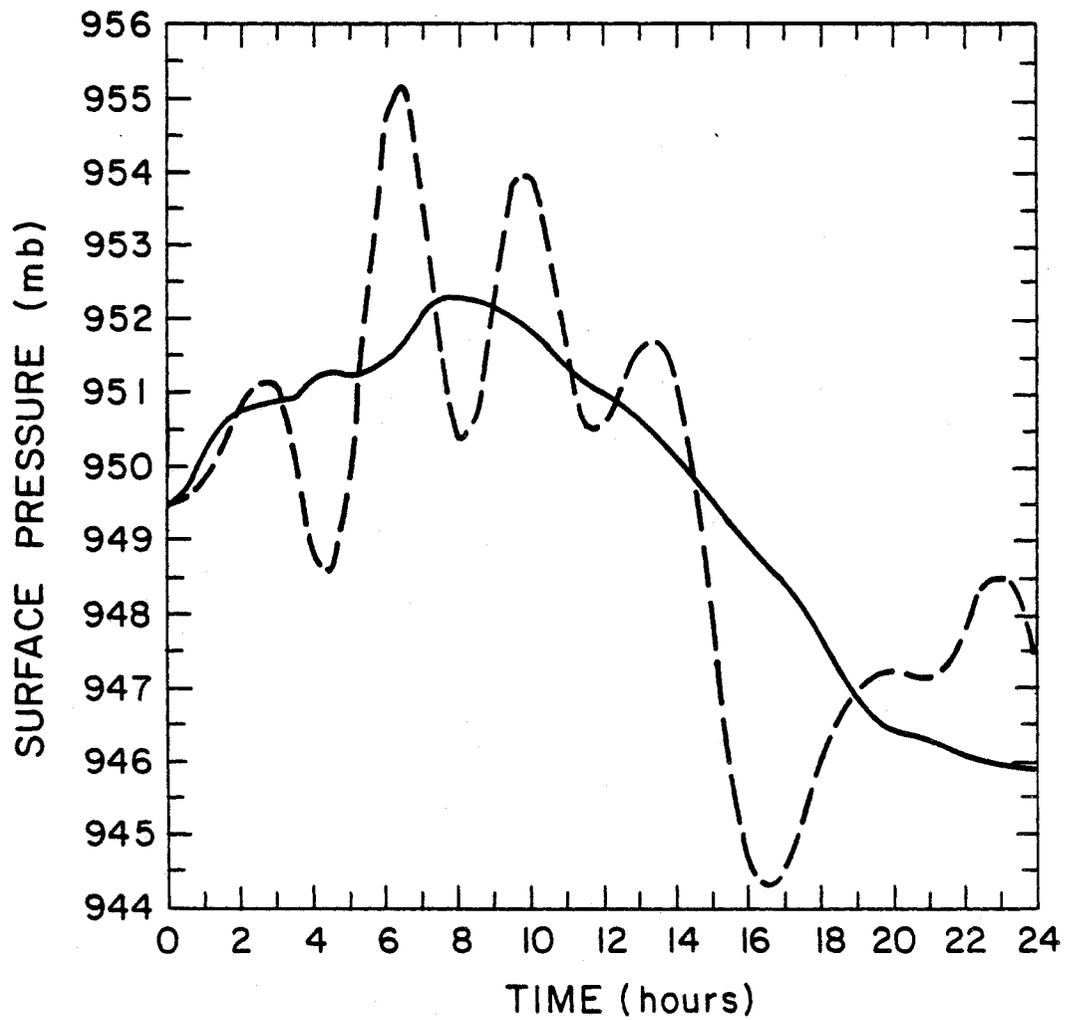


Fig. 3 Time series of the surface pressure at 43.6°N, 97.5°W for forecasts starting from analysis (dashed line) and initialized (solid line) data.

```

*
*           SET NUMBER OF ITERATIONS J1
SET(J1=j)
*           SPECIFY RADIATION DATA FILE
ACQUIRE.DN=FT02.PDN=qqqqq2.MF=TB.
*
*           END OF USER CHANGES IN THE JCL
*           ACQUIRE MODEL LIBRARY, UPDATE IT, AND CREATE LOAD MODULE.
ACQUIRE.DN=PL.PDN=CCMRDA.MF=TB.DF=TR.
UPDATE.P=PL.F.*=$.S=SR.C=0.I=0.NR.
REWIND.DN=SR.
RELEASE.DN=PL.
ACQUIRE.DN=$NPL.PDN=CCMZBA.MF=TB.DF=TR.
UPDATE.F.P=$NPL.*=$.C=0.N=NPL.
UPDATE.P=NPL.F.*=$.
CFT.I=$CPL.B=MODEL.OFF=CST.
*           UPDATE NORMAL MODE ROUTINES AND RUN STARTING ITERATION.
ACQUIRE.DN=BALANC.PDN=INITAL.MF=TB.
UPDATE.P=BALANC.C=CINITL.*=$.F.ID.ED.
CFT.I=CINITL.B=BINITL.L=0.
LDR.DN=BINITL.
*           START LOOP
LOOP.
  EXITLOOP(J1.EQ.0)
  REWIND.DN=FT02.
  REWIND.DN=$IN.
  SKIPF.DN=$IN.NF=4.           SKIP TO MODEL INPUT FILE.
  LDR.DN=MODEL.
  SKIPF.DN=$IN.NF=1.         SKIP TO INITIALIZATION INPUT FILE.
  LDR.DN=BINITL.
  SET(J1=J1-1)
ENDLOOP.
EXIT.
\EOF

```

Fig. 1a CRAY JCL structure for a nonlinear initialization.
(The user must substitute values for j and qqqqq2.)

```

$/ MODS TO RADIATION PROGRAM LIBRARY GO HERE
$READ SR
$/ END OF MODS TO RADIATION PROGRAM LIBRARY
$/ -----
\EOF
$/ MODS FOR RUNNING INITIALIZATION OF MODEL
$ID INITL1
$/
$/ ELIMINATE DISPOSE OF HISTORY TAPE
$D SAVDISP.24,47
    CALL RELEASE(IERR,2HDN,KOPY1)
    WRITE (INOUT,91) LTAPE,IPASWD
    91 FORMAT('O***** VOLUME ',A8,' SAVED ONLY      PASSWORD=''.A8)
$/
$/ ELIMINATE WRITE TO SAVE TAPE
$D STPN.118
$/
$/ ELIMINATE COPY OF SAVE TAPES
$D LINMS.230
    GO TO 270
$/
$/ ELIMINATE SAVE AND DISPOSE OF SAVE TAPE
$D LINMS.254
$/
$/ RELEASE UNIT 13
$I DUMMIES.61
    IFAIL=1
    CALL RETURN(IFAIL,13)
$/
$/ -----
$ID NOWRITE
$/
$/ MOD TO REMOVE PRINTED OUTPUT
$D COS1.29
    NPRINT=67
$/
$/ -----
$/ END OF MODS FOR INITIALIZATION
$/ -----
$ID RESMODS
$/
$/ MODS FOR CHANGING RESOLUTION
$/
$D PARAMS.13,14
    S PTRM=15,PTRN=15,PTRK=30,PLEV=9,
    T PLON=48,POREC=40,
$D PARAMS1.3,4
    PARAMETER (JG=20,JF=48,JG2=40,JCF=24)
    PARAMETER (MLOCD=16)
$D SETUP.67
C
C MHS IS RESET TO MATCH DTIME=1800 AND IRAD=24
    MHS=24
$/
$/ END OF ID RESMODS
$/ -----
$/ ALL OTHER MODS TO MODEL GO HERE
$/
$/ -----
\EOF

```

Fig. 1b CRAY UPDATE of the model for nonlinear initialization.

```
$ID USER
$/
SD INITIAL.11.18          MODS FOR INITIALIZATION ROUTINE FOLLOW
  PARAMETER (NLEV=11.MMAX=mm.NMAX=nn.KMAX=kk.NLAT=na.NLON=no)
  PARAMETER (ITMAX=j.DTIME=120.LFIX=lf.CUTOFF=cc.)
  PARAMETER (FLVECT=6Hqqqqq3.FLDATA=6Hqqqqq4.FLOUT=6Hqqqqq5)
  PARAMETER (ITYPES=3.ISYMT=-1.LVIRT=.FALSE.IBUF=2)
$/
$/ -----
\EOF
```

Fig. 1c CRAY UPDATE of the normal-mode software (lower case letters indicate where values must be substituted).

```

$LIST1          : list for input to starting initialization program
INLTYP=0        : =1 for linear init. to start. =0 otherwise
PASWRD=6Hwwwww. : set wwwwww identically on two lines below
$
\EOF
$EXPDEF         : first list for input to model program
NSVSN=6HDUMMYF. : save dataset not referenced in initialization
NSREST=0
NSBUFS=2H44.
NSWRPS=6Hwwwww. : set password identically to that above
$
CARD1. MACHENHAUER INITIALIZATION
CARD2. CFM
CARD3. ROTATIONAL MODES FIXED
CARD4. CHOSEN GRAVITY MODES BALANCED
$NEWRUN         : second list for input to model program
LTAPES=6Hqqqqq6. : set this to some (unique) dummy file name
LABDAT=6Hqqqqq5. : set this identically to FILOUT in program INITAL
MFILT=2.
MFILS=2.
DTIME=60..     : set this to some small time. like 60. (sec)
NESTEP=2.      : set DTIME parameter in INITAL to =NESTEP*DTIME
NWTIME=2.      : set equal to NESTEP
IRAD=24.
DIF=2.5E5.    : set diffusion coef. in diabatic initialization
EPS=0.06.
NLPHYS=.FALSE.. : set .TRUE. if diabatic initialization
$
INITIALIZATION OF DATA
00
\EOF
$LIST1          : list for input to initialization program
INLTYP=2.       : set =2. since nonlinear Machenhauer initial.
PASWRD=6Hwwwww. : set password identically to those two above
FTDIN=6Hqqqqq6. : set identically to value of LTAPES above
$
\EOD

```

Fig. 1d Data input for nonlinear initialization. (There are three files: the first for creating a starting iterate for Machenhauer's scheme; the second for making a 120-second forecast with the model; the last for executing the next iteration of Machenhauer's scheme.)

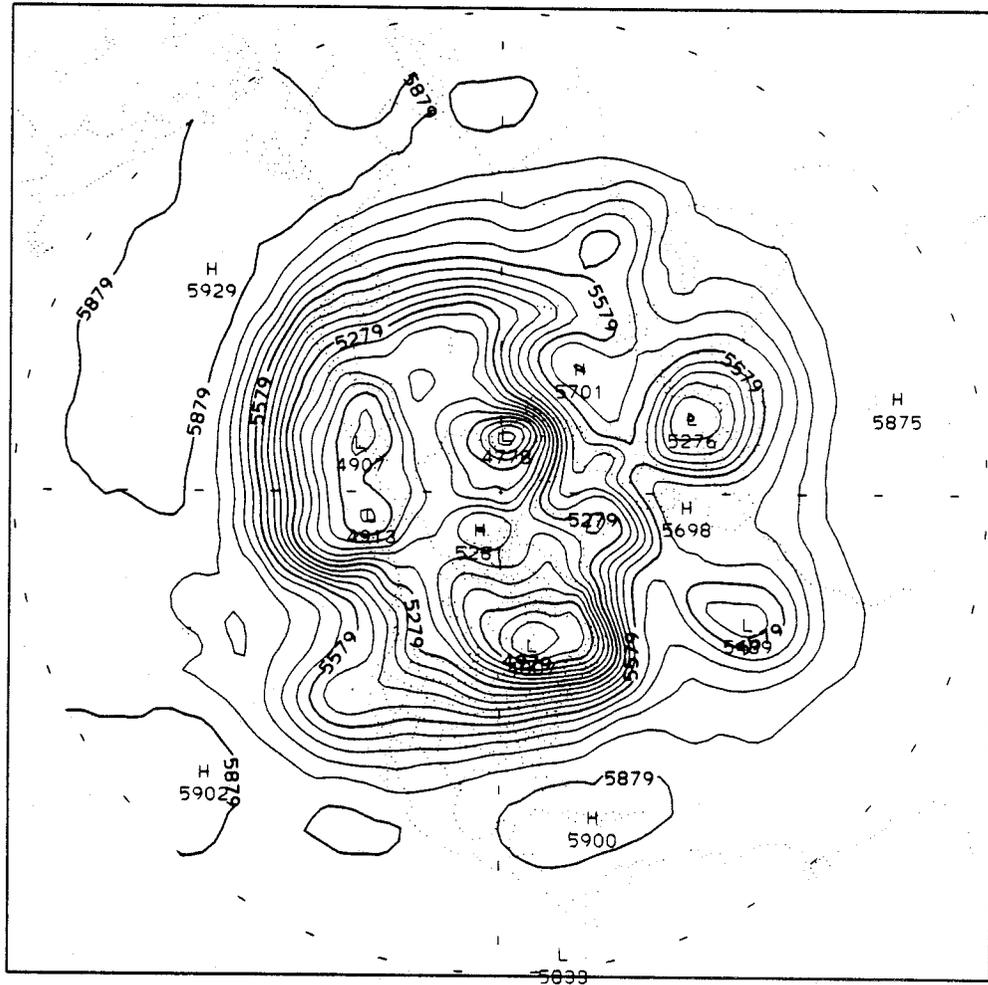


Fig. 4b The analyzed 500 mb height field for the Northern Hemisphere on 16 January 1979 at 02, for comparison with Fig. 4a.

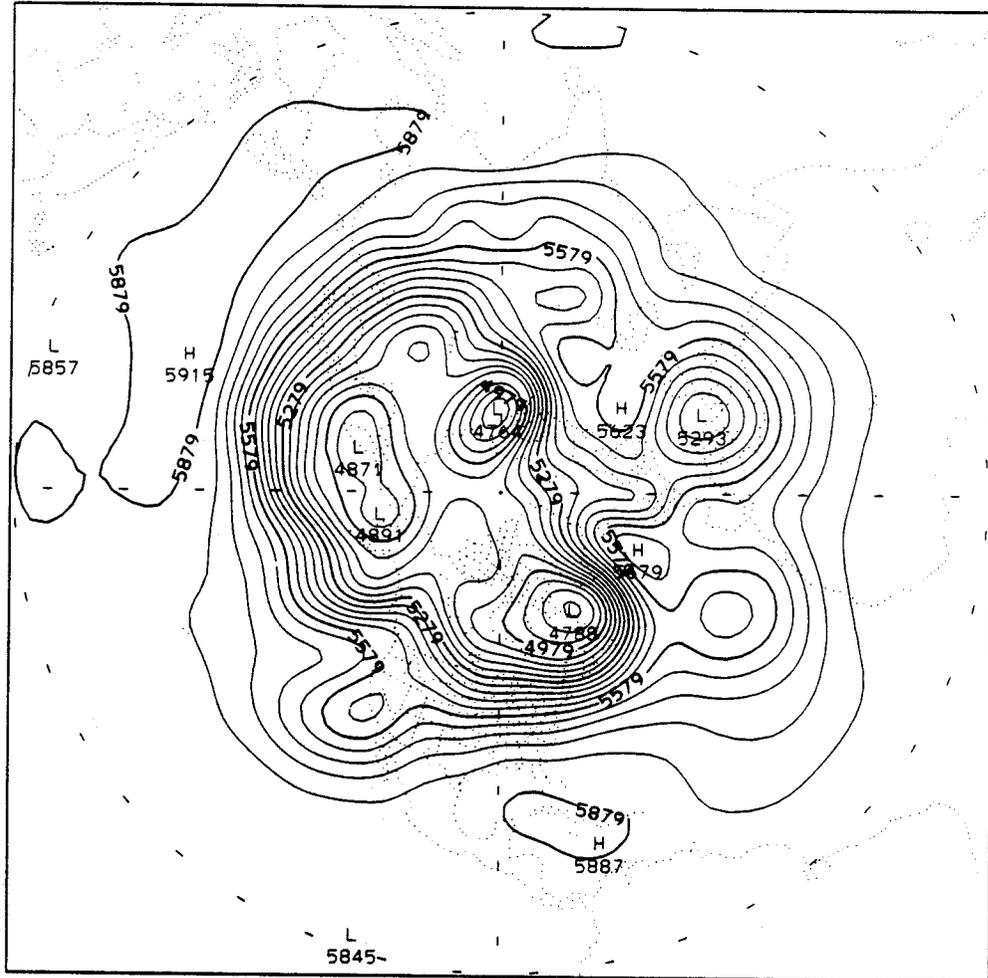


Fig. 5a The forecast 500 mb height field for the Northern Hemisphere after 24 hours (verifying time: 17 January 1979 at 0Z), starting from initialized data.

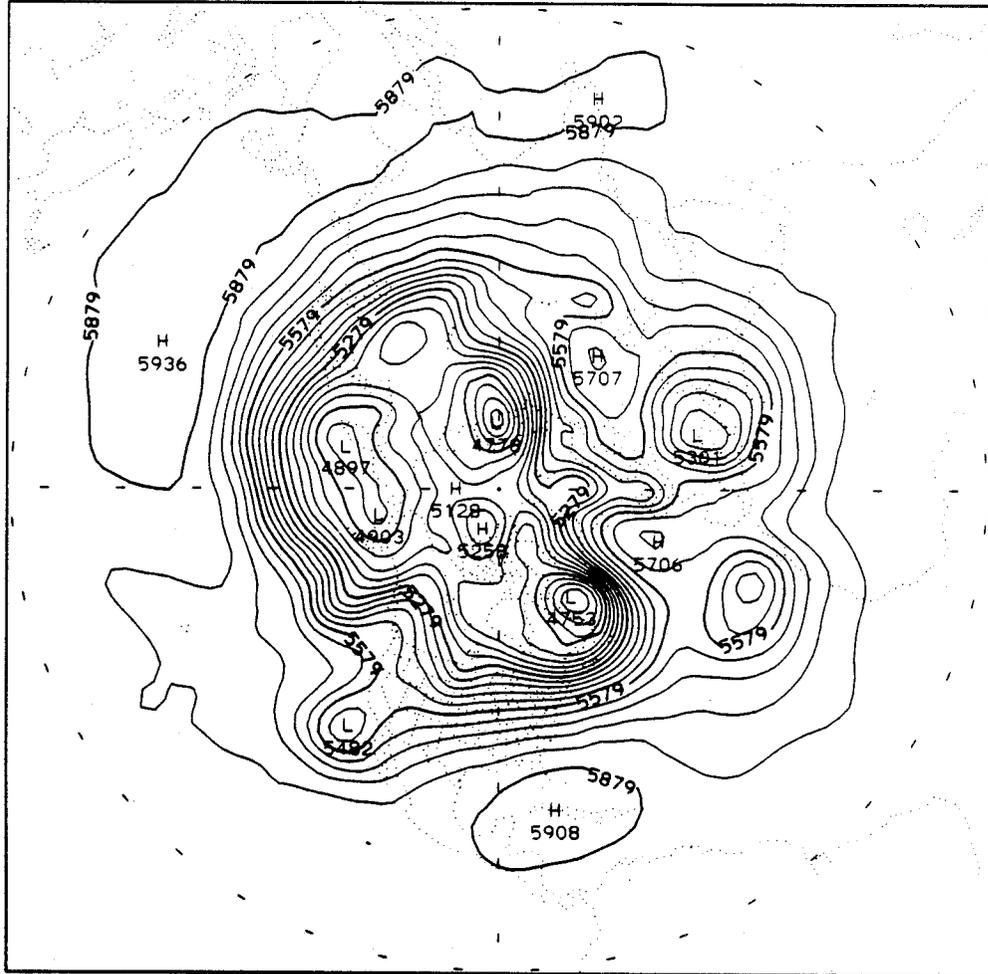


Fig. 5b The analyzed (verifying) 500 mb height field for the Northern Hemisphere on 17 January 1979 at 0Z, for comparison with Fig. 5a.

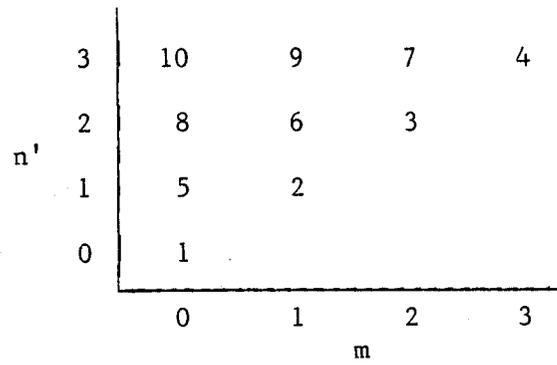


Fig. 6a Values of INDEX as functions of $m=M-1$ and $n'=N+M-2$ for T3 truncation.

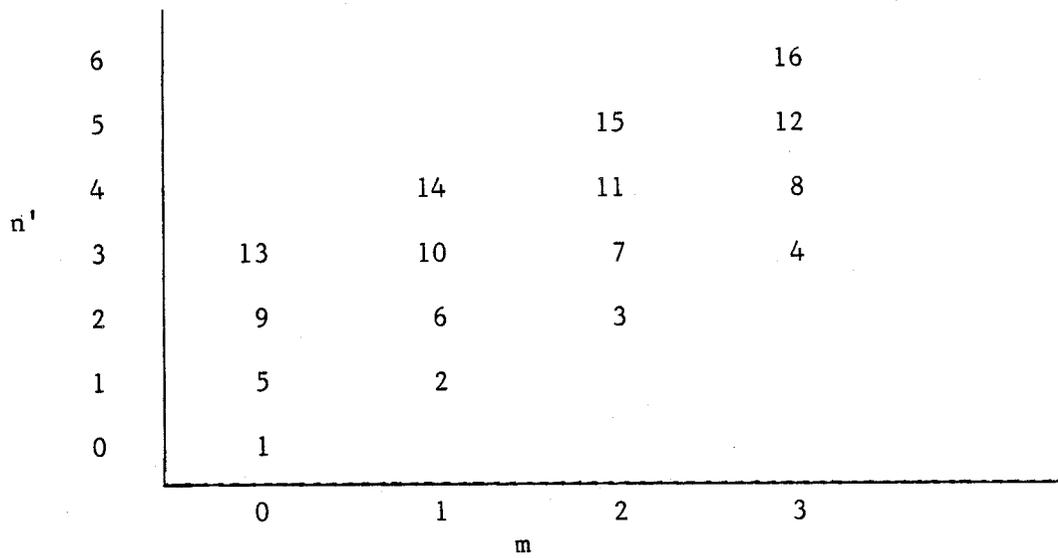


Fig. 6b Values of INDEX as functions of $m=M-1$ and $n'=N+M-2$ for R3 truncation.