General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)
Documentation of the
Fourth Order Band Model

Eugenia Kalnay-Rivas

David Hoitsma
Sigma Data Services Corp.

November 1979

Modeling and Simulation Facility
Laboratory for Atmospheric Sciences
NASA/Goddard Space Flight Center
Greenbelt, MD 20771
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>i</td>
</tr>
<tr>
<td>I. Introduction</td>
<td>I-1</td>
</tr>
<tr>
<td>II. Primitive Equations of Motion</td>
<td>II-1</td>
</tr>
<tr>
<td>Derivation of the Equations at the Poles</td>
<td>II-3</td>
</tr>
<tr>
<td>Computation of the Horizontal Pressure Gradient as suggested by N. A. Phillips</td>
<td>II-7</td>
</tr>
<tr>
<td>III. Finite Difference Variables and Grid</td>
<td>III-1</td>
</tr>
<tr>
<td>Periodic Filtering of Short Waves</td>
<td>III-3</td>
</tr>
<tr>
<td>IV. Boundary Conditions</td>
<td>IV-1</td>
</tr>
<tr>
<td>V. Finite Difference Equations</td>
<td>V-1</td>
</tr>
<tr>
<td>The Zonal (U) Momentum Equation</td>
<td>V-1</td>
</tr>
<tr>
<td>The Meridional (v) Momentum Equation</td>
<td>V-2</td>
</tr>
<tr>
<td>The Thermodynamic Energy Equation</td>
<td>V-3</td>
</tr>
<tr>
<td>The Moisture Equation</td>
<td>V-4</td>
</tr>
<tr>
<td>The Pressure-Tendency Equation</td>
<td>V-5</td>
</tr>
<tr>
<td>VI. The Forecast Equations at the Poles</td>
<td>VI-1</td>
</tr>
<tr>
<td>Zonal (U) Momentum Equation (Poles)</td>
<td>VI-3</td>
</tr>
<tr>
<td>Meridional (V) Momentum Equation (Poles)</td>
<td>VI-3</td>
</tr>
<tr>
<td>The Thermodynamics Energy Equation (Poles)</td>
<td>VI-4</td>
</tr>
<tr>
<td>The Moisture Balance Equation (Poles)</td>
<td>VI-4</td>
</tr>
<tr>
<td>The Pressure Tendency Equation (Poles)</td>
<td>VI-5</td>
</tr>
<tr>
<td>The Vertical Velocity Equation (Poles)</td>
<td>VI-5</td>
</tr>
<tr>
<td>VII. Diagnostic Equations (ϕ,ω,p)</td>
<td>VII-1</td>
</tr>
<tr>
<td>VIII. The Time Differencing Scheme</td>
<td>VIII-1</td>
</tr>
<tr>
<td>IX. Documentation of the Code (Preliminary)</td>
<td>IX-1</td>
</tr>
<tr>
<td>X. Flow Charts</td>
<td>X-1</td>
</tr>
<tr>
<td>Appendix A Reprints of Three Papers</td>
<td>A-1</td>
</tr>
<tr>
<td>A Fourth-Order Forecasting Model</td>
<td>A-1</td>
</tr>
<tr>
<td>The 4th Order GISS Model of the Global Atmosphere</td>
<td>A-3</td>
</tr>
<tr>
<td>The Effect of Accuracy, Conservation and Filtering on Numerical Weather Forecasting</td>
<td>A-16</td>
</tr>
<tr>
<td>Appendix B Program Listing</td>
<td>B-1</td>
</tr>
</tbody>
</table>
Preface

We have decided to compile a preliminary documentation of the new GLAS Fourth Order General Circulation Model. The present documentation has not been subjected to a careful editing process; we hope that its possible usefulness will compensate for some of its defects. The model dynamics (COMP0, COMP1 and COMP2) is still undergoing minor improvements, especially in the time differencing scheme which we hope will improve its efficiency. The "physics" routine (COMP3) has not been documented because it is being thoroughly revised. The present version of COMP3, similar to the one used in the GLAS/GISS models (see the documentation by Tsang and Karm), with modifications introduced by Y. Sud (1979) is included in the code. Criticisms and suggestions for improvement will be readily appreciated, since a final documentation will be prepared in 1980.

We are very grateful to all the people that have helped us generously. In particular, Dr. N. Rushfield had a major impact in the process of making the model operational. W. Connelly, D. Edelmann, D. Han, S. Breining, P. Anolick and M. Almeida were very helpful in the development of the model. The documentation was expertly and cheerfully typed by S. Mathis; D. Edelmann and D. Rosen have also cooperated in its compilation. We want to express our special gratitude toward Dr. Y. Sud, who offered us generously both his advice and his time in the development of the "physics" routine, and most especially to Dr. M. Halem without whose many useful suggestions, constant encouragement, and long patience, this work would not have been finished.

Eugenia Kalnay-Rivas

November 1979
I. Introduction

The band fourth order model is a GCM which uses quadratically conservative, fourth order horizontal space differences on an unstaggered grid and second order vertical space differences with a Matsuno (forward-backward) or a smooth leapfrog time scheme to solve the primitive equations of motion.

This program numerically solves these equations one latitude band at a time which greatly reduces the amount of computer core storage needed to run the program. It also uses the same variable names, order of computations, I/O, post-processing as the standard second order GCM. Appropriate modifications have been made for the fourth order differences and leapfrog scheme. (See the 1978 Goddard Modeling and Simulation Research Review for an overview of the fourth order band model.)

The main feature of this model is that fourth order approximations are used for all the horizontal derivatives. The derivative $\frac{\partial q}{\partial x}$ is approximated by

$$\frac{4}{3} \left( \frac{q(x+\Delta x) - q(x-\Delta x)}{2\Delta x} \right) - \frac{1}{3} \left( \frac{q(x+2\Delta x) - q(x-2\Delta x)}{4\Delta x} \right)$$

and the derivative $\frac{\partial q^T}{\partial x}$ by

$$\frac{4}{3} \left[ \frac{(T(x)+T(x+\Delta x))(q(x)+q(x+\Delta x)) - (T(x)+T(x-\Delta x))(q(x)+q(x-\Delta x))}{4\Delta x} \right]$$

$$- \frac{1}{3} \left[ \frac{(T(x)+T(x+2\Delta x))(q(x)+q(x+2\Delta x)) - (T(x)+T(x-2\Delta x))(q(x)+q(x-2\Delta x))}{8\Delta x} \right]$$

I-1
The second approximation is derived by averaging the flux $qT$ to yield a conservative form of the dynamic equations. Note that if $T$ is equal to 1 the second equation reduces to the first.

The primary variables are the horizontal components of the wind velocity, $W=(u,v)$, the temperature, $T$, the specific humidity, $q$, and the shifted surface pressure, $\pi$, ($\pi=\rho_s-\rho_{top}$, $\rho_{top}=10$ mb).

The secondary variables are the geopotential, $\phi$, the vertical wind velocity, $\dot{\phi}$, and the pressure, $p$.

The following pages give the differential equations of motion for the GCM model with the initial and boundary conditions. This is followed by the equations with the corresponding fourth order approximations which use the same notation as the current second order model. A complete description of the primitive equations with the $\sigma$ coordinate system is found in the Arakawa UCLA notes (1976).
II. Primitive Equations of Motion

1.2. Horizontal momentum equations

\[ \frac{d\pi}{dt} + \frac{dV}{dt} = \frac{\partial}{\partial t} (\pi V) + \frac{\partial}{\partial \sigma} (\pi \sigma V) \]

\[ = -\pi \nabla \phi - \pi \sigma \frac{RT}{P} \nabla \pi - (f + \frac{u \tan \phi}{a}) \mu x \pi + \eta \pi \]

3. Continuity equation

\[ \left( 3.1 \right) \frac{3 \pi}{3 t} + V \cdot (\pi V) + \frac{\partial}{\partial \sigma} (\pi \sigma V) = 0 \]

or

\[ \left( 3.2 \right) \frac{3 \pi}{3 t} = -\int_{0}^{1} V \cdot (\pi V) d\sigma = -\int_{0}^{1} \pi V d\sigma \]

4. Equation of state

\[ \alpha = \frac{RT}{p} \]

5. First law of thermodynamics

\[ \frac{3 \pi T}{3 t} + V \cdot (\pi V) + \frac{\partial}{\partial \sigma} (\pi \sigma T) = \frac{\pi \omega}{C_P} + \frac{\pi Q}{C_P} \quad (\omega = \frac{\partial p}{\partial t}) \]

From \( \phi = T/p^k \), \( p = p \tau + \sigma \pi \), \( \omega = \sigma \pi + \sigma \sigma \), \( \pi = \frac{3 \pi}{3 t} \mu x \nabla \pi \), \( k = R/C_P \) we get

\[ \frac{3 \pi \sigma T}{3 \sigma} = p^k \frac{3 \pi \sigma \theta}{3 \sigma} + \frac{\pi \sigma T}{C_P} \]

Replacing in 5,

\[ \frac{3 \pi T}{3 t} + V (\pi V) + p^k \frac{3 \pi \delta \theta}{3 \sigma} = \frac{\pi \sigma k T, \pi \sigma}{p \left( \frac{3 \pi}{3 t} + V \cdot V \pi \right) + \frac{\pi Q}{C_P}} \]

6. Humidity equation

\[ \frac{3 \pi \sigma}{3 t} + V \cdot (\pi \sigma q) = 0 \]

7. Hydrostatic equation

\[ \frac{3 \Phi}{3 p} = -C_P \theta \]

(from \( \frac{3 \Phi}{3 \phi} = -p = -\frac{1}{\alpha} \))

II-1
Of the variables \( \pi, u, v, T, q, \phi, \alpha, \sigma \) we update the 5 primary variables \( \pi, u, v, T, \) and \( q \) using equations 1, 2, 3, 2, 5 and 6. From equations 3.1, 4, and 7 we can obtain \( \phi, \alpha, \) and \( \sigma \) which are our secondary variables. Note that \( p = p_{\pi} + p_{\text{top}}. \)

Sea level pressure (used only in the smoothing routine SSMAP)

Hydrostatic eq. \( \frac{\partial p}{\partial \phi} = -p = -\frac{1}{\alpha} = -\frac{p}{RT} \therefore \log \left( \frac{\text{SLP}}{p} \right) = -\int_{0}^{\phi} \frac{d\phi}{RT} \)

\( \therefore \text{SLP} = p(\sigma=1) \exp \left( \frac{\phi}{RT} \right) \)
DERIVATION OF THE EQUATIONS AT THE POLES

Consider the continuity equation

\[ \frac{\partial \pi}{\partial t} + \nabla \cdot (\pi \mathbf{v}) + \frac{\partial \pi \sigma}{\partial \sigma} = 0 \]

coupled with a conservation equation \( \frac{\partial T}{\partial t} = S \) which can be expanded into

\[ \frac{\partial \pi T}{\partial t} + \nabla \cdot (\pi \mathbf{v} T) + \frac{\partial \pi \sigma T}{\partial \sigma} = \pi S \]

If we integrate this equation over a polar cap of radius \( \Delta \phi \)

\[ \int_{\pi/2}^{\pi/2 - \Delta \phi} \int_{0}^{2\pi} \frac{\partial \pi T}{\partial t} a^2 \cos \phi d\phi d\lambda = -\int_{\pi/2}^{\pi/2 - \Delta \phi} \int_{0}^{2\pi} \nabla \cdot (\pi \mathbf{v} T) a^2 \cos \phi d\phi d\lambda \]

\[ -\int_{0}^{2\pi} \int_{0}^{\pi/2} \frac{\partial \pi \sigma T}{\partial \sigma} a^2 \cos \phi d\phi d\lambda - \int_{0}^{\pi/2} \int_{0}^{\pi/2 - \Delta \phi} \pi a^2 \cos \phi d\phi d\lambda \]

and we assume the value of \( \frac{\partial \pi T}{\partial t} \) to be approximately constant over the polar cap

\[ \int_{\pi/2}^{\pi/2 - \Delta \phi} \int_{0}^{\pi/2} \frac{\partial \pi T}{\partial t} a^2 \cos \phi d\phi d\lambda \approx \left( \frac{\partial \pi T}{\partial t} \right) \int_{0}^{\pi/2} \int_{0}^{\pi/2} a^2 \cos \phi d\phi d\lambda \]

\[ = 2\pi a^2 (1 - \cos \Delta \phi) \left( \frac{\partial \pi T}{\partial t} \right) \text{NP} \]

The first term in the rhs is, using Gauss' theorem

\[ -\int_{\pi/2}^{\pi/2 - \Delta \phi} \int_{0}^{2\pi} \nabla \cdot (\pi \mathbf{v} T) a^2 \cos \phi d\phi d\lambda = -\int_{\pi}^{\pi \mathbf{v} T} a \sin \Delta \phi d\lambda \]
This can be approximated as
\[
-\frac{2\pi}{IM} \sum_{i=1}^{IM} \pi_i v_i T_i \sin \Delta \phi
\]

The third term on the rhs is
\[
\int_0^{2\pi/2} \int_{-\Delta \phi}^{\Delta \phi} \frac{\partial T}{\partial \phi} a^2 \cos \phi d\phi d\lambda \approx 2\pi a^2 (1 - \cos \Delta \phi) \left( \frac{\partial T}{\partial \phi} \right)_{NP}
\]
and similarly with the source term.

From these equations we obtain
\[
\left( \frac{\partial T}{\partial t} \right)_{\text{Pole}} = -(-1)^m \frac{\cot \frac{\Delta \phi}{2}}{aIM} \sum_{i=1}^{IM} \left( \pi_i v_i T_i \right) \frac{\pi}{2} \Delta \phi - \frac{\partial T}{\partial \phi} \right)_{\text{Pole}} + S
\]
since
\[
\frac{2\pi a \sin \Delta \phi}{IM} / 2\pi a^2 (1 - \cos \Delta \phi) = \frac{2\sin \Delta \phi / 2 \cos \Delta \phi / 2}{IM a^2 \sin^2 \frac{\Delta \phi}{2}} = \frac{\cot \frac{\Delta \phi}{2}}{aIM}
\]

This formulation is used in the continuity, momentum temperature and moisture equations. Note that the first term changes sign in the South Pole (m=1). In the momentum equations we make use of the transformation
\[
U_i = -\sin \lambda u_i - (-1)^m \cos \lambda v_i
\]
\[
V_i = (-1)^m \cos \lambda u_i - \sin \lambda v_i
\]
where m=1 for the South Pole and m=2 for the North Pole. \(U_i, V_i\) are the "cartesian" velocities at longitude \(\lambda_i\), and \(u_i, v_i\) the corresponding spherical velocities.

The pressure gradient terms are computed making use of Green's Theorem:
\[
\int \int \frac{\partial \Phi}{\partial x} \, dx \, dy = \int dy \int \frac{\partial \Phi}{\partial y} \, dx \, dy = \int P \, dx
\]

II-4
For example

\[
\int_0^{\pi/2} \int_0^{2\pi/2-\Delta \phi} \frac{\partial}{\partial x} a^2 \cos \phi \sin \lambda d\lambda = -\phi \frac{a \Delta \phi}{\pi/2} (-\cos \phi) d\phi
\]

\[
\phi = \frac{\pi}{2} - \Delta \phi
\]

\[
\Delta \phi = \frac{a \Delta \phi 2\pi}{\sin \phi} \sum_{i=1}^{IM} \phi_i \cos \lambda_i
\]

In the U-momentum equation we have then the following pressure terms

\[
\frac{\partial \pi U}{\partial t} = -a \frac{\Delta \phi}{\sin \phi} \sum_{i=1}^{IM} \left( \phi_i + \frac{(\sigma R T)}{\rho} \pi_i \right) \cos \lambda_i
\]

and similarly for \( \pi V \).

In the model we have approximated \( \Delta \phi \approx 2 \sin \frac{\Delta \phi}{2} \)

Then

\[
\frac{\Delta \phi}{a (1 - \cos \phi) IM} \approx \frac{1}{a \sin \frac{\Delta \phi}{2} IM}
\]

Based on this formulation we construct the fourth order scheme at the Poles by taking \( \frac{4}{3} \) of the differences evaluated at \( \Delta \phi \) from the Poles (as expanded here), minus \( \frac{1}{3} \) of the differences at \( 2 \Delta \phi \) from the Poles.

This formulation is not conservative at the Poles. However we have found that this has had no noticeable effect in the conservation of mass or energy in the model. In our shallow water experiments we studied a set of equations that were quadratically conservative, but inconsistent at the Pole, and another scheme analogous to the GFDL scheme, which is both quadratically conservative and consistent.
at the Poles, but suffers from a serious truncation error near the Poles in the pressure gradient term. The scheme that we chose gave better results than the other two (Kalnay-Rivas, 1976).
Computation of the horizontal pressure gradient as suggested by N. A. Phillips

(1) Let $\theta = \bar{\theta} + \theta'$, $\bar{T} = 280^\circ K/1000^k$ i.e. constant

(2) $\phi = \bar{\phi} + \phi'$, $k = R/C_p = .286$

$$\frac{\partial \phi}{\partial p} = -C_p \bar{\phi} \quad \bar{\phi} = \phi_0 - C_p \bar{\phi}_p^k \quad \text{with} \quad \phi_0 = C_p \bar{\phi}_p 1000^k$$

(3) $T = \bar{T}(p) + T'$, $\bar{T}(p) = \bar{\phi}_p^k$

Thus our new dependent variables are

$\phi' = \phi + C_p \bar{\phi} (p^k - 1000^k)$

$T' = T - \bar{\phi}_p^k$

In this way $\pi(\nabla \phi + \frac{\sigma RT}{\bar{p}^k} \nabla \pi)$, the pressure gradient in the momentum equations gets transformed into

(4) $\pi(\nabla \bar{\phi} + \nabla \phi' + \sigma R \bar{\phi}_p^k \nabla \pi + \frac{\sigma RT'}{\bar{p}^k} \nabla \pi)$

$$= \pi(\nabla \phi' + \frac{\sigma RT'}{\bar{p}^k} \nabla \pi) + \pi(\nabla \bar{\phi} + \sigma R \bar{\phi}_p^k \nabla \pi)$$

But the second parenthesis is zero:

$$\nabla \bar{\psi} + \sigma R \bar{\phi}_p^k \nabla \pi = -C_p \bar{\phi}_p^k \nabla \pi = -C_p \bar{\phi}_p^k + \frac{\sigma RT'}{\bar{p}^k} \nabla \pi = 0$$

In regions of steep orography, the second parenthesis in (4) is much larger than the first. When the horizontal pressure gradient terms are computed in their original form, the near cancellation of the two terms introduces large truncation errors. The procedure suggested by Phillips greatly reduces this truncation error. We have chosen a simpler definition of $\phi_0$ than the one suggested by Phillips.
III. **Finite Difference Variables and Grid**

The notation used in fourth band model is the same as the standard GLAS(GISS) second order GCM except that we use a non-staggered horizontal grid. A complete description of the variables can be found in the TSANG-KARN documentation of the GISS 9 level model.

Let $u_{ijk}$, $v_{ijk}$, $T_{ijk}$, $q_{ijk}$, $\pi_{ijk}$ be the finite difference approximations to the primary variables $u, v, T, q,$ and $\pi$ at the mesh point $(i\Delta \lambda, j\Delta \phi, (k-\frac{1}{2})\Delta \sigma)$. Also the scaled variables $\pi u, \pi v, \ldots$ are approximated by $\pi_{ijk} u_{ijk}$, $\pi_{ijk} v_{ijk}$, $\ldots$.

The finite difference equations also use the following geometric arrays:

\[
\begin{align*}
DXP(j) &= m_j = a \cos \phi_j \Delta \lambda, \\
DYU(j) &= n_j = a \Delta \phi_j,
\end{align*}
\]

We use a factor of 12 in $DXYP$ to make our scaled fourth order differences simpler.

\[
\begin{align*}
DXYP(j) &= 12 \cdot n_j, \\
\Pi_{ijk} &= DXYP(j) \cdot \pi_{ij}, \\
U_{ijk} &= n_j \pi_{ij} U_{ijk}
\end{align*}
\]

\[
\begin{align*}
S_{ijk} &= \Pi_{ij} \phi_{ijk}, \\
ADLDP &= 12 a \Delta \lambda \Delta \phi, \\
V_{ijk} &= m_j \pi_{ij} V_{ijk}
\end{align*}
\]

\[
F_{ijk} = DXYP(j) f_j + ADLDP \cdot \sin^2 \phi_j U_{ilj}
\]

**Horizontal Grid**

The fourth order band model uses an unstaggered grid in the horizontal direction.

The intervals $j=JML(N.\text{Pole})$ and $j=1(S.\text{Pole})$ represent the JM intervals between south pole and north pole. The intervals $i=1, i=2, \ldots, i=IM, i=IM+1$ represent the IM intervals in the longitudinal direction and periodic, where $i=IM+1$ is identical to $i=1$. 

III-1
\[ \Delta \lambda = \frac{\Delta \text{LONG}}{IM} \quad \Delta \phi = \frac{\Delta \text{LAT}}{JM} \]

\[ \lambda_i = (i-1) \Delta \lambda - \pi = (i-1)5^\circ - 160^\circ \quad \phi_j = (j-1) \Delta \phi - \frac{\pi}{2} = (j-1)4^\circ - 90^\circ \]

JSP = 1  JNP = JM1 = JM+1

**Vertical Grid**

The vertical grid is staggered; the values of all the variables \( u,v,T,q,\pi, \ldots \) except \( \delta \) are computed at the center of each layer. The values of \( \delta \) are computed at the edges of the layers.

![Vertical Grid Diagram]

In the case of uniform vertical resolution and NLAY=9 vertical layers,

\[ \sigma_k = \text{SIG}(K) = \frac{k-1}{9}(0,1/9,\ldots,1) \]

\[ \sigma_{k^*} = \text{SIGE}(K) = \frac{2k-1}{18} = \left(\frac{1}{18},\ldots,\frac{17}{18}\right) \]

and \( \Delta \sigma = 1/9 \).

\( \delta_{ijk} \) and its scaled version \( \delta_{ijk} \) are defined at the eight interior edges, i.e. for \( \sigma_k \) with \( k=1 \) to 8 since \( \delta(0)=\delta(1)=0 \) from the boundary conditions. The pressure \( p_{k^*} = \pi \sigma_{k^*} + p_{\text{TOP}} \) is defined at the same level as \( \delta_{k^*-1} \).

Note for level \( k \) we need \( \delta_{ijk} \) and \( \delta_{ijk-1} \) to form the second order vertical differences:

\[ \delta_{ijk} \left(V_{ijk} + V_{ijk+1}\right) - \delta_{ijk-1} \left(V_{ijk} + V_{ijk-1}\right) \]
III-2 Periodic Filtering of Short Waves

An integral part of the numerical scheme is the periodic application (every ISMTH steps, generally 2 hours) of a 16th order Shapiro filter. This has the effect of removing waves shorter than $4\lambda x$, which are not resolved in the model, while waves longer than $4\lambda x$, which are accurately computed by the difference scheme, are not affected by the filter (Kalnay-Rivas and Hoitsma, 1979).

The filter is applied to an array $q_j$ in the following way:
Let $d_+(q_j) = q_{j+1} - q_j$, $d_-(q_j) = q_j - q_{j-1}$

Then a Shapiro filter of order $2N$ is given by

$$\tilde{q}_j = q_j - (-1)^N (d_+ d_-)^N q_j$$

The response of the filter applied to a wave of the form $q_j = Q \exp\left(\frac{2\pi}{L} \lambda x_j\right)$ is

$$\tilde{q}_j = (1 - \sin^2 \frac{\pi \lambda x_j}{L}) q_j$$

The 2-dimensional filter is applied as a product of 1-dimensional filters (first in longitude, then in latitude). In latitude we filter the fields on great circles formed by meridians of longitude $\lambda$ and $\lambda + \pi$, where $0 \leq \lambda < \pi$. We are presently filtering only potential temperature and sea level pressure. These fields were chosen because they are not very affected by orography. Winds are not currently filtered, because the adjustment between mass and velocity fields does not allow the development of short waves in the winds alone. However, in the tropics, where the adjustment of winds to the mass field is minimal, the winds are somewhat noisy, and we may opt to filter them too.
IV. Boundary Conditions

Periodicity in the Zonal (East-West) direction

\[ \Pi_{IM+mj} = \Pi_{mj} \quad m=1,2,3,\ldots, j=2,\ldots,JM \]

\[ Q_{IM+mjk} = Q_{mjk} \quad m=1,2,3,\ldots, j=2,\ldots,JM, \quad k=1,\ldots,NLAY \]

for \( Q=u,v,T,q,\phi,\delta,\nu,\pi,\ldots \)

Boundary conditions at the north and south poles. Define the array INDEX as follows:

\[ \text{INDEX}(i) = i + \frac{IM}{2} \quad i=1,2,\ldots,\frac{IM}{2} \]

\[ \text{INDEX}(i+\frac{IM}{2}) = i \quad i=\frac{IM}{2}+1,\ldots,IM \]

i.e., INDEX: \( \frac{IM}{2}+1, \frac{IM}{2}+2,\ldots,IM,1,2,\ldots,\frac{IM}{2} \)

Then we can define

\[ \Pi_{iJM+2} = \Pi_{\text{INDEX}(i)JM} \]

for the points needed "beyond" the North Pole, and

\[ \Pi_{i0} = \Pi_{\text{INDEX}(i)2} \]

for the points "beyond" the South Pole

and similarly for \( T,q,\phi \).

For the horizontal velocity \( V=(u,v) \) we have

\[ V_{iJM+2k} = -V_{iJMk} \]

\[ V_{i0k} = -V_{i2k} \]
V. Finite Difference Equations

THE ZONAL (U) MOMENTUM EQUATION

$$\frac{\partial u}{\partial t} = \frac{1}{\cos \phi} \left[ \frac{\partial}{\partial x} \left( \frac{\partial (\pi u \cdot u)}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\partial (\pi \cos \phi \cdot u)}{\partial y} \right) \right] - \frac{\pi}{\cos \phi} \left[ \frac{\partial u}{\partial x} \right] + \frac{\sigma RT}{\rho} \left[ \frac{\partial u}{\partial x} \right] + \left( f + \frac{u \tan \phi}{a} \right) \pi v + \pi F_x$$

$$\frac{\partial \pi u}{\partial t} = \left\{ \begin{array}{l}
\text{COMP1: HA: PU1}_{i-1/2} \left[ (U_{ijk} + U_{i-1,ijk})(U_{ijk} + U_{i-1,ijk}) - (U_{ijk} + U_{i+1,ijk})(U_{ijk} + U_{i+1,ijk}) \right] \\
\text{COMP2: PG:} \\
\text{VA:} \\
\text{COMP3: C:}
\end{array} \right.$$

Note: COMP1, COMP2, COMP3 are the names of the three subroutines where the different terms are computed. HA: Horizontal advection terms. VA: Vertical advection terms. PG: Pressure gradient terms. C: Coriolis term.

Also note that PV2 is set equal to zero for j=0 and j=JM, i.e., there is no transport of mass across the poles.
THE MERIDIONAL (V) MOMENTUM EQUATION

\[
\frac{\partial \pi V}{\partial t} = -\frac{1}{a \cos \phi} \left[ \frac{\partial \left( \pi u \cdot v \right)}{\partial \lambda} + \frac{\partial \left( \pi \nabla \cos \phi \cdot v \right)}{\partial \phi} \right] - \frac{\partial \pi \phi}{\partial \sigma} - \frac{\pi}{a} \frac{\partial \phi}{\partial \phi} + \frac{\sigma RT}{P} \frac{\partial \pi}{\partial \phi} - \left( f + \frac{\pi u}{a} \right) u + \pi F_y
\]

COMP1: HA:

\[
\frac{\partial \pi V_{ijk}}{\partial t} = \left( 4 \cdot S_{ijk} \left( U_{ijk}^* + U_{ijk} + 1 \right) - \left( U_{ijk}^* + U_{ijk} - 1 \right) \right) (V_{ijk} + V_{ijk} + 1) - (U_{ijk}^* + U_{ijk} + 2) (V_{ijk} + V_{ijk} + 2)
\]

COMP2: P:

\[
+ \left( \pi_i^* m_j^* \left( \phi_i^* - \phi_i^* - 1 \right) + \frac{\sigma_{ijk} RT_{ijk}}{P_{ijk}} \left( \pi_{ijk} + \pi_{ijk} + 1 \right) \right)
\]

COMP3:

\[
-F_{ijk} \pi_{ijk} U_{ijk} + (\pi F_y)
\]
THE THERMODYNAMIC ENERGY EQUATION

\[ \frac{\partial \Pi}{\partial t} = -\frac{1}{\cos \phi} \frac{\partial \Pi U}{\partial \lambda} + \frac{\partial \Pi V \cos \phi \cdot T}{\partial \phi} - p^k \frac{\partial \delta T/p^k}{\partial \sigma} + \frac{\pi \sigma K T}{p} \left( \frac{\partial \Pi}{\partial t} + \frac{U}{\cos \phi} \frac{\partial \Pi}{\partial \lambda} + \frac{V}{a} \frac{\partial \Pi}{\partial \phi} + \frac{\pi Q}{C_p} \right) \]

COMP1: HA:

\[ \frac{\partial \Pi_{ijk}}{\partial t} = \{ 4^* \left[ \left( U_{ijk} + U_{i-1,jk} \right) \left( T_{ijk} + T_{i-1,jk} \right) - \left( U_{ijk} + U_{i+1,jk} \right) \left( T_{ijk} + T_{i+1,jk} \right) \right] - \}

\[ -0.5 \left[ \left( U_{ijk} + U_{i-2,jk} \right) \left( T_{ijk} + T_{i-2,jk} \right) - \left( U_{ijk} + U_{i+2,jk} \right) \left( T_{ijk} + T_{i+2,jk} \right) \right] + 

\[ 4^* \left[ \left( V_{ijk} + V_{ij-1k} \right) \left( T_{ijk} + T_{ij-1k} \right) - \left( V_{ijk} + V_{ij+1k} \right) \left( T_{ijk} + T_{ij+1k} \right) \right] - 

\[ -0.5 \left[ \left( V_{ijk} + V_{ij-2k} \right) \left( T_{ijk} + T_{ij-2k} \right) - \left( V_{ijk} + V_{ij+2k} \right) \left( T_{ijk} + T_{ij+2k} \right) \right] \}

VA:

\[ 0.5 \left( S_{ijk} \left( \frac{T_{ijk}}{P_{ijk}} + \frac{T_{ijk-1}}{P_{ijk-1}} - S_{ijk} \left( \frac{T_{ijk}}{P_{ijk}} + \frac{T_{ijk+1}}{P_{ijk+1}} \right) \right) \right) \]

COMP2:

\[ \left( \frac{\sigma_K T_{ijk}}{P_{ijk}} \right) \left( \frac{\partial \Pi_{ijk}}{\partial t} + \left( U^*_{ijk} \left( 8^* \left[ \pi_i+1j-\pi_{i-1}+1j \right] + \pi_{i-1}+1j+2j \right) \right) \right) + 

\[ V^*_{ijk} \left( 8^* \left[ \pi_{ij}+1j-\pi_{ij-1}+1j \right] + \pi_{ij-1}+1j+2j \right) \]

COMP3:

\[ \frac{\pi Q_{ijk}}{C_p} \]
THE MOISTURE EQUATION

\[ \frac{3}{3t} \frac{\partial q}{\partial t} = - \frac{1}{\cos \phi} \left[ \frac{3}{3\lambda} \frac{\partial Uq}{\partial \lambda} + \frac{3}{3\phi} \frac{\partial V \cos \phi \cdot q}{\partial \phi} \right] - \frac{3}{\pi \sigma} + \pi (E-C) \]

COMP1: HA

\[ \frac{3}{3t} \frac{\partial q_{ijk}}{\partial t} = \{4.*[(U^*_{ijk} + U^*_{i-1jk})(q_{ijk}^* + q_{i-1jk})^1 - (U^*_{ijk} + U^*_{i+1jk})(q_{ijk}^* + q_{i+1jk})] \]

\[ -0.5*[(U^*_{ijk} + U^*_{i+2jk})(q_{ijk}^* + q_{i+2jk}) - (U^*_{ijk} + U^*_{i+2jk})(q_{ijk}^* + q_{i+2jk})] \]

\[ +4.*[(V^*_{ijk} + V^*_{ijk-1})(q_{ijk}^* + q_{ijk-1}) - (V^*_{ijk} + V^*_{i+1k})(q_{ijk}^* + q_{ij+1k})] \]

\[ -0.5*[(V^*_{ijk} + V^*_{ijk-2k})(q_{ijk}^* + q_{ijk-2k}) - (V^*_{ijk} + V^*_{i+2k})(q_{ijk}^* + q_{ij+2k})] \]

\[ +0.5*[S_{ijk-1}(q_{ijk}^* + q_{ijk-1}) - S_{ijk}(q_{ijk}^* + q_{ijk+1})]/\Delta \sigma_k \]

COMP3:

\[ +\pi(E_{ijk} - C_{ijk}) \]

Note: The current transport scheme for the moisture field is being modified.
THE PRESSURE-TENDENCY EQUATION

\[
\frac{\partial \pi}{\partial t} = - \frac{L}{\cos \phi} \sum_{k=1}^{L} \frac{\partial (\Delta \sigma_k)}{\partial \lambda} \left( \frac{\partial \pi U}{\partial \lambda} + \frac{\partial \pi V \cos \phi}{\partial \phi} \right)
\]

COMPL:

\[
\text{CONV}_{i+l,j} = \Delta \sigma_k \left\{ 8 \cdot (U_{i-1,j} - U_{i+1,j}) + U_{i+2,j} - U_{i-2,j} \right. \\
+ 8 \cdot (V_{i,j-1} - V_{i,j+1}) + V_{i,j+2} - V_{i,j-2} \}
\]

Thus

\[
\Delta \sigma_k \frac{\partial \pi}{\partial t} = \text{CONV}_{ikj} - \Delta S_{ijk}
\]

giving:

\[
\Delta S_{ijk} = \left\{ \Delta S_{ijk-1} + \text{CONV}_{ikj} - \Delta \sigma_k \frac{\partial \pi}{\partial t} \right\}
\]

THE VERTICAL VELOCITY EQUATION

\[
\frac{\partial \pi}{\partial t} = - \frac{L}{\cos \phi} \left( \frac{\partial \pi U}{\partial \lambda} + \frac{\partial \pi V \cos \phi}{\partial \phi} \right) - \frac{\partial}{\partial \sigma} (\pi \delta)
\]

Thus

\[
\Delta \sigma_k \frac{\partial \pi}{\partial t} = \text{CONV}_{ikj} - \Delta S_{ijk}
\]

giving:

\[
\Delta S_{ijk} = \left\{ \Delta S_{ijk-1} + \text{CONV}_{ikj} - \Delta \sigma_k \frac{\partial \pi}{\partial t} \right\}
\]
VI. The Forecast Equations at the Poles

The fourth order band model uses a spherical cap at the poles, and the finite difference approximations to the equations of motion must be derived for this spherical region. Stereographic projection is used to give us a well-defined velocity vector at the poles.

From trigonometry, (Fig. VIa) if the vector \((U, V)\) in \(X-Y\) coordinates is represented by \((U', V')\) in \(X'-Y'\) coordinates where the prime coordinate axes are rotated by an angle \(\lambda\) then,

\[
\begin{align*}
U &= U'\cos\lambda - V'\sin\lambda \\
V &= U'\sin\lambda - V'\cos\lambda
\end{align*}
\]

(1) \(U\) \(\equiv\) \(U'\cos\lambda - V'\sin\lambda\)
(2) \(V\) \(\equiv\) \(U'\sin\lambda - V'\cos\lambda\)

We want to interpret "polar stereographic" velocities \((U_{NP}, V_{NP})\) also as "spherical" velocities \((U, V)\), as we approach the North Pole along a meridian of longitude \(\lambda\).

From Fig. VIb, which shows a unit vector in both coordinate systems, we see that, at the North Pole, spherical coordinates are rotated by an angle \((\Pi + \lambda)\) with respect to the polar stereographic coordinates. Therefore,

\[
\begin{align*}
U_{NP} &= U\cos(\lambda + \frac{\Pi}{2}) - V\sin(\lambda + \frac{\Pi}{2}) \\
V_{NP} &= U\sin(\lambda + \frac{\Pi}{2}) + V\cos(\lambda + \frac{\Pi}{2})
\end{align*}
\]

or

\[
\begin{align*}
U_{NP} &= -U\sin\lambda - V\cos\lambda \\
V_{NP} &= U\cos\lambda - V\sin\lambda
\end{align*}
\]

Similarly, for the South Pole:

\[
\begin{align*}
U_{SP} &= -U\sin\lambda + V\cos\lambda \\
V_{SP} &= -U\cos\lambda - V\sin\lambda
\end{align*}
\]

also,

\[
\begin{align*}
U &= -U_{NP}\sin\lambda + V_{NP}\cos\lambda \\
V &= -U_{NP}\cos\lambda - V_{NP}\sin\lambda
\end{align*}
\]

(5) \(U\) \(\equiv\) \(-U_{NP}\sin\lambda + V_{NP}\cos\lambda\)
(6) \(V\) \(\equiv\) \(-U_{NP}\cos\lambda - V_{NP}\sin\lambda\)
The initial values for $U_{NPk}$, $V_{NPk}$, $U_{SPk}$, $V_{SPk}$ are obtained from equations 3 and 4 by averaging in the zonal direction on the line of latitude $j=45$ for the north pole and $j=2$ for the south pole. In the forecast stereographic velocities are advected and transformed back into spherical velocities after each time step by equations 5 and 6.

In the following equation we will denote the polar velocities by $V_{P_k,m'}$, $V_{P_k,m}$ and the temperature and other variables in a similar way, $T_{P_k,m'}$, $\pi_{P_k,m'}$, ... where $m=1$ for the south pole and $m=2$ for the north pole. The following constants are used,

$$COEF_1 = (-1)^m \quad COEF_2 = -COEF_1 \quad RADIM = 3 \cdot a \cdot IM$$

$$CON_1 = 4 \cot(\tfrac{1}{2} \Delta \phi) / RADIM$$
$$CON_2 = -\cot(\Delta \phi) / RADIM$$
$$CON_3 = 4DT / (RADIM \sin(\tfrac{1}{2} \Delta \phi))$$
$$CON_4 = DT / (RADIM \sin(\Delta \phi))$$

$$JPOL(K,M) = \begin{cases} 2 JM \quad & \text{if } \quad K = 1 \quad \text{and } \quad M = 1 \\ 3 JM-1 \quad & \text{if } \quad K = 1 \quad \text{and } \quad M = 2 \end{cases}$$

$r = JPOL(1,m) =$ first interior value of $j$ (2 for the S. Pole, JM for the N. Pole)

$s = JPOL(2,m) =$ second interior value of $j$ (3 for the S. Pole, JM-1 for the N. Pole).
ZONAL (U) MOMENTUM EQUATION (POLES)

\[ \frac{\partial \pi \mathbf{p}_{k,m}}{\partial t} = (\text{COEF1} \cdot [\text{CON1} \cdot \sum \pi \cdot \mathbf{v}_{irk} (U_{irk} \sin \lambda_i + \text{COEF1} \cdot \mathbf{v}_{irk} \cos \lambda_i) + \sum \pi \cdot \mathbf{v}_{isk} (U_{isk} \sin \lambda_i + \text{COEF1} \cdot \mathbf{v}_{isk} \cos \lambda_i)]
\]

\[ + \text{CON2} \cdot \sum \pi \cdot \mathbf{v}_{isk} (U_{isk} \sin \lambda_i + \text{COEF1} \cdot \mathbf{v}_{isk} \cos \lambda_i)]
\]

\[ + .5 [\dot{\mathbf{P}}_{k-1} (\mathbf{P}_k + \mathbf{P}_{k-1}) - \dot{\mathbf{P}}_k (\mathbf{P}_k + \mathbf{P}_{k+1})] / \Delta \sigma_k \}
\]

COMP2:

\[ + \pi P \cdot [-\text{CON3} \cdot \sum \phi \cdot \frac{\sigma \cdot \mathbf{rtp}^k}{\mathbf{p}^k} \cdot \pi \cdot \mathbf{c} \text{os} \lambda_i]
\]

\[ + \text{CON4} \cdot \sum \phi \cdot \frac{\sigma \cdot \mathbf{rtp}^k}{\mathbf{p}^k} \cdot \pi \cdot \mathbf{c} \text{os} \lambda_i\]

\[ + f \cdot \pi P \cdot \mathbf{v}_{P,m} + (\pi P \cdot \mathbf{S}^P_{k,m}) \]

COMP3:

\[ \text{MERIDIONAL (V) MOMENTUM EQUATION (POLES)} \]

\[ \frac{\partial \pi \mathbf{p}_{k,m}}{\partial t} = (\text{COEF1} \cdot [\text{CON1} \cdot \sum \pi \cdot \mathbf{v}_{irk} (U_{irk} \sin \lambda_i + \text{COEF1} \cdot \mathbf{v}_{irk} \cos \lambda_i) + \sum \pi \cdot \mathbf{v}_{isk} (U_{isk} \sin \lambda_i + \text{COEF1} \cdot \mathbf{v}_{isk} \cos \lambda_i)]
\]

\[ + .5 [\dot{\mathbf{P}}_{k-1} (\mathbf{P}_k + \mathbf{P}_{k-1}) - \dot{\mathbf{P}}_k (\mathbf{P}_k + \mathbf{P}_{k+1})] / \Delta \sigma_k \}
\]

COMP2:

\[ + \pi P \cdot [-\text{CON3} \cdot \sum \phi \cdot \frac{\sigma \cdot \mathbf{rtp}^k}{\mathbf{p}^k} \cdot \pi \cdot \mathbf{c} \text{os} \lambda_i]
\]

\[ + \text{CON4} \cdot \sum \phi \cdot \frac{\sigma \cdot \mathbf{rtp}^k}{\mathbf{p}^k} \cdot \pi \cdot \mathbf{c} \text{os} \lambda_i\]

\[ + f \cdot \pi P \cdot \mathbf{v}_{P,m} + (\pi P \cdot \mathbf{S}^P_{k,m}) \]

COMP3:

\[ - f \cdot \pi P \cdot \mathbf{p}_{k,m} + (\pi P \cdot \mathbf{S}^P_{k,m}) \]

VI-3
THE THERMODYNAMICS ENERGY EQUATION (POLES)

\[
\frac{\partial \pi P_{k,m}}{\partial t} = \{\text{COEF1} \times (\text{CON1} \times \sum_{i=1}^{\pi_{ir \text{irk}}} \text{ir}_{\text{irk}} + \text{CON2} \times \sum_{i=1}^{\pi_{i \text{isk}}} \text{i}_{\text{isk}}) \\
+ 0.5 P_{k,m} \times (S_{P_{k-1,m}} - S_{P_{k,m}}) / \Delta \sigma_{k}\}
\]

\[
\text{COMP2:}
\]

\[
\frac{\partial P_{k,m}}{\partial t} = \{\text{CON5} \times \sum_{i=1}^{\pi_{ir \text{ir}}} \text{i}_{\text{ir}} \times \text{COEF1} \times \text{UP}_{k,m} \times \text{COS}_{i} + \text{VP}_{k,m} \times \text{SIN}_{i}\}
\]

\[
\text{COMP3:}
\]

\[
+ (\text{Q} P_{k})
\]

THE MOISTURE BALANCE EQUATION (POLES)

\[
\frac{\partial \pi q_{k,m}}{\partial t} = \{\text{COEF1} \times (\text{CON1} \times \sum_{i=1}^{\pi_{ir \text{qirk}}} \text{ir}_{\text{qirk}} + \text{CON2} \times \sum_{i=1}^{\pi_{i \text{qisk}}} \text{i}_{\text{qisk}}) \\
+ 0.5 (S_{P_{k-1,m}} \times (qP_{k,m} + qP_{k-1,m}) - S_{P_{k,m}} \times (qP_{k,m} + qP_{k+1,m}) / \Delta \sigma_{k}\}
\]

\[
\text{COMP3:}
\]

\[
+ (\pi_{m} \times \text{EP}_{k})
\]

VI-4
THE PRESSURE TENDENCY EQUATION (POLES)

\[
\text{CONVPL}_{k,m} = \text{COEF1} \times (\text{CON1} \times \sum_{i=1}^{IM} V_{i,k} + \text{CON2} \times \sum_{i=1}^{IM} V_{i,k} \Delta \sigma_k)
\]

\[
\frac{\partial \pi_{P,m}}{\partial t} = \sum_{k=1}^{NLAY} \text{CONVPL}_{k,m}
\]

THE VERTICAL VELOCITY EQUATION (POLES)

\[
\dot{V}_{k,m} = \dot{P}_{k-1,m} + \text{CONVPL}_{k} - \Delta \sigma_k \frac{\partial \pi_{P,m}}{\partial t}
\]
VII. Diagnostic Equations ($\phi$, $\sigma$, $p$)

Once the updated values of $\pi U$, $\pi V$, $\pi T$, $\pi q$ are found we unscale:

$$U_{ijk}^{n+1} = \frac{U_{ijk}^{n+1}}{\pi U_{ijk}^{n+1}}$$

for all $i,j,k$.

Similarly for $V,T,q$. We also filter the fields near the poles to prevent linear instability (see subroutine AVRXX). $\sigma$ is obtained from $\dot{S}$ by unscaling also.

We determine from $\pi_{ij}^{n+1}$ and $\sigma_k$, $P_{ijk}^{n+1}$

$$P_{ijk}^{n+1} = \sigma_k \pi_{ij}^{n+1} + p_{TOP}$$

$p_{TOP}$=constant.

$\phi_S$ is the surface geopotential (a function only of latitude and longitude)

For the Phillips geopotential we define $(p)^k$ at the center of the layer in the following way:

$$(p)^k_{ijk} = \frac{P_{ijk}^{k+1} - P_{ijk}^{k+1'}}{(k+1)(P_{ijk}^{k+1} - P_{ijk}^{k+1'})}$$

$$p_{ijk}^{k+1} = (SIGE(k) \pi_{ij} + p_{TOP})^{k+1}$$

$$(SIGE(k) = \sigma_k')$$

$p_{ijk}^{k+1}$ is obtained by exponentiation and $(p)^k$ by differences.

The following equations represent the geopotential calculations used in the old fourth order model.
Let
\[ C_{ijk} = \frac{\eta_{ij}^v K R \Delta \sigma_k}{P_{ijk}} - \frac{C_p}{2} \left[ \sigma_k \left( \frac{P_{ijk+1} - P_{ijk}}{P_{ijk}} \right) + \sigma_{k-1} \left( \frac{P_{ijk} - P_{ijk-1}}{P_{ijk}} \right) \right] \]

for \( k = 1, \ldots, \text{NLAY} \)

with
\[ P_{ij0}^k = P_{ij1}^k \text{ and } P_{ij\text{NLAY}+1}^k = P_{ij\text{NLAY}}^k \]

An optimized version of \( C_{ijk} \) is:
\[ C_{ijk} = \frac{\eta_{ij}^v K R \Delta \sigma_k}{P_{ijk}} - \frac{5C_p}{P_{ijk}} \left[ \sigma_k (P_{ijk+1} - P_{ijk}) + \sigma_{k-1} (P_{ijk+1} - P_{ijk-1}) \right] \]

Rather than compute \( \phi \) and then subtract \( \tilde{\phi} \) we do everything at once:

(1) \[ \phi_{ij\text{NLAY}}' = \phi_S' - CPTH \ast (PSKAPA - P^k_{ij\text{NLAY}}) + \sum_{\ell=1}^{\text{NLAY}} C_{ij\ell} T_{ij\ell} \]

(2) \[ \phi_{ij\ell} = \phi_{ij\ell+1}' + CPTH \ast (P_{ij\ell+1} - P_{ijk}) \left( \frac{T_{ij\ell}}{P_{ijk}} + \frac{T_{ij\ell+1}}{P_{ijk+1}} - 2 \right) \]

where
\[ CPTH = C_p \cdot \bar{\sigma} \quad PSKAPA = 1000^k = p_s^k \]

VII-2
The fourth order band model has the option of using the Matsuno time scheme or the smooth leapfrog scheme (see MWR-Vol 100 (487-490) R. Asselin).

Let $Q^n$ represent a typical variable that is to be updated to time $n+1$, and let $D(Q^n)$ represent the nonlinear space differences. The Matsuno (Euler-backward) scheme is as follows:

$$\tilde{Q} = Q^n + \Delta t D(Q^n)$$

$$Q^{n+1} = Q^n + \Delta t D(\tilde{Q})$$

The standard leapfrog scheme is given by

$$Q^{n+1} = \frac{Q^n - Q^{n-1}}{2\Delta t} = D(Q^n)$$

For the smooth leapfrog scheme we replace $Q^{n-1}$ by $\bar{Q}^{n-1}$

$$Q^{n+1} = \bar{Q}^{n-1} + 2\Delta t D(Q^n)$$

with

$$\bar{Q}^n = (1-v)Q^n + .5v(Q^{n-1} + Q^{n+1})$$

Equation (3) represents a simple time filter except $\bar{Q}^{n-1}$ is used instead of $Q^{n-1}$ in order to save core storage. The above equations (2) and (3) represent the order in which the smooth leapfrog scheme is evaluated. For $n=1$ we define $\bar{Q}^0 = Q^0$ then we update in equation (2) followed by the filtering in equation (3) which is needed for the next time step.
The smoothing step introduces dissipation with respect to time, controlled explicitly by the parameter $v$, as compared to the implicit dissipation in the Matsuno scheme. The amplification factor can be found in the paper by Asselin.

A further modification must be made to the smooth leapfrog scheme when source terms are included. Essentially the idea is that we must include the source term effect over two steps rather than one. If we do not do this, then the source effects (COMP3) are included only in every other step which will introduce large discretization errors. (For details see the attached report, Appendix B.)

If the source terms are called every NCOMP3 steps, then for step $n = \text{NCOMP3}$

$$Q_n^* = Q^{n-2} + 2\Delta tD(Q^{n-1})$$

$$Q_{n-1}^* = (1-v)Q^{n-1} + .5v(Q^{n-2} + Q^n)$$

Then compute the source terms $S^n$ and include in both steps $n$ and $n-1$,

$$Q^n = Q_n^* + \text{NCOMP3} \cdot \Delta tS^n$$

$$Q_{n-1}^* = Q_{n-1}^* + \text{NCOMP3} \cdot \Delta tS^n$$

The actual code is complicated by the fact that we actually use scaled and unscaled variables, but the generalization is straightforward.
IX. Documentation of the Code (Preliminary)

The band fourth model uses special equivalences and nonstandard dimensions in order to have the variables P, U, V, T, SH stored in contiguous locations for each line of latitude. Thus, we desire to have the variables stored as follows.

\[
\begin{array}{cccccccc}
\text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} \\
\text{PHIS} & U & TS & V & \text{SHS} & T & P & \text{SH} & \text{PHIS} \\
1 & \text{IM+1} & (\text{NLAY+1})*\text{IM+1} & & & & & & \text{U} \\
\end{array}
\]

The scaled variables PT, UT, VT, TT, SHT are to be stored as follows:

\[
\begin{array}{cccccccc}
\text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} & \text{NLAY*IM} & \text{IM} \\
\text{PHIS} & U & TS & V & \text{SHS} & T & P & \text{SH} & \text{PHIS} \\
1 & \text{IM} & (\text{NLAY+1})*\text{IM+1} & & & & & & \text{U} \\
\end{array}
\]

The above storage designation is accomplished by dimensioning PHIS(4*(NLAY+1)*IM,JNP) instead of PHIS(IM,JNP) (for a fine grid we have PHIS(2880,46), for ultrafine we have PHIS(4800,72)). Then we equivalence NLAY*IM locations of the first line of latitude of U with PHIS(IM+1,1) to PHIS((NLAY+1)*IM,1) (the first IM locations of PHIS are used for the quantity PHIS itself).

Similarly, we equivalence TS(1,1) to TS(IM,1) with PHIS((NLAY+1)*IM+1,1) to PHIS((NLAY+2)*IM,1) and so forth. (See the enclosed computer code for the exact values in the fine and ultrafine versions.)
In order to have the successive lines of latitude arranged properly in storage we dimension our variables to achieve this purpose. Thus, we have \( U(IM,4\times(NLAY+1),1) \) instead of \( U(IM,NLAY,JM) \), similarly for \( V,T,SH,UT,VT,TT, \) and \( SHT \). For \( P \) we have \( P(4\times(NLAY+1)\times IM,1) \) instead of \( P(IM,JNP) \) and similarly for \( TS, SHS,GT,GW, \) and \( PT \). Note that the variables \( U,UT,... \) are only computed at \( IM\times NLAY\times JNP \) points, the special dimensions are needed to properly align the variables in storage. It is important to recall that we are using two properties of the FORTRAN Compiler:

First, by equivalencing two elements of two different arrays we implicitly equivalence the other elements of the arrays. Second, that last array dimension can be left as 1 as long as it is dimensioned properly in the calling routine or it is equivalenced to a properly dimensioned array. Note we could have dimensioned \( U \) as \( U(IM,4\times(NLAY+1),JNP) \) we would use the same amount of storage. It is crucial to have the \( IM\times (NLAY+1) \) dimension because it would cause the computations on \( U \) for the successive lines of latitude to be shifted \( IM\times (NLAY+1) \) locations where the next line of latitude of \( U \) are stored. For clarity and simplicity in programming we use the standard equivalence of \( U,V,T,SH \) with \( Q(I,L,N,J) \) with \( U \) equivalent to \( Q(I,L,1,J) \) and so forth. Since we want \( P,U,V,T, \) and \( SH \) in contiguous storage locations and the equivalence of \( Q \) with \( U,V,T, \) and \( SH \), then we must dimension \( Q \) as \( Q(IM,NLAY+1,4,1) \) instead of \( Q(IM,NLAY,4,1) \). We fill in the extra locations by including \( PHIS,TS,SH, \) and \( P \).
COMPØ Description

This subroutine contains the time schemes and controls the calling sequences of the routines COMP1, COMP2, COMP3, and the polar filtering routine AVRX.

The logic of this subroutine involves three considerations. First, it permits one to choose either the Matsuno or the smooth leapfrog time scheme. Second, the latter scheme involves the use of storage arrays PSM and QSM. Third, the unscaling, smoothing of the updated variables QT and the call of subroutine COMP3 occur at the value JS2=J-2 when J is the value of the current line of latitude which is being computed. The routine COMP1 and COMP2 require the values of U,V,... at JS2 in order to compute the updated values at J because fourth order differences are used in the meridional (J) direction. Only after the COMP1 and COMP2 are called for value J can we unscale, smooth, and finish processing the variables at JS2.

The code for the poles is identical in format with the code for the other J values except the variables are scaled by the pressure PPOL only.

The main program contains two calls to COMPØ which cannot be treated independently because of the calling sequence: COMPØ(Q,QT), COMPØ (QT,Q). If the first call is a leapfrog step (LF), the second one must be also LF. If the first call is Matsuno predictor (MP), the second one must be Matsuno corrector (MC). This is represented symbolically by LF→LF, or MP→MC. The second call can be followed by either MP or LF. Each of these combinations requires different transfers.
Description of the Time Step Sequence Parameters

NSTEP: Counts the time steps. Starts and restarts both begin with NSTEP=0.

NSEQ: The number of steps (combined matsuno and leapfrog) in each (repeated) sequence of time steps.

MLF(I): MLF(I)=0 or 1 according to whether the Ith step in the sequence is Leapfrog or Matsuno, respectively. First step is always Matsuno (MLF(1)=1).

ISMTH: Smoothing routine (SMSPHAP) is called MOD(NSTEP-NSM1, NSM1: ISMTH); if ISMTH=0, there is no smoothing.

NCOMP3: Physics routine (COMP3) is called MOD (NSTEP-NCM1, NCM1: NCOMP3).

Sample Runs

Matsuno only:

NSEQ=1, MLF(1)=1, MATSUN=1, DT=750., NSM1=0, NCM1=0
BCINO3=4, ISMTH=8

Leapfrog only:

NSEQ=1, MLF(1)=1, MATSUN=0, DT=600., NSM1=0, NCM1=0
NCOMP3=5, ISMTH=10

1 Matsuno, 4 Leap-Frog:

NSEQ=5, MLF=(1,0,0,0,0), MATSUN=not needed, NSM1=0, NCM1=0, DT=600., NCOMP3=5, ISMTH=10
The COMP1 subroutine contains the horizontal and vertical advection differences. The DO loops over I are arranged to make use of the periodicity of the variables in the zonal (I) direction. For example, suppose we are to compute \( D(I) = Q(I+1) - Q(I) \) for \( I = 1, \ldots, IM \). Then the corresponding code is

\[
\begin{align*}
I &= IM \\
\text{DO } 10 \ IP1 = 1, IM \\
D(I) &= Q(IP1) - Q(I) \\
I &= IP1 \\
10 \ CONTINUE
\end{align*}
\]

Where we used the periodicity \( Q(IM+1) = Q(1) \).

In the meridional (J) direction we compute our difference approximations in stages in order to make maximum use of each line of latitude of a typical variable when it is in core. The fourth order difference approximation to \( \frac{\partial Q}{\partial \phi} \) for \( j \) is

\[
\frac{4}{3} \left( \frac{Q(i, j+1, k) - Q(i, j-1, k)}{2\Delta\phi} \right) - \frac{1}{3} \left( \frac{Q(i, j+3, k) - Q(i, j-3, k)}{4\Delta\phi} \right)
\]

Thus we see that \( Q(i, j) \) will be needed in the difference approximations to \( \frac{\partial Q}{\partial \phi} \) for \( j-2 \), \( j-1 \), \( j+1 \), and \( j+2 \). The corresponding code is

\[
\begin{align*}
\text{DO } 20 \ I &= 1, IM \\
QFLUX1 &= 4 \times (Q(I, L, JP1) + Q(I, L, J)) \\
QFLUX2 &= -0.5 \times (Q(I, L, JP2) + Q(I, L, J)) \\
D(I, L, JP2) &= D(I, L, JP2) - QFLUX2 \\
D(I, L, JP1) &= D(I, L, JP1) - QFLUX1 \\
D(I, L, J) &= D(I, L, J) + QFLUX1 + QFLUX2 \\
20 \ CONTINUE
\end{align*}
\]

For simplicity the array \( D \) is initialized to zero and \( Q \) is scaled so that (1) contains no divisions.

At the poles \( (j=1 \text{ or } J=JNP=46) \) we use the values given in section IV on boundary conditions. We have special code for these cases denoted \( J=2 \) or \( J=JM \) corrections.
COMP2 Description

This routine contains the Coriolis force term, the geopotential calculation (which should be made into a separate subroutine), and the pressure gradient and energy term calculations.

The geopotential PHI is dimensioned PHI(72,9,5) since we only need at most five storage locations for any computation. We use modular arithmetic (MOD5) to compute the indices JMOD, JP1MOD, JP2MOD which correspond to the standard index values of J,J+1, and J+2. Thus PHI(I,L,6) is stored in PHI(I,L,1), and we avoid shifting array values by using the JMOD index as a pointer.

For the south pole calculation we need geopotential values at J=2 and 3, thus, for J=1 we compute PHI for J=1,2, and 3. For successive values of J we need only compute PHI at JP2 which is needed in the pressure gradient calculation at J. Therefore, the calculation of PHI and the associated array PK are coded for calculation at JP2. Except for the first J value we are only computing the geopotential at one latitude value for each pressure gradient calculation.
X: Flow Charts

COMPO (Q, QT)

JS1=1
JS2=1

((MAIN LOOP))
DO 10 J=1, JM

((COMPUTE ALL J-PARAMETERS))

IF (J>1) GO TO 18

JP2=JP2MOD=2

((SAVE QTPOL(M) IN QSMPOL(M), M=1, 2))

18

IF (J>1) GO TO 18

((SAVE QJ (JP2) IN QSM(JP2MOD)))

IF (JP2>2) GO TO 25

((INCREMENT JP2, JP2MOD))

GO TO 18

25

CALL COMP1(Q, QT, J)
CALL COMP2(Q, QT, J)

((ELIMINATE NEGATIVE HUMIDITIES))

IF (PT4(400, 1100.)) STOP
IF (J<3) GO TO 200
IF (J=3) GO TO 70

29

((UNSCALE QT(JS2)))

CALL AVRQ(QT, JS2)

((LEAP FROG, MATSUNO AS RELATED TO THE SEQUENCE OF 2 CALLS TO COMPO: 1st CALL ⇒ LF ⇒ LF or MP ⇒ MC, 2nd CALL ⇒ LF ⇒ LF or LF ⇒ MP or MC ⇒ MP))

IF (LP=LF) GO TO 45
IF (LP=MP OR MP=MC) GO TO 58

((STATUS: MP=MC OR MC=LF))

IF (MC=LF)

((STATUS: MP=MC))

(contin.)
COMP1 \((Q, QT)\)

58 \[ ((P(I, JS2)+PT(I, JS2), Q-QT*DXYP*PT)) \] GO TO 64

45 \[ ((P(I, JS2)+*P(I, JS2)+a*(PSM+PT), Q-QT*DXYP*P*QT*DXYP*PT)) \]

63 CONTINUE
\[ ((\text{SOURCE TERM CORRECTION FOR LEAP FROG}) ) \]

IF (NOT COMP3 CALL)
\[ ((Q-QT*DXYP*PT)) \]

64 IF (NOT COMP3 CALL)
\[ ((Q-QT*DXYP*PT)) \]

IF (MATSUNO PREDICTOR STEP)
\[ CALL \text{COMP3}(Q, JS2) \]
\[ ((\text{COMPLETE SOURCE TERM CORRECTION, Q-QT*DXYP*PT}) ) \]

67 IF (J<JM) GO TO 200
\[ ((\text{INCREMENT JS2, JS2MOD})) \]

IF (JS2<JM) GO TO 29

70 \[ ((\text{POLES})) \]

200 JS2=JS1
JS1=J

10 CONTINUE
\[ ((\text{END OF J LOOP}) ) \]
RETURN

________________________

COMPL \((Q, QT, J)\)

________________________

((COMPUTE JP2-JS2MOD))

IF (J>2) GO TO 2150

JS1=JS2-JS1MOD=JS2MOD=1

(contin.)
COMPI (Q, QT, J)

2150 IF (J=JM)    \((\text{COMPUTE } PV_1, PV_2 = V^*_j + V^*_{j+2})\) \(\quad\rightarrow\) \(\text{GO TO 2158}\)

\[\text{IF}(J=1)\quad\rightarrow\quad\text{GO TO 2225}\]

2158 \((\text{COMPUTE } PU_1, PU_2 = U^*_1 + U^*_{1+2})\)
\((\text{COMPUTE HORIZ. ADV. - IN LONG. DIREC.})\)

\[\text{IF} (J = JM)\quad\rightarrow\quad\text{GO TO 2237}\]

2225 \((\text{COMPUTE HORIZ. ADV. - IN LAT. DIREC.})\)
\((J=1)\quad\rightarrow\quad\text{GO TO 2290}\)

2290 \((\text{COMPUTE HORIZ. ADV. - IN LAT. DIREC.})\)
\((J<JM)\quad\rightarrow\quad\text{GO TO 2405}\)

2237 \((\text{COMPUTE HORIZ. ADV. - IN LAT. DIREC.})\)
\((J=JM)\quad\rightarrow\quad\text{GO TO 2405}\)

2405 \((\text{COMPUTE HORIZ. ADV. - IN LAT. DIREC.})\)
\((J<JM)\quad\rightarrow\quad\text{GO TO 2405}\)

2405 \((\text{CONV. CALC. FOR CONT. EQ.})\)
\((J=JM)\quad\rightarrow\quad\text{GO TO 2405}\)

\[\text{IF} (J<JM) \rightarrow \text{GO TO 2405}\]

2225 \((\text{CONV. CALC. FOR CONT. EQ.})\)
\((J=1)\quad\rightarrow\quad\text{GO TO 2600}\)

2600 \((\text{POLES, } M=1 \text{ or } 2)\)
\((\text{HORIZ. ADV.})\)
\((\text{CONT. E.Q.})\)
\((\text{SIGDOT PL, PTPOL})\)
\((\text{VERT. ADV.})\)

\[\text{IF} (J<JM) \rightarrow \text{RETURN}\]

END OF COMPL

Alternative code from \text{---} to \text{---}

IF(J.EQ.1)    \((\text{COMPUTE } PU_1, PU_2)\)
\((\text{COMPUTE HA}^I)\)
\((J=2)\quad\rightarrow\quad\text{GO TO 2235}\)

2222 \((\text{J=2 CORREC})\)

2235 IF (J<JM) \((\text{J=JM CORREC})\)
\((J=JM)\quad\rightarrow\quad\text{GO TO 2235}\)

2225 \((\text{HORIZ. ADV.})\)
\((\text{CONT. E.Q.})\)

2405 \((\text{CONV. CALC. FOR CONT. EQ.})\)
\((J<JM)\quad\rightarrow\quad\text{GO TO 2405}\)

\[\text{RETURN}\]

END OF COMPL
COMP2 (Q,QT,J)

((COMPUTE JP2=... ))

IF (J>1) (JP2=JP2MOD=JPKP2=1))

3001 ((CORIOLIS ))

IF (J>JM) ((FIRST MAIN LOOP IN L))

3005 ((DO 3030 LX=1,NLAY ))

IF (L<NLAY) GO TO 3055

3007 ((COMPUTE PK(I,LL,JPKP2) FOR LL=1,NLAY))

3055 ((COMPUTE PHI' AT JP2MOD FOR L<NLAY))

3065 IF (JP2=1 or JP2=JNP) ((COMPUTE W(JP2MOD)))

3030 ((INCREMENT LX ))

((IF J=1 GO TO 3005 AND COMPUTE PHI(2),W(2))))

((2nd MAIN LOOP IN L))

3032 (DO 3031 LX=1, NLAY))

IF (J=1) ((2nd MAIN LOOP IN L))

IF (J > 2) ((J=2: VT,TT CORREC.))

3085 IF (J=20xJM) ((P) FOR U EQ.))

3135 IF (J < JM) (( CORREC. TO (P) J=JM))

3111 ((POLES))

3031 ((CONTINUE))

END

X-4
A FOURTH-ORDER FORECASTING MODEL*

(E. Kalnay-Rivas, D. Hoitsma, and P. Anolick)

The GISS fourth-order model (Kalnay-Rivas, et al., 1977), which is a fourth-order, energy-conserving CCM on an unstaggered grid, had shown promising capabilities. It produced forecasts that showed an improvement over the second-order GISS forecasts with the same fine grid (4° x 5°) resolution, but that were somewhat inferior to the "ultrafine" forecasts. However, the first version of the model required excessive amounts of computer memory and time for execution.

The model has been reprogrammed into the "fourth-order band model." The new program solves the primitive equations one latitude band at a time. The arrays are stored in an interlaced way, with all arrays being updated at the same latitude stored contiguously, and similarly for all arrays used in the computation of the time derivatives. This design of the program makes effective use of the virtual memory capability of the Modeling and Simulation Facility's IBM 370/165 or Amdahl computers. The virtual memory facility permits the execution of programs whose core size is larger than the one available, by placing the excess on disk and reading in those pages of information needed in the current calculations. The band fourth-order data structure and computations were constructed to optimize this virtual I/O process; a possible improvement may be to interlace also the arrays being updated with those used to compute the time derivatives. The use of the virtual memory facility avoids the explicit I/O used in the current Kern model, and yields a simpler program. (The band structure was suggested by G. Russell.)

The band fourth-order model computations have been optimized so that each time step is computed in the Amdahl in half the time required by the old model. The array structure has also been designed to reduce the amount of overall storage and high-speed memory by a factor of two (see Table 1).

Table 1. Comparison of Fourth-Order Model Computing Requirements.

<table>
<thead>
<tr>
<th></th>
<th>Original 4th-Order Model</th>
<th>Band 4th-Order Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core (bytes)</td>
<td>3500K</td>
<td>1500K</td>
</tr>
<tr>
<td>CPU time per step (COMPl, COMPl2)</td>
<td>34 sec</td>
<td>17 sec</td>
</tr>
</tbody>
</table>

The model has been programmed so that it can use both the Matsuno and the leapfrog time schemes, the latter with the Robert time smoother. If \( Q^n \) and \( D(Q^n) \) denote the fields at time \( n\Delta t \) and the corresponding time derivative computed from the space differences, the smoothed leapfrog scheme is

\[
Q^{n+1} = \tilde{Q}^{n-1} + 2\Delta t \, D(Q^n)
\]

\[
\tilde{Q}^{n} = Q^n + .5v(Q^{n-1} + Q^{n+1} -2Q^n)
\]

with \( \tilde{Q}^{0} = Q^{0} \). The use of a smoothing coefficient requires a slightly smaller time step. For example, with \( v = .1 \), the model is marginally stable with \( \Delta t = 288 \) sec., compared to \( \Delta t = 300 \) sec. for the Matsuno and leapfrog schemes.

The smoothed leapfrog scheme is further modified to include the subgrid "physics" terms, and scaling and spatial smoothing procedures. Since the "physics" is called every few time steps, unless the leapfrog scheme is restarted after every call to the "physics," only one of the two consecutive fields will be affected. The restarting procedure is time-consuming, so it has been replaced by the following algorithm

\[
\tilde{Q}^{n} = \tilde{Q}^{n-1} + Q^n
\]

\[
Q^{n+1} = Q^{n-1} + S^n
\]

where \( S^n \) corresponds to the "physics" terms. This procedure, which ensures that the physics is applied to two consecutive time steps, has been tested with good results.

The model can be extended into an "ultrafine" version in a straightforward way.

Preliminary Results. The first numerical integrations performed with the new band fourth-order model show dramatic improvements over forecasts made with the GLAS model with the same resolution ("fine grid" or 4° latitude by 5° longitude). The quality of the forecasts is now comparable with those produced with the "ultrafine" (2.5° latitude by 3° longitude) version of the model. Results from a 3-day numerical integration are presented in Figure 1. During an extended 8-day integration of the new fourth-order model, the atmospheric systems remained remarkably smooth, exhibiting a realistic behavior both with respect to position and intensities.

References

The 4th Order GISS Model of the Global Atmosphere*

E. Kalnay-Rivas 1) and 2), A. Bayliss 3) and J. Storch 3)

1) Massachusetts Institute of Technology Dept. of Meteorology, Cambridge, MA 02139, USA
2) Goddard Institute for Space Studies NASA, New York, NY 10025, USA
3) Present affiliation: ICASE, NASA Langley Research Center Hampton, Virginia 23665, U.S.A.

Abstract: The new GISS 4th order model of the global atmosphere is described. It is based on 4th order quadratically conserving differences with the periodic application of a 16th order filter on the sea level pressure and potential temperature equations, a combination which is approximately enstrophy conserving. Several short range forecasts indicate a significant improvement over 2nd order forecasts with the same resolution (~ 400 km). However the 4th order forecasts are somewhat inferior to 2nd order forecasts with double resolution. This is probably due to the presence of short waves in the range between 1000 km and 2000 km, which are computed more accurately by the 2nd order high resolution model. An operation count of the schemes indicates that with similar code optimization, the 4th order model will require approximately the same amount of computer time as the 2nd order model with the same resolution.

It is estimated that the 4th order model with a grid size of 200 km provides enough accuracy to make horizontal truncation errors negligible over a period of a week for all synoptic scales (waves longer than 1000 km).

1. Introduction

It is generally accepted that the use of 4th order finite differences is more efficient in reducing space truncation errors than the use of higher resolution on a 2nd order model (Kress and Oliger, 1972). Linear analyses and shallow water type of experiments give an upper limit of the improvement that can be expected from the reduction of errors in the horizontal differences. For example, Table 1 corresponds to a linear wave equation with phase speed c = 11 m s⁻¹, typical of atmospheric motions. It provides a measure of the "computational predictability period" after which the errors introduced by horizontal truncation alone become very serious. The table suggests that for waves longer than 2000 km a 4th order -- 400 km grid model is preferable to a 2nd order -- 200 km model. Waves shorter than 2000 km are forecast more accurately by a 2nd order -- 200 km grid model. In order to insure that horizontal truncation errors are small in the 1-3 week period of atmospheric predictability for all synoptic scale waves it is necessary to use either 4th order differences with a grid resolution of the order of 200 km or 2nd order differences with a grid of the order of 100 km. Numerical experiments with simple nonlinear models (Williamson and Browning, 1973, Kalnay-Rivas 1976a, from now on I also indicate dramatic reduction of errors by the use of 4th order differences. All these studies indicate that a considerable improvement in forecasting skill is to be expected from the use of 4th order differences if a substantial portion of the forecasting errors is due to horizontal truncation errors. In actual numerical forecasts there are several other important sources of errors: vertical truncation errors, errors in the initial conditions, and poor "physics" (i.e. parameterization of physical processes like radiation, dissipation, subgrid transports, cumulus convection, boundary layers, etc.)

* (Paper presented during the DMG-AMS Meeting, Hamburg 1976; see Preface to Issue 1 2/1977)

Table 1. Flapped time $T$ after which a wave of wavelength $L$ moving with a phase speed $c = 11 \text{ m s}^{-1}$ lags by more than 100 km due to space truncation errors.

<table>
<thead>
<tr>
<th>$\Delta x = 400 \text{ km}$</th>
<th>$\Delta x = 200 \text{ km}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2nd order differences</td>
<td>4th order differences</td>
</tr>
<tr>
<td>$L = 2000 \text{ km}, T = 0.4 \text{ days}$</td>
<td>$L = 2000 \text{ km}, T = 1.6 \text{ days}$</td>
</tr>
<tr>
<td>$L = 4000 \text{ km}, T = 1.6 \text{ days}$</td>
<td>$L = 4000 \text{ km}, T = 6.4 \text{ days}$</td>
</tr>
<tr>
<td>$L = 2000 \text{ km}, T = 1.5 \text{ days}$</td>
<td>$L = 2000 \text{ km}, T = 21 \text{ days}$</td>
</tr>
<tr>
<td>$L = 4000 \text{ km}, T = 21 \text{ days}$</td>
<td>$L = 4000 \text{ km}, T = 328 \text{ days}$</td>
</tr>
</tbody>
</table>

At the Goddard Institute for Space Studies (GISS), New York, we have developed a 4th order general circulation model (GCM) with the expectation that it will yield more accurate short range forecasts, and probably more realistic climate simulations, but with the improvement limited by the other sources of errors. The model is described in Section 2. The results of several experimental short range forecasts are discussed in Section 3. A sample 36-hour forecast is presented and compared with forecasts by the standard 2nd order GISS model, which has the same resolution as the 4th order model, and by the "ultrafine" 2nd order GISS model, which has twice the horizontal resolution. Section 4 contains the conclusions and a discussion of future work.

2. Description of the model

The 4th order GISS global atmospheric model is a primitive equation model using longitude ($\lambda$), latitude ($\phi$) and sigma ($\sigma$) coordinates. The basic equations and the parameterizations of physical processes are the same as those of the standard 2nd order GISS global model (Somerville et al., 1974). The finite difference scheme is quite different and is described in the following subsections.

2.1. Finite-difference scheme

The horizontal grid is uniform (constant $\Delta \lambda$ and $\Delta \phi$) and non-staggered. There are two ways to define such a grid, depending on whether variables are defined at the poles, as done by WILLIAMSON and BROWNING (1973), or half a grid size away from the poles (HOLLOWAY et al., 1973). We chose the former method because in the absence of smoothing near the poles it allows a time step twice as large as the latter. The singularity that spherical coordinates have at the poles is explicitly avoided by the use of a polar cell where "stereographic" velocities are used.

In the vertical direction we use a staggered grid with $\sigma$, the vertical velocity, defined at the boundaries of each layer, and all other variables in the center of the layer. The vertical grid can be non uniform although experiments have been made so far with constant $\Delta \sigma$. Experiments reported in I indicated that no significant improvement in accuracy was obtained when a 4th order staggered conservative scheme was used in the vertical direction. Therefore the model has 2nd order vertical differences. The computation of the geopotential $\Phi$ is performed as indicated by ARAKAWA (1972), with a modification suggested by PHILLIPS (1974).

a) Forecast equation away from the poles

Several systems of horizontal difference were tested as reported in I. The scheme chosen for the basis of these experiments consists of the simplest possible quadratically conservative 4th order differences.
We define the finite difference horizontal divergence operator at the vertical level $k$ as

$$D_k(b) = \frac{1}{\cos\phi} \left[ \frac{4}{3} \delta_\lambda (H u^\lambda) - \frac{1}{3} \delta_{2\lambda} (H u^{2\lambda}) \right]$$

and the finite difference gradient operator as

$$\nabla_b = \frac{i}{\cos\phi} \left[ \frac{4}{3} \delta_\lambda b^\lambda - \frac{1}{3} \delta_{2\lambda} (b^{2\lambda}) \right]$$

and the finite difference gradient operator as

$$\nabla_b = \frac{i}{\cos\phi} \left[ \frac{4}{3} \delta_\lambda b^\lambda - \frac{1}{3} \delta_{2\lambda} (b^{2\lambda}) \right]$$

Here II = $p_S - p_T$ is the difference between the pressure at the surface and the constant pressure at the top of the model, $a$ is the radius of the earth, $w = u\phi + v\psi$ is the horizontal velocity vector in spherical coordinates, and we use the finite difference notation

$$\delta_\lambda b = \left[ b(\lambda + n\Delta\lambda/2, \phi, a, t) - b(\lambda - n\Delta\lambda/2, \phi, a, t) \right]/(n\Delta\lambda),$$

and similar formulas for the other independent variables. With this notation, the continuity equation away from the poles is

$$\frac{\partial II}{\partial t} = -D_k(1) - II \delta_\phi \delta_k$$

or, if we integrate it in the vertical and make use of the boundary condition

$$u = 0 \text{ at } 0 = 0, 1,$$

$$\frac{\partial II}{\partial t} = -\sum_{k=1}^{K} D_k(1) \Delta \phi_k$$

where $K$ is the number of vertical layers. Equation (2.4b) is the forecast equation for II. The momentum equation is

$$\left( \frac{\partial h}{\partial t} \right)_k = -D_k(w) - II \delta_\phi \left( \frac{\partial w}{\partial \phi} \right)_k + \left( f + \frac{u_k \tan\phi}{a} \right) I_k \times (I\omega)_k$$

The first law of thermodynamics is

$$\left( \frac{\partial T}{\partial t} \right)_k = -D_k(T) - \left( p \frac{\partial}{\partial p} \left[ \frac{\partial}{\partial p} \right] \right)_k$$

$$+ \left( \frac{\partial}{\partial p} \left[ \frac{\partial T}{\partial \phi} + w_\phi \right] \right)_k + \frac{\partial}{\partial \phi} (T)Q.$$
Here $T$ is absolute temperature, $F$ is the frictional force, $k = R/C_p$, $Q$ is the diabatic heating per unit mass, $p$ is the pressure and $p^k$ is computed as indicated in Subsection 2.1c. The moisture equation for the water vapor mixing ratio $q$ is

$$
\begin{align*}
\left( \frac{\partial H_q}{\partial t} \right)_k &= D_k(q) + II \delta_p (\alpha q^0)_k + II (F - C).
\end{align*}
$$

(2.7)

$F$ and $C$ are the rates of evaporation and condensation.

In the momentum equation (2.5) we follow a device suggested by PHILLIPS (1974) to alleviate the difficulties of the computation of the pressure gradient in $\alpha$-coordinates in the vicinity of orography. We define $T^* = T - T(p), \Phi^* = \Phi - \Phi(p)$, where $T = \theta p^k, \theta = 280 K/(1001 \text{ mb})^k, \Phi = \Phi_0 - C_p \theta p^k$ and $\Phi_0 = [(1001 \text{ mb})^k - 1]^{1/2}$. We have simplified PHILLIPS' expression for $\Phi_0$ since its precise value is not important. With this procedure, in the pressure gradient term there is an exact cancellation of the terms $\nabla \Phi + \frac{\partial T^*}{\partial p} \nabla H$, and this implies a significant reduction of truncation errors in regions with steep orography.

b) Forecast equations at the Poles

We have followed the method used by WILLIAMSON and BROWNING (1973) and define a polar cap of radius $\Delta \phi$ on which we use "stereographic" (or rather "cartesian") velocity components defined by the transformation

$$
U_r, \Phi = -u \sin \lambda + v \cos \lambda
$$

$$
V_r, \Phi = -u \cos \lambda - v \sin \lambda
$$

(2.8)

with inverse

$$
U = U_r, \Phi \sin \lambda + V_r, \Phi \cos \lambda
$$

$$
V = V_r, \Phi \cos \lambda - U_r, \Phi \sin \lambda
$$

(2.9)

where the top and bottom signs correspond to the north and south poles respectively. The positive $x$-axis of the cartesian coordinates coincides with the meridian of longitude $\lambda = 0$. The difference in signs at the south pole is due to the choice of a right handed system of coordinates with the vertical unit vector pointing outwards. Formulas (2.8) and (2.9) are used to define "spherical" or "cartesian" velocity components wherever they are required in the finite difference equations.

The finite difference horizontal divergence operator at the poles is defined by an average over all longitudes $\lambda$:

$$
D_{r, \Phi} \Phi \left( \gamma \right) = \frac{1}{2} \sum_{i=1}^{2} \left[ \frac{4}{3} \Delta \phi, (\nabla g) \lambda_{i,} + \left( \frac{\pi}{2} - \Delta \phi \right) \lambda_{i,} \right]
$$

$$
\frac{1}{3} \Delta \phi, (\nabla g) \lambda_{i,} + \left( \frac{\pi}{2} - 2 \Delta \phi \right) \lambda_{i,}
$$

(2.10)

where $\lambda_{l} = (i - 3) \Delta \lambda, \Delta \lambda = \frac{2\pi}{3}$, and $\Delta \phi = \Delta \lambda \sin(n \Delta \phi)/[2 \pi \cos(n \Delta \phi)]$. The finite difference gradient operator at the poles is also defined by an average over all longitudes:

$$
\nabla_{r, \Phi} = \frac{2}{\Delta \phi} \sum_{i=1}^{2} \left[ \frac{4}{3} \Delta \lambda, + \left( \frac{\pi}{2} - \Delta \phi \right) \Delta \phi, \lambda_{i,} + \left( \frac{\pi}{2} - 2 \Delta \phi \right) \lambda_{i,} \right] \cdot \left[ \cos \lambda_{i,} \sin \lambda_{i,} \right]
$$

(2.11)
Then the continuity equation at the poles is
\[
\frac{\partial \Pi_{l,p}}{\partial t} = \sum_{k=1}^{K} D_{l,p,k} (1) \Delta \sigma_k,
\] (2.12)
the momentum equation is
\[
\left( \frac{\partial \Pi V}{\partial t} \right)_{l,p,k} = + D_{l,p,k} (V) - \Pi_{l,p} \delta_{\sigma} \left( \sigma_{l,p} \nabla \Pi_{l,p} \right) + f_{l,p} \kappa (\Pi V) \nabla \Pi_{l,p,k}
\]
\[= \Pi_{l,p} \left[ \nabla \Pi_{l,p} \Phi' + \frac{\partial RT'}{p} \nabla \Pi_{l,p} \right] + \Phi_{l,p,k}
\] (2.13)
where \( V_{l,p} = U_{l,p} + V_{l,p} j \). The thermodynamic equation is
\[
\left( \frac{\partial \Pi T}{\partial t} \right)_{l,p,k} = + D_{l,p,k} (T) - \left( \Pi_{l,p} \frac{\Pi_{l,p}}{\Pi_{l,p}} \frac{\sigma_{l,p}}{\Pi_{l,p}} \right) \left( T_{l,p} + \nabla \Pi_{l,p} \right)
\]
\[+ \frac{\Pi_{l,p}}{C_p} \left[ \frac{\partial \Pi T}{\partial t} + \nabla \Pi_{l,p} \nabla \Pi_{l,p} \right] + \frac{\Pi_{l,p}}{C_p} \Phi'_{l,p}
\] (2.14)
and the moisture equation is similar.

c) Diagnostic equations

In \( \sigma \)-coordinates, the pressure is defined by \( p = \sigma \Pi + p_T \). Following a suggestion by PHILLIPS (1974) we define at the center of a layer \( p_{l,p}^k = 1/(k+1) \delta_{\sigma} p_{l,p}^k + 1/\delta_{\sigma} p_{l,p}^k \), instead of \( p_{l,p}^k = (p_{l,p}^k)^{\sigma} \) as in ARAKAWA (1972). This formula was derived under the assumption that the potential temperature varies in a \( \sigma \)-layer much less than either the temperature of the pressure. PHILLIPS indicated that a more accurate relationship between temperature and geopotential can be expected from this formula.

The geopotential \( \Phi \) is determined following ARAKAWA (1972): If \( \Phi_S \) is the surface geopotential, then the geopotential at the center of the lowest layer is
\[
\Phi_k = \Phi_S + \sum_{k=1}^{K} \Delta \phi_k \nabla \phi_k
\] (2.15)
and at other levels
\[
\Phi_k = \Phi_{k+1} + C_p \Delta \sigma_k + \frac{1}{2} \delta_{\sigma} \left( p_{l,p}^{k+1/2} \right) \left( \frac{T_k + 1/2}{p_k^{k+1/2}} \right)
\] (2.16)
The coefficients \( C_k \) are determined from
\[
C_k = \frac{\Pi_0 \delta_{\sigma} \Delta \sigma_k p_k^{k+1/2}}{p_k \Delta \sigma_k p_k^{k+1/2}} - \frac{C_p \delta_{\sigma} \Delta \sigma_k p_k^{k+1/2}}{p_k \Delta \sigma_k p_k^{k+1/2}}
\] (2.17)
The vertical velocity in \( \sigma \)-coordinate is determined from (2.4a) and (2.4b), and the boundary condition \( \delta_{\sigma} = 0 \) at \( \sigma = 0, 1 \):
\[
\delta_{\sigma} \sigma_k = \frac{1}{\Pi} \left\{ \sum_{k=1}^{K} D_k (1) \Delta \sigma_k - D_k (1) \right\}
\] (2.18)
d) Boundary conditions

In the east-west direction we use periodicity: \( g(\lambda + 2\pi, \phi) = g(\lambda, \phi) \) for all variables. When the value of a variable is needed "beyond" the poles, we define it by continuation along the same meridian:

\[
w(\lambda, \pm \left(\frac{\pi}{2} + \Delta\phi\right)) = -w\left(\pi + \lambda \pm \left(\frac{\pi}{2} - \Delta\phi\right)\right) ;
g\left(\lambda, \pm \left(\frac{\pi}{2} + \Delta\phi\right)\right) = g\left(\pi + \lambda \pm \left(\frac{\pi}{2} - \Delta\phi\right)\right),
\]

where \( g \) represents all variables other than the two horizontal velocity components. In the vertical the top and bottom are material surfaces through which there is no flux \((\partial = 0 \text{ at } \sigma = 0, 1)\) except for subgrid boundary layer fluxes of momentum and heat included in \( F \) and \( Q \).

2.2. Filtering near the poles and high order filtering

The CFL computational stability condition requires a very small time step unless linearly unstable short waves are filtered out near the poles. For this purpose several alternative procedures have been tried but so far the method found to give most satisfactory results is the fourier filtering of the prognostic variables \( u, v, T \) and the indirect smoothing of \( H \) through the filtering of the sea level pressure (SLP) field.

The fourier filtering is performed polewards of 66° latitude. The amplitudes of the fourier components of zonal wavenumber \( n \) are multiplied by a transfer function which is 1 for \( n \leq N \) and decreases linearly to zero between \( n = N \) and \( n = N + 5 \). The number of retained modes is defined by \( N(\phi) = \text{integer part of } (90 \cos \phi) \). We have tried filtering the stereographic velocities \( U, V \) instead of \( u, v \) but no improvement was obtained.

Since both the surface geopotential and the surface pressure fields contain large amplitude short wave components due to the presence of orography, an artificial smoothing of these fields represents a distortion of the real geometry of the boundary. On the other hand, the SLP is an intrinsically smooth field, and its high wavenumber components more closely represent atmospheric waves. Therefore in our model the SLP is filtered near the poles and \( H \) is recovered by solving the transcendental equation used to relate them.

It was found that the 4th order model forecasts were less smooth than those of the 2nd order GISS model, especially in regions with steep orography. Based on the same considerations we have introduced a periodic application of a high (16th) order filter \((\text{SHAPRO, 1970})\) on the SLP and potential temperature fields, which are not very affected by orography.

The filter is of the form \( g = (1 - (F^2 \phi)^3)(1 - (F^2 \phi)^4) g \), where \( F^2 \phi(\theta) = (\theta_i + \theta_j - 2\theta_i \theta) \) and has a response \( F^2 \phi \exp(ik\lambda) = -\sin^2(k\Delta\lambda/2) \exp(ik\lambda) \). This filter eliminates waves shorter than \( 4\Delta\lambda \) and even after hundreds of applications has negligible damping effect on waves longer than \( 4\Delta\lambda \).

In the meridional direction there are three simple alternative forms for \( F^2 \phi \):

\[
F^2_{\phi h} = [(\theta_i + \theta_j - 2\theta_i \theta_j - 1)]/4
\]

\[
F^2_{\phi h} = [(\theta_i + \theta_j - 2\theta_i \theta_j - 1)]\cos \phi_j + 1/2 - (\theta_i - \theta_i - 1) \cos \phi_j - 1/2)/[(4 \cos \phi_j)
\]

and

\[
F^2_{\phi h} = [F^2_{\phi h} (\theta_i \cos \phi_j)]/\cos \phi_j
\]

FRANCIS \((1975)\) used \( F^2_{\phi h} \), which is the only one that conserves the area weighted average of \( g \). However, we have found that the three meridional filters produced virtually identical results. This could
be expected because the waves affected by the filter are too short to be strongly influenced by the
convergence of the meridians. Since $F_{el}^2$ and $F_{diff}^2$ can be programmed as efficiently as $F_{el}^2$ we find them
preferable. In the model we use $F_{el}^2$, with $g$ continued “beyond” the poles as indicated in Subsection
2.1d. At the poles $g_{i,p}$ is defined as the average over all longitudes $\lambda_i$ of the filtered values $g_{i,p}$. At the
present we are applying the high order filter to the SLP and potential temperature fields once every two
hours.

2.3. Enstrophy constraint

BAYLISS and ISAACSON (1975) have developed a simple method to make any finite difference
scheme conservative with respect to any quantity. The method has been tested in our model by forcing
conservation of enstrophy on the dry adiabatic version of the model, although in such a model it is
potential enstrophy that is conserved. The procedure is the following: Let the vectors $U$ and $V$ represent
the values of the velocity components discretized over the grid and let the functional $G(U, V)$ denote
the consistent 4th order approximation of the mean square vorticity. At each time step a correction
$U', V'$ is added to the predicted values $U, V$ such that $G(U + U', V + V') = G(U_0, V_0)$ where $U_0, V_0$
de note the velocities at time $t = 0$. Since this equation cannot be solved explicitly, it is linearized about
the predicted values $U, V$:

$$
\left( \frac{\partial G}{\partial U} \right)_{U', V'} \cdot U' + \left( \frac{\partial G}{\partial V} \right)_{U, V'} \cdot V' = G(U_0, V_0) - G(U, V)
$$

(2.19)

If we assume

$$(U', V') = \alpha \left( \frac{\partial G}{\partial U}, \frac{\partial G}{\partial V} \right)_{U, V}
$$

(2.20)

then $\alpha$ can be determined from (2.19):

$$
\alpha = \frac{G(U_0, V_0) - G(U, V)}{\left| \frac{\partial G}{\partial U} \right|^2 + \left| \frac{\partial G}{\partial V} \right|^2}
$$

(2.21)

It may be easily shown that the choice (2.20) minimizes $\| U' \| + \| V' \|$, the norm of the correction vector.
It has been found that this procedure improves the forecasting skill of the model. However, when it is
used in combination with the periodic application of the high order filter the improvement is marginal.
Therefore this option is not included in the standard version of our model.

2.4. Summary of the properties of the model

The finite differences of the model have the following properties:

a) Horizontal differences are performed on a nonstaggered grid and have 4th order truncation errors.
b) Vertical differences are performed on a staggered grid and have 2.5d order truncation errors.
c) The differences are conservative in the sense that in the absence of diabatic and dissipation, mass and
energy are conserved except for marginal terms at the poles. Non-conservative differences similar to
those used by WILLIAMSON and BROWNING (1973), starting from real data, proved to be nonlinearly
unstable after less than one day of integration.
d) Unlike the “box method”, the horizontal differences remain 4th order near the poles (KALNAY-RIVAS,
1976b)
e) The model contains no horizontal diffusion except for the possible use of a dissipative time scheme and for the periodic application of the high order filter. It has been shown in I that the combination of a 4th order quadratically conservative scheme with the periodic application of a high order filter replaces successfully the use of an enstrophy-conserving scheme. This is because waves shorter than four times the grid size are the ones subject to aliasing and to large truncation errors, and they are removed by the filter before they attain finite amplitude. Waves longer than four times the grid size are accurately computed by the 4th order scheme and are not affected by the filter.

3. Numerical forecasts

3.1. Analytical initial conditions

The model was tested using 3 dimensional extensions of both the nontrivial steady state solution and the Rossby-Haurwitz wave initial conditions used in I. The forecasts remained smooth during the several days of iteration. Errors in the steady state case were an order of magnitude larger than in I because of the use of single precision arithmetic on the IBM/360-95 computer. However, it was found necessary to use double precision for the longitudinally averaged pressure gradient at the poles in order to avoid excessive local error growth.

3.1. Forecasts with real initial data

As mentioned before, the parameterization of physical processes included in the terms $F, Q, C$ and $E$ of Equations (2.5) - (2.7) has been adapted from the standard 2nd order GISS model (SOMERVILLE et al, 1974). The only change that has been introduced is the reduction by a factor of 2 of the surface drag coefficient. This was found to be necessary in order to avoid an excessive damping of the pressure systems after one or two day forecasts. We believe that the fact that in the staggered grid the surface winds are obtained by a horizontal average reduces the relative effect of friction in the standard model.

At the present time (November 1976), we have performed 5 experimental short range forecasts. Since they have been made with slightly different versions of the 4th order model, we don’t yet have a reliable measure of the model’s forecasting skill. The initial data so far has been available in the staggered grid used by the standard model so that winds have been linearly interpolated to the nonstaggered grid. This is a source of error which may have adversely affected the 4th order forecasts.

We have performed a single 4-day forecast which indicated that the model remains very stable and that synoptic systems show no tendency to become unrealistically weak.

All the numerical 4th order forecasts show a significant improvement over the 2nd order forecasts performed with the same resolution. This improvement appears as a better estimation of the changes in position and intensity of several pressure systems. This in an encouraging result, especially in view of the study made by BAUMHEINER and DOWNEY (1976) which indicates that the standard 2nd order GISS model forecasts compare favorably with those made by the NMC and NCAR models with similar resolution.

On the other hand, and contrary to our expectations, the forecasts made with the 2nd order GISS model with double horizontal resolution were found to be either similar or superior to the 4th order forecasts. This is more true in the sea level pressure than in the 500 mb forecasts. We consider that there are several possible reasons for this result. One is the extra errors introduced in the 4th order model initial
data by the averaging of the winds. A second reason (and probably the most important one) is the existence of waves with significant amplitudes and wavelengths between 1000 and 2000 km. These short waves are better detected in the initial data and more accurately forecast by the 2nd order model with double resolution. The nonlinear interaction of these short waves with longer synoptic waves which according to linear theory are better forecasted by the 4th order model is clearly very important. A third important reason is the fact that errors introduced by the parameterization of subgrid processes become less important as the grid size is reduced.

We present an example of a 36 hour forecast of the sea level pressure corresponding to February 22, 1976, 12Z. Figure 1 is the verification sea level pressure map. Figure 2 is the forecast computed with the standard 2nd order GISS model, which has a resolution \( \Delta \lambda = 5^\circ, \Delta \phi = 4^\circ, \Delta \sigma = 1/9 \). Figure 3 is the 4th order forecast computed with the same resolution, and Figure 4 is the forecast obtained using the “ultrafine” 2nd order GISS model which has a resolution \( \Delta \lambda = 2.5^\circ, \Delta \phi = 2^\circ, \Delta \sigma = 1/9 \).

The most striking improvement showed by the 4th order forecast is in the position and intensity of the low in the eastern portion of North America. In fact, this low and the high over the Atlantic Ocean have been forecasted better by the 4th order model than by the 2nd order model with double resolution. On the other hand the developing cold high in central Canada and the intense low pressure center south of Greenland have been predicted better by the double resolution 2nd order model. As usual in this
type of comparison, the three forecasts share several important deficiencies. For example the intensity of the high over southwest USA has been overpredicted by the three models, probably because of difficulties introduced by the orography. The low south of Alaska has been erroneously split by the three models. This splitting, which is slightly less pronounced in the 4th order forecast, may be due to both errors in the initial data and orographic and coastal problems.

4. Summary and conclusions

We have described the characteristics of the 4th order GISS model of the global atmosphere. It is based on a quadratically conservative scheme with the periodic application of a 16th order filter on the sea level pressure and potential temperature fields. As shown in I this combination is approximately enstrophy-conserving. An operation count of the numerical schemes indicates that with similar code optimization the 4th order model will require approximately the same amount of computer time as the 2nd order enstrophy-conserving GISS model with the same resolution. We also plan to introduce a simplified semi-implicit scheme, and to study the possibility of using the combination of a nonstaggered vertical grid with the Kreiss 4th order method described in I.
The results of several short range forecasts indicate a significant improvement over the 2nd order forecast with the same resolution. This improvement is shown in the estimations of changes in position and intensity of several pressure systems. We plan to study the impact that the greater accuracy of 4th order differences has on the forecasting skill of variables of more practical importance, such as temperature and precipitation.

It has been found that the 4th order forecasts are somewhat inferior to the forecasts made with a 2nd order model with double horizontal resolution. We consider that one of the most important reasons for this result are the presence of waves in the range between 1000 km and 2000 km which are computed more accurately by the high resolution 2nd order model than by the 4th order model. Another important reason is that errors introduced by the parameterization of subgrid processes become smaller as the size is reduced.

It is estimated that the 4th order model with a grid size of 200 km provides enough accuracy to make horizontal truncation errors negligible over a period of a week for all synoptic scales (waves longer than 1000 km).
Figure 4. 36 hr. using the "ultrafine" 2nd order GISS model with double horizontal resolution!

Acknowledgements

We are grateful to Drs. M. Halem, W. Quirk and Profs. J. Charney and P. Stone for their continuous support and encouragement. Prof. E. Isaacson provided generous advice and Prof. Dr. Randall made useful comments on the manuscript. Prof. F. Sanders comments on the numerical forecasts are also gratefully acknowledged.

References

ARAKAWA, A. 1972: Design of the UCLA GCM. Techn. Rept. No. 7, Dept. of Meteorology, UCLA.


1. INTRODUCTION

The design of a numerical model for atmospheric simulation is not a straightforward procedure. Both in the areas of mathematical and numerical analysis, and in the parameterization of physical processes not explicitly resolved, the modeler faces several difficult choices between equally reasonable methods, and sometimes between similarly unsatisfactory methods.

In this paper we discuss the considerations leading to the numerical design of the GLAS Fourth-Order Global Atmospheric Model. This model, which was briefly described in Kalnay-Rivas et al. (1977), has been restructured, and several minor changes were introduced.

The computation time and memory requirements for the fourth-order model are now similar to those of the present second-order GLAS model with the same 4° latitude, 5° longitude, and 9 vertical-level resolution (Sommerville et al., 1974). However, the fourth-order model forecast skill is significantly better than that of the current GLAS model, and after 3 days it is comparable or better than that obtained with the 2.5° by 3° version of the GLAS model.

A discussion of several of the basic characteristics of the model design is contained in section 2. For each of them we present some of the possible alternatives, their advantages and disadvantages, and the reason for our choice. In section 3, we discuss the effect on numerical forecasts of changes in the accuracy, resolution, and conservation properties of the models. Section 4 contains some final remarks.

(1) As of this writing, several of the numerical experiments are not complete.

(2) Present affiliation: Department of Mathematics, Wesleyan University of Connecticut

2. DISCUSSION OF THE DESIGN OF THE MODEL

Different numerical analysts and atmospheric modelers often take different approaches in the design of a numerical model for weather prediction. Sometimes they even have different basic philosophies. For example Arakawa (1966, 1972), has been a pioneer in the development and use of numerical schemes that reproduce as closely as possible the conservation properties of the continuous equations of fluid dynamics that determine the motion of the atmosphere. On the other extreme, Kreiss and Oliger (1972, 1973) have advocated the use of more accurate schemes even when they don't formally satisfy any conservation properties.

The current GLAS model is based on Arakawa's [1972] second-order scheme using a staggered grid B. The scheme is energy conserving and approximately enstrophy conserving for non-divergent flow. In this section we isolate the areas in which the Fourth-Order model is different from the GLAS model, discuss some of the alternatives, and the justification of our choice.

2.1 Accuracy

There is a consensus among modelers that for finite difference models with second-order accuracy, horizontal resolution of about 400 km, and about 10 vertical levels, horizontal truncation errors are the most important source of errors. The truncation errors can be reduced by any of the following methods:

a. Increased horizontal resolution, retaining second-order differences.

Advantages: If the horizontal grid size is reduced by a factor of 2, truncation errors are reduced by a factor of 4. A comparison of the effect of truncation errors on the computational phase speed and group velocity of a linear wave is presented in Fig. 1. Another advantage is that increasing
the resolution allows smaller but possibly important scales to be explicitly included in the model.

**Disadvantages:** The reduction of error is slow, and the computation time is increased by a factor of 8.

**Advantages:** These fourth-order differences are considerably more accurate than the fourth-order differences of the first kind (Fig. 1). The crossover with double-resolution second-order differences in the phase speed error occurs at $3\Delta x$, and in the group velocity error at about $4\Delta x$.

**Disadvantages:** In the simplest case, Kreles' fourth order differences require the solution of tri-diagonal matrices. Finite element schemes require the solution of at least block tri-diagonal matrices. Even though there are efficient methods to perform these inversions, they are still computationally expensive.

The extra accuracy can be compensated with 4th order differences of the first kind by increasing the resolution, which has other advantages, as we mentioned earlier. These schemes, as well as in higher order finite difference schemes, waves in the range of $2\Delta x$ to $4\Delta x$ are still grossly misrepresented.

**Advantages:** The basis of the spectral expansion are the eigenfunctions of the wave equation; spectral schemes have no phase speed errors. Because of this they require less resolution than finite difference schemes.

**Disadvantages:** Because of the large number of computations required for the nonlinear terms, spectral schemes are competitive with finite difference schemes only in combination with the use of less resolution and semi-implicit time schemes.

Our choice: We chose to use fourth order finite differences of the first kind because they are computationally efficient and have small truncation errors except in the range of waves with wavelengths between $2\Delta x$ and $4\Delta x$.

### 2.2 Type of Grid

Both staggered and unstaggered grids have been widely used by atmospheric modellers.

**a. Unstaggered grid:** The advantage of this grid is its simplicity. Higher order schemes are easily developed with this grid. Its disadvantage is that all centered differences have to be computed over a distance of $2\Delta x$.

**b. Staggered grids:** Several staggered grid configurations are possible as reviewed by Arakawa [1972]. The one he called scheme C, which is the most commonly used, has the pressure defined at the center of a grid cell, and the velocity components $u$ and $v$ defined at their corresponding normal walls. This grid has the advantage that the pressure gradient and velocity divergence terms are computed over a distance of only $1\Delta x$, so that inertia gravity waves are computed with double resolution. Therefore, as pointed out by Arakawa, geostrophic adjustment is best represented in this grid. On the other hand, advection terms are computed with no more accuracy than in the unstaggered grid, and in the Coriolis' acceleration term, it is necessary to take horizontal averages of the velocities. The
higher resolution of inertial gravity waves reduces the maximum time step for explicit time schemes by a factor of two. Full fourth order schemes can be developed with staggered schemes but they are very involved (Kainay-Rivas, 1976). To date, models using staggered grids have introduced fourth order differences only in the advection terms.

Our choice: During extended range forecasts, second order errors in the non-advection terms may become important. For this reason we chose to use an untaggered grid that allows the use of a simple, full fourth-order scheme.

2.3 Conservation Properties and the Use of Horizontal Diffusion

With respect to conservation properties, there are basically 3 types of finite difference schemes for the primitive equations: a) Nonconservative schemes, the simplest of which is the one based on the advective form of the equations; b) quadratically or energy conserving schemes; and c) enstrophy conserving schemes. Adective and quadratically conservative schemes can be easily developed using staggered or untaggered grids [Lilly, 1965; Bryan, 1966]. Enstrophy conserving schemes for the primitive equations have been developed on a grid C by Grammeltvedt [1969] and Arakawa and Mintz [1974]. Sodarum [1965a, b] constructed a potential enstrophy conserving scheme on grid C, and Arakawa [1978] has recently developed a potential enstrophy and energy conserving scheme also on grid C.

Nonconservative schemes require a procedure to damp waves shorter than 4Ax, which otherwise grow spuriously causing catastrophic nonlinear instability [Phillips, 1959]. This has usually been done by means of linear or nonlinear horizontal diffusion, or by using dissipative numerical schemes such as the Lax-Wendroff, or schemes that contain explicit horizontal averaging.

Quadratically conservative schemes avoid the unbounded growth of the solutions associated with catastrophic nonlinear instability. However, as Arakawa [1972] and Sodarum [1975a, b] have pointed out, in the course of long integrations, there is still a spurious build up of energy in the shorter waves, which appears as an unbounded growth of the total enstrophy. In the absence of horizontal diffusion, this type of slow nonlinear instability will completely distort the solution. Enstrophy conserving schemes, on the other hand, impose a stronger constraint on the growth of the smallest scales present in the model. For this reason, the UCLA and the CLAS models, which use enstrophy conserving schemes, do not need to include horizontal diffusion.

It should be emphasized that conservation of enstrophy does not necessarily imply a more accurate or realistic simulation. In the real atmosphere, the constraint of quasi-geostrophic motion implies that very little of the energy generated in the baroclinically unstable scales can reach the smallest scales and be eventually dissipated (Charney, 1972).

In a numerical model, the finite resolution imposes an artificial "wall" at the short end of the spectrum, inducing an excessive accumulation of energy in the shortest waves. This problem is worst in nonconservative schemes, but it appears even in alias-free, energy- and enstrophy-conserving spectral models. This justifies using some parameterization of the unresolved subgrid eddies to withdraw energy from the smallest resolved scales.

For this purpose, Leith [1972] suggested the use of nonlinear horizontal diffusion in which the eddy diffusion coefficient is proportional to the local gradient of vorticity. This formulation is consistent with the transfer of energy to subgrid scales in two-dimensional turbulence. In another widely used formulation, suggested by Smagorinsky [1963] and based on a three-dimensional turbulent cascade theory, the diffusion coefficient is proportional to the deformation tensor. These formulations are better than the use of linear diffusion, but they both share the following problems: a) the diffusion coefficient is computed inaccurately for the shortest waves; and b) when the diffusion coefficient is large enough to avoid the spurious growth of the smallest scales, it produces excessive damping of the larger scales (Marleeha, 1975; Williamson, 1978). Furthermore, at the short end of the spectrum (scales of the order of 100 km), neither quasi-geostrophic nor 3-dimensional isotropic turbulence theories are really justified. Williamson [1978] generalized a higher order diffusion of the form KV², suggested by Kressa and Oliger [1972], still using the deformation type of diffusion coefficient. This formulation has the advantage that, because it is more scale dependent, there is less damping of the longer waves.

In our model, we have taken an approach closer to Fourier filtering the shortest waves, as suggested by Phillips [1959]. Our "subgrid parameterization" is based on the following argument: The 4th order scheme in adequate accurate for waves longer than 4Ax, but grossly inaccurate for waves between 2Ax and 3Ax. Since the shortest waves cannot provide any useful information in a finite difference scheme, we filter them out of the system while their amplitude is still small. Even though they are filtered out, the shortest waves still play an important role in the model: they act as a buffer or "sponge layer" in the spectrum domain, allowing energy to trickle down from longer waves and avoiding the accumulation of energy that would otherwise occur at the short wave cutoff.

The elimination of the short waves is performed in the model with the periodic use of a 16th order Shapiro [1970] filter, first introduced in GCMs by Francis [1975]. Figure 2 indicates the response of Shapiro filters of order 4, 8, and 16. It can be observed that in the case of a 16th order filter, waves longer than 4 are scarcely affected at all
even after 128 applications, which in our model correspond to a 10 day integration. Waves shorter than 4 are virtually eliminated. Lower order filters introduce too much decay at long scales.

At: 4th order, non-conservative, no smoothing.
B1: 4th order, quadrat. conserv., no smoothing.
B2: 8th order, 16th order filter/4 haps.
B3: 4th order, 8th order filter/4 time step.
B4: 2nd order, no smoothing.

**TABLE 1: Characteristics of the different runs in Fig. 3.**

We found that all stable runs conserved total energy with a high degree of accuracy. Fig. 3 shows the variation in time of the total potential enstrophy, which is conserved exactly in the continuous equations. The results indicate that when the 16th order filter is applied every time step (10 minutes) even a formally nonconservative scheme conserves both total energy and total potential enstrophy during a long integration (Run A4). The quadratically conservative scheme controls better the amount of energy going into the smallest scales, so that in Run B2 it was enough to apply the filter every 4 hours to conserve potential enstrophy within 0.052, even though such conservation is not formally guaranteed in the scheme.
It may be questioned whether the application of a high order filter eliminates small scale features like frontal zones or the effects of orographic or cumulus convection forcing on small scales. Fig. 4 indicates that this is true for lower order filters. However, a 16th order filter eliminates only those components which are not resolved anyway, and still allows for the formation of sharp gradient zones and strong local maxima.

10 PASSES (ONE DAY) OF SHAPIRO FILTER

"FRONT"

16th ORDER

8th ORDER

4th ORDER

2nd ORDER

10 PASSES (ONE DAY) OF SHAPIRO FILTER

"ITCZ"

16th ORDER

8th ORDER

4th ORDER

2nd ORDER

Fig. 4 Effect of 10 applications of a 16th order Shapiro filter on a) a step function b) a spike.

3. EXPERIMENTAL FORECASTS: PRELIMINARY RESULTS

We plan to perform an extensive series of forecasts to study the effect of using different schemes and varying resolution on the quality of actual weather forecasts. In this section we present some preliminary results.

We tested the Fourth Order model by making 3-day forecasts from several initial conditions. In every case the model performed much better than the GLAS model with the same 4° by 5° resolution. After 3 days the forecasts were comparable to the 2.5° by 3° version of the GLAS model in the sea level pressure maps, and had slightly less phase errors in the 500 mb maps.

Fig. 5a shows the verification sea level pressure map corresponding to a 3-day forecast with February 19, 1976, OZ (a case that has been studied in detail by Atlas et al., 1979). Fig. 5b is the 3-day forecast with the Fourth Order 4° by 5° model verifying on the same date. The excessive gradients, especially at high latitudes are due to a faulty computation of the ground temperature by the radiation routine used in that run. Figs. 5c and 5d are the forecasts generated by the GLAS models with 4° by 5° and 2.5° by 3° resolution respectively. Fig. 6 displays the corresponding 500 mb maps.

In Fig. 7a, we present the verification sea level pressure map corresponding to a 3-day forecast with February 1, 1976, OZ initial conditions. The results of six different 3-day forecasts are shown in Figs. 7b to 7g. The forecast in Fig. 7b was computed with the new full Fourth-Order model, using the same resolution and parameterization of physical processes (except for the long-wave radiation routine) as in the 4° by 5° GLAS model (Fig. 7c). In Fig. 7d, the model was the same as in 7b, but full Second-Order accuracy was used. If we compare these three made with the same resolution, we see that the fourth order model did considerably better, especially in forecasting the development and motion of the cyclone southwest of Greenland. The two second-order forecasts are close to each other.

(3) The idea of filtering these fields is due to Dr. A. Bayliss.
Fig. 7 shows the forecast made with the 2.5° by 3° GLAS model starting from the NASA initial conditions with assimilation of satellite data [Atlas et al., 1979]. All other forecasts were made from NMC's Global Analysis initial conditions. The cyclogenesis forecast was poorer than with the fourth order model, but over continental North America the forecast was better. This may be due to the higher resolution or, possibly, improved initial conditions.

Fig. 7f presents the forecast made with NMC's 6-layer, 380 km resolution, which has resolution comparable to our 4° by 5° grid. For these initial conditions, NMC's forecast was better than the GLAS second order forecast, although it shared some of its errors (such as a spurious anticyclonic over the Great Lakes).

Fig. 7g shows a forecast made with a slightly different version of the fourth order model. The differences were as follows: in the 7g forecast we used a Matsuno time step, a surface drag that was 0.75 of that of the standard model, the Shapiro was applied every two hours, and a slightly different scheme for the vertical advection of moisture was used. In the 7b forecast we used a combined Matsuno-leap-frog scheme, the surface drag was the same as in the standard model and the filter was applied every hour. The two forecasts are quite similar, but from other experiments it seems that the positions of the oceanic lows west of Spain and south of Alaska were somewhat affected by the change in frequency of the Shapiro filter.

4. FINAL REMARKS

Although these are preliminary results, the new Fourth-Order model seems to have very good forecasting skill. Similar excellent results with fourth order schemes were reported by Campana [1978, 1979], and by Williamson (1979). Campana found that most of the improvement over the second order model was obtained just by introducing fourth order accuracy on the horizontal averages performed on the Shuman-Hovermale model. Campana also obtained that for 2-day forecasts, fourth order differences were important in the advection terms not in the pressure and continuity terms. We made shallow water equation experiments that showed some deterioration of the solution after about 8 days when only the advection terms were written with fourth order accuracy. Since baroclinic models are much more unstable than the SWE, Campana's results may not hold during extended forecasts. We are performing experiments to study this possibility in our model. We are also repeating some of the forecasts using a vector invariant form of the momentum equations. The NMC 6-layer PE model used in Fig. 7f is based on such a formulation, and we want to determine if it affects the solution.

We are performing integrations with a 2.5° by 3° version of the Fourth-Order model. From linear theory, this should make horizontal truncation errors negligible for most synoptic scales for periods of a week or longer. Still, narrow atmospheric features like those due to sharp orographic forcing on the ITCZ will remain poorly resolved.

We have found the fourth order model to be extremely sensitive to the parameterization of physical processes. For example, it became unstable through excessive cooling near the surface, apparently due to a flaw in the radiation routine. None of the second order models was sensitive to this problem.

In this paper we have discussed the use of formally nonconservative schemes coupled with periodic filtering of the shortest waves as an alternative to the use of conservative schemes. It is clear from the experimental results that formal enstrophy conservation has had no beneficial impact on the quality of the forecast (Figs. 7c and 7d). Conservation of potential enstrophy [Arakawa, 1978] might have a more positive effect.

We think that the use of nonconservative schemes with high order periodic filtering is also justified in climate simulations as long as the rates of energy or enstrophy loss due to filtering remain much smaller than the observed rates of generation and dissipation.

Acknowledgements: Dr. A. Bayliss' contribution was crucial in the development of the model. The authors have benefited from stimulating discussions with Profs. M. Cane, E. Isaacson and D. Randall. The authors are grateful to Dr. M. Halem for his patience, encouragement and many useful suggestions. Dr. N. Rushfield and Mr. W. Connelly were extremely helpful in the development of the program. Dr. M. Wu provided us with her accurate radiation routine.

Fig. 5 Sea level pressure maps corresponding to 3-day forecasts with February 19, 1976, 02 initial conditions.

Fig. 6 Same as Fig. 5 but 500 mb geopotential heights.

Fig. 7 Same as Fig. 5 but for February 1, 1976, 02 initial conditions.
REFERENCES


Arakawa, A., 1972: Design of the UCLA GCM, Report No. 7, Dept. of Meteorology, UCLA.

Arakawa, A. and Y. Mintz, 1974: The UCLA atmospheric GCM. Workshop Notes, Dept. of Meteorology, UCLA.

Atlas, R., M. Halem and M. Gelil, 1979: Subjective evaluation of the combined influence of satellite temperature sounding data and increased model resolution on numerical weather forecasting. To be presented at the 4th Conference on Numerical Weather Prediction, Silver Spring, MD, October.

Campana, K., 1976: Real data experiments with a fourth-order version of the operational seven-layer model. NMC Office Note 186.


NAME TAG TYPE ADD | I | C | U | S | D | IM | N | P | S | F | D | F
---|---|---|---|---|---|---|---|---|---|---|---|---
00000 | 00000 | 000E70 | 00090C | 00090C | 00090C | 000030 | 00004C | 000030 | 000030 | 000030 | 000030
00000 | 00030 | 00004C | 00090C | 00090C | 00090C | 000030 | 00004C | 000030 | 000030 | 000030 | 000030
00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000 | 00000

SIZE OF PROGRAM: 32116C HEXADECMAL BYTES
<p>| NAME   | TYPE | REL. | ADDR. | ME | NAME   | TYPE | REL. | ADDR. | ME | NAME   | TYPE | REL. | ADDR. | ME | NAME   | TYPE | REL. | ADDR. | ME | NAME   | TYPE | REL. | ADDR. | ME |
|--------|------|------|-------|----|--------|------|------|-------|----|--------|------|------|-------|----|--------|------|------|-------|----|--------|------|------|-------|----|--------|------|------|-------|----|
| JMIN   | I<em>4  |     | N</em>0   |   | JMAX   | I<em>4  |     | N</em>0   |   | JSUN   | I<em>4  |     | N</em>0   |   | SMTH   | I<em>4  |     | N</em>0   |   | NSEQ   | I<em>4  |     | 000000 |   | NLF    | I</em>4  |     | 000004 |   | MATSNK | I<em>4  |     | 00002C |   | NSMI   | I</em>4  |     | 000030 |   |</p>
<table>
<thead>
<tr>
<th>NAME OF COMMON BLOCK</th>
<th># of BLOCKCOMMON</th>
<th>005E24 HEXADECIMAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAR NAME</td>
<td>TYPE</td>
<td>REL</td>
</tr>
<tr>
<td>NSTT</td>
<td>R44</td>
<td>000010</td>
</tr>
<tr>
<td>SPTOL</td>
<td>000014</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td># of BLOCKCOMMON</td>
<td>003C30 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>VAR NAME</td>
<td>TYPE</td>
<td>REL</td>
</tr>
<tr>
<td>JSP</td>
<td>J4</td>
<td>000000</td>
</tr>
<tr>
<td>DUMMY</td>
<td>J4</td>
<td>0000139</td>
</tr>
</tbody>
</table>

EQUIVALENCED VARIABLES WITHIN THIS COMMON GROUP:

<table>
<thead>
<tr>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
</tr>
</thead>
<tbody>
<tr>
<td>00007D</td>
<td>0024F0</td>
<td>001980</td>
<td>00000</td>
</tr>
</tbody>
</table>

NAME OF COMMON BLOCK | # of BLOCKCOMMON | 001000 HEXADECIMAL BYTES |
| VAR NAME | TYPE | REL | ADDR | VAR NAME | TYPE | REL | ADDR | VAR NAME | TYPE | REL | ADDR | VAR NAME | TYPE | REL | ADDR | VAR NAME | TYPE | REL | ADDR | VAR NAME | TYPE | REL | ADDR |
| PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 | PU | R44 | 000000 |

EQUIVALENCED VARIABLES WITHIN THIS COMMON GROUP:

<table>
<thead>
<tr>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
</tr>
</thead>
<tbody>
<tr>
<td>00007D</td>
<td>0024F0</td>
<td>001980</td>
<td>00000</td>
</tr>
</tbody>
</table>

NAME OF COMMON BLOCK: COMMON INFORMATION
<table>
<thead>
<tr>
<th>LABEL</th>
<th>ADDR</th>
<th>LABEL</th>
<th>ADDR</th>
<th>LABEL</th>
<th>ADDR</th>
<th>LABEL</th>
<th>ADDR</th>
<th>LABEL</th>
<th>ADDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>002F6</td>
<td>10</td>
<td>002F6</td>
<td>20</td>
<td>0035A</td>
<td>30</td>
<td>0035A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*OPTIONS IN EFFECT*  
NAME = MAIN, OPT = 02, INCHCT = 55, SIZE = 100K.

*OPTIONS IN EFFECT*  
SOURCE, EBCDIC, NOIST, NODECK, LOAD, MAP, NOEDIT, ID, NOXREF

*STATISTICS*  
SOURCE STATEMENTS = 71  
PROGRAM SIZE = 1660

*STATISTICS*  
NO DIAGNOSTICS GENERATED

00K BYTES OF CORE NOT USED

LEVEL 19, APR 71  
OS/360 FORTRAN M AT GISS  
DATE 12/12/79-073104

<table>
<thead>
<tr>
<th>LABEL</th>
<th>ADDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0035A</td>
</tr>
<tr>
<td>50</td>
<td>0035A</td>
</tr>
<tr>
<td>60</td>
<td>0035A</td>
</tr>
<tr>
<td>70</td>
<td>0035A</td>
</tr>
</tbody>
</table>
C*** COPY INPUTZ NAMELIST CNCG CORE TAPE AND TITLE PAGE

ISN 0104
WRITE (3,903) XLABEL

ISN 0105
WRITE (3,904) XLABEL

C*** READ INPUTZ NAMELIST CNCG CORE TAPE AND TITLE PAGE

ISN 0106
20 READ (3,904) RECORD

ISN 0107
WRITE (3,905) RECORD

ISN 0108
IF(RECOR Bit) NE ANDE) GO TO 20

ISN 0109
REWIND 11

ISN 0111
READ (11,INPUTZ)

ISN 0112
IF(START=EOx)=8) WRITE (11,930) START

ISN 0114
IF(START=EOx)=8 AND PZEME=0) WRITE (11,930) START

C*** READ IN VARIABLE ALGOC

ISN 0116
READ(*100) (ALBEDO(i,j),J=1,1,NJP) I=1,IM

ISN 0117
READ IN SURFACE GEOPOTENTIAL

ISN 0118
READ(17) (PHS(i,j),J=1,1,NJP) I=1,IM

ISN 0119
READ(6) (PH(i,j),J=1,1,NJP) I=1,IM

ISN 0120
1900 FORMAT(*4,11,*) J=1,1,NJP

ISN 0121
WRITE(*100) (ALBEDO(i,j),J=1,1,NJP)

ISN 0122
1941 FORMAT(*4,11,*) ALBEDO(i,j)=/2/(1/1,3613/7)

ISN 0123
DAYS = TAU + DINT * TAU

ISN 0124
IF(PZEME EOx)=1) PAUSE ' MOUNT TAPE5

ISN 0125
CALL SWITCH (5,595)

ISN 0127
IF(KSM5 EOx)=1) START=8

ISN 0129
KTR=8

ISN 0130
C * * * ON INITIAL START ENDFILE MODEL OUTPUT TAPE

ISN 0132
GO TO (32,120,120,40,110,110,40,50,90,1) START

ISN 0133
30 STOP

C*** SET DEPENDENT QUANTITIES

ISN 0134
40 DLON2=20.0PIJ/M

ISN 0135
LAT=DP1/3

ISN 0136
FIN=3

ISN 0137
KMA=40/LAY+5

ISN 0138
KMS=M-3

ISN 0139
IN=6/46

ISN 0140
JSPP=JS+1

ISN 0141
JNP=JNP+1

ISN 0142
JMP1 = JNP + 1

ISN 0143
JMP2 = JNP + 1

ISN 0144
JSB=1

ISN 0145
JSBPI = JSB + 1

ISN 0146
JNB=2

ISN 0147
JNB1=JNB-1

ISN 0148
NLAY=NLAY-1

ISN 0149
NLAY1=NLAY1

C --- IDFOSI(i)xNE(i) = 1 GO TO 60

ISN 0150
DO 50 L=1,NLAY

ISN 0151
50 DSIGLL=1/LAY

ISN 0152
50 SIGLL=4/0

ISN 0153
DO 70 L=1,NLAY

ISN 0154
SICEG=L+SIGLL+DSIGLL

ISN 0155
70 SIGLL=5*(SIGLL+SIGLL(L))

ISN 0156
DO 80 L=1,NLAYM

ISN 0157
80 DSIGLL=SIGLL(L)+SIGLL

ISN 0158
IF(START=EOx)=6) GO TO 120

ISN 0159
IF(START=EOx)=7) GO TO 110

C*** MACHINE CHECK RESTART, I = START = 6 OR 9

ISN 0162
90 IF(KSM5 EOx)=1) GO TO 100

ISN 0164
WRITE (*10,906)

ISN 0166
READ (*10,907) TAU

ISN 0167
WRITE (*10,908) TAU

ISN 0168
GO TO 120

C*** INITIAL CONDITIONS FROM UNIT 12, I = START = 5, 6, OR 7

ISN 0169
110 KTR=12

ISN 0170
40 IFG=1

ISN 0171
TAP=TAUI

ISN 0172
IF(TAPLTGxEOx) TAP=TAUI

ISN 0174
TAPM=DISK

ISN 0175
IF(PZEME EOx)=1 GO TO 120

ISN 0176
WRITE (*10,909)

ISN 0177
READ (*10,910) TAPM

ISN 0178
WRI Te (3,910) TAPM

C READ TAPE ON UNIT KTR

ISN 0179
GO TO 120

C*** READKTR,ERR=000,END=130)TAUX,C1,(P[XJ],J,J,J,J,NJP)1,(I,IM)

ISN 0180
ISN 0190
CALL EXIT

ISN 0191
130 IF (START GE SIGATO) GOTO 830

ISN 0194
BACKSPACE KTR

ISN 0194
TAU = TAU

ISN 0194
TAU = TAU

ISN 0196
DO 145 J=1,JM

ISN 0196
BACKSPACE KTR

ISN 0197
145 GOTO 140

ISN 0201
IF (START EQ 0) OR (START EQ 7) GC TO 160

ISN 0205
COPY C ARRAY FROM CI ARRAY

ISN 0209
READ (11, INPUT?)

ISN 0210
160 IF (KTR EQ 0) CI141=CI111

ISN 0210
IF (KTR EQ 0) XIABEL(2)=XIABEL(2)

ISN 0210
IF (KTR EQ 0) XIABEL(2)=TAPNUM

C### CALULATE DISTANCE PROJECTION ARRAYS

ISN 0216
150 FJEN=4*JLJNPI

ISN 0217
DO 200 J=J1,JNP

ISN 0218
200 LAT/JJ1OLAT(J-JFJEN)

ISN 0219
DO 210 J=J1,JNP

ISN 0220
SINJ=SIMULAT(JJ)

ISN 0221
210 CDJ(J)=COSLAT(JJ)

ISN 0222
DO 220 J=J1,JNP

ISN 0223
220 DYPJ=RAD*DOLN*COSLJ

ISN 0224
DO 230 J=J1,JNP

ISN 0225
DYIJ=RAD*DOL

C 230 DUVIJ=RADDOLAT(J-JLJNPI)

ISN 0226
CONTINUE

ISN 0227
JMP 102 = [JM + 1/2]

ISN 0228
JNP+1 = JNP + 1

ISN 0229
DO 240 J=2,JMP102

ISN 0230
DYP(J)=DYP(J-12)+DP;J+DYIJ

ISN 0231
240 DYP(J+1)=DYP(J)

ISN 0232
DYP(J+12)=0

ISN 0233
DYP(J+13)=3

ISN 0234
WHITE(J,2200)(JXP1(J)+J)=JNP

ISN 0235
2200 FORMAT(1X, DXP1,ﾉF/14.8,10(J=2,12))

C 0236
RETURN

C * * AUG 23 1977 ADDITIONS HERE TILL RETURN

C 0237
ENTRY INPUT4

ISN 0238
INC = 1

ISN 0239
BETA = 1 - 2 * ALPHA

ISN 0240
P506 = 54 / 64

ISN 0241
F17 = 17 / 16

ISN 0242
F9 = 325 / 961

ISN 0243
R = 1445

ISN 0244
RAD = 3 * RAD * IM

ISN 0245
CON1 = 4 * COTAN*OLAT / RAD

ISN 0246
CON2 = -COTAN*OLAT / RAD

ISN 0247
CON1 = 1 / (RAD+DOLAT)

ISN 0248
INC = 1

ISN 0249
JDR1(J-12)=2

ISN 0250
JDR1(J+12)=3

ISN 0251
JDR1(J+21)=JM

ISN 0252
JDR1(J+21)=JM

ISN 0253
JDR1(J+12)=2

ISN 0254
JDR1(J+11)=3

ISN 0255
JDR1(J+12)=MOD(JM+1/2,1)

ISN 0256
JDR1(J+12)=MOD(JM+2/2,1)

ISN 0257
MODP(1,1)=2

ISN 0258
MODP(1,1)=3

ISN 0259
MODP(1,2)=MOD(JM+1/2,1)

ISN 0260
MODP(1,2)=MOD(JM+2/2,1)

JNP=JM1

ISN 0261
JDR2(J)=2

ISN 0262
JDR2(J)=2

ISN 0263
JDR2(J)=2

ISN 0264
JDR2(J)=2

ISN 0265
JDR2(J)=2

ISN 0266
ADLDP = 12 * RAD * DOLAT * DOLN

ISN 0267
IMD=IM2/2

ISN 0268
IMDP = IMD+1

ISN 0269
JN2 = JM - 1

ISN 0270
JMS2 = JM - 2

ISN 0271
JMS2 = JM

ISN 0272
JMIN = 2 + JMS

ISN 0273
JMAX = JM - JMIN

ISN 0274
JSMN = JMAX + JMIN

ISN 0275
JWUM = 2 + JM

ISN 0276
JLIV = JMIN - 1
DO 301 J = 1, JMAX
JDN = J + 1
IF (JGEJMAX) JDN = JMAX + 1 - JMIN

DO 302 J = 1, JMO2
SMTH(J,J-1) = 1
IF (JGEJMAX) SMTH(J,J-1) = FLOA(1-IJNAX) / 6
IF (IIPGTE3) SMTH(J,J-1) = 0

310 CONTINUE

301 CONTINUE

PIKAPA = KAPA + 1
DO 320 L = 1, I, I4
DFFL(I) = PIKAPA * DSIGL(I)

320 CONTINUE

PSKAPA = EXPBYK(IDD,1)
THBAR = 200 / PSKAPA
CPH = ARAS / KAPA
CP = 0.85 * CP
CPH = CP * THBAR
PRE(r,9778)PSKAPA,THBAR,RGAS,KAPA,CAP,CPEC

9778 FORMAT(IIX,4PSKAPA,THBAR,RGAS,KAPA,CAP,CPEC)
PHSL(I) = PSI(1,1)
PHSL2(I) = PSI(1,1,1,1)
IM2 = IN / 2

330 CONTINUE

TWPI = 6.2563303715587
DO 306 I = 1, I2
AFLAT(I-1) = TWPI / FLOAT(IN)

306 CONTINUE

C *** ALONG = FLOAT(I-1) * TWPI / FLOAT(IN) - PI

C *** INITIAL VELOCITIES GIVEN ON A STAGGERED GRID.
C *** INTERPOLATION TO UNSTAGGERED GRID IS USED.
C *** MAKE USE OF EQUIVALENCING AND OVERLAPPING THE U ARRAY.
### Program Information

#### Name of Common Block

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>314 Hexadecimal BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0000400</td>
</tr>
</tbody>
</table>

#### Flags

<table>
<thead>
<tr>
<th>Name of Flag</th>
<th>Type</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>000040</td>
</tr>
</tbody>
</table>

#### Name of Common Block

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>5512 Hexadecimal BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>000000</td>
</tr>
</tbody>
</table>

#### Flags

<table>
<thead>
<tr>
<th>Name of Flag</th>
<th>Type</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>000000</td>
</tr>
</tbody>
</table>
### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
</tbody>
</table>

### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
</tbody>
</table>

### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
</tbody>
</table>

### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
</tbody>
</table>

### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
</tbody>
</table>

### Equivalence Variables within this Common Group

<table>
<thead>
<tr>
<th>Name of Common Block</th>
<th>Size of Block Common</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
<th>Variable Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>name of block name 1</td>
<td>000400 hexadecimal bytes</td>
<td>name of variable 1</td>
<td>name of variable 2</td>
<td>name of variable 3</td>
</tr>
<tr>
<td>name of block name 2</td>
<td>000500 hexadecimal bytes</td>
<td>name of variable 4</td>
<td>name of variable 5</td>
<td>name of variable 6</td>
</tr>
<tr>
<td>name of block name 3</td>
<td>000600 hexadecimal bytes</td>
<td>name of variable 7</td>
<td>name of variable 8</td>
<td>name of variable 9</td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>SIZE OF BLOCKCOMMON</td>
<td>HEXADECIMAL BYTES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NSEC</td>
<td>194</td>
<td>0000003</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NFMI</td>
<td>194</td>
<td>0000034</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>INVRIE OF BLOCKCOMMON</td>
<td>00000030</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR= NAME TYPE RELA ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C0MRT</td>
<td>194</td>
<td>0000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNR</td>
<td>194</td>
<td>0000019</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#5667E OF BLOCKCOMMON</td>
<td>00000030</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR= NAME TYPE RELA ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWITCH</td>
<td>194</td>
<td>0000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#7ITE OF BLOCKCOMMON</td>
<td>00000030</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR= NAME TYPE RELA ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITU</td>
<td>194</td>
<td>0000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#5657E OF BLOCKCOMMON</td>
<td>00000030</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR= NAME TYPE RELA ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IS5MTH</td>
<td>194</td>
<td>0000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#7FE OF BLOCKCOMMON</td>
<td>00000030</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAR= NAME TYPE RELA ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LABEL</td>
<td>ADDR</td>
<td>LABEL</td>
<td>ADDR</td>
<td>LABEL</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>--------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>20</td>
<td>001430</td>
<td>40</td>
<td>001020</td>
<td>70</td>
</tr>
<tr>
<td>60</td>
<td>001900</td>
<td>100</td>
<td>001498</td>
<td>120</td>
</tr>
<tr>
<td>140</td>
<td>001264</td>
<td>160</td>
<td>001086</td>
<td>180</td>
</tr>
<tr>
<td>240</td>
<td>001056</td>
<td>260</td>
<td>001038</td>
<td>280</td>
</tr>
<tr>
<td>320</td>
<td>001040</td>
<td>340</td>
<td>001020</td>
<td>360</td>
</tr>
</tbody>
</table>

*OPTIONS IN EFFECT*  
NAME= MAIN,OPT=02, LINES=555, SIZE=1000K,  
SOURCE= EBCDIC, NOLIST, NODECK, LOAD, MAP, NOEDIT, ID, NOKRF  
*STATISTICS*  
SOURCE STATEMENTS = 421 PROGRAM SIZE = 12944  
*STATISTICS*  
NO DIAGNOSTICS GENERATED  
******* END OF COMPILATION *******  
LEVEL 19.6 APR 71  
05/360 FORTRAN M AT GISS  
DATE 12/12/79-0703.20  

**772K BYTES OF CORE NOT USED**  

| ISN 0002 | 8400370 |  
| ISN 0003 | 8400380 |  
| ISN 0004 | 8400390 |  
| ISN 0005 | 8400400 |  
| ISN 0006 | 8400410 |  
| ISN 0007 | 8400420 |  
| ISN 0008 | 8400430 |  
| ISN 0009 | 8400440 |  
| ISN 0100 | 8400450 |  
| ISN 0101 | 8400460 |  
| ISN 0102 | 8400470 |  
| ISN 0103 | 8400480 |  
| ISN 0104 | 8400490 |  
| ISN 0105 | 8400500 |  
| ISN 0106 | 8400510 |  
| ISN 0107 | 8400520 |  
| ISN 0108 | 8400530 |  
| ISN 0109 | 8400540 |  
| ISN 0110 | 8400550 |  
| ISN 0111 | 8400560 |  
| ISN 0112 | 8400570 |  
| ISN 0113 | 8400580 |  
| ISN 0114 | 8400590 |  
| ISN 0115 | 8400600 |  
| ISN 0116 | 8400610 |  
| ISN 0117 | 8400620 |  
| ISN 0118 | 8400630 |  
| ISN 0119 | 8400640 |  
| ISN 0120 | 8400650 |  
| ISN 0121 | 8400660 |  
| ISN 0122 | 8400670 |  
| ISN 0123 | 8400680 |  
| ISN 0124 | 8400690 |  
| ISN 0125 | 8400700 |  
| ISN 0126 | 8400710 |  
| ISN 0127 | 8400720 |  
| ISN 0128 | 8400730 |  
| ISN 0129 | 8400740 |  
| ISN 0130 | 8400750 |  
| ISN 0131 | 8400760 |  
| ISN 0132 | 8400770 |  
| ISN 0133 | 8400780 |  
| ISN 0134 | 8400790 |  
| ISN 0135 | 8400800 |  
| ISN 0136 | 8400810 |  
| ISN 0137 | 8400820 |  
| ISN 0138 | 8400830 |  
| ISN 0139 | 8400840 |  
| ISN 0140 | 8400850 |  
| ISN 0141 | 8400860 |  
| ISN 0142 | 8400870 |  
| ISN 0143 | 8400880 |  
| ISN 0144 | 8400890 |  
| ISN 0145 | 8400900 |  
| ISN 0146 | 8400910 |  

70 WRITE [88, END=80, ERR=80] TAUR, C,[P[J], J=1, NJP, I=1, IM].
<table>
<thead>
<tr>
<th>JV</th>
<th>EN</th>
<th>RM</th>
<th>LN</th>
<th>UTAH</th>
<th>INHIT</th>
<th>EN</th>
<th>RM</th>
<th>LN</th>
<th>UTAH</th>
<th>INHIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVOT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>JAYS</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>DLT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>ITAU</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>TOPDAY</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>NSTEP</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>TAUD</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>DMTLTLT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>KAPA</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>NML</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>SINO</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>DUMMY</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>SIG</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>JPSU</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>JSN</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>DXP</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
<tr>
<td>PHIS</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
<td>EN</td>
<td>RM</td>
<td>LN</td>
<td>UTAH</td>
<td>INHIT</td>
</tr>
</tbody>
</table>

EQUIVALED VARIABLES WITHIN THIS COMMON GROUP

VARIABLE OFFSETE  VARIABLE OFFSETE  VARIABLE OFFSETE
U  0000F70  TS  001B90  V  001080  SH  000300  C  000000
MACHIN  00016C

NAME OF COMMON BLOCK = COMMON1 OF BLOCKCOMMON  00000000  00H  00000000  00H  00000000

EQUIVALED VARIABLES WITHIN THIS COMMON GROUP

VARIABLE OFFSETE  VARIABLE OFFSETE  VARIABLE OFFSETE
P  0034FD  T  0031FE  G  00194E

NAME OF COMMON BLOCK = COMMON3 OF BLOCKCOMMON  00000000  00H  00000000  00H  00000000

EQUIVALED VARIABLES WITHIN THIS COMMON GROUP

VARIABLE OFFSETE  VARIABLE OFFSETE  VARIABLE OFFSETE
J  0000AD  SH  0000AD  C  0000AD

NAME OF COMMON BLOCK = COMMON4 OF BLOCKCOMMON  00000000  00H  00000000  00H  00000000

EQUIVALED VARIABLES WITHIN THIS COMMON GROUP

VARIABLE OFFSETE  VARIABLE OFFSETE  VARIABLE OFFSETE
I  007F00  T  007F00  C  007F00
C *** (FLEAP+ANDNEXT STEP IS LEAP)
IF(MATSUN*MATSNX*EQ.0)GO TO 45
C *** (FLEAP+ANDNEXT CALL IS TO MATS PRED.)
IF(NPC*Eq.0)AND(MATSUN*Eq.116010)GO TO 58
ISN 0124 DO 29 = 1,IM1,INC
PT(J,J2J) = PSM(J,J2MD)
ISN 0126 28 CONTINUE
ISN 0127 DO 30 = 1,4
ISN 0128 DO 30 = 1,4,INC
ISN 0129 DO 30 = 1,4,INC
ISN 0130 G11,L,N,J2J = GSM(11L,N,J2MD)
ISN 0131 30 CONTINUE
ISN 0132 IF(NPC*Eq.1160 TO 63) GO TO 67
ISN 0133 58 CONTINUE
ISN 0134 DO 61 = INC,IM,INC
PT(I,J2J) = PT(I,J2J)
ISN 0136 61 CONTINUE
ISN 0137 DO 62 = INC,IM,INC
ISN 0138 DO 62 = INC,IM,INC
ISN 0139 G11,L,N,J2J = G11,L,N,J2MD
ISN 0140 62 CONTINUE
ISN 0141 GO TO 68
ISN 0143 C *** SMOOTH Q1,J2J FOR SMOOTH LEAPFROG TIME SCHEME
C *** ALPHA = 0.5 * NU
C *** BETAN = 1 - 2*ALPHA
ISN 0144 DO 48 = 1,INC,INC
PT(J,J2J) = ALPHA * PT(J,J2J) + PT(J,J2J)* (PSM(J,J2MD) + PT(I,J2J))
ISN 0146 48 CONTINUE
ISN 0147 DO 50 = 1,4
ISN 0148 DO 50 = 1,4,INC
ISN 0149 DO 50 = 1,4,INC
ISN 0150 G11,L,N,J2J = G11,L,N,J2MD
ISN 0151 50 CONTINUE
ISN 0152 C *** SOURCE TERM CORRECTION DUE TO LEAPFROG TIME SCHEME
C ***
ISN 0153 IF(MOD(NSTEP-NCM1,NCMP3))NE0160 TO 67
ISN 0155 DO 57 = 1,4
ISN 0157 DO 57 = 1,4,INC
ISN 0159 Q1(I,L,N,J2J) = Q1(I,L,N,J2J) + Q1(I,L,N,J2J) * DXYP(J2J)*PT(I,J2J)
ISN 0160 57 CONTINUE
ISN 0162 IF(MOD(NSTEP-NCM1,NCMP3))NE0160 TO 67
ISN 0164 CALL COMP3UT,WT,TF,SG,PM,PT,POL,STPOL,STPOL,STPOL,
* GTPOL,J2J
C *** CALL COMP3UND OR COM3 STEP
ISN 0165 IF(MATSUN*Eq.0)GO TO 45
ISN 0167 DO 65 = 1,4
ISN 0169 DO 65 = 1,4,INC
ISN 0170 Q1(I,L,N,J2J) = Q1(I,L,N,J2J) + DXYP(J2J)*PT(I,J2J)
ISN 0171 65 CONTINUE
ISN 0172 IF (J Lt, JN) GO TO 200
ISN 0174 J2J = J2J + 1
ISN 0175 J2MD2 = MOD(J2J-1,5)+1
ISN 0176 IF (J2J Ft, JN) GO TO 29
C *** POLES
C ***
ISN 0178 70 = 1
ISN 0179 IF (J*Eq.0) JN = 1
ISN 0181 COFF1 = -1.0 * 80 * 0
ISN 0182 C *** IFLEAP+ANDNEXT STEP IS LEAP
ISN 0184 IF(MATSUN*Eq.1)AND(MATSUN*Eq.0)GO TO 78
ISN 0186 PPROL(IN) = PPROL(IN)
ISN 0187 DO 75 = 1,4
ISN 0189 75 CONTINUE
ISN 0190 IF (G11POL(L,N,M) = GTPOL(L,N,M)) CONTINUE
ISN 0191 GO TO 54
ISN 0196 78 PPROL(IN) = PPROL(IN)
ISN 0198 DO 80 = 1,4
ISN 0190 80 CONTINUE
ISN 0197 IF(NPC*Eq.1160 TO 93)
GO TO 54
ISN 0199
ISN 0182
IF (NLAY.EQ.2) GO_TO 2440

ISN 0183
GO To 2440

ISN 0193
IF(IX4.EQ.1) GO TO 2450

ISN 0195
TPJX = TPJX + PJX

ISN 0208
CONTINUE

ISN 0209
SFLUX = SFLUX + SIGL

ISN 0211
CONTINUE

ISN 0214
SIGL = SIGL(1) + SIGL(2)

ISN 0216
TFLUX = TFLUX + SIGL

ISN 0218
CONTINUE

ISN 0219
RETURN

ISN 0220
CONTINUE

ISN 0221
M = 1

ISN 0224
IF (L.EQ.0) M = 2

ISN 0226
PSTRM = 0.0

ISN 0227
COEFF = (1.0)**(1/2)

ISN 0228
COEFF = -COEFF

ISN 0229
DO 2505 L = 1,NLAY

ISN 0230
DO 2505 N = 1,JS

ISN 0231
SUM(N) = 0.0

ISN 0233
CONTINUE

ISN 0234
CONTINUE

ISN 0235
CONTINUE

ISN 0236
CONTINUE

ISN 0237
CONTINUE

ISN 0239
CONTINUE

ISN 0240
CONTINUE

ISN 0241
CONTINUE

ISN 0242
CONTINUE
<table>
<thead>
<tr>
<th>LABEL ADDR</th>
<th>LABEL ADDR</th>
<th>LABEL ADDR</th>
<th>LABEL ADDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>2150</td>
<td>00046</td>
<td>2154</td>
<td>00043</td>
</tr>
<tr>
<td>110</td>
<td>00037</td>
<td>112</td>
<td>00033</td>
</tr>
<tr>
<td>2225</td>
<td>0007C</td>
<td>2250</td>
<td>00112</td>
</tr>
<tr>
<td>2240</td>
<td>00110</td>
<td>2255</td>
<td>00150</td>
</tr>
<tr>
<td>126</td>
<td>0014AC</td>
<td>2410</td>
<td>00168</td>
</tr>
<tr>
<td>7450</td>
<td>0019A</td>
<td>2450</td>
<td>00196</td>
</tr>
<tr>
<td>2490</td>
<td>00104</td>
<td>2520</td>
<td>00202</td>
</tr>
<tr>
<td>2510</td>
<td>00206</td>
<td>2539</td>
<td>00224</td>
</tr>
<tr>
<td>2540</td>
<td>00250</td>
<td>2599</td>
<td>0025C</td>
</tr>
</tbody>
</table>

**OPTIONS IN EFFECT**
- NAME = MAIN
- OPT = 0
- LINECNT = 55
- SIZE = 100K

**OPTIONS IN EFFECT**
- SOURCE
- EBCDIC
- NOSTMT
- NOCHK
- NOLOAD
- LSTR
- MAP
- NDE
- ID
- NOXREF

**STATISTICS**
- SOURCE STATEMENTS = 284
- PROGRAM SIZE = 1024

**STATISTICS**
- NO DIAGNOSTICS
- GENERATED

---

**LEVEL 19.6-APR 71**

**COMPILED OPTIONS**
- NAME = MAIN
- OPT = 0
- LINECNT = 55
- SIZE = 100K

**ISN**
- 0002
- 0003
- 0004
- 0005
- 0006
- 0007
- 0008
- 0009
- 0010
- 0011
- 0012
- 0013
- 0014
- 0015
- 0016
- 0017
- 0018
- 0019
- 0020
- 0021
- 0022
- 0023
- 0024
- 0025
- 0026
- 0027
- 0028

**DATE 12/12/79-078361**

**780K BYTES OF CORE NOT USED**

**COMPILED OPTIONS**
- NAME = MAIN
- OPT = 0
- LINECNT = 55
- SIZE = 100K

**ISN**
- 0002
- 0003
- 0004
- 0005
- 0006
- 0007
- 0008
- 0009
- 0010
- 0011
- 0012
- 0013
- 0014
- 0015
- 0016
- 0017
- 0018
- 0019
- 0020
- 0021
- 0022
- 0023
- 0024
- 0025
- 0026
- 0027
- 0028
| ISN 0086 | 3141 CONTINUE | B4837590 |
| ISN 0087 | C IF(J=EQ.INJ) GO TO 3030 | B4837590 |
| ISN 0089 | C C C COMPUTE PK AT LATITUDE JPS USING GISS METHOD | B4837660 |
| ISN 0090 | C (I+LL+NLAY) GO TO 3055 | B4837990 |
| ISN 0091 | 3057 DC 3010 I = 0,M | B4837640 |
| ISN 0092 | PII = SIGE(I) * (P1(JLP2) + PTOP) | B4837660 |
| ISN 0093 | PKII = PII * EXPBK(PLII) | B4837660 |
| ISN 0094 | LL = 1 | B4837660 |
| ISN 0095 | DC 3010 LLPI = 2, NLAYPI | B4837660 |
| ISN 0096 | PKI = SIGE(LLPI) * (P1(JLP2) + PTOP) | B4837790 |
| ISN 0097 | PKII = PKII * EXPBK(LLPI) | B4837790 |
| ISN 0098 | PKI(JLP2,JKP2) = (PKI - PKII) / (SIGFILL) * PI(JLP2) | B4837710 |
| ISN 0099 | PKII = PKII | B4837710 |
| ISN 0100 | LL = LLPI | B4837740 |
| ISN 0101 | 3010 CONTINUE | B4837750 |
| ISN 0102 | C C C COMPUTATION OF PHI (GEODETICAL) AT JPS FOR LNLAY. | B4837760 |
| ISN 0103 | C C PKSAP = 10000, SIGE(I)=0 | B4837770 |
| ISN 0104 | C C C HERE PHI IS NORMALIZED; SIGE=SIGE/STANDARD PHI-PHIBAR. | B4837790 |
| ISN 0105 | IF(II,LL+NLAY) GO TO 3055 | B4837600 |
| ISN 0106 | DO 3050 I = 1,MINC | B4837610 |
| ISN 0107 | PHI(I,JLP2MOD1) = PMI51(JLP2) + CTNP*(PKI(JLP2,JKP2) - PKSAP) | B4837620 |
| ISN 0108 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0109 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0110 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0111 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0112 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0113 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0114 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0115 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0116 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0117 | 3050 CONTINUE | B4837620 |
| ISN 0118 | 3055 DC 3060 I = 0,MINC | B4837620 |
| ISN 0119 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) + PHI(I,JLP2MOD1) | B4837620 |
| ISN 0120 | PHI(I,JLP2MOD1) = PHI(I,JLP2MOD1) | B4837620 |
| ISN 0121 | 3060 CONTINUE | B4837620 |
| ISN 0122 | C C FOR PHI(I,JL) USE INDEX(I). | B4837620 |
| ISN 0123 | C C COMPUTE W(I,JL,JPMD) FOR PRESSURE GRADIENT TERM AND SKY. | B4837620 |
| ISN 0124 | C C PIV FOR ENERGY TERM IN THERMODYNAMICS EQ. | B4837620 |
| ISN 0125 | 3065 IF(JP2.EQ.0 OR JP2.EQ.JP2) GO TO 3030 | B4838150 |
| ISN 0126 | DO 3070 I = 1,MINC | B4838040 |
| ISN 0127 | TBAR = THBAR * PKII(JLP2) | B4838040 |
| ISN 0128 | TPRIME = TLL(JLP2) - TBAR | B4838040 |
| ISN 0129 | 3070 CONTINUE | B4838040 |
| ISN 0130 | 3080 IF (JL.GE.2) GO TO 3030 | B4838040 |
| ISN 0131 | C C IF : I=11 RETURN AND COMPUTE PHI(I,JL2,JL2,JL2). | B4838230 |
| ISN 0132 | C C FOR SOUTH POLE CALCULATIONS | B4838230 |
| ISN 0133 | 3030 CONTINUE | B4838230 |
| ISN 0134 | 3070 IF(JL2.GE.2) GO TO 3030 | B4838230 |
| ISN 0135 | JP2MOD = JP2MOD + 1 | B4838230 |
| ISN 0136 | JP2MOD = JP2MOD + 1 | B4838230 |
| ISN 0137 | JP2MOD = JP2MOD + 1 | B4838230 |
| ISN 0138 | JP2MOD = JP2MOD + 1 | B4838230 |
| ISN 0139 | C C PRESSURE GRADIENT (F EQUATION) FOR J2= | B4838230 |
| ISN 0140 | 3032 DO 3031 LL = 1,NLAY | B4838230 |
| ISN 0141 | L = NLAYPL - LL | B4838230 |
| ISN 0142 | IF(JL.EQ.1) GO TO 3111 | B4838230 |
| ISN 0143 | 3080 IF (JL.GE.2) GO TO 3030 | B4838230 |
| ISN 0144 | C DD 3083 I=INC,MINC | B4838230 |
C
PIV = DXP(JJ) * V(J,J)
F1 = PIV(INDEX(J,J))
GI = PIV(INDEX(J,J))
P2 = PIV(J,J)
G2 = PIV(J,J)
WI = W(J,J)
VTIL(J,J) = VTIL(J,J) + DT * DXP(JJ) * P1(J,J) * (G)
S = [P2 - PHII(J,J)MOD] + WI / [G2 - P(J,J)]
S + [PHII(J,J)MOD] - FI + [G2 - P(J,J)]

ISN 0153

3083 CONTINUE

ISN 0154
C
• COMPUTATION OF THE PRESSURE GRADIENT FOR J1 = J2

C
• MOMENTUM EQs

ISN 0155
ISN 0156
ISN 0157
ISN 0158
ISN 0159
ISN 0160
C
• NEXT TWO LINES HAVE ADDED I's

ISN 0161
VTIL(J,J) = VTIL(J,J) + DT * DXPI(J,J) * P1(J,J) * (G)
S = [PHII(J,J)MOD] - PHII(J,J)MOD + WI / [P1(J,J)]
S = [P1(J,J)] + PHII(J,J)MOD - PHII(J,J)MOD + WI
S = [P1(J,J)] - P1(J,J)

ISN 0162
ISN 0163
ISN 0164
ISN 0165
ISN 0166
ISN 0167
C
• COMPUTATION OF THE PRESSURE GRADIENT FOR J1 = J2

C
• MENTAL IVI MOMENTUM Eq

ISN 0168
ISN 0169
ISN 0170
ISN 0171
ISN 0172
C
3100 CONTINUE

ISN 0173
C
3125 IF(J1+J2) = 0 GO TO 3031

ISN 0175
C
3031 IF(J1+J2) = 0 GO TO 3035

ISN 0176
ISN 0177
ISN 0178
ISN 0179
ISN 0180
ISN 0181
C
3150 CONTINUE

C
• CALCULATIONS AT THE POLES

C
• MOMENTUM AND TEMPERATURE AT SOUTH POLE J1 = 11 AND AT

C
• NORTH POLE J1 = 21

C
• (I = 11 CORRESPONDS TO THE LATITUDE LINE NEAREST THE POLE

C
• J1=MOD(I,4) = I-11)*0.51

C
• CONJ = 4*0781394RAD*WSIN(INDEG*DOLAT)

C
• CON2 = 0781394RAD*DOLAT

ISN 0182
C
3111 N = 1

ISN 0183
C
3111 N = 1

ISN 0185
ISN 0186
ISN 0187
ISN 0188
ISN 0189
C
3111 N = 1

C
• COROLIS TERM AT THE POLES

C
• F1 = F1EN(11) * PPOL(1)

ISN 0191
ISN 0192
ISN 0193
ISN 0194
ISN 0195
ISN 0196
ISN 0197
ISN 0198
C
3200 CONTINUE
<table>
<thead>
<tr>
<th>NAME OF COMMON BLOCK</th>
<th>#00_7E OF BLOCKCOMMON</th>
<th>00160C HEXADECIMAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAR. NAME TYPE REL. ADDR.</td>
<td>VAR. NAME TYPE REL. ADDR.</td>
<td>VAR. NAME TYPE REL. ADDR.</td>
</tr>
<tr>
<td>PU</td>
<td>R04</td>
<td>000000</td>
</tr>
<tr>
<td>EQUIVALENCED VARIABLES WITHIN THIS COMMON GROUP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VARIABLE OFFSETE</td>
<td>VARIABLE OFFSETE</td>
<td></td>
</tr>
<tr>
<td>GT</td>
<td>000040</td>
<td></td>
</tr>
<tr>
<td>GA</td>
<td>00168C</td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#05_7E OF BLOCKCOMMON</td>
<td>002004 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>SWITCH</td>
<td>I84</td>
<td>N0R.</td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#057E OF BLOCKCOMMON</td>
<td>0381A8 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>SDOT R04</td>
<td>000000</td>
<td></td>
</tr>
<tr>
<td>OMEGA I82</td>
<td>019E00</td>
<td></td>
</tr>
<tr>
<td>MM151S I84</td>
<td>N0R.</td>
<td></td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#057E OF BLOCKCOMMON</td>
<td>0381A8 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>RAO7</td>
<td>I82</td>
<td>N0R.</td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#0F7E OF BLOCKCOMMON</td>
<td>000000 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>ISGTH</td>
<td>I84</td>
<td>N0R.</td>
</tr>
<tr>
<td>NAME OF COMMON BLOCK</td>
<td>#1479E OF BLOCKCOMMON</td>
<td>000000 HEXADECIMAL BYTES</td>
</tr>
<tr>
<td>VAR. NAME TYPE REL. ADDR.</td>
<td>VAR. NAME TYPE REL. ADDR.</td>
<td>VAR. NAME TYPE REL. ADDR.</td>
</tr>
</tbody>
</table>
ISN 0037  DUMMY[i]=Q(i-L-N,J)+ALPHA[i][PINC[i-1,N,J]+(MINC[i-1,N,J)-
   Q(i-L,N,J)]=Q(i-L,N,J)  B4M02350
ISN 0038  Q(i-L,N,J) = Q(i-L,N,J)  B4M02360
ISN 0039  10 IP = INC + INC       B4M02370
ISN 0040  DO 20 IP = INC, INC     B4M02380
ISN 0041  20 G1 = O(i-L,N,J) + DUMMY(i)  B4M02390
ISN 0042  30 CONTINUE            B4M02400
ISN 0043  IMINC = IM - INC       B4M02410
ISN 0044  I = IM                B4M02420
ISN 0045  DO 35 IP = INC, INC, INC B4M02430
ISN 0046  DUMMY[i]=Q[i+1,N,J]+ALPHA[i][PINC[i+1,N,J]+(MINC[i+1,N,J)]-Q[i+1,N,J])  B4M02440
ISN 0047  IMINC = I               B4M02450
ISN 0048  35 I = IP               B4M02460
ISN 0049  DO 40 I = INC, INC, INC B4M02470
ISN 0050  P(I,J) = DUMMY(J)      B4M02480
ISN 0051  40 CONTINUE            B4M02490
ISN 0052  GO TO 60               B4M02500
C C C C FOURIER SMOOTHING NEAR POLES
C C C C1/2 VERSION 3 SMOOTH ON F AND T
ISN 0054  1  I + J = 1             B4M02520
ISN 0056  2  I - J = 1             B4M02540
ISN 0057  = (33NN) / 2             B4M02550
ISN 0058  DO 650 I = 1, NLAY      B4M02560
ISN 0059  DO 650 J = 1, NLAY      B4M02570
ISN 0061  650 CONTINUE            B4M02590
ISN 0062  DO 670 I = 1, NLAY      B4M02600
ISN 0063  TRANN = SSTR[I,N,J] * TRANN      B4M02610
ISN 0064  670 CONTINUE            B4M02620
ISN 0065  DO 680 I = 1, NLAY      B4M02630
ISN 0066  CALL FOURT2[CATAM+1,1,1,0]  B4M02640
ISN 0067  680 CONTINUE            B4M02650
ISN 0068  DO 690 I = 1, NLAY      B4M02660
ISN 0069  DO 690 J = 1, NLAY      B4M02670
ISN 0070  DATA[I] = DATA[I] / FLOAT[N]  B4M02680
ISN 0071  690 CONTINUE            B4M02690
ISN 0072  IF[I][J][N][M] = DATA[I]    B4M02700
ISN 0073  CALL SMSMAP               B4M02710
ISN 0074  60 CONTINUE            B4M02720
ISN 0075  100 CONTINUE            B4M02730
C C C C TRANSFORM TO SEA LEVEL PRESSURE
C C ISN 0076  ISN 0077  TSURF = T[0]SPL[I,J][T][T][N][NAY][T][T][N][NAY][T]  B4M02740
ISN 0078  SPL[I,J] = SPL[I,J] + PTOP + SSTR[I][PH[I,J],TSURF]  B4M02750
ISN 0079  110 CONTINUE            B4M02760
ISN 0080  IF[I][J][N][M] = 20 GO TO 781  B4M02770
ISN 0081  DO 760 I = 1, NLAY      B4M02780
ISN 0082  DO 760 J = 1, NLAY      B4M02790
ISN 0083  DATA[I] = SPL[I,J]     B4M02800
ISN 0084  760 CONTINUE            B4M02810
ISN 0085  CALL FOURT2[CATAM+1,1,1,0]  B4M02820
ISN 0086  DO 770 I = 1, NLAY      B4M02830
ISN 0087  DO 770 J = 1, NLAY      B4M02840
ISN 0088  TRANN = SSTR[I,N,J] * TRANN      B4M02850
ISN 0089  770 CONTINUE            B4M02860
ISN 0090  CALL FOURT2[CATAM+1,1,1,1]  B4M02870
ISN 0091  DO 780 I = 1, NLAY      B4M02880
ISN 0092  DO 780 J = 1, NLAY      B4M02890
ISN 0093  SLPIJ = DATA[I] / FLOAT[N]  B4M02900
ISN 0094  780 CONTINUE            B4M02910
ISN 0095  781 CONTINUE            B4M02920
C C C C SMOOTHING ALONG LONGITUDE [SEE SMSMAP]
C C ISN 0096  NSM = 8     B4M02930
ISN 0100  C[N] = 4.44444 NSM     B4M02970
ISN 0101  DO 130 I = 1, NLAY      B4M02980
ISN 0102  DATA[I] = SPL[I,J]     B4M02990
ISN 0103  130 CONTINUE            B4M03000
ISN 0104  DC[N] = 4.44444 NSM     B4M03010
ISN 0105  DO 145 I = 1, NLAY      B4M03020
ISN 0106  C[N] = 1     B4M03030
ISN 0107  IS = 1     B4M03040
ISN 0108  DO 145 IP = 1, NLAY      B4M03050
ISN 0109  CATA[I] = DATA[I] + CATA[I]     B4M03060
ISN 0110  I = IP     B4M03070
ISN 0111  145 CONTINUE            B4M03080
ISN 0112  IS = 1     B4M03090
ISN 0113  DO 150 I = 1, NLAY      B4M03100
ISN 0114  DATA[I] = CATA[I] - CATA[I]     B4M03110
ISN 0115  IS = 1     B4M03120
ISN 0116  150 CONTINUE            B4M03130
ISN 0117  140 DO 160 I = 1, NLAY      B4M03140
ISN 0118  160 CONTINUE            B4M03150
<table>
<thead>
<tr>
<th>NAME OF COMMON BLOCK</th>
<th># of BLOCKCOMMON</th>
<th>NAME OF BLOCKCOMMON</th>
<th>EQUVALENCES WITHIN THIS COMMON GROUP</th>
<th>VARIABLE OFFSET</th>
<th>VARIABLE OFFSET</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMON1</td>
<td></td>
<td>COMMON2</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>JPOL</td>
<td>000438</td>
<td>JPOL</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>INDO</td>
<td>000414</td>
<td>INDO</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>PSIN</td>
<td>000428</td>
<td>PSIN</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>DFIN</td>
<td>000232</td>
<td>DFIN</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>SFIN</td>
<td>000438</td>
<td>SFIN</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>CSOL</td>
<td>000428</td>
<td>CSOL</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>DYP</td>
<td>000232</td>
<td>DYP</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>DYP</td>
<td>000438</td>
<td>DYP</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>DYP</td>
<td>000428</td>
<td>DYP</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
<tr>
<td>DYP</td>
<td>000232</td>
<td>DYP</td>
<td>EQUVALENCES WITHIN THIS COMMON GROUP</td>
<td>VARIABLE OFFSET</td>
<td>VARIABLE OFFSET</td>
</tr>
</tbody>
</table>
C

* * * SHOULD PUT :=-1) FOR WIND SMOOTHING IN II26

C

C 0107
C 1124 DO 1126 J = 1, JN

C 0109
C DATA(J) = POT(I, J, NF)

C 0110
C DATA(J, NF) = POT(I, MD2, J, NF)

C 0111
C 1126 CONTINUE

C

C 0115
C JS = JMT2

C 0116
C DO 1130 = 1, JMT2

C 0117
C DATA(J) = DATA(JP1 - DATA(J))

C 0118
C JS = J

C 0119
C CONTINUE

C 0120
C CONTINUE

C 0121
C IF [NF * GT 0] GO TO 1154

C 0122
C DO 1150 J = 1, JN

C 0123
C SLPI(I, J) = SLPI(I, J) - DATA(J) / CY

C 0124
C CONTINUE

C 0125
C CONTINUE

C 0126
C IF [NF * GT 0] GO TO 1164

C 0127
C S1 = 0

C 0128
S2 = 0

C 0129
C DO 1150 J = 1, JN

C 0130
S1 = S1 + SLPI(I, J)

C 0131
S2 = S2 + SLPI(I, J)

C 0132
CONTINUE

C 0133
CONTINUE

C 0134
IF [NF * LT 0] GO TO 1164

C 0135
S1 = 0

C 0136
S2 = 0

C 0137
C DO 1150 J = 1, JN

C 0138
S1 = S1 + SLPI(I, J)

C 0139
S2 = S2 + SLPI(I, J)

C 0140
CONTINUE

C 0141
CONTINUE

C 0142
CONTINUE

C 0143
CONTINUE

C 0144
CONTINUE

C 0145
C DO 1150 J = 1, JN

C 0146
SLPI(I, J) = S1

C 0147
SLPI(I, JNP) = S2

C 0148
CONTINUE

C 0149
CONTINUE

C 0150
CONTINUE

C 0151
CONTINUE

C 0152
CONTINUE

C 0153
CONTINUE

C 0154
CONTINUE

C 0155
CONTINUE

C 0156
CONTINUE

C 0157
CONTINUE

C 0158
CONTINUE

C 0159
CONTINUE

C 0160
CONTINUE

C 0161
CONTINUE

C 0162
CONTINUE

C 0163
IF [NF * LT 0, NLAY] GO TO 1000

C

C

* * * SECOND MAIN J-LOOP

C

C

* * * TRANSFORM BACK TO SURFACE PRESSURE

C

C 0165
C MAXIT = 50

C 0166
C DO 2005 J = 1, JNP

C 0167
C DO 1320 I = 1, I

C 0168
C PHISX = PHIS(I, J)

C 0169
C PTOP = SLPI(I, J) - PTOP

C 0170
C DO 1330 NITX = 1, MAXIT

C 0171
P1 = PSCF

C 0172
T8 = POT(I, J) + EXPYK * PSCF * SIG(J) + PTOP

C 0173
T9 = POT(I, J) + EXPYK * PSCF * SIG(J) + PTOP

C 0174
T5URF = 100 * PSCF * T8, T9

C 0175
T5URF = SLPI(I, J) / SLPI(I, JNP) - PTOP

C 0176
IF [ABS [PSCF - 0.005 * WEWE = 1]] NO GO TO 1340

C 0177
CONTINUE

C

C 0178
CONTINUE

C 0179
PRINT 1335, MAXIT

C 0180
CONTINUE

C 0181
CONTINUE

C 0182
PRINT 7990, I

C 0183
PRINT 7990, I

C 0184
PRINT 7990, I

C 0185
CONTINUE

C 0186
CONTINUE

C 0187
CONTINUE

C 0188
CONTINUE

C 0189
CONTINUE

C 0190
CONTINUE

C 0191
CONTINUE

C

* * *