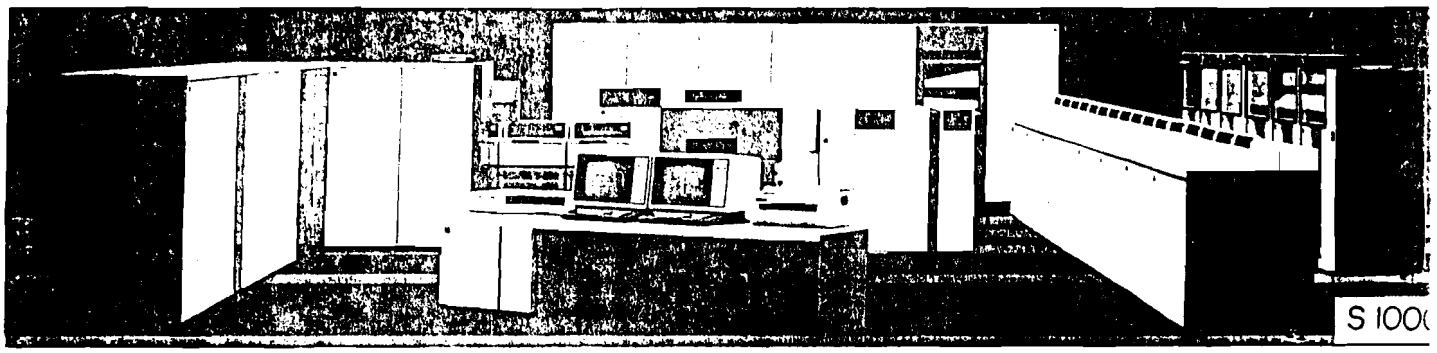
PN-ABU-008

# A DOCUMENTATION OF THE FSU REGIONAL PREDICTION MODEL




A.K. BOHRA
M. C. SINHA
ARUN KUMAR
JIAN SHENG
and
T.N. KRISHNAMURTI

DEPARTMENT OF METEOROLOGY
FLORIDA STATE UNIVERSITY
TALLAHASSEE, FLORIDA 32306

BEST AVAILABLE COPY

# A DOCUMENTATION OF THE FSU REGION PREDICTION MODEL

A.K. Bohra

M.C. Sinha

Arun Kumar

Jian Sheng

and

T.N. Krishnamurti

Department of Meteorology

Florida State University

Tallahassee, Florida 32306

BEST AVAILABLE COPY

Contents

Part I Model documentation

1. Model stucture and Boundary Conditions

1.1 Introduction

1.2 Model Structure

1.2.1 Vertical Structure

1.2.2 Horizontal Structure

1.3 Boundary Conditions

1.3.1 Vertical

1.3.2 Horizontal

2. Model Formulation

2.1 Introduction

2.2 Governing Equations

2.3 Transformation of Governing Equations

2.4 Governing Equations in Finite Difference Form

2.5 Solution of Helmholtz Equation

2.6 Interpolation Scheme

2.7 Integration Scheme and Boundary Conditions

3. Planetary Boundary Layer (PBL)

(a) Bulk Aerodynamic Calculation over ocean

(b) The roughness Parameter $Z_0$

(i) Ocean

(ii) Land

(c) Surface Fluxes from Similarity Analysis and Planetary Boundary Layer

(d) Coupling of Surface fluxes from the Surface Energy Balance and the Similarity Theory

(e) The Planetary Boundary Layer and the Vertical Distribution of Surface Fluxes

4. Parameterization of Shallow Convection

5. Parameterization of Deep Moist Convection

6. Large Scale Condensation

6.1 Different Formualtions of Large Scale Condensation Heating

(a) Removal of Supersaturation

(b) Simplest Direct Calculation

(c) A more Rigrous Formulation

7. Radiation

7.1 Introduction

7.2 Basic Features of the Scheme

7.3 Structure of the Radiation Scheme

7.4 Computations of Radiation Effects

7.4.1 Specification of Clouds

7.4.2 Longwave Radiation

7.4.3 Shortwave Radiation Flux

7.4.4 Radiative Heat Balance at the Earth's Surface

7.5 Input to the Radiation Scheme

8. Dry Convective Adjustment

PART I

MODEL DOCUMENTATION

# 1. Model Structure and Boundary Conditions

## 1.1 Introduction

In this chapter the structure and boundary conditions of the F.S.U. limited area grid point primitive equation model are presented.

## 1.2 Model Structure

### 1.2.1 Vertical Structure

The dimensionless $\sigma$ co-ordinate has been chosen in the vertical in which

$$\sigma = \frac{p}{p_s} \tag{1}$$

where p is the pressure and $p_s$ is the pressure at the earth's surface.

The vertical extent of the model ranges from $\sigma$=0.1 at top of the model to $\sigma$=1 at the earth's surface. There are 9 layers in the vertical. The thickness of each layer measured by pressure is uniform. Figure 1 shows the vertical structure of the model.

All the variables of the model (u,v,q,T) are defined at the centre of $\sigma$ layers (dashed levels). The vertical velocity $\dot{\sigma}$ is defined at the layer interfaces (solid levels). The layer values are represented by quantities with hat e.g. $\hat{u}$ whereas values at layer interface are represented by quantities without hat e.g. $\sigma$ etc.

### 1.2.2 Horizontal Structure

Figure 2 shows the forecast area of the model.

In the horizontal the model has been cast on a 2°x2° latitude longitude grid. The model domain covers the area 30°S to 50°N and

$\sigma_1=0.1$ $P_1,\ \dot{\sigma}_1=0$

$\tilde{u}_1,\ \tilde{v}_1,\ \tilde{T}_1,\ \tilde{q}_1$ $\tilde{\sigma}_1$

$\sigma_2=0.2$ $P_2,\ \dot{\sigma}_2$

$\tilde{u}_2,\ \tilde{v}_2,\ \tilde{T}_2,\ \tilde{q}_2$ $\tilde{\sigma}_2$

$\sigma_3=0.3$ $P_3,\ \dot{\sigma}_3$

$\tilde{u}_3,\ \tilde{v}_3,\ \tilde{T}_3,\ \tilde{q}_3$ $\tilde{\sigma}_3$

$\sigma_4=0.4$ $P_4,\ \dot{\sigma}_4$

$\tilde{u}_4,\ \tilde{v}_4,\ \tilde{T}_4,\ \tilde{q}_4$ $\tilde{\sigma}_4$

$\sigma_5=0.5$ $P_5,\ \dot{\sigma}_5$

$\tilde{u}_5,\ \tilde{v}_5,\ \tilde{T}_5,\ \tilde{q}_5$ $\tilde{\sigma}_5$

$\sigma_6=0.6$ $P_6,\ \dot{\sigma}_6$

$\tilde{u}_6,\ \tilde{v}_6,\ \tilde{T}_6,\ \tilde{q}_6$ $\tilde{\sigma}_6$

$\sigma_7=0.7$ $P_7,\ \dot{\sigma}_7$

$\tilde{u}_7,\ \tilde{v}_7,\ \tilde{T}_7,\ \tilde{q}_7$ $\tilde{\sigma}_7$

$\sigma_8=0.8$ $P_8,\ \dot{\sigma}_8$

$\tilde{u}_8,\ \tilde{v}_8,\ \tilde{T}_8,\ \tilde{q}_8$ $\tilde{\sigma}_8$

$\sigma_9=0.9$ $P_9,\ \dot{\sigma}_9$

$\tilde{u}_9,\ \tilde{v}_9,\ \tilde{T}_9,\ \tilde{q}_9$ $\tilde{\sigma}_9$

$\sigma_{10}=1$ $P_s,\ P_{10},\ \dot{\sigma}_{10}=0$

Fig. 1 Vertical Discretization

30°E to 150°E. Arakawa A type grid has been chosen where all the prognostic variables are unstaggered in the horizontal.

## 1.3 Boundary Conditions

### 1.3.1 Vertical

At top of the model

$\sigma = 0.1$

$\dot{\sigma} = 0$

At bottom of the model

$\sigma = 1$

$\dot{\sigma} = 0$

$Z = Z(\lambda,\phi)$ over land

$Z = 0$ over ocean

$T = T_s(\lambda,\phi)$ over ocean

where $T_s$ is sea surface temperature

### 1.3.2 Horizontal

Open upstream boundary conditions are used at the horizontal boundaries. Also at these boundaries

$\nabla \cdot \vec{V} = 0$ (Divergence)

$\dot{\sigma} = 0$ (Vertical velocity)

## 2. Model Formulation

2.1 Introduction In this chapter the dynamical formulation of the model is presented. A semi-Lagrangian semi-implicit scheme is used for the integration of the model. Semi Lagrangian method has been used in atmospheric models earlier (Krishnamurti, 1962, 1969, 1985; Mathur 1970, 1983,; Robert 1981, 1982). Semi implicit scheme has also been used for economy in time (Kudon 1978, Robert 1974).

Section 2.1 summarizes the governing differential equations of the model. In section 2.2 these equations are transformed into form suitable for semi-Lagrangian semi-implicit formalism. Finite difference forms of the transformed equations along with the appropriate boundary conditions are presented in section 2.3. In section 2.4 formulation of a 3-dimensional Helmholtz equation in one of the model parameters (P) and its solution are presented. Section 2.5 describes the solution of Helmholtz equation by matrix method. In section 2.6 the 4th order interpolation scheme used in the model is presented. Section 2.7 describes the integration schemes and the boundary conditions.

### 2.2 Governing equations

The governing equations for a Lagrangian formalism in $\sigma$ coordinate system are written as follows:

Equation of Motion

$$\frac{Du}{Dt} = -\dot{\sigma}\frac{\partial u}{\partial \sigma} + v\left(f+\frac{u}{a}\tan\phi\right) - g\frac{\partial z}{\partial x} - RT\frac{\partial \ln p_s}{\partial x} + F_x \equiv r_1 \tag{1}$$

$$\frac{Dv}{Dt} = -\dot{\sigma}\frac{\partial v}{\partial \sigma} - u\left(f+\frac{u}{a}\tan\phi\right) - g\frac{\partial z}{\partial y} - RT\frac{\partial \ln p_s}{\partial y} + F_y \equiv r_2 \tag{2}$$

Thermodynamic equation

$$\frac{DT}{Dt} = -\dot{\sigma}\frac{\partial T}{\partial \sigma} + kT\left[\frac{\dot{\sigma}}{\sigma} + \frac{D\ln p_s}{Dt}\right] + \frac{H+D_\theta}{C_p} \tag{3}$$

Mass Continuity equation

$$\frac{D\ln p_s}{Dt} = -\frac{\partial\dot{\sigma}}{\partial\sigma} - \nabla\cdot\vec{V} \tag{4}$$

Moisture Continuity Equation

$$\frac{Dq}{Dt} = -\dot{\sigma}\frac{\partial q}{\partial\sigma} + e - r + D_q \tag{5}$$

Equations (1)-(5) form the prognostic equations of the model. If equation (5) is dropped the remaining equations form the prognostic equations of the dry version of the model.

To close the system of equations (1)-(5), the following two diagnostic equations are used

Hydrostatic equation

$$\frac{\partial z}{\partial\sigma} = -\frac{RT}{g\sigma} = -\frac{C_p\theta}{g}\frac{\partial\pi^k}{\partial\sigma} \tag{6}$$

Equation of State

$$\alpha = \frac{RT}{p} = \frac{RT}{\sigma p_s} \tag{7}$$

The quantity $\pi^k$ in equation (6) is obtained from the Poisson Equation:

$$\frac{T}{\theta} = \pi^k = \left[\frac{p}{1000\text{mb}}\right]^k \tag{8}$$

On integrating equation of mass continuity in the vertical, Equation for surface pressure tendency is obtained

$$\frac{\partial p_s}{\partial t} = -\left[\frac{1}{1-\sigma_T}\right]\int_{\sigma_T}^{1}\nabla\cdot(p_s\vec{V})d\sigma \tag{9}$$

On eliminating $\frac{\partial p_s}{\partial t}$ between (4) and (9) equation for vertical velocity is obtained.

$$\frac{\partial\dot{\sigma}}{\partial\sigma} = (\tilde{\vec{V}} - \vec{V})\cdot\nabla\ln p_s + \nabla\cdot(\tilde{\vec{V}} - \vec{V}) \tag{10}$$

where

$$\tilde{\vec{V}} = \frac{1}{(1-\sigma_T)} \int_{\sigma_T}^{1} \vec{V} d\sigma \tag{11}$$

The operator $\frac{D}{Dt}$ in equations (1)-(5), which represents the Lagrangian operator, can be expressd, in special polar co-ordinate system as

$$\frac{D}{Dt}() = \frac{\partial}{\partial t}() + \frac{u}{a\cos\phi}\frac{\partial}{\partial\lambda} + \frac{v}{a}\frac{\partial}{\partial\phi}() \tag{12}$$

or, in Cartesian co-ordinate system as

$$\frac{D}{Dt}() = \frac{\partial}{\partial t}() + u\frac{\partial}{\partial x}() + v\frac{\partial}{\partial y}() \tag{13}$$

The list of variables in the governing equations (1)-(13)

| | |
|---|---|
| $\sigma$ | dimensionless vertical coordinate |
| p | pressure |
| $p_s$ | pressure at the earth's surface |
| u | x-component of wind vector |
| v | y-component of wind vector |
| t | time |
| $\dot{\sigma}$ | (= $d\sigma/dt$) vertical velocity in sigma coordinate |
| f | (= $2\ \Omega\ \sin\phi$) Coriolis parameter |
| $\Omega$ | earth's rotation rate |
| $\phi$ | latitude, postive northward from equator |
| $\lambda$ | longitude, positive eastward from Greenwich |
| a | earth's radius |
| g | gravity |
| z | height |
| R | Specific gas constant (for dry air) |
| T | Temperature |
| $F_x$ | x-component of surface momentum flux |

| | |
|---|---|
| $x$ | x-coordinate |
| $y$ | y-coordinate |
| $F_y$ | y-component of surface momentum flux |
| $k$ | $= R/C_p$ |
| $C_p$ | Specific heat (for dry air) at constant pressure |
| $H$ | Vertical eddy flux of heat |
| $D_\theta$ | diabatic heating (cooling) per unit mass |
| $\theta$ | potential temperature |
| $q$ | specific humidity |
| $e$ | vertical eddy flux of water vapor |
| $r$ | rainfall |
| $D_q$ | decrease (increase) of water vapor per unit mass due to condensation (evaporation) |
| $\vec{V}$ | Horizontal wind vector |
| $\sigma_T$ | $\sigma$ coordinate value at top of the model |
| $\alpha$ | specific volume |
| $\pi$ | $=\left[\frac{p}{1000\text{mb}}\right]^k$ |

Operators

| | |
|---|---|
| $(\tilde{\ })$ | vertical integration operator $= \frac{1}{1-\sigma_T}\int_{\sigma_T}^{1}(\quad)d\sigma$ |
| $\frac{D}{Dt}$ | Horizontal individual change operator |

## 2.3 Transformation of Governing Equations

A one step, fourth order, semi-Lagrangian advection scheme is used to integrate the model. A special feature of this scheme is that it accounts for changes in accelerations of dependent variables and

advecting velocity over the trajectory traced by the parcel in time step $\Delta t$. The inclusion of this formulation of the advective processes is expected to give better prediction of the phase and amplitude of rapidly developing disturbances as is generally the case in tropics. From physical point of view it is a direct approach. To make this scheme economical this scheme is coupled with semi-implicit time integration scheme.

Equations (1) to (5) are transformed into form suitable for the semi-Lagrangian semi-implicit formalism using the following definitions.

For any variable F,

$$F^* = \frac{\int\int F \cos\phi \, d\lambda d\phi}{\int\int \cos\phi d\lambda d\phi} \tag{14}$$

gives the domain average over the model domain.

$$F' = F - F^* \tag{15}$$

gives the deviation of the variable from its area average.

Combining the thermodynamic energy equation (3) and the mass continuity equation (4), a model parameter 'P' is defined as

$$P = gz + RT^* \ln p_s \tag{16}$$

also, $$P_s = gz_s + RT_s^* \ln p_s \tag{17}$$

where $z_s$ and $T_s^*$ are the surface topography and area averaged surface temperature respectively.

Lapse rate parameter '$\gamma$' is defined as

$$\gamma = \frac{R}{\sigma}\left[\frac{kT}{\sigma} - \frac{\partial T}{\partial \sigma}\right] \tag{18}$$

also

$$\gamma^* = \frac{R}{\sigma}\left[\frac{kT^*}{\sigma} - \frac{\partial T^*}{\partial \sigma}\right] \tag{19}$$

therefore

$$\gamma' = \gamma - \gamma^* \tag{20}$$

On applying (14)-(20) to (1)-(3) and (5) the latter can be written as

$$\frac{Du}{Dt} = -\dot{\sigma}\frac{\partial u}{\partial \sigma} + v\left(f+\frac{u}{a}\tan\phi\right) - g\frac{\partial z}{\partial x} - R(T^*+T')\frac{\partial \ln p_s}{\partial x} + F_x$$

$$= -\dot{\sigma}\frac{\partial u}{\partial \sigma} + v\left(f+\frac{u}{a}\tan\phi\right) - \frac{\partial}{\partial x}(gz+RT^*\ln p_s) - RT'\frac{\partial \ln p_s}{\partial x} + F_x$$

or $$\frac{Du}{Dt} + \frac{\partial P}{\partial x} = \dot{\sigma}\frac{\partial u}{\partial \sigma} + v\left(f+\frac{u}{a}\tan\phi\right) - RT'\frac{\partial \ln p_s}{\partial x} + F_x \equiv s_1 \tag{21}$$

Similarly

$$\frac{Dv}{Dt} + \frac{\partial P}{\partial y} = -\dot{\sigma}\frac{\partial v}{\partial \sigma} - u\left(f+\frac{u}{a}\tan\phi\right) - RT'\frac{\partial \ln p_s}{\partial y} + F_y \equiv s_2 \tag{22}$$

$$\frac{DT}{Dt} + \dot{\sigma}\frac{\partial T^*}{\partial \sigma} - kT^*\left[\frac{\dot{\sigma}}{\sigma} + \frac{D\ln p_s}{Dt}\right]$$

$$= -\dot{\sigma}\frac{\partial T'}{\partial \sigma} + kT'\left[\frac{\dot{\sigma}}{\sigma} + \frac{D\ln p_s}{Dt}\right] + \frac{H+D_\theta}{C_p}$$

or $$\frac{D}{Dt}(T-kT^*\ln p_s) - \frac{\gamma^*\sigma\dot{\sigma}}{R} = \dot{\sigma}\left[\frac{kT'}{\sigma} - \frac{\partial T'}{\partial \sigma}\right] - kT'\left[\frac{\partial \dot{\sigma}}{\partial \sigma} + \nabla\cdot\vec{V}\right] + \frac{H+D_\theta}{C_p}$$

$$\frac{D}{Dt}\left[-\frac{g\sigma}{R}\frac{\partial z}{\partial \sigma} - kT^*\ln p_s\right] - \frac{\gamma^*\sigma\dot{\sigma}}{R} = \dot{\sigma}\left[\frac{kT'}{\sigma} - \frac{\partial T'}{\partial \sigma}\right] - kT'\left[\frac{\partial \dot{\sigma}}{\partial \sigma} + \nabla\cdot\vec{V}\right] + \frac{H+D_\theta}{C_p}$$

$$-\frac{\sigma}{R}\frac{D}{Dt}\left[\frac{\partial p}{\partial \sigma} + \sigma\gamma^*\ln p_s\right] - \frac{\gamma^*\sigma\dot{\sigma}}{R} = \dot{\sigma}\left[\frac{kT'}{\sigma} - \frac{\partial T'}{\partial \sigma}\right] - kT'\left[\frac{\partial \dot{\sigma}}{\partial \sigma} + \nabla\cdot\vec{V}\right] + \frac{H+D_\theta}{C_p}$$

or $$\frac{D}{Dt}\left[\frac{1}{\gamma^*}\frac{\partial P}{\partial \sigma} + \sigma\ln p_s\right] + \dot{\sigma} = -\frac{\gamma'\dot{\sigma}}{\gamma^*} + \frac{kRT'}{\sigma\gamma^*}\left[\frac{\partial \dot{\sigma}}{\partial \sigma} + \nabla\cdot V\right] - \frac{R(H+D_\theta)}{C_p\sigma\gamma^*} \equiv s_3 \tag{23}$$

$$\frac{Dq}{Dt} = -\dot{\sigma}\frac{\partial q}{\partial \sigma} + e - r + D_q \equiv s_4 \tag{24}$$

In equation (21)-(24) all the non-linear terms appear on the right hand side.

A few more definitions required for semi-Lagrangian formulation are given below

$F = F(x,y,\sigma,t)$ at any point x, y and at current time step t (25)

$F^{+} = F(x,y,\sigma,t+\Delta t)$ at point x, y and at a future time step $t+\Delta t$ (26)

$F^{-} = F(x,y,\sigma,t-\Delta t)$ at point x, y and at a previous time step $t-\Delta t$ (27)

$F^{\circ} = F(x-a,y-b,\sigma,t)$ at time t at a point $\zeta$, $\eta$ from where the parcel arrives at x, y at time $t+\Delta t$ (28)

The semi-Lagrangian horizontal advection scheme is defined as

$$\frac{DF}{Dt} \equiv \frac{d_H F}{dt} = \frac{F^{+}-F^{\circ}}{\Delta t} \tag{29}$$

Semi-implicit time integration scheme is defined as

$$\bar{F} = \frac{F^{+}+F^{\circ}}{2} \tag{30}$$

Equation (29) shows that knowledge of forcing terms at time step $(t+\Delta t)$ is essential for applying semi-Lagrangian semi-implicit integration scheme. As the Lagrangian scheme involves interpolation of forcings from grid points to origin of the parcel, an iterative procedure has to be adopted involving computation of forcing terms F more than once. The proposition of using this physically sound but computationally inefficient scheme, then becomes less attractive. To avoid such a situation, a simple linear extrapolation of forcings is used

$$F = \frac{F^{+}+F^{-}}{2}$$

or $$F^{+} = 2F - F^{-} \tag{31}$$

(31) and (29) give

$$\{F\} = \frac{F^{+}+F^{\circ}}{2} = \frac{F^{\circ}+2F-F^{-}}{2} \tag{32}$$

On application of semi-Lagrangian and semi-implicit scheme the equations (21)-(24) and (4) transform into the following equations.

$$\frac{d_H u}{dt} + \frac{\partial P}{\partial x} = \{s_1\} \tag{33}$$

$$\frac{d_H v}{dt} + \frac{\partial \bar{P}}{\partial y} = \{s_2\} \tag{34}$$

$$\frac{d_H}{dt}\left[\frac{1}{\gamma^*}\frac{\partial P}{\partial \sigma} + \sigma \ln p_s\right] + \bar{\dot{\sigma}} = \{s_3\} \tag{35}$$

$$\frac{d_H q}{dt} = \{s_4\} \tag{36}$$

and

$$\frac{d_H \ln p_s}{dt} + \overline{\nabla \cdot V} + \overline{\frac{\partial \dot{\sigma}}{\partial \sigma}} = 0 \tag{37}$$

Using equations (29) and (32), equations (33)-(37) can be expressed as a set of 5 symbolic algebraic equations

$$\overset{+}{W}_j = \overset{\circ}{X}_j + Y_j + \bar{Z}_j \qquad j = 1(1)5 \tag{38}$$

As described earlier, the superscript + refers to future value at grid point, ∘ refers to current value at origin, - refers to past value at grid point. The subscript j indicates equations corresponding to equations (33)-(37).

The individual terms in the equation (38) can be expanded as

$\underline{W_j}$

$$W_1 = u + \frac{\Delta t}{2}\frac{\partial P}{\partial x} \tag{39}$$

$$W_2 = v + \frac{\Delta t}{2}\frac{\partial P}{\partial y} \tag{40}$$

$$W_3 = \frac{1}{\gamma^*}\frac{\partial P}{\partial \sigma} + \sigma \ln p_s + \frac{\Delta t}{2}\dot{\sigma} \tag{41}$$

$$W_4 = q \tag{42}$$

$$W_5 = \ln p_s + \frac{\Delta t}{2}\left[\nabla\cdot\vec{V} + \frac{\partial\dot{\sigma}}{\partial\sigma}\right] \tag{43}$$

$\underline{X_j}$

$$X_1 = u - \frac{\Delta t}{2}\left[\frac{\partial P}{\partial x} - s_1\right] \tag{44-a}$$

$$X_1 = u + \frac{\Delta t}{2} r_1 \quad \text{(see (1) )} \tag{44-b}$$

$$X_1 = \frac{\xi}{\Delta t} \tag{44-c}$$

$$X_2 = v - \frac{\Delta t}{2}\left[\frac{\partial P}{\partial y} - s_2\right] \tag{45-a}$$

$$X_2 = v + \frac{\Delta t}{2} r_2 \quad \text{(see (2) )} \tag{45-b}$$

$$X_2 = \frac{\eta}{\Delta t} \tag{45-c}$$

$$X_3 = \frac{1}{\gamma^*}\frac{\partial P}{\partial\sigma} + \sigma\ln p_s - \frac{\Delta t}{2}(\dot{\sigma} - s_3) \tag{46}$$

$$X_4 = q + \frac{\Delta t}{2} s_4 \tag{47}$$

$$X_5 = \ln p_s - \frac{\Delta t}{2}\left[\nabla\cdot\vec{V} + \frac{\partial\dot{\sigma}}{\partial\sigma}\right] \tag{48}$$

$\underline{Y_j}$

$$Y_1 = \Delta t s_1 \tag{49-a}$$

$$Y_1 = 2(X_1 - u) + \Delta t\left(g\frac{\partial z}{\partial x} + RT^*\frac{\partial}{\partial x}\ln p_s\right) \tag{49-b}$$

$$Y_2 = \Delta t s_2 \tag{50-a}$$

$$Y_2 = 2(X_2 - u) + \Delta t\left(g\frac{\partial z}{\partial y} + RT^*\frac{\partial}{\partial y}\ln p_s\right) \tag{50-b}$$

$$Y_3 = \Delta t S_3 \tag{51-a}$$

$$Y_3 = 2\left[X_3 - \frac{1}{\gamma^*}\left[g\frac{\partial z}{\partial\sigma} + \frac{kRT^*\ln p_s}{\sigma}\right]\right] + \Delta t\dot{\sigma} \tag{51-b}$$

$$Y_4 = \Delta t S_4 \tag{52-a}$$

$$Y_4 = 2(X_4 - q) \tag{52-b}$$

$$Y_5 = 0 \tag{53}$$

$\underline{Z_j}$

$$Z_1 = -\frac{\Delta t}{2} s_1 \tag{54-a}$$

$$Z_1 = -(X_1-u) - \frac{\Delta t}{2}\left[g\frac{\partial z}{\partial x} + RT^*\frac{\partial \ln p_s}{\partial x}\right] \tag{54-b}$$

$$Z_2 = -\frac{\Delta t}{2} s_2 \tag{55-a}$$

$$Z_2 = -(X_2-v) - \frac{\Delta t}{2}\left[g\frac{\partial z}{\partial y} + RT^*\frac{\partial \ln p_s}{\partial y}\right] \tag{55-b}$$

$$Z_3 = -\frac{\Delta t}{2} s_3 \tag{56-a}$$

$$Z_3 = -\left[ X_3 - \frac{1}{\gamma^*}\left[g\frac{\partial z}{\partial \sigma} + \frac{kRT^*}{\sigma} \ln p_s\right]\right] - \frac{\Delta t}{2}\dot{\sigma} \tag{56-b}$$

$$Z_4 = -\frac{\Delta t}{2} s_4 \tag{57-a}$$

$$Z_4 = -(X_4-q) \tag{57-b}$$

$$Z_5 = 0 \tag{58}$$

By eliminating various quantities from equations (39)-(43) a single equation for the model parameter P is obtained

$$\frac{\Delta t^2}{4}\nabla^2 P + \frac{\partial}{\partial\sigma}\left[\frac{1}{\gamma^*}\frac{\partial P}{\partial\sigma}\right] = \frac{\Delta t}{2}\vec{\nabla}\cdot W + \frac{\partial W_3}{\partial\sigma} - W_5 \equiv Q \tag{59}$$

where $\vec{W} = \hat{i}_\lambda W_1 + \hat{i}_\phi W_2$ (60)

Equation (59) is a 3 dimensional Helmholtz equation. This equation can be solved with appropriate boundary conditions for $P.W^+$ can be obtained from equation (38) making use of equations (44-a)-(58). equations (39)-(43) can then be used to give the prognostic variables. At upper and lower boundaries the vertical velocity $\sigma$=0 and at horizontal boundaries $\dot{\sigma}$=0 and $\nabla\cdot\vec{V}$= 0

At upper boundary, therefore, (41) becomes

$$\frac{1}{\gamma^*}\frac{\partial P}{\partial \sigma} = W_3 - \sigma \ln p_s$$

At lower boundary

$$\frac{1}{\gamma^*}\frac{\partial P}{\partial \sigma} = W_3 - \ln p_s \qquad \text{where } \sigma=1$$

2.4 Governing Equations in Finite Difference Form

From the vertical structure of the model, given in figure 1, the following parameters can be defined

N: Number of $\sigma$ layers

NP1: Number of $\sigma$ layer boundaries

NP1 = N+1

$\sigma_k$ (K=1(1)NP1): Value of $\sigma$ at layer boundaries

$\delta_k$ (K=1(1)N): Value of $\sigma$ at layers

$$\Delta\sigma_k = \delta_k - \delta_{k-1} \qquad k=2(1)N \tag{61}$$

$$\Delta\delta_k = \sigma_{k+1} - \sigma_k \qquad k=1(1)N \tag{62}$$

$$P_k = gz_k + RT_k^* \ln p_s \tag{63}$$

$$P_{NP1} = gz_s + RT_s^* \ln p_s \tag{64}$$

or

$$\ln p_s = \frac{P_{NP1}}{RT_s^*} - \frac{gz_s}{RT_s^*} \tag{65}$$

In writing the equations in finite difference form, the following boundary conditions are used

Top of the model

$$\sigma=0.1, \ \dot{\sigma}=0 \tag{66}$$

Bottom of the model

$$\sigma=1, \ \dot{\sigma}=0 \tag{67}$$

Also

$$\nabla\cdot\vec{V} = 0 \text{ at horizontal boundary points} \tag{68}$$

and $\dot{\sigma} = 0$ at horizontal boundary points (69)

The equations (39)-(43) are written in finite difference form as follows:

(i) <u>Equations of motion</u>

$$W_1 = u + \frac{\Delta t}{2}\frac{\partial P}{\partial x} \tag{70}$$

$$W_2 = v + \frac{\Delta t}{2}\frac{\partial P}{\partial y} \tag{71}$$

At general levels

$$\tilde{W}_{1k} = \tilde{u}_k + \frac{\Delta t}{2}\left[\frac{\frac{\partial P}{\partial x}_{k+1} + \frac{\partial P}{\partial x}_{k}}{2}\right] \tag{72}$$

$$\tilde{W}_{2k} = \tilde{v}_k + \frac{\Delta t}{2}\left[\frac{\frac{\partial P}{\partial y}_{k+1} + \frac{\partial P}{\partial y}_{k}}{2}\right] \tag{73}$$

(72) and (73) a single vector equation of motion is obtained

$$\vec{W}_k = \hat{i}_\lambda \tilde{W}_{1k} + \hat{i}_\phi \tilde{W}_{2k}$$

$$= \vec{V}_k + \frac{\Delta t}{2}\left[\frac{\nabla P_{k+1} + \nabla P_k}{2}\right] \tag{74}$$

Operating (74) with $\nabla\cdot$

$$\nabla\cdot\vec{W}_k = \nabla\cdot\vec{V}_k + \frac{\Delta t}{2}\left[\frac{\nabla^2 P_{k+1} + \nabla^2 P_k}{2}\right] \tag{75}$$

From (75)

$$\nabla\cdot\vec{W}_{k-1} = \nabla\cdot\vec{V}_{k-1} + \frac{\Delta t}{2}\left[\frac{\nabla^2 P_k + \nabla^2 P_{k-1}}{2}\right] \tag{76}$$

$\big((75) + (76)\big)/2$ gives

$$\frac{1}{2}\left(\nabla\cdot\vec{W}_k + \nabla\cdot\vec{W}_{k-1}\right) = \frac{\left(\nabla\cdot\vec{V}_k + \nabla\cdot\vec{V}_{k-1}\right)}{2} + \frac{\Delta t}{2}\left[\frac{\nabla^2 P_{k+1} + 2\nabla^2 P_k + \nabla^2 P_{k-1}}{4}\right] \tag{77}$$

At <u>upper boundary</u> ($k=1$, $\sigma=0.1$, $\dot{\sigma}=0$)

equation (75) takes the form

$$\nabla\cdot\vec{W}_1 = \nabla\cdot\vec{V}_1 + \frac{\Delta t}{2}\left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2}\right] \tag{78}$$

At <u>lower boundary</u> ($k=N$, $\sigma=1$, $\dot{\sigma}=0$)

(75) takes the form

$$\nabla\cdot\vec{W}_N = \nabla\cdot\vec{V}_N + \frac{\Delta t}{2}\left[\frac{\nabla^2 P_{NP1} + \nabla^2 P_N}{2}\right] \tag{79}$$

At <u>horizontal boundaries</u>, (75) takes the form

$$\frac{\Delta t}{2}\left(\nabla^2 P_{k+1} + \nabla^2 P_k\right) = 0 \tag{80}$$

$$\nabla\cdot\vec{V} = 0 = \nabla\cdot\vec{W} \text{ at horizontal boundaries}$$

(ii) Thermodynamic Energy Equation

$$W_3 = \frac{1}{\gamma^*}\frac{\partial P}{\partial\sigma} + \sigma \ln p_s + \frac{\Delta t}{2}\dot{\sigma} \tag{81}$$

At general levels

$$\tilde{W}_{3k} = \frac{1}{\tilde{\gamma}_k^*}\left[\frac{P_{k+1} - P_k}{\Delta\tilde{\sigma}_k}\right] + \tilde{\sigma}_k \ln p_s + \frac{\Delta t}{2}\left[\frac{\dot{\sigma}_{k+1} + \dot{\sigma}_k}{2}\right] \tag{82}$$

$$\tilde{W}_{3k-1} = \frac{1}{\tilde{\gamma}_{k-1}^*}\frac{P_k - P_{k-1}}{\Delta\tilde{\sigma}_{k-1}} + \tilde{\sigma}_{k-1}\ln p_s + \frac{\Delta t}{2}\frac{\dot{\sigma}_k + \dot{\sigma}_{k-1}}{2} \tag{83}$$

[(82) - (83)] / $\Delta\sigma_k$ gives

$$\frac{\tilde{W}_{3k} - \tilde{W}_{3k-1}}{\Delta\sigma_k} = \frac{1}{\Delta\sigma_k}\left[\frac{1}{\tilde{\gamma}_k^*}\frac{P_{k+1} - P_k}{\Delta\tilde{\sigma}_k} - \frac{1}{\tilde{\gamma}_{k-1}^*}\frac{P_k - P_{k-1}}{\Delta\tilde{\sigma}_{k-1}}\right]$$

$$+ \ln p_s + \frac{\Delta t}{2}\left[\frac{\dot{\sigma}_{k+1} - \dot{\sigma}_{k-1}}{2\Delta\sigma_k}\right] \tag{84}$$

At the <u>upper boundary</u>

(82) takes the form

$$\frac{1}{\tilde{\gamma}_1^*}\left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \tilde{\sigma}_1 \ln p_s + \frac{\Delta t}{2}\frac{\dot{\sigma}_2}{2} = \tilde{W}_{31}$$

or $$\frac{1}{\tilde{\gamma}_1^*}\left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \left[\sigma_1 + \frac{\Delta\tilde{\sigma}_1}{2}\right]\ln p_s + \frac{\Delta t}{2}\frac{\dot{\sigma}_2}{2} = \tilde{W}_{31}$$

or $$\frac{1}{\tilde{\gamma}_1^*}\frac{1}{\Delta\tilde{\sigma}_1}\left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \frac{\sigma_1}{\Delta\tilde{\sigma}_1}\ln p_s + \frac{1}{2}\ln p_s + \frac{\Delta t}{2}\frac{\dot{\sigma}_2}{2\Delta\tilde{\sigma}_1} = \frac{\tilde{W}_{31}}{\Delta\tilde{\sigma}_1} \tag{85}$$

At the lower boundary

$$\frac{1}{\tilde{\gamma}_N^*}\frac{1}{\Delta\tilde{\sigma}_N}\frac{P_{NP1}-P_N}{\sigma_{NP1}-\sigma_N}+\sigma_{NP1}\frac{\ln p_s}{\Delta\tilde{\sigma}_N}-\frac{1}{2}\ln p_s+\frac{\Delta t}{2}\frac{\sigma_N}{2\Delta\tilde{\sigma}_N}=\frac{\tilde{W}_{3N}}{\Delta\tilde{\sigma}_N} \tag{86}$$

At <u>horizontal boundaries</u> (82) takes the form

$$\frac{1}{\tilde{\gamma}_k^*}\left[\frac{P_{k+1}-P_k}{\Delta\tilde{\sigma}_k}\right]+\tilde{\sigma}_k\ln p_s=\tilde{W}_{3k} \tag{87}$$

At upper boundary $\frac{1}{\tilde{\gamma}_1^*}\left[\frac{P_2-P_1}{\sigma_2-\sigma_1}\right]+\tilde{\sigma}_1\ln p_s=\tilde{W}_{3k}$

At lower boundary $\frac{1}{\tilde{\gamma}_N^*}\left[\frac{P_{NP1}-P_N}{\sigma_{NP1}-\sigma_N}\right]+\tilde{\sigma}_N\ln p_s=\tilde{W}_{3N}$

Equation of Continuity

$$W_5=\ln p_s+\frac{\Delta t}{2}\nabla\cdot\vec{V}+\frac{\Delta t}{2}\frac{\partial\dot{\sigma}}{\partial\sigma} \tag{88}$$

At general levels

$$\tilde{W}_{5k}=\ln p_s+\frac{\Delta t}{2}\nabla\cdot\vec{V}_k+\frac{\Delta t}{2}\left[\frac{\dot{\sigma}_{k+1}-\dot{\sigma}_k}{\Delta\tilde{\sigma}_k}\right] \tag{89}$$

$$\tilde{W}_{5k-1}=\ln p_s+\frac{\Delta t}{2}\nabla\cdot\vec{V}_{k-1}+\frac{\Delta t}{2}\left[\frac{\dot{\sigma}_k-\dot{\sigma}_{k-1}}{\Delta\tilde{\sigma}_{k-1}}\right] \tag{90}$$

In the present case $\Delta\tilde{\sigma}_k=\Delta\tilde{\sigma}_{k-1}$

[(89) + (90)] / 2 gives

$$\frac{\tilde{W}_{5k}+\tilde{W}_{5k-1}}{2}=\ln p_s+\frac{\Delta t}{2}\left(\frac{\nabla\cdot\vec{V}_k+\nabla\cdot\vec{V}_{k-1}}{2}\right)+\frac{\Delta t}{2}\left[\frac{\dot{\sigma}_{k+1}-\dot{\sigma}_{k-1}}{2\Delta\tilde{\sigma}_k}\right] \tag{91}$$

using (77), (91) takes the form

$$\begin{aligned}\frac{\tilde{W}_{5k}+\tilde{W}_{5k-1}}{2}&=\ln p_s+\frac{\Delta t}{2}(\nabla\cdot\vec{W}_k+\nabla\cdot\vec{W}_{k-1})\\&-\left[\frac{\Delta t}{2}\right]^2\left[\frac{\nabla^2P_{k+1}+2\nabla^2P_{k+1}+\nabla^2P_{k-1}}{2}\right]\\&+\frac{\Delta t}{2}\left[\frac{\dot{\sigma}_{k+1}-\dot{\sigma}_{k-1}}{\Delta\tilde{\sigma}_k}\right]\end{aligned} \tag{92}$$

At <u>upper boundary</u> (89) takes the form

$$\tilde{W}_{51}=\ln p_s+\frac{\Delta t}{2}\left[\nabla\cdot\vec{W}_1-\frac{\Delta t}{2}\left[\frac{\nabla^2P_2+\nabla^2P_1}{2}\right]\right]+\frac{\Delta t}{2}\frac{\dot{\sigma}_2}{\Delta\tilde{\sigma}_1} \tag{93}$$

or

$$\ln p_s + \frac{\Delta t}{2}\frac{\dot{\sigma}_2}{\Delta\sigma_1} = \tilde{W}_{51} - \frac{\Delta t}{2}\nabla\cdot\vec{W}_1 + \left[\frac{\Delta t}{2}\right]^2\left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2}\right] \tag{94}$$

At lower boundary (89) gives

$$\tilde{W}_{5N} = \ln p_s + \frac{\Delta t}{2}\left[\nabla\cdot\vec{W}_N - \frac{\Delta t}{2}\left[\frac{\nabla^2 P_{NP1} + \nabla^2 P_N}{2}\right]\right] - \frac{\Delta t}{2}\frac{\dot{\sigma}_N}{\Delta\sigma_N} \tag{95}$$

or

$$\ln p_s - \frac{\Delta t}{2}\frac{\dot{\sigma}_N}{\Delta\sigma_N} = \tilde{W}_{5N} - \frac{\Delta t}{2}\nabla\cdot\vec{V}_N = \tilde{W}_{5N} - \frac{\Delta t}{2}\nabla\cdot\vec{W}_N + \left[\frac{\Delta t}{2}\right]^2\left[\frac{\nabla^2 P_{NP1} + \nabla^2 P_N}{2}\right] \tag{96}$$

At horizontal boundaries

$$\tilde{W}_{5k} = \ln p_s \tag{97}$$

By eliminating various terms from the finite difference forms of equations of motion, thermodynamic energy equation and the mass continuity equations a single 3-dimensional Helmholtz equation in the finite difference form can be obtained.

Helmholtz equation

$$\left[\frac{\Delta t}{2}\right]^2 \nabla^2 P + \frac{\partial}{\partial\sigma}\left[\frac{1}{\gamma^*}\frac{\partial P}{\partial\sigma}\right] = \frac{\Delta t}{2}\nabla\cdot\vec{W} + \frac{\partial W_3}{\partial\sigma} - W_5 \equiv Q \tag{98}$$

Combining Equations (77), (84) and (91) in appropriate way, Helmholtz equation in finite difference form can be written

$$\frac{1}{\Delta\sigma_k}\left[\frac{1}{\tilde{\gamma}_k^*}\left[\frac{P_{k+1} - P_k}{\Delta\sigma_k}\right] - \frac{1}{\tilde{\gamma}^*_{k-1}}\left[\frac{P_k - P_{k-1}}{\Delta\sigma_{k-1}}\right]\right]$$

$$+ \left[\frac{\Delta t}{2}\right]^2\left[\frac{\nabla^2 P_{k+1} + 2\nabla^2 P_k + \nabla^2 P_{k-1}}{4}\right]$$

$$= \frac{\Delta t}{2}\left(\frac{\nabla\cdot\vec{W}_k + \nabla\cdot\vec{W}_{k-1}}{2}\right) + \left[\frac{\tilde{W}_{3k} - \tilde{W}_{3k-1}}{\Delta\sigma_k}\right] - \left[\frac{\tilde{W}_{5k} + \tilde{W}_{5k-1}}{2}\right] \tag{99}$$

At upper boundary (99) gives, using (78), (85) and (94)

$$\frac{\Delta t}{2}\frac{\nabla\cdot\vec{W}_1}{2} + \frac{\tilde{W}_{31}}{\Delta\sigma_1} - \frac{\tilde{W}_{51}}{2}$$

$$= \frac{\Delta t}{2}\left[\frac{\nabla\cdot\vec{V}_1}{2}\right] + \left[\frac{\Delta t}{2}\right]^2\left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2x2}\right]$$

$$+ \frac{1}{\tilde{\gamma}_1^*} \frac{1}{\Delta\tilde{\sigma}_1} \left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \ln p_s + \frac{1}{2} \ln p_s + \frac{\Delta t}{2} \left[\frac{\dot{\sigma}_2}{2\Delta\tilde{\sigma}_1}\right]$$

$$- \frac{\ln p_s}{2} - \frac{\Delta t}{2} \left[\frac{\nabla\cdot\vec{V}_1}{2}\right] - \frac{\Delta t}{2x2} \left[\frac{\sigma_2}{\Delta\tilde{\sigma}_1}\right]$$

$$\frac{1}{\tilde{\gamma}_1^*} \frac{1}{\Delta\tilde{\sigma}_1} \left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \ln p_s + \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2}\right] \tag{100}$$

Substituting value of $\ln p_s$ in (100) from (65), the former gives

$$\frac{1}{\tilde{\gamma}_1^*} \frac{1}{\Delta\tilde{\sigma}_1} \left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \frac{P_{NP1}}{RT_S^*} + \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2x2}\right]$$

$$= \frac{\Delta t}{2} \frac{\nabla\cdot\vec{W}_1}{2} + \frac{\tilde{W}_{31}}{\Delta\tilde{\sigma}_1} - \frac{\tilde{W}_{51}}{2} + \frac{gz_s}{RT_S^*} \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \tag{101}$$

At lower boundary (79), (86), (96) and (65) give

$$- \frac{1}{\tilde{\gamma}_N^*} \frac{1}{\Delta\tilde{\sigma}_N} \left[\frac{P_{NP1} - P_N}{\sigma_{NP1} - \sigma_N}\right] - \frac{P_{NP1}}{RT^*_{NP1}} \frac{1}{\Delta\tilde{\sigma}_N} + \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_{NP1} + \nabla^2 P_N}{2x2}\right]$$

$$= - \frac{\tilde{W}_{3N}}{\Delta\tilde{\sigma}_N} - \frac{\tilde{W}_{5N}}{2} + \frac{\Delta t}{2} \frac{\nabla\cdot\vec{W}_N}{2} - \frac{gz_s}{RT^*_{NP1}} \frac{1}{\Delta\tilde{\sigma}_N} \tag{102}$$

## 2.5 Solution of Helmholtz equation

In the semi-Lagrangian semi-implicit scheme, as has been seen in earlier section, a 3-dimensional Helmholtz equation is obtained, which can be written as:

$$\left[\frac{\Delta t}{2}\right]^2 \nabla^2 P^+ \frac{\partial}{\partial\sigma} \left[\frac{1}{\gamma^*} \frac{\partial P}{\partial\sigma}\right] = \frac{\Delta t}{2} (\nabla\cdot\vec{W}) + \frac{\partial W_3}{\partial\sigma} - W_5 \equiv Q \tag{103}$$

To simplify the numerical solution of (103), it is transformed into a set of 2-dimensional linearly independent set of Helmholtz equations which can then be solved using standard relaxation techniques. For this purpose the above equation is written in finite difference form.

$$\frac{1}{\Delta\sigma_k} \left[ \frac{1}{\tilde{\gamma}_k^*} \left[\frac{P_{k+1} - P_k}{\Delta\tilde{\sigma}_k}\right] - \frac{1}{\tilde{\gamma}^*_{k-1}} \left[\frac{P_k - P_{k-1}}{\Delta\tilde{\sigma}_{k-1}}\right]\right]$$

$$+ \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_{k+1} + 2\nabla^2 P_k + \nabla^2 P_{k-1}}{4}\right]$$

$$= \frac{\Delta t}{2} \frac{(\nabla\cdot\vec{W}_k + \nabla\cdot\vec{W}_{k-1})}{2} + \frac{\tilde{W}_{3k} - \tilde{W}_{3k-1}}{\Delta\sigma_k} - \frac{\tilde{W}_{5k} + \tilde{W}_{5k-1}}{2} \qquad (104)$$

$$k=2(1)N$$

At upper boundary

$$\frac{1}{\tilde{\gamma}_1^*} \frac{1}{\Delta\tilde{\sigma}_1} \left[\frac{P_2 - P_1}{\sigma_2 - \sigma_1}\right] + \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \frac{P_{NP1}}{RT_S^*} + \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_2 + \nabla^2 P_1}{2x2}\right]$$

$$= \frac{\Delta t}{2} \frac{(\nabla\cdot\vec{W}_1)}{2} + \frac{\tilde{W}_{31}}{\Delta\tilde{\sigma}_1} - \frac{\tilde{W}_{51}}{2} + \frac{gz_s}{RT_S^*} \frac{\sigma_1}{\Delta\tilde{\sigma}_1} \qquad (105)$$

At lower boundary

$$- \frac{1}{\tilde{\gamma}_N^*} \frac{1}{\Delta\tilde{\sigma}_N} \left[\frac{P_{NP1} - P_N}{\sigma_{NP1} - \sigma_N}\right] - \frac{P_{NP1}}{RT_{NP1}^*} \frac{1}{\Delta\tilde{\sigma}_N} + \left[\frac{\Delta t}{2}\right]^2 \left[\frac{\nabla^2 P_{NP1} + \nabla^2 P_N}{2x2}\right]$$

$$= - \frac{\tilde{W}_{3N}}{\Delta\tilde{\sigma}_N} - \frac{\tilde{W}_{5N}}{2} + \frac{\Delta t}{2} \left[\frac{\nabla\cdot\vec{W}_N}{2}\right] - \frac{gz_s}{RT_{NP1}^*} \frac{1}{\Delta\tilde{\sigma}_N} \qquad (106)$$

At horizontal boundaries (104) becomes

$$\frac{1}{\Delta\sigma_k} \left[ \frac{1}{\tilde{\gamma}_k^*} \frac{P_{k+1} - P_k}{\Delta\tilde{\sigma}_k} - \frac{1}{\tilde{\gamma}_{k-1}^*} \left[\frac{P_k - P_{k-1}}{\Delta\tilde{\sigma}_{k-1}}\right] \right]$$

$$= \frac{\tilde{W}_{3k} - \tilde{W}_{3k-1}}{\Delta\sigma_k} - \frac{\tilde{W}_{5k} + \tilde{W}_{5k-1}}{2} \qquad (107)$$

Equations (104) to (107) show that the Helmholtz equation (103) can be represented as a set of NP1 Helmholtz equations for a model have NP1 vertical levels. However, these equations are coupled and need to be made linearly independent before iterative method is applied for their solution.

The set of Helmholtz equations can be represented by a single matrix equation

$$\underline{\underline{A}}\underline{P} + \nabla^2\underline{\underline{B}}\underline{P} = \underline{F} \qquad (108)$$

A and B are (NP1xNP1) matrices. B is a tridiagonal matrix. P is a column vector of length NP1. $\underline{F}$ is also a column vector of length NP1.

(108) can be written as

$$\underline{P} + \nabla^2 \underline{A}^{-1}\underline{B}\underline{P} = \underline{A}^{-1}\underline{F} \tag{109}$$

Let $\underline{A}^{-1}\underline{B} = \underline{C}$ (110)

Then (109) becomes

$$\underline{P} + \nabla^2 \underline{C}\underline{P} = \underline{A}^{-1}\underline{F} \tag{111}$$

Matrix $\underline{C}$ can be diagonalized using similarity transformation.

If two matrices T and $T^{-1}$ can be determined such that

$$T\underline{C}T^{-1} = \lambda \tag{112}$$

then

$$C = T^{-1}\lambda T \tag{113}$$

where $\lambda = \begin{vmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & & \\ & & & \lambda_{NP1} \end{vmatrix}$

is a diagonal matrix representing eigen values of matrix $\underline{C}$.

Substituting (113) in (111)

$$\underline{P} + \nabla^2 \underline{T}^{-1}\lambda \underline{T}P = \underline{A}^{-1}\underline{F} \tag{114}$$

T and $T^{-1}$ are lower triangular matrices.

Operating (114) by $\underline{T}$

$$\underline{T}\underline{P} + \nabla^2 \lambda \underline{T}\underline{P} = \underline{T}\underline{A}^{-1}\underline{F}$$

Let $T\underline{P} = \underline{R}$ and $TA^{-1} = E$ (115)

$$\underline{R} + \nabla^2 \lambda \underline{R} = \underline{E}\underline{F} \tag{116}$$

(116) is the matrix representation of a set of linearly independent Helmholtz equations. Solution of (115) by relaxation technique gives NP1 values of $\underline{R}$ and using (115). NP1 values of $\underline{P}$ can be obtained.

Let $\underline{V}$ be the eigen-vectors of matrix $\underline{C}$ corresponding to eigen values $\lambda$ i.e.

$$CV = V\lambda \tag{117}$$

or $$C = V\lambda V^{-1} \tag{118}$$

But $$C = T^{-1}\lambda T$$

therefore $T^{-1} = V$ = Eigen vector matrix

For finding the eigen values ($\lambda$) and eigen vectors ($T^{-1}$) of matrix C ($=A^{-1}B$), the elements of matrices A and B are required. These can be written from equations (104), (105), (106) and (107) as follows.

Matrix $A_{ij}$ ($i=1,NP_1$ ; $j=1,NP_1$)

$$A_{1,1} = -\frac{1}{\tilde{\gamma}_1^*}\frac{1}{\Delta\tilde{\sigma}_1}\frac{1}{\Delta\tilde{\sigma}_1} \tag{119-a}$$

$$A_{1,2} = \frac{1}{\tilde{\gamma}_1^*}\frac{1}{\Delta\tilde{\sigma}_1}\frac{1}{\Delta\tilde{\sigma}_1} \tag{119-b}$$

$$A_{1,NP1} = \frac{1}{\Delta\tilde{\sigma}_1}\frac{1}{RT^*_{NP1}} \tag{119-c}$$

$$A_{i,i-1} = \frac{1}{\tilde{\gamma}^*_{i-1}}\frac{1}{\Delta\tilde{\sigma}_{i-1}}\frac{1}{\Delta\sigma_i} \tag{119-d}$$

$$A_{i,i} = -\frac{1}{\tilde{\gamma}_i^*}\frac{1}{\Delta\tilde{\sigma}_i}\frac{1}{\Delta\sigma_i} - \frac{1}{\tilde{\gamma}^*_{i-1}}\frac{1}{\Delta\tilde{\sigma}_{i-1}}\frac{1}{\Delta\sigma_i} \tag{119-e}$$

$$A_{i,i+1} = \frac{1}{\tilde{\gamma}_i^*}\frac{1}{\Delta\tilde{\sigma}_i}\frac{1}{\Delta\sigma_i} \tag{119-f}$$

$$A_{NP1,N} = \frac{1}{\tilde{\gamma}_N^*}\frac{1}{\Delta\tilde{\sigma}_N}\frac{1}{\Delta\tilde{\sigma}_N} \tag{119-g}$$

$$A_{NP1,NP1} = -\frac{1}{\tilde{\gamma}_N^*}\frac{1}{\Delta\tilde{\sigma}_N}\frac{1}{\Delta\tilde{\sigma}_N} - \frac{1}{RT^*_{NP1}}\frac{1}{\Delta\tilde{\sigma}_N} \tag{119-h}$$

Matrix B

$$B(1,1) = 0.25\ (\Delta t/2)^2 \tag{120-a}$$

$$B(1,2) = 0.25\ (\Delta t/2)^2 \tag{120-b}$$

$$B_{i,i-1} = 0.25\ (\Delta t/2)^2 \tag{120-c}$$

$$B_{i,i} = 0.50\ (\Delta t/2)^2 \tag{120-d}$$

$$B_{i,i+1} = 0.25\ (\Delta t/2)^2 \tag{120-e}$$

$$B_{NP1,N} = 0.25\ (\Delta t/2)^2 \tag{120-f}$$

$$B_{NP1,NP1} = 0.25\ (\Delta t/2)^2 \tag{120-g}$$

At horizontal boundaries the matrix equation reduces to

$$\underline{P} = \underline{\underline{A}}^{-1}\underline{F} \tag{121}$$

i.e. For boundaries the values of $\underline{P}$ are obtained without solving the Helmholtz equation.

2.6 Interpolation Scheme

The semi-Lagrangian advection scheme requires computation of the distance traversed and the total forcings experienced by the parcel in time step $\Delta t$, for prediction. For this a fourth order accurate finite difference scheme based on Taylor series expansion is used in this model. This is also equivalent to a 25 point stencil Lagrangian interpolation.

For any function F, according to definition given in equation (28), $F^\circ$ represents the value of function at time t and at a point $(\zeta,\eta)$ from where the parcel arrives at a grid point (x,y) at time $t+\Delta t$ i.e.

$$F^\circ = F\ (x-\zeta\ y-\eta\ \sigma,t) \tag{122}$$

As shown in figure 3, the origin of the parcel is at point O and it arrives at the grid point P. In the present scheme it is not necessary for the parcel to be less than one grid interval away from the grid point at which it arrives in time step $\Delta t$. However, once the origin is determined nearest 25 point stencil is used for interpolation.

Interpolated value of any function F from grid point to the origin $\zeta,\eta$ is obtained in the following way.

$$F^\circ = F(x-\zeta,\ y-\eta,\ \sigma,\ t)$$

$$= (I-G_{ij})\ (I-H_{ij})\ F(x,y,\sigma,t) \tag{123}$$

Where $G_{ij}$ and $F_{ij}$ are the fourth order spatial finite difference operators. These can be written, from figure 3, as

$$G_{ij} = \zeta_{ij}\overset{x}{\delta}_{ij} - \frac{\zeta_{ij}^2}{2!}\overset{2x}{\delta}_{ij} + \frac{\zeta_{ij}^3}{3!}\overset{3x}{\delta}_{ij} - \frac{\zeta_{ij}^4}{4!}\overset{4x}{\delta}_{ij} \tag{124}$$

$$H_{ij} = \eta_{ij}\overset{y}{\delta}_{ij} - \frac{\eta_{ij}^2}{2!}\overset{2y}{\delta}_{ij} + \frac{\eta^3}{3!}\overset{3y}{\delta}_{ij} - \frac{\eta_{ij}^4}{4!}\overset{4y}{\delta}_{ij} \tag{125}$$

$\overset{x}{\delta}_{ij}$, $\overset{2x}{\delta}_{ij}$, $\overset{3x}{\delta}_{ij}$, $\overset{4x}{\delta}_{ij}$ are first, second, third, and fourth order differential operators in x directions and $\overset{y}{\delta}_{ij}$, $\overset{2y}{\delta}_{ij}$, $\overset{3y}{\delta}_{ij}$, and $\overset{4y}{\delta}_{ij}$ are corresponding quantities in y direction. They can be written as

$$\overset{x}{\delta}_{ij} = \frac{(\ \ )_{i+1,j} - (\ \ )_{i-1,j}}{2\delta x_{i,j}} \tag{126}$$

$$\overset{y}{\delta}_{ij} = \frac{(\ \ )_{i,j+1} - (\ \ )_{i,j-1}}{2\delta y_{i,j}} \tag{127}$$

$$\overset{2x}{\delta}_{ij} = \frac{(\ \ )_{i+1,j} - 2(\ \ )_{i,j} + (\ \ )_{i-1,j}}{(\delta x_{ij})^2} \tag{128}$$

$$\overset{2y}{\delta}_{ij} = \frac{(\ \ )_{i,j+1} - 2(\ \ )_{i,j} + (\ \ )_{i,j-1}}{(\delta y_{ij})^2} \tag{129}$$

Also

$$\overset{mx}{\delta}_{ij}\ (\overset{nx}{\delta}_{ij}) = \overset{nx}{\delta}_{ij}\ (\overset{mx}{\delta}_{ij}) \tag{130-a}$$

$$\overset{m}{\delta}_{ij}\ (x) = \overset{m_1x}{\delta}_{ij}\ \overset{m_2x}{\delta}_{ij}\ ; \qquad m_1 + m_2 = m \tag{130-b}$$

$$\overset{mx}{\delta}_{ij}\ (\overset{ny}{\delta}_{ij}) = \overset{ny}{\delta}_{ij}\ \overset{mx}{\delta}_{ij} \tag{130-c}$$

$$\overset{my}{\delta}_{ij}\ (\overset{ny}{\delta}_{ij}) = \overset{ny}{\delta}_{ij}\ (\overset{ny}{\delta}_{ij}) \tag{130-d}$$

$$\overset{my}{\delta}_{ij} = \overset{m_1y}{\delta}_{ij}\ \overset{m_2y}{\delta}_{ij}\ ; \qquad m_1 + m_2 = m \tag{130-e}$$

$\zeta$ and $\eta$ are the distances of origin from the grid point at which the parcel arrives. The first guess values of $\zeta$ and $\eta$ are computed using

$$X_{ij} = u_{ij}\Delta t + A_{ij}\frac{\Delta t^2}{2} \tag{131}$$

$$Y_{ij} = v_{ij}\Delta t + B_{ij}\frac{\Delta t^2}{2} \tag{132}$$

$X_{ij}$ and $Y_{ij}$ are the quantities corresponding to $\zeta$ and $\eta$ computed using the forcings, $A_{ij}$ and $B_{ij}$ at the grid point in X and Y direction respectively. The first guess $u^\circ$ and $v^\circ$ and forcings $A^\circ$ and $B^\circ$ at the departure points are then determined using

$$u^\circ = (I-G_{ij})(I-H_{ij})u_{ij} \tag{133a}$$

$$v^\circ = (I-G_{ij})(I-H_{ij})v_{ij} \tag{133b}$$

and

$$A^\circ = (I-G_{ij})(I-H_{ij})A_{ij} \tag{134a}$$

$$B^\circ = (I-G_{ij})(I-H_{ij})B_{ij} \tag{134b}$$

where

$$G_{ij} = X_{ij}\delta^{x}_{ij} - \frac{X^2_{ij}}{2!}\delta^{2x}_{ij} + \frac{X^3_{ij}}{3!}\delta^{3x}_{ij} - \frac{X^4_{ij}}{4!}\delta^{4x}_{ij} \tag{135}$$

$$H_{ij} = Y_{ij}\delta^{y}_{ij} - \frac{Y^2_{ij}}{2!}\delta^{2y}_{ij} + \frac{Y^3_{ij}}{3!}\delta^{3y}_{ij} - \frac{Y^4_{ij}}{4!}\delta^{4y}_{ij} \tag{136}$$

With these known quantities, the second guess distance of origin from the grid point are then determined using (131) and (132) but with values at the grid points replaced by the values at the departure points. Once these are known, a 25 point stencil is considered around this origin and any function is interpolated using these 25 points to this origin using (123).

This formulation is equivalent to a 4th order Lagrangian interpolation formula but it is more efficient than latter in the sense that cross terms of the Lagrangian interpolation are not required to be computed explicitly.

The scheme described above is equivalent to a Taylor series expansion, a Lagrangian interpolation formula or a 4th degree polynomial. The relationship is established in the following paragraphs.

If the value of a function is known at a point than its value at a point at a very small distance from it can be expressed as a Taylor series. Let us consider a two dimensional case in which

$\delta F = F(x,y,t) - F(x-\zeta,\ y-\eta,\ t-\Delta t)$

is the change in property F of a parcel as it arrives from point $(x-\zeta,\ y-\eta)$ to point x,y in time $\Delta t$. The semi-Lagrangian advection scheme seeks to determine the value of the variable at the point $(x-\zeta,\ y-\eta)$ through a fourth order interpolation formula, which can be expressed as a two-dimensional Taylor series

$$F_0 = F(x-\zeta,\ y-\eta,\ t-\Delta t)$$

$$= F(x,y,t-\Delta t) - \zeta\frac{\partial F}{\partial x} - \eta\frac{\partial F}{\partial y} + \frac{1}{2!}\ \zeta^2\ \frac{\partial^2 F}{\partial x^2} + \zeta\eta\frac{\partial^2 F}{\partial x\partial y} + \eta^2\frac{\partial^2 F}{\partial y^2}$$

$$- \frac{1}{3!}\ \zeta^3\frac{\partial^3 F}{\partial x^3} + 3\zeta^2\eta\frac{\partial^3 F}{\partial x^2\partial y} + 3\zeta\eta^2\frac{\partial^3 F}{\partial x\partial y^2} + \eta^3\frac{\partial^3 F}{\partial y^3}$$

$$+ \frac{1}{4!}\ \zeta^4\frac{\partial^4 F}{\partial x^4} + 4\zeta^3\eta\frac{\partial^4 F}{\partial x^3\partial y} + 6\zeta^2\eta^2\frac{\partial^4 F}{\partial x^2\partial y^2} + 4\zeta\eta^3\frac{\partial^4 F}{\partial x\partial y^3} + \eta^4\frac{\partial^4 F}{\partial y^4}$$

This can also be expressed as

$$= F_{i,j} - (\zeta\ \delta^{x}_{ij}\ F_{ij} - \eta\delta^{y}_{ij}) + 1/2!\ (\zeta^2\delta^{2x}_{ij} + 2\zeta\eta\delta^{y}_{ij}\ \delta^{x}_{ij} + \eta^2\delta^{2y}_{ij})F_{ij}$$

$$- \frac{1}{3!}\ (\zeta^3\delta^{2x}_{ij}\delta^{x}_{ij} + 3\zeta^2\eta\delta^{2x}_{ij}\delta^{y}_{ij} + 3\zeta\eta^2\delta^{x}_{ij}\delta^{2y}_{ij} + \eta^3\delta^{2y}_{ij}\delta^{y}_{ij})F_{ij}$$

$$+ \frac{1}{4!}(\zeta^4\delta^{2x}_{ij}\delta^{2x}_{ij} + 4\zeta^3\eta\delta^{2x}_{ij}\delta^{x}_{ij}\delta^{y}_{ij} + 6\zeta^2\eta^2\delta^{2x}_{ij}\delta^{2y}_{ij} + 4\zeta\eta^3\delta^{x}_{ij}\delta^{y}_{ij}\delta^{2y}_{ij} + \delta^{2y}_{ij}\delta^{2y}_{ij})F_{ij}$$

We also note that

$$\delta^{mx}_{i,j}\ (\delta^{ny}_{i,j})F_{ij} = \delta^{ny}_{i,j}\ (\delta^{mx}_{i,j})F_{ij}$$

where $\delta^{x}_{ij} = \dfrac{(F_{i+1,j} - F_{i-1,j})}{(2\Delta X_j)}$, $\delta^{2x}_{ij} = \dfrac{(F_{i+1,j} - 2F_{i,j} + F_{i-1,j})}{(\Delta x_j)^2}$ etc.

If we consider the split form i.e. we take the interpolation formula as made up of two, one in i direction and another in j direction, then we can get a compact looking formulae in which the explicit computation of the cross terms (like $\delta_{ij}^{2x}\,\delta_{ij}^{y}$) is not necessary.

Let x-direction interpolation formula

$$= (I - G_{ij})F_{ij}$$

$$= F_{ij} - \zeta\delta_{ij}^{x} F_{ij} + \frac{\zeta^2}{2!}\delta_{ij}^{2x} F_{ij} - \frac{\zeta^3}{3!}\delta_{ij}^{3x} F_{ij} + \frac{\zeta^4}{4!}\delta_{ij}^{4x} F_{ij}$$

Then

$$F(x-\zeta, y-\eta) = (I - G_{ij})(I - H_{ij})F$$

In this case first the interpolation is carried out in one direction and then the second operator operates upon the interpolated function. For easiness and clarity let us consider a second order formulation.

$$(I - G_{ij})(I - H_{ij})F$$

$$(I - H_{ij})F = (I - \eta\delta_{ij}^{y} + \frac{\eta^2}{2!}\delta_{ij}^{2y})F_{ij}$$

$$(I - G_{ij})(I - H_{ij})F = (I - \zeta\delta_{ij}^{x} + \frac{\zeta^2}{2!}\delta_{ij}^{2x})(I - \eta\delta_{ij}^{y} + \frac{\eta^2}{2!}\delta_{ij}^{2y})F$$

$$= \Big[ I - \zeta\delta_{ij}^{x} - \eta\delta_{ij}^{y} + \frac{\zeta^2}{2!}\delta_{ij}^{2x} + \zeta\eta\delta_{ij}^{x}\delta_{ij}^{y} + \frac{\eta^2}{2!}\delta_{ij}^{2y} - \frac{\zeta^2\eta}{2!}\delta_{ij}^{2x}\delta_{ij}^{y} - \frac{\zeta\eta^2}{2!}\delta_{ij}^{x}\delta_{ij}^{2y} + \frac{1}{2!2!}\zeta^2\eta^2\delta_{ij}^{2x}\delta_{ij}^{2y} \Big]$$

An equivalent representation can be obtained in form of a fourth order polynomial. Let us consider only one direction.

We can expresss the value at any point x as

$$P_n(x) = a_o + a_1(x-x_o) + a_2(x-x_o)^2 + a_3(x-x_o)^3 + a_4(x-x_o)^4$$

This polynomial must be exactly defined at 5 points to get 5 coeffs.

$$a_2 = \frac{1}{2}\,\frac{(y_1 - 2y_0 + y_{-1})}{h^2}$$

$$- \frac{1}{24h^2}\,[y_2 - 4y_1 + 6y_0 - 4y_{-1} + y_{-2}]$$

$$= \frac{1}{24h^2}\,[12y_1 - 24y_0 + 12y_{-1} - y_2 + 4y_1 - 6y_0 + 4y_{-1} - y_{-2}]$$

$$= \frac{1}{24h^2}\,[-y_2 + 16y_1 - 30y_0 + 16y_{-1} - y_{-2}]$$

$$y_1 - y_{-1} = 2a_1h + 2a_3h^3$$

$$y_2 - y_{-2} = 4a_1h + 16a_3h^3$$

$$a_1 = \frac{y_1 - y_{-1}}{2h} - a_3h^2$$

$$y_2 - y_{-2} = 4\left[\frac{y_1 - y_{-1}}{2h} - a_3h^2\right]h + 16a_3h^3$$

$$= 2y_1 - 2y_{-1}\ 4a_3h^3 + 16a_3h^3$$

$$\text{or}\quad 12a_3h^3 = y_2 - 2y_1 + 2y_{-1} - y_{-2}$$

$$\text{or}\quad a_3 = \frac{1}{12h^3}\,[y_2 - 2y_1 + 2y_{-1} - y_{-2}]$$

$$a_1 = \frac{y_1 - y_{-1}}{2h} - \frac{1}{12h^3}\,[y_2 - 2y_1 + 2y_{-1} - y_{-2}]h^2$$

$$= \frac{1}{12h}\,[6y_1 - 6y_{-1} - y_2 + 2y_1 - 2y_{-1} + y_{-2}]$$

$$= \frac{1}{12h}\,[-y_2 + 8y_1 - 8y_{-1} + y_{-2}]$$

If we want to use a fourth order accurate formula in 2-dimensions, then the polnomial can be represented as

$$P_n(x,y) = a_0 + a_1(x-x_0) + a_1(y-y_0) + a_2(x-x_0)^2 + a_2(y-y_0)^2$$
$$+ a_3(x-x_0)^3 + a_3(y-y_0)^3 + a_4(x-x_0)^4 + a_4(y-y_0)^4.$$

The calculation of interpolated value can be made more efficient if the computation is split into 2 steps. First the x-direction inter-

polation is done and then on the interpolated values the y direction interpolation is done. In this way the explicit computation of cross-terms can be avoided. This is what has been done in the present case.

## 2.7 Scheme of Integration

The integration of the limited area, grid point primitive equation model using the semi-Lagrangian semi-implicit scheme involves the following steps.

1. Forcing, $s_j$ at grid points are calculated and stored in $S_j$.
2. $\zeta$ and $\eta$ are calculated using an efficient fourth order interpolation scheme given in section 2.6.
3. Forcings $s_j$ at grid point are interpolated to origin of parcel using $\zeta,\eta$ by a similar interpolation scheme and stored in $s_j°$.
4. Total effective forcings are calculated using $s_j$, $s_j°$ and $s_j$ and stored in $W_j$.
5. Forcings for Helmholtz equation are calculated using $W_j$'s and stored in Q.
6. 3-dimensional Helmholtz equation is transformed into a set of 2 dimensional linearly independent Helmholtz equations and these are solved by standard iterative techniques to obtain model parameter P. At horizontal boundary points, due to boundary conditions, $\sigma=0$ and $\nabla\cdot V=0$, Helmholtz equations reduce to simple algebraic equations from which model parameter P can be determined directly.
7. From $P^+$ and $W^+$ prognostic and diagnostic variables are calculated as follows:

u-component of wind

$$u^+ = W_1^+ - \frac{\Delta t}{2}\frac{\partial P^+}{\partial x} \tag{137}$$

At horizontal boundary points, the derivative $\frac{\partial P}{\partial x}$ is obtained by upstream open boundary conditions. For example, at west boundary, as shown in figure 2, the derivative is written as

$$\left[\frac{\partial P}{\partial x}\right]_1 = \frac{P_2 - P_1}{\Delta x} \tag{138}$$

v-component of wind

$$v^+ = W_2^+ - \frac{\Delta t}{2}\frac{\partial P^+}{\partial y} \tag{139}$$

Surface pressure

$$\ln p_s^+ = \frac{1}{RT_s^*}(P_s^+ - gz_s) \tag{140}$$

Moisture Variable

$$q^+ = W_4^+ \tag{141}$$

Geopotential Height

$$Z^+ = \frac{1}{g}(P^+ - RT^* \ln p_s) \tag{142}$$

Temperature

$$T^+ = -\frac{\sigma g}{R}\frac{\partial Z^+}{\partial \sigma} \tag{143}$$

Vertical velocity measure ($\sigma$)

$$\sigma^+ = \frac{2}{\Delta t}\left[ W_3^+ - \frac{1}{\gamma^*}\frac{\partial P^+}{\partial \sigma} - \sigma \ln p_s \right] \tag{144}$$

Steps 1 to 7 are repeated to obtain forecast at desired time.

3. Planetary Boundary Layer (PBL)

Within the PBL and the surface layer one has to consider the fluxes of momentum, sensible heat and moisture due to turbulent eddy motion. A separation between the land and the oceanic regions is necessary. We shall present the so-called similarity analysis approach for the specification of the surface fluxes. It is worth noting that the radiation code in a numerical weather prediction model is usually completed with a calculation of the surface energy balance. This surface energy balance over land areas invokes a balance between the net radiation and the surface fluxes of heat and moisture for the determination of soil temperature. Thus, it is quite apparent that the parameterization of the planetary boundary layer has to be compatible with the parameterization of the radiative processes.

a) Bulk aerodynamic calculation over ocean. Surface fluxes of sensible heat, water vapor and momentum can be expressed by bulk formulae: These are expressed by,

$$F_\theta = C_p \rho C_H \, | V_a | \, (T_w - T_a) \qquad \text{(sensible heat)}$$

$$F_q = L \rho C_Q \, | V_a | \, (q_w - q_a) \qquad \text{(latent heat)} \qquad (1)$$

$$F_m = \rho C_D \, | V_a | \; V_a \qquad \text{(momentum)}$$

(A list of symbols is provided in Table 2)

Here $V_a$, $T_a$ denote the wind speed and the air temperature at the anemometer level, and $T_w$, $q_w$ denote the water temperature and the saturation specific humidity. The values of the non-dimensional exchange coefficients that are widely used are:

$$C_H = 1.4 \times 10^{-3}$$

$$C_Q = 1.6 \times 10^{-3} \qquad (2)$$

PAGE 36 MISSING

$$C_D = 1.1 \times 10^{-3}$$

These were determined during the GARP Atlantic Tropical Experiment.

The choice of units is quite important for the calculation of these fluxes. For most meteorological purposes it is desirable to express $F_\theta$ and $F_q$ in units of watts/m$^2$, while $F_m$ is usually expressed by the familiar units of Pascal. For these units one can simplify the bulk formulae (using $\rho$ = 1.23 Kg/m$^3$) to read

$$F_\theta = 1.72 \, | Va | \, (T_w - T_a)$$

$$F_q = 4.92 \times 10^3 \, | Va | \, (q_w - q_a) \qquad (3)$$

$$F_m = 1.35 \times 10^{-2} \, V_a^2$$

Here the wind speed is measured in ms$^{-1}$, the temperature in °K or °C and the specific humidity q is measured in units of gm/kg.

In situations of strong wind speeds the waves over the ocean exert a large drag on the air and one finds it desirable to allow for a variation of the drag coefficient as a function of the wind speed, especially if the wind speeds exceed 10ms$^{-1}$. The variation of surface drag coefficient as a function of the wind speed is incorporated via the variation of a roughness parameter $Z_o$ described below.

b. <u>The roughness parameter Zo</u>

Zo varies in space and is also regarded as a function of time.

Given a domain of calculation $0 < x < L$, $0 < y < M$ one needs a tabulation of a land-ocean Matrix LxM and a tabulation of orographic height h.

The tabulation of h can be obtained for instance from Gates and Nelson (1975).

i) <u>Oceans</u>

The well-known Charnock (1955) formula is an accepted method for defining Zo over oceans:

$$Zo = M \frac{U_*^2}{g} \tag{4}$$

where M is a constant and has a value around 0.04. One cannot calculate Zo from this formula until U* the friction velocity is known, thus an iteration procedure is necessary.

Note that:

$$U_*^2 = (\overline{-u'w'}) = \tau_o/\rho_o \tag{5}$$

where the overbar is an area average, $\tau_o$ is the surface stress and $\rho_o$ is the density of air. A first guess field of U* can be obtained from the Bulk aerodynamic representation provided that one allows for the variation of the drag as a function of the wind speed. Some accepted values of the drag coefficients are:

$$\begin{aligned} C_D = C_{DO} &= 1.1 \times 10^{-3} \text{ for } V < 5.8 ms^{-1} \\ &= C_{DO}\ (0.74 + 0.046V) \text{ for } 5.8 \leqslant V \leqslant 16.8 ms^{-1} \\ &= C_{DO}\ (0.94 + 0.034V) \text{ for } V > 16.8\ ms^{-1}. \end{aligned} \tag{6}$$

Thus the first guess field of Zo is obtained from the observed wind speed V via the following steps.

$$V \rightarrow C_D \rightarrow (=C_D\ V^2) \rightarrow U_* \rightarrow Zo$$

Using this first guess Zo one may next follow the similarity approach (described below) to determine the surface fluxes including U* which in turn defines the final value of Zo the roughness parameter.

ii) <u>Land</u>

Over land a method defined by Delsol et al. (1971) allows for a

variation of the roughness parameter as a function of the elevation based on the mesoscale variance of mountain heights. In its simplest form it is expressed by the relation,

$$Z_o = 0.15 + 0.2E - 8(236.8 + 18.42*h)**2 \text{ meters} \tag{7}$$

parameter Zo are m. For numerical weather prediction it is desirable to restrict an upper limit of Zo to about 100cm.

c. <u>Surface fluxes from similarity analysis, and planetary boundary layer</u>

The starting point of this analysis is an application of the similarity principle to the nondimensional wind shears $\phi_m$ and the thermal gradients $\phi_h$, i.e.,

$$\frac{kz}{U*}\frac{\partial \bar{u}}{\partial z} = \phi_m \tag{8}$$

and

$$\frac{kz}{\theta*}\frac{\partial \bar{\theta}}{\partial z} = \phi_h \tag{9}$$

A similar equation also may be written down for the vertical variation of the nondimensional humidity variable. Here k is the Von Karman's constant, U* defines the friction velocity which is related to the surface stress and the momentum flux.

The quantity $-U*\theta* = \overline{W'\theta'})_o$ is a measure of the surface heat flux. The functional form on the right hand side of the Eqs. (8) and (9) are usually determined by an empirical best fit curve from observations of the nondimensional wind shears and potential temperature gradient to the nondimensional length scale Z/L where L is the Monin Obukhov length defined by,

$$L = U_*^2/K\beta\theta_* \tag{10}$$

Here $U_*$ is the friction velocity and $\theta_*$ is a characteristic temperature scale at the surface and $\beta$ is the buoyancy parameter, $\beta = g/\theta_0$. Here a distinction between stable and unstable surface layers can be made, i.e.

$$L > 0 \quad \text{stable}$$

$$L < 0 \quad \text{unstable} \tag{11}$$

Thus, following Businger et al. (1971) we may write,

a) For the unstable case ($Z/L < 0$),

$$Kz/U_* \frac{\partial \bar{U}}{\partial z} = (1 - 15\ z/L)^{-1/4} \tag{12}$$

$$Kz/\theta_* \frac{\partial \bar{\theta}}{\partial z} = 0.74\ (1 - 9\ z/L)^{-1/2} \tag{13}$$

and (b) For the stable case ($Z/L > 0$)

$$\frac{KZ}{U_*} \frac{\partial \bar{U}}{\partial z} = 1.0 + 4.7\ z/L \tag{14}$$

$$KZ/\theta_* \frac{\partial \bar{\theta}}{\partial z} = 0.74 + 4.7\ z/L \tag{15}$$

The parameterization of the surface fluxes thus requires a determination of $U_*$, $\theta_*$, and $q_*$ (the humidity counterpart).

Next we shall define two measurement levels $Z_1$ and $Z_2$, assumed to lie within the "constant flux layer" adjacent to the earth's surface. If we denote $\Delta Z = Z_2 - Z_1$, $\bar{U}_2$, $\bar{U}_1$, $\bar{\theta}_1$, $\bar{\theta}_2$ as mean quantities of levels 2 and 1 respectively, then a finite form of the nondimensional relation from the stable and neutral case may be expressed by:

$$K\ \frac{(\bar{U}_2 - \bar{U}_1)}{U_*} = \ln\left[\frac{z_2}{z_1}\right] + 4.7\left[\frac{\Delta z}{L}\right] \tag{16}$$

and

$$K\ \frac{(\bar{\theta}_2 - \bar{\theta}_2)}{\theta_*} = 0.74\ \ln\left[\frac{z_2}{z_1}\right] + 4.7\left[\frac{\Delta z}{L}\right] \tag{17}$$

At this stage we have three unknowns in this problem, namely $U*$, $\theta*$ and $\Delta Z/L$. From Eq. 10 we can express $\Delta Z/L$ by,

$$\left[\frac{\Delta z}{L}\right] = \frac{\Delta z k \beta \theta *}{U_*^2} \tag{18}$$

The above three equations describe a closed system for the three unknowns, and, for this stable case, upon elimination of $\theta*$ and $U*$ a single equation for $\Delta Z/L$ is frequently obtained, (Chang, 1978).

$$\frac{\Delta z}{L} = \ln(z_2/z_1) \quad \frac{9.4R_{iB} - 0.74 + \sqrt{4.888\ R_{iB} + 0.5476}}{9.4 - 44.18\ R_{iB}} \tag{19}$$

Here $R_{iB}$ denotes the Bulk Richardson number, which also distinguishes the sign of the stability (or instability) of the surface layer. It is easy to see that a critical value of the Bulk Richardson number for the stable case lies between 0 and 0.212, the latter defines a critical Bulk Richardson number. Thus, the solution $\Delta Z/L$ from equation (19) may be substituted into equation (16) and (17) to determine $U*$ and $\theta*$. The fluxes for the stability dependent equivalent exchange coefficients for the stable case are expressed by the relations,

$$F_m = \rho C_D\ (\bar{U}_2 - \bar{U}_1)^2 \equiv U_*^2 \tag{20}$$

$$F_\theta = \rho C_p C_H\ (\bar{U}_2 - \bar{U}_1)(\bar{\theta}_2 - \bar{\theta}_1) \equiv -\ U*\theta* \tag{21}$$

$$F_q = \rho L_q C_q (\bar{U}_2 - \bar{U}_1)(\bar{q}_2 - \bar{q}_1) \equiv -U*q* \tag{22}$$

where the exchange coefficients for the stable and neutral case are

$$C_D = \frac{K^2}{[\ln\ (z_2/z_1) + 4.7\ \Delta z/L]^2} \tag{23}$$

$$C_H = C_Q = \frac{-K^2}{[\ln\ (z_2/z_1) + 4.7\ (\Delta z/L)][0.74\ln\ (z_2/z_1) + 4.7\ (\Delta z/L)]} \tag{24}$$

The analysis for the unstable cases is not as straight forward because of the fractional powers (equations 12 and 13) in the empirical fits of the nondimensional shear and thermal gradient data sets. As a result, a simple algebraic equation for $\Delta Z/L$ (similar to above equation) does not exist. A number of different approaches have appeared in recent literature suggesting iterative solutions for the three unknowns $\Delta Z/L$, $U*$ and $\theta*$ for the unstable case; here we follow the analysis based on Chang (1978). For a two-level representation $Z_1$, $Z_2$, as before, an equation for $Z_2/L$ is obtained

$$\frac{Z_2}{L} = R_{iB} \frac{Z_2}{\Delta z} \frac{[\ln(z_2/z_1) - \psi_1(z_2/L,\ z_2/z_1)]^2}{0.74\ [\ln\ (z_2/z_1) - \psi_2\ (z_2/L),(z_2/z_1)]} \tag{25}$$

The unknown in the above equation is the stability parameter $Z_2/L$. The functions $\psi_1$ and $\psi_2$ have rather complicated forms:

$$\psi_1 = \int_{z_1}^{z_2} \frac{1}{z} \left[1 - (1 - 15\ z/L)^{-1/4}\right] dz \tag{26}$$

and

$$\psi_2 = \int_{z_1}^{z_2} \frac{1}{z} \left[1 - (1 - 9z/L)^{-1/2}\right] dz \tag{27}$$

By introducing new variables, Chang obtained solutions for these integrals,

$$\psi_1 = \ln \left[ \frac{1+E_2^2}{1+E_1} \frac{1+E_2^2}{1+E_1^2} \right] - 2\tan^{-1}E_2 + 2\tan^{-1}E_1 \tag{28}$$

$$\psi_2 = 2\ln \left[ \frac{1+\Upsilon_2}{1+\Upsilon_1} \right] \tag{29}$$

where

$$E_1 = \left[ 1 - 15 \left(\frac{z_2}{L}\right) \Big/ \left(\frac{z_2}{z_1}\right) \right]^{1/4}$$

$$E_2 = (1 - 15 \quad z_2/L)^{1/4}$$

$$\Upsilon_1 = \left[ 1 - 9 \left(\frac{z_2}{L}\right) \Big/ \left(\frac{z_2}{z_1}\right) \right]^{1/2} \tag{30}$$

$$\Upsilon_2 = (1 - 9 \; z_2/L)^{1/2}$$

Equation 25 is a transcendental equation for $z_2/L$,

$$\frac{z_2}{L} = R_{iB} \; \frac{z_2}{\Delta z} \; \frac{[\ln (z_2/z_1) - \psi_1]^2}{0.74 \; [\ln (z_2/z_1) - \psi_2]} \tag{31}$$

By assuming a large range of values of $z_2/z_1$ and $z_2/L$, one can via backward substitution obtain a corresponding large range of values of $R_{iB}$. Chang used such a data set to provide a cubic least square regression for $\ln(-R_{iB} \; z_2/\Delta z)$ and $\ln(-z_2/L)$. This turned out to be a quasilinear relationship, since the value of the quadratic and cubic coefficients were quite small. The regressed relation was suggested as a means for a solution of the transcendental relation in practical applications. Chang showed that for observed range of values of $R_{iB}$ this relationship has an accuracy of around 98%. We have adopted this relation in our analysis.

It is now possible to write down the exchange coefficients for the unstable case, these would be counterparts of the Eqs. 23, 24 for the stable or neutral case. Thus, for the unstable case: ($R_{iB} < 0$)

$$C_D = \frac{K^2}{[\ln (z_2/z_1) - \psi_1]^2} \tag{32}$$

$$C_H = C_q = \frac{- K^2}{0.74[\ln (z_2/z_1) - \psi_2][\ln (z_2/z_1) - \psi_1]} \tag{33}$$

Here the exchange coefficients for heat and moisture fluxes are assumed to be equal. $\psi_1$ and $\psi_2$ are functions of $Z_2/L$; $Z_2/Z_1$ is a

known quantity.

In the actual calculation $Z_1$ is taken to be equal to Zo , $Z_2$ is the height of the lowest computational level. At the earth's surface $U_1$ and $V_1$ are both set to zero. $U_2$ and $V_2$ are the surface winds, i.e. at the lowest computational level. $T_1$ denotes the ground temperature over land or the ocean temperature. $T_2$ is the temperature of air at the lowest computational level at the surface. $Q_1$ is the saturation specific humidity at the ocean or land surface temperature. It is multiplied by a ground wetness parameter GW whose value is set to 1.0 over ocean, whereas over land its value is determined from empirical relations. $Q_2$ is the specific humidity of air at the lowest computational level next to the ground. The lowest computational level is set to 10 meters from the ground. $T_2$ and $Q_2$ are usually extrapolated downwards from a first computational level which is well above the 10 meter level.

d. <u>Coupling of Surface Fluxes From the Surface Energy Balance and the Similarity Theory</u>:

The surface energy balance equation can be solved for the ground temperature $T_g$ without any reference to the surface fluxes from the similarity theory. If that were done than we would have two separate solutions for the surface fluxes of sensible and latent heat i.e. one from the surface energy balance and the other from the similarity theory. In order to avoid this inconsistency an iterative solution of the two problems is proposed that couples the two solutions.

The surface energy balance equation may be expressed by,

$$F_L - \sigma T_g^4 + (1-\alpha)F_s + \rho C_p\, u_*\theta_* + \rho L\, u_* q_* = 0 \qquad (34)$$

Here $F_L$ denotes the downward flux of long wave radiation,

$F_S$ denotes the downward flux of short wave radiation,

$\alpha$ denotes the surface albedo

$u*$ denotes the friction velocity at the ground

$\theta*$ and $q*$ respectively denote the characteristic temperature and specific humidity of the surface layer. Furthermore it should be noted that $u*\theta*$ denotes the flux of sensible heat from the earth's surface to the atmosphere, and $u*q*$ denotes the flux of latent heat. In section (3.1c) we have discussed a procedure for the solutions of $u*$, $\theta*$ and $q*$ for stable and unstable conditions. Stability is determined from the sign of the Monin Obukhov length L which is defined by,

$$L = \frac{u_*^2}{K\beta\theta*}$$

where $u*\theta* = -\overline{\omega'\theta'})_0$ the surface heat flux, which is positive if upward for unstable situations i.e. $L < 0$. Given a first guess (superscript 1) value of the surface temperature $T_g^{(1)}$ one can evaluate the corresponding surface moisture $q_g^{(1)}$ using the relation for saturation specific humidity, ground wetness and surface albedo. Stability is next calculated from the magnitude of the Bulk Richardson's number,

$$R_{iB}(1) = \frac{\beta[T_a - T_g]}{[u_a - u_g]^2} \Delta Z$$

Here $\beta$ is a constant $= g/\overline{\theta}$ ; $T_a$ is an interpolated air temperature at the top of the Prandtl layer ($\Delta Z$ = 30 meters) where the wind speed is $u_a$. The ground temperature is $T_g$ and the wind speed at the surface $u_g$

$= 0$. One next solves for $(\Delta Z/L)^{(1)}$ for the stable ($R_{iB} > 0$) and unstable ($R_{iB} < 0$) situations from Eqs. (19) and (25) respectively.

The stability exchange coefficients $C_D^{(1)}$, $C_H^{(1)}$ and $C_q^{(1)}$ are next evaluated utilizing the first guess value of the Monin Obukhov length $L^{(1)}$. It is now possible to update the values of the surface fluxes $-u_*$, $\theta_*$ and $-u*q*$ as a function of the Monin Obukhov length and stability, here we use the Eqs. (20), (21) and (22). At this point $u*$, $\theta*$ and $q*$ are substituted into the energy balance equation and one solves for an updated ground temperature $T_g^{(2)}$. This cycle is repeated to minimize the difference $|(T_g^{\nu+1} - T_g^{\nu})|$ as a function of the scan $\nu$. This procedure converges very rapidly yielding a coupling among the surface energy balance and the surface similarity theory.

e. <u>The Planetary Boundary Layer and the Vertical Distribution of Surface Fluxes</u>:

Within the planetary boundary we are guided by the GATE observation for the parameterization of the vertical distribution of fluxes. The humidity flux, according to GATE observations, increases with height during disturbed conditions. This increase of vertical eddy flux of humidity is generally attributed to downdrafts from deep convection that tend to dry the planetary boundary layer. The deep convective parameterization of the present study, a modified Kuo's scheme, lacks an explicit treatment of this phenomenon.

During undisturbed conditions the eddy flux of moisture in the planetary boundary layer is known (GATE observation) to decrease with height.

From an inspection of these data sets the following empirical

expression for the vertical distribution of humidity flux was proposed for disturbed conditions.

$$F_q(P) = F_q(P_o) \left[ 1 - \frac{1}{\Delta p} \int_P^{P_o} \overline{\left[ \frac{Q_2 - Q_2^K}{\overline{Q_2} - \overline{Q_2^K}} \right]} dp \right] \tag{35}$$

Here $\Delta p$ denotes a depth of the troposphere, $F_q(P_o)$ is the surface flux as determined from similarity theory. In the second term of the above expression, $Q_2$ denotes the total apparent moisture sink while $Q_2^K$ denotes that from the deep convection alone - as determined from the Kuo's theory,

$$Q_2^K = -L \left[ a_q \frac{q_s - q}{\Delta \tau} + \omega \frac{\partial q}{\partial p} - \eta \frac{I_L}{(PB - PT)} \right] \tag{36}$$

The overbars in the denominator of Eq. 35 denote a vertical average through the depth of the troposphere, and $\overline{[\ ]}$ is the time average. This structure function

$$\left[ 1 - \frac{1}{\Delta p} \int_P^{P_o} \overline{\left[ \frac{(Q_2 - Q_2^K)}{(\overline{Q_2} - \overline{Q_2^K})} \right]} dp \right]$$

is determined entirely from GATE observations. It reflects the vertical structure for the eddy flux of moisture, for all processes not including those covered by the simple Kuo's parameterization. Here we define 'disturbed situations' as those where the precipitation rates exceed 5 mm/day. Figure 3 illustrates the GATE structure function for the eddy flux of moisture for disturbed conditions. This allows for a drying of the subcloud layer and a moistening of the troposphere. Here we assume that this structure function provides a parameterization of the downdrafts from deep convective clouds as well as the moistening

by shallow convective clouds that coexist with the deep clouds. This precipitation rate is here determined from an application of the Kuo's scheme, discussed in the next section, and is given by the relation,

$$R_T = I_L(1+\eta)(1-b).$$

Equation (35) satisfies the following constraints,

Surface flux = $F_q(P_o)$

flux at the top ($P = p_T$) is zero.

Total flux convergence = $F_q(P_o)$

If $R_T \leq 5$ mm/day then we assume that an undisturbed boundary layer is present, in which case the vertical distribution of eddy fluxes of humidity within the PBL, based on GATE, may be approximated by,

$$F_q(P) = F_q(P_o) \left[\frac{P - 1.03\ P_{LCL}}{P_o - 1.03\ P_{LCL}}\right] \tag{37}$$

Where $P_o$ denotes the surface pressure, P the pressure at a reference level and $P_{LCL}$ the pressure at the lifting condensation level. In these situations the GATE measurements were only made in the undisturbed subcloud layer. The flux vanishes at a level close to the lifting condensation level. Usually shallow convection is present above the LCL, these clouds extend to the base of an inversion. The treatment of shallow convection is given in the next section.

The vertical fluxes of sensible heat and momentum for disturbed and undisturbed conditions are based on GATE observations. These are given by the following relations, undisturbed situations ($R_T \leq 5$ mm/day):

An equation similar to (35) is used for the distribution of sensible heat for disturbed conditions. For undisturbed conditions we set

$$F_\theta(P) = F_\theta(P_o) \left[\frac{P - P_{LCL}}{P_o - P_{LCL}}\right] \tag{38}$$

The surface flux from the similarity theory is given by $F_\theta(P_o)$ and it decreases to zero at the lifting condensation level.

The best fit relation, for the wind stress for undistured conditions, based on GATE observations is very similar to the other relation,

$$F_m(P) = F_m(P_o) \left[\frac{P - 1.08P_{LCL}}{P_o - 1.08P_{LCL}}\right] \tag{39}$$

The stress vanishes close to the lifting condensation level.

For disturbed situations the flux of momentum is expressed by

$$F_m(P) = F_m(P_o) \{ap + b\} \tag{40}$$

$$a = \frac{0.54}{P_o - P_{LCL}} \tag{41}$$

and $$b = \frac{0.46P - P_{LCL}}{P_o - P_{LCL}} \tag{42}$$

The flux in the disturbed situations vanishes close to the lifting condensation level.

4. Parameterization of Shallow Convection:

Following Tiedke (1984) we assume a simple diffusive form for the parameterization of shallow convection. Shallow convection is invoked over conditionally unstable undisturbed situation ($R_T$ < 5 mm/day) via a K theory. The humidity and the thermal equations are expressed by,

$$\frac{\partial \bar{q}}{\partial t} = \frac{\partial}{\partial p} K_q \frac{\partial \bar{q}}{\partial p}, \tag{1}$$

and

$$\frac{\partial \bar{\theta}}{\partial t} = \frac{\partial}{\partial p} K_\theta \frac{\partial \bar{\theta}}{\partial p} \tag{2}$$

Furthermore, following Tiedke (1984), we set, $K_q = K_\theta = Kg^2\rho^2$, the coefficient varies with height as the square of the density. Based on numerous studies with the GATE, ATEX and AMTEX data sets Tiedke suggests a value of K = 25 $m^2\ s^{-1}$. However setting a constant value of K does not provide a smooth continuity with the fluxes at the lifting condensation level. It is desirable that $Kg^2\rho^2 \frac{\partial \bar{q}}{\partial p}$ at the lifting condensation level be consistent with the planetary boundary layer flux at that level. If we define

$$K_q = \frac{F_q(LCL)/L}{\rho^2 g \left.\frac{\partial \bar{q}}{\partial p}\right)_{LCL}} \tag{3}$$

then the humidity flux in the subcloud layer and the shallow convective cloud layer are smoothly connected. A similar definition of $K_\theta$ is also used for the same reasons. Above the lifting condensation level in the cloud layer $K_\theta \frac{\partial \bar{\theta}}{\partial p}$ is negative, implying a downward eddy flux of sensible heat from the base of inversion to the cloud base. On the other hand $K_q \frac{\partial \bar{q}}{\partial p}$ is essentially positive in the shallow convective cloud layer.

Since the humidity flux is small both at the cloud base and at the cloud top, over the lower half of the clouds $\frac{\partial}{\partial p} K_q \frac{\partial \bar{q}}{\partial p}$ is negative and is positive over the upper half of the clouds. Thus these shallow convective clouds tend to moisten the layers of the atmosphere over the upper half of the clouds. Heckley (1984) has summarized the results of recent numerical predictions made with the ECMWF model. The experience at ECMWF shows that the inclusion of shallow convection improves the overall tropical forecasts slightly. In its absence the planetary boundary layer tends to be too moist and the region below the inversion layer tends to be too dry. Although the present formulation is very simple it is shown to provide an overall improvement of the thermodynamical structure over the undisturbed areas. In the present formulation we have included the vertical eddy flux of momentum in the planetary boundary layer, however, it is not included within the cloud layer.

It is easy to see that moisture is conserved by this parameterization of shallow convection. The total convergence of flux of humidity in the cloud layer is equal to Fq(LCL), i.e. the flux at the cloud base.

It is important to assure that these shallow clouds are nonprecipitating and do not systematically produce saturation on the large scale. To assure that it is desirable to set an upper limit on the final specific humidity $q_F$ produced by the process,

i.e. $q_F/q_s \leqslant 0.8$

This is easy to implement formally although it violates the conser-

vation slightly. However this is necessary in order to avoid the eventual formation of layer clouds and associated large scale condensation.

The present formulation of shallow convection is very simplistic, and deserves to be investigated in further detail.

5. Parameterization of deep moist convection:

The large scale convergence of flux of moisture is expressed by the relation,

$$C_M = -\nabla\cdot\overline{Vq} - \frac{\partial\,\overline{\omega q}}{\partial p} \tag{1}$$

or in advective form by,

$$C_M = -\bar{V}\cdot\nabla\bar{q} - \bar{\omega}\frac{\partial\bar{q}}{\partial p} \tag{2}$$

In the Kuo type cumulus parameterization theories the supply of moisture is usually defined from a vertical integral of the above expression. The supply is then used to define the moist adiabatic cloud elements in various versions of the Kuo's scheme. Krishnamurti et al (1983a) and Anthes (1984) have noted that the supply of moisture for the definition of clouds may be expressed simply by the second term of the above equation. The first term, namely the horizontal advection is used for a direct moistening of air on the large scale. Thus we define the supply of moisture by the relation,

$$I_L = -\frac{1}{g}\int_{P_T}^{P_B} \omega\frac{\partial q}{\partial p}dp \tag{3}$$

Here $P_T$ and $P_B$ denote the cloud top and the cloud based respectively, they are defined in terms of the vanishing buoyancy level and the lifting condensation level respectively. The other symbols are the usual ones and are given in Table 2.

Based on our experience, Krishnamurti et al (1980, 1983a), we have noted that the above definition for the large scale supply is a

close measure of the rainfall rate, and thus sufficient supply is not available to account for the observed moistening of the vertical columns. These statements are based on semiprognostic studies made with the GATE observations. These lead us to propose a mesoscale convergence parameter $\eta$ and a moistening parameter b.

We define the total supply by the relation,

$$I = I_L(1+\eta) \tag{4}$$

where $I_L\eta$ denotes the net mesoscale moisture supply.

The total moisture supply is partitioned into the precipitating part and the moistening part via the following relations,

$$R = I(1-b) = I_L(1+\eta)(1-b) \tag{5}$$

$$M = Ib = I_L(1+\eta)b \tag{6}$$

Following Krishnamurti et al (1983a) we shall define the total supply of moisture required to produce grid scale moist - adiabatic sounding by the expression,

$$Q = \frac{1}{g}\int_{P_T}^{P_B} \frac{(q_s-q)}{\Delta\tau}dp + \frac{1}{g}\int_{P_T}^{P_B}\left[\frac{C_pT(\theta_s-\theta)}{L\theta\Delta\tau} + \omega\frac{C_pT}{L\theta}\frac{\partial\theta}{\partial p}\right]dp \tag{7}$$

$\Delta\tau$ denotes a cloud time scale ≈ 30 minutes. The two respective terms are denoted by,

$$Q = Q_q + Q_\theta \tag{8}$$

The total supply I may likewise be split into the moistening and heating parts by the relation,

$$I_q = Ib = I_Lb(1+\eta) \tag{9}$$

and $I_\theta = I(1-b) = I_L(1-b)(1+\eta)$ (10)

The thermal and the humidity equations are expressed by:

$$\frac{\partial\theta}{\partial t} + V\cdot\nabla\theta + \omega\frac{\partial\theta}{\partial p} = a_\theta\left(\frac{\theta_s-\theta}{\Delta\tau} + \omega\frac{\partial\theta}{\partial p}\right) \tag{11}$$

$$\frac{\partial q}{\partial t} + V\cdot\nabla q = a_q\left(\frac{q_s-q}{\Delta\tau}\right) \tag{12}$$

Where $a_\theta$ and $a_q$ are defined by the relations,

$$a_\theta = \frac{I_\theta}{Q_\theta} = \frac{I(1-b)}{Q_\theta} = \frac{I_L(1+\eta)(1-b)}{Q_\theta} \tag{13}$$

$$a_q = \frac{I_q}{Q_q} = \frac{Ib}{Q_q} = \frac{I_L b(1+\eta)}{Q_q} \tag{14}$$

The parameterization is closed if

b and $\eta$ are somehow determined then $a_\theta$ and $a_q$ may be evaluated from the relations,

$$a_\theta = \frac{I_\theta}{Q_\theta} = \frac{I(1-b)}{Q_\theta} = \frac{I_L(1-b)(1+\eta)}{Q_\theta} \tag{15}$$

and

$$a_q = \frac{I_q}{Q_q} = \frac{Ib}{Q_q} = \frac{I_L b(1+\eta)}{Q_q} \tag{16}$$

Krishnamurti et al (1983a) proposed a closure for b and $\eta$ based on a screening multiregression analysis of GATE observations. Here we regressed normalized heating and moistening $R/I_L$ and $M/I_L$ against a number of large scale variables.

Based on GATE observations we noted significant correlations for the heating and moistening from the following relations.

$$\frac{M}{I_L} = a_1\zeta + b_1\bar{\omega} + c_1 \tag{17}$$

$$\frac{R}{I_L} = a_2\zeta + b_2\bar{\omega} + C_2 \tag{18}$$

Where $a_1$, $b_1$, $c_1$, $a_2$, $b_2$, and $c_2$ are regression constants whose magnitudes may be found in Krishnamurti et al (1983a). $\zeta$ and $\bar{\omega}$ are respectively the relative vorticity and the vertically integrated vertical velocity.

Thus, in numerical weather prediction, $\zeta$ and $\omega$ determine $M/I_L$ and $R/I_L$;

since $$\frac{M}{I_L} = b(1+\eta) \tag{19}$$

and $$\frac{R}{I_L} = (1+\eta)(1-b) \tag{20}$$

These two relations determine b and $\eta$.

we can find $a_\theta$ and $a_q$ from the relations,

$$a_\theta = \frac{I_L(1-b)(1+\eta)}{Q_\theta} = \frac{R}{Q_\theta} \tag{21}$$

and

$$a_q = \frac{I_L\ b(1+\eta)}{Q_q} = \frac{M}{Q_q} \tag{22}$$

It is also of interest to note that the apparent heat source $Q_1$ and the apparent moisture sink $Q_2$, for this formulation, may be expressed by:

$$Q_1 = a_\theta \left[C_p \frac{T}{\theta} \frac{(\theta_s-\theta)}{\Delta\tau} + \omega C_p \frac{T}{\theta} \frac{\partial\theta}{\partial p}\right] - C_p \frac{T}{\theta} \left[H_R + H_s\right] \tag{23}$$

$$Q_2 = -L \left[a_q \frac{(q_s-q)}{\Delta\tau} + \omega\frac{\partial q}{\partial p}\right] \tag{24}$$

The total convection precipitation is given by,

$$P_T = \frac{1}{g} \int_{P_T}^{P_B} a_\theta\ C_p \frac{T}{\theta} \frac{(\theta_s-\theta)}{\Delta\tau} dp \tag{25}$$

## 6. Large Scale Condensation

Large scale condensation is usually invoked in a numerical weather prediction experiment if dynamic ascent of stable saturated air occurs at any level of the atmosphere.

The ascent is usually a consequence of large scale dynamics such as differential vorticity advection, thermal advection, slow orographic ascent or even the buoyancy driven ascent from the lower troposphere. In the latter case convective and nonconvective clouds could coexist over the same region.

The Stable air refers to absolute stability where both the potential and the equivalent temperatures increase with height. The Saturation refers to the ratio of specific humidity to the saturation specific humidity which is close to unity, although one does use a saturation ratio of the order of 0.8 to take into account possible subgrid scale saturation.

We may express these conditions via the relations,

$$\omega<0 \quad \text{ascent,}$$

$$-\frac{\partial\theta}{\partial p}>0; \quad -\frac{\partial\theta_e}{\partial p}>0 \quad \text{stable,} \tag{1}$$

and

$$\frac{q}{q_s} > 0.8 \quad \text{saturation}$$

(a list of symbols is provided in Table 2). In the following we shall consider regions where these three conditions are met.

### 6.1 Different formulations of large scale condensation heating:

a) Removal of supersaturation:

In a multilevel numerical prediction model the simplest formulation requires an estimate of saturation at each time step.

Let $\Delta q = q - q_s$ (2)

If $\Delta q > 0$, then at that level of atmosphere one can set the contributions from large scale condensation in the first law of thermodynamics and the water vapor continuity equation by the relations,

$$C_p \frac{T}{\theta} \frac{\partial \theta}{\partial t} = + \frac{L \Delta q}{\Delta t} \tag{3}$$

and

$$\frac{\partial q}{\partial t} = - \frac{\Delta q}{\Delta t} \tag{4}$$

Thus the supersaturation is simply condensed out with an equivalent heat release in the thermal equation at that level of the atmosphere. This is usually done at the end of each time step, $\Delta t$, when other processes in the model have contributed to a positive $\Delta q$.

Calculation of the saturation specific humidity $q_s$ is usually carried out from the use of various approximations such as the Tetens formulae,

vapor pressure $$e_s = 6.11 \exp \left[ \frac{a(T-273.16)}{(T-b)} \right] \tag{5}$$

and saturation specific humidity $$q_s = \frac{0.622 e_s}{p - 0.378 e_s} \tag{6}$$

where the constants a and b are defined in terms of saturation over water (a = 17.26, b = 35.86) and over ice (a = 21.87, b = 7.66). The Tetens formulae has been tested and found to be a reasonable approximation for construction of moist adiabats in the troposphere.

b) <u>Simplest direct calculation</u>:

In the simplest formulation for nonconvective heating,

$$H_{NC} = -L \frac{dq_s}{dt}, \tag{7}$$

is approximated by,

$$H_{NC} = -L\omega \frac{\partial q_s}{}$$

$$\partial p \tag{8}$$

where $\frac{\partial q_s}{\partial p}$ is measured along a local moist adiabat. From the Tetens formula, we obtain,

$$\frac{\partial q_s}{\partial p} = \frac{-0.622 \times 6.11}{p} \exp\left[\frac{a(T-273.16)}{(T-b)}\right] \left[\frac{1}{p} - \frac{a}{(T-b)} - \frac{a(T-273.16)}{(T-b)^2} \frac{\partial T}{\partial p}\right] \tag{9}$$

Conservation of moist static energy along a moist adiabat can be expressed by,

$$gZ_s + C_pT_s + Lq_s = E_s. \tag{10}$$

Upon differentation with respect to pressure we obtain,

$$\frac{g\partial Z_s}{\partial p} + C_p\frac{\partial T_s}{\partial p} + L\frac{\partial q_s}{\partial p} = 0 \tag{11}$$

or

$$-\frac{RT_s(1+0.61q_s)}{p} + C_p\frac{\partial T_s}{\partial p} + L\frac{\partial q_s}{\partial p} = 0 \tag{12}$$

Upon elimination of $\frac{\partial T_s}{\partial p}$ from equation (9) and (10) we obtain the relation,

$$\frac{\partial q_s}{\partial p} = \frac{-0.622 \times 6.11}{p} \exp\left[\frac{a(T-273.16)}{(T-b)}\right] \left[\frac{1}{p} - \frac{a}{(T-b)} - \frac{a(T-273.16)}{(T-b)^2} \times \frac{RT}{C_pP}(1+0.61\, q_s) - \frac{L}{C_p}\frac{\partial q_s}{\partial p}\right] \tag{13}$$

we can solve for $\frac{\partial q_s}{\partial p}$ from this relation and obtain,

$$\frac{\partial q_s}{\partial p} = -\frac{C_1C_2}{1+\frac{L}{C_p}C_1C_3}$$

where

$$C_1 = \frac{-0.622 \times 6.11}{p} \exp\left[\frac{a(T-273.16)}{(T-b)}\right] \tag{14}$$

$$C_3 = \left[\frac{a}{(T-b)} - \frac{a(T-273.16)}{(T-b)^2}\right] \tag{15}$$

and

$$C_2 = \frac{1}{p} - C_3 \left[\frac{RT}{C_p P}(1+0.61q_s)\right] \tag{16}$$

c) <u>A more rigorous formulation</u>:

Kanamitsu (1975) proposed that all of the thermodynamic processes that modify the temperature or humidity instantaneously should be incorporated in a more rigorous formulation of the nonconvective heating. A number of thermodynamic processes such as other heat sources and advection modify the temperature and then the saturation specific humidity. Likewise in the humidity relation a number of processes such as eddy fluxes, evaporation and even convective processes can modify the humidity. The proper definition of $q_s$ or $q$ requires that these are handled appropriately.

Following Kanamitsu (1975) we shall first obtain an equation for the relative humidity r,

we define $$r = \frac{q}{q_s} \tag{17}$$

or

$$q = rq_s \tag{18}$$

we can write,

$$\left[\frac{\partial}{\partial t} + V_H \cdot \nabla\right] q = r \frac{dq_s}{d\theta}\left[\frac{\partial}{\partial t} + V_H \cdot \nabla\right]\theta + q_s \left[\frac{\partial}{\partial t} + V_H \cdot \nabla\right] r \tag{19}$$

We shall define the saturation specific humidity by the approximate relation,

$$q_s = \frac{\epsilon e_s}{p} \tag{20}$$

differentiate with respect to temperature

$$\left.\frac{\partial q_s}{\partial T}\right]_p = \frac{\epsilon}{p}\left.\frac{\partial e_s}{\partial T}\right]_p \approx \frac{\epsilon L q_s}{RT^2} \tag{21}$$

or

$$\left.\frac{\partial q_s}{\partial \theta}\right]_p = \frac{\epsilon L q_s}{RT^2}\left.\frac{\partial T}{\partial \theta}\right]_p$$

$$= \frac{\epsilon L q_s}{RT\theta} \tag{22}$$

Now we can rewrite Eq. 19

$$\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)q = \frac{\epsilon L r q_s}{RT\theta}\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)\theta + q_s\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)r \tag{23}$$

or

$$\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)r = -\frac{1}{q_s}\frac{\epsilon L q}{RT\theta}\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)\theta + \frac{1}{q_s}\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)q \tag{24}$$

This can be further simplified using the first law and the moisture continuity equation. The first law may be written in the form,

$$C_p\frac{T}{\theta}\left[\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)\theta + \omega\frac{\partial\theta}{\partial p}\right] = \sum_i H_i + g\frac{\partial F_\theta}{\partial p} \tag{25}$$

Where $H_i$ denotes the various heating terms, these include; Radiative $H_R$, convective $H_C$, and non-convective heating $H_{NC}$. The humidity equation is written as,

$$L\left[\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)q + \omega\frac{\partial q}{\partial p}\right] = L(E - P_C - P_{NC}) + g\frac{\partial F q}{\partial p} \tag{26}$$

The last terms in the above two equations denote the eddy convergence of fluxes. Upon substitution from Eqs. (23) and (24) into Eqs. (25) and (26) we obtain,

$$\left(\frac{\partial}{\partial t} + V_H \cdot \nabla\right)r = -\frac{1}{q_s}\left[\left\{\frac{\epsilon L q}{RT\theta}\left(-\omega\frac{\partial\theta}{\partial p} + g\frac{\partial F\theta}{\partial p} + \sum H_i\right)\right\}\right.$$

$$- \left\{-\omega\frac{\partial q}{\partial p} + g\frac{\partial F_q}{\partial p} + E - P_C - P_{NC}\right\}\Big]$$

$$= \underbrace{-\frac{\omega}{q_s}\frac{\partial q}{\partial p}}_{I} - \underbrace{\frac{\epsilon L r}{RT\theta}\left\{-\omega\frac{\partial\theta}{\partial p} + g\frac{\partial F_\theta}{\partial p} + \Sigma H_i\right\}}_{II}$$

$$+ \underbrace{\frac{1}{q_s}\left\{g\frac{\partial F_q}{\partial p} + E - P_C - P_{NC}\right\}}_{III} \tag{27}$$

Term I denotes the change in relative humidity arising from vertical advection of specific humidity.

Term II denotes the change in relative humidity due to change in temperature.

Term III denotes a change in relative humidity due to convergence of vertical eddy flux of moisture and condensation processes.

The expression for nonconvective heating $H_{NC}$ is obtained by setting $\frac{\partial r}{\partial t} = 0$ in the above relation, (supersaturation is not permitted).

Thus

$$(V_H \cdot \nabla)r = -\frac{\omega}{q_s}\frac{\partial q}{\partial p} - \frac{\epsilon L r}{RT\theta}\left\{-\omega\frac{\partial\theta}{\partial p} + g\frac{\partial F_\theta}{\partial p} + H_R + H_C + H_{NC}\right\}$$

$$+ \frac{1}{q_s}\left[g\frac{\partial F_q}{\partial p} - \frac{H_C}{L} - \frac{H_{NC}}{L}\right] \tag{28}$$

Upon solving for the nonconvective heating $H_{NC}$ we obtain,

$$H_{NC} = \frac{1}{\frac{C_p T}{\theta L q_s} + \frac{\epsilon L}{RT\theta}}\left[\frac{-\omega}{q_s}\frac{\partial q_s}{\partial p} + \frac{\omega\epsilon L}{RT\theta}\frac{\partial\theta}{\partial p}\right] + R_E \tag{29}$$

Here $R_E$ denotes the remaining terms.

The estimate of $\frac{\partial\theta}{\partial p}$ may be obtained from the equation,

$$C_p\frac{\partial T_s}{\partial p} = \frac{RT_s}{p}\left(1 + 0.61q_s\right) - L\frac{\partial q_s}{\partial p} \tag{30}$$

Let $\pi = \left[\frac{p}{1000}\right]^{R/C_p}$; $T_s = \pi\theta_s$, (31)

hence

$$C_p \frac{\pi\partial\theta_s}{\partial p} + C_p\theta_s\frac{\partial\pi}{\partial p} = \frac{RT_s}{p}\left(1 + 0.61q_s\right) - L\frac{\partial q_s}{\partial p} \tag{32}$$

or

$$\frac{\partial\theta_s}{\partial p} = -\frac{1}{C_p\pi}\left[C_p\theta_s\frac{\partial\pi}{\partial p} - \frac{R\pi\theta_s}{p}\left(1 + 0.61q_s\right) + L\frac{\partial q_s}{\partial p}\right] \tag{33}$$

The subscript s denotes a moist adiabat.

Here the first term in equation (29) describes the moist adiabatic heating, while the last term $R_E$ includes all other effects that level to produce supersaturation (such as eddy flux convergence, other terms of heating $H_R$, $H_S$, $H_C$, large scale horizontal advection).

Thus this more accurate form of nonconvective heating requires estimates of these several other processes. If we set $R_E$ to zero and also assume that the terms containing $\epsilon$ are small compared to the term adjacent to it, we obtain,

$$H_{NC} = -\frac{1}{C_p\frac{T}{\theta}} L\omega\frac{\partial q_s}{\partial p} \tag{34}$$

which is the simpler form of nonconvective heating discussed earlier. Since the large scale dry static stability $-\frac{\partial\theta}{\partial p} > 0$ hence the inclusion of the terms containing $\epsilon$ reduces the magnitude of $H_L$. Thus the simpler form tends to over-estimate the intensity of the so called stable rain and the associated nonconvective heating rate.

Scheme c) seems to be the most accurate among the three above schemes. Scheme a) does not assure a local moist adiabatic sounding $\left(-\frac{\partial\theta_s}{\partial p}, \frac{\partial q_s}{\partial p}\right)$ at the level where the excess moisture is condensed out

since that process is neither isobaric nor does it occur on a constant sigma surface. Scheme b) is evidently an approximation of scheme c) for $\epsilon = 0$ and $R_E = 0$. In the middle latitudes when large scale ascent of stable saturated air occurs, Krishnamurti (1968), the moist upward vertical motion are about 50% larger than estimates made from the use of dry static stability. The rainfall rates are accordingly larger when the local moist adiabatic process is taken into account. In the tropics, where convective and nonconvective processes coexist, estimates of the ratios of nonconvective to convective rain vary over different regions. According to Rosenthal (1969) this ratio is of the order of 0.40 for hurricanes. According to Quah (1985) this ratio is of the order of 0.25 for the monsoon depressions. Based on an examination of GATE disturbances over the Atlantic we find that this ratio for weak Atlantic easterly waves is around 0.15. Thus it appears that as we go towards the more intense disturbances from the waves to depression and to a hurricane the proportion of nonconvective rain appears to increase. That is not surprising since more anvil rain tends to fall from a larger number of tall cumulonimbi in the more intense disturbances.

# 7. RADIATION:

## 7.1 Introduction.

Inclusion of radiative effects in a numerical weather prediction model is important as these give rise to vertical and horizontal thermal gradients which in turn affect the wind circulation. This happens because of radiative heating and cooling and the sensible heat supply from the ground.

Atmospheric heating and cooling due to radiation stems from three sources. These are heating due to the absorption of solar radiation by water vapor, cooling due to long wave radiation and sensible heat supply from the earth. In the third cause the radiative effect is present but only implicitly. Over the land, however, the surface temperatures of the ground becomes high in the day by means of solar radiation.

In this chapter we describe the radiation parameterization scheme used in our model.

## 7.2 Basic Features of the Scheme.

### 7.2.1 Interactive scheme.

Our scheme is an interactive radiative scheme. It is able to interact with humidity and cloud distribution (predicted from the model variables) and thus predict the form of feedback mechanisms involved.

### 7.2.2 Number of cloud layers.

If random overlap of clouds is assumed then the number of possible paths of the solar beam from the top of the atmosphere to

ground is $2^n$ where n is number of cloud layers. In our case we have 3 layers of clouds; low, middle and high; therefore, in all we have 8 possible cloud configurations.

7.2.3 Absorption of radiation.

Only water and water vapour are considered as radiatively active gas in calculating radiative heating rates.

7.2.4 Partition of radiative scheme.

Radiative scheme is essentially divided into 2 parts. Shortwave and longwave. It is possible to do since there is very little overlap in terms of wavelengths between the incoming solar radiation and outgoing terrestrial radiation.

7.2.5 Diurnal changes.

Calculation of zenith angle allows for the seasonal, diurnal and latitudinal changes in the radiative effects.

7.2.6 Emissivity method.

Emissivity method is used for longwave radiation transfer.

7.3 Structure of the Radiation Scheme.

The structure of the radiation scheme consists of 9 program units. Figure 3.1 shows the systematic relationship of each unit with the others. As the radiation calculations are computationally expensive, they are carried out every 3 hours during integration of the model.

7.4 Computations of radiation effects.

The computation of radiative effects has been divided into 4 parts:

1. Specification of clouds.

2. Long wave radiation transfer.

3. Solar short wave radiation and

4. Surface energy balance.

We describe these parts one by one.

7.4.1 Specificaton of clouds.

The specification of clouds is based on certain threshold values of model relative humidity. Three cloud types are defined; low, middle and high. The critical relative humidity for low, middle and high clouds are assigned as 0.66, 0.50, and 0.40 respectively. These values have been derived from average tropical conditions. The fractional low, middle or high clouds are defined as follows:

$$\left.\begin{aligned} &\text{Fractional low cloud:} \quad CL = \frac{(\overline{RH} - RHCL)}{(1 - RHCL)} \\ &\text{Fractional middle cloud:} \quad CM = \frac{(\overline{RH} - RHCM)}{(1 - RHCM)} \\ &\text{Fractional high cloud:} \quad CH = \frac{(\overline{RH} - RHCH)}{(1 - RHCH)} \end{aligned}\right] \qquad (1)$$

where RHCL, RHCM, RHCH are the critical relative humidity for low, middle and high clouds respectively. $\overline{RH}$ is the average relative humidity taken over the layers in which the initial or predicted relative humidity exceeds the critical relative humidity.

The upper limit of fractional area covered by any type of cloud is kept as 1 and lower limit as zero.

There are eight possible cloud configurations in the present scheme:

i) Clear

ii) Low clouds only

iii) Low and middle clouds

iv) Low, middle and high clouds

v) Middle cloud only

vi) High cloud only

vii) Middle and high clouds

viii) Low and high clouds

For given model temperature and humidity values radiative fluxes are first calculated for all of cloud configuration present. Different weights are assigned to these fluxes depending upon the cloud amounts within a grid square area. The weights are:

$$\left.\begin{aligned}
\text{Coefficients (i)} &= (1-CL)\ (1-CM)\ (1-CH) \\
\text{Coefficients (ii)} &= CL\cdot(1-CM)\ (1-CH) \\
\text{Coefficients (iii)} &= CL\cdot CM\cdot(1-CH) \\
\text{Coefficients (iv)} &= C_L\cdot C_M\cdot C_H \\
\text{Coefficients (v)} &= (1-C_L)\cdot C_M\ (1-C_H) \\
\text{Coefficients (vi)} &= (1-C_L)\ (1-C_M)\ C_H \\
\text{Coefficients (vii)} &= (1-C_L)\ C_M\cdot C_H \\
\text{Coefficients (viii)} &= C_L\cdot(1-C_M)\cdot C_H
\end{aligned}\right] \qquad (2)$$

We also put the following constraints on specification of clouds.

1. The bases of the clouds are fixed.
2. The maximum allowed thickness is 200mb.
3. The cloud layers considered have to be thicker than 50mbs.

The allowed bases and tops of the clouds are as follows:

| | Base (mb) | Top (mb) |
|---|---|---|
| Low cloud | 950 | 750 |

| | | |
|---|---|---|
| Middle cloud | 750 | 450 |
| High cloud | 450 | 150 |

It was noticed that CL, CM and CH overspecified the cloud fractional area. The following modification is presently used.

$$C_x = C_x^2 \tag{3}$$

x is cloud type - low, middle or high. The cloud specification is carried out in the subroutines RADN and CLOUD.

7.4.2 Long wave radiation flux.

Longwave radiation is absorbed and emitted by atmospheric gases, clouds and the earth's surface. The absorption by water drops is so strong that most clouds in the lower troposphere absorb essentially all the incident radiation flux and emit a flux which is similar to that from a blackbody at the cloud boundary temperature. Cloud radiative properties in the longwave region are thus independent of wavelength for practical purposes.

Emissivity method is used for the computation of long wave radiation fluxes. All of the long wave radiation originates at the earth's surface or from the atmosphere and clouds. We consider in our scheme absorption by water vapor only. We neglect absorption by carbon-dioxide because it is well mixed and its role in the tropical troposphere is not large. The role of ozone warming is also not significant for a tropospheric model.

Cloud free case

Upward flux: The total upward flux of long wave radiation at a reference level in cloud free case is given as sum of two parts.

$$F_i\uparrow = \sigma T_g^4 \left(1 - \epsilon[W_B - W_i]\right) + \int_{W_i}^{W_B} \sigma T^4 \frac{\partial \epsilon[W - W_i]}{\partial W} dW \tag{4}$$

The first term on right hand side gives the part that comes up to the level i from the earth's surface (ground) and the second term gives the part that is emitted by layers between the ground and the reference levels.

$\sigma$: Stefan Boltzmann constant

$W_B$: path length at the ground

$W_i$: path length at the reference level

$\epsilon$: The emissivity as function of the path length.

<u>Downward flux</u>: It is given by just one term.

$$F_i\downarrow = -\int_0^{W_i} \frac{\sigma T^4 \, \partial\epsilon(W_i - W)}{\partial W} \, dW \qquad (5)$$

$\frac{\partial\epsilon}{\partial w}$ in equations is a measure of change of emissivity with respect to the path length.

<u>Cloudy case</u>: Let us consider that there is one cloud layer above the reference level. Then the cloud will affect the downward flux at reference level.

$$F_i\downarrow = \sigma T_{cb}^4 \, (1 - \epsilon[W_i - W_{cb}]) - \int_{W_{cb}}^{W_i} \frac{\sigma T^4 \, \partial\epsilon[W_i - W]}{\partial W} \, dW \qquad (6)$$

$T_{cb}$: temperature at the cloud base

$W_{cb}$: path length at cloud base

When there is one cloud layer below the reference level, upward flux at the level is computed as:

$$F_i\uparrow = \sigma T_{ct}^4 \, (1 - \epsilon[W_{ct} - W_i]) - \int_{W_i}^{W_{ct}} \frac{\sigma T^4 \, \partial\epsilon[W - W_i]}{\partial W} \, dW \qquad (7)$$

$T_{ct}$: Temperature at cloud top

$W_{ct}$: Path length at cloud top level

For multiple cloud layer the same equations (6) and (7) are applied in similar ways. These fluxes are determined in the routine FLUX. We obtain the emissivity values from the tables of emissivity as a function of path length, prepared by Staley and Jurica. The error estimates are of the order of 0.1°C/day in the atmosphere. The emissivity tabulations are provided in subroutine EMTAB. This table expresses the emissivity $\epsilon$ as a function of the path length which in turn depends on humidity, temperature and pressure. The real data or the predicted data would, in general, fall within the range of tabulated values of the path length. Thus, we need to interpolate the values of emissivity corresponding to the desired values of the path length.

As mentioned earlier, we consider absorption only by water vapor. The path length is measured from the top of the model atmosphere, $P_T = 0.1P_s$ to the reference level, $P_i$. The contribution from the region above the model atmosphere ($0 \leq P \leq P_T$) is constructed from a standard atmospheric sounding. The path length is defined as:

$$W(P) = \frac{1}{g} \int_{P_T}^{P_i} q \left(\frac{P}{P_o}\right)^{0.85} \left(\frac{T_o}{T}\right)^{0.5} dp + W_T \qquad (8)$$

q: specific humidity

$W_T$: Path length above the model top. We assume an isothermal stratosphere (220°K) and a front point of 210°K. This provides a measure of water vapor in the stratosphere. This is defined in routine SLR and its value is $4.7 \times 10^5$.

$\left[\frac{P}{P_s}\right]^{0.85}$ gives the pressure reduction effect and $\left[\frac{T_s}{T}\right]^{0.5}$ gives the

correction for variation of temperature.

7.4.3 Short-wave radiation flux.

The solar radiation at the top of the atmosphere is divided into scattered and absorbed part. Short wave radiation is depleted due to absorption by water vapor and Rayleigh scattering by aerosols.

$$\text{Scattered part:} \quad S^s = 0.651\ S_o \cos \zeta \tag{9}$$

$$\text{Absorbed part:} \quad S^a = 0.349\ S_o \cos \zeta \tag{10}$$

So is the solar constant, $\zeta$ is the zenith angle.

From empirical studies, an absorptivity function A[W] is defined, which given the amount of incoming solar radiation that is depleted by the absorption constituent.

$$A[W] = 0.271\ [W \sec \zeta]^{0.303} \tag{11}$$

The direct solar radiation reaching a reference level i is computed as follows:

$$S^a\ (1 - A[W_i \sec \zeta]) \tag{12}$$

Only the absorbed part of direct solar radiation is attenuated by the process.

To estimate the net downward flux of short wave radiation, we compute the amount of diffuse radiation that comes up from the earth's surface. For this albedo of earth's surface and the reference level is taken into account.

The amount of diffuse short wave radiation reflected from the earth's surface is:

$$S^a\{1 - A[W_o]\}\alpha_s \tag{13}$$

$W_o$: Path length at earth's surface.

Diffuse radiation reaching the reference level is determined as:

$$S^a\{1 - A[W_o \sec \zeta]\}\alpha_s \{1 - A[1.66(W_o - W_i)]\} \quad (14)$$

The absorptivity of diffuse radiation is multiplied by 1.66 to account for the increased path length experienced by the diffuse radiation.

The total downward flux is determined as:

$$S_i^a = S^a [1 - A[wSec\ \zeta]] - S^a\{1 - A[W_o \sec \zeta]\}\alpha_s \quad (15)$$
$$x\{1 - A[1.66(W_o - W_i)]\}$$

When there is a cloud layer present below the reference level i, the net downward flux is computed using the relation:

$$S_i^a = S^a(1 - A[W_i \sec \zeta]) - S^a(1 - A[W_{ct} \sec \zeta])\alpha_c$$
$$x(1 - A[1.66(W_{ct} - W_i)] \quad (16)$$

where $\alpha_c$ is albedo of cloud.

For computation of the amount of short wave radiation passing through a cloud layer absorptivity of cloud is defined. Since in a cloud both liquid water and water vapor exist, absorptivity function is given as $A[W_{ci}^*]$ where $W_{ci}^*$ is the augmented path length which accounts for the equivalent amount of water vapor within the cloud.

The amount of direct solar radiation reaching below a single cloud atmosphere is computed as:

$$S^a(1 - A[W_{ct} \sec \zeta])\ (1 - \alpha_c)\ (1 - A[W_{ci}^*]) \quad (17)$$

For determining the downward flux of short wave below a single cloud in the atmosphere the upward flux of diffuse radiation rising from the earth's surface is also considered. The total downward flux of absorbed short wave radiation at a reference level below a single cloud layer is computed as:

$$S_i^a = S^a(1 - A[W_{ct} \sec \zeta])\ (1 - \alpha_c)\ \{1 - A[W_c^* + 1.66(W_i - W_{cb})] -$$
$$(1 - A[W_c^* + 1.66(W_o - W_{cb})])\alpha_s\ (1 - A[1.66(W_o - W_i)])\} \quad (18)$$

When more than one cloud layer is present the same procedure is logically applied.

These fluxes are computed in routine SLR. The absorptivity function A[W] is computed in function subroutine FCN.

Scattered Part

Rate of warming of the atmosphere by scattered part $S^s$ of the incoming solar radiation, is very small. However it is not negligible in the heat balance at the earth's surface.

For considering scattering effects, albedo of air is calculated using an empirical relation.

$$\alpha_o = \text{ALBAIR} = 0.085 - 0.245 \log \left(\frac{P_s}{1000} \cos \zeta\right) \tag{19}$$

Downward flux of scattered solar radiation in case of clear sky, at the earth's surface is computed by the following relation.

$$S_i{}^s = \frac{S^s\ (1 - \alpha_o)}{(1 - \alpha_o \alpha_s)} \tag{20}$$

$\alpha_s$: albedo of the earth's surface.

When clouds are present the following relationship is used:

$$S_i{}^s = S^s\ (1 - \beta)/(1 - \beta\alpha_s) \tag{21}$$

where $\beta = 1 - (1 - \alpha_o)\ (1 - \alpha_{c_i})$ when cloud $c_i$ is present. The same formula is applied when multiple clouds are present. In that case $\beta$ is determined extending the above example.

The scattered part of the flux computation is carried out in the main routine RADN. The constants required for the radiation computations are defined in subroutines CONST and CALPHI.

Radiative heating rate

Long wave warming (cooling) rate is computed as the divergence (or convergence) of the net long wave fluxes at the interface of the layers (i.e. $\sigma$ levels).

$$\left(\frac{\partial T}{\partial T}\right)_{\text{long wave}} = \frac{g}{cp} \frac{\partial F}{\partial p} \tag{22}$$

The short wave warming rate at a level is computed as divergence of the net short wave fluxes at the layer interface.

$$\left(\frac{\partial T}{\partial t}\right)_{\text{short wave}} = \frac{g}{cp} \cdot \frac{\partial S_i{}^a}{\partial p} \tag{23}$$

where $S_i{}^a$ includes absorbed part of direct and reflected diffuse radiation.

These computations are carried out in routine RADN. First the computations are made for each cloud configuration. Then these are modified by the coefficients of various cloud configurations to give net warming and cooling rates. The sum of long wave warming/cooling rates and short wave cooling rates gives the net temperature change due to radiation. This is returned in array TEMPT giving change of temperature at N $\sigma$ levels of the model, due to radiative effects.

7.4.4 Radiative Heat Balance at the Earth's Surface.

We carry out the radiative heat balance at the earth's surface for the points that lie over land. We assume that land has zero heat capacity and set the time rate of change of energy accumulation at the earth's surface to zero. Then, we solve for the ground temperature using a balance condition. The balance is among the net radiative and net soil heat fluxes at the earth's surface. The surface energy is expressed by:

$$F_L + F_S + H_S + H_L = 0 \tag{24}$$

i.e. Net long wave radiation + net short wave radiation + sensible heat flux + latent heat flux = 0.

The net downward long wave flux at earth surface is computed as:

$$F_L = F_{LG} - \sigma T_g^4 \tag{25}$$

$F_{LG}$ is downward flux

The net short wave flux at earth surface is computed as:

$$F_S = F_{SG} - \alpha_S F_{SG} \tag{26}$$

where $\alpha_S$: earth's albedo. $F_{SG}$ includes both absorbed and scattered part.

The sensible and latent heat fluxes are computed using bulk formulae in routine SFXPAR. These depend on the difference $T_g - T_s$ and $q_g - q_s$.

$$H_S = \rho_o C_p C_D | V_o | (T_g - T_s) \tag{27}$$

$$H_L = \rho C_q L | V_o | (q_g - q_s) \tag{28}$$

$T_g$, $q_g$: ground temperature and specific humidity

$T_s$, $q_s$: air temperature and specific humidity

A first guess of $T_g$ is provided as surface temperature T 1000. The first guess of air temperature is obtained by extrapolating temperature of lowest $\sigma$ layer of the model. The ground specific humidity is defined by the relation:

$$q_g = GWq_{sg} \tag{29}$$

where $q_{sg}$ is the saturation specific humidity at ground. GW is ground wetness parameter and is empirically defined by the relation:

$$GW = (\alpha_{max} - \alpha) \cdot (1 - R \cdot H \cdot surface) / (\alpha_{max} - 0.05) \tag{30}$$

where $\alpha_{max}$: maximum albedo over the domain

$\alpha$: albedo at a given point

The parameter GW is restricted to lie between 0 and 1. The surface energy balance equation is written as:

$$FTS = T_g^4(1 - \alpha_s) - \sigma T_g^4 + x_s(T_s - T_g) + x_L(q_s - q_g) = 0 \quad (31)$$

where $x_s = \rho_o\ C_p\ C_D\ V_o$ (32)

$x_L = \rho_o\ C_q\ C_D\ V_o$ and (33)

$V_o$ = surface velocity

The energy balance equation is solved by Newton Raphson method to get value of $T_g$. The solution also provides measure of long wave radiation and fluxes of sensible and latent heat. The Newton Raphson method is an iterative procedure aimed at minimising FTS. We start with a first guess value of $T_g$ and calculate:

i) $$\frac{\partial FTS}{\partial T_g} = -4\sigma T_g^3 - x_s - x_L q_g \frac{(273.16x25.22-5.33T_g)}{T_g^2} \quad (34)$$

ii) Update a value of $T_g$

$$T_{g_{new}} = T_{g_{old}} - \frac{\partial FTS}{\partial T_g} \quad (35)$$

iii) Specific humidity at ground, $q_g$ is computed $q_g = q_{sq}GW$

The steps (i) - (iii) are repeated and usually convergence is achieved in a couple of scans. This calculation is done in the routine TG.

Subroutine RADN is the main routine in the entire calculation. The final results of fluxes are expressed in units of watts/$m^2$ and the cooling (or warming) rates are expressed in deg c/day. These calculations are done in subroutine RADN.

7.5 Input to the Radiation Scheme.

7.5.1 Model variables.

Temperature values are needed at the boundaries of the layers where the fluxes are computed. They are derived from the layer tem-

peratures ($\sigma$ levels) by simple averaging. At bottom and top levels of the model the temperature values are linearly extrapolated from the inner layer temperatures of the vertical domain. The relative humidity is obtained from the model variable specific humidity and averaged to layer boundaries. When humidity is needed at layers, the interpolated values are averaged again. This ensures that the moisture input to the radiation scheme is as smooth as the temperature input.

7.5.2 Clouds.

The cloud cover at each level is obtained from the relative humidity at that level. Procedure to obtain the cloud cover has been described earlier.

It has been found that because of poor vertical resolution, boundary layer clouds are badly represented in the model and produce wrong feedback between radiation, vertical diffusion and condensation. Therefore any cloud that would be present within a convective boundary layer (i.e. whose potential temperature is smaller than that of the ground) is eliminated from the input to the radiation scheme.

7.5.3 Ground albedo.

Monthly mean values of surface albedo obtained from NCAR are presently used.

7.5.4 Solar zenith angle.

The solar zenith angle is given by the relation:

$$\cos \zeta = \sin \phi \sin \delta + \cos \phi \cos \delta \cos h \qquad (36)$$

where $\zeta$: zenith angle

$\phi$: latitude

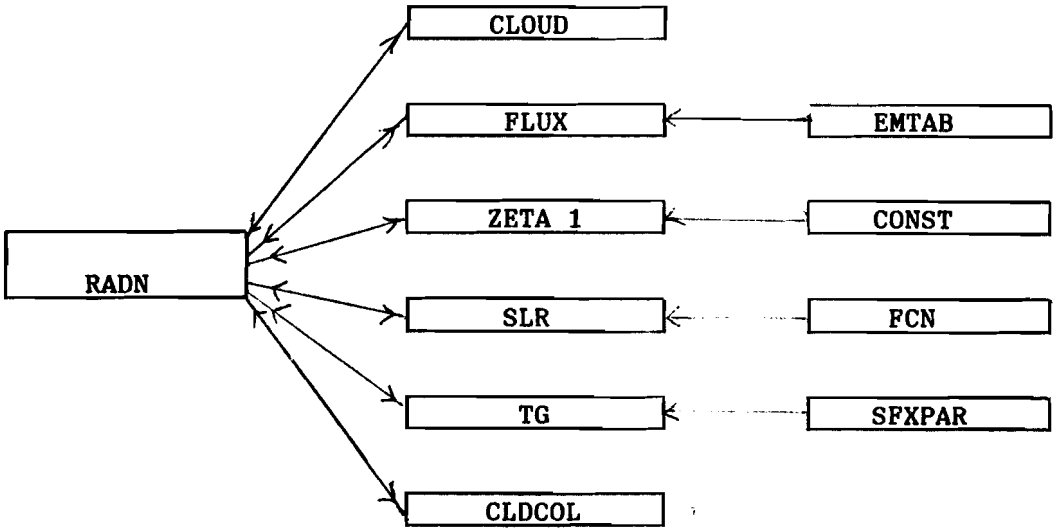
$\delta$: declination

h: hour angle measured from the local solar noon

The seasonal change in the radiation computations is brought in through the declination. The diurnal change comes through the hour angle, h, and the latitudinal effect appears through $\phi$.

The zenith angle is computed in routine ZETA1. The declination angle and cos and sin of latitude $\phi$ are computed in routine CONST. The hour angle is determined in main radiation routine RADN. The zenith angle is averaged over 3 hours for use in radiation scheme.

Figure 4




The structure of radiative scheme.

8. Dry Convective Adjustment

Although superadiabatic lapse rates can exist in the atmosphere over some places for short periods, this is not desirable from the point of numerical weather prediction. If a region of superadiabatic lapse rate is present locally, small perturbations can lead to buoyancy accelerations that require for a total description a very fine mesh model. Larger mesh size models will see this growth of local buoyancy accelerations as noise. These coarse meshes do not resolve the energetics of growing dry convective elements, and uncontrolled growth of noise can occur. This can be prevented via a strong vertical diffusion in a numerical weather prediction model. More often one makes use of the dry convective adjustment procedure which replaces the superadiabatic lapse rate by a neutral sounding such that the total dry static energy remains an invariant before and after the adjustment over the vertical. There exists a pressure level, $p_r$, above the level of minimum dry static energy, which determines the top of the moist convective adjustment. At this level if the observed dry static energy is Er then

$$E_r\,(p_o - p_r) = \int_{p_r}^{p_o} E dp \qquad (1)$$

where E is the dry static energy of the sounding, and $p_o$ is the surface pressure assumed to be 1000 mb. The procedure consists of an integrative E for a level pr = po - $n\Delta p$ (starting from n = 1 where $\Delta p$ is the grid interval along the vertical) to the pressure level po. One continues this until the left hand side of above equation becomes

equal to or greater than the right hand side. When this is done the adjusted sounding can be recovered from the conservation of dry static energy, the Poisson equation and the hydrostatic law. The adjustment cools the lower layers and warms the upper layers of the adjusted sounding. The warming of the upper layer can produce dry instability above that layer, and one has to successively remove that as well by a similar adjustment procedure. In practice we take the following steps.

The adjustment is done over successive layers, in which one compares the potential temperature over two adjacent layers for instability. This continues until one finds a stable layer. An average of the dry static energy is computed over that depth (i.e., up to the level where the sounding becomes stable), and a constant theta in the vertical is the adjustment. The adjusted potential temperature is computed at every level using the average dry static energy for that depth. The checking is done over again but this time T(k) = T(k+1) since the adjusted sounding is neutral ($\theta$ constant with height) over the first depth and k is the level measured from bottom up.

<u>Some useful equations</u>

Dry static energy given by: $E = gz + C_pT$

$$\theta = \frac{po}{p}^{\kappa}T \; ; \; \kappa = R/c_p \; ; \; 1/p = \frac{RT}{p} \qquad (2)$$

Differentiating with respect to pressure we get,

$$\frac{\partial E}{\partial p} = g\frac{\partial z}{\partial p} + C_p \frac{\partial T}{\partial p} \qquad (3)$$

and

$$\frac{\partial \theta}{\partial p} = \frac{po^{\kappa}}{p} \frac{\partial T}{\partial p} + T(p_o)^{\kappa} \left(-\kappa p^{(-\kappa-1)}\right) \qquad (4)$$

or

$$\frac{\partial \theta}{\partial p} = \frac{po^{\kappa}}{p} \frac{\partial T}{\partial p} - \frac{TR}{C_p p} \qquad (5)$$

using the hydrostatic equation, $\frac{\partial z}{\partial p} = -\frac{1}{\rho g}$ ; $g\frac{\partial z}{\partial p} = -\frac{RT}{p}$ (6)

we get $\frac{1}{cp}\frac{\partial E}{\partial p} = -\frac{RT}{pC_p} + \frac{\partial T}{\partial p}$ (7)

or $= \frac{p}{po}^{\kappa}\frac{\partial \theta}{\partial p}$ (8)

or $\frac{\partial E}{\partial p} = C_p \frac{p}{po}^{\kappa}\frac{\partial \theta}{\partial p}$ (9)

The code is provided below.

Anthes, R.A., 1984: Cumulus parameterization. The Dynamics of clouds and Mesoscale Meteorological System (To be published by Academic Press).

Businger, J.A., J.C. Wyngaard, Y. Azumi and E.F. Bradley, 1971: Flux profile relationship in the atmospheric surface layer. J. Atmos. Sci., 28, 181-189.

Chang, L.W., 1978: Determination of surface flux of sensible heat, latent heat and momentum utilizing the bulk Richardson number. Papers in Meteor. Res., 1, 16-24.

Charnok, H., 1955: Wind stress on a water surface. Quar. J. Roy. Met. Soc., 81, pp. 639-640.

Delsol, F., K. Miyakoda and R. Clarke, 1971: Parameterization processes in the surface boundary layer of an atmospheric circulation model. Quar. J. Roy. Met. Soc., 97, 181-208.

Heckley, W.A., 1984: A review of numerical experimentation at the ECMWF with emphasis on Tropical prediction. GARP Special Report No. 44, WMO, Geneva, pp. IV-1 - IV-21.

Kanamitsu, M. 1975: On numerical prediction over a global tropical belt. Report No. 75-1, Dept. of Meteorology, Florida State University, Tallahassee, Florida, 32306, pp.1-282.

Krishnamurti, T.N., 1962: Numerical intergration of primitive equations by a quasi-Lagragian advective scheme. J. Appl. Meteor., 1, 503-521.

Krishnamurti, T.N., 1968: A diagnostic balance model for studies of weather systems of high and low latitudes, Rossby number < 1. Mon. Wea. Rev., 96, pp. 197-207.

Krishnamurti, T.N., 1969: An experiment in numerical prediction in equatorial latitudes. Quart. J. Roy. Meteor. Soc., 95, 594-620.

Krishnamurti, T.N., 1985: Numerical Weather Prediction in Low Latitudes. Advances in Geophysics, Vol. 28. Academic Press.

Krishnamurti, T.N., Y. Ramanathan, H. Pan, R. Pasch, and J. Molinari, 1980: Cumulus parameterization and rainfall rates I. Mon. Wea. Rev., 111, 815-828.

Krishnamurti, T.N., S. Low-Nam and R. Pasch, 1983. Cumulus parameterization and rainfall rates II. Mon. Wea. Rev., 111, 815-828.

Kudoh, T., 1978: A semi-implicit formulation of Prognostic equations in a sigma-coordinate system. The Geophysical Magazine, 38, 45-79.

Mathur, M.B., 1970: A note on an improved quasi-Langragian advective scheme for primitive equations. Mon. Wea. Rev., 98, 214-219.

Mathur, M.B., 1983: A quasi Lagrangian regional model designed for operational weather prediction. Mon. Wea. Rev., 111, 2087-2098.

Quah, L.C., 1985: On the Asian Monsoon Heat Sources and Energetics. Ph.D. Dissertation, pp. 1-200. Dept. of Meteorology, Florida State University, Tallahassee.

Robert, A., 1981: A stable numerical integration scheme for the primitive meteorological equations. Atmos. Ocean, 19, 35-46.

Robert, A., 1982: A semi-Lagrangian and semi-implicit scheme for the primitive meteorological equations. J. Meteor. Soc. of Japan, 60, 319-325.

Robert, A., J. Henderson and C. Turnbull, 1972: An implicit time integration scheme for baroclinic models of the atmosphere. Mon. Wea. Rev., 100, 329-335.

Robert, A., T.L. Yee and H. Ritchie, 1985: A semi-Lagrangian and semi-implicit numerical integration scheme for multilevel atmospheric models. Mon. Wea. Rev., 113, 388-394.

Rosenthal, S.L., 1969: Numerical Experiments with a mult-level primitive equation model designed to simulate the development of Tropical Cyclones. Experiment I. U.S. Department of Commerce, ESSA Technical Memorandum NO NHRL-82, 32pp.

Staley, D.O., and G.M. Jurica, 1970: Tables of Flux emissivity water vapor, carbon dioxide, and ozone. J. Appl. Meteor., 9, 365-372.

Tiedke, M., 1984: The sensitivity of the time-mean large-scale flow to cumulus convection in the ECMWF model. Workshop on convection in large-scale numerical model. European Centre for Medium Range Weather Forecast pp. 297-317.

Part II

## CODE DOCUMENTATION

Contents


| Item No. | Routine |
|---|---|
| 1 | MAIN-PROGRAM FSULAM |
| 2 | Input Routines |
| 2.1 | SUBROUTINE START |
| 2.2 | SUBROUTINE RESTART |
| 3 | Thermodynamics |
| 3.1 | SUBROUTINE TBAR |
| 3.2 | SUBROUTINE LAPSE |
| 3.3 | SUBROUTINE GUESS |
| 4 | Dynamics |
| 4.1 | SUBROUTINE YCAL |
| 4.2 | SUBROUTINE YZADD |
| 4.3 | SUBROUTINE XCAL1 |
| 4.4 | SUBROUTINE XYZW |
| 4.5 | SUBROUTINE QCAL1 |
| 4.6 | SUBROUTINE DIAGNOS |
| 4.7 | FUNCTION XYINT4 |
| 4.8 | FUNCTION XYINT |
| 4.9 | SUBROUTINE VADV |
| 5 | Convective Adjustment |
| 5.1 | SUBROUTINE DRYCONT |
| 5.2 | SUBROUTINE T2TT |
| 5.3 | SUBROUTINE SUPSEA |
| 5.4 | SUBROUTINE SUPADJ |

| | |
|---|---|
| 5.5 | SUBROUTINE SUPADJ1 |
| 5.6 | SUBROUTINE SPHUM |
| 6 | Radiation |
| 6.1 | SUBROUTINE RADN |
| 6.2 | SUBROUTINE CLOUD |
| 6.3 | SUBROUTINE FLUX |
| 6.4 | SUBROUTINE SLR |
| 6.5 | SUBROUTINE TG |
| 6.7 | SUBROUTINE EMTAB |
| 6.8 | SUBROUTINE ZETA1 |
| 6.9 | FUNCTION FCN(X) |
| 7 | Cloud parameterization |
| 7.1 | SUBROUTINE CVHEAT |
| 7.2 | SUBROUTINE SSHEAT |
| 7.3 | SUBROUTINE HST |
| 7.4 | SUBROUTINE SHACON |
| 8 | Planetary Boundary Layer |
| 8.1 | SUBROUTINE PBLFLX |
| 8.2 | SUBROUTINE LCLSUR |
| 8.3 | SUBROUTINE SFLX |
| 8.4 | SUBROUTINE SFXPAR |
| 9 | Forecast and Diagnostics |
| 9.1 | SUBROUTINE FORCST |
| 9.2 | SUBROUTINE RAINFAL |
| 9.3 | SUBROUTINE CHECK |
| 10 | Service routines |
| 10.1 | SUBROUTINE BSM1 |

| | |
|---|---|
| 10.2 | SUBROUTINE BSMOOTH |
| 10.3 | SUBROUTINE CALPHI |
| 10.4 | SUBROUTINE COFF |
| 10.5 | SUBROUTINE CONST |
| 10.6 | SUBROUTINE EIGEN |
| 10.7 | SUBROUTINE ELLIPT |
| 10.8 | SUBROUTINE INVERT |
| 10.9 | SUBROUTINE ITERATE |
| 10.10 | SUBROUTINE MOVE |
| 10.11 | SUBROUTINE MULT |
| 10.12 | SUBROUTINE RUTISH |
| 10.13 | SUBROUTINE ZERO |

Program: FSULAM

Abstract: This is main program of the grid point multi-level $\sigma$ co-ordinate primitive equation model. Various constants and arrays required by various subroutines are defined or initialized in this routine. This routine calls various subroutines for carrying out semi-Lagrangian advection, integration or calculating forcing terms due to convection, radiation, friction etc. The input data is read into the memory in this routine. The output is also written in this routine. It has 9 common blocks through which it communicates with other parts of the program.
Besides integration this program also calls routines for intialization of input fields.
Various diagnositics are printed for monitoring the program.

Subroutines called: CONST, CALPHI, COFF, INOUT, TBAR, LAPSE, START, RESTART, SPHUM, CHECK, GUESS, ZERO, YCAL, YZADD, MOVE, QCAL1, ELLIPT, BSMOOTH, DIAGNOS, SUPADJ, FORCST, RAINFAL

Called by: None

Description of program card

TAPE 91: Input file. 46 records of initial input are read from this tape. Input is read from this tape more than once. When initialization is invoked, initialized input is written on this.

TAPE 92: Initialized input and other diagnostics of initialization routines are written on this file.

TAPE 93: Intermediate values of non-linear tendencies are written on this file, and then read as values of previous time step.

TAPE 95: Constant fields of surface albedo, sea surface temperatures and smooth terrain are read from this file at the initial time step in routine START.

TAPE 96: Constant fields are written on this file like grid interval, latitude and Coriolis force and the $\tilde{\sigma}$ layer and $\sigma$ level values etc.

TAPE 97: This is main output file. On this 55 records of forecast fields are written at every forecast time step.

TAPE 98: On this output file the rainfall arrays and surface heat and moisture fluxes are written as 5 spearate records at every rainfall output time step.

TAPE 99: On this file the arrays TB, TBB, GAMAB, ALBD, TERR, TSUB, QSUB and GNEXT are written and these are read from this file whenever the program is restarted.

Description of parameter statement

| | |
|---|---|
| L | Number of grid points in x-direction |
| LM1 | L-1 |
| M | Number of grid points in y-direction |
| MM1 | M-1 |
| N | Number of layers in vertical |
| NM1 | N-1 |
| NP1 | N+1 |
| LP1 | L+1 |

Description of common blocks

Unlabelled common

| | |
|---|---|
| UU(L,M,N) | U component of wind. Prognostic field |
| VV(L,M,N) | V component of wind. Prognostic field |
| PP(L,M,N) | Geopotential heights are read in this variable. During the integration it represents model parameter. $P = gz + RT^* \ln P_s$ At the time of output again it contains geopotential height. Prognostic Field. |
| WW(L,M,NM1) | Vertical velocity $\dot{\sigma}$ |
| QQ(L,M,N) | Specific humidity |
| PL(L,M) | Log of surface pressure '$p_s$' |
| A1(L,M,N) | Working array |
| A2(L,M,N) | Working array |
| A3(L,M,N) | Working array |
| A4(L,M,N) | Working array |
| A5(L,M,N) | Working array |
| B1(L,M,N) | Working array |
| B2(L,M,N) | Working array |

| | |
|---|---|
| B3(L,M,N) | Working array |
| B4(L,M,N) | Working array |
| Common /WORK/ | |
| DT | Time step |
| DX(M) | Grid distance in x-direction |
| DY | Grid distance in y-direction |
| DLAM | Grid interval in terms of °longitude |
| DPHI | Grid interval in terms of ° latitude |
| PHI(M) | Latitude values |
| SIG(NP1) | $\sigma$ values of layer boundaries. Solid $\sigma$ levels. |
| SIGT(N) | $\sigma$ values of layers. Dotted levels |
| COR(M) | Coriolis parameter. $2\ \Omega\ \mathrm{Sin}\phi$ |
| RAD | Earth's radius |
| G | Gravity |
| R | Gas constant for dry air |
| CP | Specific heat at constant pressure |
| PI | 3.1415 |
| TB(N) | Temperature of $\tilde{\sigma}$ layers. Area averaged |
| TBB(NP1) | Temperature of layer boundaries. Area averaged |
| GAMAB(N) | Parameter $\gamma$ (a function of lapse rate) in layers. Area averaged |
| T(N) | Temperatures of $\sigma$ layers |
| TT(NP1) | Temperature of layer boundaries |
| GAMA(N) | $\gamma$ in sigma layers |
| EGN(NP1) | Eigen values representing NP1 modes in vertical. Where NP1 is number of layer boundaries |
| TRANS(NP1,NP1) | Transformation matrix. Intermediary array |
| EE(NP1,NP1) | Working array |

| | |
|---|---|
| TINV(NP1,NP1) | Eigen vector matrix corresponding to EGN |
| AINV(NP1,NP1) | Working array |
| T1000 | Temperature at the lowest layer boundary. Also surface temperature over oceans |
| Q1000 | Specific humidity at the lowest layer boundary. Also surface humidity over oceans. It is function of T1000 |
| RH(N) | Relative humidity in $\sigma$ layers |
| TSUB(L,M) | Surface temperature array |
| QSUB(L,M) | Surface humidity array |
| Common /RSR/ | |
| RADC | Degree to radians conversion factor |
| GAM | Dry adiabatic lapse rate $g/C_p$ |
| GINV | 1/G |
| ALBAIR | Albedo of air for scattering part of solar direct radiation |
| ROMEGA | Earth's rotation rate in degrees/hour |
| RS | Solar constant |
| DELTA | Declination angle |
| COSD | Cos of declination |
| SIND | Sine of declination |
| SINPHI(M) | Sin$\phi$; $\phi$ is latitude |
| COSPHI(M) | Cos$\phi$ |
| FACTOR | Multiplication factor |
| SIGMA | Stefan Boltzmann constant |
| RTOP | Radiation at top of atmosphere |
| TEMPT(N) | Temperature (heating) due to radiative effects |
| ALBD(L,M) | Surface albedo array. It is constant during integration. Monthly mean albedos are used |

| | |
|---|---|
| SST(L,M) | Sea surface temperature array. It is constant during integration. Monthly mean values are used |
| TERR(L,M) | Smooth topography array. It s constant during integration |
| SLAT | Southern-most latitude. Southern boundary |
| WLONG | Western-most longitude. Western boundary |
| DLAMD | Grid interval in °longitude |
| DYPHI | Grid interval in °latitude |
| I | x coordinate index of a point |
| J | y coordinate index of a point |
| N1 | Number of layers in vertical. Used in radiation package. At other places N1 = N-1 |
| RHO | Density of air at surface |
| YL | Latent heat of evaporation |
| KT | Time step count |
| KRAD | Radiation time step count |
| TIME | Time of beginning of integration of the model |
| DZHOUR | Map time of the beginning of integration |
| MON | Month |
| NDAY | Starting day of integration |

Common /ALA/

| | | | |
|---|---|---|---|
| TEMP(N) | Temperature in $\sigma$ layers with order reversed i.e. lowest $\sigma$ layer to uppermost $\sigma$ layer | | |
| Q(N) | Specific humidity in $\sigma$ layers with order reversed | | |
| PZ(NP1) | Pressure at layer boundaries with order reversed | | |
| PT(N) | Pressure in $\sigma$ layers with order reversed | | |
| DPZ(N) | PZ(K+1) - PZ(k) | k=1(1)N | Bottom to top |
| DPT(N) | $\frac{PZ(k) + PZ(k+1)}{2}$ | K=1(1)N | Bottom to top |

| | |
|---|---|
| DPTIV(N) | 1.0/DPT(N) |
| DPZIV(N) | 1.0/DPZ(N) |
| Z10(L,M) | A working array. Used to represent surface topography |

Common /CVH/

| | |
|---|---|
| WBAR | Vertically averaged vertical velocity $\dot{\sigma}$ |
| ZETA | Relative vorticity at 0.8 $\sigma$ level |
| LCLF | Level of vanishing buoyancy. Cloud top |
| LCONV(N) | Instability index of $\sigma$ layers. Initialized as 89 in routine YCAL. Modified in routine CVHEAT, SSHEAT and SUPADJ |
| HLC(N) | Convective heating in $\sigma$ layers from bottom layer to top layer |
| QDIF(N) | Difference between specific humidity in cloud and environment |
| HLS(N) | Stable heating |
| LCLB | Cloud bottom |
| RAINF | 24 hours rainfall for determining whether planetary boundary layer is unstable |
| ISTAB(N) | Instability index |
| ETAIL | Working array |

Common /SOLR/

| | |
|---|---|
| U1 | Wind velocity at ground |
| U2 | Wind velocity at anemometer level (Extrapolatd from lowest layer velocity). |
| T1 | Ground temperature |
| T2 | Surface temperature extrapolated from model's lowest layer |
| Q1 | Specific humidity corresponding to T1 |
| Q2 | Specific humidity corresponding to T2 |
| ZOMAX | Maximum roughness parameter. A specified quantity |

| | |
|---|---|
| XHTI | Topographic height |
| ALBX | Albedo at a point |
| RIB | Bulk Richardson number |
| Z1 | Height of level 1 in constant flux layer |
| Z2 | Height of level 2 in constant flux layer |

Common /RAIN/

| | |
|---|---|
| RAINCV(L,M) | Convective rainfall |
| RAINLS(L,M) | Large scale rainfall |
| RAINSV(L,M) | Supersaturation rainfall |
| SURFLX(L,M) | Surface moisture flux |
| TEMFLX(L,M) | Heat flux |

Common /PHY/

| | |
|---|---|
| RAD2(L,M,N) | Temperature change due to radiation |
| TTFLUX2(L,M,N) | Temperature change due to planetary boundary layer temperature flux including shallow convection |
| QQFLUX2(L,M,N) | Moisture change due to PBL moisture flux including effect of shallow convection |
| CVH2(L,M,N) | Heating due to convective rainfall |
| ALS2(L,M,N) | Heating due to supersaturation adjustment |
| CVQ2(L,M,N) | Moistening due to cumulus convection |
| UFL(L,M,N) | X momentum flux |
| VFL(L,M,N) | Y momentum flux |
| ALSSU(L,M,N) | Moisture change due to supersaturation adjustment |
| DADJ(L,M,N) | Temperature change due to dry convective adjustment |
| LCON2(L,M,N) | Instability index |
| HLSS2(L,M,N) | Heating due to large scale precipitation |

Other relevant variables and arrays

| | |
|---|---|
| GNEXT(L,M,N) | This array contains the product TP i.e. product of transform of eigen vector TINV and the model parameter P at the current time step |

Description of subroutine

A-B

Various parameters for the integration are defined. Initial, last, forecast and rainfall time steps are calculated and stored in KT1, KT2, KT3, KT4. Two more counters KT5 and KT6 are defined for the purpose of restarting the integration. Parameters for initialization (INITIA, NTIMES, ICOUNT, KTT2 and DTT) are also defined here. Time step for radiation is stored in KRAD. All time steps are printed.
If initialization is to be done time step DT is reassigned value = 300 and KT2 = 1.

B-C

Rainfall arrays and forcing arrays are initialized in DO loop 1420

C-D

Subroutines CONST, COFF and CALPHI are called to define various constants for integration and coefficients for smoothing. The constants defined in CONST are mainly used in radiation package. Subroutine CALPHI gives grid intervals. In DO loop 1421 some parameters are calculated to economize computations.

D-E

For time step KT1=1 the input fields are read from tape ID=91 as 46 records. Subroutine TBAR is called to compute area averaged temperatures of $\sigma$ layers. Routine LAPSE is called to calculate model parameter $\gamma^*$ which is area averaged lapse rate parameter. Area averaged temperatures at layer boundaries TBB are also calculated in this routine. Subroutine START is called to read specified field arrays albedo, sea surface temperature and terrain. For time step other than 1 routine RESTART is called to read the necessary fields for restart from file 99. These fields are printed. A few more parameters are computed in DO loop 1422 and 1423.

E-F

Subroutine EIGEN is called to calculate eigen valuess and eigen functions of the coefficient matrix of Helmholtz equation

$$\underline{P} + \frac{\Delta t^2}{4} \nabla^2 A^{-1} B \underline{P} = A^{-1} \underline{F} = E$$

As array UU is used in routine EIGEN, input tape is rewound and input fields are reread. Constants RADIA(=1/1800), DY2(=2DY), DY2I, DYI and RADI = 1/RAD are defined.

F-G

At initial time KT1=1, the vertical velocity field is set to zero at boundaries. The upper and lower values of relative humidity are set to 95% and 20% respectively.

G-H

In DO loop 81 surface temperature and humidity are determined. Over land these values are set to zero. Over ocean (albedo < 0.06) surface temperature is assigned the sea surface temperature value and corresponding specific humidity are calculated. These are stored in TSUB and QSUB arrays. In inner do loop 82 specific humidity is determined from input relative humidity field and stored in array QQ.
Subroutine check is called to calculate and print average quantities for monitoring the program.

H-I

In DO loop 1426 maximum albedo of the domain is determined and stored in ALBMAX. Main DO loop 11 starts. Hour for hour angle calculation that appears in the zenith angle computations, is calculated. DAY is also calculated here from time step KT1.

```
TIME = (KT1 - 1) * DT
DAY = IFIX(TIME/86400)
HOUR = IFIX((TIME - DAY*86400)/3600)
ZZHOUR = HOUR + DZHOUR
```

The hour angle is stored in ZZHOUR. If it exceeds 24, value 24 is subtracted from it to get correct hour. The time step, time, day and hour are printed.

I-J

For time step KT=1 routine GUESS is called to determine the model parameter $P = gz + RT^* \ln p_s$ also in do loop 1450 the array A1 obtained from routine GUESS is stored in array GNEXT and the working arrays B1, B2, B3 and B4 are initialized to zero.
For time step other than zero the working arrays B1, B2, B3, B4 which contain non linear tendencies of previous time step are read from tape KD(=93).

J-K

Routine YCAL is called for calculation of non linear tendencies at current time step including all physical forcings. All the non linear tendencies A1 to A4 are written on KD after rewinding it. The tendencies A1 to A4 and B1 to B4 are added by calling routine YZADD. For time step 1 the multiplying coefficients are CA = 0.5, CB = 0.0. For all other time steps the coefficients are CA = 1.0, CB = -0.5. Routine XCAL1 is called to compute linear tendencies at grid points and to add these to output of YZADD. These values are stored in A1 to A4.

K-L

The total tendencies are interpolated to origin of the parcel in routine XYZW. This routine is called for determining origin and then finding the tenden-

cies at origin. At this stage the total tendencies are moved into arrays B1 to B4 by calling routine MOVE. The tendency A1 is temporarily stored in array UU. In DO loop 1451 the surface topography is computed using relation

$$(P - RT^* \ln p_s)/g$$

and stored in array B(I,J,1). At this stage routine QCAL1 is called to calculate the forcing of the Helmholtz equation in P.

L-M

Statement 71. After this routine ELLIPT is called to obtain P at next time step by solving the Helmholtz equation by relaxation. Routine DIAGNOS is called to obtain values of prognostic variables U, V, Q and diagnostic variable W. Routine BSMOOTH is called for applying smoother to all the fields UU, VV, PP, QQ and WW.

M-N

Routine SUPADJ is called to determine points at which supersaturation is occuring. In do loop 15 boundaries values of WW are reinitialized to zero.
If time step KT=KT3 routine FORCST is called to write the forecast fields on output tape 97. At first time step and every radiation time step routine CHECK is called for printing average values of some fields. When KT=KT4 rainfall is written on output file 98.
Every radiation time step the parameters required for restarting of the forecast are written on tape 99. 46 records of forecast variables and other arrays TB, TBB, GAMAB, ALBD, TERR, TSUB, QSUB and GNEXT are written on tape 99.
Do loop 11 ends here.

N-O

If INITIA=999 routine INIT is called for initialization of the input variables.

O-P

Prognostic and diagnostic fields are written on input tape. Variables required for restart of integration are written on tape 99.

## 2.1 Subroutine: START

Abstract: In this routine the prescribed input fields albedo, sea surface temperature and surface topography are read from mass storage unit 95 at the beginning of integration.

Called by: Main program FSULAM

Subroutines called: None

Input via COMMON: None

Input via argument

| Variable | Meaning |
|---|---|
| L | Number of grid points in x direction |
| M | Number of grid points in y direction |

Output variables

| Variable | Meaning |
|---|---|
| A(L,M) | Surface albedo |
| B(L,M) | Sea surface temperature |
| C(L,M) | Smooth topography |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Unit 95 is rewound. Surface albedo, sea surface temperature and smooth topography arrays are read from unit 95 as separate 3 records. Control passes to calling program. |

## 2.2 Subroutine: RESTART

Abstract: In this routine the parameters required to restart the integration of the model after any forecast time are read

Called by: Main program FSULAM

Subroutines called: None

Input via COMMON: None

Input via argument list

| Variable | Meaning |
|---|---|
| KT5 | Counter for number of forecast time for which the records on the unit 97 are to be skipped |
| KT6 | Counter for number of rainfall output time for which the records on the unit 98 are to be skipped. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Unit 99 is rewound. Then from this unit the arrays TB, TBB, GAMAB, ALBD, TERR, TSUB, QSUB and GNEXT are read and the unit is rewound again. |
| B-C | Unit 97 is rewound. Number of records to be skipped for each time step are calculated and stored in KS<br>KS = 6N+1<br>If (KT5.NE.0) i.e. if already records of one or more forecast time are written on unit 97 then in DO loop 10, KT5KS records are skipped by dummy read of 97. |
| C-D | Rewind 98. If (KT6.NE.0) i.e. if already records of one or more rainfall time are written on unit 98 then in DO loop 20, KT65 records are skipped on unit 98 by dummy read<br>Control passes to calling program. |

3.1 Subroutine: TBAR

Abstract: This routine computes the area averaged temperature from the geopotenial heights at the beginning of intergration of the model (KT = 1). The computed values are stored in array TB at N $\sigma$ levels. The formula used is

$$\frac{\sum_{i=1}^{L} \sum_{j=1}^{M} -\frac{\sigma}{R} \; g \; \frac{(ZZ_{k+1} - ZZ_k)}{(\sigma_{k+1} - \sigma_k)} \; DX(J)}{\sum_{i=1}^{L} \sum_{j=1}^{M} DX(J)}$$

Subroutines called:

Input variables via common:

| | |
|---|---|
| DX(M) | E-W direction grid interval, as function of latitude |
| R | Gas constant |
| G | Gravity |
| SIGT(N) | N values of $\sigma$ levels |

Input variables via arguments list:

| Variables | Meaning |
|---|---|
| ZZ(L,M,NP1) | L*M*NP1 values of geopotential heights |

Output variables:

| | |
|---|---|
| TB(N) | N values of area average temperatures at $\sigma$ levels |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Variable AREA is initialized. DO loop 11 is set up in which the area average temperature TB(k) are initialized at $\sigma$ levels 1 to N. |
| B-C | DO loop 12 is set up to index J through 1 to M and I through 1 to L<br>AREA = AREA + DX(J)<br>Loop 12 is initialed for indexing k through 1 to N and the temperature is computed corresponding to point I, J at level K and added to TB(K), stored in TB(K)<br>DO loop 12 ends. |

C-D

In DO loop 13 the area average temperatures are computed at N $\sigma$ levels and stored in TB. These values are also written in a formatted way (11F10.3) on file 96. The control returns to calling program.

3.2 Subroutine: LAPSE

Abstract: In this routine area average temperatures TBB at NP1 $\sigma$ levels are obtained from area average temperatures $\tilde{T}B$ at N $\tilde{\sigma}$ levels by averaging and extrapolation. Then these are used to obtain the lapse rates GAMAB using the formula

$$\gamma^* = \frac{R}{\sigma}\left[\frac{kT^*}{\sigma} - \frac{\partial T^*}{\partial \sigma}\right]$$

where k = R/Cp
This lapse rate is defined for linear parts of the equation

Subroutines called: None

Input variables via common:

Input variables via argument list

| | |
|---|---|
| $\tilde{T}B(N)$ | Area average temperatures at N $\tilde{\sigma}$ levels |
| SIGT(N) | N values of $\tilde{\sigma}$ levels |
| SIG(NP1) | NP1 values of $\sigma$ levels |
| R | Gas constant |
| Cp | Specific heat of air at constant pressure |
| N | Number of $\tilde{\sigma}$ levels |
| NP1 | Number of $\sigma$ levels |
| NM1 | N - 1 |

Output variables

| Variable | Meaning |
|---|---|
| TBB(NP1) | Area averaged temperature at NP1 $\sigma$ levels |
| GAMAB(NP1) | NP1 values of linear lapse rate |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Temperature TBB at upper boundary is obtained using TBB(1) = 1.5 * $\tilde{T}B(1)$ - 0.5 * $\tilde{T}B(2)$ DO loop 12 is set up for indexing K through 2 to N and TBB at these levels are obtained by averaging TB values of adjacent levels. TBB at NP1 level is obtained using TBB(NP1) = 1.5 * $\tilde{T}B(N)$ - 0.5 * $\tilde{T}B(NM1)$ |

B-C

In DO loop 11 the linear lapse rate values are obtained and stored in array GAMAB(N). The values of TB, TBB and GAMAB are written in formatted way (11E12 5) on file 96. Control is returned to calling routine.

3.3 Subroutine: GUESS

Abstract: This routine computes at the beginning of integration the model parameter P, given by the relation:
$P = gz + RT \ln p_s$
The quantity $v^{-1}P$ is also computed in this routine where $v^{-1}$ is the inverse of eigen vector matrix v.

Subroutines called:

Input variables via common

| Variables | Meaning |
|---|---|
| TBB(NP1) | Area averaged temperature $T^*$ at NP1 $\sigma$ level |
| TRANS(NP1,NP1) | The inverse eigen vector matrix |

Input variables via argument list

| | |
|---|---|
| PP(L,M,N,NP1) | The geopotential heights at the LxMxNP1 grid points. |
| PL(L,M) | Log of surface pressure at LxM grid points. |

Output variables

| Variable | Meaning |
|---|---|
| PP(L,M,N) | The model parameter 'P' at LxMxNP1 points |
| PTRANS(L,M,NP1) | $\sum_{j=1}^{NP1} v^{-1}_{ij} \quad P_j = TP$ |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | DO loop 11 is set up to index K through 1 to NP1 vertical levels. J through 1 to M and I through 1 to L. The model parameter is computed and stored in PP. $P_k = gz_k + R\,T_k \ln p_s$. DO loop 11 ends. |
| B-C | DO loop 12 is set up to index J through 1 to M, I through 1 to L and within this do loop 13 is set up to index K through 1 to NP1 vertical levels. The variable TT(k) is initialized for one k level at a time and do loop 13 is set up for a counter KK to index through 1 to NP1. For every value of K, a sum of the product $v^{-1}$ PP(I,J,KK) is found over 1 to NP1 levels and stored in array TT(K). DO loop |

13 ends. DO loop 12 is set up for the variable K to index through 1 to NP1. The contents to array TT(K) are transferred to array PTRANS(I,J,K). DO loop 12 ends and control is transferred to calling program.

PAGES 107-109 MISSING

## 4.1 Subroutine: YCAL

Abstract: This is the main subroutine of the model. In this routine the non-linear tendencies of the momentum equations, thermodynamic energy equations, and moisture continuity equation are calculated. These tendencies also include effects of physical forcings. These forcing terms are calculated in various physics routine called by YCAL. In this routine various arrays and constants are printed to monitor the performance of the model during integration.

Called by: Main program FSULAM

Subroutines called: RADN, SFLX, CVHEAT, SUPADJ1, DRYCON1, LCLSUR, PBLFLX

Input via common

| | |
|---|---|
| UU(L,M,N) | U component of wind |
| VV(L,M,N) | V component of wind |
| PP(L,M,NP1) | Model parameter P |
| WW(L,M,NM1) | Vertical velocity ($\sigma$) at layer boundaries |
| QQ(L,M,N) | Specific humidity |
| DT | Time step |
| DX(M) | Grid distance in x-direction |
| DY | Grid distance in y-direction |
| DLAM | x-grid interval in terms of degrees longitude |
| DPHI | y-grid interval in terms of degrees latitude |
| PHI(M) | Latitudes |
| SIG(NP1) | $\sigma$ values at layer boundaries |
| SIGT(N) | $\sigma$ values in layers ($\tilde{\sigma}$ levels) |
| COR(M) | Coriolis parameter |
| RAD | Earth's radius |
| G | Gravity |
| R | Gas constant for dry air |
| CP | Specific heat at constant pressure |

| | |
|---|---|
| PI | 3.141592654 |
| TB(N) | Area averaged temperature in layers (at $\tilde{\sigma}$ levels) |
| TBB(NP1) | Area averaged temperature at layer boundaries (at $\sigma$ levels) |
| GAMAB(N) | Area averaged parameter $\gamma^*$ in layers (at $\tilde{\sigma}$ levels) |
| T(N) | Temperature of a column in $\sigma$ layers (at $\tilde{\sigma}$ levels) |
| TT(NP1) | Temperature of a column at $\sigma$ layer boundaries (at $\sigma$ levels) |
| GAMA(N) | Parameter $\gamma$ in $\sigma$ layers (at $\tilde{\sigma}$ levels) |
| T1000 | Surface temperature |
| Q1000 | Specific humidity at surface |
| RH(N) | Relative humidity at N $\sigma$ layers (at $\tilde{\sigma}$ levels) |
| TSUB(L,M) | Surface level temperature |
| QSUB(L,M) | Surface level specific humidity |
| U1 | Surface wind velocity. Level 1 of constant flux layer |
| U2 | Surface wind velocity. Level 2 of constant flux layer |
| T1 | Temperature at level 1 of constant flux layer |
| T2 | Temperature at level 2 of constant flux layer |
| Q1 | Specific humidity corresponding to T1 |
| Q2 | Specific humidity corresponding to T2 |
| XHT1 | Height of surface topography |
| PZ(NP1) | Pressure at $\sigma$ layer boundaries (at $\sigma$ levels) |
| PT(N) | Pressure in $\sigma$ layers (at $\tilde{\sigma}$ levels) |
| ALBX | Albedo at a point |
| ALBMAX | Maximum albedo over the domain |
| PL(I,J) | $\ln p_s$ where $p_s$ is surface pressure |
| HLC(N) | Heating in $\sigma$ layers of a column due to cumulus convection (at $\tilde{\sigma}$ levels) |

| | |
|---|---|
| QDIF(N) | Difference of humidity between cloud and environment in a column |
| HLS(N) | Heating due to large scale rainfall in a column |
| QFLX(N) | Moisture flux in a column |
| TFLX(N) | Temperature flux in a column |
| FSX | Sensible heat flux at surface |
| FLT | Latent heat flux at surface |
| FMX | x momentum flux at surface |
| FMY | y momentum flux at surface |
| UFLX(N) | x momentum flux in a column |
| VFLX(N) | y momentum flux in a column |
| HQSUP(N) | Moisture change due to supersaturation adjustment |
| HTSUP(N) | Temperature change due to supersaturation adjustment |
| DADJ1(N) | Heating due to dry convective adjustment |

Output

| | |
|---|---|
| A1(L,M,N) | Individual change of u. Non-linear tendency of u equation |
| A2(L,M,N) | Individual change of v |
| A3(L,M,N) | Individual change of P. Non-linear tendency of thermodynamic equation |
| A4(L,M,N) | Individual change of q |
| RAINCV(L,M) | Convective rainfall |
| RAINLS(L,M) | Large scale rainfall |
| RAINSU(L,M) | Supersaturation rainfall |
| SURFLX(L,M) | Surface latent heat flux |
| TEMFLX(L,M) | Surface sensible heat flux |
| RAD2(L,M,N) | Temperature change due to radiative effects |
| TTFLUX2(L,M,N) | Temperature (sensible heat) flux |

| | |
|---|---|
| QQFLUX2(L,M,N) | Moisture (latent heat) flux |
| CVH2(L,M,N) | Heating (temperature change) due to convection |
| ALS2(L,M,N) | Heating (temperature change) due to moist asjustment |
| CVQ2(L,M,N) | Moisture change due to convection |
| UFL(L,M,N) | x momentum flux |
| VFL(L,M,N) | v momentum flux |
| ALSSU(L,M,N) | Moisture change due to moist adjustment |
| DADJ(L,M,N) | Temperature change (heating) due to dry convective adjustment |
| LCON2(L,M,N) | Instability index |
| HLSS2(L,M,N) | Heating due to large scale rainfall i.e. stable heating |
| TSEC(13) | Time taken by each of the physics routines |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Upto statement 1301, maximum u and v components of lowest $\sigma$ layer are determined and stored in UMAX1 and VMAX1. Corresponding grid points indices are also stored. These values are printed on output file |
| B-C | Major DO loop 1151 is initiated for computation of forcings due to various physical processes. The index J runs from 1 to M and I runs from 1 to L. Conditions at boundaries are set up. |
| C-D | DO loop 1938 is initiated for calculation of vertical advection operator $\dot{\sigma}\frac{\partial}{\partial\sigma}$. The index k runs from 1 to N for N $\sigma$ layers. Here in uppermost layer if $\dot{\sigma} > 0$ i.e. downward advection $\dot{\sigma}\frac{\partial}{\partial\sigma}$ is forced to zero value. Similarly at lower boundary layer if $\dot{\sigma} < 0$ i.e. upward motion $\dot{\sigma}\frac{\partial}{\partial\sigma}$ is forced to zero. For general levels, at upper layer boundary $\dot{\sigma}\frac{\partial}{\partial\sigma}\text{upper} = \frac{0.5\ \dot{\sigma}_{k-1}}{\sigma_k - \sigma_{k-1}}$ |

Lower boundary of layer

$$\dot{\sigma}\frac{\partial}{\partial\sigma}\text{lower} = \frac{0.5\ \dot{\sigma}_k}{\tilde{\sigma}_{k+1} - \tilde{\sigma}_k}$$

At upper boundary

$$\dot{\sigma}\frac{\partial}{\partial\sigma} = \dot{\sigma}_1/(\tilde{\sigma}_2-\tilde{\sigma}_1)$$

At lower boundary

$$\dot{\sigma}\frac{\partial}{\partial\sigma} = \dot{\sigma}_{NM1}/(\tilde{\sigma}_N-\tilde{\sigma}_{NM1})$$

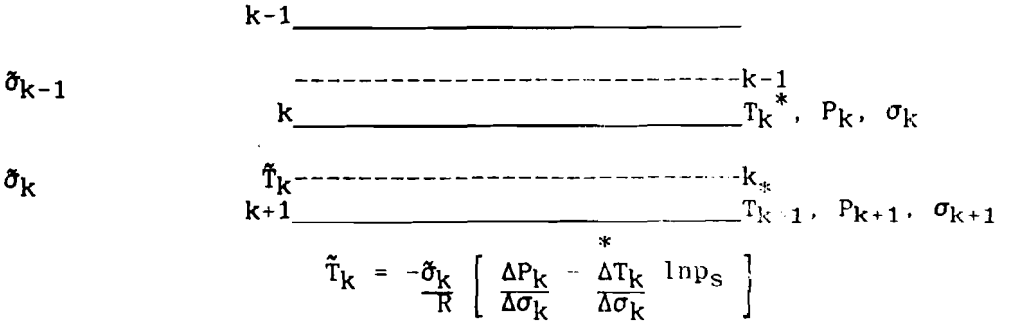
A factor 0.5 appears in all expressions to take care of vertical averaging later on. These operators are stored for a column in WD11 and WD21 respectively. Do loop 1938 closes. All computations of physics are done columnwise.

D-E

DO loop 12 to DO loop 52. In DO loop 12 temperatures in $\sigma$-layers are determined using hydrostatic relation.

$$\tilde{T}_k = -\frac{\tilde{\sigma}_k}{R}\ (P_{k+1} - P_k) - R(T^*_{k+1} - T^*_k)$$

$$\cdot\ \frac{\ln p_s}{(\sigma_{k+1} - \sigma_k)}$$




$$\tilde{T}_k = -\frac{\tilde{\sigma}_k}{R}\left[\frac{\Delta P_k}{\Delta\sigma_k} - \frac{\Delta T^*_k}{\Delta\sigma_k}\ \ln p_s\right]$$

Temperature at layer boundaries are obtained by averaging (DO loop 13). At top and bottom levels, by linear extrapolation of layer temperatures

$$T_1 = 1.5 \cdot \tilde{T}_1 - 0.5 \cdot \tilde{T}_2$$

$$T_{NP1} = 1.5 \cdot \tilde{T}_N - 0.5 \cdot \tilde{T}_{N-1}$$

$$T_k = 0.5 * \{\tilde{T}_k + \tilde{T}_{k-1}\}$$

In DO loop 13 pressure in $\tilde{\sigma}$ layers $PT_k$ are obtained by reversing the order of index k. Also temperatures at layer boundaries are obtained $PZ_k$

$$PT_k = \tilde{\sigma}_{kk}\, p_s$$

$$kk = N - k+1;\ k=1(1)N$$

$$PZ_{k+1} = \sigma_{kk}\, p_s$$

In DO loop 53 ΔPZ are obtained and in DO loop 51 ΔPT are obtained

$$\Delta PZ_k = PZ_k - PZ_{k+1}$$
$$\Delta PT_k = PT_{k-1} - PT_k$$

At lower boundary

$$\Delta PT_1 = PZ_1 - PT_1$$

In DO loop 52 inverse of ΔPT and ΔPZ are obtained and stored in DPTIV and DPZIV respectively.

E-F

Statement 52 to 422. Parameters of constant flux layer are obtained which are later used in boundary layer parameterization.
Albedo of the point for given I,J is stored in ALBX. We difine 2 levels in the constant flux layer. Level 1 is lower (surface) and level 2 is upper (air) but is much lower than lowest $\sigma$ layer value.

$U1 = 0$ Surface velocity

$$U2 = (\tilde{U}_N{}^2 + \tilde{V}_N{}^2)^{1/2}$$

Lower limit of U2 is set to 0.1.
Air temperature $T_2$ is extrapolated from temperature of lowest $\sigma$ layer.

$$T_2 = \frac{\tilde{T}_N}{(\tilde{\sigma}_N)^{R/c_P}}$$

Pressure of lower boundary $\sigma$ level is stored in PTX and of lowest $\tilde{\sigma}$ layer in PTXN. Temperature of lowest $\sigma$ layer is stored in TTXN. Corresponding specific humidities are determined and stored in QSX and QSXN, respectively.
Relative humidity is determined.

$$RHX = QQ_N/QSNX$$

Lower limit of relative humidity is set = 0.0 and upper limit = 1.0.
Specific humidity at level 2 is determined

$$Q_2 = RHX * QSX$$

The surface specific humidity Q1000 is assigned value $Q_2$ under constant flux assumption.
Maximum roughness length is defined as ZOMAX = 4000 * 0.01 metres.
Calculation of roughness length $Z_0$
Over land areas surface topography of the point I,J is stored in XHT1, density of air is determined and stored in RHO. Ground wetness parameter GW is defined by empirical relation.

$$GW = \frac{(ALBMAX - ALBX)}{(ALBMAX - 0.05)} \cdot (1.0 - RHX)$$

Over oceans (ALBX $\leqslant$ 0.06)
$Z_2 = 10$
CDO = 0.0011 defines drag coefficients. It is made a function of wind speed.
If U2 < 5.8 $ms^{-1}$ CDD = CDO
5.0 < U2 $\leqslant$ 16.8; CDD = CDO·(0.74+0.046·U2)
16.8 < U2 CDD = CDO·(0.94+0.034·U2)
Friction velocity first guess is calculated as

$$USTSQ = CDD \cdot U2^2$$

Then the roughness length $Z_0$ is calculated using Charnock's formula

$$Z_0 = 0.04 \cdot USTSQ/g$$

$$Z_1 = Z0$$

Lower limit of Z1 is set as $1.0 \times 10^{-4}$m.
If $Z_1 > Z_2$ then Z2 = Z2 + Z1
Over land (Statement 413)

$$Z_2 = 100 \text{ m and}$$

$$Z_0 = 0.15 + 0.2 \cdot (236.8 + 18.42 \cdot XHT1)^2 \cdot 1.0 \times 10^{-8} \text{ metres}$$

Upper limit of $Z_0$ is kept as = 1.0

$$Z_1 = Z_0$$

If $Z_1 \geqslant$ ZOMAX A1 = ZOMAX

F-G 422 continue. If time step is integral multiple of KRAD, subroutine 'RADN' is called for calculation of temperature change due to radiation. The radiative temperature is stored in array RAD2. The order of the index K is reversd to store from lowest to uppermost layer.

G-H Statement 62. The surface temperature and specific humidity are stored in variables T1 and Q1. T2, Q2, Z1, Z2 are known. Routine SFLX is called to calculated surface fluxes. The x and y surface momentum fluxes are stored in FMX and FMY respectively.

H-I WBAR = 0. Upto statement 230 and two statements after that vertically averaged $\sigma$ is determined. Relative vorticity at layer interface of N-2 and N-3 is determined.

$$\zeta = \frac{(V_{i+1,j,N-3} - V_{i-1,j,N-3}) + (V_{i+1,j,N-2} - V_{i-1,j,N-2})}{DXX}$$

$$- \frac{(V_{i+1,j,N-3} - U_{i-1,j,N-3}) + (U_{i+1,j,N-2} - V_{i-1,j,N-2})}{DYY}$$

$$+ (U_{i,j,N-3} + U_{i,j,N-2}) \cdot TAN\left(\frac{PHI(J)*PI/180.0}{RAD}\right)$$

$$\cdot \frac{COR(J)}{ABS\ (COR(J) + 1.0E - 10)}$$

$$\zeta = \frac{\partial \bar{V}^{\sigma}}{\partial x} - \frac{\partial \bar{u}^{\sigma}}{\partial y} + \bar{U}^{\sigma} \frac{\tan\phi_j}{a} f$$

Subroutine CVHEAT is called to calculate convective heating. Instability index LCONV are obtained and stored in LCONV1 with order reversed. Routine SUPADJ1 is called for moist adjustment and the routine DRYCON1 for dry convective adjustment. In DO loop 233 the heating due to convection, large scale rainfall, moisture difference between cloud and environment, moisture change and temperature change due to supersaturation removal are stored in arrays HLC1, HLS2, QDIF1, HQSUP1, HTSUP1 with order of vertical levels reversed.

I-J

Routine LCLSUR is called to get LCL to be used later in planetary boundary layer parmaeterization and shallow convection. In this routine LCL level (LCLN) is obtained. Routine PBLFLX is called for PBL flux distribution in the layers. In DO loop 127 the moisture flux, U-flux, V-flux and temperature flux are stored in column vectors QFLX1, UFLX1, VFLX1 and TFLX1 with order reversed

$$kk = N - k+1 \quad k= 1(1)N$$

J-K

To get the total effects of heating and moistening in a column due to convection, stable heating and supersaturation, variables SUM, SUM2 and SUM1 are initialized. In DO loop 118 the pressure weighted quantities are added in vertical.

$$SUM = SUM + HLC(k)\ \left(\frac{\Delta PZ}{G}\right) \cdot 100 \text{ etc.}$$

These sums are added to arrays RAINCV, RAINLS and RAINSU to get convective rainfall, large scale rainfall and supersaturation rainfall contribution of all columns in one time step.

$$RAINCV(I,J) = RAINCV(I,J) + \frac{C_P}{L} SUM \cdot DT$$

$$RAINLS(I,J) = RAINLS(I,J) + \frac{C_P}{L} SUM2 \cdot DT$$

$$RAINSU(I,J) = RANSU(I,J) - SUM1 \cdot DT$$

(SUM1 is computed from moisture change)
Surface flux of temperature and moisture are also obtained.

$$TEMFLX(I,J) = TEMFLX(I,J) + FSX \cdot DT$$
$$SURFLX(I,J) = SURFLX(I,J) + FLT \cdot DT/L$$

K-L

For points that are monitored the following variables and arrays are printed. Points monitored are sea point, desert point and high ground point.

(i) Title HRAD, HLCO, HLS, TFLX, QFLX, COOLKUO, UFLX, VFLX, LCONV
(ii) KT: Time step
(iii) Title from YCAL
(iv) Sea point W desert point of high ground
(v) Surface variables (for $\sigma_N$ layer) T(N), QQ(I,J,N), FMX, FMY, UU(I,J,N), VV(I,J,N)
Under the title
TAIR, QAIR, FMX, FMY, UU(I,J,N), VV(I,J,N)
(vi) Arrays
HRAD1, HLC1, HTSUP1, TFLX1, QFLX1, QDIF1, UFLX1, VFLX1, DADJ1, LCONV1
(N rows of values are printed)
(vii) In DO loop 1217 for the lowest 2 $\sigma$ layers and $\sigma$ layer boundaries T, TT, PT, PZ, GAMA and TFLX1 are printed.
Printing stops at statement 181

L-M

DO loop 1019. The values of various physical forcings in a column are stored in 3 dimensional arrays. For radiation time step for the point of maximum surface u and v velocity repectively
FSX, FLT, FMX, FMY, UU(I,J,N), VV(I,J,N), R1, T2, Q1, Q2, U1, U2, QQ(I,J,N), TFLX1(N), QFLX1(N), UFLX1(N), VFLX1(N), LCLN and PZ(1), ALBX are printed

M-N

1305 continue. For points that are monitored and for radiation time step the following variables and arrays are printed. For N $\sigma$ layers
UU, VV, QQ, WW, T, PZ, TT, PT, PL(I,J), HRAD1, TFLX1, QFLX1, UFLX1, VFLX!, HTSUP1, HQSUR1, HLS2, GAMA, QDIF1,HLC1, ETAIL, LCONV1, FSX, FLT, FMX, FMY, T1, T2, Q1, Q2, U1, U2, TSUB(I,J), QSUB(I,J), ALBD(I,J), CHS, CQS, CDS, RIB, RHO, GW
Physics computations are complete. DO loop 1151 ends.

N-O

Routines BSM1 is called for smoothing of temperature and moisture distribution

O-P

Another major DO loop 11 is initiated for calculation of non-linear tendencies of momentum, thermodynamic and moisture conservation equations. The index JJSM runs from 1 to M and index JJSM runs from 1 to M and index IISM from 1 to L. Conditions at boundaries are specified

IP1 = I+1; at I=L, IP1 = I
IM1 = I-1; at I=1, IM1 = I
JP1 = J+1; at J=M, JP1 = J
JM1 = J-1; at J=1, JM1 = 1
DYY = 2 * DY; at J=1 or J=M DYY = DY
DXX = 2 * DX(J); at I=1 or I=L DXX = DX(J)

P-Q In DO loop 1212 temperatures in N $\sigma$ layers are calculated using hydrostatic equation

$$\tilde{T}_k = -\frac{\tilde{\sigma}_k}{R} \cdot \left[\left[\frac{\partial P}{\partial \sigma}\right]_k - R\left[\frac{\partial T^*}{\partial \sigma}\right]_k \ln p_s\right]$$

From these temperatures temperatures at layer boundaries are obtained by averaging

$$TT_k = .5 \cdot (\tilde{T}_k + \tilde{T}_{k-1})$$

At upper and lower boundaries temperatures are obtained by extrapolation.
$TT_1 = 1.5 \cdot \tilde{T}_1 - 0.5 \cdot \tilde{T}_2$
$TT_{NP1} = 1.5 \cdot \tilde{T}_N - 0.5 \cdot \tilde{T}_{NM1}$
In DO loop 14 the lapse rate parameter $\gamma$ is calculated for N $\sigma$ layers.

$$\tilde{\gamma}_k = \frac{R}{\tilde{\sigma}_k}\left[\frac{R}{C_P}\frac{\tilde{T}_k}{\tilde{\sigma}_k} - \frac{\Delta T_k}{\Delta \tilde{\sigma}_k}\right]$$

Q-R In DO loop 49 for N $\tilde{\sigma}$ layers the total physical forcing is determined by adding contributions from radiation, convection, temperature flux, supersatureation adjustment, large scale heating, and dry convective adjustments. These are stored in DTDT(k). The values of these contributions in individual columns for a point I,J ae stored in one-dimensional arrays. DO loop 49 closes.

R-S DO loop 11 for index k is initiated. The index k runs from 1 to N. The vertical advection operators are calculated as described under C-D.
At boundaries the divergence DIV is assigned value 0. AT other points divergence DIV is calculated using the formula

$$DIV_k = \frac{U_{i+1,j,k} - U_{i-1,j,k}}{2\Delta x} + \frac{V_{i,j+1,k} - V_{i,j,k}}{2\Delta y} - V_{i,j,k}\frac{\tan\phi_j}{a} + \left[\frac{\partial \dot{\sigma}}{\partial \sigma}\right]_k$$

Non linear tendencies for u equation are calculated and stored in A1

$$A1_{i,j,k} = \Delta t \left[ -\sigma\frac{\partial}{\partial\sigma} \cdot [U_{i,j,k} - U_{i,j,k-1}] - \sigma\frac{\partial}{\partial\sigma} \cdot [U_{i,j,k+1} - U_{i,j,k}] + U_{i,j,k} \cdot \left[ f_j + \frac{U_{i,j,k} \tan\phi_j}{a} \right] - R \cdot 0.5 \cdot \overline{(T - T^*)}^{\sigma}_k \cdot \frac{(\ln p_s)_{i+1,j} - (\ln p_s)_{i-1,j}}{2\Delta x_j} + UFLX1(k) \right]$$

A similar expression is used for calculating non-linear tendency of v equation and the tendency is stored in A2

The non-linear tendency for thermodynamic equation is calculated using the expression

$$DT * \frac{\left[ - (\sigma)_k (\gamma_k - \gamma_k^*) + \frac{R^2}{C_P \tilde{\sigma}_k} \cdot (\tilde{T}_k - TB_k) \cdot (\nabla \cdot \vec{V})_k - \frac{R}{\tilde{\sigma}_k} \frac{(H + D_\theta)}{C_P} \right]}{\gamma^*}$$

where $\frac{(H + D_\theta)}{C_P}$ is represented by variable $DTDT_k$

DO loop 11 ends. Control passes to calling program FSULAM.

4.2 Subroutine: YZADD

Abstract: This routine computes the underlined expression of the following equation

$$W_j^{+} = X_j^{o} + \underline{Y_j + \bar{Z}_j}$$

by adding the $Y_j$ (current step) and $\bar{Z}_j$ (previous step). $Y_j$'s and $Z_j$'s are brought through argument list.
As Z's are expressed as

$$Z_i = \frac{-\Delta t}{2} s_i = \frac{-Y_i}{2}$$

two weighting coefficients CA and CB are brought into the routine through argument list. For time step 1 $Z_i$'s are not available, so CB = 0 and therefore CA = 0.5 otherwise CA = 1.0 and CB = 0.5. The routine carries out the summation and returns values in array B for every point along a latitude.

Subroutines called: None

Input via common: None

Output variables: $B_j(L,M,N)$: values of $Y_j + Z_j^{-1}$ at L M N points.

Description of subroutine

| Designator | Text |
|---|---|
| A | Loop 11 is set up to index I from 1 to L. The summation of arrays A and B is carried out and sum is stored in array B.<br>$B_i = CA * A_i + CB * B_i$<br>Control passes to calling program FSULAM |

4.3 Subroutine: XCAL1

Abstract: This routine is called in the routine FSULAM and it computes the X's of the following semi-implicit, semi-Lagrangian scheme:

$$W_j^+ = X_j^\circ + Y_j + Z_j^-$$

These represent the values of the forecast variables at the origin and are given as

$$X_1 = u - \frac{\Delta t}{2} \frac{\partial \bar{P}^\sigma}{\partial x} + \frac{s_1}{2}$$

$$X_2 = v - \frac{\Delta t}{2} \frac{\partial \bar{P}^\sigma}{\partial y} + \frac{s_2}{2}$$

$$X_3 = \frac{1}{\gamma^*} \frac{\partial \bar{P}^\sigma}{\partial \sigma} + \dot{\sigma} \ln p_s - \frac{\Delta t}{2} \bar{\delta}^\sigma + \frac{s_3}{2}$$

$$X_4 = q + \frac{s_4}{2}$$

$$X_5 = \ln p_s - \frac{\Delta t}{2} \nabla \cdot \vec{V} + \frac{\partial \dot{\sigma}}{\partial \sigma} \text{ ; At horizontal boundaries } X_5 = \ln p_s$$

where

$$P = gz + RT^* \ln p_s$$

$$\tilde{\gamma}^* = \frac{R}{\tilde{\sigma}} \left[ \frac{k\tilde{T}^*}{\sigma} - \frac{\partial T^*}{\partial \sigma} \right]$$

The superscript $\circ$ means that

$$X_j^\circ \rightarrow X(X_j - \alpha,\ Y_j - \beta,\ t)$$

In this routine the values of X's are computed at the grid points. In routine XYZW, these are interpolated at origin of the parcel. The values of the computed are stored in arrays A.

Subroutines called: None

Input variables via common

| Variable | Meaning |
|---|---|
| UU(L,M,N) | U component of wind at N $\tilde{\sigma}$ levels |
| VV(L,M,N) | V component of wind at N $\tilde{\sigma}$ levels |
| PP(L,M,NP1) | The parameter P defined as $gz + RT^* \ln p_s$, at NP1 $\sigma$ levels |

| | |
|---|---|
| WW(L,M,NM1) | The vertical velocity $\sigma$ at NM1 $\sigma$ levels |
| QQ(L,M,N) | The specific humidity at N $\tilde{\sigma}$ levels |
| A1(L,M,N) | $S_1$ |
| A2(L,M,N) | $S_2$ |
| A3(L,M,N) | $S_3$ |
| A4(L,M,N) | $S_4$ |
| A5(L,M,N) | Working array |
| DT | Time step (= 1800 sec) |
| DX(M) | The grid interval in E-W direction<br>$DX(J) = d\lambda\cos\phi_J$ |
| DY | Grid interval in N-S direction |
| SIG(NP1) | NP1 values of $\sigma$ levels |
| PHI(M) | M values of latitude |
| SIGT(N) | N values of $\tilde{\sigma}$ levels |
| COR(m) | M values of coriolis parameter |
| GAMAB(N) | N values of parameter $\gamma^*$ at N $\tilde{\sigma}$ levels |
| RAD | Radius of earth |
| PI | Value of $\pi$ |

Output variables

| Variable | Meaning |
|---|---|
| A1(L,M,N) | Value of X1 at grid points. X1 is described in abstract. |
| A2(L,M,N) | Value of X2 at grid points. |
| A3(L,M,N) | Value of X3 at grid points. |
| A4(L,M,N) | Value of X4 at grid points. |
| A5(L,M,N) | Value of X5 at grid points. |

Description of subroutine

| Designator | Text |
|---|---|

A-B DO loop 11 is set up to index J from 1 to M. For computation of finite differences JP1 = J+1 and JM1 = J-1 are defined with JP1 = M at northern and JM1 = 1 at southern boundaries. DYY = 2.0 * DY is defined with DYY = DY at N and S boundaries.

B-C DO loop 11 is set up to index I from 1 to L. IP1 = I+1 and IM1 = I-1 are defined with IP1 = L and IM1 = 1 at E and W boundaries.

C-D DO loop 11 is set up for k to index through vertical levels 1 to N. For levels between 2 and NM1 the value of $\sigma_k$ is stored in W1 and $\sigma_{k-1}$ in W2, for k=1 0 in $W_1$ and $\sigma_1$ in $W_2$ for level k=N $\sigma_{NM1}$ in W1 and 0 in W2.

D-E X1 to X5 are computed according to formulae given in abstract. As the values of X's are obtained at $\dot{\sigma}$ levels P values are averaged in the vertical for obtaining derivatives like $\partial P/\partial x$ etc. Divergence $(\nabla \cdot V)$ for $X_5$ is obtained accourding to expression

$$\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} - \frac{v \tan\phi}{a}$$

At boundary points $\nabla \cdot \vec{V} + \frac{\partial \dot{\sigma}}{\partial \sigma} = 0$ and $X_5 = PL(I,J)$

4.4 Subroutine: XYZW

Abstract: In this routine, to obtain $\overset{+}{W}_j$ of the equation

$$\overset{+}{W}_j = \overset{o}{X}_j + Y_j + \bar{Z}_j$$

the values of $X_j$'s are computed (interpolated) from their values at grid point to the point of origin of the parcel and added to the array B's at grid point which contain $Y_j + \bar{Z}_j$. To carry out the interpolation functions XYINT4 and XYINT are called. It should be noted that the point of origin need not be very close to the grid point considered but can be more than several grid intervals away. Also, the distances are obtained in terms of grid intervals and once the location of origin is obtained, the surrounding 25 points stencil is considered for interpolation of X's to this point.

Subroutines called: Function XYINT4 and Function XYINT

Input variables via common

| Variable | Meaning |
|---|---|
| $A_i$(L,M,N) | The values of X's of equation given above at the grid points. |
| $B_i$(L,M,N) | Array in which the values $y_j + \bar{Z}_j$ are stored for j = 1,4. |
| VV(L,M,N) | Array in which the values $Y_j + \bar{Z}_j$ are stored for j = 5. |
| DX(J) | The grid interval in E-W direction in terms of distance. |
| DY | Grid interval in N-S direction in terms of distance. |
| DLAM | E-W grid interval in terms of degrees longitude. |
| DPMI | N-S grid interval in terms of degrees latitude. |

Output variables

Description of routine

| Designator | Text |
|---|---|
| A-B | DO loop 11 is set up to index K from 1 to N, J from 1 to M and I from 1 to L. Rough location of the origin is obtained using value's of $X_1$, $X_2$ (which have at this stage dimension of velocity) and |

stored in XDLAM and YDPHI respectively. These values represent distances in terms of degrees longitudes and latitudes. The nearest grid point coordinates to the origin are determined and stored in I1 and J1. Distance of the origin from the grid point I1, J1 is determined in terms of fraction of grid interval DLAM and DPHI and stored in XA and YB.

B-C The lower limits (I1=L and J1=1) and upper limits (I1=L and J1=M) are assigned. The values of X's (here A's) are interpolated to the point at distance XA, YB from I1, J1. From these X's at the origin the distance traveled by the parcel in time $\Delta t$ is determined in terms of degrees longitude and latitudes. For points, I=3 to L-2, J=3 to M-2, XYINT4 is called for interpolation while for inner boundary points (.....) XYINT is called and for boundary

```
 ------------------
|  X X X X X X X  |
|  X X X X X X X  |
|  X X X X X X X  |
 ------------------
```

points (--) the values of X's at the boundary grid points are used without interpolation. These distances are stored in XDLAM and XDPHI.

C-D The nearest grid point location is determined and stored in I1, J1 and distance of (new) exact location of origin is determined from I1, J1 in terms of degrees longitude and latitudes and limits are set for I1 as 1 and L and for J1 as 1 and M.

D-E Values of X's are interpolated to the exact origin using the nearest 25 points stencil and 4th order (or second order) interpolation formula as in B-C and these values are added to array B (representing $Y + Z^-$ at this stage) and the results are stored in B1 and B4 and UU. Arrays B1 to B4 and array UU at this stage represents W's. Control passes to calling program.

4.5 Subroutine: QCAL1

Abstract: This subroutine computes the forcing function (Q) of the Helmholtz's equation in P

$$\frac{\Delta t^2}{4} \nabla^2 P + \frac{\partial}{\partial \sigma} \frac{1}{\gamma^*} \frac{\partial P}{\partial \sigma} = Q$$

where $Q \equiv \frac{\Delta t}{2} \nabla \cdot \vec{W} + \frac{\partial \tilde{w}_3}{\partial \sigma} - \tilde{W}_5$

$$\vec{W} = i\lambda \hat{\tilde{W}}_1 + i\phi \hat{\tilde{W}}_2$$

W's are obtained according to equation

$$\overset{+}{W}_j = \overset{o}{X}_j + Y_j + \bar{Z}_j$$

in routine XYZW.
In this computation the divergence at the boundary points is taken as zero and only the last two terms are computed. The results are stored in array PP

Subroutine called: None

Input variables via common

| Variable | Meaning |
|---|---|
| $A_j(L,M,N)$ | $\overset{+}{W}_j = \overset{o}{X}_j + Y_j + \bar{Z}_j$; values of W's are stored in this array. |
| DX(M) | Grid distance in E-W direction |
| G | Gravity |
| R | Gas constant |
| B1(L,M,N) | Surface topography as determined by model at the current time step. |
| SIGT(N) | N values of $\tilde{\sigma}$ levels |
| TBB(NP1) | NP1 values of average temperatures at $\sigma$ levels |
| RAD | Earth's radius |
| PHI(M) | Latitude values |
| PI | Value of $\pi$ |
| DY | Grid distance in N-S direction |
| DT | Time step $\Delta t$. |

Output variables:

Description of routine

| Designator | Text |
|---|---|
| A-B | DYY = 2.0 DY<br>DO loop 11 is set up to index J through 1 to M; JP1 = J+1, JM1 = J-1, DXX = 2.0 * DX(J). DO loop 11 is also set up for indexing I through 1 to L with IP1 = I+1 and IM1 = I-1. The values of forcing functions are determined at all horizontal boundary points. At boundary points divergence and $\sigma$=0 therefore the calculation reduces to |

$$\frac{P_{N+1} - P_N}{\gamma_N * \Delta\sigma_N} + \frac{P_{N+1}}{RT_s *} = W_{3N+1} + \frac{gZ_s}{RT_s *}$$

At upper boundary

$$PP(I,J,1) = \frac{\tilde{W}_{31}}{\Delta\sigma_1} - \frac{\tilde{W}_{51}}{2} + \frac{\sigma_1 \cdot g \cdot Z_s}{RT_s *} \cdot \frac{1}{\Delta\sigma_1}$$

At lower boundary

$$PP(I,J,NP1) = \frac{-\tilde{W}_{3N}}{\Delta\sigma_N} - \frac{\tilde{W}_{5N}}{2} - \frac{gZ_s}{RT_s *} \cdot \frac{1}{\Delta\sigma_N}$$

And from level 2 to N

$$= \frac{\partial\tilde{W}_3}{\partial\sigma_k} - 0.5\,\bar{\tilde{W}}_5\dot{\sigma}$$

B-C

For points other than boundaries the forcing function is determined using

$$\frac{\Delta t}{2}\,\nabla\cdot\tilde{W}_k + \frac{\partial\tilde{W}_{3k}}{\partial\sigma_n} - \tilde{W}_5\dot{\sigma}$$

$$PP1 = 0.5\,\Delta t\,\frac{\partial\tilde{W}_1}{\partial x} + \frac{\partial\tilde{W}_2}{\partial y}\Big|_1$$

$$- \tilde{W}_2\tan\phi + \sigma_1\,\frac{g\,Z_s}{RT_s *}\,\frac{1}{\Delta\sigma_1}$$

$$PP_{NP1} = 0.5\,\Delta t\,\frac{\partial\tilde{W}_1}{\partial x} + \frac{\partial\tilde{W}_2}{\partial y}\Big|_N$$

$$- (\tilde{W}_2)_N\tan\phi - \frac{(\tilde{W}_3)_N}{\Delta\sigma_N} - 0.5(\tilde{W}_5)_N$$

$$- \frac{gZ_s}{RT_s *}\,\frac{1}{\Delta\sigma_N}$$

$$PP_k = 0.5 \cdot \Delta t \cdot \frac{\partial \tilde{W}_1 \tilde{\sigma}}{\partial x} + \frac{\partial \tilde{W}_{2\tilde{\sigma}}}{\partial y}$$

$$- \tilde{W}_2 \tan\phi + \frac{\partial W_3}{\Delta \tilde{\sigma}_k} - 0.5 \cdot \tilde{W}_{5\tilde{\sigma}}$$

The DO loop 11 ends. The control is transferred to calling routine.

## 4.6 Subroutine: DIAGNOS

Abstract: In this routine the values of forecast variables at time step n+1 are obtained using the following statements.

$$u^{+} = w_1^{+} - \frac{\Delta t}{2} \overline{\frac{\partial P^{+}}{\partial x}}^{\sigma}$$

$$v^{+} = w_2^{+} - \frac{\Delta t}{2} \overline{\frac{\partial P^{+}}{\partial y}}^{\sigma}$$

$$\ln p_s^{+} = \frac{1}{RT_{s*}} (P^{+} - gz_s)$$

$$\dot{\sigma}^{+} = \frac{2}{\Delta t} \left[ w_3^{+} - \frac{1}{\gamma^*} \frac{\partial P}{\partial \sigma} - \sigma \ln p_s \right]$$

$$q^{+} = w_4^{+}$$

$$T^{+} = - \frac{\sigma g}{R} \frac{\partial z^{+}}{\partial \sigma}$$

where

$$z^{+} = \frac{1}{g} (P^{+} - RT^{*} \ln p_s^{+})$$

This routine is called after obtaining the values of $P^{+}$ in routine 'ELLIPT'. The vertical velocity values'$\sigma$' at boundary are initialized as zero.

Subroutines called:

Input variables via common:

| | |
|---|---|
| A(L,M,N) | 5 arrays containing the values w's of the equation $\overset{+}{w}_j = \overset{o}{x}_j + y_j + \overset{-}{z}_j$ These represent forcing. |
| P(L,M,N) | Values of the model parameter P at time n+1 |
| DY | Grid interval in N-S direction |
| DX(m) | Grid interval in E-W direction as function of latitude |
| DT | Time step '$\Delta t$' |
| G | Gravity |
| B1(L,M,1) | Surface topography '$z_s$' as determined by model. |

| | |
|---|---|
| SIG(NP1) | NP1 values of '$\sigma$' |
| TBB(NP1) | NP1 values of area average temperature $T^*$ at $\sigma$ levels. |
| GAMAB(N) | N values of lapse rate GAMAB at $\sigma$ levels. |
| R | Gas constant |

Output variables

| Variable | Meaning |
|---|---|
| UU(L,M,N) | L•M•N values of u-component of wind at time n+1 |
| VV(L,M,N) | L•M•N values of humidity variable (specific humidity) at time n+1. |
| PL(L,M) | L•M values of $\log p_s$ at time n+1 |
| WW(L,M,NM1) | L•M•NM1 values of vertical velocity $\sigma$ at time n+1 |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The DO loop 11 is set up to index J through 1 to M and I through 1 to L. The values JP1 = J+1, JM1 = J-1, DYY = 2.0Dy, IP1 = I+1, IM1 = I-1, DXX = 2.0DX(J) are assigned with values at boundaries as JP1 = M, JM1 = 1, IP1 = L, IM1 = 1, DYY = DY and DXX = DX(J). |
| B-C | The DO loop 12 is set up within loop 11 to index K through 1 to N vertical $\sigma$ levels. The values of u, and v component are computed using the relation $u = A_1 - \frac{1}{2}\Delta t \cdot \frac{\overline{PP}^{\sigma}}{\partial x}$ $v = A_2 - \frac{1}{2}\Delta t \cdot \frac{\partial \overline{PP}^{\sigma}}{\partial y}$ and specific humidity using $A = A_4$. The DO loop 12 ends. The log of surface pressure $p_s$ is computed using $PL_{i,j} = \frac{1}{R} \cdot \frac{1}{T_s^*} \{PP_{NP1} - gz_s\}$; B1(I,J,1) = $Z_s$(I,J) DO loop 11 ends. |
| C-D | DO loop 14 is set up to index J through 1 to M and I through 1 to L. The values JP1 = J+1, JM1 = J-1, DYY = 2.0DY, IP1 = I+1, IM1 = I-1, DXX = 2.0DX(J) |

are assigned with values at boundaries as JP1=M, JM1=1, IP1=L, IM1=1, DYY=DY and DXX=DX(J).

D-E In DO loop 15 within the DO loop 14 the vertical velocity $\sigma$ is initialized as zero at NM1 $\sigma$ levels at the boundary points. For points other than boundary points, first the velocity $\sigma$ is obtained at level 1 and then in DO loop 16 at level 2 to NM1 using formulae

$$\dot{\sigma}_1 = \Delta\tilde{\sigma}\ 2.0\frac{[W_5 - \ln P_s]}{\Delta t}$$

$$- \left[ \frac{\partial u}{\partial x} - \frac{\partial u}{\partial y} \right] - \frac{u\tan\phi_j}{a}$$

$$\dot{\sigma}_k = \dot{\sigma}_{k-1} - \Delta\tilde{\sigma}_k\ \ 2.0\ \frac{[W_{5_k} - \ln P_s]}{\Delta t}$$

$$- - \left[ \frac{\partial v}{\partial x} - \frac{\partial v}{\partial y} \right] - \frac{v\tan\phi_j}{a}$$

DO loop 16 ends.
Control passes to calling program.

4.7 Subroutine: FUNCTION XYINT4

Abstract: In this function routine the value of a given variable F is interpolated to a point (I,J) at given short distance using a 4th order interpolation formula given in terms of a polynomial, say

$$P_n(X) = a_0 + a_1(X-X_0) + \frac{a_2}{2}(X-X_0)^2 + \frac{a_3}{6}(X-X_0)^3 + \frac{a_4}{24}(X-X_0)^4 \quad (1)$$

and

$$P_n(Y) = a_0 + a_1(Y-Y_0) + \frac{a_2}{2}(Y-Y_0)^2 + \frac{a_3}{6}(Y-Y_0)^3 + \frac{a_4}{24}(Y-Y_0)^4 \quad (2)$$

For using this formula, the polynomial is required to be defined exactly at 5 points. On applying this condition, the following 5 coefficients are obtained.

$$a_0 = F_{i,j}(x,y,t) \quad (3)$$

$$a_1 = 1/12\ [-F_{i+2,j} + 8F_{i+1,j} - 8F_{i-1,j} + F_{i-2,j}] \quad (4)$$

$$a_2 = 1/24\ [-F_{i+2,j} + 16F_{i+1,j} - 30F_{i,j} + 16F_{i-1,j} - F_{i-2,j}] \quad (5)$$

$$a_3 = 1/12\ [F_{i+2,j} - 2F_{i+1,j} + 2F_{i+1,j} - F_{i-2,j}] \quad (6)$$

$$a_4 = 1/24\ [F_{i+2,j} - 4F_{i+1,j} + 6F_{i,j} - 4F_{i-1,j} + F_{i-2,j}] \quad (7)$$

This interpolation formula can equivalently be expressed as

$$= (I - G_{ij})\ (I - H_{ij})\ F_{ij} \quad (8)$$

where $(I - G_{ij})$ is interpolation operator in X direction and $(I - H_{ij})$ in the Y direction. In this routine first the interpolation is carried out in X-direction, then in Y-direction. For this a stencil of 25 points is set up.

Subroutines called: None

Input variables via common

| Variable | Meaning |
|---|---|
| DLAM | Grid interval in terms of degrees longitude, in east-west. |

DPHI . Grid interval in terms of degrees latitude, in north-south.

Output variables

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The fractional distances $\zeta$ and $\eta$ (i.e.,$(X-X_0)$ $(Y-Y_0)$) are determined and stored in RX and RY. The -ve sign allows the parcel to come from any side of the grid point. It also takes care of -ve signs of odd powered terms. |
| B-C | DO loop 11 is set up to index counter JJ from 1 to 5. This way a stencil of 5X5 is set up, aroung the grid point (I, J) which (from the calling routine) is in fact the nearest grid point to the origin of parcel. The counter J1 takes up values such that grid point (I,J) is at middle of 5 J1 values. Say J = 3 then J1 = 1 to 5. The corresponding coefficients of equations 3, 4, 5, 6 and 7 are determined for every X-direction row stored in B(1) to B(5) and these in turn are used in computation of A(1) to A(5) given by the relation $$A_i = \frac{a_{4i}}{24}\zeta^4 + \frac{a_{3i}}{12}\zeta^3 + \frac{a_{2i}}{24}\zeta^2 + a_{1i}\zeta + a_{0i}$$ This is achieved by initializing A(JJ) = B(5) and using a recursive relation by setting up DO loop 11 to index I1 through 1 to 4. DO loop 11 closes giving 5 values of interpolated values of function in X-direction. |
| C-D | Taking A(3) as the central value the coefficients for Y direction are dtermined using relations (3)-(7) and stored in B(1) to B(5). XYINT4 (giving the final interpolated value) is intialized as B(5) and DO loop 12 is set up to index I1 through 1 to 4. The relation $$XYINT4 = \frac{a_4}{24}\eta^4 + \frac{a_3}{12}\eta^3 + \frac{a_2}{24}\eta^2 + a_1\eta + a_0$$ gives the interpolated value. DO loop 12 ends and the interpolated value is returned to the calling program. |

4.8 Subroutine: FUNCTION XYINT

Abstract: This subroutine interpolates the given function to the point near a given point using a second order interpolation polynomial. This is used for the inner boundary points i.e. I=2 and LM1 J=2 and BMM1). The interpolation is first carried out in x-direction, then in y-direction using the relation.

$$F_o = F_{i,j} - \xi\frac{\partial F}{\partial x} - \frac{1}{2}\xi^2\frac{\partial^2 F}{\partial x^2} \qquad (1)$$

$$F_o = F_{i,j} - \eta\frac{\partial F}{\partial y} + \frac{1}{2}\eta^2\frac{\partial^2 F}{\partial y^2} \qquad (2)$$

Subroutines called:

Input via common

| Variable | Meaning |
|---|---|
| DLAM | Grid interval in degrees longitude in E-W direction. |
| DPHI | Grid interval in degrees latitude in N-S direction. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | DO loop 11 is set up to index JJ through 1 to 3. Another index J1 is defined which keeps the given point I,J at middle of the domain of interpolation. The interpolation is carried out in x-direction using (1) and stored in A(JJ). DO loop 11 ends and interpolation is carried out in y direction using relation (2) and the interpolated value of the given function is returned in XYINT. |

4.9 Subroutine: FUNCTION VADV

Abstract: In this function routine vertical advection of moisture at a given point is calculated.

Subroutines called: None

Input via COMMON: None

Input via argument list

| | |
|---|---|
| WD1 | $0.5 * \dot{\sigma}\ \partial/\partial\sigma$ at upper boundary of a layer. |
| WD2 | $0.5 * \dot{\sigma}\ \partial/\partial\sigma$ at lower boundary of a layer. |
| QQ(L,M,N) | Specific humidity |
| I | Index of x coordinate of a point |
| J | Index of y coordinate of a point |
| K | Index of $\sigma$ coordinate of a point |
| KM1 | K-1 |
| KP1 | K+1 |
| L | Maximum number of grid points in E-W |
| M | Maximum number of grid points in N-S |
| N | Number of $\sigma$ layers |

Output

| Variable | Meaning |
|---|---|
| VADV | Vertical advection of moisture |

Description of Subroutine

| Designator | Text |
|---|---|
| A-B | Vertical advection of moisture is calculated using the relation |

$$\overline{\dot{\sigma}\ \frac{\partial q}{\partial \sigma}}^{\sigma}$$

Control passes to calling routine

## 5.1 Subroutine: DRYCON1

Abstract: In this subroutine the dry convective adjustment is carried out. The adjustment removes any superadiabatic lapse rate in a column by making use of the conservation of dry static energy.

Called by: YCAL

Subroutines called: T2TT

Input via common: None

Input via argument list

| Variable | Meaning |
|---|---|
| T | Temperatures at specified $\sigma$ levels. N values from top to bottom |
| PT | Pressure at specified $\sigma$ levels N values from bottom to top. |
| DT | Time step |
| NS | Number of $\sigma$ levels |

Output

| Variable | Meaning |
|---|---|
| DADJI | Temperature change due to dry convective adjustment in a column |

**Description of subroutine**

| Designator | Text |
|---|---|
| A-B | Interpolate the pressure and temperature values at N specified $\sigma$ levels to get additional levels between these levels. For this the parameters are defined.<br>NDIVI = 5<br>NN = (N-1) • NDIVI + 1 = 41<br>NN1 = NN - NDIVI = 36<br>In DO loop 10, the heating/cooling due to dry convective adjustment is initialized to '0' and pressure value of specified levels is stored in an array PA such that<br>PA(KK) = PT(K)<br>where KK = N - K+1<br>i.e. order is changed from top to bottom. Subroutine T2TT is called at this stage to give pressure and temperature value at NN levels. |

B-C DO loop 20. Potential temperature at NN levels is computed using Poisson equation and stored in array TT. Iteration counter ITT is initialized

ITT = 1

and the control transfers to 235

C-D Statement 205. The control comes here when ITT ‡ 1. The interation counter is increased by 1
ITT = ITT+1 and if ITT = 20 control transfers to statement 225. Otherwise the control transfers to statement 235.

D-E Statement 225. The control comes to this statement if ITT = 20 i.e. convergence is not achieved in 19 iterations. This warning is printed in Format 11. Temperatures at NN interpolated levels is printed. Subroutine T2TT is called to compute pressure and temperature at Nn interpolated levels. In DO loop 1000 potential teperature at N specified levels are computed and printed.
In DO loop 1010 potential temperature at NN interpolated pressure levels are computed and printed. The control transfers to calling program.

E-F Statement 235. Control comes here before the computations of dry convective adjustments start for each iteration. For adjustment computations are made from <u>top to bottom</u>.
Lowest layer temperature differnece $\Delta\theta_{NN}$ is assigned value = '0'. In DO loop 30 the temperature chages from higher to next lower level are computed and stored in DTH array. The counter in this loop runs from 1 to NN-1

F-G INDEX = 1. This initialized value implies that there is no unstable layer. Then the search for unstable layer is made in do loop 40.
If $\Delta\theta_k = (\theta_{k+1} - \theta_k) > 0$
INDEX is switched to - ve value.

INDEX = -1

DO loop 40 continues
If iteration = 1 and INDEX = 1 implying that no unstable layer exists so no adjustment is necessary and control returns to the calling program. If ITT ‡ 1 but INDEX = 1 i.e. the convergence has already been achieved, then control passes to statement 305

G-H Control comes to DO loop 130 when INDEX = -1 i.e. some unstable layer exists. Search for unstable layers is started.
In DO loop 130 top of the unstable layer is searched. $\Delta\theta_k > 0$ gives level of instability. The level value is stored inKK1. KK1 lies between 1

and NN-1

$1 \leqslant KK1 \leqslant NN-1$

when $\Delta\theta_k > 0$ control transfers to statement 15. Otherwise loop 130 continues

H-I

Statement 15. Bottom of the unstable layer is searched.

$\Delta\theta_k < 0$

gives the bottom level of unstable layer. The bottom level k is stored in KK2. When such a level is reached control passes to statement 25. Otherwise DO loop 140 continues.

$KK1 \leqslant KK2 \leqslant NN-1$

I-J

To determine whether extension of adjustment layer is possible to levels above KK1 and below KK2.
If KK2 = NN-1 the adjustable layer cannot be extended beyond KK2 and therefore control passes to statement 105. If KK1 = 1 the adjustable layer cannot be extendd above it and, therefore, control passes to statement 315.

J-K

Average temperature of layers between KK1 and KK2 is determined in DO loop 50 and stored in SUM.

K-L

Search for adjustable levelabove the top of the unstable layer. Initialize the desired level

KK3 = 1

In DO loop 60 search for such a level is made. Adjustable layer is extended to level up to which $\theta_k$ < average temperature of layer (KK1, KK2), say $\theta_k$ i.e.
If $\theta_k$ < SUM
GO to 35 and KK3 = K-1 at this stage. The counter K of the loop 60 runs from 1 to KK1

L-M

Similarly adjusted level below the bottom of unstable layer is determined according to criteria

$\theta_k$ > SUM

The level is stored in KK4. This means that now the adjusted temperature will be assigned to levels from KK3 to KK4 instead of KK1 to KK2.

M-N

Statement 55. The average potential temperature of layer KK1, KK2 is assigned to levels between KK3 and KK4 in DO loop 80

$\theta_k$ = SUM

Loop 50 ends
Then the control goes to statement 205 for next iteration

N-O

Statement 105. Here the control come when KK2 = NN-1 the adjustable layer cannot be exended

beyond bottom of unstable layer. In such a case average temperature of layer KK1 to level NN is computed in DO loop 100. Search for adjustable level KK3 is made above KK1 in DO loop 90 and the adjusted temperature SUM is assigned to levels between KK3 and NN in DO loop 160.
The control passes to statement 205 for next iteration.

O-P

Statement 315. Control comes to this statement when KK1 = 1 i.e. adjustable layer cannot be extended above top of unstable layer. In such a case search for adjustable level KK4 is made below KK2 level and the average temperature of layer KK1, KK2 is assigned to levels between 1 and KK4.
The control passes to statement 205 for next iteration.

P-Q

Statement 305. DO loop 190 starts. The index runs from 1 to NN. The control comes here when the superadiabatic lapse rate has been removed from a column indicated by INDEX = 1. In DO loop 190 temperature TT at NN pressure levels are recovered using Poisson equation

Q-R

In DO loop 300 the adjusted temperature are stored in specified level temperature array TN. DO loop 300 closes. In DO loop 310 the tendency of temperature due to dry convective adjustment is computed

$$DADJI(K) = \frac{\partial T}{\partial t} = (TN_k - T_k)/\Delta t$$

DO loop 310 closes
Control transfers to calling program

5.2 Subroutine: T2TT

Abstract: In this subroutine the interpolation of pressure and temperature is carried out to give 4 additional levels between any two specified pressure levels for fine resolution for the purpose of dry convective adjustment. Temperature at specified pressure levels is also interpolated linearly on log of pressure scale.

Called by: DRYCON1

Subroutines called: None

Input via common: None

Input via arguments list

| Variable | Meaning |
|---|---|
| T | Temperature at N $\sigma$ levels i.e. specified levels |
| P | Pressure at N $\sigma$ levels i.e. specified levels |
| N | Number of specified pressure levels |
| NN | Number of interpolate pressure levels |
| NDIVI | Number of parts in which the pressure interval between two specified pressure levels is divided. |
| NN1 | NN - NDIV |

Output

| Variable | Meaning |
|---|---|
| TT | Temperature at NN interpolated levels |
| PP | Pressure at NN interpolated levels. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | DO loop 10. The index K runs from 1 to N specified pressure levels. A counter K1 is defined as<br>K1 = (K-1) * NDIVI + 1<br>so that<br>PP(K1) = P(K)<br>therefore PP(1) = P(1), PP(NDIVI+1) = P(2) and so on. |
| B-C | DO loop 20. Outer DO loop, the counter K runs from 1 to NN1 at interval NDIVI. In this DO loop the |

pressure at interpolated pressure levels between two consecutive PP levels defined in A-B are computed. For this first the interpolation interval is computed.

DP = PP(K+NDIVI) - PP(K)/NDIVI

```
------------------------- PP(K) TT(K)
------------------------- PP(K+1)
------------------------- PP(K+2)
------------------------- PP(K+3)
------------------------- PP(K+4)
------------------------- PP(K+NDIVI)
                          TT(K+NDIVI)
```

Inner DO loop 100 is initiated with counter KK running from 1 to NDIVI - 1. This DO loop gives (NDIVI-1) additional levels between any two given levels

K1 = K + KK

K comes from do loop 20

PP(K1) = P(K) + KK * DP

DO loop 100 closes

DO loop 20 closes

C-D

Like pressure in A-B, temperatures at interpolated pressure levels are defined. The DO loop 40 starts. The counter K runs from 1 to N specified pressure levels

K1 = (K-1) * NDIVI + 1

and

TT(K1) = T(K)

i.e. TT(1) = T(1), TT(6) = T(2) and so on NDIVI = 5

D-E

DO loop 30 starts the counter K runs from 1 to NN interpolated pressure levels. In this loop log of pressure is found and stored again in PP(K)

E-F

DO loop 50 starts with counter K running from 1 to NN1 at interval of NDIVI (=5). This is outer DO loop. DO loop 110 starts, the counter KK running from 1 to NDIVI - 1. This gives values at additional levels between two given levels

K1 K KK

K comes from loop 50

$$TT(K1) = TT(K) + \{TT(K+NDIVI) - TT(K)\} \times \frac{(PP(K1) - PP(K))}{(PP(K-NDIVI) - PP(K))}$$

DO loop 110 ends

DO loop 50 ends

F-G DO loop 60. In this loop the pressure values at interpolated levels are recovered by finding expotential of PP which at this stage has log of pressure. Control returns to calling program.

5.3 Subroutine: SUPSEA

Abstract: In this routine super saturation in a column is removed and resulting adjusted temperature is computed.

Called by: SUPADJ1

Subroutines called: FUNCTION SPHUM

Input via common: None

Input via argument list:

| Variable | Meaning |
|---|---|
| TO | Temperature at a given level |
| QO | Specific humidity corresponding to TO |
| PO | Pressure at the level of TO and QO |
| I | x coordinate of the given point |
| J | y coordinate of the given point |
| KSP | Level of TO, QO and PO |

Output

| Variable | Meaning |
|---|---|
| TO | Adjusted temperature |
| QO | Adjusted saturation specific humidity at TO |
| PO | Pressure at the level of TO and QO |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | A = 17.27, B = 35.86<br>If TO < 273.16<br>A = 21.87, B = 7.66<br>$L/C_p$ is computed and stored in variable H. Given temperature TO is stored in TG. |
| B-C | DO loop 10. The counter K runs from 1 to 100 indicating 100 iterations are to be carried out. Saturation specific humidity corresponding to TO and PO is computed by calling function SPHUM and stored in QS.<br>The super saturation, when removed raises the tem- |

perature. The modified temperature is found by Newton Raphson's method.
A guess of the heating is computed as
(Modified Temperature - Previous Step Temperature) $+ \frac{L}{C_p}$ (QS - QO)
and stored in FT
Derivative of FT is computed and stored in DFT

$$DFT = 1 + \frac{L}{C_p} q_s \frac{A}{TG-B} - \frac{A*(TG-273.16)}{(TG-B)^2}$$

New guess GTS = TG - FT/DFT
If the difference between new guess and previous value is < 0.1 control goes out of DO loop 10 to statement 15. Otherwise
TG = GTS
and iterative process continues

C-D Control comes here after all the iterations are over. The convergence in the routine has not been achieved. Values of I,J, KSP, TO and TG are printed and control goes to statement 31

D-E Control comes here once convergence is achieved. Then TO = GTS and
QO = 1.0 • SPHUM(PO,TO)
QO gives saturation specific humidity at temperature TO and pressure PO.
Statement 31. Control passes to calling program.

5.4 Subroutine: SUPADJ

Abstract: In this routine point of maximum specific humidity is determined. Other calculations in the routine, for example, supersaturation rainfall are no longer in use.

Subroutines called: SPHUM

Input via COMMON:

| Variable | Meaning |
|---|---|
| PL(L,M) | Log of surface pressure |
| PP(L,M,NP1) | Model parameter P |
| SIG(NP1) | Values of sigma at sigma levels |
| SIGT(N) | Values of sigma at sigma layers |
| R | Gas constant |
| G | Gravity |

Output

| Variable | Meaning |
|---|---|
| RMAX | Maximum relative humidity value |
| IMAX | x-coordinate of RMAX |
| JMAX | y-coordinate of RMAX |
| KMAX | Vertical coordinate of RMAX |
| NPOINT | Number of grid points at which supersaturation is occuring |
| NPOINT1 | Number of grid points higher than layer 5 at which super saturation is occuring |
| NPOINT2 | Number of grid points at or lower than layer 5 at which the super saturation is occuring |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The variables, NPOINT, NPOINT1, NPOINT2, RMAX, IMAX, JMAX, KMAX are initialized as '0' |
| B-C | DO loop 20 starts. Major DO loop. Index J runs from point 1 to M and I runs from point 1 to L. Index IADJ is assigned value 1. |

C-D Pressure at NP1 layer boundaries are computed in DO loop 50. DO loop 40 starts. Index K runs from layer 1 to layer N. In this DO loop pressure at the layers are determined and then corresponding temperatures are computed using relation

$$T_k = \frac{-\sigma_k}{R} \frac{\partial p}{\partial \sigma} - R \cdot \frac{\partial T^*}{\partial \sigma} \ln p_s$$

By calling the function routine SPHUM, saturation specific humidity corresponding to layer temperature and pressure is determined and stored in $Q_s$

D-E For points where model predicted specific humidity Q(K) is > QS the counter NPOINT is increased by 1. Depending upon high or low point counter NPOINT1 or NPOINT2 is increased by 1. Relative humidity is calculated and stored in RMAX. The corresponding grid point indices I, J, K are stored in IMAX,, JMAX and KMAX respectively. Control passes to statement 40.

E-F 40 continue. Control comes here either from C-D or if Q(k) ≤ QS. DO loop 40 ends.

F-G The indices for maximum relative humidity point are printed. Control passes to calling program.

5.5 Subroutine: SUPADJ1

Abstract: In this subroutine the heating and moistening due to super saturation is computed.

Called by: YCAL

Subroutines called: SUPSEA

Input via common:

| Variable | Meaning |
|---|---|
| PT(K) | Values of presure at N, $\sigma$ levels |
| T(K) | Values of temperature at N $\sigma$ levels with order reversed i.e. bottom to top. |
| QQ(I,J,KK) | Values of specific humidity at N_ $\sigma$ levels with order reversed i.e. bottom to top. |
| DT | Time step |

Input via argument list

| Variable | Meaning |
|---|---|
| LCONV1(K) | Value of stability index at N $\sigma$ levels |

Output

| Variable | Meaning |
|---|---|
| HTSUP(K) | Value of heating due to super saturation at N $\sigma$ levels. |
| HQSUP(K) | Value of moistening due to super saturation at N $\sigma$ levels. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | DO loop 50. The heating and moistening parameter at N $\sigma$ levels are initialized. |
| B-C | DO loop 40. The counter K runs from 1 to N $\sigma$ levels. A variable<br>KK = N - K+1<br>is defined to reverse the order of parameters. When LCONV1(KK) = 899, this large scale stable rain has been computed. In that case heating due to super saturation is not computed and control |

passes to statement 40. When LCONV1(KK) = 899

```
Q(K) = QQ(I,J,KK)
```

Q(K) is specific humidity at N $\sigma$ levels from bottom to top.

The saturation specific humidity at T(KK) is computed and stored in QS.

If Q(K) > QS then

```
     TO = T(KK)
     QO = Q(K)
```

and routine SUPSEA is called to remove super saturation and provide adjusted values TO an QO using the adjusted values, net heating and moistenig are computed and stored in

```
     HQSUP(K) = (QO - Q(K))/Δt
```

and

```
     HTSUP(K) = (TO - T(KK))/ Δt
```

The value of stability index is reassigned as 799.

DO loop 40 closes.

Control passes to calling program.

## 5.6 Subroutine: FUNCTION SPHUM

Abstract: In this function routine saturation specific humidity for a given pressue 'p' and temperature TT is computed.

Subroutines called: None

Input via COMMON: None

Input via arguments list

| Variable | Meaning |
|---|---|
| P | Pressure at a given grid point |
| TT | Temperature at a given grid point |

Output

| Variable | Meaning |
|---|---|
| SPHUM | |

| Designator | Text |
|---|---|
| A-B | The temperature is converted to °C from °K. Constants A and B of Tetan's formula are determined depending upon whether temperature is more than 0°C or less than that. Saturation specific humidity is then calculated using Tetan's formula. |

$$e_s = 6.11 \cdot e^{\frac{A \cdot T}{(T+273.16-B)}}$$

$$SPHUM = 0.622 \cdot \frac{e_s}{(p-0.378 \cdot e_s)}$$

Control passes to calling program.

## 6.1 Subroutine: RADN

Abstract This is main subroutine of the radiation package. This is called by rouinte YCAL. This routine specifies the various categories of clouds, evaluates the net short and long wave fluxes, short and long wave convergences, total heating, and the cooling and warming rates. This is a one dimensional column routine and prints all values in vertical besides printing the surface balance.

Subroutines called: (i) CLOUD (ii) FLUX (iii) ZETA1 (iv) SLR (v) TG (vi) CLDCOL

Input variables via common:

| Variable | Meaning |
|---|---|
| ALBD(I,J) | Albedo at the point I,J. This refers to surface albedo. In the routine this value is assigned to variable ALBSFC. |
| I | x-coordinate grid point count i.e. in east-west. |
| J | y-coordinate grid point count i.e. in N-S direction |
| QQ(L,M,N) | The specific humidity over all grid points of the domain L•M•N |
| KT | Time step count. This variable is compared with KRAD to decide printing of variables |
| KRAD | Radiation variables printing time step count. At integral multiple of KRAD, radiation variables are printed |
| T(N) | The temperature at N sigma at ($\sigma$) levels. These values are assigned to array 'TEMP' in the routine with order reversed i.e. TEMP(1) = Temperature at $\sigma$ = 0.95 level. TEMP(1) = T(N) etc. |
| TT(NP1) | The temperature at NP1 sigma levels. These values are assigned to array 'TZ' in the routine with order reversed. |
| PT(N) | Pressures at N($\sigma$) levels with order from bottom to top. |
| PZ(NP1) | Pressures at NP1($\sigma$) levels with order from bottom to top |
| DPZ(N) | $\Delta p = (PZ_k - PZ_{k+1})$ N values. These values are available at N $\sigma$ levels. |

| | |
|---|---|
| T2 | Air temperature at surface level. This value is assigned to T1000 in this routine |
| COSD | Cosine of angle of declination |
| SIND | Sine of angle of declination |
| COSPHI(M) | Array representing Cosine of latitude $\phi$. |
| SINPHI(M) | Array representing Sine of latitude $\phi$. |
| MON | Month |
| NDAY | Starting day of integration of model |
| DLAMD | Grid interval along E-W in degrees longitude |
| GAM | $g/C_p$ |
| WLONG | Western most longitude of the domain |
| RS | Insolation received at the top of the atmosphere. Given by the relation<br>$RS = (1.935 \times 4.2\times10^7/60/d^{-2})$ Erg $Cm^{-2}$ $sec^{-1}$<br>$= 1.38\times10^3$ $Wm^{-2}$<br>where 1.935 is solar constant in langleys (Cal/min)<br>$d^2$ is mean constant sun distance = $1.50\times10^{11}$m |
| ZZHOUR | Hour of the day |
| ROMEGA | 360/24 Earth's angular velocity in degrees/HR |
| RADC | Conversion factor for degrees to radians conversion = $\pi/180$. |
| DPHI | Grid interval in N-S direction in terms of degrees latitude. |
| TSUB(L,M) | Surface temperature. Over the oceans this quantity represents sea surface temperature. |
| QSUB(L,M) | Surface specific humidity |
| RHO | Density of air. At 0°C and 1000 mb pressure |
| SIGMA | Stefan-Boltzmann's constant = $5.6696\times10^{-8}$ $Wm^{-2}deg^{-4}$ |
| PI | Value of $\pi$ |
| ALBAIR | Albedo of air |

| | |
|---|---|
| GINV | 1/g |
| UU(L,M,N) | U component of wind at Nth (i.e. $\sigma$ = 0.95) level |
| VV(L,M,N) | V component of wind at Nth $\sigma$ level. |
| FDW(NP1,NM1) | Downward flux of radiation at NP1 levels for 8 configurations of clouds |
| ASO(NP1,NM1) | DIRECT Solar insolation absorbed by the atmosphere at NP1 $\sigma$ levels for NM1(=8) cloud configurations. |
| CLD(N,NM1) | The array indicating presence or absence of the NM1 (=8) configurations of clouds at N $\sigma$ levels. |
| ICLB | Bottom of low clouds |
| ICLT | Top of low clouds |
| ICMB | Bottom of middle clouds |
| ICMT | Top of middle clouds |
| ICHB | Bottom of high clouds |
| ICHT | Top of high clouds |
| FUD(NP1,NM1) | Upward longwave radiation fluxes at NP1 $\sigma$ levels for NM1(=8) cloud configurations |
| RSO(N,NM1) | Reflected solar insolation for the eight cloud configuration |

Input variables from data statements

| Variable | Meaning |
|---|---|
| ALBC | Albedo of clouds. This array represents 8 configurations of clouds |
| ALBSC | Array representing albedo of clouds for Rayleigh scattering. This also represents 8 configurations of clouds. |

Output Variables

| Variable | Meaning |
|---|---|
| FLG | Downward flux of longwave radiation at surface (ground) |
| FLGU | Upward flux of longwave radiation at surface (ground) |

| | |
|---|---|
| SGDN | Downward flux of shortwave radiation at surface (ground) |
| SGUP | Upward flux of shortwave radiation at surface (ground) |
| SEN | Surface flux of sensible heat (surface temperature flux). Computed in routine 'TG'. SEN = $\rho C_H C_p$ $Vel_{surface} \cdot (T_{1000} - T_{surf})$ (Argument list) |
| CON | Surface flux of latent heat (surface moisture flux). Computed in routine TG. CON = $\rho C_q C_p$ $Vel_{surface} \cdot (q_{1000} - q_{surface})$ (Argument list) |
| FLGU | Upward longwave flux from the ground = $-\sigma T_g^4$. Computed in routine Tg. (Argument list) Surface (ground) temperature. Computed in routine Tg as $T_s$. Comes in routine RADN as TSFC through arguments list. |
| QSFC | Ground specific humidity. Computed in routine Tg as $Q_s$. (Argument list) |
| SNET(NP1) | Net shortwave fluxes at NP1 $\sigma$ levels (Net shortwave irradiances) (Layer boundaries) |
| SWUP(NP1) | Upward shortwave fluxes at NP1 $\sigma$ levels (Layer boundaries) |
| SWDN(NP1) | Downward shortwave fluxes at NP1 $\sigma$ levels (Layer boundaries) |
| LNET(NP1) | Net longwave fluxes at NP1 $\sigma$ levels. (Net longwave irradiances) (Layer boundaries) |
| LWUP(NP1) | Upward longwave fluxes at NP1 $\sigma$ levels (Layer boundaries) |
| LWDN(NP1) | Downward longwave fluxes at NP1 $\sigma$ levels (Layer boundaries) |
| SCONV(N) | Shortwave convergence at N $\tilde{\sigma}$ levels (Layers) |
| LCONV(N) | Longwave convergence at N $\tilde{\sigma}$ levels (Layers) |
| WARM(N) | Warming rate at N $\tilde{\sigma}$ levels after averaging over NM1(=8) configurations of clouds. |
| COOL(N) | Cooling rate at N $\tilde{\sigma}$ levels after averaging over NM1(=8) configurations of clouds. |
| TEMPT(N) | Net temperature change (heating/cooling) at N $\tilde{\sigma}$ levels which is sum of warming and cooling rate at these levels |

| | |
|---|---|
| DTC(N,NM1) | Cooling rate for 8 different cloud configurations |
| DTW(N,NM1) | Warming rate for 8 different cloud configurations |

Other relevant variables and array names appearing in the routine 'RADN'

| Variable | Meaning |
|---|---|
| EL | Correction index due to the differnce between air temperature and ground temperature |
| ALBC(1) | Albedo of cloud configuration no. 1 i.e. clear sky. It is assigned the value of surface albedo |
| T1000 | Ground temperature |
| Qs1000 | Saturation specific humidity for ground temperature T1000 |
| ICLR | Index for clear sky = 1 |
| ICCL | Index for only low cloud = 2 |
| ICLM | Index for low and middle cloud = 3 |
| ICLMH | Index for low, middle and high cloud = 4 |
| ICCM | Index for only middle cloud = 5 |
| ICCH | Index for only high clouds = 6 |
| ICMH | Index for middle and high clouds = 7 |
| ICLH | Index for low and high clouds = 8 |
| Q(N) | Specific humidity at N $\sigma$ levels from ground to top. |
| TZ(NP1) | Temperature at NP1 $\sigma$ levels from ground upward |
| TEMP(N) | Temperature at N $\sigma$ levels from ground, upwards |
| RH(k) | Relative humidity at N $\sigma$ levels from ground, upwards. |
| RW(k) | Optical path length at N $\sigma$ levels upwards from ground. Computed from top to bottom, indexed from bottom to top. |
| F(NP1) | Blackbody flux at NP1 $\sigma$ levels. It is function of temperatures 'TZ' at these levels. |
| CL | Fractional low cloud cover |

| | |
|---|---|
| CM | Fractional middle cloud cover |
| CH | Fractional high cloud cover |
| | $0 \leqslant (C_L, C_M, C_H) \leqslant 1$ |
| CLR | Fractional cover of clear sky |
| CCL | Fractional cover of low cloud |
| CLM | Fractional cover of low and middle clouds |
| CLMH | Fractional cover of low, middle and high clouds |
| CCM | Fractional cover of middle clouds |
| CCH | Fractional cover of high clouds |
| CMH | Fractional cover of middle and high clouds |
| CLH | Fractional cover of low and high clouds |
| COE(8) | Weight coefficient for eight different cloud configurations |
| FDG | Downward longwave flux reaching ground giving equal weight to all eight configurations of cloud. It is multiplied by COE(NM1) and added to give total downward longwave flux at ground |
| RSS | Scattered solar radiation |
| RSA | Absorbed solar radiation |
| DTW(N,NM1) | Warming rate at N $\sigma$ levels for 8 different configurations. |
| FSGA | Total absorbed part of insolation at the surface. This is sum of absorbed solar radiation for eight configuration of clouds multiplied by the respective coefficients |
| FSGS | Total scattered part of insolation at the surface. |
| SSO(NM1) | Scattered solar radiation for the eight configuration of clouds |
| ABSV | Wind velocity at ground |
| RESG | Saturation vapor pressure at ground |
| RQSG | Saturation Specific humidity at ground |

| | |
|---|---|
| GW | Ground wetness parameter |
| FNET(NP1,NM1) | Net flux of radiation at NP1 $\sigma$ levels for eight cloud configurations |
| DTC(N,NM1) | Cooling rate for 8 different cloud configurations at N $\tilde{\sigma}$ levels bottom to top |
| COOL(N) | Cooling rate after averaging over 8 different configurations |
| SLONG | Longitude of the point at which zenith angle is computed |
| NHOUR | Number of hours on which zenith angle is to be averaged. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Constants, N1 (number of levels), EL, ALBSFC, ALBC(1) and indices for 8 configurations of clouds are defined. Temperature T1000 is assigned the air temperature value T2. Specific humidity Q1000 at T1000 is computed. Blackbody radiation flux at the <u>top</u> of the model is computed: F(NP1) |
| B-C | In DO loops 32 and 33 the orders of temperatures (T and TT) and humidity (QQ) are reversed to bottom to top. In DO loop 35 corresponding R.H. are computed and upper and lower limits are set to 1.0 and 0.1 respectively. The path lengths RW(k) for water vapour are computed. The path length at the top of the model is taken as RW(NP1) = $4.7 \times 10^{-5}$ and at other levels $RW_k = RW_{k+1} + q_k x \Delta p_k x \frac{10.0}{g} \times \frac{p_k}{1000}^{0.85} \times \frac{T1000}{T}^{0.5}$ Here $\left[\frac{p}{p1000}\right]^{0.85}$ takes into account the pressure reduced factor. Irradiances at remaining N levels: F(N) are computed RH1000 is defined. |
| C-D | Fractional cloud cover and corresponding 8 weight coefficients are computed. The fractional cloud cover and bottom and top of clouds are determined in subroutine CLOUD |
| D-E | Initialize FUP(NP1,NM1), FDW(NP1,NM1), DTC(N,NM1) in loop 25. Initialize ground temperature TSFC = 0 |

E-F For the cloud configurations that are present compute the downward longwave fluxes by calling routine FLUX (This routine uses the emissivities from the various cloud configurations. The flux is largest in case of clear sky.) and store in array FDW. If FDW is zero the flux is taken same as at ground. Then a weighted sum is found and stored in FLG. FLG is total downward flux of longwaves at surface.

F-G To compute the net solar heating at all the model levels, the zenith angle

$\cos\zeta = \sin\phi\sin\delta + \cos\phi\cos\delta\cos h$

is computed by calling routine 'ZETA1' then the scattered and absorbed parts of the solar radiation are computed:

RSS = scattered part = $0.651 x S_o x \cos\zeta$

RSA = absorbed part = $0.349 x S_o x \cos\zeta$

Initialize the warming rate DTW for the eight configuration. The direct solar insolation and reflected solar insolation are computed by calling routine SLR for the cloud configuration that are present. The net warming for each configuration is computed and stored in DTW at N $\sigma$ levels.

$$DTW = \frac{g}{C_p} \times 10^{-2} \times \frac{0.001}{\Delta p_{k-1}} \times \Delta ASO_{k,k-1} + \Delta RSO_{k-1,k}$$

$10^{-2}$ (mb to Pa conversion factor). .001 erg/cm$^2$ to J/m$^2$ conversion factor. The weighted sum of DTW's is found to give warming rate at N levels. The rate at top is taken as zero. DTW is the flux divergence of the absorbed and reflected solar radiation.

G-H Computes the sum of scattered and absorbed solar insolation at ground. For absorbed part, a weighted sum of ASO is found in loop 137. For scattered part computations are made if zenith angle is somewhat < 90°. For $\cos\zeta > 0.01$, FSGS=0. For FSGS computations, albedo of air is obtained using $0.085 - 0.245 \log10\cos\zeta * \frac{P_{sfc}}{P_{1000}}$ limit is set = 1.0. In loop 138 contributions of all the eight configuration are added to get FSGS. $R_1$, $R_2$, $R_3$ denote the reflected solar insolation from lower, middle and high clouds. The formula evaluated is

$$R_{123} = 1 - \frac{(1-R_1)(1-R_2)(1-R_3)}{(1 - R_1R_2 - R_2R_3 - R_3R_1 + 2R_1R_2R_3)}$$

$X = (1 - R_{123})$/albedo of air.

Then scattered solar radiation, SSO = (1-X)/(1-X*albedo surface).
Then sum of FSGA and FSGS is found and stored in FSG. If zenith angle ($\cos\zeta$) $\leqslant$ 0.01, FSG = 0.

Multi cloud albedo for scattered radiation

$$R_{123} = 1 - \frac{(1-R_{12})(1-R_3)}{(1-R_{12}R_3)}$$

$$= 1 - \frac{\dfrac{(1-R_1)(1-R_2)}{1-R_1R_2}\,(1-R_3)}{1 - \left[1-\dfrac{(1-R_1)(1-R_2)}{1-R_1R_2}\right] R_3}$$

$$= 1 - \frac{(1-R_1)(1-R_2)(1-R_3)}{(1-R_1R_2) - [1-R_1R_2-(1-R_1)(1-R_2)]R_3}$$

$$= 1 - \frac{(1-R_1)(1-R_2)(1-R_3)}{(1-R_1R_2) - [-2R_1R_2-R_1-R_2]R_3}$$

$$= 1 - \frac{(1-R_1)(1-R_2)(1-R_3)}{1 - R_1R_2 - R_1R_3 - R_2R_3 + 2R_1R_2R_3}$$

Surface down: $(1-\alpha_A) + (1-\alpha_A)\alpha_s\alpha_A + (1-\alpha_A)(\alpha_s\alpha_A)^2 + \ldots = \dfrac{1-\alpha_A}{1-\alpha_s\alpha_A}$

up: $(1-\alpha_A)\alpha_s + (1-\alpha_A)(\alpha_s\alpha_A)\alpha_s + (1-\alpha_A)(\alpha_s\alpha_A)^2\alpha_s + \ldots = \dfrac{(1-\alpha_A)\alpha_s}{1-\alpha_s\alpha_A}$

Top up: $\alpha_A + (1-\alpha_A)^2\alpha_s + (1-\alpha_A)^2\alpha_s\alpha_s\alpha_A + \ldots = \alpha_A + \dfrac{(1-\alpha_A)^2\alpha_s}{1-\alpha_s\alpha_A}$

$$= \frac{\alpha_A + \alpha_s - 2\alpha_A\alpha_s}{1-\alpha_s\alpha_A} \quad (1)$$

Absorbed at surface: $\dfrac{(1-\alpha_A)(1-\alpha_s)}{1-\alpha_s\alpha_A}$

$$= \frac{(1-\alpha_s-\alpha_A+\alpha_s\alpha_A)}{(1-\alpha_s\alpha_A)} \quad (2)$$

$$(1) + (2) = 1$$

H-I Over the land, surface ground temperature, sensible heat flux and latent heat flux and upward longwave flux at the ground are determined in the routine

$T_g$, for the surface heat balance computations. The condition of albedo $\leqslant 0.06$ is tested for this purpose. Over the land surface the values of ground temperature and ground specific humidity are stored in TSUB and QSUB respectively. Over the oceans the variables TSFC and QSFC are assigned values from the arrays TSUB and QSUB respectively.

I-J Upward fluxes of longwave radiation are computed for the cloud configurations that are present. In loop 3 the sum of the upward and downward longwaves radiation are found at NP1 $\sigma$ levels and for NM1 cloud configurations.

J-K Longwave cooling rate DTC due to flux divergence of radiative energy in degrees per day is calculated here. Before this the routine CLDCOL is called which fixes flags CLD at levels in vertical allowing cooling only at the bottom and top of the cloud, so making the net flux effectively zero inside the cloud. The divergence

$$\frac{g}{C_p} \frac{\partial F}{\partial P} \times CLD$$

is computed at N levels from 1 to N in DO loop 4. In loops 14, 15 the net cooling at N levels due to all the cloud configurations are calculated.

K-L At the surface (ground) the (i) downward shortwave radiation, (ii) upward S.W. radiation and the (iii) net shortwave radiation are computed.

(i) SGDN = FSGA+FSGS

(ii) SGUP = $-\alpha_{surface}$ * SGDN

(iii) SGNET = SGDN+SGUP

L-M For radiation time step i.e. when KT is integer multiple of KRAD various radiation parameters (variables) are printed for a point (25,27).
PRINT 300 statement prints the title SURFACE BALANCE. 310 prints titles SWDN, SWUP, LWDN, LWUP, SEN, LATENT, TSURF, QSURF. 441 prints values of these variables. 320 prints K, PRES, TEMP, REL HUM, SWDN, SWUP, LWDN, LWUP, SNET, LNET, SCONV, LCONV, WARM, COOL, NETRATE titles. 330 prints MB, (k), (%), 2(W/MSQ), 4(W/MSQ), (W/MSQ), (W/MSQ), 2(k/D), (k/D) titles.

M-N If time step is not radiation step the above printing is skipped. After printing or without it, the control comes to 399. Loop 444 is initiated in this net shortwave radiation (SNET), longwave upward flux LWUP, longwave downward flux (LWDN), net longwave flux (LNET), upward shortwave flux (SWUP), downward shortwave flux (SWDN), and except

for lowest (uppermost) level, short wave convergence (SCONV) and longwave convergence are initiated to zero. In inner loop 44, the above quantities are obtained at all levels except NP1 level. Loop 44 closes. Except for level NP1, the net warming/cooling rates are computed for all levels and are stored in TEMPT. for radiation time step the above quantities are printed according to formats 442 and 443. Loop 444 closes.

N-O After the format statements, the control is returned to calling program YCAL.

## 6.2 Subroutine: CLOUD

Abstract: This subroutine determines the fractional areas of low middle and high clouds on the basis of critical (prescribed) relative humidity values. The top and bottom levels of all cloud types are determined. Flags CLD (level, cloud configuration) are also obtained determining presence or absence of a configuration at given level.

Subroutines called: None

Input variables via common: None

Input variables via argument list

| Variable | Meaning |
|---|---|
| PT | Pressure at N $\sigma$ levels |
| RH | Relative humidity at N $\sigma$ levels |

Input variables in data statements

| | |
|---|---|
| PCLB | Pressure at bottom limit of low cloud |
| PCLT | Pressure at top limit of low cloud |
| PCMB | Pressure at bottom limit of middle cloud |
| PCMT | Pressure at top limit of middle cloud |
| PCHB | Pressure at bottom limit of high cloud |
| PCHT | Pressure at top limit of high cloud |
| RHCL | Critical relative humidity for low clouds |
| RHCM | Critical relative humidity for middle clouds |
| RHCH | Critical relative humidity for high clouds |

Output variables

| Variable | Meaning |
|---|---|
| ICLB | Bottom level of low clouds |
| ICLT | Top level of low clouds |
| ICMB | Bottom level of medium clouds |
| ICMT | Top level of medium clouds |

| | |
|---|---|
| ICHB | Bottom level of high clouds |
| ICHT | Top level of high clouds |
| CL | Fractional area of low cloud cover |
| CM | Fractional area of medium cloud cover |
| CH | Fractional area of high cloud cover |
| CLD (N,NM1) | A flag that determines whether at a given level a particular configuration of clouds exist or not. This is used in flux computations. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | R.H. and top level indices of low, medium and high clouds are initialized as zero. Bottom level indices of the clouds are initialized as N. |
| B-C | DO loop 1 is initiated to index the counter IL from 1 through N. A check is made whether the given pressure level lies between the limits of lower cloud. If yes, then the R.H. value at that level is checked against RHCL. If it is ⩾ RHCL then the bottom of low cloud is chosen as minimum out of current value of IL and ICLB. The top is assigned as IL+1. Control is transferred to statement 1, DO loop limit. In this part total R.H. of low cloud column s also determined (RHL) |
| C-D | Statement No. 3. A procedure similar to B-C is adopted for medium clouds. Values of ICMB and ICMT are determined. |
| D-E | Statement 5. A procedure similar to B-C and C-D is adopted for high clouds. Values of ICHB and ICHT are determined. Loop 1 closes. |
| E-F | DO loop 20. The counter JR (for cloud configuration) is indexed through 1 to NM1 and counter K through 1 to N. The values of the Flag CLD(N,NM1) are initialized as 1. |
| F-G | Upto statement number 121 the fractional area for low cloud is computed and flag CLD is assigned appropriate values. If the cloud bottom ICLB > ICLT, it implies that low cloud does not exist and therefore $C_L = 0$ (statement 121). CL is determined as: $$\frac{RHL}{((ICLT-ICLB) - RCHL)} \cdot \frac{1}{(1-RHCL)}$$ |

If the square of CL exceeds 1 it is kept as 1. ICLT1 is defined as ICLT-1. From bottom of cloud up to ICLT1 level flags for configurations 2, 3, 4 and 8 are changed to zero and control is transferred to statement 221. The flag CLD(,)=0 idicates that the given level is within the cloud

G-H Statement 221. A procedure similar to F-G is adopted for medium cloud and glags for configuration 3, 4, 5 and 7 are changed to zero.

H-I Statement 222. A procedure similar to F-G is adopted for high clouds and flags for configuration 4, 6, 7 and 8 are changed to zero.

I-J If the initialized values of ICLT, ICMT, ICHT remain unchanged, the values of ICLB, ICLT, ICMB, ICMT, ICHB, and ICHT are changed to 1. Control transfers to calling routine RADN.

## 6.3 Subroutine: FLUX

Abstract: This routine computes the upward and downward longwave radiation fluxes for a given cloud configuration using the emissivity method, at the top, bottom and middle of clouds.

Subroutines called: EMTAB

Input variables via COMMON

| Variable | Meaning |
|---|---|
| RW(NP1) | The path length (for water vapor only) of radiation at NP1 $\sigma$ levels. |
| F(NP1) | The blackbody irradiances at NP1 $\sigma$ levels given as $\sigma T^4$. |
| TZ(NP1) | The temperatures at NP1 sigma levels from bottom to top. |

Input via data statement: None

Input via arguments list:

| Variable | Meaning |
|---|---|
| ICAT | Index for configuration of cloud ranging from 1 for clear sky to 8 for low and high clouds |
| ITOP | Top of the cloud category ICAT |
| ICLB | Bottom level of low cloud |
| ICLT | Top level of low cloud |
| ICMB | Bottom level of middle cloud |
| ICMT | Top level of middle cloud |
| ICHB | Bottom level of high cloud |
| ICHT | Top level of high cloud |
| TSFC | Surface temperature. But here if called before computation of TSFC in routine 'TG' it is just an indicator that downward fluxes are to be computed. When called after computation of TSFC in 'TG' it indicates that upward fluxes are to be computed. |

Output variables

| Variable | Meaning |
|---|---|

FUP(N,NM1) Upward longwave fluxes at N $\tilde{\sigma}$ levels for NM1 cloud configurations.

FDW(N,NM1) Downward longwave fluxes at N $\tilde{\sigma}$ levels for NM1 cloud categories.

INDEX

Other relevant variable and array names appearing in the routine

| Variable | Meaning |
|---|---|
| SIGMA | $5.66 \times 10^{-5}$ ..... Stefan-Boltzmann's constant. |
| FBDYU | Blackbody irradiance at given temperature |
| WB | Difference of path lengths<br>Either<br>RW(Top of cloud) - RW(given level)<br>when upward flux is to be computed or<br>RW(IBOT1) - RW(given level)<br>when downward flux is to be computed. |
| FU | I time: In computation of upward fluxes<br>$= \sigma T^4_{cTop} \cdot (1 - \epsilon[WC_T - W_i])$<br>and II time = I time result $-\int_{W_i}^{W_{cT}} T^4 \frac{\partial t}{\partial w} [W - W_i] dw$<br>It in fact represents upward flux for a given category of cloud.<br>$FU = F_i = T_{cT}^4 (1 - \epsilon[W_{cT} - W_i]) - \int_{W_i}^{W_{cT}} T^4 \frac{\partial t}{\partial w} [W - W_i] dw$ |

<u>I Time</u>

| | |
|---|---|
| DWT | Difference of path length between IL+1 level and K level. Upward flux |
| DWB | Difference of path length between IL level and K level. Upward flux |
| EMB | Emissivity corresponding to DWB |
| EMT | Emissivity corresponding to DWT |

<u>II Time</u>

| | |
|---|---|
| DWT | Path length at K level or difference of path length between K level and IL+1 level |

| | |
|---|---|
| DWB | Difference of path lengths between K level and NP1 level or difference of path lengths between K level and IL level. |
| IBOT | Bottom of high, middle or low cloud |
| IBOT1 | Either ground or top of low cloud or top of middle cloud |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The variable N1 is assigned the value NP1-1 where NP1 refers to top of the model. Blackbody irradiance is computed at temperature of top level of the given cloud configuration. |

B-C

```
_______________________________TZ(NP1)  F(10)

-------------------------------
_______________________________TZ(9)    F(9)

-------------------------------
_______________________________TZ(8)    F(8)

-------------------------------
_______________________________TZ(7)    F(7)

-------------------------------
_______________________________TZ(6)    F(6)

-------------------------------
_______________________________TZ(5)    F(5)

-------------------------------
_______________________________TZ(4)    F(4)

-------------------------------
_______________________________TZ(3)    F(3)

-------------------------------
_______________________________TZ(2)    F(2)

-------------------------------
_______________________________TZ(1)    F(1)
```

Loop 10 is set up. kounter K indexes through ITOP to NP1 level. TSFC is checked against 'Q'. The path difference between the cloud top and relevant level is found. Corresponding emissivity is obtained from 'EMTAB'. Effective upward flux (blackbody) found between cloud top and level i.e.

$$T_{cT}^4 \cdot (1 - \epsilon[W_{cT}-W_i])$$

Variable K1 is defined as K-1. Except for K=ITOP DO loop 1 is set up in which index IL indexes through ITOP to K1. In this the difference of path lengths of bottom and top of the IL layer from K level are found, corresponding emissivities are found and such layer contributions are added to FU

$$\int_{W_i}^{W_{cT}} T^4 \frac{\partial \epsilon}{\partial w} [W-W_i] dw$$

The blackbody irradiances at levels IL are provided in variable F. Loop 1 closes. The flux FU for given K and given category is then stored in array FUP .

C-D

Statement 4. Loop 10 continues. TSFC is checked against zero. For zero value downward fluxes are computed. Path length difference of level K top of atmosphere is stored in DWT and difference of path lengths between level K and level NP1 in DWB. Emissivities are obtained. The effective downward blackbody flux is computed at level K using value at the top of the model and stored in FD

FD = F(NP1)•(EMT-EMB)

For levels except K=NP1, loop 2 is set up with counter IL to index through K to N1 (i.e. NP1-1). Path differnce between level K and IL+1 is found and stored in DWT. Path difference between K and IL is found and stored in DWB. Corresponding emissivities are computed from EMTAB and contributions of all layers are added and stored in FD. Do loop 2 closes. The downward flux for level K and given cloud configuration is stored in array FDW(NP1,NM1) at NP1 $\sigma$ levels. The convention is +ve for downward flux, -ve for upward flux. Loop 10 closes. The formula that is computed from statement 4 to 10 is

$$Fi = \sigma T_{c_b}^4 (1-\epsilon[W_i-W_{c_b}]) - \int_{W_{c_b}}^{W_i} \sigma T^4 \frac{\partial \epsilon}{\partial w} [W_i-W] dw$$

D-E

From statement 10 to statement 18. Indexes are fixed for the cloud categories that are present. If the case is that of clear sky i.e. ITOP=1 the control is transferred to calling program. If ICHB $\neq$ 1 i.e. high clouds are present then INDEX = 3, IBOT = ICHB and IBOT1 is maximum out of top of low

cloud or medium cloud. If high clouds are absent (statement 11) and medium clouds are present, INDEX = 2, IBOT = ICMB and IBOT1 is maximum out of ICLT and clear sky. If no medium or high but low clouds then INDEX = 1. This is done to find upward and downward fluxes in between the clouds.

E-F Blackbody irradiance is computed at temperature of level IBOT1 and stored in FBDYU.

F-G Set up DO loop 30 to index the counter K through IBOT1 to IBOT i.e. between the lower cloud top to immediately above cloud bottom. A procedure similar to BC is followed to compute the upward fllux in DO loop 21 after checking TSFL against zero.

G-H Statement 24. A procedure similar to C-D is adopted to compute the downward flux in DO loop 22.

H-I A check is made for IBOT1. If it is not that of ground i.e. 1 then INDEX = INDEX - 1 and depending upon the value of index the control transfers to 13: for computation between low cloud and ground or to 11: for computation between middle and low clouds. In this way the upward and downward fluxes are computed when there are more than one layer of clouds.

## 6.4 Subroutine: SLR

Abstract: This routine computes the reflected and absorbed solar radiation by water vapor which is the only optically active gas considered in the model. The downward irradiances of absorbed solar radiations are computed for the case of clear sky and with multiple cloud layers. The absorption by cloud droplets takes place in the near i.r. spectra.

Routines called: None

Input via COMMON

| Variable | Meaning |
|---|---|
| RSA | Absorbed solar insolation |
| RSS | Scattered solar insolation |
| RW | Optical path length |
| SECZ | Secant of zenith angle |
| ALBC | Cloud albedo for 8 configurations of clouds |

Input via argument list

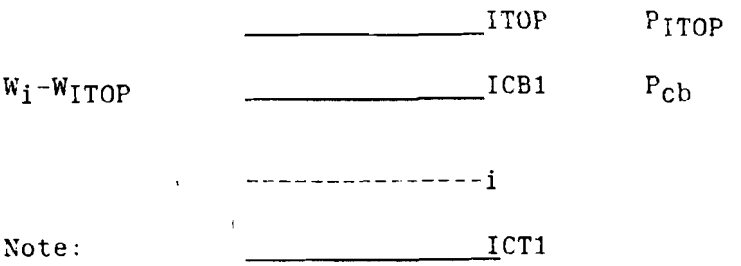
| Variable | Meaning |
|---|---|
| ICAT | Index of cloud configuration. Varies from 1 to 8 |
| ICLB | Bottom level of low cloud |
| ICLT | Top of low cloud |
| ICMB | Bottom of middle cloud |
| ICMT | Top of middle cloud |
| ICHB | Bottom of high cloud |
| ICHT | Top of high cloud |
| PZ(k) | Pressure at NP1 $\sigma$ levels |

Output variables

| Variable | Meaning |
|---|---|
| ASO(NP1,NM1) | Absorbed insolation at NP1 $\sigma$ levels by NM1 categories |
| RSO(NP1,NM1) | Reflected insolation at NP1 $\sigma$ levels by NM1 categories |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Definitions of various quantities are given<br>INDEX = 1; high clouds are present<br>INDEX = 2; no high clouds but middle and/or low clouds could be present<br>INDEX = 3; no high or middle clouds but low clouds could be present<br>ASO: Absorbed insolation. Also represents net downward shortwave flux<br>RSO: Reflected insolation. Represents diffused reflected radiation reaching a level, i<br>RW: Optical path length<br>DW: Optical depth<br>EQCW: Equivalent optical depth<br>ALBC: Cloud albedo for different configuration<br>REASO: Reflected solar radiation<br>SECZ: Secant of zenith angle<br>First the cloud base is initialized at the top level<br>ICB1 = ITOP = NP1<br>In DO loop 6, the reflected (diffuse) radiation is initialized as zero. Direct solar radiation at the top is initialized as the absorbed part of solar radiation at the top of atmosphere<br>ASO(ITOP,ICAT) = RSA |
| B-C | Top level is now set equal to the maximum value among the low, middle and high cloud and assigned to ICT1. In case of clear sky ICT1 = 1. Then a search is made for cloud type present and appropriate index is assigned. In case of clear sky INDEX = 3 as in case of low clouds |
| C-D | The downward solar radiation flux is computed. Statement 100. DO loop 1 is started with counter IL running from level ICT1 (top of lower cloud/earth's surface) to ICB1 (bottom of higher cloud/model top). Except for IL=ICB1, the direct solar radiation is computed.<br>If ITOP = NP1<br>(DW) Optical depth = $W_i$secz at level i<br>If ITOP ‡ NP1<br>$DW = 1.66(W_i - W_{ITOP}) + 1.4 \cdot (P_{cb} - P_{ITOP})/200.0$<br>where $C_p$ = cloud bottom of the higher cloud<br>$C_t$ = ITOP = Top of the lower cloud<br>The optical depth value is augmented by multiplying the path length by a factor 1.66 and the optical depth value 1.4 is the value for a cloud 200 mb thick. In the above formula, the various levels are as indicated below. |




(The path length $W_{ITOP}$ should be corrected to $W_{CB1}$)
ICT1 = ICHT/ICMT/ICLT/1
ICB1 = NP1/ICHB/ICMB/ICLB/
ITOP = NP1/ICHT/ICMT/ICLT/
For a given category, at levels IL=ICT1 to IL=ICB1 the downward flux of shortwave radiation is computed as following

$$AA = ASO_{ITOP} - RSO_{ITOP}$$

giving net downward radiation at cloud top (or at model top in case of clear sky)

$$ASO_i = AA \left\{ 1 - A[W_i \sec z] \right\}$$

DO loop 1 ends

D-E Reflected diffuse radiation at lower cloud top level ICT1 is computed using

$$REASO = \alpha_{ICAT}\ ASO_{ICT1}$$

In case of clear sky albedo of surface is used (Also ICT1=1 in this case).

E-F DO loop 5 starts. The counter IL runs from lower cloud top to bottom of upper cloud. In this loop the upward reflected radiation, RSO, is computed. The optical depth is augmented by a factor 1.66

$$DW = 1.66\ (W_{ct} - W_i)$$

Then

$$RSO_i = REASO \cdot (1 - A[DW])$$

i.e.

$$RSO_i = s^a\ \ 1 - A[W_{ct} \sec z] \ \cdot \alpha_s$$

$$\cdot\ \ 1 - A[1.66(W_{ct} - W_i)]$$

Which is the upward diffuse radiation reaching level i.

F-G In case of clear sky or ICT1=1 the control transfers to calling routine. Otherwise, if high cloud

is present (INDEX=1) control goes to statement 10. If no high cloud but middle cloud is present, (INDEX=2), control goes to statement 11. If no high or middle cloud control passes to statement 12.

G-H

Statement 10. High cloud is present. Here the control comes after computation of solar radiation between high cloud top and model top have been complete. Search for the highest cloud base and store in ICB1.
If bottom of high cloud (ICHB) is equal to top of middle cloud (ICMT) then this level exists inside cloud hence ICB1 is changed to ICB1 = ICMB and also the index is changed to INDEX = INDEX+1 i.e. no computation between middle and high clouds are made.
If ICHB = ICMT and
ICMB = ICLT
this implies continuity of cloud layers and no level is outside the cloud between ICLB and ICHT therefore the ICB1 is changed to
ICB1 = ICLB
and index is increased by 1. The control is passed to statement 3.

H-I

Statement 11. Here the control comes only if high cloud is not existing and middle cloud is existing and computations between top of middle cloud and model top have been completed. The maximum cloud bottom height is searched and stored in ICB1. If both low and middle clouds exist and their extents are such that
ICMB = ICLT
then ICB1 = ICLB
and index is increased by 1. Control passes to statement 3.

I-J

Statement 12. Here the control comes only if high and middle clouds do not exist and computation between low cloud top and model top have been completed. In such case ICB1 = ICLB (bottom of low cloud) and then the control passes to statement 3.

J-K

Statement 3. DO loop 2 starts for computation of downward short wave flux within the clouds. The index IL runs from cloud bottom to top of contiguous cloud top.
If the case is of clear sky (ICT1 = 1) or the level ICT1 = 1 is reached the control passes to statement 2. Otherwise, equivalent

$$EQCW = 1.4 \times (P_i - P_{ICT1}) /200$$

Optical depth

$$DW = EQCW + 1.66\ (W_i - W_{ICT1})$$

and downward solar radiation is computed as

$$ASO = 1 - \alpha_c \times ASO_{ICT1} \times [1-A[DW]]$$

DO loop 2 ends.

K-L

ICB1 = ICHB then new cloud top height is searched among 1, ICLT, and ICMT and stored in ICT1. The high cloud top height is stored in ITOP.
If at this stage ICB1 = ICMB then new cloud top height is searched between 1 and ICLT and stored in ICT1 and ITOP is assigned ICMT.
If at this stage ICB1 = ICLB then ICT1 is assigned value = 1 and ITOP = ICLT. The index is increased by 1 and control is transferred to statement 100 to repeat computations up to this stage for next INDEX. Once INDEX > 3 the control returns to calling program.

## 6.5 Subroutine: TG

Abstract: This subroutine computes the surface temperature through an iterative process using the Newton Raphson's method. The balance is between longwave radiation, the sensible heating and the latent heating from the ground. This subroutine makes the radiation package consistent with the boundary layer dynamics. The earth's surface is treated as a black-body.

Called by: RADN

Subroutines called: SFXPAR

Input via common

| Variable | Meaning |
|---|---|
| U1 | Wind speed at ground (level 1) |
| U2 | Wind speed at surface (level 2) |
| T1 | Ground temperature |
| T2 | Air temperature |
| Q1 | Specific humidity at temperature T1 |
| Q2 | Specific humidity at temperature T2 |
| ZOMAX | 4000 x 0.01 = 40 metres. Maximum roughness height |
| XHT1 | Topography at a given point |
| ALBX | Albedo at a given point |
| RIB | Bulk Richardson number |
| Z1 | Height of the lower level in surface layer |
| Z2 | Height of the upper level in surface layer |

Input via arguments list

| | |
|---|---|
| T1000 | Ground temperature |
| Q1000 | Ground specific humidity |
| CP | Specific heat at constant pressure |
| II | x coordinate of the computation column |
| JJ | y coordinate of the computation column |

| | |
|---|---|
| KT | Time step count |
| RHO | Density of air |
| P1000 | Pressure at $\sigma$=1 i.e. PZ(1) |
| FLG | Longwave downward flux at ground |
| FSG | Shortwave downward flux at ground |

Output

| Variable | Meaning |
|---|---|
| TS | Ground temperature |
| QS | Ground specific humidity |
| FLGU | Upward flux of longwave radiation at ground |

Description of the subroutine

| Designator | Text |
|---|---|
| A-B | The Stefan-Boltzman constant is stored in variable SIGM and the ground temperature 'TS' is assigned value T1000 as the first guess. The variable VELT is assigned value U2 i.e. the upper level velocity in surface flux layer. INDEX is initialized as 1. The ground temperature 'TS' is to be obtained from the balance among the net radiation and the net soil heat fluxes at the earth surface under the assumption of zero heat capacity of ground. The downward flux of longwave radiation at ground 'FLG' and downward shortwave flux at ground 'FSG' are computed in calling routine 'RADN' in C.G.S. units (ergs sec/cm$^2$). These are converted into M.K.S. units (Watts/m$^2$) and their sum stored in variable<br>X2 = (FSG+FLG) x 0.001<br>Where 0.001 is conversion factor. Constant factor required in Tetan's formula is stored in X3<br>X3 = 25.22 x 273.16 |
| B-C | DO loop 100 starts. The counter IL is to run from 1 to 100 i.e. 100 iterations are assigned for Newton Raphson's method by which ground temperature $T_s$ is to be computed. To obtain the surface fluxes the routine SFXPAR is called to get fresh values of CH, CQ, VELT (=U2), CHP, CQP, $U^*$ ant $T^*$. Switch SWCH is assigned value = 1 and surface sensible and latent heat fluxes multiplication factors are computed using<br>$XS = \rho C_H C_p U_2$ |

$XL = \rho C_H L U_2$

Using Tetan's formula surface specific humidity is computed and stored in QG.

Intermediary variables $TX = TS^2$ is defined.

C-D

The sensible and latent heat fluxes depend upon differences $(T_{1000}-T_s)$ and $(q_{1000}-q_g)$ where $T_{1000}$ and $q_{1000}$ are air temperature and specific humidity and $T_s$, $q_s$ are ground temperature and specific humidity. The balance relation is

$$FTS = (FLG + FSG) - \alpha_s FSG - \sigma T_s^4$$

$$+ XS(T_{1000}-T_s) + XL(q_{1000}-q_g) = 0$$

The ground temperature $T_s$ is obtained using an iterative method (Newton Raphson's method). This minimizes the value of FTS. It is a 3 step method. For this, with the current guess value, the FTS and its derivative are computed. First the derivative is computed

$$FDTS = \frac{\partial FTS}{\partial T_s}$$

$$= 4\sigma T_s^3 - XS - XL \cdot q_q \frac{273.16x25.22-5.33xT_s}{T_g^2}$$

where $q_g = q_{sg} \cdot GW$

GW is ground wetness, $q_{sg}$ is saturation specific humidity at ground at temperature (current guess) $T_s$. FTS is computed using the formula given above with the help of intermediary variables A11, A22, A33 and A44, and stored in variable FTS.

D-E

The value TS is updated and stored in TS1

TS1 = TS - FTS/DFDTS

If the difference of previous guess and the updated value of $T_s$ is < 0.1 control passes to statement 101, otherwise TS is assigned new value TS1 and iteration continues. DO loop 100 ends.

E-F

After statement 100 CONTINUE. The control comes to this statement if convergence is not achieved. In such case FSG, FLG, FW, U2, T1000, Q1000, RHO, P1000, T1, Q1, XHT1, ALBX, Z1, Z2 are printed. The ground temperature is assigned value = T1000 and the control passes to statement 411 where warning is printed that convergence is not achieved in the routine.

F-G Statement 311. If ground temperature $T_s > 323$, $T_s$ is assigned upper limit value = 323, and corresponding ground specific humidity is obtained from Tetan's formula.
The value of 'U1', 'VELT', GRTEMP1', 'GRTEMP(NEW)', 'Q1000', 'QS', 'AIRTEMP', 'CH', 'CQ' are printed according to format 111.

G-H Sensible heat flux and moisture flux are computed using new values of $T_s$ and $Q_s$.

$$SEN = XS * (T1000-T_g)$$

$$CON = XL * (Q1000-Q_s)$$

The upward longwave flux at ground is computed

$$FLGU = -\sigma T_s^4$$

The control passes to calling program

## 6.6 Subroutine: CLDCOL

Abstract: In this subroutine the cloud index array CLD(N,8) for eight categories of cloud is modified in such a way that the cooling effect by longwave radiation within the cloud is eliminated. The only contribution to cooling, computed in array DTC in routine RADN, then comes from top and bottom of a cloud layer.

Called by: RADN

Calls: None

Input via common: None

Input via argument list

| Variable | Meaning |
|---|---|
| CLD | An array that flags the presence of a cloud configuration at a given layer. |
| COE | Array of weight factors for eight different configuration |

Output

| Variable | Meaning |
|---|---|
| CLD | Modified cloud flag array |

Description of subroutine

| Text | Designation |
|---|---|
| A-B | Define two variables<br>N1 = N-1<br>N2 = N-2<br>Where N=Number of layers in vertical |
| B-C | DO loop 100 starts to run the index JR over NM1 cloud configuration. DO loop 210 is also initiated to run the index KR over N layers. The array CLD for one configuration is stored temporarily in array CLDD. DO loop 210 closes. |
| C-D | DO loop 220 is initiated to run the index from 1 to N1.<br>KR1 = KR+1<br>Difference in CLD arrays at two consecutive values of KR is found and stored in CD. Absolute value of CO is found and stored in CO1. |

D-C

If absolute differences is < 0.1 both levels are outside cloud and the array CLD need not be modified, the control passes to statement 220. If difference CO < 0.0

level KR is within cloud. Then

CLDD(KR) = 1.0

If difference CO > 0.0 then

level KR1 is within cloud. Then

CLDD(KR1) = 1.0

| | | | | |
|---|---|---|---|---|
| K-1 | 0 | 0 | 1 | 1 |
| K | 0 | 1 | 0 | 1 |
| CO | 0 | 1 | -1 | 0 |

DO loop 220 closes.

E-F

DO loop 230 is initiated to run index KR from 1 to N levels. The array CLDD(KR) is stored in array CLD for given configuration JR. DO loop 230 closes.

DO loop 200 computations are repeated for NM1 configurations and DO loop 200 closes.

Control returns to calling routine.

## 6.7 Subroutine: EMTAB

Abstract: In this routine emissivity values as a function of optical depth, DW, in log scale is computed.

Called by: FLUX

Calls: None

Input via common: None

Input via argument list

| Variable | Meaning |
|---|---|
| DW | Optical depth |

Output

| Variable | Meaning |
|---|---|
| EMIT | Emissivity value corresponding to given DW |

Description of routine

A-B The emissivity is intialized as 0.0. If optical depth < 0.0 the control passes to statement 5 where this value is printed and the routine returns '0' value of emissivity. If optical depth is 0, then also '0' value is returned

B-C For values of DW > 0 interpolation of emissivity is caried out on log scale of DW.
Log of DW is computed and stored in WD
If the value of WD is higher then Wlog(1), the control passes to statement 2, otherwise EMIT = EPS(1) and the control returns to calling program

C-D Statement 2. DO loop 3 starts to run index IL from 1 to N1 to determine the two consecutive values of WLOG within which DW lies. At every step, the counter IL is stored in variable K. If the value of WD is outside the limits of consecutive WLONG's the DO loop continues.
DO loop 3 closes.

D-E Value of ∂E/∂W is interpolated on the log scale of W

$$\frac{\partial E}{\partial W} = DEMDW = \frac{\epsilon_{K+1} - \epsilon_K}{\log W_{K+1} - \log W_K}$$

Emissivity is computed

$$EMIT = \epsilon_K + (WD - \log W_K) \cdot \frac{\partial E}{\partial W}$$

Control returns to calling program.

## 6.8 Subroutine: ZETA1

Abstract: In this subroutine average zenith angle over a period of NH (=3) hours is computed. This is required in computation of shortwave radiation part of the radiation package.

Subroutines called: None

Input via COMMON: None

Input via argument list

| Variable | Meaning |
|---|---|
| SIND | Sine of declination angle |
| COSD | Cosine of declination angle |
| SINL | Sine of latitude angle |
| COSL | Cosine of latitude angle |
| NH | No. of hours on which average zenith angle is to be computed. |
| ZH | HOUR + DZHOUR |
| DGD | Earth's angular velocity |
| SL | Longitude at which zenith angle is computed. |
| RAD | Degree to radian conversion factor |
| PI | $\pi$ |
| II | x-coordinate of point at which zenith angle is to be computed |
| JJ | y-coordinate of point at which zenith angle is to be computed |

Output

| Variable | Meaning |
|---|---|
| COS | Cosine of zenith angle |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The zenith angle cos(z) is initialized as zero. Also |

$X1 = \sin(\delta)\sin(\phi_j)$
$X2 = \cos(\delta)\cos(\phi_j)$

The formula for zenith angle is

$$\cos(z) = \sin(\phi)\sin(\delta) + \cos(\phi)\cos(h)$$

The hour angle cos(h) is computed next. For this the following information is used.

DZHOUR = 12

which gives the noon hour at G.M.

DAY = IFIX(TIME/86400)

gives DAY as 0, 1, 2 etc.

HOUR = IFIX((TIME-DAY•86400)/3600)

From which ZZHOUR

is computed as

ZZHOUR = ZH = HOUR - DZHOUR

If ZZHOUR > 24 ZZHOUR = ZZHOUR-24.

Zenith angle is averaged over NH (in the present case, 3) hours. DO loop 1 starts to run the index IL over NH hours in step of 1 hour.

An average around the ZZHOUR (=ZH) is found. For this the following formula is used.

ZZ = ZH + IL - (NH/2 + 1)

which gives average of

ZH - 1, ZH and ZH + 1 hours.

Hour angle is computed at given longitude after converting the earth's angular velocity • hour and longitude in radians

TAGL = (ZZ x DGDY + SL) x RAD - PI

= (hour x $\Omega$ + longitude) x conversion factor - 3.14

Then

$\cos(z) = \cos(z) + \sin(\phi)\sin(\delta) + \cos(\phi)\cos(\delta)(TAGL)$

DO loop 1 closes.

Average zenith angle is computed. Lower limit of the angle is set equal to 0.01

6.9 Subroutine: FUNCTION FCN(X)

Abstract: In this function subprogram the computed function gives the equivalent amount of water vapor of the model cloud as the absorptivity function

Called by: SLR

Subroutines called: None

Input via common: None

Input via argument list

| Variable | Meaning |
|---|---|
| X | Path length for computing equivalent optical depth. $= W\mathrm{Sec}\zeta$ |

Output

| Variable | Meaning |
|---|---|
| FCN | Function giving equivalent optical depth |

Description of routine

| Designator | Text |
|---|---|
| A-B | The absorptivity function FCN is computed using the formula $FCN = A[W] = 0.271(W\mathrm{Sec}\zeta)^{0.303}$ Control returns to calling program |

7.1 Routine: CVHEAT

Abstract: This routine computes the cumulus convection contribution to heating and moistening rate of the model atmosphere. Kuo's cumulus parameterization scheme as modified by Krishnamurti making use of moistening parameter 'b' and mesoscale convergence parameter 'y' is employed. The parameters 'y' and 'b' are obtained through regression equations coefficients of which are derived from GATE observations.

Subroutines called: SSHEAT

Input variables via common:

1. PT(K): K=1(1)N: Pressure values at N $\tilde{\sigma}$ levels from lower to upper levels
   PT(K) = $\tilde{\sigma}_{KK} * p_s$ KK = N+1-K K=1(1)N

2. QQ(K): Specific humidity values at $\tilde{\sigma}$ levels

3. T(K): Temperature at $\tilde{\sigma}$ levels

4. SIG(K): NPI values of $\sigma$ (Layer boundaries)

5. SIGT(K): N values of $\tilde{\sigma}$ (Model layer)

6. PL(I,J): Log of surface pressure over the domain LxM

7. WBAR: Vertically averaged vertical velocity $\dot{\sigma}$

8. ZETA: Vorticity at 0.8 $\sigma$ level

Input variables via arguments list

1. WD11: $\dot{\sigma}\ \partial/\partial\sigma$ at lower $\sigma$ level of a layer

2. WD21: $\dot{\sigma}\ \partial/\partial\sigma$ at upper $\sigma$ level of the layer

Output variables

1. LCLF: Cloud top; level of vanishing buoyancy

2. LCONV(k): Index for instability. 1: Unstable; 0: Neutral; -1: Stable

3. HLC(k): Heating due to convection

4. QDIF(k): Difference of humidity between cloud and environment

5. HLS(k): Stable heating

6. LCLB: Cloud base

7. RAINF: Total convective rainfall in 24 hours

8. ISTAB: Index for presence of conditional instability

9. ETAIL: Working array

10. QS(k): Saturation specific humidity in cloud

11. PCONST(k): $\frac{\theta}{C_p T} = \frac{1}{C_p} \cdot \left[\frac{1000}{p}\right]^{R/Cp}$

12. THETA(k): Potential temperature

13. TCNVT(k): $\frac{T}{\theta} = \frac{p}{1000}^{R/Cp}$

14. TADV(k): $\dot{\sigma} \frac{\partial\theta}{\partial\sigma}$

15. TEMP(k): Environmental temperature from bottom layer to top layer of model

16. Q(k): Environmental specific humidity from bottom layer to top layer of model

17. CCONV: Moisture supply: $I_L$

Data

1. GRAV: Gravitational acceleration

2. RHC: Critical relative humidity

3. RHCRIT: Critical relative humidity for presence of cloud

4. CDT: Cloud time scale

5. RCP: $R/C_p$

6. EPS: 0.622

Desription of subroutine

| Designator | Text |
|---|---|
| A-B | Define various constants like gas constant (GASR), CPINV($1/C_p$), CDTINV(1/CDT), CD1(6.11 x 0.622), constants required in Tetan's formula (CD1, CD2, CD3, CD4, CD5), constants required in defining specific humidity and latent heating (CD6 and CD7) and constants for regression equation determining $\eta$ and b (AX1, AX2, AX3, BX1, BX2, BX3). Assign N1 = N-1. |

B-C Initialize various arrays; RAINF and in do loop 756 - HLC, QDIF, QADV, QSADV, TADV, FXY, ETAIL, FDQ, ISTAB for N $\tilde{\sigma}$ levels.

C-D DO loop 756. Determine w (≐ dp/dt) using $w = \dot{\sigma}\, p_s$ In this loop, also determine $DP_k = \Delta\tilde{\sigma}_{kk} \times p_s$ i.e. pressure differences between $\tilde{\sigma}$ levels and their inverses, $DPS_{k+1} = \Delta\sigma_{kk} \times P_s$ i.e. pressure differences between $\sigma$ levels and their inverses.

D-E Compute the conversion factors for T to $\theta$ and other multiplication factors.

$$TCONVT_k = \left[\frac{p_k}{1000}\right]^{R/Cp} = \frac{\tilde{T}_k}{\theta_k}$$

$$PCONST_k = \frac{1}{C_p} \cdot \frac{\theta}{\tilde{T}}$$

$$PTF_k = p^{-k}$$

in DO loop 10.

E-F In DO loop 20 determine the specific humidity ($QS_k$) using Tetan's formula relative humidity. Change the order of temperature and humidity (T, QQ) and assign to $TEMP_k$ and $Q_k$. Calculate potential temperature corresponding to temperatures $TEMP_k$.

F-G In DO loop 957 initialize the instability index at N $\tilde{\sigma}$ levels as '89'. Store the values of QS in QS1 and Q in Q1 and THETA in THETA1 after changing the order i.e. from top layer to bottom layer of model.

G-H In DO loop 765 compute the vertical advection saturation specific humidity QS1, specific humidity Q1 and potential temperature THETA as

$$\frac{\dot{\sigma}_k \frac{\Delta QS1_k}{\Delta\tilde{\sigma}_k} + \dot{\sigma}_{k+1} \frac{\Delta QS1_k}{\Delta\tilde{\sigma}_{k+1}}}{2} = QSADVS_k$$

etc.

In DO loop 766 reverse the order and store the advection values of $q_s$, q and $\theta$ in QSADV, QADV, TADV respectively.

H-I Determine the presence or otherwise of conditional instability in DO loop 30. For this find the averge vertical velocity w (= dp/dt) at $\sigma$ levels. If w is > 0 then control passes to statement 31 and $LCONV_k$ is assigned value -1 i.e. stable case. Otherwise compute the stability

. . .

$$\frac{\theta}{C_pT} L \cdot \dot{\sigma} \frac{\partial q}{\partial \sigma} - \dot{\sigma} \frac{\partial \theta}{\partial \sigma} = \dot{\sigma}\theta \left[ \frac{\partial L q_s}{\partial C_p \tilde{T} \sigma} + \frac{\partial \ln \theta}{\partial \sigma} \right] = x$$

For $x \geqslant 0$ LCONV(k) = 0 i.e. Neutral
$x < 0$ LCONV(k) = 1 i.e. Unstable
and ISTAB(k) = 1 (to be used in ).
Store the array LCONV in LCONVS.

I-J Initialize cloud base LCLB = 100. In DO loop 40 indexing K from 1 to N1 first check for instability; for existing instability PLCL = PT(k). Then check if relative humidity (RH(k) is > RHCRIT. If yes then assign T(k), Q(k), QS(k) to TLCL, QLCL and QSLCL respectively and compute RHLCL. If RHLCL is $\geqslant$ RHCRIT then LCLB is assigned value k. LCLFC level of free convection is assigned value = LCLB and if LCLB $\geqslant$ N1 the control goes to statement 400 implying convection is not invoked.

J-K In this section the moist adiabat is computed i.e. a covective cloud is defined. For this in DO loop 70 multiplication factor is computed using expression

$$-R/C_p \frac{L}{Cp} \times (p_k) \times \frac{0.622 \times 6.11}{p_k}$$

and stored in CCONV. And the cloud temperature array TS(K) is assigned the environmental temperature values (TEMP(K)). LCLBP is taken one level higher to bottom of the cloud and in DO loop 80 which indexes K from LCLBP through N1, cloud temperatures TS(K), are computed using Newton Raphson's method.
For this

$$F(TK) = -T_{K-1}\, p_{K-1}^{k} + T_K p_K^{-k} - \frac{L}{Cp} q_s(T_{K-1}) p_K^{-k} + \frac{L}{Cp} q_s(T_K) p_K^{-k}$$

$$\frac{DF}{dT_K} = \frac{L}{Cp} \frac{dq_s}{dT} p_K^{-k} + p_K^{-k}$$

At temperature TS(K), compute the cloud humidity QS(K) do loop 80 closes. Initialize top of cloud LCLT as LCLB. In DO loop 897, index K from LCLB+1 to N1. Compare TS(K) with environmental temperature and the last level up to which TS(K) $\geqslant$ TEMP(K) assign index K of that level to LCLT otherwise if TS(K) < TEMP(K), control passes to 890 and LCLT is compared with LCLB+1, if LCLT $\leqslant$ LCLB+1 convection is not invoked and control passes to statement 400.

K-L

Initialize available moisture supply (I) as CONV = 0. Compute

$$\Delta p = 1/[\{(p_{Bottom})^3 - (p_{Top})^3\} \cdot 100]$$

In DO loop 50 compute the vertically integrated moisture supply (convergence)

$$CONV = I = \int_{PLCLB}^{PLCLT} -q \frac{\partial q}{\partial \sigma} \Delta p \cdot \frac{100}{g}$$

If moisture supply is $\leqslant 0$ convection is not invoked and control goes to statement 400. Otherwise store CONV in CONVC.
Compute ETA($\eta$): the mesoscale convergence parameter and YB(b): the moistening parameter from the regression equations as
$\eta = (a_1 + b_1)\zeta + (a_2 + b_2)\bar{\omega} + (a_3 + b_3) - 1$
$b = (a_1\zeta + a_2\omega + a_3)/(\eta + 1)$
Set the upper and lower limits of $\eta$ as 0.5 and -0.1 and upper limit of YB as 0.15. Also compute
$\eta_{10}$ = ETA10 = $\eta/(1+\eta)$
Set lower limit of b as $\eta_{10}$. If b < 0 then set b = 0. Compute the total rainfall in 24 hours using RAINF = (1 - b) x (1 + $\eta$) x I x 816400.00 for use in PBLFLX routine.

L-M

Initialize XQT ($Q_\theta$), XQQ ($Q_q$) and XQCOOL as zero. Compute XXX = Cp/L$\Delta$t from cloud bottom (LCLB) to cloud top (LCLT). Compute moisture supply that modifies temperature

$$X = \frac{Cp}{L} \frac{T}{\theta} \dot{\sigma} \frac{\partial \theta}{\partial \sigma}$$

which is underlined part of $Q_\theta$

$$Q_\theta = \frac{1}{g} \int_{P_T}^{P_B} \frac{Cp}{L} \left[\frac{T_s - T}{\Delta t}\right] + \underline{\frac{Cp}{L} \frac{T}{\theta} \dot{\sigma} \frac{\partial \theta}{\partial \sigma}} \; dp$$

At cloud bottom vertical advection of $\theta$ is not allowed (x=0 at k=LCLB).

DTEMP = Cloud temperature
- Environmental temperature

DQ = Cloud specific humidity
- Environmental specific humidity

For all levels within cloud

$$FDQ(k) = \frac{(q_s - q)_k}{\Delta t}$$

$$FXY(u) = XY = \frac{Cp}{L} \frac{1}{\Delta t} \cdot (T_s - T)$$

Net cooling due to cumulus vertical transport of temperature is stored in XQCOOL as

$$XQCOOL = XQCOOL - \int_{P_T}^{P_B} \frac{Cp}{L} \frac{T}{\theta} \dot{\sigma} \frac{\partial\theta}{\partial\sigma}$$

In DO loop 832 compute $Q_\theta$ and $Q_q$ according to equations:

$$Q_\theta = \frac{1}{g} \int_{P_T}^{P_B} \frac{Cp}{L} \left[ \frac{T_s - T}{\Delta t} + \frac{T}{\theta} \dot{\sigma} \frac{\partial\theta}{\partial\sigma} \right] dp = XQT$$

$$Q_q = \frac{1}{g} \int_{P_T}^{P_B} \frac{(q_s - q)}{\Delta t} dp = XQQ$$

which are the moisture supply required to raise the large scale temperature to cloud temp, XQT, and moisture supply required to increase moisture to saturation at the cloud temperature, XQQ, respectively.

For a monitoring point (4,18) print $Q_\theta$ and $Q_q$. If $Q_\theta > 0$ control transfers to statement 93', otherwise, the partitioning factor for potential temperature, YAT, is assigned 0 value.

$a_\theta = YAT = 0$

and if $Q_q \leqslant 0$, convection is not invoked and control passes to statement 400², otherwise, the partitioning factor for specific humidity is calculated.

$a_q = YAQ = I * (1+\eta) * b/Q_q$

and control passes to statement 94.

Statement 93. If $Q_q > 0$ the control passes to statement 96 otherwise convection is not invoked and control passes to statement 400.

Statement 96.

$a_\theta = YAT = I \cdot (1+\eta)(1-b)/Q_\theta$

$a_q = YAQ = I \cdot (1+\eta) \cdot b/Q_q$

Statement 94

$X1 = a_q/\Delta t$

M-N In DO loop 200 the heating and moistening rate due to cumulus convection is computed according to

$$\frac{D\theta}{Dt} = -\frac{\partial q}{\partial\sigma} + a_\theta \left[ \dot{\sigma} \frac{\partial q}{\partial\sigma} + \frac{\theta}{T} \frac{(T_s - T)}{\Delta t} \right]$$

$$\frac{Dq}{Dt} = a_q \frac{q_s - q}{\Delta t}$$

Index k runs from cloud bottom to cloud top.
DTEMP = Cloud temperature - environmental temperature

$$X = a_\theta \frac{(T_s - T)}{\Delta t} \times \frac{1000}{p}^{R/C_p}$$

DQ = $q_{cloud} - q_{environment}$
heating due to cumulus convection, HLC is calculated using

$$HLC = x + \omega \frac{\partial \theta}{\partial p} \cdot \frac{p}{1000}^{R/C_p}$$

At cloud bottom, heating = zero
If heating < 0, heating = zero
moistening - QDIF(k)

$$= \frac{a_q}{\Delta t} \cdot (q_s - q) - \eta \cdot I \cdot g \cdot 3.0 \cdot p_k^2 \cdot \Delta p$$

where $\Delta p = 1/\left((p_{cloud\ bottom} - p_{cloud\ top})^3 \cdot 100\right)$
DO loop 200 closes and control passes to statement 300.

M-N Statement 400. When convection is not invoked, control comes to this statement then LCLF = LCLB. If LCLB ⩾ N1 LCLF = N1 and array LCONV is restored from LCONVS. If convection has been invoked the array LCONV has value 999 at this stage.

N-O Statement 300. Call SSHEAT which computes stable heating. Control transfers to calling routine, YCAL.

7.2 Subroutine: SSHEAT

Abstract: In this routine contribution due to non-convective clouds (stable heating) is computed. The computations are carried out only when the instability index LCONV(k) is zero.

Routines called: Function HST

Input via common

| Variable | Meaning |
|---|---|
| w(k): | NM1 values of vertical velocity on isobaric surfaces i.e. dp/dt |
| QS(N): | Saturation specific humidity |
| THETA(k): | N values of potential temperature at $\sigma$ levels |
| TADV(k): | Vertical advection of potential temperature |

Input via argument list

| Variable | Meaning |
|---|---|
| PT(k) | Pressure at $\sigma$ layers |
| LCONV(k) | Instability index |
| T(k) | Temperature at $\sigma$ layers |

Data input

| | |
|---|---|
| CP: | Specific humidity at constant pressure |
| R: | Gas constant |
| YL: | Latent heat of vaporization |
| EPS: | Multiplication factor for determination of specific humidity |

Output

| Variable | Meaning |
|---|---|
| HLS(k) | Stable heating |
| LCONV(k) | Instability index |

Description of subroutine

| Designator | Description |
|---|---|

A-B In DO loop 10 determine the vertical velocity at constant pressure (dp/dt) at NPI $\sigma$ surfaces by transfering array w(k) and shifting one level i.e. OME1(k) = w(k-1) etc. Upper and lower boundary level values are assigned '0'. In DO loop 130 find the average dp/dt at $\bar{\sigma}$ leels in array OME2(u).

B-C In DO loop 30 initialize stable heating array HLS(k) to zero. DO loop 40 starts, indexing k from 1 to N. Assign kk1 = N-k+1. If the stability index LCONV $\neq$ 0, stable heating at that level is not invoked and control passes to statement 40. Otherwise, if RH is LT1 then also control passes to 40. In case RH $\geqslant$ 1, function HST is called and the stable heating is stored in HLS(k) and the instability index is changed to '899'. DO loop 40 ends and control passes to calling routine.

7.3 Subroutine: Function HST

Abstract: This function routine returns stable heating at a point under the conditions

i) absolutely stable atmosphere

i.e. $-\frac{\partial\theta_e}{\partial p} > 0$ , or $-\frac{\partial\theta_e}{\partial p} > 0$

ii) The level in question must be saturated

iii) Upward vertical motion must exist

$w < 0$

the stable heating rate is measured from the time rate of change of specific humidity

$$H_{ST} = -L\frac{dq_s}{dt} = -Lw\frac{\partial q_s}{\partial p}$$

Routines called: None

Input via common

Input via argument list

| Variable | Meaning |
|---|---|
| QS: | Saturation specific humidity at $\tilde{\sigma}$ levels |
| T: | Temperature at $\tilde{\sigma}$ levels |
| AT: | Potential temperature |
| TAD: | Vertical advection of potential temperature |
| P: | Pressure at $\tilde{\sigma}$ levels |
| CP: | Specific heat at constant pressure |
| YL: | Latent heat of vaporization |
| R: | Gas constant |
| OME: | Vertical velocity w(= dp/dt) at $\tilde{\sigma}$ levels |

Output

| Variable | Meaning |
|---|---|
| HST | Stable heating rate |

Description of Routine

| Text | Designator |
|---|---|

A-B . Stable heating rate is computed

$$HST = - Lw \frac{\partial q_s}{\partial p}$$

where $\partial q_s/\partial p$ is given as

$$\frac{\partial q_s}{\partial p} = \frac{-(0.622)(6.11)}{p} \times \exp \frac{a(T - 273.16)}{T - b}$$

$$\left[ \frac{1}{p} - \frac{a}{T - b} - \frac{T - 273.16}{(T - b)^2} \cdot \frac{RT}{Cp\,p} (1 + 0.61 q_s) - \frac{L}{Cp} \frac{\partial q_s}{\partial p} \right] = \frac{C_5 C_6}{1 + \frac{L}{Cp} C_3 C_5}$$

$$C_5 = \frac{-0.622 \times 6.11}{p} \exp \frac{a(T - 273.16)}{T - b}$$

$$C_6 = \frac{1}{p} - C_3 \left[ \frac{RT}{C_p p} (1 + 0.61 q_s) \right]$$

$$C_3 = \left[ \frac{a}{T - b} - \frac{a(T - 273.16)}{(T - b)^2} \right]$$

Where a and b are the constants of Tetan's formula

7.4 Subroutine: SHACON

Abstract: In this routine the contribution of shallow convetive clouds to temperature and moisture flux divergence are computed.

Called by: PBLFLX

Subroutines called: None

Input via common

| | |
|---|---|
| Q(k) | Specific humidity at N $\bar{\sigma}$ levels from bottom to top. |
| PZ(k) | Pressures at NP1 $\sigma$ levels. |
| PT(k) | Pressure at N $\bar{\sigma}$ levels |
| TT(k) | Temperature at NP1 $\sigma$ levels. |
| TEMP(k) | Temperature at N $\bar{\sigma}$ levels, from bottom to top |

Input via argument list

| Variable | Meaning |
|---|---|
| FXQ | Moisture flux at 1 level below PBL top |
| LCLP | Level 1 below the top of boundary layer. |
| LCLP1 | Top of the boundary layer |
| NPH | Number of levels of shallow convection computation |

Output

| | |
|---|---|
| SCUM | Moisture flux divergence due to shallow convection at NPH levels. |
| SCUH | Temperature flux divergence due to shallow convection at NPH levels. Where NPH comes via argument list |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Define<br>$FCP = R/C_p$<br>In DO loop 10 potential temperature at N $\bar{\sigma}$ levels are computed<br>$\theta = T \cdot \left[\frac{1000}{p}\right]^{R/C_p}$ |

B-C DO loop 20. The moisture flux WDQDP and temperature flux HDTDP are initialized as zero at N $\sigma$ levels. The parameter FK (diffusion coefficient) is initialized as zero.

C-D Number of levels between top of boundary layer up to top of model atmosphere is stored in KK1. Density of air at top of the boundary layer, where pressure and temperature are PZ(LCLP1) and TT(KK1) respectively is determined.

D-E A vertical moisture gradient at the top of boundary layer is determined and stored in XXLCL

$$XXLCL1 = \rho^2 LCL \; g \cdot \frac{\partial q}{\partial q}$$

The value of diffusion coefficient at top of the boundary layer is determined and stored in FK

$$FK = \frac{\text{flux at 1 level below PBL top}}{L} \cdot \frac{1}{\zeta^2 LCL \cdot g \cdot \frac{\partial q}{\partial q}}$$

If the moisture flux FK < 0.0 shallow convection contribution is not computed and control returns to calling program.

E-F Upper limit of FK is set = 25. For every 6th time step, FK and FXQ are printed for monitored point.

F-G DO loop 30. In DO loop 30 the temperature and moisture fluxes are computed between top of PBL up to upper limit of shallow convection i.e. level 4

$$KK1 = NP1 - K+1$$

$$\zeta^2 = \frac{100 \cdot PZ_k}{RT}^{\;2}$$

$$X1T = -C_p \zeta^2 gk \; \frac{p}{1000}^{\;k}$$

$$= - C_p \zeta^2 gk \; \frac{T}{\theta}$$

is temperature flux at level k.
$S1Q = -\zeta^2 gk$ L moisture flux
The temperature and moisture fluxes are computed and stored in HDTDP and WDQDP

$$HDTDP = X1T \cdot \frac{\partial \theta}{\partial P}$$

$$= -C_p \zeta^2 gk \frac{T}{\theta} \frac{\partial \theta}{\partial P} = F_\theta$$

$$WDQDP = X1Q \frac{\partial q}{\partial P}$$

$$= -L\zeta^2 gk \frac{\partial q}{\partial P}$$

G-H DO loop 40. The index K runs from PBL top to shallow convection upper limit (4). Moisture and temperature fluxes are computed and stored in SCUM and SCUH

$$SCUM = -\frac{\partial F}{\partial P}\frac{g}{C_p} \qquad SCUH = -\frac{g}{L}\frac{\partial F}{\partial P}$$

Control transfers to calling program.

## 8.1 Subroutine: PBLFLX

Abstract: In this subroutine the sensible heat, latent heat and momentum surface fluxes are vetically distributed in the planetary boundary layer according to profiles based on stability of the layer. If rainfall in the PBL is > 5 min. the layer is defined as undisturbed. The profiles of momentum, sensible heat and latent heat fluxes are in the form of structure functions derived from GATE observations.

Called by: YCAL

Subroutines called: SHACON

Input via common

| | |
|---|---|
| FSX | Sensible heat flux at surface |
| FLT | Latent heat flux (moisture flux) at surface |
| RAINF | 24 hr. covective rainfall, computed in CVHEAT |
| PZ | Pressure values at $\sigma$ levels from bottom to top |
| ISTAB(k) | Stability index, determined in CVHEAT |

Input via argument list

| | |
|---|---|
| LCLN | LCL level, determined in routine LCLSUR |

Input via data

| | |
|---|---|
| ETAP | Structure function for moisture (Q) in case of undisturbed PBL condition at N $\sigma$ levels. From bottom to top $\eta_{qu}$ |
| ETAX | Structure function for moisture (Q) in case of disturbed PBL condition at N $\sigma$ levels. From bottom to top $\eta_{qd}$ |
| EPSP | Structure function for temperature (T) in case of disturbed PBL condition at N $\sigma$ levels. From bottom to top $\eta_{TD}$ |
| ETSP | Structure function for temperature (T) in case of undisturbed PBL condition at N $\sigma$ levels. From bottom to top $\eta_{TD}$ |

Output

| Variable | Meaning |
|---|---|
| QFLX(k) | The moisture flux divergence at N $\sigma$ levels |

| | |
|---|---|
| TFLX(k) | The temperature (or sensible heat) flux divergence at N $\tilde{\sigma}$ levels |
| SCUM(k) | Moisture flux due to shallow convection |
| SCUH(k) | Sensible heat flux due to shallow convection |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Normalization to GATE moisture and heat structure functions under disturbed and undisturbed boundary conditions. The flux at a level is given as $$F = F_o \left[ 1 - \frac{1}{\Delta p} \int_p^{p_o} \eta(p)dp \right]$$ where $\eta$ can be ETAP, ETAX, EPSP or ETSP. For example $$\eta(p) = \frac{Q_2 - Q_2{}^k}{\bar{Q}_2 - \bar{Q}_2{}^k}$$ In this equation $Q_2$ and $\bar{Q}_2$ are from Ooyama and $Q_2{}^k$ and $Q_2{}^k$ are from FSU diagnostic model.<br>For normalization of various fluxes variables SUM, SUM1, SUM, and SUM3 are initialized to zero.<br>In DO loop 130 vertically averaged $\eta$ is found.<br>In DO loop 120 the flux for each of the N layers are determined<br>$\tilde{\eta} = \eta/\bar{\eta}$ such that $\bar{\tilde{\eta}} = 1$<br>Thus, ETAP, ETAX, EPSP and ETSP at this stage represent normalized fluxes of moisture and heat. |
| B-C | RCP = R/CP<br>The variable LCLP is intialized as LCLF (level of free convection). The upper limit of LCLP is set = N. |
| C-D | In DO loop 10 the arrays<br>GG1 (temperature structure function)<br>GG2 (moisture structure function)<br>FX1 (temperature flux)<br>FX2 (moisture flux)<br>are initialized to zero.<br>In DO loop 20 arrays QFLX, TFLX, UFLX, VFLX, SCUM and SCUH are initialized to zero. Their meaning is explained in OUTPUT. |
| D-E | Determination of top of planetary boundary layer. The maximum allowed height of PBL LPBMAX = 3 |

The height of PBL

LPBL = LCLN

where LCLN is the 1 level of condensation obtained in routine LCLSUR. If LCL is at a level 5 or above, height of PBL is restricted to

LPBL = LPBMAX

Two additional variables are defined

LCLP1 = LPBL giving height of planetary boundary layer.

LCLP11 = LCLP1 - 1 i.e. one level below the top of PBL

Pressure at LCL level is intialized as

PLCL = PZ(LCLP1)

The multiplying factors for sensible heat flux, moisture flux and momentum flux are defined

ALPHAS = 1.0

ALPHAL = 1.03

ALPHAM = 1.00

These values are used in equations

$$F = F_o \frac{P - \alpha P_{LCL}}{P_o - \alpha P_{LCL}}$$

The values of the $\alpha$'s indicates that momentum and temperature fluxes are zero at PLCL and moisture flux is zero at a level slightly lower than LCL.

E-F

Defining disturbed or undisturbed PBL.

If the surface flux of moisture is negative it is set to zero.

If rainfall obtained in CVHEAT > 5 mm control passes to statement 11, otherwise to statement 21.

F-G

Statement 11. Disturbed PBL. In DO loop 30, the temperature an moisture fluxes at NP1 $\sigma$ levels are determined. The fluxes from the solid lower boundary are taken as zero. The quantities computed are

$$G = \frac{1}{\Delta p} \int_p^{p_o} \eta(p)dp$$

GG1 → temperature flux

GG2 → moisture flux

In DO loop 40 the surface fluxes of temperature and moisture are distributed vertically, using

$$F = F_o \left[ 1 - \frac{1}{\Delta p} \int_p^{p_o} \eta(p)dp \right]$$

FX1 → temperature flux

FX2 → moisture flux

$$F = F_o \frac{0.54}{p_o - p_{LCL}} p_{LCL} + \frac{0.46 p_o - p_{LCL}}{p_o - p_{LCL}}$$

$= 0.46F_o$

G-H DO loop 50. The temperature and moisture flux divergence are computed at N $\sigma$ levels using $\frac{g}{C_p}\frac{\partial F}{\partial P}$ and $\frac{g}{L}\frac{\partial F}{\partial P}$ respectively and stored in TFLX and QFLX respectively. The control passes to statement 101.

H-I Undisturbed planetary boundary condition. R.F $\leqslant$ 5.0 mm. Based on GATE observations it is possible to express the fluxes by a linear relation.

$$F = F_o(ap+b)$$

where $p_o > p \geqslant p_{LCL}$

| flux | a | b |
|---|---|---|
| Momentum | $\frac{0.54}{p_o - p_{LCL}}$ | $\frac{0.46p_o - p_{LCL}}{p_o - p_{LCL}}$ |
| Sensible heat | $\frac{3.50}{p_o - p_{LCL}}$ | $\frac{-2.5p_o - p_{LCL}}{p_o - p_{LCL}}$ |
| Latent heat | $\frac{-2.0}{p_o - p_{LCL}}$ | $\frac{3p_o - p_{LCL}}{p_o - p_{LCL}}$ |

At $p = p_{LCL}$

$F_m = 0.46F_{mo}$

$F_T = -0.25F_{To}$

$F_q = 3F_{qo}$

$$\frac{\partial F}{\partial p} = \frac{F_o}{\Delta p} \qquad \Delta p = (p_o - \alpha p_{LCL})$$

DO loop 60. The fluxes of moisture at NP1 levels are computed from the GATE structure function and stored in GG2.

In DO loop 70 the vertical distribution of temperature and moisture fluxes is computed

Temperature flux using

$$Fx1 \equiv F = FSX * \left[\frac{p - \alpha p_{LCL}}{p_o - \alpha p_{LCL}}\right]$$

and moisture flux, as in disturbed case

$$Fx2 = FLT * \left[1 - \frac{1}{\Delta p}\int_{p_o}^{p} \eta(p)dp\right]$$

I-J Computation of flux divergence of temperature. In DO loop 80 temperature flux divergence is computed using

$$\frac{g}{C_p}\frac{\partial F}{\partial p}$$

$\frac{C_p \partial P}{}$

and stored in TFLX array from surface to 1 level from top of planetary boundary layer. A multiplication factor 0.01 appears on right hand side of statement 80 because of conversion from mb to pascal.

J-K

Conditions for shallow convection are defined.
(i) The boundary layer is undisturbed
(ii) Moisture flux at the surface is positive
(iii) LCL $\leqslant$ 5. Level 5 is the upper limit of shallow convection
(iv) Instability index at 1 level below PBL top is 1 i.e. conditional instability exists at that level. If these conditions are not met, shallow convection is not invoked and control transfers to statement 901. Otherwise,

FXQ = FX2(LCLP11)

i.e. flux of moisture at level 1 below PBL top is stored in FXQ. Shallow convection routine is called to determine its contribution to moisture and temperature flux divergence.

K-L

In Do loop 100 the moisture flux divergence is computed in the boundary layer.
In DO loop 110 the contributions of shallow convection to temperature flux divergence (SCUH) and to moisture flux divergence (SCUM) are added and the resultant values are stored in TFLX and QFLX at N $\sigma$ levels. The control passes to statement 101.

L-M

Statement 201. The control comes here if PBL is undisturbed and shallow convection is not invoked. In DO loop 90 the temperature and moisture flux divergence are computed at N $\sigma$ levels using

$$\frac{g}{C_p}\frac{\partial F}{\partial p} \quad \text{and} \quad \frac{g}{OL}\frac{\partial F}{\partial P}$$

respectively.

M-N

For a monitored point, and every 6th time step vertically integrated temperature and moisture fluxes are computed from lowest $\sigma$ levels up to N $\sigma$ levels in DO loop 140.

$$SUM0 = \sum_{k=1}^{N} \frac{C_p}{g} \cdot 100 \cdot TFLX \cdot (\Delta P)$$

and

$$SUM1 = \sum_{k=1}^{N} \frac{C_p}{g} \cdot 100 \cdot TFLX \cdot (\Delta P)$$

The parameters
KT, LCLP11, ISTAB(LCLP11), SUM, SUM1, FLT and FSX are printed.

N-O

Computations of momentum flux. DO loop 150. The intermediary arrays FX1 and FX2 are initialized for x and y momentum fluxes.

$$XX1 = 1/(P2(1) - P_{LCL})$$

In DO loop 160 the momentum fluxes are computed

$$XX = (PZ_k - P_{LCL}) \cdot XX1$$

$$FX1(k) = FMX \cdot XX$$
$$FX2(k) = FMY \cdot XX$$

In DO loop 170 the x and y momentum flux divergences are computed and stored in UFLX and VFLX respectively. Control returns to calling program, YCAL

## 8.2 Subroutine: LCLSUR

Abstract: In this subroutine lifting condensation level in a given column is determined.

Called by: YCAL

Subroutines called: SPHUM

Input via common

| Variable | Meaning |
|---|---|
| Q1000 | Ground specific humidity |
| PL(I,J) | Log of surface pressure |

Input via arguments list

| | |
|---|---|
| T2 | Temperature at surface |

Output

| Variable | Meaning |
|---|---|
| LCLN | Lifting condensation level |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | For computation of LCL the beginning is made at surface. The pressure temperature and humidity values are assigned surface values and stored in variables<br>PLCLP = $e^{PL}$ = surface pressure<br>TLCLP = $T_2$ = Air temperature<br>QLCLP = Q1000 = surface humidity<br>Potential temperature at suface is computed and stored in THLCLP |
| B-C | A count, K, is initialized<br>K = N<br>Pressure, temperature and sat. specific humidity at $\sigma_N$ level are found and stored in PNEXT, TNEXT and QSN.<br>In case<br>QLCLP > QSN<br>i.e. humidity of parcel is > higher level control passes to statement 21 as saturation is reached. Otherwise,<br>K = K-1<br>If counter K=1 the control passes to statement 21, |

otherwise to statement 11 for checking saturation specific humidity value at next $\sigma$ level, against specific humidity of parcel starting from surface.

C-D

Statement 21.

LCLN = NP1 - K+1

The LCL is computed and stored in LCLN. The control transfers to calling program.

## 8.3 Subroutine: SFLX

Abstract: In this subroutine surface fluxes of sensible heat, latent heat and momentum are computed via stability dependent bulk aerodynamic formulation. The roughness parameter, $Z_0$, is computed from Charnock's formula. In the computation of the friction velocity, U*, the drag coefficient, $C_D$, is made a function of the wind speed of the surface layer, over the ocean. Over land the roughness parameter is made a function of terrain height.

Called by: YCAL

Subroutines called: SFXPAR

Input via COMMON: None

Input via argument list

| Variable | Meaning |
|---|---|
| U1 | Surface velocity in m/s (level 1) |
| U2 | Air speed in m/s (level 2) |
| T1 | Potential temperature at surface (level 1) |
| T2 | Air potential temperature (level 2) |
| Q1 | Specific humidity value of surface corresponding to temperature T1 (level 1) |
| Q2 | Specific humidity value of air corresponding to temperature T2 (level 2) |
| ZOMAX | Maximum roughness length. In the present case its value is 40 cm. |
| XHT1 | Terrain height in metres |
| ALBX | Albedo of surface at a given point (II,JJ) |
| II | x coordinate of given point |
| JJ | y coordinate of given point |
| KT | Time step |
| RHO | Density of air at surface |
| KRAD | Time step for radiation computation |
| IUM | x coordinate of point of maximum U component |

| | |
|---|---|
| JUM | y coordinate of point of maximum U component |
| IVM | x coordinate of point of maximum V component |
| JVM | y coordinate of point of maximum V component |
| GW | Ground wetness parameter |

Data input

| Variable | Meaning |
|---|---|
| G | Acceleration due to gravity |
| VKC | von Karmann constant |
| CP | Specific heat at constant pressure |
| HL | Latent heat of evaporation |
| ALPHA | Charnoch's parameter |

Output

| Variable | Meaning |
|---|---|
| $C_D$ | Drag coefficient for momentum (Bulk transfer stress coefficient) |
| $C_H$ | Bulk transfer sensible heat flux coefficient |
| $C_Q$ | Bulk transfer latent heat flux coefficient |
| OL | Monin Obukhov length |
| USTR | Frictional velocity |
| TSTR | Frictional temperature |
| RIB | Bulk Richardson number |
| FSX | Sensible heat flux |
| FLT | Latent heat flux |
| FMX | x - Momentum flux |
| FMY | y - Momentum flux |

Description of subroutine

| Designator | Text |
|---|---|

A-B The parameters INDX, CHP (slope of bulk transfer sensible heat flux coefficient) and CHQ (slope of bulk transfer latent heat flux coefficient) are initialized.

INDX = CHP = CHQ = 0

For land surface computation of roughness parameter, $Z_0$, the control passes to statement 10. (ALBX > 0.06)

B-C Ocean surface computation of $Z_0$ Height of level $Z_2$ is initialized as

$Z_2 = 10$

Drag coefficient, $C_D$, is initialized as

CDO = 0.0011

The drag coefficient is made a function of wind speed. For $U_2 < 5.8$ CD = CDO

$-5.8 \leqslant U_2 \leqslant 16.8$ CD = CDO x $(0.74-0.046xU_2)$

$U_2 > 16.8$ CD = CDO x $(0.94+0.034xU_2)$

First guess of friction velocity. U* is determined

USTRSQ = CD*U2**2

Using Charnock's formula first guess of roughness parameter $Z_0$ is obtained

$Z_0 = MU*^2/g$

The lower limit of $Z_0$ is set = 0.0001. The level 1 height $Z_1$ is assigned value $Z_0$

$Z1 = Z\phi$

If Z1 ⩾ Z2 Z2 = Z2+Z1

Iteration count is initialized as 0. Iterative process of dermining final value of $Z_0$ is initiated with statement 12 where subroutine SFXPAR is called to provide next guess of U* using which $Z_0$ is determined. The values of $Z_1$ and $Z_2$ are reassigned, the lower limit of $Z_0$ is always kept as 0.0001. The iterative process is repeated 4 times. Once the final value of $Z_0$ is obrained for the given point on ocean the control transfers to statement 15.

C-D Determination of roughness parameter, $Z_0$, on land. Statement 10. When the given point is a land point, the roughness parameter, $Z_0$, is determined as a function of terrain height XHT1.

$$Z_0 = 0.15 + 0.2 \times (236.8+18.42xXHT1) ** 2.0 \times 1.0 \times 10^{-8} \text{ metres}$$

$$Z_1 = Z_0$$
$$Z_2 = 100.$$

D-E Statement 15. Using the final value of $Z_0$, $Z_1$ and $Z_2$ the drag coefficients $C_D$, $C_H$, $C_Q$ are determined by callling the routine SFXPAR

E-F Surface fluxes for sensible heat (FSX), latent heat (FLT), X momentum flux (FMX), y momentum flux (FMY) are obtained.

$FSX = \rho C_H C_p U_2 (T_2 - T_1)$

$FLT = \rho G W C_{DL} (Q_2 - Q_1)$

$FMX = \rho C_D U_2$

$FMY = \rho C_D U_2$

For time steps other than radiation time step, the control passes to statement 1001

F-G For radiation time steps, at the monitored points, the following variables are printed

CH, CQ, CD, Z1, Z2, Z0

FSX, FLT, U1, U2, T1, T2, Q1, Q2

Also for maximum surface wind point

CH, CQ, CD are printed under heading SFLX Max.

G-H Statement 1001. Control transfers to calling program

8.4 Subroutine: SFXPAR

Abstract: This subroutine computes the transfer coefficients of sensible latent and momentum fluxes. The subroutine returns the parameters USTR, TSTR, OL, CD, CH, CQ, RIB, RIM described under output.

Called by: SFLX, TG

Subroutines called: None

Input via common: None

Input via arguments list

| Variable | Meaning |
|---|---|
| U1 | Velocity at ground (level 1) |
| U2 | Velocity at level 2. |
| T1 | Potential temperature at level 1 |
| T2 | Potential temperature at level 2 |
| Z1 | Height of level 1 |
| Z2 | Height of level 2 |
| II | x coordinate of given point |
| JJ | y coordinate of given point |
| CHP | Slope of parameter CH |
| CQP | Slope of parameter CQ |
| INDX | |

Input via data statement

| Variable | Meaning |
|---|---|
| G | gravity |
| VKC | von Karmann constant |
| ALPHA | Charnock's parameter |

Output

| Variable | Meaning |
|---|---|

| | |
|---|---|
| USTR | Friction velocity |
| TSTR | Friction temperature |
| RIB | Bulk Richardson number |
| RIM | Modified Bulk Richardson number |
| B | Buoyancy parameter |
| DZL | Stability parameter for stable or neutral case |
| Z2L | Stability parameter for unstable case |
| CD | Bulk transfer stress coefficient |
| CH | Bulk transfer sensible heat flux coefficient |
| CQ | Bulk transfer sensible heat flux coefficient |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Determination of Bulk Richardson number. For this Buoyancy parameter B is defined as $B = \frac{G}{T}$ where $T = \frac{(T_1+T_2)}{2}$ The difference in heights, temperatures and speed of two levels in constant surface flux layer are determined as $\Delta Z = DZ = Z_2 - Z_1$ $\Delta U = DU = U_2 - U_1$ $\Delta T = DT2T1 = T_2 - T_1$ Richardson number is obtained $R_{iB} = RIB = \Delta T \cdot B \cdot \frac{\Delta Z}{(\Delta U)^2} = \frac{g}{\theta} \frac{\Delta\theta}{(\Delta U)^2}$ A multiplication factor $ETA = \ln(Z_2/Z_1)$ is also defined. For unstable case (RIB < 0) control passes to statement 20. For stable case (RIB ⩾ .212) control passes to statement 10. |
| B-C | Stable or neutral case $0 < RIB < 0.212$ The stability parameter, DZL, is computed using the formula $\frac{\Delta Z}{L} = \ln\frac{Z_2}{Z_1} \left[ \frac{9.4RIB - 0.74 + \sqrt{4.88RIB + 0.5476}}{9.4 - 44.18RIB} \right]$ |

Monin Obukhov length, L, is determined

$$OL = \frac{DZ}{DZL} = \frac{\Delta Z}{\Delta Z/L} = \frac{\Delta ZKB\theta*}{U*^2}$$

Using the stability parameter (Bulk Richardson number RIB), the friction velocity and temperature are determined

$$U* = k \frac{(\bar{U}_2 - \bar{U}_1)}{\ln\frac{Z_2}{Z_1} + 4.7\frac{\Delta Z}{L}}$$

$$\theta* = k \frac{(\bar{\theta}_2 - \bar{\theta}_1)}{0.74 + \ln\left[\frac{Z_2}{Z_1}\right] + 4.7 \cdot \frac{\Delta Z}{L}}$$

Bulk transfer stress coefficient, $C_D$, is determined

$$CD = \frac{K^2}{\ln\left[\frac{Z_2}{Z_1}\right] - 4.7\frac{\Delta Z}{L}^{\ 2}}$$

C-D

Exchange coefficient of moisture for stable and neutral case. If INDX = 1 derivative of CQ is to be computed, control passes to statement 701. This is the case when this routine is called by routine 'TG'. When routine is called by SFLX

$$CH = \frac{-K^2}{\ln\left[\frac{Z_2}{Z_1}\right] + 4.7\frac{\Delta Z}{L} \quad 0.74\ln\left[\frac{Z_2}{Z_1}\right] + 4.7\frac{\Delta Z}{L}}$$

If drag coefficient < $0.5\times10^{-3}$ the moisture exchange coefficient is not computed and control passes to statement 10. Otherwise, CQ = CH/1.7 and control passes to statement 703

D-E

Statement 701. The control comes to this statement in case of stable and neutral situation and when derivatives of CH and CQ are to be obtained (i.e. INDX = 1)

$$CH = \frac{-K^2}{\ln\left[\frac{Z_2}{Z_1}\right] + 4.7\frac{\Delta Z}{L} \quad 0.74\ln\left[\frac{Z_2}{Z_1}\right] + 4.7\frac{\Delta Z}{L}}$$

If the exchange coefficient CH < $0.2\times10^{-3}$ this implies moisture flux is downward and the control passes to statement 702. If CH ⩾ $0.2\times10^{-3}$ the positive moisture flux exists and CQ = CH/1.7
In case of stable and neutral value of Richardson number RIB CHP = CQP = 0, and the control returns to calling program.

E-F

Statement 10. Extreme stable case (RIB $\geqslant$ 0.212). No transfer of fluxes.
INDX = 1 Derivatives are to be computed and therefore, control passes to statement 702 otherwise USTR = TSTR = 0
OL = 999
Lower limit of CD is defined
CD = $0.5 \times 10^{-3}$
CH = -CD/(7•RIB)
CQ = CH
Control passes to statement 704
CH from

$$CH = \frac{K^2}{0.74 \ \ln\left[\frac{Z_2}{Z_1}\right] - \psi_2 \quad \ln\left[\frac{Z_2}{Z_1}\right] - \psi_1}$$

Also CQ = CH/1.7
and the control transfers to statement 706.

I-J

Statement 705. Control comes to this statement when INDX = 1 Derivatives of CH and CQ are to be computed.
CH and CQ are computed as in section H-I.
Derivatives of the stability parameter Z2L is obtained w.r.t. $\theta$

$$Z2LP = -e^y \ \Delta y$$

$$= -Z2L \cdot [A1 + 2A_2X + 3A_3X^2]$$

$$\cdot \frac{1}{RIM} \cdot \frac{B \ Z_2}{(U_2-U_1)^2}$$

Similarly derivatives of E1, E2, G1, G2 PS1 and PS2 are obtained

$$E1P = 3.75 \times Z2LP \ / \ (Z_2/Z_1) \ / \ E1^3$$

$$E2P = -3.75 \times Z2LP \ / \ E2^3$$

$$G1P = -4.5 \times Z2LP \ / \ (Z_2/Z_1) \ / \ G1$$

F-G

Statement 702. Extreme stable case. Control comes to this statement when INDX = 1 Derivatives of CH and CQ are to be computed.
Lower limit of CH is set
CH = $-0.2 \times 10^{-3}$
CQ = CH/1.7
CHP = CQP = 0.
Control passes to calling program.

G-H

Statement 20. Unstable case. RIB < 0. Modified Richardson number RIM is computed

$$RIM = RIB \cdot Z_2/\Delta Z$$

$$= \left[ \frac{B(\theta_2-\theta_1)\Delta Z}{(U_2-U_1)^2} \right] \frac{Z_2}{\Delta Z}$$

In unstable case, the stability parameter, Z2L is given as

$$Z_2L = \frac{Z_2}{L}$$

$$= RIM \left[ \frac{[\ln Z_2/Z_1) - \psi_1(Z_2/L,\ Z_2/Z_1)]^2}{[0.74[\ln(Z_2/Z_1) - \psi_2(Z_2/L,\ Z_2/Z_1)]]} \right]$$

where

$$\psi_1(Z_2/L,\ Z_2/Z_1) = \ln\left[ \frac{1+\epsilon_2}{1+\epsilon_1}^2 \cdot \frac{1+\epsilon_2^2}{1+\epsilon_1^2} \right]$$

$$- 2\tan^{-1}\epsilon_2 + 2\tan^{-1}\epsilon_1$$

and

$$\psi_2(Z_2/L,\ Z_2/z_1) = 2\ln \frac{1+\gamma_2}{1+\gamma_1}$$

$$\epsilon_1 = (1 - 15Z_2/L\ /\ Z_2/Z_1)^{1/4}$$

$$\epsilon_2 = (1 - 15Z_2/L)^{1/4}$$

$$\gamma_1 = (1 - gZ_2/L\ /\ z_2/Z_1)^{1/2}$$

$$\gamma_2 = (1 - gZ_2/L)^{1/2}$$

Equation for $Z_2/L$ is solved by a polynomial least square fitting method.

$$y = a_0 + a_1X + a_2X^2 + a_3X^3$$

where $y = \ln(-Z_2/L)$
and $x = \ln(-RIM)$
The coefficients $a_0$, $a_1$, $a_2$, $a_3$ are functions of $Z_2/Z_1$ only. These have been obtained by CHANG (1978). The stability parameter Z2L is computed from y.

$$Z2L = -e^y$$

Monin Obukhov length is determined

$$OL = Z_2/Z2L$$

Then the parameters required for solution of equation for Z2L are computed
E1 = $\epsilon$1, E2 = $\epsilon$2, G1 = $\gamma$1, G2 = $\gamma$2, $\psi$1 = PS1 and $\psi$2 = PS2 according to formulae given above.

H-I When INDX = 0 derivatives of CH and CQ are not required to be computed. U* = USTR is obtained from equation

$$\frac{k(U_2-U_1)}{U*} = \ln\left[\frac{Z_2}{Z_1}\right] - \psi_1 \frac{Z_2}{L} \frac{Z_2}{Z_1}$$

and $\theta*$ = TSTR from

$$\frac{k(\theta_2-\theta_1)}{0.74\theta*} = \ln\left[\frac{Z_2}{Z_1}\right] - \psi_2 \frac{Z_2}{L} \frac{Z_2}{Z_1}$$

CD from

$$CD = \frac{K^2}{\left[\ln\left[\frac{Z_2}{Z_1}\right] - \psi_1\right]^2}$$

and

G2P = -4.5xZ2LP/G2

Derivatives of $\psi 1$ and $\psi 2$ are given as

Derivatives of Bulk exchange coeficient are computed

$$CHP = CH* \frac{\partial\psi_2}{\frac{Z_2}{Z_1} - \psi_2} + \frac{\partial\psi_1}{(\eta-\psi_1)}$$

CQP = CHP/1.7

The control passes to calling program.

9.1 Subroutine: FORCST

Abstract: In this subroutine the forecast variables UU, VV, ZZ, WW, QQ, T are calculated and written on the output file 97.

Subroutines called: Function SPHUM

Input via COMMON

| Variable | Meaning |
|---|---|
| UU(L,M,N) | U component of wind |
| VV(L,M,N) | V component of wind |
| PP(L,M,N) | Model parameter P |
| WW(L,M,N) | Vertical velocity at $\sigma$ surfaces ($\sigma$) |
| QQ(L,M,N) | Specific humidity |
| PL(L,M) | Log of surface pressure |
| SIGTK(N) | N values of multiplication factor $\frac{\sigma_k}{R\ (\sigma_{k+1} - \sigma_k)}$ |
| TBB(NP1) | Area average temperature at NP1 $\sigma$ layer boundaries |
| SIGT(N) | N values of $\sigma$ layers |

Output

| Variable | Meaning |
|---|---|
| UU(L,M,N) | U-component of wind |
| VV(L,M,N) | V component of wind |
| PP(L,M,N) | Geopotential heights |
| WW(L,M,N) | Vertical velocity $\dot{\sigma}$ |
| QQ(L,M,N) | Relative humidity |
| PL(L,M) | Log of surface pressure |
| WORK1(L,M,NP1) | Temperature at $\sigma$ layer |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | U and V components of wind are written on unit 997 for N $\sigma$ layers as separate records |
| B-C | Temperature at N $\sigma$ layers are calculated in DO loop 40 and30 using relation $$-\frac{\dot{\sigma}}{R}\frac{\partial p}{\partial \sigma} - R\frac{\partial T^*}{\partial \sigma}\ln p_s$$ and written on unit 97 as N separate records. |
| C-D | In DO loop 60 and inner loop 50 the geopotential heights are calculated using relation $$(P - R * T^* \ln p_s)/g$$ and written on unit 97 as separate N records |
| D-E | DO loop 70. Surface pressure is retrieved from log of surface pressure, $p_L$. Then in DO loop 90 and inner do loop 80, saturation specific humidities corresponding to layer temperature and pressure are calculated. From these and model predicted specific humidities Q, the relative humidities in $\sigma$ layers are calculated and written on unit 97 as N records |
| E-F | DO loop 100. Vertical velocity $\dot{\sigma}$ are written on unit 97 as NM1 records. DO loop 100 closes. |
| F-G | Surface pressure values are written on unit 97 as 1 record.<br>The control passes to the calling program |

## 9.2 Subroutine: RAINFAL

Abstract: In this routine the rainfall amounts due to convection, large scale dynamic ascent, super saturation and surface flux of moisture and heat are written as separate records on mass storage file 98. Then the arrays representing these quantities are initialized.

Called by: FSULAM

Subroutines called: None

Input via common

| Variable | Meaning |
|---|---|
| RAINCV(#I,#J) | Convective rainfall |
| RAINLS(#I,#J) | Large scale rainfall |
| RAINSU(#I,#J) | Super saturation rainfall |
| SURFLX(#I,#J) | Surface flux of moisture |
| TEMFLX(#I,#J) | Surface flux of heat |

Output

| Variable | Meaning |
|---|---|
| RAINCV(#I,#J) | Convective rainfall |
| RAINLS(#I,#J) | Large scale rainfall |
| RAINSU(#I,#J) | Super saturation rainfall |
| SURFLX(#I,#J) | Surface moisture flux |
| TEMFLX(#I,#J) | Surface heat flux |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Arrays RAINCV, RAINLS, RAINSU SURFLX and TEMFLX are written on mass storage file 98. In DO loop 10 the above arrays are initialized to zero. Control passes to the calling program. |

9.3 Subroutine: CHECK

Abstract: In this routine average quantities of some of the output fields are calculated to monitor the performance of the model. This helps in checking the model in case of trouble. It is called at the starting time, after first time step and every 3 hours after that as well as at the forecast time.

Called by: Main program FSULAM

Subroutines called: None

Input via COMMON

| Variable | Meaning |
|---|---|
| UU(L,M,N) | U component of wind |
| VV(L,M,N) | V component of wind |
| PP(L,M,N) | Geopotential Height Z |
| WW(L,M,N) | Vertical velocity $\sigma$ |
| QQ(L,M,N) | Relative humidity |
| PL(L,M) | Log of surface pressure |
| RAINCV(L,M) | Convective rainfall |
| RAINLS(L,M) | Large scale rainfall |
| RAINSU(L,M) | Supersaturation rainfall |
| SURFLX(L,M) | Surface moisture flux |
| TEMFLX(L,M) | Surface temperature flux |
| RAD2(L,M) | Temperature change due to radiation |
| TTFLUX2(L,M,N) | Temperature change due to PBL flux of temperatue |
| UFL(L,M,N) | x component of momentum flux |
| VFL(L,M,N) | y component of momentum flux |
| ALSSU(L,M,N) | Moisture change due to supersaturation adjustment |
| DADJ(L,M,N) | Temperature change due to dry convective adjustment |
| QQFLUX2(L,M,N) | Moisture change due to PBL moisture flux |

CVH2(L,M,N) Heating due to convective rainfall

ALS2(L,M,N) Heating due to supersaturation adjustment

CVQ2(L,M,N) Moistening due to cumulus convection

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Heading 'FROM CHECK' is printed. Then second heading TIME STEP. UU, VV, QQ, WW, PP is printed. Time step KT is also printed in second heading. DO loop 10 is initiated to run index K from 1 to NP1. In this loop average u component v component, moisture content in layers, vertical velocity $\sigma$, geopotential height PP are calculated and printed for each level. In this do loop total precipitable water is also calculated in SUM5. |
| B-C | Statement 31. After this statement in DO loop50 average convective rain, large scale rain, supersaturation rain, surfae moisture flux and surface temperature flux are calculated and printed according to format 41. In DO loop 170 average surface temperature is calculated and printed in Format 171. In DO loop 70 average log of surface pressure is calculated and printed in format 51. |
| C-D | DO loop 80. In this DO loop temperature geopotential heights of $\sigma$ levels are computed and average of these are calculated and printed. $$T_k = -\delta_k \cdot \frac{\Delta P}{\Delta\sigma} - R \cdot \frac{\Delta T}{\Delta\sigma} \cdot \ln p_s$$ and $$Z_k = \frac{(P - R*T^* * \ln p_s)}{g}$$ |
| D-E | DO loop 150. First, the title RAD, TFLUX, QFLX, CVHEAT, LARGE, CVQMDI is printed. Then the average quantities of<br>(i) Temperature change due to radiation<br>(ii) Temperature change due to PBL temperature flux<br>(iii) x momentum flux<br>(iv) y momentum flux<br>(v) Moisture change due to supersaturation adjustment<br>(vi) Temperature change due to dry convective adjustment<br>(vii) Moisture change due to PBL moisture flux |

(viii) Heating due to convective rainfall
(ix) Heating due to supersaturation rainfall
(x) Moistening due to cumulus convection

are calculated for all $\sigma$ layers and printed according to format 151. Control returns to calling program FSULAM.

10.1 Subroutine: BSM1

Abstract: In this routine a 5 point smoother is applied to quantity at a given point.

Subroutines called: None

Input via COMMON: None

Input via arguments list

| Variable | Meaning |
|---|---|
| F(L,M,N) | Given variable array which is required to be smoothed |

Output

| Variable | Meaning |
|---|---|
| F(L,M,N) | Smoothed array |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Outer DO loop 100 is initiated to run index k from 1 to N. Inner DO loop 10 is initiated to run indx J from 2 to MM1 and I from 2 to LM1. The given variable array F is smoothed using nearest 5 points in two steps. $A = F_{i-1,j,k} + F_{i+1,j,k} + F_{i,j-1,k} + F_{i,j+1,k} + 4F_{i,j,k}$ $Work_{ij} = 0.125A$ DO loop 100 closes |
| B-C | In DO loop 20 the values of array WORK(I,J) are stored in array F(I,J,K). DO loop 20 ends. DO loop 100 ends. Control passes to calling routine |

## 10.2 Subroutine: .BSMOOTH

Abstract: In this routine the forecast variables are subject to smoothening minimum in the inner domain and smoothening progressively increasing from 5 points inside boundaries towards the boundaries.

Subroutines called: None

Input variables via common

| Variable | Meaning |
|---|---|
| DX(M) | M values of grid interval in E-W direction as function of latitude. |
| DY | Grid interval in N-S direction |

Input variable via argument list

| Variable | Meaning |
|---|---|
| F(L,M,N1) | The variable to which smoother is to be applied |
| A1 | Smoother for points I•EQ•1 or I•EQ•L or J•EQ•1 or J•EQ•M |
| A2 | Smoother for points I•EQ•2 or I•EQ•L-1 or J•EQ•2 or J•EQ•M-2. |
| A3 | Smoother for points I•EQ•3 or I•EQ•L-2 or I•EQ•3 or J•EQ•M-2. |
| A4 | Smoother for points I•EQ•4 or I•EQ•L-3 or J•EQ•4 or J•EQ•M-3. |
| A5 | Smoother for points I•EQ•5 or I•EQ•L-4 or J•EQ•5 or J•EQ•M-4. |
| A6 | Smoother for the inner domain i.e. I > 5 or I < 4 or J >5 or J < M-4. |
| N1 | Number of vertical levels. |

Output variables

| Variable | Meaning |
|---|---|
| F(L,M,N1) | Smoothed forecast variable. |

Description of subroutine

A-B Main DO loop 11 is set up to index K from 1 to N1 levels in vertical. In the inner loop of 12,

values are assigned as
JP1 = J+1; at J=M, JP1=M
JM1 = J-1; at J=1, JM1=1
IP1 = I+1; at I=L, IP1=L
IM1 = I-1; at I=1, IM1=1
The smoother is then applied and the smoothed values are stored in array work.

$$WORK(I,J) = (1-4.0A)F_{i,j} + AF_{i+1,j} + F_{i,j+1} + F_{i-1,j} + F_{i,j-1}$$

The coefficients A take value as described in 'INPUT'. DO loop 12 closes.

B-C DO loop 11 is initiated for J to index through 1 to M and I to index through 1 to L. In this DO loop contents of WORK are transferred to F(I,J,U). DO loop 11 ends. Control transfers to calling program.

10.3 Subroutine: CALPHI

Abstract: In this subroutine the horizontal grid intervals $\Delta x(M)$ and $\Delta y$ are defined on a longitude-latitude grid. Coriolis parameter is determined and vertical sigma ($\sigma$) and sigma hat ($\tilde{\sigma}$) levels are defined besides defining the time step, $\Delta t$.

Subroutines called:

Input variables via common:

Input variables via arguments list:

| Variables | Meaning |
|---|---|
| PHI1 | Southern-most latitude of the model domain |
| PHI2 | Northern-most latitude of the model domain |
| DELAM | Grid interval in E-W direction in terms of degrees longitude. |

Output Variables

| Variables | Meaning |
|---|---|
| DT | Time step in seconds |
| DLAM | Grid interval in E-W direction in terms of degrees longitude |
| DPHI | Grid interval in N-S direction in terms of degrees latitude |
| DX(M) | The grid interval in E-W direction as function of latitude |
| COR(M) | Values of coriolis parameter as function of latitude |
| DY | Grid interval in N-S direction |
| SIG(NP1) | NP1 values of $\sigma$ levels |
| SIGT(N) | N values of $\tilde{\sigma}$ levels |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Through data statement the values of lower boundary ($\sigma_{N1} = 1.0$) and upper boundary ($\sigma_1 = 0.1$) values |

are defined. Time step $\Delta t$ is defined. The N-S grid interval is computed and stored in DPHI

B-C DO loop 11 is set up to index J from 1 to M. Values of latitudes representing N-S grid are obtained and stored in PHI. Using this DX(J) and COR(J) are computed as

$DX(J) = DLAM \times 111.1 \times 10^3 \times \cos(\phi_j \times PI/1800)$

$COR(J) = 2\sin(\phi_j \times PI/1800);\ J = 1(1)M$

DO loop 11 ends

The North south grid interval DY is computed and stored in DY.

C-D DO loop 12 is initialized to index k through 1 to N. In this DO loop NP1 values of $\sigma$ levels and N values of $\tilde{\sigma}$ levels are obtained. Loop 12 closes.

D-E The values DT, DLAM, DPHI, and DY are printed. DX, PHI, COR, SIG and SIGT are written on tape 96 in formatted form (10E12.6)

10.4 Subroutine: COFF

Abstract: In this routine the coefficient for smoothing are assigned to array ASM depending upon the distance of given point from the boundary.

Subroutines called: None

Input via COMMON: None

Input via arguments list

| Variable | Meaning |
|---|---|
| L | Maximum number of grid points in E-W |
| M | Maximum number of grid points in N-S |

Output

Variable

| | |
|---|---|
| ASM(L,M) | Array of smoothig coefficients |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The following smoothing coefficients are defined.<br>A6 = 0.0006<br>A5 = 0.002<br>A4 = 0.0075<br>A3 = 0.015<br>A2 = 0.03<br>A1 = 0.06 |
| B-C | DO loop 100. For grid points 5 points away from boundary the smoothing coefficient (SC) ia A6.<br>For grid points 4 points away from boundary the SC is A5<br>For grid points 3 points away from boundary the SC is A4<br>For grid points 2 points away from boundary the SC is A3<br>For grid points 1 point away from boundary the SC is A2<br>For grid points at boundary the smoothing coefficient is A1<br>The smoothing coefficients are stored in array ASM. The DO loop 100 closes. The control passes to calling program. |

10.5 Subroutine: CONST

Abstract: In this routine various constants required for the model are defined. Solar declination angle and the incident solar radiation at the top of the atmosphere is also computed here.

Subroutines called: None

Input variables via common:

| Variables | Meaning |
|---|---|
| NDAY | Starting day of the forecast |
| MON | Month |
| DYPHI | Grid interval in N - S direction in degress latitude. |
| SLAT | Southern most latitude of domain |

Output variables:

| Variable | Meaning |
|---|---|
| RAD | Radius of earth: $6.37122 \times 10^6$ meters. |
| G | Accelaration due to gravity 9.81 $msec^{-2}$ |
| R | Gas constant 287 $J \circ K^{-1} Kg^{-1}$ |
| CP | Specific heat at constant pressure 1004 $J \circ K^{-1} Kg^{-1}$ |
| YL | Latent heat of vaporization at 0°C $2.5 \times 10^6$ $J\ Kg^{-1}$ |
| GINV | 1/G |
| PI | Value of $\pi$; 4.0 x ATAN (1.0) |
| RADC | Conversion factor for degree to radius conversion $\pi/180$. |
| GAM | Dry convective lapse rate g/Cp = °K/meters |
| ALBAIR | Albedo of air = 1.0 |
| ROMEGA | Angular velocity of earth in °/hr. $\frac{360°}{24}$ /hr. |
| RS | Solar radiation incident at the top of the atmosphere = $\frac{1.935 \times 4.2 \times 10^7}{60 \times d^2}$ |

$= 1.38\ Wm^{-2}$

| | |
|---|---|
| DELTA | Solar declination angle |
| SIND | Sine of angle of solar declination |
| COSD | Cosine of angle of solar declination |
| SINPHI(M) | Sine of latitude |
| COSPHI(M) | Cosine of latitude |

Factor

| | |
|---|---|
| SIGMA | Stefan - Boltzmann constant($\sigma$)<br>$= 5.6678 \times 10^{-5}\ erg\ cm^{-2}\ sec^{-1}\ deg^{-4}$<br>$= 5.6696 \times 10^{-2}\ Wm^{-2}\ deg^{-4}$ |

RTOP: Not used

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Various constants as described under 'OUTPUT VARIABLES' are defined. JUN22, the day of maximum angle of declination and JULY1, the day of farthest distance of earth from sun are defined. Earth's orbit's eccentricity is defined as 0.0178 |
| B-C | From the given value of NDAY and MONTH the number of the days from 1 January is computed in NDAY. Its difference from JUN22 is stored in ADAY and from JULY1 in BDAY. Maximum declination is computed in radians as<br>DECM = (23 + 27/60) • RADC |
| C-D | Square of earth - Sun distance is computed using<br>$\rightarrow$ $\sqrt{\Omega \geqslant \Delta\Omega\Pi}^{6}\ \left(1 + eccn \cdot \cos \frac{2 \cdot PI \cdot BDAY}{365}\right)^2$<br>and using this the solar irradiance incident at the top of the atmosphere is computed<br>RS = 1.935 x 4.2 x $10^7$/60/ESDIST<br>where 1.935 is solar constant in ly, 4.2 x $10^7$ is conversion factor from calories to erg and division by 60 converts to $sec^{-1}$. |
| D-E | Declination angle is calculated<br>DELTA = $\delta$ = (Max. declination) • $\cos(\frac{2\pi ADAY}{365})$<br>from this cosine and sine of declination angles are computed. |

| | |
|---|---|
| E-F | Using SLAT and DYPHI, latitude values ($\phi$'s) at grid points in N-S direction are computed and corresponding cos $\phi$'s and sin $\phi$'s are computed |
| F-G | Some more constants are defined and the control returns to the calling program |

10.6 Subroutine: EIGEN

Abstract: This subroutine computes the eigen values and eigen vectors of the coefficient matrices of the Helmholtz equation for model parameter 'P' as given below

$$AP + \frac{\Delta t^2}{4} BP = Q$$

Using the Rutishauser method.
This equation is written in a finite difference form as

$$\frac{1}{\delta\sigma_K} \frac{P_{K+1} - P_K}{\tilde{\gamma}_K \Delta\tilde{\sigma}_K} - \frac{P_K - P_{K-1}}{\gamma^*_{K-1} \Delta\sigma_{K-1}} - \frac{\Delta t^2}{4} \nabla^2 P_K = Q_K$$

where $\delta\sigma_K = \tilde{\sigma}_K - \tilde{\sigma}_{K-1}$
$\Delta\tilde{\sigma}_K = \sigma_{K-1} - \sigma_K$

Matrix A has the following elements

$$A_{11} = - \frac{1}{\tilde{\gamma}_1^*} \Delta\tilde{\sigma}_1 \cdot \frac{1}{\delta\sigma_2}$$

$$A_{12} = \frac{1}{\tilde{\gamma}_1^*} \frac{1}{\Delta\tilde{\sigma}_1} \frac{1}{\delta\sigma_2}$$

$$A_{1,NP1} = \frac{\sigma_1}{RT^*_{NP1}} \frac{1}{\delta\sigma_2}$$

$$A_{i,i-1} = \frac{1}{\gamma^*_{i-1}} \frac{1}{\Delta\tilde{\sigma}_{i-1}} \frac{1}{\delta\sigma_i}$$

$$A_{i,i} = \frac{-1}{\tilde{\gamma}_i^*} \frac{1}{\Delta\tilde{\sigma}_i}$$

$$A_{i,i+1} = \frac{1}{\gamma_i^*} \frac{1}{\Delta\tilde{\sigma}_i} \frac{1}{\delta\sigma_i}$$

$$A_{NP1,N} = \frac{1}{\gamma_N^*} \frac{1}{\Delta\tilde{\sigma}_N} \frac{1}{\delta\sigma_N}$$

$$A_{NP1,NP1} = - \frac{1}{\gamma_N^*} \frac{1}{\Delta\tilde{\sigma}_N} + \frac{1}{RT_S^*} \frac{1}{\delta\sigma_N}$$

Matrix B has the following elements

$B_{11} = 0.25$, $B_{12} = 0.25$, $B_{NP1,N} = 0.25$, $B_{NP1,NP1}$ 0.25,
$B_{K,K-1} = 0.25$, $B_{K,K} = 0.5$, $B_{K,K+1} = 0.25$

The forcing function $Q_K$ of the Helmholtz equation can be expressed as

$$Q_k \equiv \frac{\Delta t}{2} \nabla \cdot w + \frac{\partial w_3}{\partial \sigma} - w_5$$

therefore we have the boundary values ($\dot{\sigma}=0$, $\dot{\nabla} \cdot \tilde{W}=0$)

$$\frac{P_2 - P_1}{\tilde{\gamma}_1^* \, \Delta\tilde{\sigma}_1} = w_{31}$$

$$\frac{P_{N+1} - P_N}{\tilde{\gamma}_N^* \, \Delta\tilde{\sigma}_N} + \frac{P_{N+1}}{RT_s^*} = w_{3N+1} + \frac{gz_s}{RT_s^*}$$

The f.d. equation and the matrix equation represent

$$\frac{\partial}{\partial \sigma} \frac{1}{\gamma} \cdot \frac{\partial P}{\partial \sigma} - \frac{\Delta t^2}{4} \nabla_H^2 P = \frac{\Delta t}{2} \nabla \cdot \tilde{w} + \frac{\partial \tilde{w}_3}{\partial \sigma} - \tilde{W}_5 \equiv Q$$

Also

$$A\underline{P} + \frac{\Delta t^2}{4} \nabla^2 B\underline{P} = \underline{F}$$

or $\underline{P} + \frac{\Delta t^2}{4} \nabla^2 A^{-1} B\underline{P} = A^{-1}\underline{F}$

Let $A^{-1}B = C = TCT^{-1} = \Lambda$
or $C = T^{-1}\Lambda T$

Then

$$CV_k = \Lambda_k V_k \qquad k=1(1)N+1$$

$$\underline{P} + \frac{\Delta t^2}{4} \nabla^2 T^{-1}\Lambda T\underline{P} = A^{-1}F$$

Premultiply this above equation with transformation matrix T.

$$T\underline{P} + \frac{\Delta t^2}{4} \nabla^2 \Lambda T\underline{P} = TA^{-1}F$$

Let $\underline{Q} = T\underline{P}$, $TA^{-1} = \underline{E}$

Then

$$\underline{Q} + \underline{\Delta t^2} \nabla^2 \Lambda \underline{Q} = E\underline{F}$$

$$CV = \Lambda V$$

$$C = V^{-1}\Lambda V$$

$$T = V^{-1}$$

$$\Lambda = V^{-1}\Lambda V = TCT^{-1}$$

Subroutine called: INVERT, MULT, RUTISH

Input variables via common:

| Variables | Meaning |
|---|---|
| SIG(NP1) | NP1 values of $\sigma$ levels. |
| SIGT(N) | N values of $\tilde{\sigma}$ levels. |
| GAMAB(N) | N values of lapse rate $\gamma^*$ at $\tilde{\sigma}$ levels. |
| TBB(NP1) | NP1 values of area average temperature at $\sigma$ levels. |
| R | Gas constant |

Output variables

| Variables | Meaning |
|---|---|
| EGN(NP1) | NP1 values of eigen values. These correspond to NP1 $\sigma$ levels. |
| TRANS(NP1, NP1) | Inverse eigen vector martix $V^{-1}$ |
| EE(NP1, NP1) | Matrix $V^{-1}A^{-1}$ |
| TINV(NP1, NP1) | Eigen vector matrix V |
| AINV(NP1, NP1) | Matrix $A^{-1}$ |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | The variable FACTOR is initialized as 1.0 and in DO loop 11 array A(NP1, NP1) is initialized as zero. Elements of array are obtained in accordance with relations given in abstract at lower and upper boundary levels and in between levels in DO loop 12. Matrix A is written on unit 96 according to format 10E 13.6. |
| B-C | In DO loop 86 $\underline{A}$ is normalized (or scaled) by dividing it By 'FACTOR'. In DO loop 71 the matrix $\underline{A}$ is temp-stored in R4 and subroutine INVERT is called to compute inverse of R4, in R4. In DO loop 72 $A^{-1}$(R4) is stored in AINV. AINV is written on unit 96. Routine MULT is called to obtain |

multiplication of A and $A^{-1}$ to give unit matrix in 'R1' which in turn is written on unit 96. (10E 13.6).

C-D In DO loop 21 all elements of matrix B (NP1, NP1) are initialized. Then the non-zero elements are assigned their respective values, before and in DO loop 22. The matrix B is written on unit 96, product $A^{-1}B$ is found in matrix 'C' by calling 'MULT' and the matrix C is written on unit 16 (10E 13.6).

D-E Subroutine 'RUTISH' is called for computation of eigen values and corresponding eigen vectors of matrix C i.e. $A^{-1}B$ in EGN and TINV. A working array AA is initialized to zero in loop 61 [K = 1(1)NP1]. In DO loop 62 [J = 1(1)NP1] for every value of K, a sum of product of eigen vectors AA = AA + TINV(J,u) • TINV(J,u) is found. DO loop 62 ends. DO loop 61 is initialized, in which the normalized eigen vectors are obtained [J = 1(1)NP1]. Normalized eigen vectors are written on unit 96. In DO loop 63 and 64 sum of square of eigen vectors of every column of normalized eigen vectors is determined and written on unit 96. Eigen values EGN are then written on unit 96.

E-F Routine 'MULT' is called to get product C*TINV = $A^{-1}B\underline{V}$ in R1. R1 is then written on unit 96. In DO loop 83 all elements of matrix R1 are initialized to zero and in loop 84, the diagonal elements of R1 are assigned the eigen values EGN. Routine 'MULT' is called to obtain product TINV*Λ = VΛ in R2, which in turn is written on unit 96.

F-G In DO loop 53 the eigen vector matrix is temporarily stored in R4. Invert is called to find inverse matrix $R_4^{-1}$ (i.e. $V^{-1}$ i.e. T) in R4. In loop 54, the contents of inverted matrix are transferred to matrix TRANS, which is written in unit 96.

G-H Product of TINB(V) and TRANS($V^{-1}$) is found by calling 'MULT' in R1. A unit matrix R1 is obtained that is written on unit 96. Product $V^{-1}A^{-1}B$ is found in R1. Product $V^{-1}A^{-1}BV$ is found in R2. R2 represents eigen vector matrix and is written on unit 96. Product $V^{-1}A^{-1}$ is found in matrix EE by calling MULT and EE is written on 96. Products EE*A = $V^{-1}A^{-1}A$ and products $V^{-1}A^{-1}AV$ are found which give unit matrix R2. Unit matrix is written on unit 16.

H-I In DO loop 85 [k = 1(1)NP1] the eigen values are scaled by multiplying each of them with FACTOR. DO

loop 85 is initiated for counter I to index through 1 to NP1. Within inner 85 loop the vectors $A^{-1}$ and $EE = V^{-1}A^{-1}$ are scaled by dividing these by FACTOR. The DO loop 85 ends and the control is transferred to Calling routine.

## 10.7 Subroutine: ELLIPT

Abstract: In this subroutine the Helmholtz equation for the model parameter p is cast in a form suitable for computation, then the subroutine 'ITERATE' is called to solve it. The Helmholtz equation to be solved is

$$\frac{\Delta t^2}{4} \nabla^2 P + \frac{\partial}{\partial \sigma} \frac{1}{\gamma} \cdot \frac{\partial P}{\partial \sigma} \equiv Q$$

which can also be written as

$$A\underline{P} + \frac{\Delta t^2}{4} \nabla^2 B\underline{P} = \underline{F}$$

or $\underline{P} + \frac{\Delta t^2}{4} \nabla^2 A^{-1}B\underline{P} = A^{-1}\underline{F}$

or $\underline{P} + \frac{\Delta t^2}{4} \nabla^2 T^{-1}\Lambda T\underline{P} = A^{-1}F$

where $A^{-1}B = C = T^{-1}\Lambda T$

or $T\underline{P} + \frac{\Delta t^2}{4} \nabla^2 \Lambda T\underline{P} = TA^{-1}F$

Let $Q = T\underline{P}$, $TA^{-1} = E$

Then

$$\underline{Q} + \frac{\Delta t^2}{4} \nabla^2 \Lambda \underline{Q} = E\underline{F} \qquad \text{(A)}$$

At boundary points $Q=E\underline{F}$ or $P=A^{-1}\underline{F}$

$T = V^{-1}$, $T^{-1} = V$, $\Lambda = V^{-1}CV = TCT^{-1}$

Subroutines called: ITERATE

Input variables via common

| Variable | Meaning |
|---|---|
| EE(NP1, NP1) | Matrix $V^{-1}A^{-1}$ $(=TA^{-1})$ where V is eigen vector matrix |
| AINV(NP1, NP1) | Matrix $A^{-1}$ |
| TRANS(NP1, NP1) | Inverse eigen vector matrix $V^{-1}$ |
| TINV(NP1, NP1) | Eigen vector matrix $T^{-1}$ or V. |

Input variable via argument list

| | |
|---|---|
| P(L,M,NP1) | The model parameter P given as $P = gz_s - RT^* \ln p_s$ |
| F(L,M) | Log of surface pressure $p_s$ |

Output variables

| Variables | Meaning |
|---|---|
| P(L,M,NP1) | The model parameter P at time $t+\Delta t$. |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | DO loop 11 is set up to index J through 1 to M and I through 1 to L, for the domain excluding boundary points, DO loop 32 is set up to index K through 1 to NP1 vertical level. For every level a working array TT(K) is initialized to zero and in inner DO loop 32 for counter KK = 1(1)NP1 a sum $TA^{-1}F$ is found and stored in TT(K). For boundary points in DO loop 13 a sum $\sum A^{-1}*F$ is found and stored in TT(K). In DO loop 20 these values TT(K) are transferred back to array P. Which is now actually $TA^{-1}F$. |
| B-C | In DO loop 14 a sum $\sum_{kk=1}^{NP1} V^{-1}P$ $\quad T_{k,kk}P_{kk}$ $\quad$ K=1(1)NP1, KK=1(1)NP1 is found for each of NP1 values of counter K and stored in array TT(K). In the inner 11 DO loop these values are transferred to array P which now contains total forcing terms of Helmholtz equation (A). DO loop 11 ends. |
| C-D | In DO loop 15 log of surface pressure is read from unit 92 and routine iterate is called to solve Helmoltz equation. The values of variable TP(i.e. $V^{-1}P$) obtained from ITERATE are written on unit 92 and also stored in array GNEXT. |
| D-E | In DO loop 21 and inner loop 22, the values of sum of product $VV^{-1}P$ is found for every level and ultimately the values of model parameter are transferred to array P from array TT(K). The control returns to the calling program. |

10.8 Subroutine: INVERT

Abstract: In this subroutine inverse of a given matrix D(N,M) is computed and the inverse is returned in the array D(N,M). The method used is Gauss Jordan.

Subroutines called: None

Input variables via common: None

Input variables via agrument list

| Variable | Meaning |
|---|---|
| D(N,M) | Matrix which is required to be inverted. |

Output variables

| Variable | Meaning |
|---|---|
| D(N,M) | Inverse matrix |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | By standard Gauss-Jordan method the inverse of given matrix D is obtained and returned in D itself. |

## 10.9 Subroutine: ITERATE

Abstract: In this routine the Helmholtz equation in P is solved by iteration. The equation that is solved is

$$Q + \frac{\Delta t^2}{4} \nabla^2 \lambda \underline{Q} = E\underline{F}$$

where

$$Q = T\underline{P},\ TA^{-1} = E,\ T^{-1} = V,\ T = V^{-1}$$

where V is eigen vector matrix.

Subroutines called: None

Input variables via common

| Variable | Meaning |
|---|---|
| DT | Time step in seconds |
| EGN(NP1) | NP1 eigen values corresponding to NP1V levels of the model. |
| DX(m) | Grid distance in E-W direction as function of latitude. |
| DY | Grid distance in N-S direction |
| RAD | Earth's radius |
| PHI(m) | Latitude values |
| PI | Value of $\pi$ |

Input variables via arguments list

| Variable | Meaning |
|---|---|
| PP(L,M) | Value of the model parameter $T\underline{P}$ at one level |
| F(L,M) | The model parameter 'P' at one vertical level K. It is stored in the unit 92 in array GNEXT from where it is read in ELLIPT. |
| K | Index of vertical level at which the Helmholtz equation is being solved. |

Output variables

| Variable | Meaning |
|---|---|
| P(L,M) | The solution of the Helmholtz equation at one level, representing $\underline{Q} = T\underline{P}$ |

Description of subroutine

| Designator | Text |
|---|---|
| A-B | Various constants are defined. The array AA is intialized to value $1.0 \times 10^{-10}$ in DO loop 18. |

$$A_1 = \frac{\Delta t^2}{4} \times \Lambda_k$$

$$A_2 = 1.0$$

$$A_3 = AA_k$$

$$A_4 = 1.0 \times 10^{-3}$$

The iteration counter is assigned a maximum value = 1000. In DO loop 15 the boundary values of input array F are replaced by corresponding values of array PP.

B-C

A DO loop 11 is set up to index KK through 1000(NN) iterations. ERR1 and ERR2 are intialized to zero. ERR1 is relative later. ERR2 is absolute error. The constant A5 is intialized to value zero.

C-D

DO loop 12 is set up to index J through 2 to MM1 and I through 2 to LM1. ERR is computed form the relation

$$= \frac{\Delta t^2}{4} \cdot \Lambda_k \left[ \frac{(F_{i+1,j} + F_{i-1,j})}{\Delta x^2} + F_{i,j+1} \frac{1}{\Delta y^2} - \frac{\tan\phi_j}{a} + F_{i,j-1} \frac{1}{\Delta y^2} + \frac{\tan\phi_j}{a} \right] + F_{i,j} \left[ 1 - 2.0 \cdot \frac{\Delta t^2}{4} \Lambda_k \cdot \frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right] - PP_{i,j}$$

which finds the difference of $(\nabla^2 F - PP)$.

D-E

The value of F is modified using ERR as following:

$$(F_{ij})_{NEW} = (F_{ij})_{old} - \frac{1.0 \cdot ERR}{\left[ 1 - 2.0 \cdot \frac{\Delta t^2}{4} \Lambda_k \cdot \frac{1}{\Delta xj^2} + \frac{1}{\Delta y^2} \right]}$$

E-F

If $|PP1 > 1.0 \times 10^{-20}$ then the error is scaled as

$E1 = |ERR/PP_{ij}|$

otherwise E1 = 0.0

E2 is defined as $|ERR|$

If E1 > ERR1 then ERR1 = E1

If E2 > ERR2 then ERR2 = E2

If $|Pp_{ij}| > A5$ then $A5 = |PP_{ij}|$

DO loop 12 ends.
If ERR1 > tolerance limit A3 and absolute error ERR2 > A4 then the DO loop 11 starts next iteration. Otherwise KK, ERR1, ERR2 and A5 are written on unit 96.

F-G

In DO loop 14 the contents of $F_{ij}$ are transferred to $PP_{ij}$. Control transfers to calling program.

10.10 Subroutine:. MOVE

Abstract: This routine moves the contents of given array A to the array B. The multiplying factors INCA and INCB are used to fix the starting record in arrays A and B respectively.

Subroutines called: None

Input variables via common: None

Description of subroutine

| Designator | Text |
|---|---|
| A | Loop 11 is set up to index 'N' through record number 1 to NWD (= LxMxN in the present case). Then B((N-1) x INCB+1) = A((N-1) x INCA+1) achieves the transfer. DO loop 11 ends and control is transferred to calling routine. |

10.11 Subroutine: MULT

Abstract: In this subroutine product of two given NxM matrices F and G is found and returned in the matrix E.

Subroutines called:

Input variables via common

Input variables via arguments list

| Variables | Meaning |
|---|---|
| F(N,N) | NxN matrix |
| G(N,N) | NxN matrix |
| N | Order of matrix F & G |

Output variables

| Variable | Meaning |
|---|---|
| E(N,N) | NxN product matrix |

Description of subroutine

| Designator | Text |
|---|---|
| A | In DO loop 30 for I = 1(1)N and J = 1(1)N the product matrix E is initialized to zero and in the inner DO loop 30 the product is found as $E_{ij} = E_{ij} + F_{ik} \cdot G_{kj}$ DO loop ends and control transfers to calling program. |

10.12 Subroutine: RUTISH

Abstract: In this routine eigen-values and eigen vectors of any given square matrix are computed. The methodology adopted is to find the largest, next largest eigen values and so on. It uses the Rutishauser method and power method.

Subroutines called: INVRT

Input variables via common

Input variables via arguments list

| Variables | Meaning |
|---|---|
| B(M,M) | Input matrix of which eigen values and eigen vectors are to be obtained. |

Output variables

| Variables | Meaning |
|---|---|
| EGN(M) | M values of eigen values |
| VECTOR(M,M) | MxM Eigen vector matrix |

Description of routine

| Designator | Text |
|---|---|
| A-B | A variable M1 is defined as M1 = M-1. The tolerance error limit is defined as EPS = 0.0000000001. In the DO loop 98 the contents of input matrix B are transferred to working matrix C |
| B-C | The main DO loop 110 is initiated to index the counter IV through 1 to M. For IV < M a variable N is defined as N = M+1-IV and N1 = N1. DO loop 100 is set up to index counter I from 1 to N, in which an array Z(I) is initiated to one. A counter KC is initialized zero. A DO loop 101 is set up to index I from 1 to N. An array Y is initialized to zero. In this array the sum of product C(I,J)•Z(J) over the counter J of inner 101 loop is stored. (J= 1(1)N). The value of first element of the array Y is stored in R. |
| C-D | DO loop 90 selects the pivotal element of array y(I) by indexing I through 1 to N. The index kc is changed to 1 for non-pivotal elements. Loop 90 closes. For the pivotal elements (kc = 0) the control is transferred to 103 where value R = y(1) is stored in EGN corresponding to counter IV. For |

kc > 1 the control passes to 104 where in DO loop 115 the contents of array y(I) and control is transferred to 99 where the process of selecting a pivotal element starts again.

D-E

In DO loop 109 the contents of array y are transferred to working array X, with lag of one element i.e. X(I) = -y(I-1) DO loop 112 is initiated to index I through 2 to N and J through 2 to N. In this loop a matrix A is obtained with elements

C(I,J) + C(I,J)*X(I-1)

and in DO loop 113 contents of A are transferred to C. Here we get the matrix one order lower i.e. 3x3 matrix reduces to 2x2. Control is transferred to 118. But if IV=M the element C(1,1) is equal to the eigen value and is stored in EGN(IV). Statement 118 writer the counter IV and the corresponding eigen value on unit 96.

E-F

DO loop 105 is set up to index I through 1 to M1. Inner loop 107 is set up to index J through 1 to M. At every step one element of input matrix B is transferred to working array A as

A(I,J) = B(I+1,J+1)

For diagonal elements an array

X-A

is formed as A(I,J) - EGN(IV) and stroed in A(I,J). Loop 107 closes. A column of input matrix B is stored in array U as U(I) = -B(I+1,1).

F-G

DO loop 915 is initialized to index counter I through 1 to M1. For I < M1 the absolute value of diagonal elements A(I,I) is checked against EPS. For errors < EPS the array U is two consective values are added and stored in U as

U(I) = U(I) + U(I+1)

In DO loop 923 the consecutive of matrix A(I,J) are added and stored in A as

A(I,J) = A(I,J) + A(I+1,J)

Diagonal element A(I,I) is chosen as divisor in statement 907. Array element U(I) and the A(I,J) are divided by the divisor. Loop 915 is initiated for counter MM to index through 1 to M1, and a variable DELT is defined as

DELT = A(MM,I)

For non-diagonal elements

U(MM) = U(MM) - U(I) • DELT

and A(MM,J) = A(MM,J) - A(I,J) • DELT

DO loop 915 closes. This DO loop has computed the eigen vectors of eigen value EGN(IV).

G-H

Array element y(1) is initialized as 1.0. In DO loop 108 the contents of array U are transferred to

y as

```
y(I) = U(I-1)
```

In DO loop 121 the contents of y(I) are transferred to VECTOR (I,IV) which gives eigen value matrix. The eigen vectors are computed column wise. Control transfers to callling program.

10.13 Subroutine: ZERO

Abstract: In this subroutine, the given array of length L (here L = LxMxN) is initialized to zero

Subroutines called: None

Input variables via common: None

Input variables via argument list

| Variables | Meaning |
|---|---|
| F(L) | The given array of length L which is to be initialized. |
| L | Length of the array |

Output variables

| Variables | Meaning |
|---|---|
| F(L) | The initialized array |

Description of subroutine

| Designator | Text |
|---|---|
| A | A DO loop 11 is set up to index counter I through 1 to L. The array F is initialized<br>F(I) = 0.0<br>and the DO loop 11 ends. |

# Part III

## Case Study

## Case Study

### 1. Introduction

A case study is reported here as an illustration of this model. The input for this case study was the data for 12 GMT of 6 July 1983. These data were taken from the ECMWF analyses. The scaled terrain heights with a maximum height of 3000 metres were used. The albedo and sea surface temperatures for July were used. The synoptic situtation for the illustration is a Monsoon case study.

### 2. Input Charts.

2.1 The input charts for topography, albedo and sea surface temperatures are given in Figure No. 1.

2.2 The input charts for winds at 850 mb, 500 mb and 200 mb for 12 GMT of 6 July 1983 are given in Figure No. 2.

2.3 Forecast charts for 24 hours for winds at levels 850 mb, 500 mb and 200 mb valid for 12 GMT of 7 July 1983 are given in Figure No. 3.

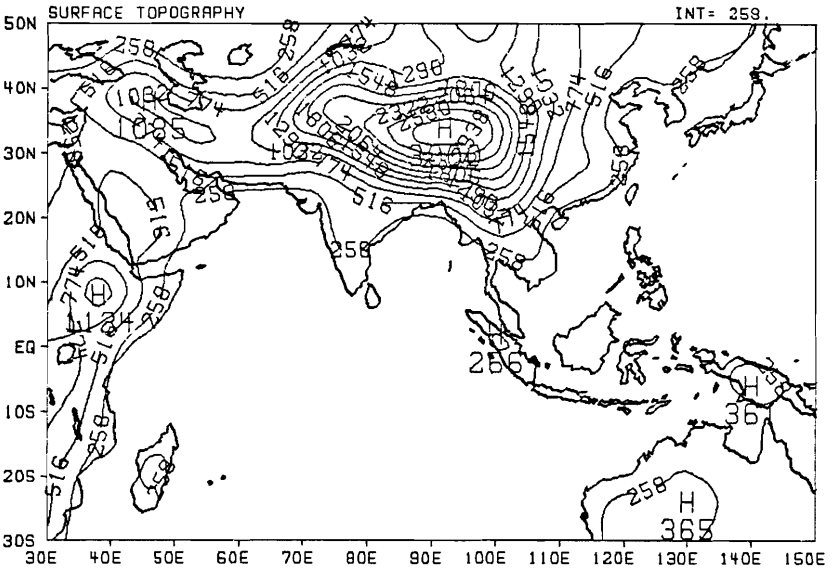
2.4 Verification charts for 12 GMT of 7 July 1983 for winds at 850 mb, 500 mb and 200 mb are given in Figure No. 4.

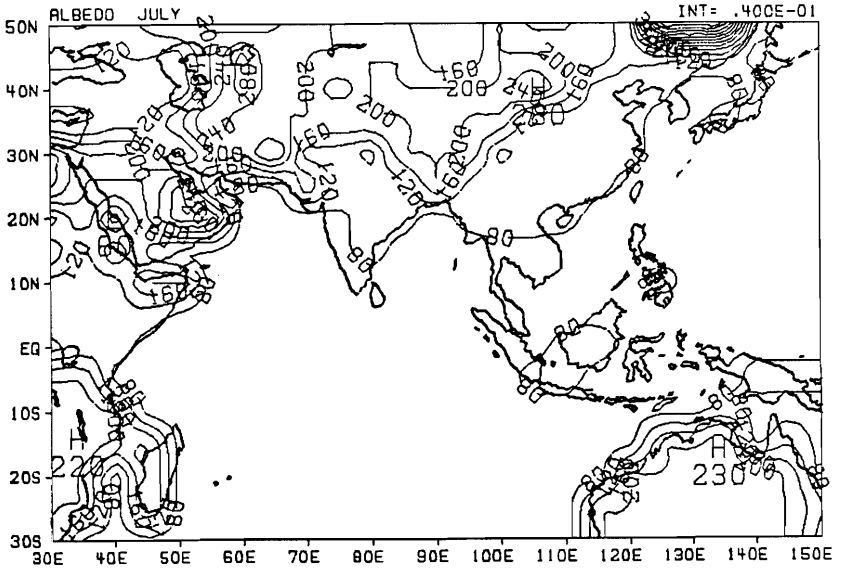
2.5 Forecast 24 hour rainfall and vertical velocities valid for 12 GMT of 7 July 1983 are given in Figure No. 5.

2.6 Forecast charts for 48 hours for winds at levels 850 mb, 500 mb and 200 mb valid for 12 GMT of 8 July 1983 are given in Figure No. 6.

2.7 Verification charts for 12 GMT of 8 July 1983 for winds at 850 mb, 500 mb and 200 mb are given in Figure No. 7.

2.8 Forecast 24 hour rainfall and vertical velocities valid for 12 GMT of 8 July 1983 are given in Figure No. 8.



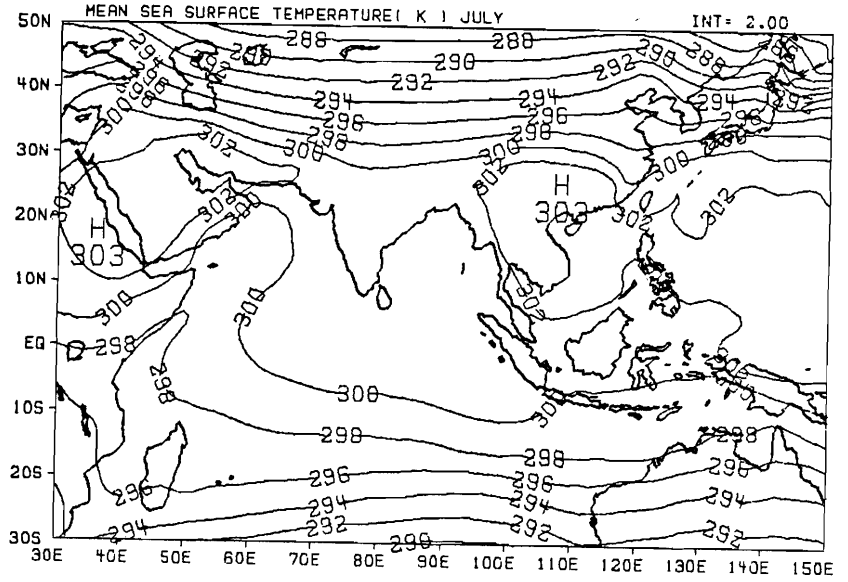




Figure No 1: Topography,Albedo and Sea surface Temperatures.



BEST AVAILABLE COPY