0-315-03589-7

51484

PERMISSION TO MICROFILM — AUTORISATION DE MICROFILMER

• Please print or type — Écrire en lettres moulées ou dactylographier

Full Name of Author — Nom complet de l'auteur

Timothy Otto Goos

Date of Birth — Date de naissance

March 19/1953

Country of Birth — Lieu de naissance

Canada

Permanent Address — Résidence fixe

12834 - 89 St.
Edmonton Alta
T5E 3J9

Title of Thesis — Titre de la thèse

A Finite-Element Method Applied to a Two-level Quasi-geostrophic Model Atmosphere

University — Université

University of Alberta

Degree for which thesis was presented — Grade pour lequel cette thèse fut présentée

M. Sc.

Year this degree conferred — Année d'obtention de ce grade

1981

Name of Supervisor — Nom du directeur de thèse

Dr. E P Lozowski

Date

April 24/81

Signature

Timothy Goos

THE UNIVERSITY OF ALBERTA

A FINITE-ELEMENT METHOD APPLIED TO A TWO-LEVEL QUASI-GEOSTROPHIC MODEL ATMOSPHERE

by

© TIMOTHY OTTO GOOS

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

IN

METEOROLOGY

DEPARTMENT OF GEOGRAPHY

EDMONTON, ALBERTA

SPRING, 1981

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR Timothy Otto Goos

TITLE OF THESIS A Finite-Element Method Applied to a Two-level Quasi-geostrophic Model Atmosphere

DEGREE FOR WHICH THESIS WAS PRESENTED Master of Science

YEAR THIS DEGREE GRANTED 1981

(Signed) Timothy Goos

PERMANENT ADDRESS:

12834 - 89 Street

Edmonton, Alberta

T5E 3J9

DATED February 27 1981

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled "A Finite-Element Method Applied to a Two-level Quasi-geostrophic Model Atmosphere" submitted by Timothy Otto Goos in partial fulfilment of the requirements for the degree of Master of Science in Meteorology.

Edward Lozowski
Supervisor

K. W. Hage

A. Craggs

Date February 27 1981

ABSTRACT

A finite-element method (using the Galerkin formulation) and a truncated spectral method are applied to find numerical solutions to the three non-linear partial differential equations describing a two-level quasi-geostrophic model atmosphere on a β-plane. This work was undertaken to provide a vehicle with which the author could study the finite-element method and its application to problems of meteorological interest. A relatively simple meteorological problem was chosen to allow concentration on the method and its application rather than complex physical interactions.

With the finite-element method, bi-linear basis functions defined on a variable resolution grid are used while sinusoidal basis functions are used with the spectral method. The domain of the problem is a channel of length 2.8×10^7m and width 4.4×10^6m with a free-slip wall boundary condition applied at the north and south boundaries and periodicity assumed in the x-direction. The grid consists of a central portion with uniformly high resolution and a uniformly changing resolution away from this sub-domain. A second-order Adams-Bashforth time integration scheme is used with both numerical techniques.

Parallel integrations of up to 48 hours duration for a set of four cases are presented and compared, using the spectral solution as a highly accurate standard. In the uniform resolution sub-domain, the finite-element solution achieves a maximum S1 score of under 25 for both height and thickness fields in three of the cases. Values of lower than 30 are generally considered to be near perfect forecasts in operational weather forecasting. However, it is to be noted that in this study, a highly simplified model is used and integrations are done from initial conditions with no inherent error. In the fourth case, numerical instability occurred due to the rapid growth of spurious short-wave-length waves generated near the boundaries. Investigations revealed that these waves were being produced by the inaccurate evaluation of normal derivatives near and on the boundaries. A possible method for overcoming this problem is discussed.

ACKNOWLEDGEMENTS

I am grateful to the people and organizations who helped me with this study.

I am indebted to my departmental supervisor, Dr. E. P. Lozowski, for his guidance, suggestions, and his thorough review of this manuscript. In particular, I wish to thank him for the confidence he placed in me in allowing me to study this particular area.

I wish to thank Dr. K. D. Hage and Dr. A. Craggs who, with Dr. Lozowski, served on my examining committee.

I wish to thank Dr. A. Staniforth of RPN in Dorval, P.Q. for supplying me copies of computer programs which have been developed in that office.

Thank you to the Department of Geography for providing me with the necessary computer resources to do this study.

This study was undertaken while I was on educational leave from the Atmospheric Environment Service of the Department of the Environment.

TABLE OF CONTENTS

CHAPTER		Page

LIST OF FIGURES

FIGURE		PAGE
5.39	Initial mean height field in metres for Case IV in (a) area B and (b) area A.	136
5.40	Initial thickness field in metres for Case IV in (a) area B and (b) area A.	137
5.41	Initial omega field in μbar/sec for Case IV in (a) area B and (b) area A.	138
5.42	The 48-hour spectral solution for the mean height field in metres for Case IV in (a) area B and (b) area A.	139
5.43	The 48-hour spectral solution for the thickness field in metres for Case IV in (a) area B and (b) area A.	140
5.44	The 48-hour spectral solution for the omega field in μbar/sec for Case IV in (a) area B and (b) area A.	141
5.45	The 48-hour finite-element solution for the mean height field in metres for Case IV in (a) area B and (b) area A.	142
5.46	The 48-hour finite-element solution for the thickness field in metres for Case IV in (a) area B and (b) area A.	143
5.47	The 48-hour finite-element solution for the omega field in μbar/sec for Case IV in (a) area B and (b) area A.	144
5.48	The (a) energies and (b) potential enstrophy of Case IV for the two solutions in the verification area.	145
5.49	The (a) kinetic energies and (b) potential energies of Case IV for the two solutions in the verification area.	146

FIGURE PAGE

LIST OF SYMBOLS

SYMBOL		PAGE
a	mean radius of the earth	6
a_i	eigenvalue of the i'th expansion function	23
APE	available potential energy	5
$b_{\ell i}$	interaction coefficient	25
$c_{\ell jk}$	interaction coefficient	25
c_p	specific heat for dry air at constant pressure	11
C	speed	60
D	domain	15
e_G	error in forecast gradient	63
$e^i(x,y)$	i'th two-dimensional basis function	38
$e^m(x)$	m'th one-dimensional basis function	41
f	Coriolis parameter	7
fo	Coriolis parameter at latitude ϕ_o	8
$F_i(x,y)$	i'th spectral expansion function	20
g	gravitational constant	17
$I(\phi)$	variational integral	42

SYMBOL		PAGE
$I^*(\phi)$	approximation to $I(\phi)$	43
J	Jacobian function	12
KE	kinetic energy	5
l_y	y-direction cosine	44
L_x	length of domain in x-direction	7
L_y	length of domain in y-direction	7
M	total mass of model atmosphere	16
N	total number of spectral expansion functions	20
NI	number of grid points in x-direction	40
NJ	number of grid points in y-direction	40
p	pressure	6
PE	potential enstrophy	16
q_i	potential vorticity at level i	18
Q_i	vorticity at level i	11
Q_g	geostrophic vorticity	9
$\overline{Q}$	vorticity of mean height field	13
$\hat{Q}$	vorticity of thickness field	13

SYMBOL		PAGE
R	gas constant for dry air	11
S1	verification score	62
$S(x_i)$	smoothing parameter at point x_i	64
t	time	9
T	temperature	11
TAU	streamfunction of the thickness field	109
u_x	x-derivative of u(x,y)	46
u_g	x-component of geostrophic wind	9
v_g	y-component of geostrophic wind	9
$\vec{V}_g$	geostrophic velocity	9
x	east-west coordinate in horizontal plane	6
y	north-south coordinate in horizontal plane	6
y_o	y-coordinate at ϕ_o	6
z	vertical coordinate	17
α	specific volume	11
β	variation of Coriolis parameter with latitude	3

SYMBOL		PAGE
Γ	boundary off domain	17
δ_{ij}	Kronecker delta	22
Δp	pressure difference between levels 1 and 3	6
θ	potential temperature	11
ρ	density	17
σ	static stability parameter	10
τ_i	i'th spectral expansion coefficient for the thickness field	23
ϕ	latitude	6
ϕ_o	latitude where tangent plane touches the earth	6
$\phi_{B,i}$	value of the function ϕ on the boundary	55
ψ_i	(1) stream function at level i	6
	(2) i'th spectral expansion coefficient for the height field	23
$\bar{\psi}$	stream function for the mean height [illegible]	12
$\hat{\psi}$	stream function for the thickness field	12
ψ_Z	zonal stream function	16
ψ_E	eddy stream function	16
ω	over-relaxation factor	59

SYMBOL		PAGE
$\tilde{\omega}$	vertical wind component in isobaric coordinates	6
ω_i	i'th spectral expansion coefficient for vertical wind component	23
$\tilde{\omega}_i$	vertical wind component at level i	10
Ω	angular speed of rotation of the earth	7
XCI	streamfunction of the mean height field	109

CHAPTER 1

INTRODUCTION

An accurate method for producing a weather forecast has been an elusive dream throughout much of man's history. The abundance of folk beliefs (Hornstein, 1978) available for predicting the weather indicates the importance man has attached to weather forecasts. During this century, much attention has been focussed on using numerical methods to solve the hydro-dynamical equations, thereby producing a weather forecast (Bjerknes 1904; Richardson, 1921; Charney et. al., 1950; Cressman, 1958, 1963; Shuman and Hovermale, 1968; Bourke, 1972, 1974; Machenhauer and Daley, 1972).

In recent years (Wang et. al., 1972; Cullen, 1973, 1974a, 1974b, 1976, 1979; Staniforth and Mitchell, 1977, 1978; Staniforth and Daley, 1977, 19[illegible], 1979), the finite-element method (FEM) has been applied to the problem of numerical weather prediction. The development of larger and faster computers has made possible the use of the FEM for the prediction of large-scale atmospheric flows. The FEM enables one to use a grid with variable spacing between grid points with relative ease. This is a distinct advantage over the finite-difference methods which have been used previously.

Numerical weather prediction began when Bjerknes (1904) recognized that the primitive equations of meteorology formed a system of non-linear partial differential equations which could, in principle, be solved to forecast the subsequent states of the atmosphere from a known initial state. He also recognized that the system did not have an analytic solution and that the available data were inadequate to specify the initial conditions. Holton (1972) shows that the primitive equations consist of the momentum equations in the horizontal (plane), the thermodynamic energy equation, the continuity equation, the hydrostatic approximation, and the equation of state. This system of equations forms the basis of large-scale numerical weather prediction. Richardson (1921) attempted the first numerical integration of these equations using a finite-difference method. Unhappily, his forecast took many months to produce and was in error by several orders of magnitude.

The development of the electronic computer in the 1940s allowed Charney et. al. (1950) to perform the first successful numerical weather forecast. They integrated, in time, the barotropic vorticity equation which is a simplified equation derivable from the primitive equations. By simplifying the primitive equations using certain assumptions about the atmosphere, one may obtain a system of equations which is more easily solved. Holton (1972) describes the assumptions necessary eliminate sound, gravity and inertia waves. Charney et. al. (1950) used the most simplified form of the primitive equations for their integration.

The field of numerical weather prediction has expanded rapidly since the first successful integration, with many more complicated finite difference models being developed (Cressman, 1958, 1963;

Shuman and Hovermale, 1968; Howcroft, 1971). Beginning in the late 1960's, the use of the spectral method for integrating the primitive equations was investigated (Robert, 1966, 1970; Elsaesser, 1966; Eliasen and Machenhauer, 1970; Baer and Alyea, 1971; Bourke, 1972). Efficient spectral models were not possible until Orszag (1970) described the use of transforms to allow the efficient evaluation of the product terms in the primitive equations. Since then, the spectral method has been used with success in numerical weather prediction (Daley et. al., 1976; Bourke, 1974).

In the 1970's, the FEM was first introduced to numerical weather prediction. Early work (Wang et. al., 1972; Cullen, 1973, 1974a), considered highly simplified atmospheric models. Recently, models using the primitive equations defined over the northern hemisphere have been integrated using the FEM (Staniforth and Daley, 1978, 1979; Cullen, 1979). Staniforth and Daley (1979) have found the finite-element model produces forecasts as accurate as those of the operational spectral model even though the finite-element model does not contain many of the physical processes present in the spectral model.

In this thesis, a FEM is used to solve a set of partial differential equations. These equations mathematically describe a two-level baroclinic model of the atmosphere on a β-plane. (The model is fully described in Chapter 2.) The equations of this model atmosphere are also solved using a spectral method. This solution is highly accurate and is used to evaluate the accuracy of the solution by the

FEM. In Chapter 5, the solutions using both methods are presented and compared. Although the meteorological problem under consideration is relatively simple, this approach provides a good framework within which my supervisor and I could learn about and apply the FEM.

CHAPTER 2

A DESCRIPTION OF THE MODEL ATMOSPHERE

2.1 The Model Equations and the Domain

The intent of this thesis is to study the application of the FEM to the prediction of synoptic scale atmospheric flows. Consequently, a model atmosphere has been chosen in which such flows are possible. The quasi-geostrophic two-level model was selected as it is the simplest synoptic scale model which includes baroclinic effects. This model has been used in the past to study atmospheric flows (e.g. Phillips (1951), Holton (1972), Stone (1974), Held (1975)) as although it is a relatively simple atmospheric model, it simulates many atmospheric processes well.

A model atmosphere which allows baroclinic effects has been employed because the baroclinic conversion of potential to kinetic energy is a major process in the development of synoptic scale storms. The total potential energy of a system is the sum of its internal energy and its gravitational potential energy. Lorenz (1955) defines the available potential energy (APE) of a column of the atmosphere to be the difference between the total potential energy and the minimum total potential energy which could be achieved by an adiabatic redistribution of the mass of that air column. The APE is that portion of the total potential energy which is available for possible conversion to kinetic energy (KE). This conversion is often an unstable process, i.e. baroclinic instability.

In the quasi-geostrophic two-level model, the atmosphere is divided into levels as shown in Figure 2.1. Pressure (p) is used as the vertical coordinate. The streamfunction (ψ) is defined at levels 1, 2 and 3 and the vertical motion ($\tilde{\omega}$) at levels 0, 2 and 4 is considered. Levels 1 and 3 are the two levels of prime importance, from which the model derives its name. The pressure difference between levels 1 and 3, Δp, is 500 mb for this model. Pedlosky (1979) has shown that this model, with one small additional assumption, has the same dynamic equations as a two layer model, in which the layer between levels 0 and 2 is assumed to have a certain uniform density and the layer between levels 2 and 4 is assumed to have another uniform density.

Prior to presenting the model equations, a few definitions will be given. When the β-plane approximation is used, the earth is approximated by a plane tangent to the earth at some latitude ϕ_o, as shown in Figure 2.2. On this plane, a three-dimensional right-handed Cartesian coordinate system is defined. The unit vector $\hat{i}$ points in the x-direction, which is east, while the unit vector $\hat{j}$ points in the y-direction, which is north. The unit vector $\hat{k}$ points vertically in the direction of lower atmospheric pressure. The origin of the coordinate system is defined to be at the point $x = 0$, $y = 0$, $p = 1000$ mb. A relation between latitude, ϕ, and distance along the y-axis may be written:

$$y - y_o = a(\phi - \phi_o) \qquad (2.1)$$

where a is the mean radius of the earth and y_o is the value of y at ϕ_o.

The plane in Figure 2.2 is assumed to have a width of $L_y/2$ so the range of y is $[0, L_y/2]$. This definition for the width of the plane is made for consistency with the expansion functions used with the spectral method discussed in Chapter 3. The plane is assumed to have a length L_x. More precisely, it is assumed that the solution to the model equations, derived later in this chapter, is periodic in x with wavelength L_x. This length of the domain is also chosen for consistency with the expansion functions.

The particular values of L_x and L_y used are chosen by considering the type of atmospheric motions to be studied. As mid-latitude synoptic scale motions will be considered in this thesis, ϕ_o is chosen to be 45^oN latitude. The domain is chosen so that $L_y/2 = 4.4 \times 10^6$m, which is approximately 55^o of latitude on the earth. This width is chosen by considering the meridional extent of typical mid-latitude long waves on the earth. It is similar to that used by others (Cullen, 1976; Grammeltvedt, 1969). The chosen length of periodicity is $L_x = 2.8 \times 10^7$m which is essentially the length of the latitude circle at 45^oN.

The Coriolis parameter, f, is defined by:

$$f = 2\Omega \sin\phi \tag{2.2}$$

where Ω is the earth's angular velocity. If only the first two terms of a Taylor expansion of f about y_o are retained, one obtains:

$$f = f_o + \beta(y-y_o) \tag{2.3}$$

where:

$$\beta = \left.\frac{df}{dy}\right|_{\phi=\phi_o} = \frac{2\Omega}{a}\cos\phi_o \tag{2.4}$$

and f_o is the Coriolis parameter at ϕ_o.

The horizontal del operator is defined by:

$$\vec{\nabla} = \hat{i}\frac{\partial}{\partial x} + \hat{j}\frac{\partial}{\partial y} \tag{2.5}$$

while the horizontal Laplacian operator is defined by:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{2.6}$$

Jeffre[illegible] 1922) defines the geostrophic velocity to be the velocity field resulting from a balance between the Cor[illegible]s force and the pressure gradient force. When pressure is used as the vertical coordinate, the pressure gradient force term is transformed [illegible]o a geopotential height gradient term. If the flow is quasi-non-divergent, the geopotential height gradient term may be replaced by a stream function gradient term.

The horizontal geostrophic velocity is quasi-non-divergent and, therefore, it may be written:

$$\vec{V}_g = \hat{k} \times \vec{\nabla}\psi_g \tag{2.7}$$

where $\vec{V}_g$ is the horizontal geostrophic velocity and ψ_g is the geostrophic stream function. In component form, this is:

$$u_g = -\frac{\partial \psi_g}{\partial y} \tag{2.8}$$

$$v_g = \frac{\partial \psi_g}{\partial x} \tag{2.9}$$

where $\vec{V}_g = \hat{i}\, u_g + \hat{j}\, v_g$.

The vertical velocity in pressure coordinates, $\tilde{\omega}$, is defined by:

$$\tilde{\omega} = \frac{dp}{dt} \tag{2.10}$$

where t is time. The geostrophic vorticity, Q_g, is defined by:

$$Q_g = \hat{k} \cdot \vec{\nabla} \times \vec{V}_g = \nabla^2 \psi_g \tag{2.11}$$

The quasi-geostrophic vorticity equation may be written as (Holton, 1972; Hoskins, 1975):

$$\frac{\partial Q_g}{\partial t} = -\vec{V}_g \cdot \vec{\nabla}\{Q_g + f\} + f_o \frac{\partial \tilde{\omega}}{\partial p} \tag{2.12}$$

It is paradoxical that although the geostrophic wind is non-divergent, the second term in (2.12) is a divergence term. Holton (1972) has shown that this term is important in keeping temperature changes hydrostatic and vorticity changes geostrophic in synoptic scale systems. The temperature and vorticity fields must remain consistent with the original assumptions of the quasi-geostrophic vorticity equation. In the derivation of this equation, it is assumed that the atmosphere is in hydrostatic balance and that the flow is quasi-geostrophic.

Eqn. (2.12) is applied at levels 1 and 3 using the following finite-difference approximations for the vertical gradient of $\tilde{\omega}$:

$$\left(\frac{\partial \tilde{\omega}}{\partial p}\right)_1 \simeq \frac{\tilde{\omega}_2 - \tilde{\omega}_0}{\Delta p} \qquad (2.13)$$

$$\left(\frac{\partial \tilde{\omega}}{\partial p}\right)_3 \simeq \frac{\tilde{\omega}_4 - \tilde{\omega}_2}{\Delta p} \qquad (2.14)$$

The vertical velocities at the top, $\tilde{\omega}_0$, and bottom, $\tilde{\omega}_4$, of the atmosphere are assumed to be zero. This assumption eliminates external gravity waves from the model.

Holton (1972) shows that the adiabatic quasi-geostrophic thermodynamic energy equation may be written:

$$\frac{\partial}{\partial t}\left(-\frac{\partial \psi}{\partial p}\right) = -\vec{V}_g \cdot \vec{\nabla}\left(-\frac{\partial \psi}{\partial p}\right) + \frac{\sigma}{f_o}\tilde{\omega} \qquad (2.15)$$

where the static stability parameter is defined by:

$$\sigma = -\frac{\alpha}{\theta}\frac{\partial\theta}{\partial p} \tag{2.16}$$

and the potential temperature is defined by:

$$\theta = T\left(\frac{1000}{p}\right)^{R/c_p} \tag{2.17}$$

In these equations, α is the specific volume and T is the temperature of the air. R is the specific gas constant for dry air and c_p is its specific heat at constant pressure. Eqn. (2.15) is applied at level 2 using the following finite-difference approximation:

$$\left(\frac{\partial\psi}{\partial p}\right)_2 \simeq \frac{\psi_3-\psi_1}{\Delta p} \tag{2.18}$$

Holton (1972), applying Eqns. (2.12) and (2.15) as indicated, finds the three model equations to be:

$$\frac{\partial Q_1}{\partial t} = - J(\psi_1,Q_1) - J(\psi_1,f) + \frac{f_o}{\Delta p}\tilde{\omega}_2 \tag{2.19}$$

$$\frac{\partial Q_3}{\partial t} = - J(\psi_3,Q_3) - J(\psi_3,f) - \frac{f_o}{\Delta p}\tilde{\omega}_2 \tag{2.20}$$

$$\frac{\partial}{\partial t}(\psi_1-\psi_3) = - J(\psi_2,\psi_1-\psi_3) + \frac{\sigma\Delta p}{f_o}\tilde{\omega}_2 \tag{2.21}$$

where the Jacobian operator, J, is defined by:

$$J(A,B) = \frac{\partial A}{\partial x}\frac{\partial B}{\partial y} - \frac{\partial A}{\partial y}\frac{\partial B}{\partial x} \tag{2.22}$$

In Eqns. (2.19) - (2.21), the stream function is assumed to be the geostrophic stream function although the subscript denoting the geostrophic value has been dropped here and will be for the remainder of this thesis. The static stability parameter is assumed to be horizontally homogeneous and independent of time.

The stream function for the mean flow, $\overline{\psi}$, and the thickness, $\hat{\psi}$, are defined to be:

$$\overline{\psi} = \frac{\psi_1+\psi_3}{2} \tag{2.23}$$

$$\hat{\psi} = \frac{\psi_1-\psi_3}{2} \tag{2.24}$$

Note that $\hat{\psi}$ is really half the thickness between levels 1 and 3. However, for convenience, $\hat{\psi}$ will be simply referred to as the thickness in this thesis. The stream function at level 2, ψ_2, is obtained by linearly interpolating between levels 1 and 3 giving:

$$\psi_2 \simeq \frac{\psi_1+\psi_3}{2} \tag{2.25}$$

Thus, in Eqn. (2.21), ψ_2 may be approximated by $\overline{\psi}$.

Forming the sum and difference of Eqns. (2.19) and (2.20) and applying Eqns. (2.23) - (2.25) to (2.19) - 2.21), one obtains:

$$\frac{\partial \overline{Q}}{\partial t} = - J(\overline{\psi},\overline{Q}) - J(\hat{\psi},\hat{Q}) - J(\overline{\psi},f) \qquad (2.26)$$

$$\frac{\partial \hat{Q}}{\partial t} = - J(\hat{\psi},\overline{Q}) - J(\overline{\psi},\hat{Q}) - J(\hat{\psi},f) + \frac{f_o}{\Delta p} \tilde{\omega}_2 \qquad (2.27)$$

$$\frac{\partial \hat{\psi}}{\partial t} = - J(\overline{\psi},\hat{\psi}) + \frac{\sigma \Delta p}{2f_o} \tilde{\omega}_2 \qquad (2.28)$$

Finally, Eqns. (2.27) and (2.28) may be combined to yield:

$$(\nabla^2-\lambda^2) \frac{\partial \hat{\psi}}{\partial t} = - J(\hat{\psi},\overline{Q}) - J(\overline{\psi},\hat{Q}) - J(\hat{\psi},f) + \frac{2f_o^2}{\sigma \Delta p^2} J(\overline{\psi},\hat{\psi}) \qquad (2.29)$$

$$(\nabla^2-\lambda^2) \tilde{\omega}_2 = - \frac{4f_o}{\sigma \Delta p} J(\hat{\psi},\overline{Q}) - \frac{2f_o}{\sigma \Delta p} J(\hat{\psi},f) \qquad (2.30)$$

where:

$$\lambda^2 = \frac{2f_o^2}{\sigma \Delta p^2} \qquad (2.31)$$

Equations (2.26), (2,29) and (2.30) form a system of three non-linear partial-differential equations in the three unknowns: $\overline{\psi}$, $\hat{\psi}$ and $\tilde{\omega}_2$. Given boundary and initial conditions, a solution to this system of equations may be found. Baer (1970) has found an analytical solution for a particular set of boundary and initial conditions. For more general solutions, however, numerical methods must be used. Chapters 3 and 4 describe the two numerical methods employed in this thesis.

2.2 The Boundary Conditions

A free-slip wall boundary condition was imposed at the northern and southern boundaries of the domain. Accordingly, there can be no flow perpendicular to the wall and there is no viscous boundary layer at the wall. Using Eqns. (2.7) - (2.9), the first of these conditions implies that:

$$\frac{\partial \psi}{\partial x} = 0 \qquad (2.32)$$

at $y = y_o + L_y/4$ and at $y = y_o - L_y/4$. The second of these conditions means that no additional terms must be used with the model equations found in Sec. 2.1 at the boundary of the domain. These extra terms would be necessary to account for the turbulence generated near the boundary if free slip was not assumed.

These conditions do not completely specify the solution as ψ is specified by them only to within an arbitrary additive constant. Pedlosky (1979) and Phillips (1954) have shown that, as a consequence of the vanishing of the normal velocity component on the boundary:

$$\int_0^{L_x} \frac{\partial^2 \psi}{\partial y \partial t}\, dx = 0 \qquad (2.33)$$

where the integration is made along the northern [illegible] southern boundaries separately. Pedlosky (1979) has shown that [illegible] (2.33) is equivalent to:

$$\iint_D \frac{\partial \psi}{\partial t}\, dxdy = 0 \qquad (2.34)$$

where the integration occurs over the domain, D.

As the boundary conditions, Eqn. (2.32) and (2.33) or (2.34), apply to the stream functions at each level, they also apply to linear combinations of these stream functions. Thus, they may be applied to the derived stream functions $\overline{\psi}$ and $\hat{\psi}$. Equation (2.33) is the form of the second boundary condition which is used in this thesis. This form requires less computer time when numerically integrated, as it must be when applied to find a solution by the FEM.

2.3 The Energy and Potential Enstrophy Relations

The study of the energetics of the atmosphere and, in particular, of the exchanges of energy among its various forms, among the various scales of motion, and among the sources and sinks of energy, provides an important tool for understanding atmospheric motion. Also, the potential enstrophy (PE) may be used as an aid in understanding atmospheric motion. The PE is defined to be the mean squared potential vorticity. Holton (1972) shows that the potential vorticity, q, is a measure of the ratio of the vorticity of a vortex to the depth of the vortex.

These quantities also provide a useful tool for studying a particular atmospheric model, and, they may be used as a means of comparing an atmospheric model with other models or with the real atmosphere. For the chosen model, there are three quantities of interest:

1) the kinetic energy (KE)
2) the available potential energy (APE)
3) the potential enstrophy

The KE and APE are considered to be present in two forms: zonal (ZKE and ZAPE) and eddy (EKE and EAPE). The atmospheric flow is considered to have a basic zonal component with disturbances called "eddies" superimposed on that flow. Thus, any stream function, ψ, may be written as the sum of a zonal, ψ_Z, and an eddy, ψ_E, stream function.

The KE of the horizontal flow is defined as:

$$KE = \int_M \frac{1}{2} (\vec{V} \cdot \vec{V}) \, dM \qquad (2.35)$$

where dM is an element of mass and the integration is over the total mass, M, of the atmosphere. For the two-level model under consideration, this may be written as:

$$KE = \frac{\Delta p}{g} \iint_D (\nabla\bar{\psi}\cdot\nabla\bar{\psi} + \nabla\hat{\psi}\cdot\nabla\hat{\psi})\ dxdy \qquad (2.36)$$

where the hydrostatic relation, $dp/dz = -g\rho$, and $dM = \rho d\,x\,dydz$ have been used. ρ is the atmospheric density, g is the acceleration due to gravity and z is the vertical coordinate parallel to the -p direction.

Differentiating Eqn. (2.36) with respect to time and using the divergence theorem and the boundary conditions, one may show that:

$$\frac{dKE}{dt} = -\frac{2f_o}{g} \iint_D \tilde{\omega}_2\hat{\psi}\ dxdy \qquad (2.37)$$

This shows that when, in the mean, warm air is ascending and cold air is descending, there is an increase in the kinetic energy, because the centre of gravity is being lowered.

For this model, Holton (1972) shows that the APE is given by:

$$APE = \frac{\Gamma}{2} \iint_D \hat{\psi}^2\ dxdy \qquad (2.38)$$

where $\Gamma = 4\, f_o^2/g\sigma\Delta p$. Differentiating Eqn. (2.38) with respect to time, one finds that:

$$\frac{dAPE}{dt} = \frac{2f_o}{g} \iint_D \tilde{\omega}_2 \hat{\psi}\; dxdy \qquad (2.39)$$

Combining Eqns. (2.37) and (2.39), one has:

$$\frac{d}{dt}(KE+APE) = 0 \qquad (2.40)$$

Thus, the total energy, TE = KE + APE, of the model is conserved but an exchange of energy between KE and APE may occur.

Relations similar to Eqns. (2.36) and (2.38) are used to calculate the eddy and zonal components of the KE and APE. Conservation relations, similar to Eqn. (2.40), yield the intuitive result that the total KE (APE) is simply the sum of the ZKE (ZAPE) and the EKE (EAPE).

The potential vorticities at levels 1 and 3 ar given by:

$$q_1 = Q_1 - \frac{f_o^2}{\sigma\Delta p^2}(\psi_1 - \psi_3) \qquad (2.41)$$

$$q_3 = Q_3 + \frac{f_o^2}{\sigma\Delta p^2}(\psi_1 - \psi_3) \qquad (2.42)$$

and the PE is given by:

$$PE = \frac{\Delta p}{2g} \iint_D (q_1^2 + q_3^2)\ dxdy \tag{2.43}$$

Differentiating Eqns. (2.42) - (2.44) with respect to time and using the divergence theorem, one finds that:

$$\frac{dPE}{dt} = 0 \tag{2.44}$$

Thus, the potential enstrophy is a conservative property of this model. When an approximate numerical solution is found for the model equations, the two conservative quantities, TE and PE, will provide a test of the conservation properties of that solution.

CHAPTER 3

THE SPECTRAL METHOD

3.1 A Description of the Method

This chapter details the method of solution of the model equations, (2.26) - (2.28), using the spectral method, which is an application of a method developed by Galerkin (1915). When this method is applied to the model equations, a highly accurate (but not analytic) solution is obtained. It is found that the solution by the spectral method conserves total energy and potential enstrophy. This spectral solution will be used to judge the accuracy of the solution by the FEM.

A simple example will be used to demonstrate the application of the spectral method to a boundary-value problem. Consider the following equation:

$$\frac{\partial f(x,y,t)}{\partial t} = g(x,y,t) \qquad (3.1)$$

This is similar in form to the model equations, (2.26) - (2.28). The functions f (x,y,t) and g (x,y,t) are assumed to be defined on a two-dimensional domain D and the problem is assumed to be properly posed. The functions f and g are represented using a set of N + 1 expansion functions, F_i (x,y), to obtain:

$$f(x,y,t) \simeq \hat{f}(x,y,t) = \sum_{i=0}^{N} f_i(t)\ F_i(x,y) \qquad (3.2)$$

$$g(x,y,t) \simeq \hat{g}(x,y,t) = \sum_{i=0}^{N} g_i(t)\; F_i(x,y) \tag{3.3}$$

Kreider et. al. (1966) show that if the expansion functions form a basis on the domain, the approximate functions, $\hat{f}$ and $\hat{g}$, will converge to the true functions, f and g, as N approaches infinity. It is assumed that the expansion functions can satisfy the boundary conditions.

The approximate functions are substituted into Eqn. (3.1) using Eqns. (3.2) and (3.3) and a residual, R (x,y,t) is defined according to:

$$R(x,y,t) = \sum_{i=0}^{N} \frac{df_i}{dt} F_i - \sum_{i=0}^{N} g_i F_i \tag{3.4}$$

Note that in Eqn. (3.4) the total derivative with respect to time is used as the expansion coefficients, f_i and g_i, depend only on time.

The explicit dependence of the expansion coefficients on time and of the expansion functions, F_i, on the two space variables have been omitted for the sake of brevity. As the approximate functions will not, in general, satisfy Eqn. (3.1) exactly, the residual is not necessarily zero.

In Galerkin's method, the residual is forced to zero with respect to the expansion functions in an average sense over the domain

(Zienkiewicz,). Thus, we insist that:

$$\iint_D F_j \left(\sum_{i=0}^{N} \frac{df_i}{dt} F_i - \sum_{i=0}^{N} g_i F_i \right) dxdy = 0 \qquad (3.5)$$

for j = 0, 1, 2, . . ., N. This technique gives a system of N + 1 ordinary differential equations for the expansion coefficients, $f_i(t)$. This system may then be numerically integrated, given initial conditions, to find the approximate solution, $\hat{f}$ (x,y,t), to Eqn. (3.1).

3.2 The Application of the Spectral Method to the Model Equations

The expansion functions must be specified and they are usually chosen with properties which are particularly appropriate to the problem under consideration. For the present problem, the following set of properties is chosen:

1) F_o is a constant over the domain
2) $\partial F_i/\partial x = 0$ along the north and south boundaries
3) the F_i are orthonormal i.e.

$$\frac{1}{A} \iint_D F_i F_j \, dxdy = \delta_{ij} \qquad (3.6)$$

where δ_{ij} is the Kronecker delta and A is the area of the domain

4) the expansion functions are eigenfunctions of the Laplacian operator so that:

$$\nabla^2 F_i = - a_i^2 F_i \tag{3.7}$$

where $a_i^2 > 0$

The first property allows one to represent a field with a non-zero mean using the expansion functions. Property 2 is chosen so that the boundary condition, Eqn. (2.32), may be satisfied. Property 3 is chosen to facilitate the integration of terms in equations similar to Eqn. (3.5). Property 4 is chosen as the Laplacian operator is present in the model equations. This property will also simplify the integration of terms in equations similar to Eqn. (3.5).

Expanding the variables $\overline{\psi}$, $\hat{\psi}$ and $\tilde{\omega}_2$, one has:

$$\overline{\psi}(x,y,t) \simeq \sum_{i=0}^{N} \psi_i(t)\ F_i(x,y) \tag{3.8}$$

$$\hat{\psi}(x,y,t) \simeq \sum_{i=0}^{N} \tau_i(t)\ F_i(x,y) \tag{3.9}$$

$$\tilde{\omega}_2(x,y,t) \simeq \sum_{i=0}^{N} \omega_i(t)\ F_i(x,y) \tag{3.10}$$

Using this expansion and Eqn. (3.7), the first model equation, Eqn. (2.26), becomes:

$$-\sum_{i=0}^{N} a_i^2 \frac{d\psi_i}{dt} F_i = \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 \psi_j \psi_k J(F_j, F_k) + \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 \tau_j \tau_k J(F_j, F_k)$$

$$- \beta \sum_{i=0}^{N} \psi_i \frac{\partial F_i}{\partial x} + R \qquad (3.11)$$

Applying the Galerkin procedure to Eqn. (3.11) by multiplying by a particular F_ℓ and integrating over the domain, one has:

$$- a_\ell^2 \frac{d\psi_\ell}{dt} = \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 \psi_j \psi_k \overline{F_\ell J(F_j, F_k)} + \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 \tau_j \tau_k \overline{F_\ell J(F_j, F_k)}$$

$$- \beta \sum_{i=0}^{N} \psi_i \overline{F_\ell \frac{\partial F_i}{\partial x}} \qquad (3.12)$$

where the averaging operator, $\overline{(\quad)}$, is defined by:

$$\overline{(\quad)} = \iint_D (\quad) \, dxdy \qquad (3.13)$$

The terms in Eqn. (3.12, which involve the averaging operator may only be evaluated once the expansion functions have been chosen. These terms, however, are independent of space and time. They are constants which depend only on the expansion functions and the domain of the problem and they may be written:

$$c_{\ell jk} \equiv \overline{F_\ell J(F_j,F_k)} \tag{3.14}$$

$$b_{\ell i} \equiv \overline{F_\ell \frac{\partial F_i}{\partial x}} \tag{3.15}$$

Applying the Galerkin procedure to the three model equations, Eqns. (2.26) - (2.28), and using Eqns. (3.14) and (3.15), one obtains:

$$-a_\ell^2 \frac{d\psi_\ell}{dt} = \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 c_{\ell jk}(\psi_j\psi_k+\tau_j\tau_k) - \beta \sum_{i=0}^{N} b_{\ell i}\psi_i \tag{3.16}$$

$$-a_\ell^2 \frac{d\tau_\ell}{dt} = \sum_{j=0}^{N} \sum_{k=0}^{N} a_k^2 c_{\ell jk}(\tau_j\psi_k+\psi_j\tau_k) - \beta \sum_{i=0}^{N} b_{\ell i}\tau_i + \frac{f_o}{\Delta p}\,\omega_\ell \tag{3.17}$$

$$\frac{d\tau_\ell}{dt} = -\sum_{j=0}^{N} \sum_{k=0}^{N} c_{\ell jk}\psi_j\tau_k + \frac{\sigma\Delta p}{2f_o}\,\omega_\ell \tag{3.18}$$

Using Eqns. (3.14) and (2.22), one may show that:

$$\sum_{j=0}^{N} \sum_{k=0}^{N} X_k Y_{jk} c_{\ell jk} = \frac{1}{2}\left(\sum_{j=0}^{N} \sum_{k=0}^{N} (X_k-X_j) Y_{jk} c_{\ell jk}\right) \tag{3.19}$$

for arbitrary X_k and for all Y_{jk} where $Y_{jk} = Y_{kj}$. X_k and Y_{jk} represent expansion coefficients or combinations of expansion coefficients which depend only on the given indices. Eqn. (3.19) may be applied to Eqns. (3.16) - (3.18) to give

$$\frac{d\psi_\ell}{dt} = \frac{1}{2}\left(\sum_{j=0}^{N}\sum_{k=0}^{N}\frac{(a_j^2-a_k^2)}{a_\ell^2}c_{\ell jk}(\text{[illegible]} + \text{[illegible]}\right) + \frac{\beta}{a_\ell^2}\sum_{i=0}^{N}b_{\ell i}\psi_i \qquad (3.20)$$

$$\frac{d\tau_\ell}{dt} = \frac{1}{2}\left(\sum_{j=0}^{N}\sum_{k=0}^{N}\frac{(a_j^2-a_k^2)}{a_\ell^2}c_{\ell jk}(\tau_j\psi_k+\tau_k\psi_j)\right) + \frac{\beta}{a_\ell^2}\sum_{i=0}^{N}b_{\ell i}\tau_i - \frac{f_o}{\Delta p}\frac{\omega_\ell}{a_\ell^2} \qquad (3.21)$$

$$\frac{d\tau_\ell}{dt} = \frac{1}{2}\left(\sum_{j=0}^{N}\sum_{k=0}^{N}c_{\ell jk}(\tau_j\psi_k-\tau_k\psi_j)\right) + \frac{\sigma\Delta p}{2f_o}\omega_\ell \qquad (3.22)$$

Eqns. (3.20) - (3.22) yield a system of 3 (N + 1) ordinary differential equations for the 3 (N + 1) expansion coefficients. Before solving these, the expansion functions must be chosen. They must satisfy the four criteria given earlier. Following Lorenz (1960), the choice of expansion functions is:

$$\begin{aligned}
F_0 &= 1 \\
F_1 &= \sqrt{2}\cos\frac{2\pi}{L_y}y \\
F_2 &= 2\sin\frac{2\pi}{L_y}y\,\sin\frac{2\pi}{L_x}nx \\
F_3 &= 2\sin\frac{2\pi}{L_y}y\,\cos\frac{2\pi}{L_x}nx
\end{aligned} \qquad (3.23)$$

This is referred to as a truncated spectral series expansion. Applying Eqn. (3.7) to these functions, one finds:

$$a_0^2 = 0 \tag{3.24}$$

$$a_1^2 = (2\pi/L_y)^2 \tag{3.25}$$

$a_2^2 = a_3^2$ [illegible] $\pi/$[illegible]$)^2$ [illegible] $(2\pi n/L_x)^2$ (3.26)

This set of functions allows [illegible] wavenumber n to exist in the x-direction. This particular wavenumber will be left to be chosen later. In the y-direction, only waves of one wavenumber are allowed and this is completely specified with the choice of L_y given earlier. In order to fully describe atmospheric flows, which are characterized by many wavelengths, many more expansion functions would be necessary. Limiting the model to waves of only one wavelength, however, simplifies the solution considerably. This set of functions does, however, allow one to study the non-linear interactions between waves of the same wavelength in the fields of the stream functions for the mean flow (or mean height) and the thickness. The amplitudes of the waves and the phase difference between the waves may be varied to simulate aspects of various atmospheric flow patterns.

The expansion function F_o is used to specify the mean height of the field. It is irrelevant to the present problem, however, and will be dropped. As proof of its irrelevance, Eqn. (3.14) demonstrates that with the chosen expansion functions, $c_{\ell jk} = 0$ if any of $\ell = 0$,

$j = 0$, or $k = 0$. Eqn. (3.15) shows that $b_{\ell i} = 0$ if $\ell = 0$ or $i = 0$. Thus, the mean of the field does not interact with the perturbation in the field and it is, therefore, not important. Consequently, the expansion coefficients associated with F_o will be assumed to be 0.

For the chosen domain and the three expansion functions, F_1, F_2 and F_3, Eqn. (3.15) gives:

$$b_{1\ell} = 0 \quad \text{for } \ell = 1,2,3 \tag{3.27}$$

$$b_{23} = -b_{32} = -2\pi n/L_x \tag{3.28}$$

Eqn. (3.14) shows that $c_{\ell jk}$ is non-zero only when the three indices are distinct and one finds that:

$$\bar{c}_{321} = -\frac{32\sqrt{2}\pi n}{3L_x L_y} \tag{3.29}$$

The dependent variables, $\bar{\psi}$, $\hat{\psi}$ and $\tilde{\omega}_2$, are expanded using the expansion functions to obtain:

$$\bar{\psi} = \psi_1 F_1 + \psi_2 F_2 + \psi_3 F_3 \tag{3.30}$$

$$\hat{\psi} = \tau_1 F_1 + \tau_2 F_2 + \tau_3 F_3 \tag{3.31}$$

$$\tilde{\omega}_2 = \omega_1 F_1 + \omega_2 F_2 + \omega_3 F_3 \tag{3.32}$$

Finally, Eqns. (3.23) - (3.32) are substituted into Eqns. (3.20) - (3.22) to obtain a system of nine ordinary differential equations in the nine expansion coefficients. These equations are non-dimensionalized as this will give a simpler form for the final equations. Thus, in the equations, an arbitrary expansion coefficient for the mean flow, ψ_i, is replaced by $L^2 f_o \psi_i$. This new coefficient is non-dimensional. Similarly, τ_i is replaced by $L^2 f_o \tau_i$; ω_i is replaced by $\Delta p f_o \omega_i$, and d/dt is replaced by f_o d/dt.

Simple algebraic manipulation allows one to rewrite the resulting nine non-dimensional equations as a system of six ordinary differential equations and three diagnostic omega equat

$$\frac{d\psi_1}{dt} = 0 \tag{3.33}$$

$$\frac{d\psi_2}{dt} = n^2\nu\mu(\psi_1\psi_3+\tau_1\tau_3) - \beta^* n\mu\psi_3 \tag{3.34}$$

$$\frac{d\psi_3}{dt} = - n^2\nu\mu(\psi_1\psi_2+\tau_1\tau_2) - \beta^* n\mu\psi_2 \tag{3.35}$$

$$\frac{d\tau_1}{dt} = - \frac{\omega_1}{\alpha^2} \tag{3.36}$$

$$\frac{d\tau_2}{dt} = n^2\nu\mu(\tau_1\psi_3+\tau_3\psi_1) - \beta^* n\mu\tau_3 - \mu\omega_2 \tag{3.37}$$

$$\frac{d\tau_3}{dt} = - n^2\nu\mu(\tau_1\psi_2+\tau_2\psi_1) + \beta^* n\mu\tau_2 - \mu\omega_3 \tag{3.38}$$

$$\omega_1 = \frac{-\nu}{\sigma^* + \frac{1}{\alpha^2}} (\tau_2\psi_3 - \tau_3\psi_2) \tag{3.39}$$

$$\omega_2 = \frac{1}{\sigma^* + \mu}\left(\nu\{n^2\mu(\tau_1\psi_3 + \tau_3\psi_1) - (\tau_3\psi_1 - \tau_1\psi_3)\} - \beta^* n\mu\tau_3\right) \tag{3.40}$$

$$\omega_3 = \frac{1}{\sigma^* + \mu}\left(\nu\{(\tau_2\psi_1 - \tau_1\psi_2) - n^2\mu(\tau_1\psi_2 + \tau_2\psi_1)\} + \beta^* n\mu\tau_3\right) \tag{3.41}$$

where the following symbol definitions have been used:

$$\alpha = L_x/L_y \tag{3.42}$$

$$L = L_x/2\pi \tag{3.43}$$

$$\gamma = 8\sqrt{2}/3\pi \tag{3.44}$$

$$\beta^* = \beta L/f_o \tag{3.45}$$

$$\sigma^* = \frac{\Delta p^2 \sigma}{2f_o^2 L^2} \tag{3.46}$$

$$\nu = \gamma\alpha n \tag{3.47}$$

$$\mu = 1/(\alpha^2 + n^2) \tag{3.48}$$

Thus, the spectral method requires the solution of six non-linear ordinary differential equations, Eqns. (3.33) - (3.38), and three omega equations, Eqns. (3.39) - (3.41), for the nine expansion coefficients. Baer (1970) has found an analytical solution to these equations to be expressible in terms of elliptic functions. The analytical solution was not used for this thesis. Instead, a numerical solution was found because a highly accurate solution could be found numerically with considerably less computational effort than an evaluation of the analytical solution would require.

Equations (3.33) - (3.38) have the general form:

$$\frac{df}{dt} = g(t) \tag{3.49}$$

In order to integrate such an equation numerically, it is written in finite-difference form. For the first time step, an Euler forward difference is used:

$$f^1 = f^0 + \Delta t g^0 \tag{3.50}$$

where f^o and g^o are the initial values of the functions f and g and Δt is the time step. Ralston (1965) shows that this scheme has first-order accuracy. For subsequent time steps, the second-order Adams-Bashforth method is used:

$$f^{n+1} = f^n + \frac{\Delta t}{2}(3g^n - g^{n-1}) \tag{3.51}$$

Lilly (1965) found the Adams-Bashforth method to be simple and efficient while giving accuracy on a par with more complicated methods when he compared 8 methods for integrating equations similar to Eqns. (3.37) - (3.42).

As the spectral solution is to be used as an accurate solution with which to compare the finite-element solution, the minimization of time truncation error is very important. Haltiner (1971) indicates that the time step must be a "small" fraction of the period of variation of the spectral amplitudes. As synoptic-scale motions are being studied, the period of the spectral amplitudes is expected to be on the order of days. Thus, the qualitative criterion of Haltiner (1971) indicates that a time step on the order of hours is appropriate. Test integrations were done with time steps of 1 hour and 0.5 hours. After 24 hours of integration, the solutions with the different time steps were compared. The spectral amplitudes were found to vary by less than 1 part in 10^4. These results suggest a time step of 0.5 hours is sufficiently small and this is used in all spectral integrations presented in Chapter 5.

The initial conditions are determined by choosing values for ψ_1^o, ψ_2^o, ψ_3^o, τ_1^o, τ_2^o and τ_3^o. In choosing these values, one determines not only the amplitudes of the waves but also the phase relationship between the waves in the stream function for the mean height, $\bar{\psi}$, and for the thickness, $\hat{\psi}$. This allows one to simulate a variety of atmospheric flow patterns. The initial conditions for the cases presented in Chapter 5 were chosen to represent a variety of flows. This is done so that the solution by the finite-element method could be tested with a variety of initial conditions in order to determine its possible strengths or weaknesses.

3.3 The Energy and Potential Enstrophy Relations

The energies and potential enstrophy discussed in Sec. 2.3 can be related to the expansion coefficients. Using the definition of the kinetic energy, Eqn. (2.36), and Eqns. (3.30) and (3.31), one finds that:

$$KE = \frac{2L_xL_y\Delta pL^4f_o^2}{g}\left\{ \frac{1}{L_y^2}(\psi_1^2+\tau_1^2) + \left(\frac{1}{L_y^2}+\frac{n^2}{L_x^2}\right)(\psi_2^2+\psi_3^2+\tau_2^2+\tau_3^2) \right\} \quad (3.52)$$

where non-dimensionalized coefficients are used. Using Eqn. (2.38), the available potential energy of the spectral solution is found to be given by:

$$APE = \frac{\Gamma L_xL_yf_o^2L^4}{4}(\tau_1^2+\tau_2^2+\tau_3^2) \quad (3.53)$$

The potential enstrophy for the spectral solution, using Eqns. (2.41), (2.42) and (2.43), is found to be:

$$PE = \frac{\Delta pL_xL_yf_o^2}{2g}\Big(\{\alpha^2\psi_1\}^2 + \{(\alpha^2+n^2)\psi_2\}^2 + \{(\alpha^2+n^2)\psi_3\}^2 +$$

$$\{(\alpha^2+1/\sigma^*)\tau_1\}^2 + (\alpha^2+n^2+1/\sigma^*)\{\tau_2^2+\tau_3^2\}\Big) \quad (3.54)$$

Differentiating Eqns. (3.52) - (3.54) with respect to time and substituting Eqns. (3.33) - (3.41) in, it is found that the total energy and the potential enstrophy are conserved by the exact spectral solution. Eqns. (3.52) - (3.54) allow the calculation of these quantities at each step of the integration. The conservation of these quantities was used as a method of verifying the accuracy of the spectral numerical solution. In practice, it was found that these quantities were conserved to better than 1 in 10^4 for a wide variety of initial conditions and lengths of integrations. In particular, it is true for all cases presented in Chapter 5.

CHAPTER 4

THE FINITE-ELEMENT METHOD

4.1 A Description of the Method

Since its initial development by structural engineers during the 1950's (e.g. Turner et. al., 1956; Argyris, 1960; Clough, 1960), the finite-element method (FEM) has become a popular means of finding approximate solutions to initial and/or boundary value problems. In recent years, (Wang, et. al., 1972; Cullen, 1976, 1979; Staniforth and Daley, 1978, 1979), the FEM has been applied to the initial-boundary-value problems of numerical weather prediction. This chapter describes the application of the FEM to the atmospheric model described in Chapter 2. A brief history of the method is given first and a discussion of the FEM follows, with a simple example used to clarify the discussion. Finally, the application of the FEM to the atmospheric model under consideration is presented.

It is difficult to determine the originator of the FEM, although Clough (1960) seems to have been the first to use the name. Until the early 1960's, the method was developed separately by mathematicians and engineers. On the engineering side, the FEM evolved from the matrix method of structural analysis (Zienkiewicz, 1977). In this method, the analysis of structures proceeded by considering the components of the structures separately. Relations between the displacements and internal forces at the nodal points of individual components were derived in matrix form with the displacements and/or the forces being unknown. (The nodal points are the places where the components were joined.) The solution for the unknowns proceeded by solving the system of equations, written in matrix form, for the unknowns.

This method provided the exact answer for the unknowns. The only assumption made was that mathematical relations could be used to describe real physical systems. By analogy, McHenry (1943) and Newmark (1949), for example, extended the matrix method to continuum problems, i.e. problems without easily identifiable components. They divided the continuum into a number of hypothetical components called elements. They then proceeded as before by writing a system of equations for the nodal displacements and forces and solving this system of equations. This was found to give a good approximate solution to the original continuum problem.

As the method evolved, it was found that the simplest procedure to ensure that the forces and displacements of the approximate solution represented accurately the true solution, was to introduce the concept of virtual work (Zienkiewicz, 1977). An arbitr[illegible] .e. virtual) nodal displacement is imposed and the internal and external work done by the various forces and stresses during that displacement are equated. It was recognized that this approach was equivalent to minimizing the total potential energy of the structure under consideration. Argyris (1960), for example, detailed the resulting matrix equations, for a rectangular panel under plane stress, in a comprehensive paper on energy theorems and matrix methods

During the late 1950's and early 19 s (e.g. Szmelter, 1959; Clough, 1965), this method was recognized as an extension of the Rayleigh-Ritz method, which had been well known to mathematicians since the publications of Rayleigh (1870) and Ritz (1909). With this recognition, it became possible to give a mathematical basis to the largely intuitive developments of the engineering profession. During the last two decades, the use of the FEM has grown rapidly and a bibliography of the FEM by Norrie and deVries (1976) provides a detailed listing of many of the developments.

When applying the FEM, the domain of the problem is divided into a number of elements. The number, size and shape of the elements are chosen after consideration of the domain of the problem and the degree accuracy required. Zienkiewicz (1977) discusses some of the possible choices for elemental shapes. Triangular and rectangular shapes have been popular as they are relatively easy to work with. There are no hard and fast rules for choosing the elements although there are a few guidelines. For a domain with a curved boundary, triangular elements offer an advantage as they can better approximate the boundary. For rectangular domains, rectangular elements may be advantageous as fewer elements are normally required for a given level of accuracy. Also, increasing the total number of elements and decreasing the size of the elements provides increased accuracy. One has the option of using small elements in areas of detailed interest while using larger elements elsewhere. When all the elements are chosen for a domain, the result is called the mesh or grid of the domain. It is advantageous to automate the choice of this mesh for a given elemental shape and domain.

In this thesis, rectangular elements are used, and Fig. 4.1 shows the mesh used for the domain defined in Sec. 2.1. The domain is divided into unequal rectangles. The central portion of the mesh has high resolution with the elements having a length of 200 km. on a side. The element length and width varies uniformly away from this central portion. Thus, the ratio of elemental lengths between neighbouring elements is a constant. The elemental length reaches a maximum of 1200 km. and the elemental width reaches a minimum of 60 km. The high resolution portion near the north and south boundaries is used to permit a more accurate implementation of the boundary conditions, Eqns. (2.33) and (2.34). A variable grid length was used so that its effect on the solution could be investigated. The portions of the grid marked A and B in Fig. 4.1 are the portions of the domain in which the FEM solution will be presented (in Chapter 5) for comparison with the spectral solution. The portion marked C is the portion on which the energies and the potential enstrophy of the two solutions will be calculated.

The elemental shape must be chosen in conjunction with the basis functions, $e^i(x,y)$, to be used and the number of nodes per element. The basis functions form an interpolatory basis with which a function may be interpolated on the domain. They are normally chosen to be low-order polynomials and are defined in a piece-wise sense on the domain, i.e. the i'th basis function is non-zero only over some (small) portion of the domain adjacent to element i. The choice of piece-wise defined basis functions is a major strength of the FEM. It leads to systems of linear equations which may be written in matrix form. These matrices are normally highly sparse and banded. This gives a significant computational advantage over systems that are nearly full or not banded.

In general, when interpolating a function, one needs the basis functions and some values of the function being interpolated. In the FEM, these values are the value of the function at the nodes. The nodes are specified points on the domain and the number of nodes chosen per element determines the order of the basis functions. Thus, with rectangular elements, if four nodes are chosen per element, the basis functions are bi-linear while the use of sixteen nodes per element would be consistent with bi-quadratic basis functions. Although the nodes are often chosen to be on the boundary of the elements, they may be in the interior of the element. The number of nodes per element, the nodal positions and the basis functions must be chosen together. The reader is referred to Zienkiewicz (1977) and Tong and Rossettos (1977) for examples of elements and their associated nodes and basis functions which have been used.

In this thesis, four nodes per element are used and the nodes are chosen to be in the corners of the elements. Each node is common to four elements.

Bi-linear basis functions are used. One may define the basis functions with respect to each element or, equivalently and more simply in this case, one may define them with respect to the nodes. If the nodes are numbered m = 1, 2, . . ., NI in the x-direction and n = 1, 2, . . ., NJ in the y-direction, the basis function for node (m,n) may be written:

$$e^{\ell}(x,y) = \begin{cases} \left(\frac{x-x_{m-1}}{h_{m-1}}\right)\left(\frac{y-y_{n-1}}{k_{n-1}}\right) & ;\ x\varepsilon\left(x_{m-1},x_m\right),\ y\varepsilon\left(y_{n-1},y_n\right) \\ \left(\frac{x_{m+1}-x}{h_m}\right)\left(\frac{y-y_{n-1}}{k_{n-1}}\right) & ;\ x\varepsilon\left(x_m,x_{m+1}\right),\ y\varepsilon\left(y_{n-1},y_n\right) \\ \left(\frac{x-x_{m-1}}{h_{m-1}}\right)\left(\frac{y_{n+1}-y}{k_n}\right) & ;\ x\varepsilon\left(x_{m-1},x_m\right),\ y\varepsilon\left(y_n,y_{n+1}\right) \\ \left(\frac{x_{m+1}-x}{h_m}\right)\left(\frac{y_{n+1}-y}{k_n}\right) & ;\ x\varepsilon\left(x_m,x_{m+1}\right),\ y\varepsilon\left(y_n,y_{n+1}\right) \\ 0 & ;\ \text{otherwise} \end{cases} \quad (4.1)$$

where:

$$h_m = x_{m+1}-x_m$$
$$k_n = y_{n+1}-y_n \quad (4.2)$$

and ℓ is the multi-integer (m,n). The vector $[X_1, X_2, \ldots, X_{NI}]$ contains the values of the x-coordinate at the nodes while $[Y_1, Y_2, \ldots Y_{NJ}]$ contains the values of the y-coordinate.

Thus, e^{ℓ} (x,y) is non-zero only over the four rectangles which have node ℓ in common. Also, e^{ℓ} (x,y) has a value of 1 at the node and diminishes to a value of zero at the other nodes of those four rectangles. With this definition, these basis functions are almost orthogonal, i.e. they interact only locally. Thus, the integral defined by:

$$I = \iint_D e^{\ell}(x,y)\ e^{k}(x,y)\ dxdy \tag{4.3}$$

where ℓ is the multi-integer (m,n) and k is the multi-integer (M,N), is non-zero only if m = M - 1, M or M + 1 and n = N - 1, N or N + 1. Finally, it should be noted that the basis functions are separable, i.e. they may be written e^{ℓ} (x,y) = e^{m} (x) e^{n} (y). se basis functions may be used to interpolate a function, g (x,y), on a domain in the following way:

$$g(x,y) \approx \sum_i g_i e^i(x,y) \tag{4.4}$$

where the summation extends over all nodes and the g_i are the nodal values of g (x,y).

The FEM is, in fact, a general class of methods and one must choose the particular method(s) to be used for a given problem. Zienkiewicz (1977) provides a discussion of some of the methods which have been used. Two large sub-classes of the FEM are the variational and the weighted residual approaches. An example of a method from each of these sub-classes is considered in this thesis. First, the Rayleigh-Ritz method (a variational approach) and then the Galerkin method (a weighted residual approach) will be discussed. The Galerkin method is the one actually used for the solution of the model equations.

To illustrate the Rayleigh-Ritz method, let us consider the two-dimensional Poisson equation:

$$\nabla^2\phi = f(x,y) \tag{4.5}$$

valid on a domain D with the boundary condition:

$$\phi(x,y) = \bar{\phi} \tag{4.6}$$

on the boundary, Γ, of the domain where $\bar{\phi}$ is a constant. A variational principle may be written:

$$I(\phi) = \frac{1}{2}\iint_D (\nabla\phi)^2 \, dxdy + \iint_D f(x,y)\phi \, dxdy \tag{4.7}$$

where $(\nabla\phi)^2 = (\partial\phi/\partial x)^2 + (\partial\phi/\partial y)^2$. Tong and Rossettos (1977) have shown that the function, ϕ, which minimizes Eqn. (4.7) and satisfies Eqn. (4.6), is also the solution of Eqns. (4.5) and (4.6). Thus, the solution of the original differential equation is also the extremum of the variational principle.

In the Rayleigh-Ritz method, an approximate solution to Eqns. (4.5) and (4.6) is found by minimizing an approximate form of the variational principle, Eqn. (4.7). The functions ϕ and f are expanded as in Eqn. (4.4) and are substituted into Eqn. (4.7). This yields an approximate form of the variational, $I^* (\phi)$, and this is minimized with respect to changes in each of the nodal values of ϕ, i.e.

$$\frac{\partial I^*}{\partial \phi_i} = 0 \qquad ; \; i=1,2,\ldots,N \tag{4.8}$$

where N is the total number of interior nodes. This procedure yields a system of linear equations which may be written in matrix form as:

$$A\underline{\phi} = B\underline{f} \tag{4.9}$$

where A and B are square N x N matrices and $\underline{\phi}$ and $\underline{f}$ are vectors of the nodal values of their respective functions. Given the nodal values of f, one may find the nodal values of ϕ by inverting A in Eqn. (4.9). This is the approximate solution to Eqns. (4.5) and (4.6).

Many boundary-value problems, however, do not have a corresponding variational principle and other methods must be used to find solutions to them. The Galerkin method is one of these. The present method is an extension of that given by Galerkin (1915). In the general weighted residual approach, the differential equation, Eqn. (4.5), is multiplied by a test function, t (x,y), and the product is integrated over the domain to yield:

$$\iint_D t\nabla^2\phi \; dxdy = \iint_D tf \; dxdy \tag{4.10}$$

In the Galerkin method, the test function is chosen to be one of the basis functions.

If the derivatives in Eqn. (4.10) are of higher order than are the basis functions, it is necessary to integrate Eqn. (4.10) by parts. In two-dimensions, this can be accomplished using Green's Theorem. Following Zienkiewicz (1977), Eqn. (4.10) may be written:

$$-\iint_D \left(\frac{\partial t}{\partial x}\frac{\partial \phi}{\partial x} + \frac{\partial t}{\partial y}\frac{\partial \phi}{\partial y}\right) dxdy = \iint_D tf \; dxdy - \oint_\Gamma n_x t\frac{\partial \phi}{\partial x}\, d\Gamma - \oint_\Gamma \ell_y t\frac{\partial \phi}{\partial y}\, d\Gamma \tag{4.11}$$

where n_x and l_y are the direction cosines between the outward normal to the boundary and the x and y axes respectively. With the boundary condition Eqn. (4.6), the last two terms of Eqn. (4.11) are zero.

Now, ϕ and f are expanded using Eqn. (4.4) and substituted into Eqn. (4.11). Upon integration, a system of linear equations similar to Eqn. (4.9) is obtained and the approximate solution may be found by inverting the new matrix A. Strang and Fix (1973) have shown that the Galerkin and Rayleigh-Ritz methods give the same system of equations if the problem has a variational form.

The right-hand-side of Eqn. (4.5) has, to this point, been treated as a simple function, f (x,y). In fact, as in the model equation (2.26) for example, the right-hand-side may be a rather complicated expression. The nodal values of this expression must be evalu[illegible]ive the nodal values of f) so that the solution may proce[illegible]Galerkin or Rayleigh-Ritz methods. Staniforth and Da[illegible]ovide a detailed explanation of how this expression may be evaluated and the following brief explanation is based on their work.

The evaluation of complicated right-hand-sides of equations such as Eqn. (2.26) is most easily understood by breaking the procedure up into a number of smaller steps. In general, the right-hand-side involves the addition or subtraction of functions, the differentiation of functions and the products of functions. Each of these will be considered in turn.

The addition or subtraction of functions proceeds node by node. Thus, if the nodal values f_i and g_i of the functions f (x,y) and g (x,y) are known, the sum or difference, b (x,y) = f (x,y) $\pm$ g (x,y), is found using:

$$b_i = f_i \pm g_i \tag{4.12}$$

for all the nodes, i = 1, 2, . . ., N.

The differentiation of functions will be considered for a one-dimensional domain first. Let us consider the determination of a multiple of the first derivative of a function, $u(x)$, in the x-direction:

$$v = \alpha u_x \tag{4.13}$$

where $u_x \equiv du/dx$ and α is a scalar constant. It is assumed that u is known at the nodal points x_i, $i = 1, 2, \ldots, N$ and that v is required at these points. The two functions, u and v, are expanded using the basis functions, $e^i(x)$, and substituted into Eqn. (4.13) to obtain:

$$v_i e^i(x) = u_i e^i_x(x) \tag{4.14}$$

where $e^i_x(x) = d\, e^i(x)/dx$ and where the Einstein summation convention for repeated indices is used. The Galerkin procedure is applied by multiplying by each of the basis functions successively and integrating over the domain, $[x_1, x_N]$, to obtain

$$v_i \int_{x_1}^{x_N} e^k(x) e^i(x)\, dx = \alpha u_i \int_{x_1}^{x_N} e^k(x) e^i_x(x)\, dx \tag{4.15}$$

for $k = 1, 2, \ldots, N$. This may be written in matrix form as:

$$P\underline{v} = \alpha P_x \underline{u} \tag{4.16}$$

where:

$$\underline{v} = (v_1, v_2, \ldots\ldots, v_N)^T$$

$$\underline{u} = (u_1, u_2, \ldots\ldots, u_N)^T$$

$$p^{ki} = \int_{x_1}^{x_N} e^k e^i \, dx \tag{4.17}$$

$$p_x^{ki} = \int_{x_I}^{x_N} e^k e_x^i \, dx$$

p^{ki} is the (k,i) element of an $\underline{N}$ x N matrix and $(\quad)^T$ is the transpose of that vector.

As the basis functions are nearly orthogonal, the matrices P and P_σ are tri-diagonal. Staniforth and Daley (1978) show that P is diagonally dominant and that the solution of this system of equations by Gaussian elimination is, therefore, stable with respect to round-off error. As a consequence of the separability of the basis functions, Staniforth and Daley (1978) find that a partial derivative in a multi-dimensional field may be obtained by taking the appropriate ordinary derivative in the direction defined by the partial derivative.

Consider now the evaluation of the product of two functions, u (x) and v (x), defined, again for the sake of simplicity, on a one-dimensional domain:

$$w = \alpha uv \tag{4.18}$$

where α is a scalar constant. The three functions, u, v and w, are expanded using the basis functions and the Galerkin procedure is again applied to obtain:

$$w_i \int_{x_1}^{x_N} e^k(x)e^i(x)\ dx = \alpha u_i v_j \int_{x_1}^{x_N} e^k(x)e^i(x)e^j(x)\ dx \tag{4.19}$$

which is valid for k = 1, 2, . . ., N. The left-hand-side will be recognized as the matrix P defined by Eqn (4.17). The right-hand-side of Eqn. (4.19) is more complicated than before and an efficient method for evaluating it is necessary. The left-hand-side is evaluated using numerical integration and, as the integrand is cubic in x, Simpson quadrature, a quadrature formula which is exact for cubics, is used. Over the range [x_n, x_{n+1}], it is found that:

$$\int_{x_1}^{x_{n+1}} e^n(x)u(x)v(x)\ dx = \frac{(x_{n+1}-x_n)}{6}\ (u_n v_n + \tfrac{1}{2}\{u_n+u_{n+1}\}\{v_n+v_{n+1}\}) \tag{4.20}$$

and that:

$$\int_{x_n}^{x_{n+1}} e^{n+1}(x)u(x)v(x)\ dx = \frac{(x_{n+1}-x_n)}{6}\ (u_{n+1}v_{n+1} + \frac{1}{2}\{u_n+u_{n+1}\}\{v_n+v_{n+1}\}) \quad (4.2.)$$

For a multi-dimensional in, Staniforth and Daley (1978) show that product terms like 8) may be reduced to weighted sums of products evaluated on one-dimensional domain. That is, one numerically integrates in one d ion and then integrates along the other direction.

Thus, the results which are necessary to evaluate the nodal values of f on the right-hand-side of Eqn. (4.9) have been established for bi-linear basis functions on a two-dimensional domain. In Section 4.2, we will consider the application of the FEM to the model equations, Eqns. (2.26), (2.29) and (2.30). The nodal values of the right-hand-sides of these equations were evaluated using the methods of this section.

4.2 The Application of the FEM to the Model Equations

The model equations, Eqns. (2.26), (2.29) and (2.30), are of the form:

$$\nabla^2\phi - \lambda^2\phi = f(x,y) \qquad (4.22)$$

which is a Helmholtz equation. Staniforth and Mitchell (1977) have shown that the variational principle corresponding to Eqn. (4.22) is

$$I(\phi) = \iint_D \left((\nabla\phi)^2 + \lambda^2\phi^2 + 2f\phi \right) dxdy \qquad (4.23)$$

given the boundary condition Eqn. (2.33). Here, $(\nabla\phi)^2 = (\partial\phi / \partial x)^2 + (\partial\phi / \partial y)^2$.

Staniforth and Mitchell (1977, 1978) have shown that with the basis functions defined by Eqn. (4.1), the approximate minimization of I (ϕ) in Eqn. (4.23), using the Rayleigh-Ritz method, yields a solution with second-order accuracy on any sub-domain with uniform grid spacing. They discuss an alternate approach using the Galerkin method which yields a fourth-order solution on any uniform sub-domain. This is the method used in this thesis.

Applying the Galerkin method to Eqn. (4.22) yields:

$$\iint_D \nabla^2\phi e^k(x)\, dxdy - \lambda^2 \iint_D \phi e^k(x)\, dxdy = \iint_D f e^k(x)\, dxdy \qquad (4.24)$$

where λ is assumed to be a constant over the domain. If ϕ and f are expanded using the bi-linear basis functions, the second-order solution obtainable from the Rayleigh-Ritz method is found. Eqn. (4.24) may be written:

$$\int_{x_{i-1}}^{x_{i+1}} \int_{y_{j-1}}^{y_{j+1}} (\phi_{xx}+\phi_{yy}) e^k \, dxdy - \lambda^2 \int_{x_{i-1}}^{x_{i+1}} \int_{y_{j-1}}^{y_{j+1}} \phi e^k \, dxdy = \int_{x_{i-1}}^{x_{i+1}} \int_{y_{j-1}}^{y_{j+1}} f e^k \, dxdy \quad (4.25)$$

as the basis function e^k for the node (i,j) is non-zero only over the four rectangles adjoining that node.

To demonstrate the technique for finding the fourth-order solution, consider the element which has the node (i,j) in its lower left hand corner, i.e. for which we have $x_i \le x \le x_{i+1}$ and $y_j \le y \le y_{j+1}$. For this element, the right-hand-side of Eqn. (4.25) may be written:

$$R = \int_{x_i}^{x_{i+1}} \int_{y_j}^{y_{j+1}} f e^k \, dxdy \quad (4.26)$$

Following Staniforth and Mitchell (1978), f is assumed to be symmetric about x_i, y_j and R is therefore evaluated as one-quarter of the doubly symmetric integral over four times the area, viz:

$$R = \frac{1}{4} \int_{x_i-H_i}^{x_i+H_i} \int_{y_j-K_j}^{y_j+K_j} f e^k \, dxdy \quad (4.27)$$

where $H_i = x_{i+1} - x_i$ and $K_j = y_{j+1} - y_j$. Note that in Eqn. (4.27), e^k must be re-defined for the new assumed grid and that f is expanded in terms of this new e^k. This interpolate for f is substituted into Eqn. (4.27) and the integration is performed yielding an expression for R in terms of the nodal values of f. Similar integrations of the right-hand-side of Eqn. (4.25) are performed over the three remaining rectangles adjoining (i,j). A similar procedure is undertaken to obtain the left-hand-side, but an integration by parts is necessary first. For the node (i,j), this yields Eqn. (4.28), which Staniforth and Mitchell (1978) show has fourth-order accuracy on any uniform sub-domain. (Eqn. (4.28) is on page 54.)

An equation similar to Eqn. (4.28) may be derived for all interior nodes. These equations form a system of linear equations in the nodal values of the functions ϕ and f. To complete this system of equations, it is necessary to impose the boundary conditions.

Denoting the north boundary with j = 1 and the south boundary with j = NJ, the first boundary condition, Eqn. (2.32), gives:

$$\phi_{1,1}=\phi_{2,1}=\cdots\cdots=\phi_{NI,1}\equiv\phi_{B,1} \tag{4.29}$$

and:

$$\phi_{1,NJ}=\phi_{2,NJ}=\cdots\cdots=\phi_{NI,NJ}\equiv\phi_{B,NJ} \tag{4.30}$$

$$\frac{1}{12}\left(\phi_{i+1,j+1}\left(\frac{k_j}{h_i}+\frac{h_i}{k_j}\right)+\phi_{i-1,j+1}\left(\frac{k_j}{h_{i-1}}+\frac{h_{i-1}}{k_j}\right)+\phi_{i-1,j-1}\left(\frac{k_{j-1}}{h_{i-1}}+\frac{h_{i-1}}{k_{j-1}}\right)+\phi_{i+1,j-1}\left(\frac{k_{j-1}}{h_i}+\frac{h_i}{k_{j-1}}\right)\right.$$

$$+\phi_{i+1,j}\left(5\frac{k_j}{h_i}-\frac{h_i}{k_j}+5\frac{k_{j-1}}{h_i}-\frac{h_i}{k_{j-1}}\right)+\phi_{i,j+1}\left(-\frac{k_j}{h_i}+5\frac{h_i}{k_j}-\frac{k_j}{h_{i-1}}+5\frac{h_{i-1}}{k_j}\right)+\phi_{i,j-1}\left(-\frac{k_{j-1}}{h_{i-1}}+5\frac{h_{i-1}}{k_{j-1}}-\frac{k_{j-1}}{h_i}\right.$$

$$\left.+5\frac{h_i}{k_{j-1}}\right)+\phi_{i-1,j}\left(5\frac{k_j}{h_{i-1}}-\frac{h_{i-1}}{k_j}+5\frac{k_{j-1}}{h_{i-1}}-\frac{h_{i-1}}{k_{j-1}}\right)-5\,\phi_{ij}\left(\frac{k_j}{h_i}+\frac{h_i}{k_j}+\frac{k_{j-1}}{h_i}+\frac{h_i}{k_{j-1}}+\frac{k_{j-1}}{h_{i-1}}+\frac{h_{i-1}}{k_{j-1}}+\frac{k_j}{h_{i-1}}+\frac{h_{i-1}}{k_j}\right)$$

$$-\frac{\lambda^2}{144}\left(h_ik_j\phi_{i+1,j+1}+h_{i-1}k_j\phi_{i-1,j+1}+h_{i-1}k_{j-1}\phi_{i-1,j-1}+h_ik_{j-1}\phi_{i+1,j-1}+5k_j(h_i+h_{i-1})\phi_{i,j+1}\right.$$

$$\left.+5h_i(k_j+k_{j+1})\phi_{i+1,j}+5h_{i-1}(k_j+k_{j-1})\phi_{i-1,j}+5k_{j-1}(h_i+h_{i-1})\phi_{i,j-1}+25\{h_i(k_j+k_{j-1})+h_{i-1}(k_j+k_{j-1})\}\phi_{ij}\right)$$

$$=\frac{1}{144}\left(h_ik_jf_{i+1,j+1}+h_{i-1}k_jf_{i-1,j+1}+h_{i-1}k_{j-1}f_{i-1,j-1}+h_ik_{j-1}f_{i+1,j-1}+5k_j(h_i+h_{i-1})f_{i,j+1}\right.$$

$$\left.+5h_i(k_j+k_{j-1})f_{i+1,j}+5h_{i-1}(k_j+k_{j-1})f_{i-1,j}+5k_{j-1}(h_i+h_{i-1})f_{i,j-1}+25(h_i+h_{i-1})(k_j+k_{j-1})f_{ij}\right)$$

Equation (4.28)

In order to impose the second boundary condition, Eqn. (2.33), the integrand in Eqn. (2.33) is expanded using the basis functions. At the northern boundary, this gives:

$$\left.\frac{\partial\phi}{\partial y}\right|_{i,1} = \frac{\phi_{i,2}-\phi_{i,1}}{k_1} \tag{4.31}$$

and at the southern boundary, it yields:

$$\left.\frac{\partial\phi}{\partial y}\right|_{i,NJ} = \frac{\phi_{i,NJ}-\phi_{i,NJ-1}}{k_{NJ-1}} \tag{4.32}$$

for $i = 1, 2, \ldots, NI$. In Eqns. (4.31) and (4.32), $\phi = \partial\psi/\partial t$. Eqn. (2.33) is of the form:

$$\int_0^{L} g \, dx = 0 \tag{4.33}$$

Expanding g with the bi-linear basis functions and integrating Eqn. (4.33), one obtains:

$$(h_{NI-1}+h_1)g_{1,j}+(h_1+h_2)g_{2,j}+\ldots\ldots+(h_{NI-2}+h_{NI-1})g_{NI-1,j} = 0 \tag{4.34}$$

Substituting Eqns. (4.29) - (4.31) into Eqn. (4.34) and noting that in Eqn. (2.34), $g = \partial\phi/\partial y$, one obtains:

$$\phi_{B,1} = \frac{1}{2L_x}\Big(\phi_{1,2}(h_{NI-1}+h_1) + \phi_{2,2}(h_1+h_2) + \ldots\ldots$$

$$+ \phi_{NI-1,2}(h_{NI-2}+h_{NI-1})\Big) \qquad (4.35)$$

where $\phi_{B,1}$ is the value of ϕ at all nodal points along the northern boundary. Similarly, the expression for the nodal values on the southern boundary, $\phi_{B,NJ}$, is found to be given by:

$$\phi_{B,NJ} = \frac{1}{2L_x}\Big(\phi_{1,NJ-1}(h_{NI-1}+h_1) + \phi_{2,NJ-1}(h_1+h_2) + \ldots\ldots$$

$$+ \phi_{NI-1,NJ-1}(h_{NI-2}+h_{NI-1})\Big) \qquad (4.36)$$

If the vectors $\underline{\phi}$ and $\underline{f}$ are defined to be vectors of all (both interior and boundary) nodal values of their respective functions, Eqns. (4.35) and (4.36) may be combined with all equations similar to Eqn. (4.28) to obtain a complete system of linear equations. This may be written in matrix form as:

$$A\underline{\phi} = B\underline{f} \qquad (4.37)$$

An approximate solution to the Helmholtz equation, Eqn. (4.22), with the boundary conditions given by Eqns. (2.32) and (2.33), may be found by solving the matrix equation, Eqn. (4.37). The procedure for doing thi is described in Section 4.4.

4.3 The Energy and Potential Enstrophy Relations

The evaluation of the kinetic energy, the available potential energy, and the potential enstrophy for the finite-element solution is done by approximating Eqns. (2.36), (2.38) and (2.43) with equations which are numerically integrated. These three equations have the form:

$$Z = \alpha \int\int_D f(x,y)\ dxdy \qquad (4.38)$$

where α is a constant and z represents the KE, PE or APE. nodal values of the integrand, f (x,y), are evaluated using the results of Section 4.2. This is done using Eqns. (2.36), (2.38) and (2.43) to define the integrand for the ..., APE and PE respectively. In order to perform the integration, Eqn. (4.38) is approximated by a simple finite-difference representation:

$$Z \approx \alpha \sum_i \sum_j f_{ij} \left(\frac{h_i + h_{i-1}}{2}\right) \left(\frac{k_j + k_{j-1}}{2}\right) \qquad (4.39)$$

where the f_{ij} are the nodal values of the integrand, f (x,y).

The limits of the summations in Eqn. (4.39) must be defined. In the definitions of KE, APE and PE in Section 2.3, we considered the integration to occur over the entire domain. Thus, the summation is with $i = 1, 2, \ldots, NI - 1$ and $j = 1, 2, \ldots, NJ$. Within these limits, the conservative properties of the solution by the FEM are checked and the results are presented in Chapter 5.

It is also of interest to calculate the KE, APE and PE in the region where the solution by the FEM has the greatest accuracy, i.e. the sub-domain with uniform resolution. T will enable a better judgement of the potential accuracy of the solution using the FEM. For the cases presented in Chapter 5, this is done for the area marked C in Fig. 4.1. These calculations are done for the spectra olution as well so the two solutions can be compared. To do this, it was necessary to calculate the nodal values of the spectral solution.

4.4 The Solution Algorithm

The five steps for the solution of the model equations (Eqns. (2.26), (2.29) and (2.30)) using the FEM are:

1) to determine the right-hand-sides of the model equations.
2) to solve the system of equations resulting from the discretization of Eqns. (2.26) and (2.29) for $\partial\bar{\psi}/\partial t$ and $\partial\hat{\psi}/\partial t$, respectively, at time t.
3) to extrapolate in time to estimate $\bar{\psi}$ and $\hat{\psi}$ at time $t + \Delta t$, where Δt is the time step.

4) to solve the system of equations resulting from the discretization of Eqn. (2.30) for $\tilde{\omega}_2$ at $t + \Delta t$.

5) to repeat steps one through four for the desired number of time steps.

The procedure used for performing the first step was described in Section 4.3. The solution of a system of linear equations, as required by steps 2 and 4 may be done using either direct or iterative methods. Tong and Rossettos (1977) provide a discussion of the advantages and disadvantages of each method when applied to finite-element problems. Their analysis indicates that, for the systems of equations in this thesis, a direct method would probably be most efficient in the use of computer resources. However, as the computer code was to be used a limited number of times, the efficient use of computer resources was judged to be less important than the efficient use of programming time. Hence, an iterative method is used.

Young (1...) describes many iterative methods and from these, successive over-relaxation (SOR) is chosen as it is relatively simple to program and is reasonably efficient in the use of computer resources. To apply SOR, the matrix A in Eqn. (4.37) is split into a lower triangular, L, and an upper triangular, U, matrix. Then, given an initial guess for the nodal values, $\underline{\phi}^o$, the approximate solution, $\underline{\phi}^n$, is found using

$$L\Delta\underline{\phi} = B\underline{f} - A\underline{\phi}^{n-1} \qquad (4.40)$$

$$\underline{\phi}^{n} = \underline{\phi}^{n-1} + \omega\Delta\underline{\phi} \qquad (4.41)$$

where ω is the over-relaxation factor. The solution proceeds by iterating with Eqn. (4.41) and (4.42) and the superscript n refers to the number of the iteration. One continues iterating until the difference between successive iterates, $\Delta\underline{\phi}$, is sufficiently small. In the present thesis, iterations are continued until the relative change from one iteration to the next at all nodes in the high-resolution sub-domain (Area C) is less than .1%.

The total number of iterations needed for a solution depends critically on the choice of the value for the over-relaxation factor. An attempt was made, using techniques described by Young (1971), to objectively determine the optimum over-relaxation factor, i.e. the one which resulted in the minimum number of iterations for a solution. This did not work well, however, and a factor of 1.795 was found, by trial and error, to be nearly optimum.

Step 3 was accomplished using the Euler and Adams-Bashforth methods described in Section 3.2. The length of the time step, Δt, for the solution by the FEM was chosen using different criteria than in Section 3.2 however. Haltiner (1971) shows that to prevent computational instability, the fastest travelling wave in the solution must move less than one grid interval, Δx, in one time step, Δt. This leads to the following condition:

$$\frac{C\Delta t}{\Delta x} \leq \frac{1}{\sqrt{2}} \tag{4.42}$$

for a two-dimensional domain where C is the speed of the fastest travelling wave. This criterion was derived using finite-difference methods rather than finite-element methods for the discretization of the space coordinates. However, the physical interpretation of this criterion presented by Haltiner (1971) suggests that it could at least be used as a guideline for the present work. For synoptic-scale systems, in quasi-geostrophic models where sound and gravity waves are not permitted, Holton (1972) estimates 50 m/s to be the maximum value of C. With this value and a time step of 1/2 hour, Eqn. (4.42) indicates that computational instability should be prevented in the present model for all portions of the grid with grid spacing greater than 127 km. Thus, only in a very narrow band, near the north and south boundaries of the present grid, is the possibility of computational instability indicated. A time step of 1/2 hour is used as it will mean that the time truncation errors in both methods of solution are the same and because the criterion have suggested that computational instability is unlikely to be a major problem.

CHAPTER 5

THE RESULT

5.1 Introduction

The model equations derived in Chapter 2 are solved using both the spectral and the finite-element methods, as discussed in Chapters 3 and 4. In this chapter, some examples of the solutions obtained using these methods, with various initial conditions, will be presented. In the accompanying discussion, some of the strengths and weaknesses of the finite-element solution will be demonstrated through a comparison with the spectral solution. Prior to presenting these results, a discussion of some of the conventions used and the numerical values of certain constants will be given.

A total of four sets of initial conditions will be presented with both methods of solution used for all four. Cases I, II and III are model atmospheres which, initially, favour the conversion of APE to KE. This conversion leads to a development (amplification) of the wave in the mean height field. The cases will be referred to as weakly, moderately, and strongly developing cases. Case IV is a model atmosphere in which a conversion of KE to APE takes place. The wave in the mean height field decays and this is called the decaying case.

As shown in Chapter 3, the spectral solution allows only waves of wavenumber n in the x-direction to exist. This wavenumber must be chosen and its value determines the scale of motion to be considered. As mid-latitude synoptic scale motions are to studied, a wavenumber of 3 to 9 is appropriate. For the chosen channel length, these wavenumbers lead to waves with wavelengths of the order of thousands of kilometers. Holton (1972) suggests that wavenumber 7 is close to the average wavenumber of mid-latitude synoptic systems. He also shows that this wave is near the wavelength of maximum baroclinic instability, i.e. the wavelength which becomes baroclinically unstable with the lowest thermal contrast. A wavenumber of 7 is used in all the cases in this chapter.

The static stability parameter, σ, is defined by Eqn. (2.16). It is calculated by writing Eqn. (2.16) in finite-difference form using a central difference about 500 mb. Tabulated values of θ and p for the ICAO standard atmosphere were used to find $\sigma = 2.8 \times 10^{-6}$ $m^4\ s^2\ kg^{-2}$. This is within 25% of the value assumed by Holton (1972) for mid-latitude synoptic scale systems.

Teweles and Wobus (1954) developed the S1 score for comparing two fields with their values determined at grid points. This score is used later in this chapter as an aid in comparing the solutions by the two methods. The S1 score compares the gradients of the fields rather than their magnitudes, for example. Such a comparison is significant meteorologically because the gradient in the stream function determines the wind speed and direction, through the geostrophic wind relation, Eqn. (2.7). The wind and the vertical wind shear are more important weather elements than the actual value of the streamfunction at a point, because they determine the potential for baroclinic and barotropic growth of weather systems.

To calculate the S1 score, one first calculates the gradients in the two fields which are being compared. In the present work, the components of the gradients of the two fields in the x and y directions are calculated for all points in the verifying area (area C of Fig. 4.1) including the boundaries of that area. If e_G is defined to be the difference between the gradient [illegible] the two fields at a point and G_L is defined to be the larger o[illegible] gradients, the S1 score [illegible] calculated using the formula:

$$S1 = 100 \frac{\Sigma |e_G|}{\Sigma |G_L|} \qquad (5.1)$$

where the summation occurs over all points in the verifying area and both components of the gradients are compared at each point. The S1 score varies from 0 to 100 with lower values indicating a greater agreement of the two fields.

The mean difference (MD) and mean absolute difference (MAD) between the finite-element and spectral solutions are also calculated over the verification area. In both the MD and the MAD, the finite-element solution is subtracted from the spectral solution at all grid points within area C, including the boundaries. The mean used is the arithmetic mean.

During initial test runs, it was found that waves of short wavelength were being generated along the northern and southern boundaries when the finite-element solution was computed. These waves grew in amplitude and moved away from the boundary, gradually contaminating the solution by the FEM even in the high-resolution sub-domain. Investigations indicated that a major cause of these waves was the poor evaluation of the vorticity near the boundary. In particular, the evaluation of the second derivative of the stream function in the y-direction was subject to a large error near the boundary. A simple 3-point smoothing algorithm was introduced to damp these waves. Thus, if the function f (x) is known at the grid points $(x_1, x_2, \ldots, x_N)$, the smoothed function, g (x), at the point x_i is given by:

$$g(x_i) = (1 - S(x_i))f(x_i) + \frac{S(x_i)}{2} (f(x_{i+1}) + f(x_{i-1})) \qquad (5.2)$$

where $S(x_i)$ is the smoothing parameter at x_i. This smoother was applied to the calculated vorticity field in the vicinity of the northern and southern boundaries. It was applied to the vorticity field each time it was evaluated, first in the y-direction and then in the x-direction. Although most smoothing was necessary in the y-direction, some smoothing in the x-direction was found to be helpful in controlling the spurious waves near the boundary. For the six grid points nearest the boundary, the smoothing parameters were .30, .50, .30, .15, .08, and .02. The largest values were near the northern and southern boundaries. The smoothing parameter was zero for all other interior grid points. The smoothing parameters for the first six interior lines of grid points in the x-direction, adjacent to and including the northern and southern boundaries, were .10, .15, .10, .05, .01 and .005.

The values for the smoothing parameters were determined by trial and error. This set of parameters seemed to have little effect on the long wavelengths of interest but controlled the spurious short wavelengths reasonably well. It was found that using this smoothing also reduced the amount of computation time by 10 - 15% as it allowed more rapid convergence to occur in the solution of the boundary value problems by relaxation.

In displaying the height and thickness fields later in this chapter, the stream functions for the mean height and thickness fields have been converted to a height in meters. This was easily done as, in this model, the gravitational constant, g, is assumed to be independent of height. Following Holton (1972), the height in meters, z, is obtained using:

$$z = \psi \frac{f_o}{g} \tag{5.3}$$

where $g = 9.806\ m\ s^{-2}$. The stream function, ψ, is either $\bar{\psi}$ or $\hat{\psi}$, depending on which field is being converted.

The omega fields displayed have the units μbar/sec. This is the unit which is commonly used as it gives values of the order of 1. It should be noted, however, that negative omega values imply ascent.

In the display of these fields, roughly two-thirds of the total domain in Fig. 4.1 is shown. Referring to Fig. 5.1, for example, the lower panel is area A of Fig. 4.1 while area B of Fig. 4.1 is the upper panel. With this display, both the high and low resolution portions of the two solutions may be easily compared.

As previously stated, area C of Fig. 4.1 is the verification area. This is the sub-domain over which the various energy quantities, the PE, the S1 score, the MD and the MAD are calculated, as shown, for example, in Figs. 5.10 - 5.12. The KE, APE and PE are also calculated over the whole domain for some of the cases.

5.2 Case I - Weak Development

Case I is an example of a weakly developing situation. Figs. 5.1 and 5.2 show that, initially, the trough in the thickness field lags (i.e. is west of) the trough in the mean height field by 90°. When the trough in the thickness field lags the trough in the height field by between 0° and 180°, the perturbations (i.e. the waves) in the mean height field will amplify. In this process, the flow becomes more meridional and the amplitude of the wave increases. Holton (1972) shows that this amplification will occur most rapidly when the lag is 90°. Holton (1972) did a linear analysis of an atmospheric model similar to the one under consideration. He found that if the gradients in the thickness field were less than a critical value, the waves would not develop, i.e. they were stable waves. With the values used in the present study, his analysis suggests the wave should be stable for this case. Due to model differences and non-linear interactions, the wave does, in fact, develop.

As was discussed in Section 2.1, this growth is due to the baroclinic conversion of energy from APE to KE through the lowering of the centre of mass of the atmosphere. This will be demonstrated more clearly when, later in this section, Fig. 5.10a is discussed.

The spectral solution (SP) after twenty-four hours of integration is given in Figs. 5.4 - 5.6. The wave in the height field has undergone the expected development and retrogressed 15°. The thickness field has developed slightly and progressed eastward 10°. Thus, the phase difference between the two fields has been reduced by 25°. The omega field has amplified somewhat during the period and has retrogressed 22°.

Figs. 5.1 - 5.2 are also the initial conditions used for the solution by the FEM. Figs. 5.7 - 5.9 are the resulting twenty-four hour forecasts for the height, thickness and omega fields, respectively. Only a very detailed comparison of the forecasts produced by the two methods reveals any differences between them over the high-resolution portion of the grid. Near the boundaries, it is apparent that the troughs and ridges in the height field of the finite-element solution (FE) have retrogressed slightly faster than those of the SP.

The diagnostic omega fields for the two solutions, Figs. 5.6 and 5.9, are considerably different, however. The maxima in the omega field are larger in the FE and are closer to the southern boundary in the FE than in the SP. The minima of the FE behave in an analogous manner, i.e. they are more negative and are closer to the northern boundary than are those of the SP. Whereas the SP has a value very near 0.0 at the boundary, the boundary value of the FE is somewhat larger than 0.0, in absolute magnitude. Investigations revealed that this remarkable difference in the two omega fields was present even at the initial time. Thus, this difference is not due to the development and growth of errors during the integration but rather due to the evaluation of the diagnostic omega field of the FE. Using the boundary condition, Eqn. (2.32), one may readily show that the right-hand-side of Eqn. (2.30) should be zero on the boundary, for the chosen initial conditions. However, since the estimate of the vorticity, $\overline{Q}$, is very poor on the boundary, the term $J(\hat{\psi},\overline{Q})$ is subject to a large error on the boundary. When the boundary value of the omega field is in error, the interior values of the omega field are also in error. The poor estimation of the vorticity near the boundary also causes an error in the evaluation of the right-hand-sides of Eqns. (2.26) and (2.29). However, this error is not as easily seen in these cases as one is dealing with an error in the forecast of a relatively small change (i.e. $\partial\overline{\psi}/\partial t$) in a large quantity (i.e. $\overline{\psi}$).

The evaluation of the boundary condition, Eqn. (2.33), also makes a contribution to the errors near the boundaries. This boundary condition requires an integration along the boundary. Thus, values from both the high and low resolution portions of the grid are used to find the boundary condition for the entire boundary. This effectively increases the truncation error in the x-direction, along the entire boundary, to that of the portion of the boundary with the lowest resolution.

The resolution in the y-direction is very high near the boundary. This band with high resolution along the boundaries was used to try to reduce the errors along the boundaries. With such resolution, a better estimate of the vorticity on and near the boundary can be obtained. In addition, the evaluation of the boundary condition, Eqn. (2.33), has greater accuracy as the truncation error present in the discretization of $\partial\psi/\partial y$ is reduced. Because of this area of high resolution, however, a shorter time-step must be used in the integration to prevent computational instability, as was discussed in Chapter 4.

Cullen (1976), Grammeltvedt (1969) and others have used different initial conditions when performing numerical experiments with this model on this domain. Their initial conditions were characterized by fields with very little variation in the stream functions near the boundary. In effect, then, there was a buffer of 600 - 800 km. between the boundary and the region of interest. This approach was not used in the present work, however, because the highly accurate spectral solution would not have been available for comparison.

Fig. 5.10a shows the evolution of the TE, KE and APE in the verification area (area C of Fig. 4.1) for the two solutions during the forecast period. The TE of the SP increases by approximately 30% during the period of integration. This is not, however, in violation of the conservation of TE, which was demonstrated in Chapter 3, as this TE has been calculated only in the verification area and not over the whole domain. As noted in Chapter 3, the conservation of the TE of the SP over the whole domain was verified for a wide variety of initial conditions. The increase of the TE of the SP in area C is due to the motion of the fields and the development of the fields during the forecast period.

Referring to Fig. 5.10a, we see that the gain in TE of the SP is due to a gain in KE while the APE has remained nearly constant. Thus, during the development of the waves, APE has been converted to KE but also, in the verification area, there has been energy imported. It is not possible to say what proportion of the gain in KE is due to importation and what proportion is due to development.

Fig. 5.10a confirms that, with respect to the three energy quantities, the FE has performed well. The TE of the FE has decreased relative to the SP in the verification area. The FE has lost approximately 2% of the TE in the verification area. This small error is seen to be due to a loss in the APE of the FE relative to the SP, while it is noted that the KE of the FE is extremely close to that of the SP. The growth rate of the error seems to be nearly linear, with the error first being noticeable after approximately ten hours of integration.

Fig. 5.10b indicates that the FE has also lost a small amount of PE relative to the SP. The total loss is only .4%. Careful scrutiny of Fig. 5.10b reveals that the FE actually gained PE relative to the SP for the first twelve hours and then lost PE for the last 12 hours. The small variation of the PE of the FE about that of the SP indicates that the FE behaves quite well in this case.

In Fig. 5.11a, the KE of the two solutions is presented in the zonal and eddy forms, while Fig. 5.11b gives the corresponding values for the APE. As noted earlier, the KE of the two solutions is essentially the same. However, the FE has lost ZKE relative to the SP and gained an equal and opposite amount of EKE. There is roughly a 1% difference in the ZKE between the two solutions. The loss in the APE of FE relative to the SP is seen to be due to a loss of ZAPE while the EAPE of the two solutions is essentially the same.

The reasons for the differences betwee he two solutions, as shown in Figs. 5.10 and 5.11, are rather hard to determine precisely. As mentioned previously, there are errors in the FE due to space and time discretization. Also, it has been shown previously that, as a result of space discretization, significant errors can occur near the boundary due to the imperfect implementation of the boundary condition. As the time discretization scheme is the same for both models, it is expected that this, by itself, makes only a very small contribution to the differences between the two solutions. It seems probable, however, that the loss in ZAPE of the FE relative to the SP is due to the poor evaluation of the boundary condition since the boundary value plays a relatively larger role in the determination of the zonal quantities than of the eddy quantities. The error in the ZKE of the FE is thought to be smaller because the boundary condition on the height field is satisfied better than the one for the thickness field. In this model, the boundary value of the height field should not change with time but that of the thickness field does. It was found that the forecast change in the thickness field on the boundary in the FE was subject to a large error but that the FE maintained the boundary value of the height field fairly well. It is to be noted that the thickness field determines the APE whereas the KE is determined by both the thickness and height fields. The small errors in the eddy forms of the energy quantities are thought to be due to a combination of the various errors. It is probable that, during the initial few hours, errors due to space discretization are most important. As the integration proceeds, errors from the boundary will gradually infiltrate the high-resolution portion of the domain and become more important.

Fig. 5.12a gives the S1 scores for the FE relative to the SP. There is nearly a linear growth of the S1 score for both the height and thickness fields with the S1 score for the height field growing more rapidly. In the development of this score, Teweles and Wobus (1954) indicated that, in the forecasting of the real atmosphere, an S1 score of under 30 could be considered a perfect forecast while an S1 score of over 70 would indicate a useless forecast. For the present integration, the S1 score has only grown to near 5 which indicates, according to Teweles and Wobus's subjective criteria, an exceptional forecast.

However, care must be exercised here. In the present study, a highly simplified model atmosphere is used and only a very limited number of types of waves are allowed, in contrast to many types of wave motion possible in the real atmosphere. Also, in the present study, the initial conditions are very well specified and this is not true of forecasts for the real atmosphere where the initial conditions must be specified by some objective analysis of actual reported data. Thus, it seems reasonable to assume that the S1 scores for the present study should be significantly lower than what might be expected of forecasts of the real atmosphere. It is difficult, however, to objectively determine how small the score should be to conclude that a "good" forecast has been generated. The differences which were seen between the two forecasts were very small and this, at least, gives one the impression that the FE was very close to the SP and that, therefore, S1 scores of about 5 or less indicate a good forecast. With the presentation of the remaining cases, an approximate upper limit for the S1 score representing a good forecast may be found.

Fig. 5.12b shows that, in the mean over the verification area, the height field of the SP is 0.24 m higher than that of the FE and that the thickness of the SP is 0.08 m lower than that of the FE. Considering that the fields under consideration have magnitudes of nearly 100 m and that some points in the height field underwent changes of nearly 30 m during the forecast period, these MD errors are quite respectable. The MD for the height field is larger than that for the thickness field because there were larger changes in the height field during the forecast period.

Fig. 5.12b also gives the MAD between the two solutions in the verification area. The MAD for both fields grows to nearly 1.10 m during the period of integration. The MAD for the thickness fiel[illegible]s nearly linear growth while that for the height field appears to be closer to exponential growth.

It is interesting to note that while the MD for the thickness field was smaller, in absolute value, than that for the height field, the MAD for the thickness field is larger than that for the height field. Thus, the errors in the thickness field are smaller than those of the height field but there is a greater bias in the errors of the thickness field than in the height field. In effect, these results suggest that although the error at individual points is, in general, smaller in the thickness field than in the height field, the mean of the thickness field is not being kept constant as well as is that of the height field. It is not possible to suggest the precise cause of this at this time.

An interesting comparison may be made between Figs. 5.12 and 5.11. Fig. 5.11 shows that the KE of the FE is closer to that of the SP than is the APE. Fig. 5.12a shows that the gradient of the thickness field is forecast well while in Fig. 5.12b, we have seen that the forecast magnitude of the thickness field has a bias in it. The KE depends on the gradient of the thickness field while the APE depends on the magnitude of the thickness field. Given that the ZAPE is in error by the largest amount, it is suggested that the errors in the forecast magnitudes of the thickness field are produced largely by a poor implementation of the boundary condition.

5.3 Case II - Moderate Development

Case II is a situation favourable for greater development than Case I. Figs. 5.13 and 5.14 show that, initially, the trough in the thickness field lags the trough in the mean height field by 90^{o}. The thickness field for this case is very similar to that of Case I, but the wave in the height field is characterized by a much larger amplitude for Case II than for Case I. As a result, the amplitude of the omega field is larger for this case, as shown in Fig. 5.15.

With the larger meridional gradient in the height field in this case, a large mean zonal wind exists in the domain. This means that the wave in the mean height field is expected to move faster than in Case I. Also, the larger meridional variation in the mean height field implies greater thermal advection in this case. This increases the rate of baroclinic development. Thus, both the rate of propagation and rate of development of the wave in the mean height field are expected to have been changed in Case II.

Figs. 5.16 - 5.18 give the mean height, thickness and omega fields, respectively, after 48 hours of integration with the spectral method. (The length of integration has been increased as this will provide a more stringent test on the conservative properties and of the accuracy of the FE. This will also allow a better investigation of the possibility of computational instability in the FE.) The wave in the mean height field has developed and progressed eastward approximately 15^{o}. The wave in the thickness field has decayed and progressed eastward approximately 45°. Thus, the phase difference between the waves in the two fields has been reduced to 60°. The wave in the omega field has decayed slightly and progressed approximately 20°.

Figs. 5.19 - 5.21 show the mean height, thickness and omega fields, respectively, for a 48 hour forecast using the FEM. The differences between the two solutions are much more apparent in this case than in Case I. Referring to Figs. 5.16 and 5.19, it is apparent that the FE has not maintained the north-south anti-symmetry present in the SP. The troughs in the FE have undergone greater development than those of the SP while the ridges of the FE have not been built as high as those of the SP. This seems to be due to problems with the boundary values of the FE. The SP has maintained a constant value for the mean height field along the boundaries while the boundary value of the FE has changed. Although any of the errors previously discussed could be the cause of this, it is probable the main cause is the poor evaluation of the vorticity near the boundary. This, as discussed in Sec. 5.2, was found to be a major source of error near the boundary.

Related to the above problem is the curvature in the trough and ridge axes of the FE. The SP has trough and ridge axes which are north-south while those of the FE are slightly curved. It appears that the speed of these waves in the FE is slower than those of the SP near the boundaries but is faster than those of the SP in the central portion of the domain. These errors are due, either directly or indirectly, to the problems near the boundaries. The overdevelopment of the troughs of the FE has led to larger zonal wind speeds in the trough in the middle of the domain. This leads to a more rapid progression of the trough axis in the central portion of the domain. Similarly, larger gradients in the height field near the ridge axes in the centre of the domain has caused a more rapid progression of the ridge axes. Near the boundaries, just the opposite has happened. Gradients have been reduced and the progress of the waves has been retarded.

Finally, the presence of a few small amplitude short wavelength waves in the height field of the FE near the boundaries should be noted. These are the waves which the smoother, discussed in Sec. 5.1, was designed to control. Also of note is the absence of these waves in the region of the domain where the grid of the FE has very poor resolution. The short wavelength waves could not be resolved in this portion of the domain.

Comparing Figs. 5.17 and 5.20, it is apparent that there are vast differences between the two solutions for the thickness field. The FE has very nearly maintained the correct value for the maximum value of the highs and the minimum value of the lows. However, the maxima and minima of the FE are much closer to the boundaries than are those of the SP. This has been caused by the problems with the boundary value of the FE. For this case, the boundary value of the SP has changed very little during the integration period while the boundary value of the FE has definitely changed during this period.

In the central portion of the grid, the FE is within a few percent of the SP however. The zero thickness isopleth is forecast very well by the FE near the centre of the grid even in those regions with relatively poor resolution in the dire : . The phase speeds in the central portion are nearly ide . the meridional amplitude of the zero isopleth.

The short wavelength waves are more notic able in the thickness field of the FE than they are in the height field. Their amplitudes are small however. The waves are not distinct enough in either field to judge whether these short wavelength waves are in a position favourab e for development or not.

The omega fields of the two solutions, shown in Figs. 5.18 and 5.21, are drastically different. The trough and ridge axes of the FE have a distinct curvature to them whereas those of the SP are straight lines oriented north-south. This curvature is simply a reflection of the curvature of the axes in the height and thickness fields of the FE found previously. The maxima/minima of the FE are more positive/ negative than are those of the SP. This is indicative of the greater development which the FE has undergone. Also, the maxima and minima of the FE are located nearer to the boundaries than are those of the SP.

There are numerous spurious short wavelength waves present in the omega field of the FE. Although they are most common and have their largest amplitudes near the boundaries, they are also present well into the domain. These waves in the interior of the domain were not visible in the height and thickness fields but they must have been there or they would not be present in the omega field. Thus, some small amplitude short wavelength waves have propagated well into the grid.

Fig. 5.22a shows the TE, APE, and KE in the verification area for this case. The TE of the SP rises during the integration period for the reasons discussed with Case I. However, the TE of the FE rises faster than that of the SP. After 48 hours, the TE of the FE is approximately 6% larger than that of the SP. This increase in TE is due largely to the increased gradients in the mean height field of the FE which were seen earlier. The gain in the TE confirms that the wave in the mean height field of the FE has developed more than the wave in the SP. Also, it should be noted that the divergence in the TE of the two solutions does not begin until after twenty-four hours.

Further study of Fig. 5.22a reveals that the gain in the TE of the FE relative to the SP is due to a large gain in the KE of the FE relative to the SP which is only partially compensated for by a loss in the APE of the FE relative to the SP. The divergence of the KE and APE of the two solutions begins after only twelve hours of integration. During the period from twelve to twenty-four hours, the KE and APE of the FE, although in error with respect to the SP, adjust themselves so that the TE of the two solutions is the same. Thus, during the first twelve hours, the two solutions behave similarly. During the twelve to twenty-four hour period, the FE converts APE to KE at a faster rate than does the SP. During the final 24 hours, the FE either imports energy into the verification area or it has created some spurious source of energy. It seems probable that there has been importation of energy. In the FE relatively large amounts of KE have been concentrated in the middle portion of the domain, while the KE near the boundaries is lower in the FE than the SP. This can be seen by comparing the gradients in the mean height field of the two solutions. A calculation of the meridional momentum transport might have given further insight here but this was not thought of until after the computation was finished.

Fig. 5.22b shows that the PE of the FE is less than that of the SP during the forecast period. The PE in the verification area changes significantly during the period and the FE under-forecasts the change in PE by nearly 20%. This result is a little surprising as, with the greater development of the FE, one would expect larger vorticities in the FE than in the SP. However, the troughs and ridges are sharper in the SP than in the FE and, thus, the vorticity of the FE is lower. Also, the thickness field of the FE has smaller values over the verification area than does the SP. These effects both act to give the FE a lower PE than the SP.

In Fig. 5.23a, it is seen that the FE gains a significant amount of ZKE relative to the SP while losing a small amount of EKE relative to the SP during the forecast period in the verification area.

The difference appears in the ZKE after fifteen hours and in the EKE after twenty-four hours. The aforementioned meridional transport of east-west momentum towards the centre of the domain in the FE has presumably led to this increase in the ZKE in the verification area. It seems that EKE in the FE has been converted to ZKE as well. This is reflected in the relatively broad troughs in the FE. This, I believe, is a natural result of the process of baroclinic development. During development, ZAPE is converted to EAPE; EAPE is converted to EKE; and, finally, EKE is converted to ZKE, as discussed by Holton (1972). The wave in the FE is at a more advanced state of development than the wave in the SP. Physically, the wave in the FE has presumably begun to slow down its rate of growth and soon, it is expected the wave will undergo the process of occlusion and begin to decay.

Fig. 5.23b shows that, in fact, the APE of the two solutions is not the same during the initial twelve hours as suggested when Fig. 5.22a was discussed. The resolution of that figure was not high enough to see that the FE continuously loses APE relative to the SP. It is important to note, however, that in this case, the APE is much smaller than the KE. Thus, for this case, a 10% error in the APE has the same effect on the TE as approximately a 1% error in the KE, i.e. the error in the APE is a less important measure of the performance of the FE than is the KE.

The loss of APE in the FE relative to the SP is seen, in Fig. 5.23b, to be due to a loss of both ZAPE and EAPE. During development, there is a conversion of ZAPE to EAPE. The FE loses ZAPE slightly faster initially than the SP does but the difference in the ZAPE between the two solutions remains relatively small However, after twelve hours, the FE begins to lose EAPE rapidly with respect to the SP. It seems that energy in the FE, during the last thirty-six hours of development, is being rapidly transferred from ZAPE through the EAPE and EKE forms to ZKE, i.e. the FE has an accelerated transfer of energy during the baroclinic development. The result is a loss of ZAPE, EAPE and EKE in the FE relative to the SP with only the ZKE of the FE larger than that of the SP. The error in the ZKE begins to grow most rapidly a few hours after the error in the EAPE begins its rapid growth. This suggests that there is a causal relationship between the two errors. The relationship is somewhat clouded by the possibility of either net energy importation to or exportation from the verification area.

Fig. 5.24a shows that the S1 score for the thickness field has grown more rapidly than for the height field. The growth of the S1 score for both fields is approximately twice as fast as for Case I. The faster growth of the S1 scores in Case II may be due either to the faster growth or to the greater motion in Case II. In both cases, the field which underwent the greatest change had the larger S1 score. The final S1 score for the height field is near 18. In the preceding discussion of the various energy quantities, it was found that errors began to grow rapidly after twenty-four hours. After thirty hours, large differences between the two solutions were common in the energies. This suggests that an S1 score of less than ten or twelve indicates a "good" forecast. This is a subjective criterion and further discussion will be presented with the final cases.

Fig. 5.24b gives the MD and MAD curves for the two models for this case. Most notable is the large growth of the MAD between the mean height fields. This is indicative of the greater development present in the FE. The MD between the mean height fields grows to approximately 7 meters. This indicates that the areas of low height have developed faster than the highs have built, in agreement with what was observed earlier. In contrast, the MD between the thickness fields stays relatively small and negative. The MAD between the thickness fields rises to almost 7 m. These results indicate that although there are large errors in the thickness field of the FE, there is very little bias towards either raising or lowering the mean of the field in the verification area.

In Fig. 5.25a, the KE, APE and TE of both solutions over the entire domain are presented. The TE of the SP changes by approximately 7% during the forecast period. The TE of the SP should be conserved and this error is due to the method used for calculating the TE here. For this figure, the values of the SP at the grid points were found and a numerical integration was performed, as described in Section 4.3. Thus, this error is due to the discretization of the SP and the truncation errors in the numerical integration. In fact, as discussed in Section 3.3, it was confirmed that the SP conserved TE to better than 1 in 10^4. During the forecast period, the TE of the FE changes by less than 3%. Considering that the SP was subject to a 7% error due to the discretization process, the FE has conserved TE quite well. The maximum difference between the TE of the two solutions is only 3%. The KE of the two solutions is essentially identical but the FE does gain APE relative to the SP. This is probably due to the problems with the boundary value of the thi ness field. During the integration period, this has caused a gain of 3% in the TE for the entire domain.

Fig. 5.25b gives the PE of the two solutions for the entire domain. Once again, there is a small error in the PE of the SP due to the numerical integration process. More important, however, is the striking gain in the PE of the FE after twenty-four hours. This large amount of PE is found in the spurious short wavelength waves generated near the boundaries. These waves were found to have vorticities which were an order of magnitude larger than vorticities elsewhere in the domain. It is apparent that the smoother which was used was not adequate to control the development and growth of these waves. It is probably pure coincidence that the error in the PE of the FE begin to grow most rapidly at approximately the same time the forecast in the verification area was found to degrade rapidly. The PE of the FE was found to be smaller than that of the SP right to the end of the integration. This indicates there was little or no contamination of the forecast in the verification area by the short wavelength waves. However, the exponential growth of the PE indicates the potential for future problems if the integration had proceeded further in time.

5.4 Case III - Strong Development

Case III is a situation favorable for strong development. Figs. 5.26 - 5.28 give the initial height, thickness and omega fields, respectively. The height and thickness fields are characterized by large gradients and the troughs in the thickness field lag those of the height field by 90°. The thermal advection, which is essential for baroclinic development, is considerably greater than in the previous cases. This will promote strong development. The initial omega field has relatively large values for synoptic-scale vertical motion, indicating the vigour of the situation under consideration.

It was intended to integrate this case for forty-eight hours but computational instability occurred in the FE and the integration had to be terminated after thirty-two hours. Fig. 5.29 shows the mean height field of the SP after twenty-four hours. The wave has developed and progressed eastward approximately 30°. Fig. 5.30 shows the thickness field for the SP after twenty-four hours. The wave has progressed approximately 50°. The meridional amplitude of the thickness field has increased which implies the conversion of ZAPE to EAPE during the forecast period. On Fig. 5.31, we see that the amplitude of the omega field has decreased by approximately one-fourth and that the wave has progressed approximately 40°.

Figs. 5.32 and 5.33 show the mean height and thickness fields of the FE after twenty-four hours. The differences in the mean height fields of the two solutions are similar to those present in Case II. The trough and ridge axes are tilted. There has been preferential development in the northern half of the domain and spurious short-wavelength waves are present near the boundaries again. The thickness fields differ in many of the ways found in Case II. Once again, the high and low centres of the FE have migrated towards the boundaries.

Figs. 5.34 and 5.35 show the mean height and thickness fields of the FE after thirty hours. It is apparent tha some of the short-wavelength waves, which were present along the boundaries, have grown explosively and overwhelmed the wave of interest, thereby ruining the forecast. This displays the effect of numerical computational instability. By thirty-three hours, the amplitude of these waves had grown so large that an overflow condition was obtained on the computer.

In Fig. 5.36a, we see that the FE has gained TE relative to the SP, as in Case II. As with Case II, the error is quite small after twenty-four hours of integration. The FE has, once again, gained KE relative to the SP but lost APE. After twenty-four hours, the FE in both Case II and Case III has nearly maintained the TE in the verification area through equal but opposite errors in the EKE and APE. However, the fields in Case III have undergone greater development and faster motion than those in Case II. Thus, it seems the growth of errors is not very dependent on either the rate of development or speed of the fi- ds.

In Fig. 5.36b the PE of the two solutions is presented. The error in the PE of the FE is approximately 5%, which is nearly identical to the error in the PE in Case II after twenty-four hours.

In Fig. 5.37, it will be seen that the zonal and eddy forms of the KE and APE of the FE behave in a similar manner to those in Case II. However, the errors are larger than in Case II. For example, the error in the ZKE of the FE after twenty-four hours is only 7% in Case II but is over 30% in Case III. Thus, although the error in the TE of the two cases had similar errors, the individual elements making up the TE are subject to greater errors in this case. In Case II, it was suggested that the FE underwent an accelerated development relative to the SP. The results of this section indicate that this has occurred in this case also, and, that the rate of accelerated development depends, at least somewhat, on the rate of development.

Fig. 5.38 shows the S1 scores and the MD and MAD curves for this case. Using the subjective criteria suggested in Section 5.3, the S1 scores suggest the forecast was "good" for approximately sixteen hours. The MD and MAD curves exhibit similar events to those of Case II. The errors have grown faster in this case.

5.5 Case IV - Moderate Decay

In Case IV, the troughs in the thickness field lead the troughs in the mean height field by 90°, as the initial conditions given in Figs. 5.39 - 5.41 show. The initial amplitudes of the waves are the same as in Case II. With such a phase relationship between mean height and thickness, the wave in the mean height field decays and there is a conversion of KE to APE. This is the opposite of what happened in the previous cases.

Figs. 5.42 - 5.44 give the mean height, thickness and omega fields, respectively, for the SP after forty-eight hours of integration. The wave in the mean height field has decayed in amplitude and has retrogressed approximately 12°. The wave in the thickness field has also decayed in amplitude and has progressed 70°. At this time, the waves are almost 180° out of phase. They appear to be in the process of adjusting themselves so that their phase becomes favourable for development. The omega field appears to have changed very little in either amplitude or position during the forecast period.

Figs. 5.45 - 5.47 give the three fields for the FE after forty-eight hours. The mean height field of the FE and the SP are very similar over the high-resolution portion of the grid. In fact, near the centre line of the domain, the FE is barely distinguishable from the SP until the grid spacing in the x-direction approaches 800 km. However, short-wavelength waves are present near the boundaries and those near the north boundary have not decayed as rapidly as those near the south boundary. Comparing this with the results of previous cases, it indicates there is a predilection for development/decay near the north/south boundary. This is probably due to problems with either the calculation of various quantities (e.g. vorticities) near the boundary or due to the implementation of the boundary condition. It seems probable that this bias is related to the sign of the height field at the boundary, i.e. the positive/negative boundary wants to become more positive/negative. The thickness field exhibits a large amount of curvature in the trough and ridge axes in this case. The precise cause of this is unknown. However, the major cause of change in the thickness field in this case was motion as opposed to decay. Thus, it seems reasonable to expect the source of the problem to be connected to the motion of the field.

The omega field of the FE has a larger amplitude than does that of the SP. This is partially due to the causes discussed in Section 5.1. A secondary cause is the increased thermal advection in the FE over the SP due to the errors in the height and thickness fields of the FE.

In Fig. 5.48a, it can be observed that the FE has gained TE relative to the SP during the forecast period in the verification area. This also occurred in Case II and is likely indicative of the bias with respect to development at the boundaries. The slc decrease in the TE of the SP simply indicates there is a net export of energy from the verification area, whereas a net import was found for the developing cases. The APE of the solutions is very nearly the same during the period but the FE has gained KE relative to the SP. This was the cause of the gain in the TE of the FE in the developing cases also. The APE changes very little during the period suggesting there is very little decay in the verification area, or that an importation of APE to the verification area has taken place.

Fig. 5.48b shows that the FE loses PE relative to the SP during the first thirty-six hours of integration but gains it back during the last twelve hours. The maximum difference between the PE of the two solutions is less than 4%.

Fig. 5.49a shows that the gain of KE by the FE is due to a gain of ZKE. The EKE of the two solutions is very similar throughout the forecast period. The error in the ZKE of the FE occurs early in the forecast period and grows slowly during the period to a maximum of 14%.

According to Fig. 5.49b, the FE loses a small amount of APE relative to the SP during the forecast period. The error is very small during the first thirty hours of integration as there are roughly equal and opposite errors in the ZAPE and EAPE. However, the errors in the ZAPE and EAPE stay relatively small throughout the forecast period.

Fig. [illegible] gives the S1 scores and the MD and MAD curves for this case. The S1 score rises rapidly for the thickness field. This is further confirmation of the errors which evolved in the FE during the rapid movement of this field. The S1 score for the height field rises at a much slower rate. Near the end of the period, the S1 score for the height field begins to rise more rapidly. It seems possible that this is a reflection of the poor thickness forecast as the two fields are connected through the model equations.

The MAD curves for the height and thickness fields are nearly parallel as they rise. They rise to a maximum of less than 5 m, which is quite respectable compared with the previous cases. Once again, the MD curves suggest the height field of the FE is subject to some effect which causes the mean height field of the FE to become progressively lower than that of the SP. As discussed in Sections 5.2 an 5.3, it is felt that this is due to problems at the boundary. This effect occurs early in the period and seems to have less importance in the thickness field forecast.

Fig. 5.51a gives the TE, KE and APE of both solutions over the entire domain. It is apparent that the FE has conserved TE very well. However, the FE does not convert as much KE to APE as does the SP. This is due to the problem with development and decay near the boundaries which was discussed earlier. Fig. 5.51b shows the PE for the two solutions over the entire domain. Once again, the PE of the FE rises rapidly during the forecast period, as a result of the short-wavelength waves generated near the boundaries.

CHAPTER 6

CONCLUSIONS

The finite-element method and the spectral method have been used to numerically integrate the equations describing a two-level quasi-geostrophic model of atmospheric flow on a β-plane. The solutions using the two numerical methods for a group of cases have been compared. The spectral solution, being believed to be highly accurate, was used to evaluate the performance of the finite-element solution.

It was found that the largest errors in the finite-element solution arose near the boundaries of the grid. In the interior of the grid, the finite-element solution compared very well with the spectral solution. Over that portion of the grid with high-resolution, only minor phase or amplitude differences were found between the two solutions. As the time of integration increased, the errors in the finite-element solution generated along the boundaries slowly infiltrated the interior of the grid.

Two major sources of the boundary errors were found. It was suggested that the major error source was the evaluation of derivatives perpendicular to the boundaries. In particular, higher-order derivatives were found to be subject to the greatest errors. Thus, vorticities were not well calculated by the finite-element method along the boundary. In an attempt to minimize this problem, a portion of the grid near the boundaries was chosen to have high accuracy in the direction perpendicular to the boundaries. Further steps could possibly be taken to minimize this source of error. For example, the vorticity on the edge could be "tied" to the first interior grid point, i.e. the vorticity on the edge could be set equal to the vorticity at the first interior grid point. The success or failure of this technique would depend on the system of equations which would result from this approximation. It was not attempted as it would have required a major rewriting of computer routines and there was no strong evidence that it would dramatically improve the results. It is, however, a technique deserving of future consideration.

The second source of error at the boundary was that due to the implementation of the boundary condition, Eqn. (2.33). This required an integration along the boundary and was subject to the truncation error in the numerical integration. The magnitude of this truncation error was determined by that portion of the boundary with the largest grid spacing in the x-direction. The effect of this error on the solution could be most effectively minimized by ensuring that the boundaries are well away from the area of interest and by, if possible, using a high-order numerical integration scheme. In this work, the first technique could not be used. The second technique was not fully studied as it seemed the poor evaluation of perpendicular derivatives at the boundary was of much greater importance.

The techniques for avoiding or preventing the numerical instability found in Case III are reasonably well known. The small grid lengths in the y-direction combined with the relatively long time-step have led to this instability. Increasing the grid length and/or decreasing the time-step would prevent this. A different time integration scheme could also be used. In the problem under consideration, the elimination of the small grid spacings near the boundary would be most economical in terms of computer time. This could be implemented when, as discussed earlier, an improved method of evaluating the derivatives perpendicular to the boundaries was found.

The two methods of solution were used for only a limited number of cases. There are other atmospheric flows which would be interesting to study. For example, it would be interesting to compare the two solutions when a barotropic atmosphere (i.e. one in which the thickness field was zero everywhere) was considered.[1] In this case, the height field would simply translate, i.e. there could be no development. By studying cases with varying speeds of translation, one could study how well the finite-element solution deals with translation and seek to determine a relation between the growth of errors and the speed of translation. A companion set of cases to these would be those in which the height field is stationary (or nearly so) but developing. Then, one could seek a relation between the growth of errors and the rate of development. These two sets of cases would help to answer some of the questions raised in Chapter 5 where the effects of translation and development could not be separated.

---

1. Note that the perturbation thickness field is implied here.

Although some difficulties have been experienced in implementing the FEM, it has demonstrated that it is a technique worthy of study when one is considering atmospheric flow problems. In the future, the author hopes to use the FEM for the solution of a more complicated atmospheric model. This model is presently being used for operational weather forecasting at the Alberta Weather Centre using finite-difference techniques.

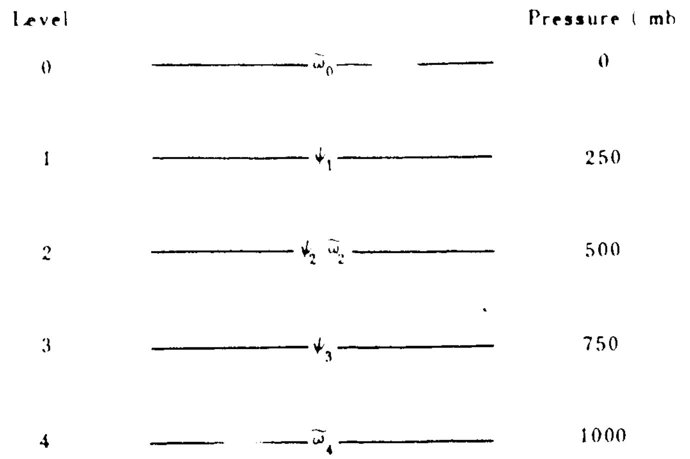
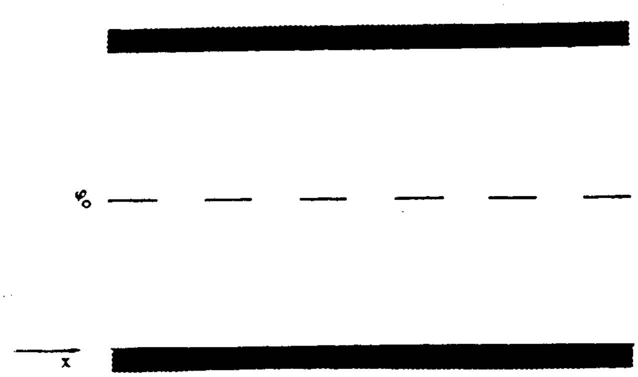


Fig. 2.1 Vertical discretization of the model showing the five levels used, with pressure as the vertical coordinate. ψ_i is the stream function at level i and $\tilde{\omega}_i$ is the vertical velocity at level i.



in he horizontal plane ϕ_i is the
at the plane is tangent to the
x- ection is east; the y-direction

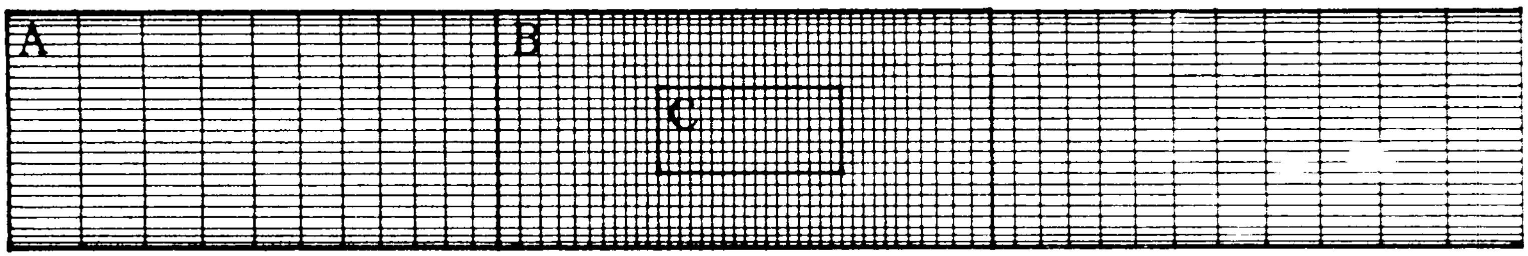
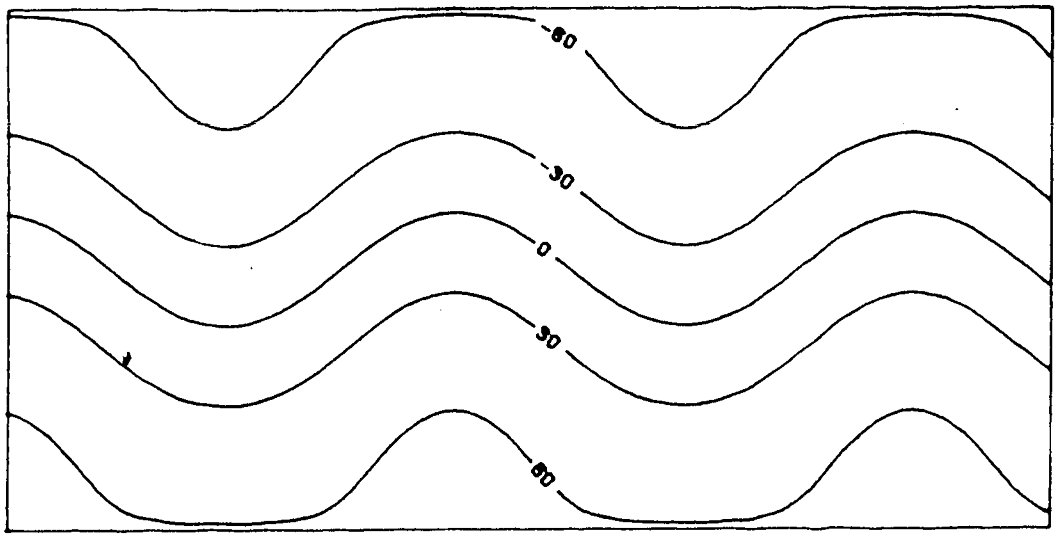


Fig. 4.1 A 62 x 27 variable grid-length mesh having a 27 x 19 sub-domain of uniform high resolution. Areas A and B are the portions of the domain used for displaying the fields in Chapter 5 while area C is the verification area.



(a)

(b)

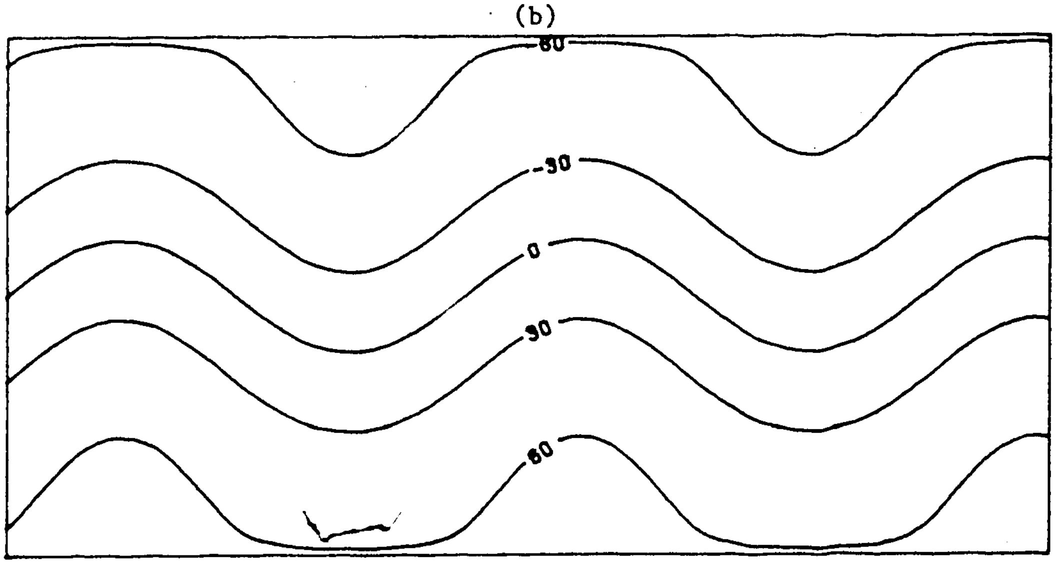
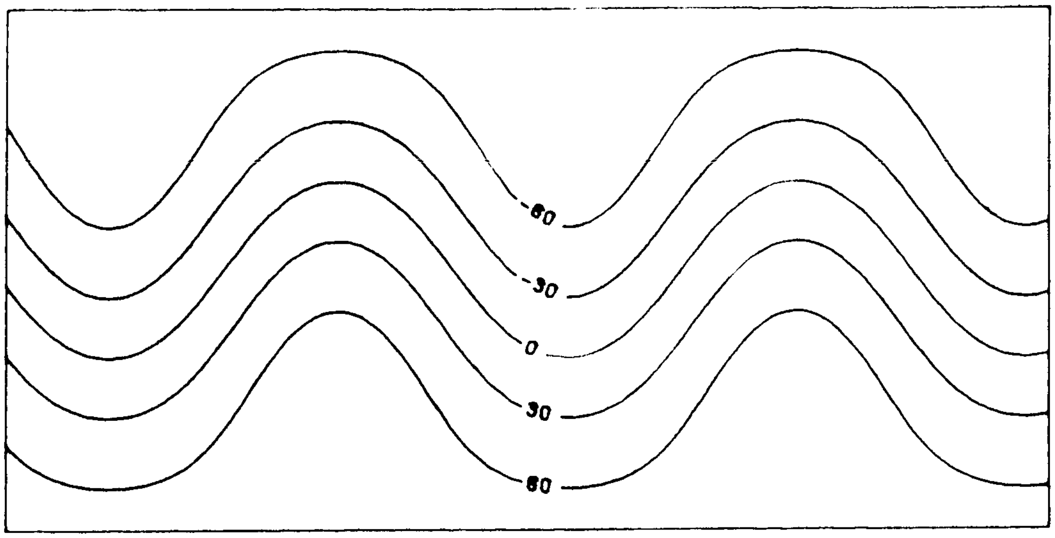


Fig. 5.1 Initial mean height field in metres for Case I in (a) area B and (b) area A.



(a)

(b)

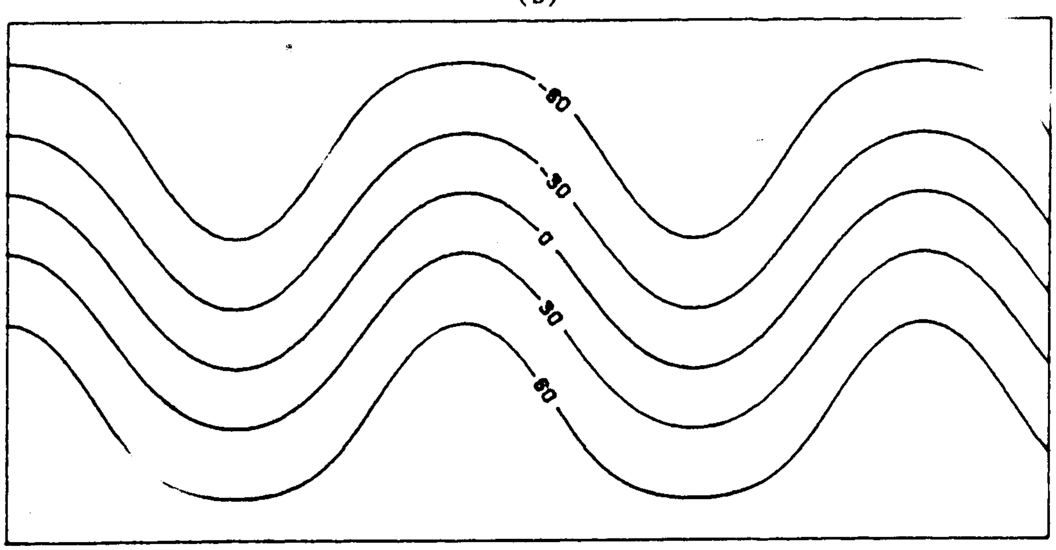
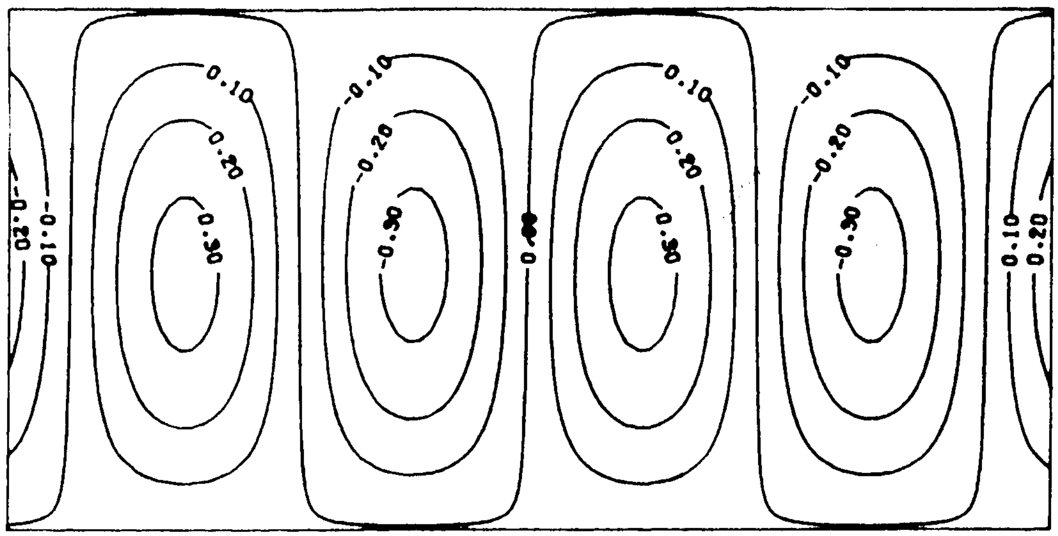


Fig. 5.2 Initial thickness field in metres for Case I in (a) area B and (b) area A.



(a)

(b)

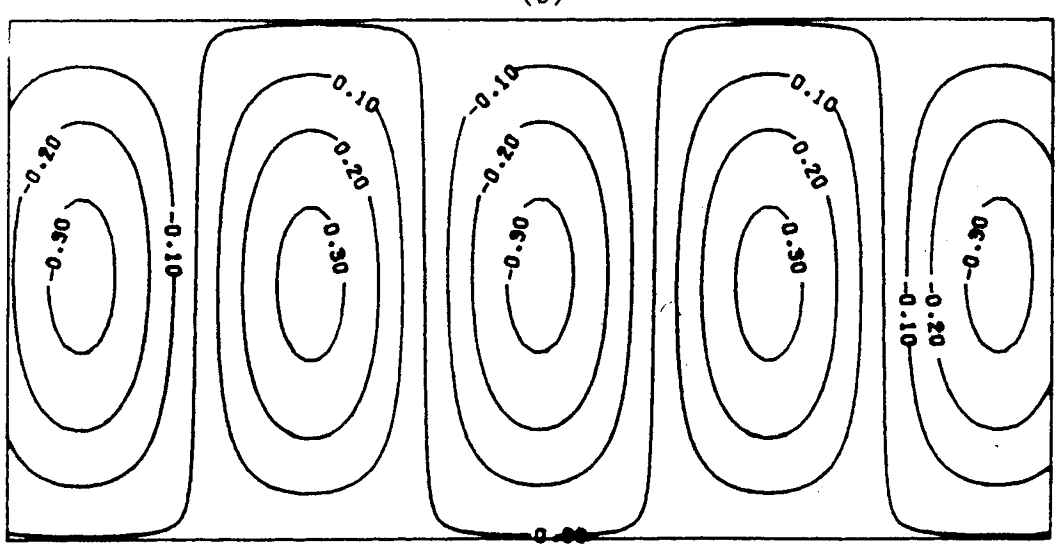
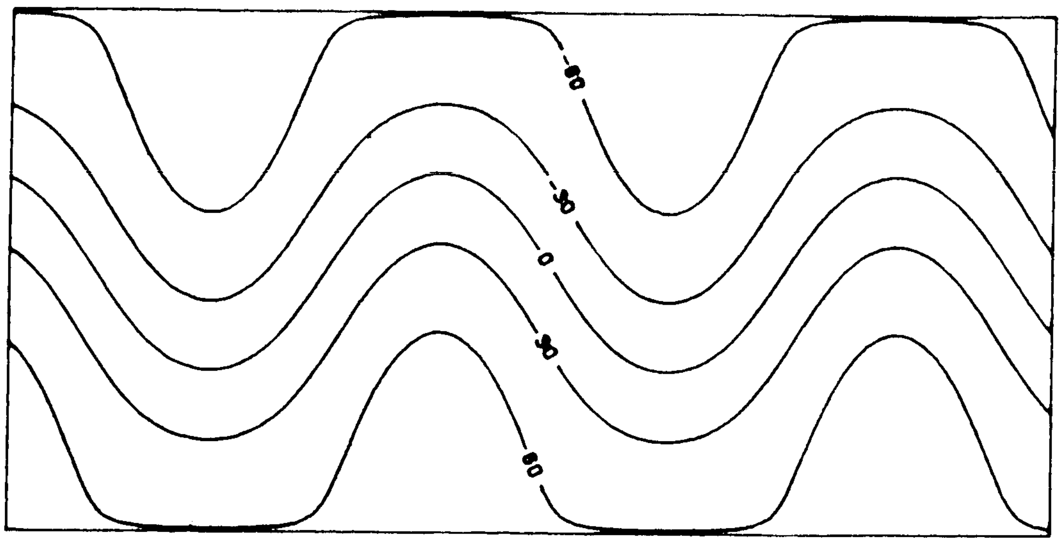


Fig. 5.3 Initial omega field in μbar/sec for Case I in (a) area B and (b) area A.



(a)

(b)

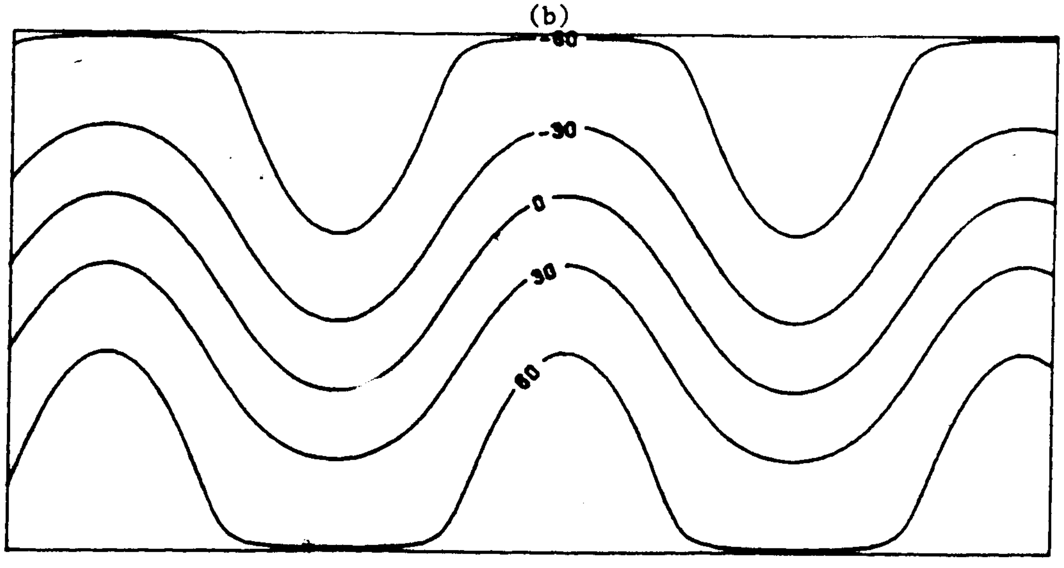
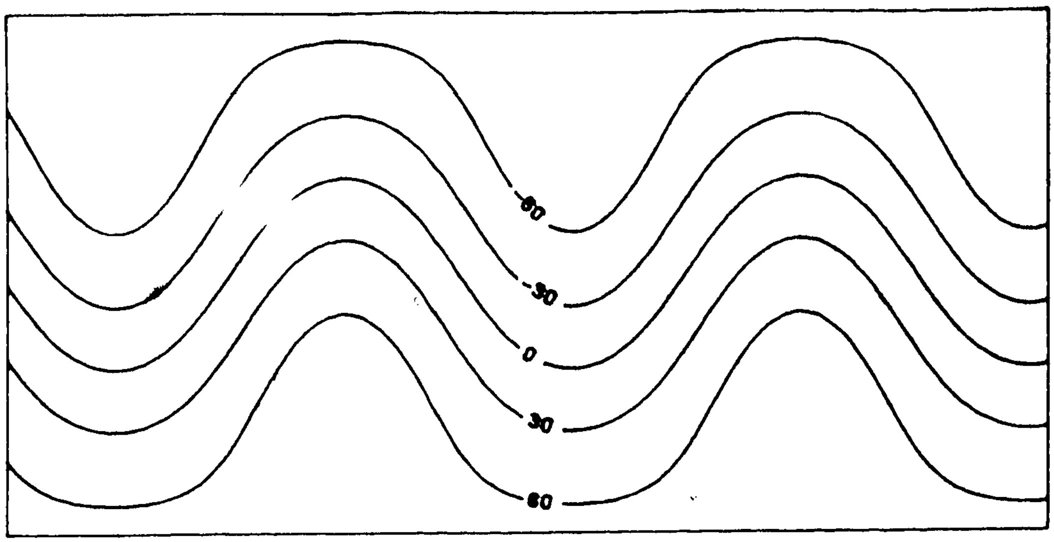


Fig. 5.4 The 24-hour spectral solution for the mean height field in metres for Case I in (a) area B and (b) area A.



(a)

(b)

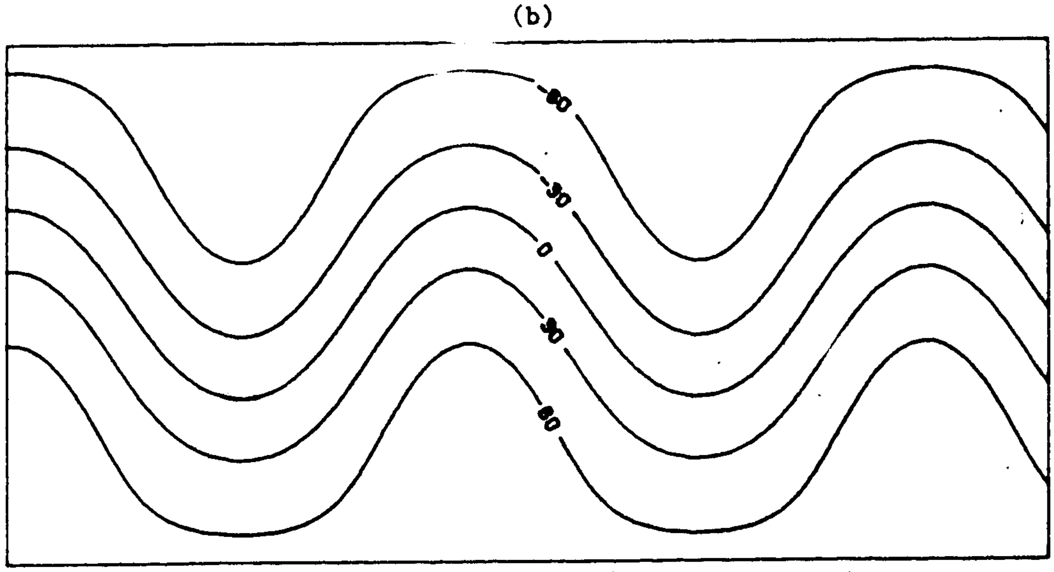


Fig. 5.5 The 24-hour spectral solution for the thickness field in metres for Case I in (a) area B and (b) area A.

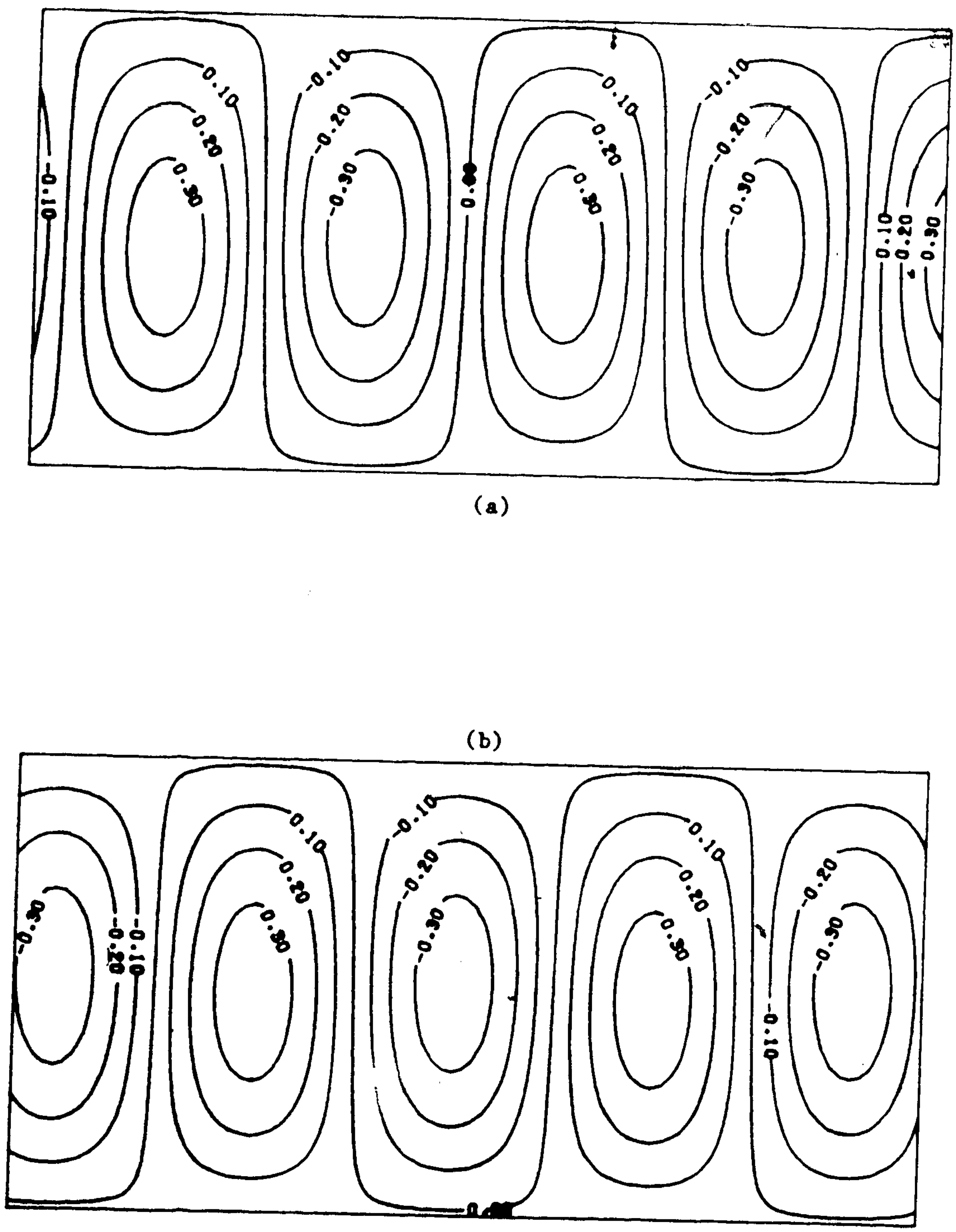
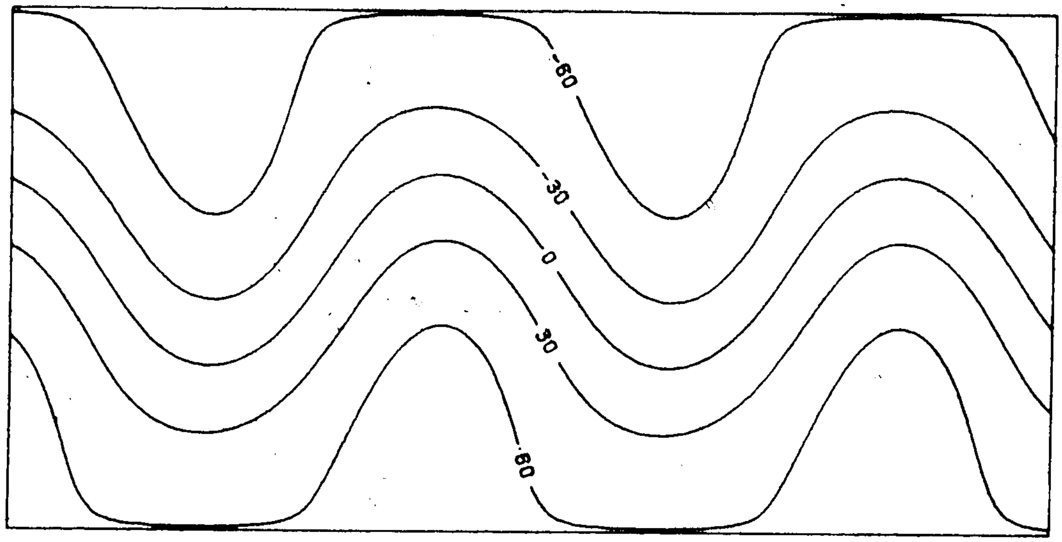


Fig. 5.6 The 24-hour spectral solution for the omega field in μbar/sec for Case I in (a) area B and (b) area A.



(a)

(b)

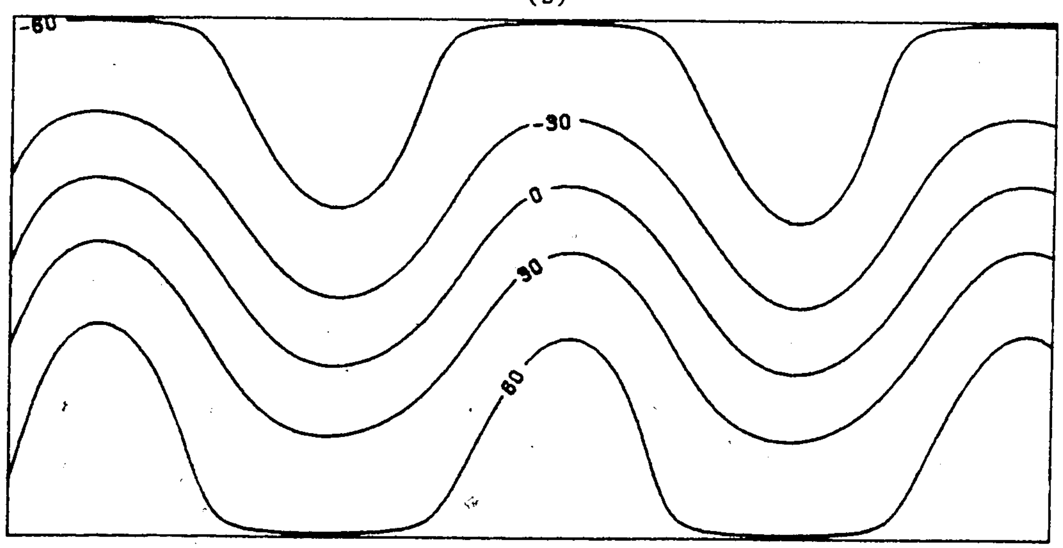
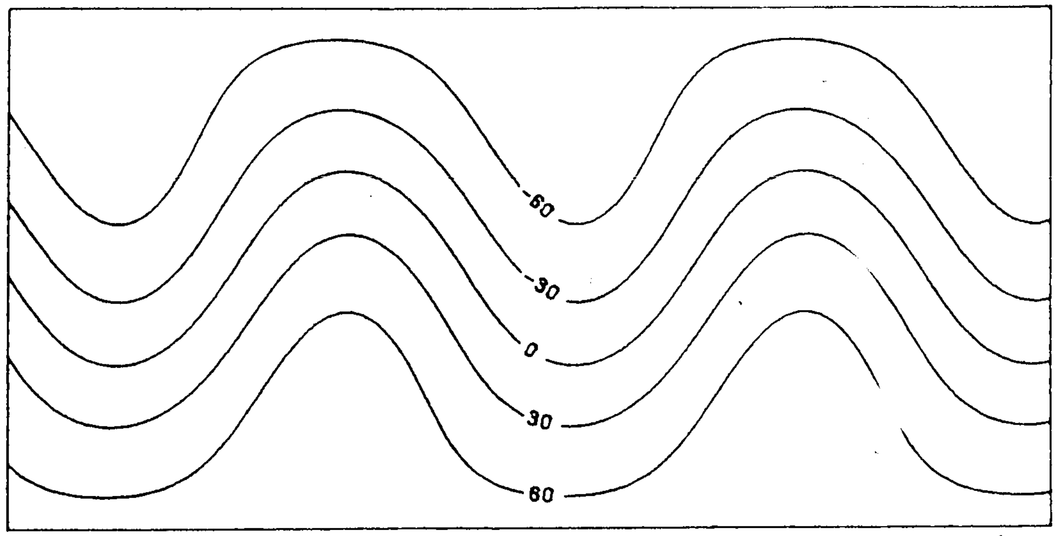


Fig. 5.7 The 24-hour finite-element solution for the mean height field in metres for Case I in (a) area B and (b) area A.



(a)

(b)

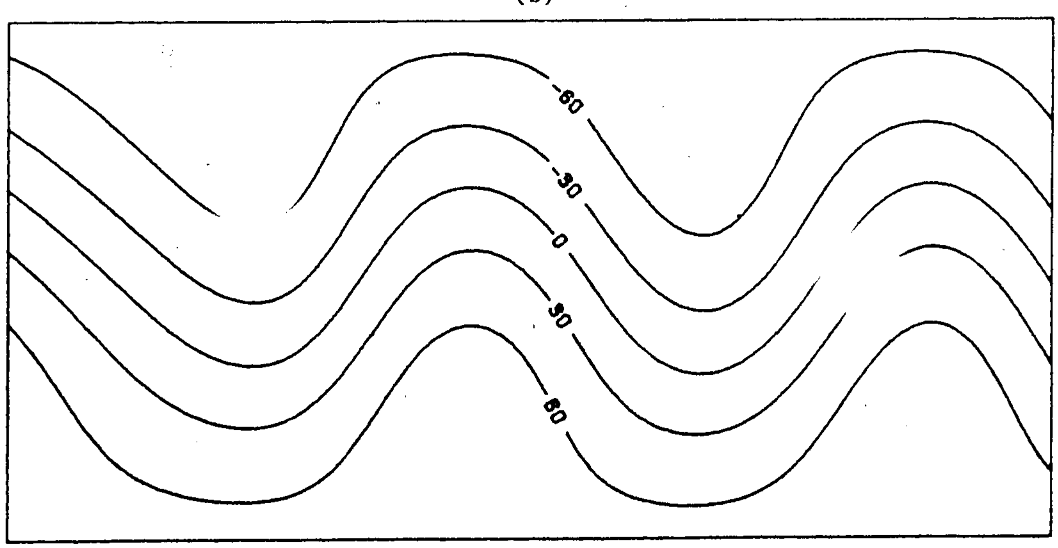
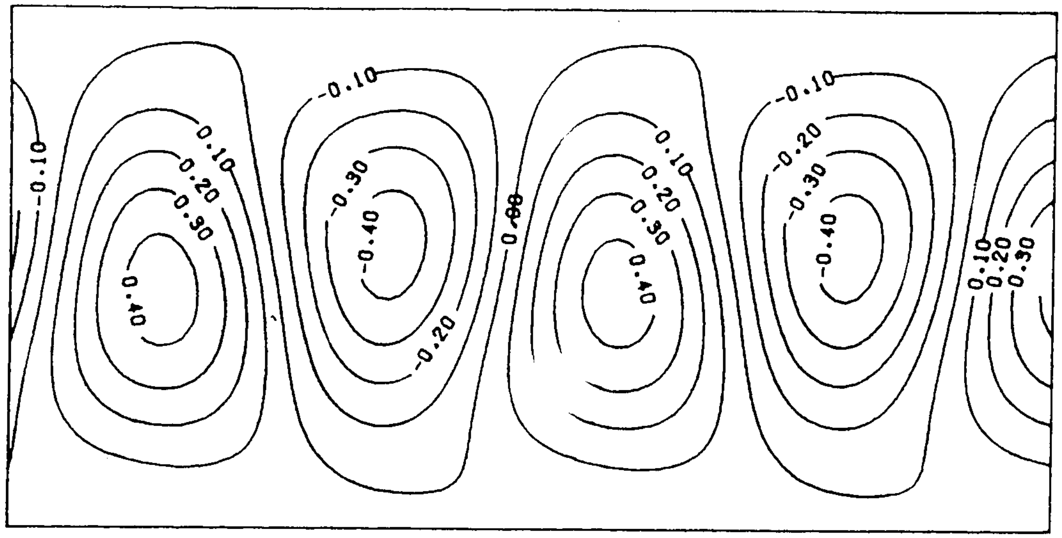


Fig. 5.8 The 24-hour finite-element solution for the thickness field in metres for Case I in (a) area B and (b) area A.



(a)

(b)

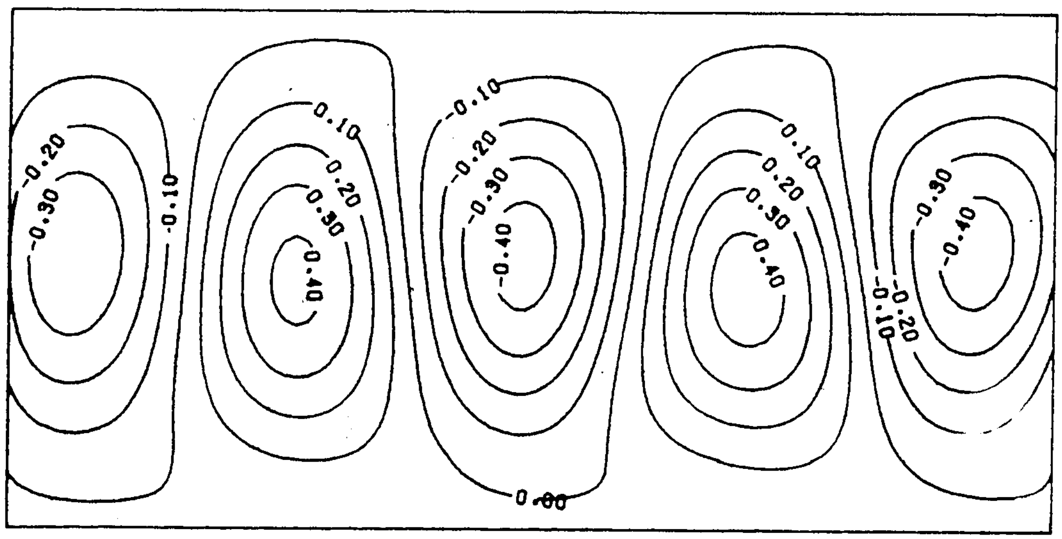


Fig. 5.9 The 24-hour finite-element solution for the omega field in μbar/sec for Case I in (a) area B and (b) area A.

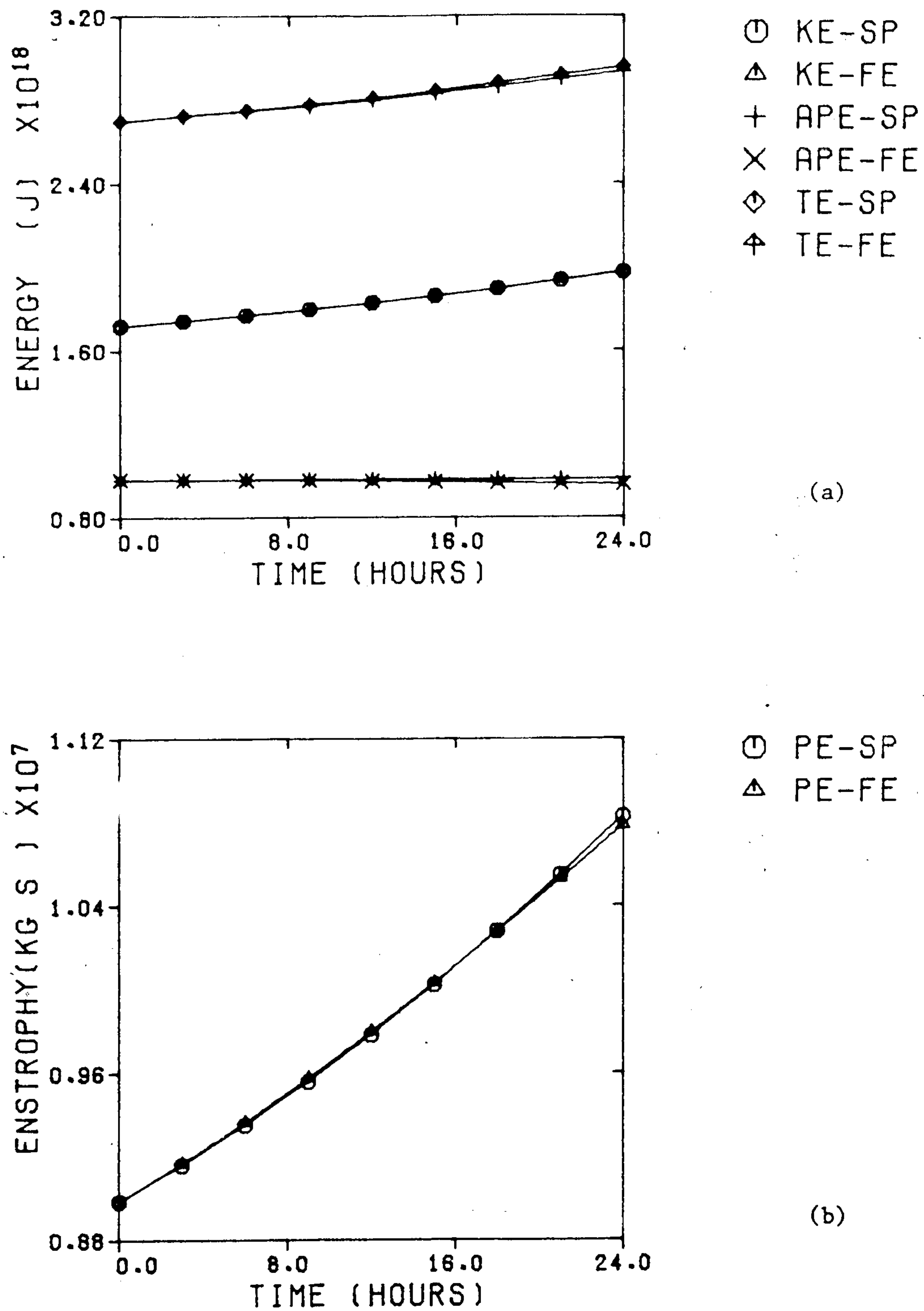


Fig. 5.10 The (a) energies and (b) potential enstrophy of Case I for the two solutions in the verification area.

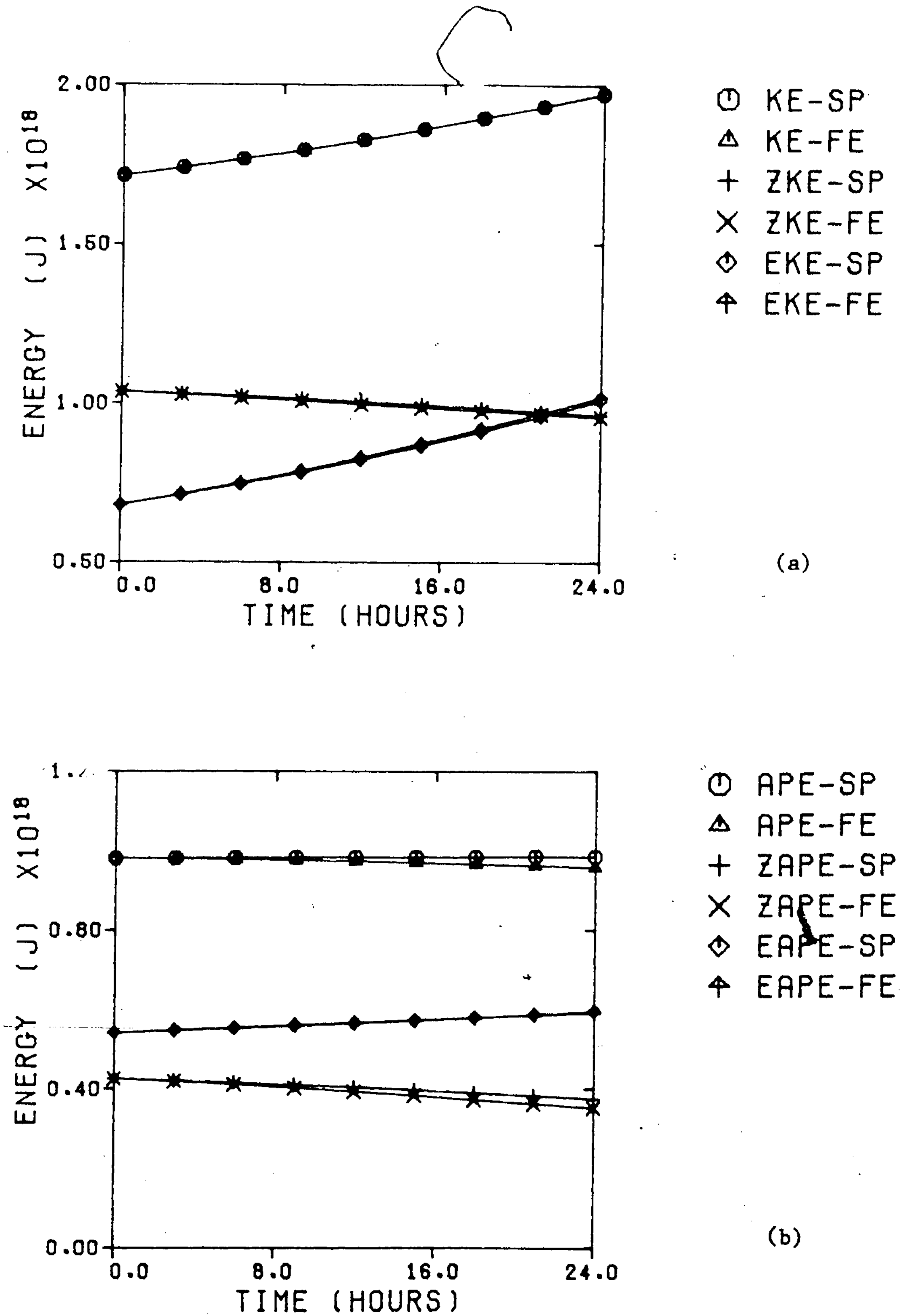


Fig. 5.11 The (a) kinetic energies and (b) potential energies of Case I for the two solutions in the verification area.

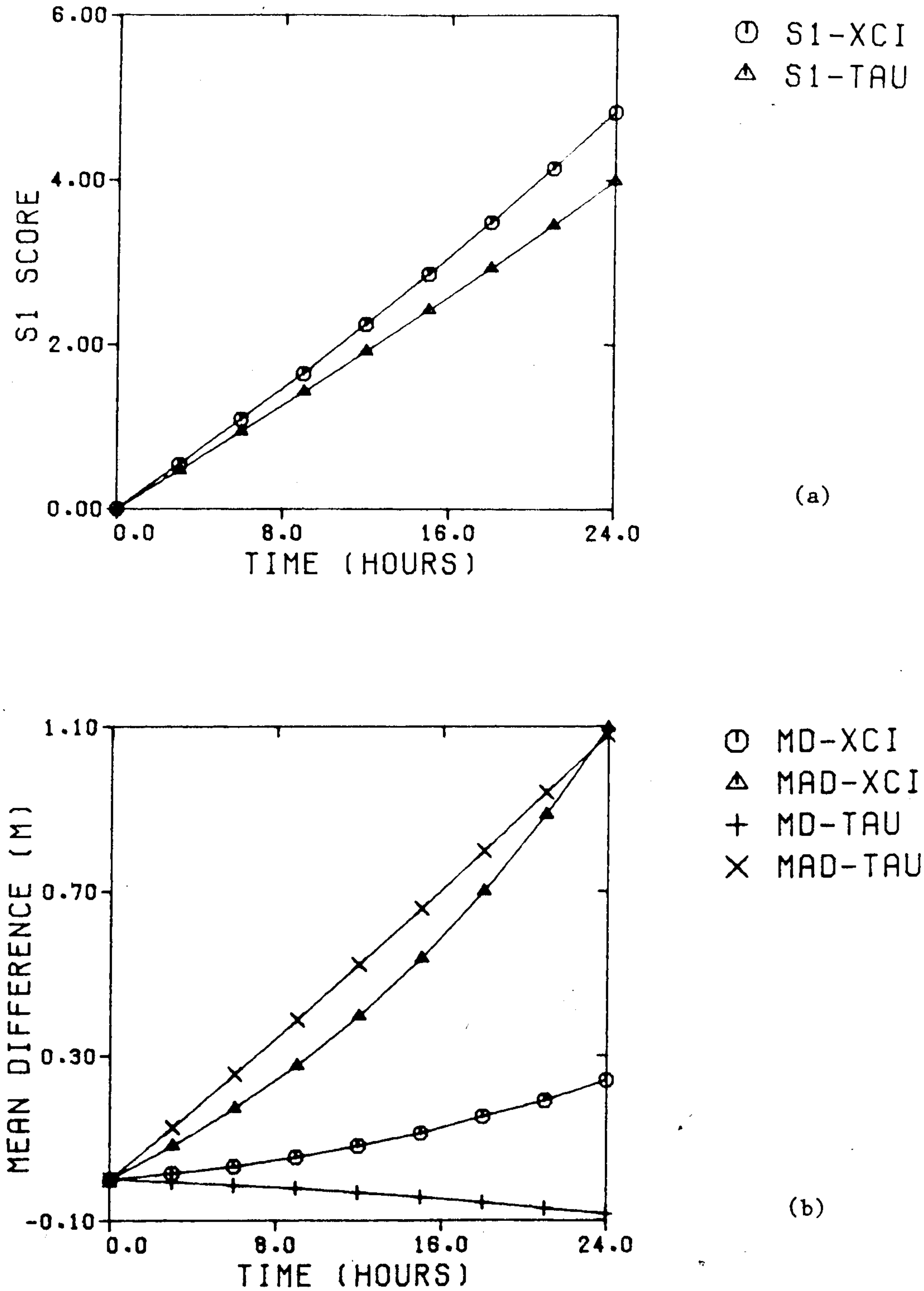
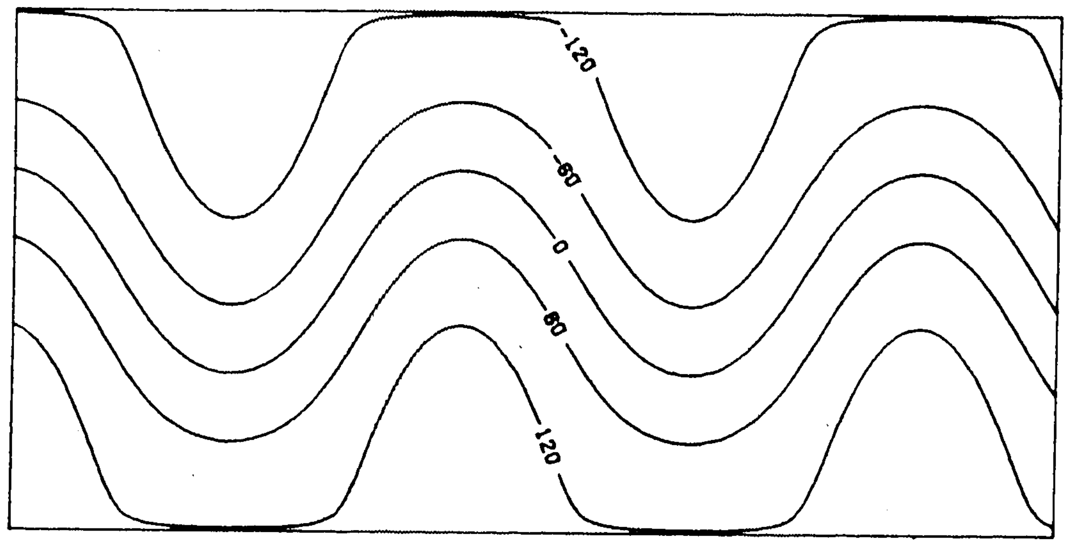


Fig. 5.12 The (a) S1 scores and (b) the MD and MAD curves for Case I for the two solutions in the verification area.



(a)

(b)

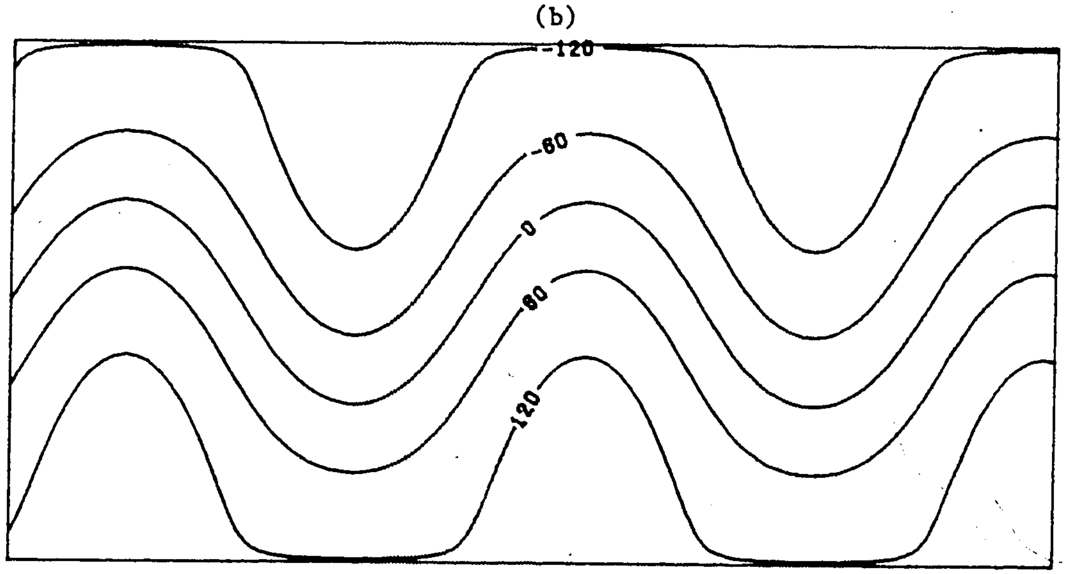
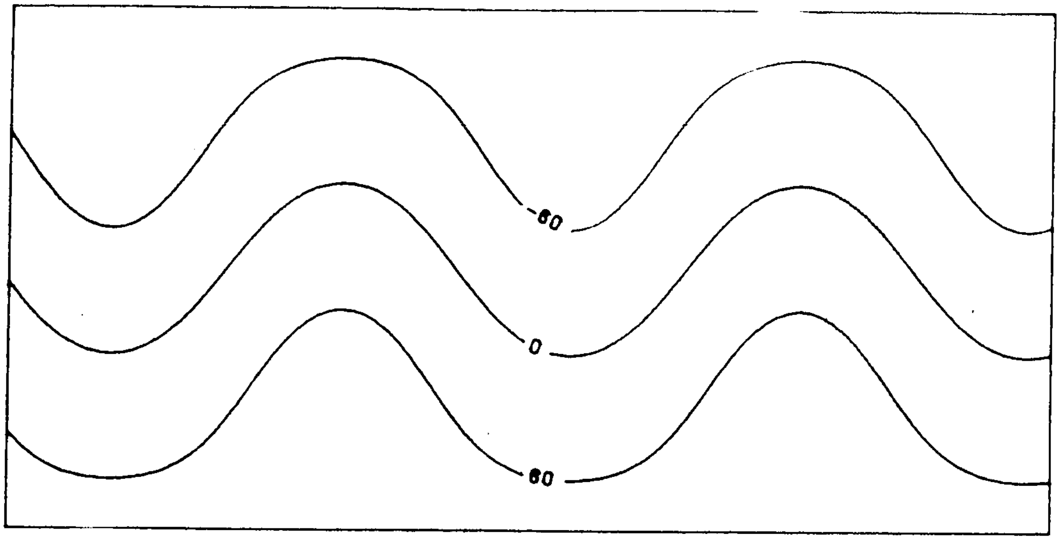


Fig. 5.13 Initial mean height field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

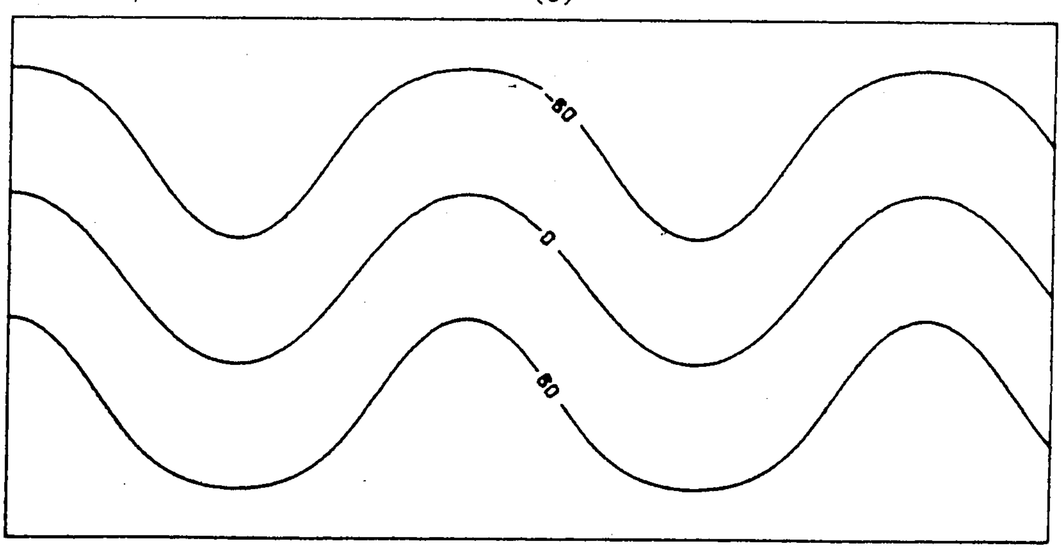
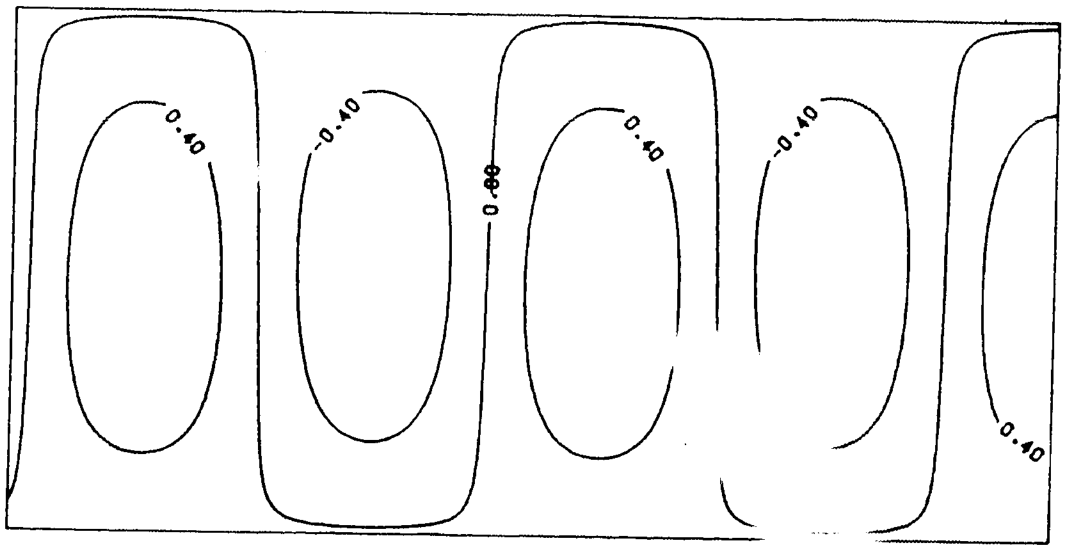


Fig. 5.14 Initial thickness field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

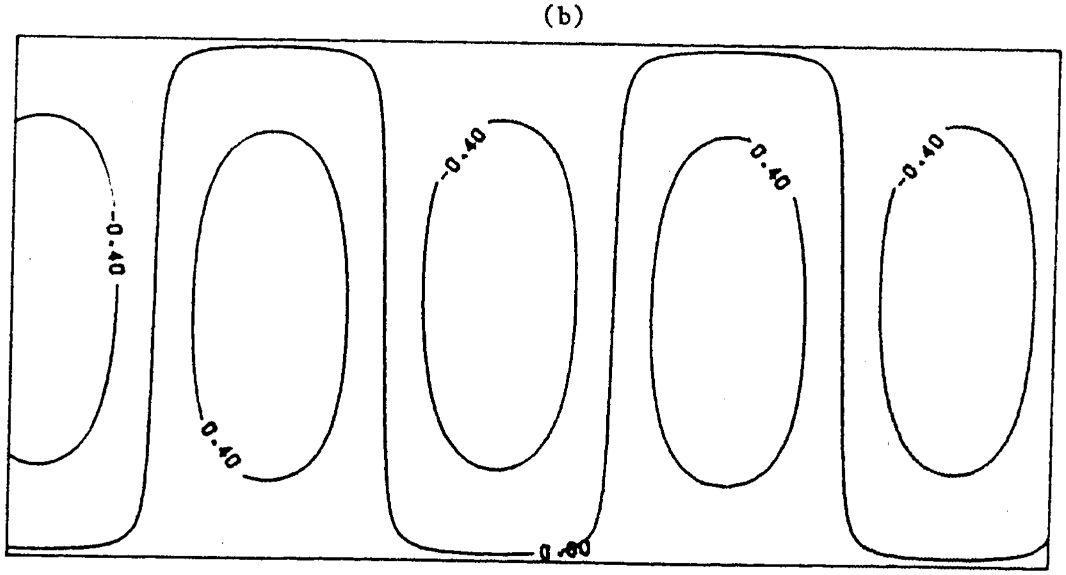
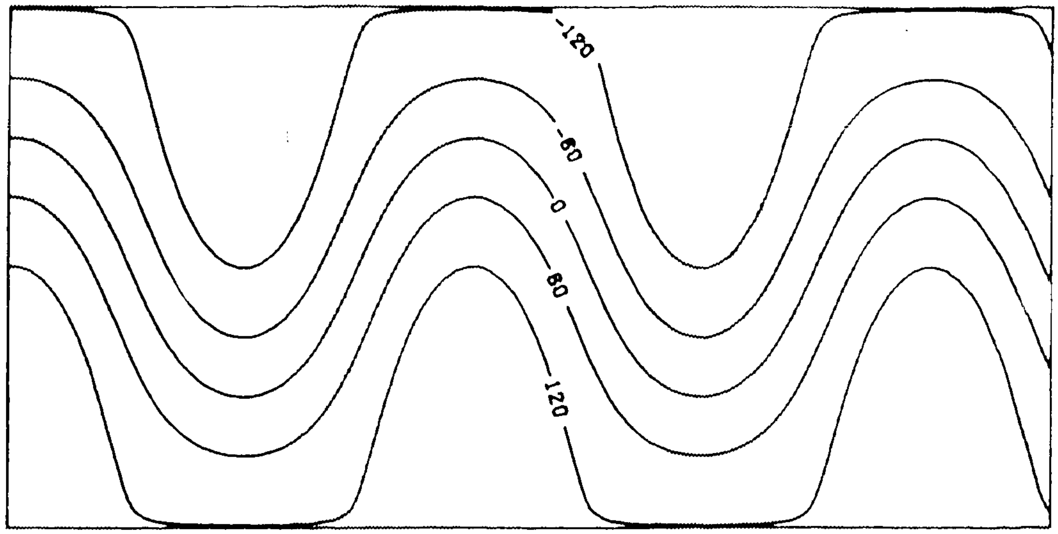


Fig. 5.15 Initial omega field in μbar/sec for Case II in (a) area B and (b) area A.



(a)

(b)

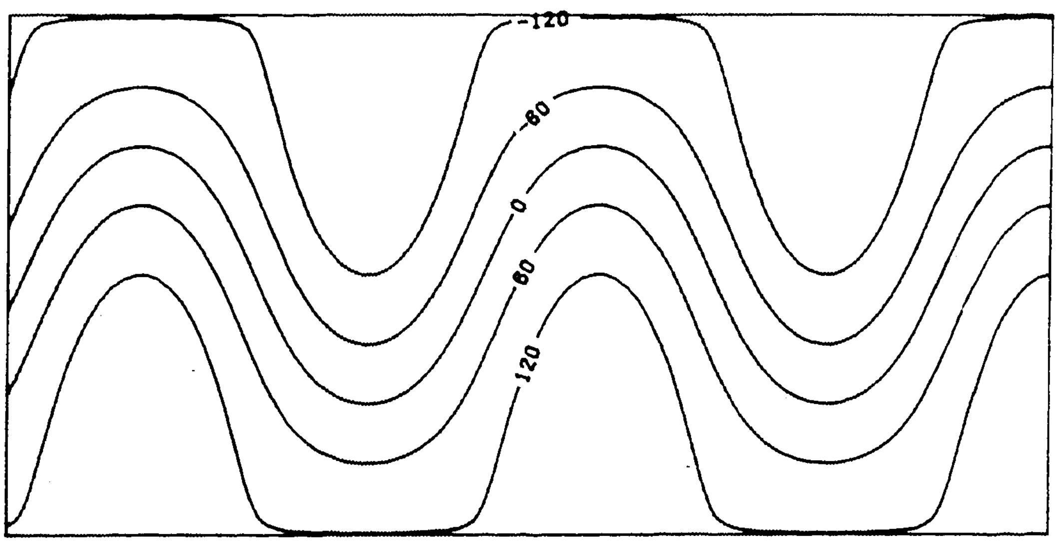
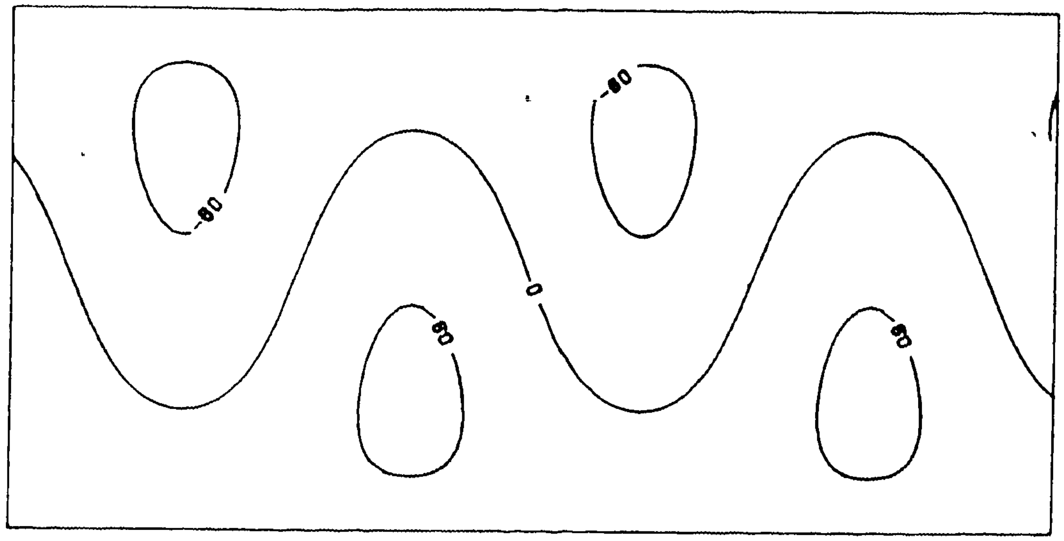


Fig. 5.16 The 48-hour spectral solution for the mean height field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

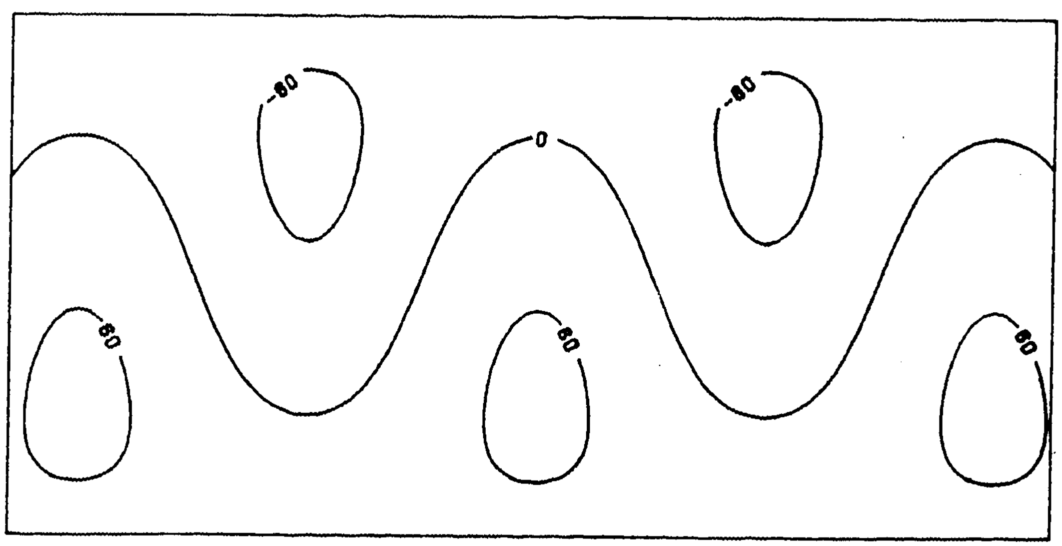
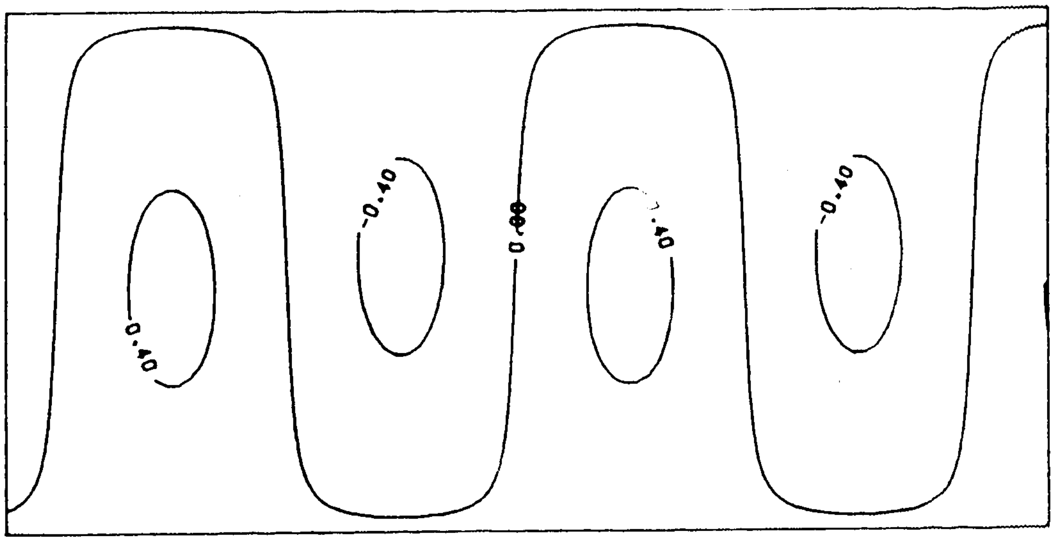


Fig. 5.17 The 48-hour spectral solution for the thickness field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

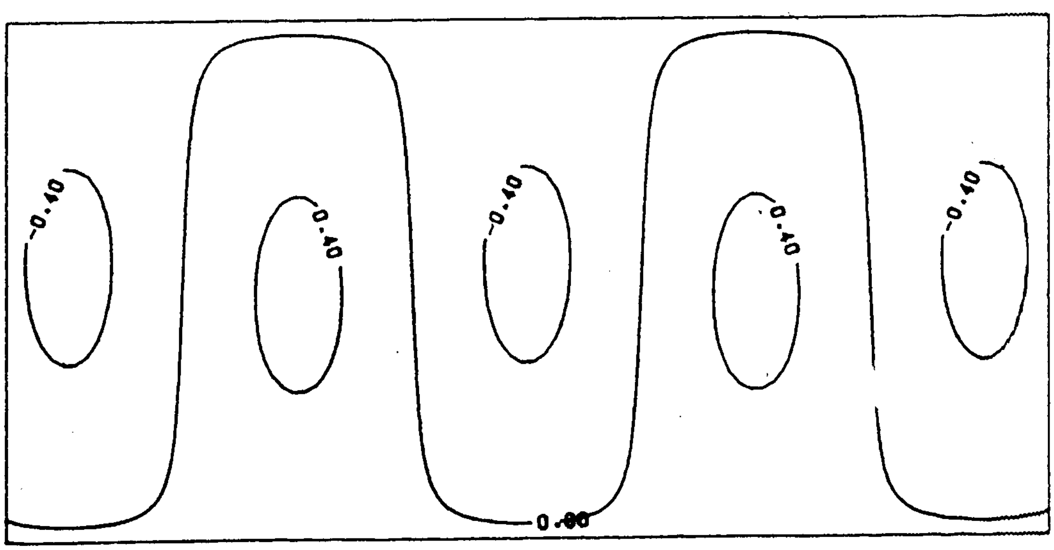
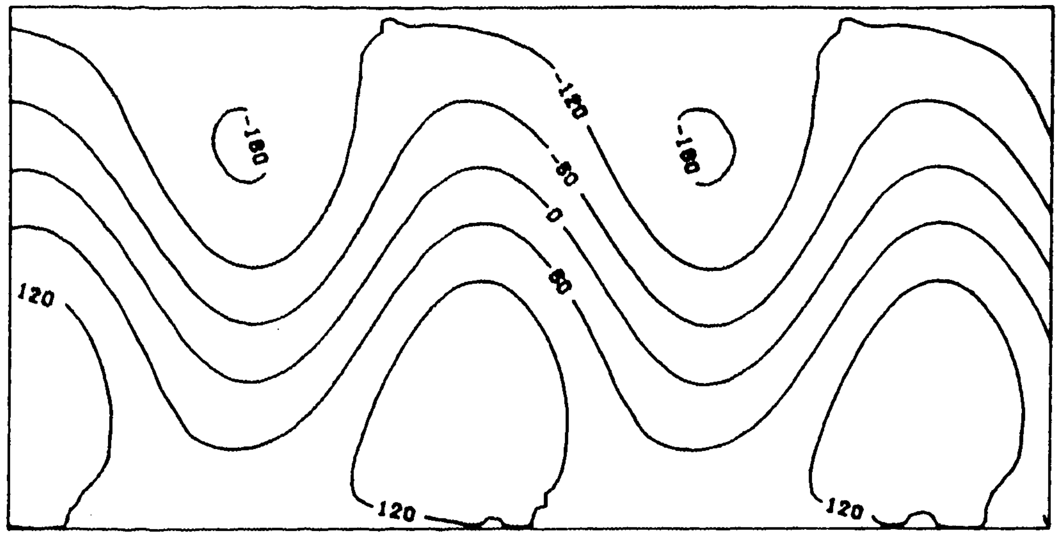


Fig. 5.18 The 48-hour spectral solution for the omega field in μbar/sec for Case II in (a) area B and (b) area A.



(a)

(b)

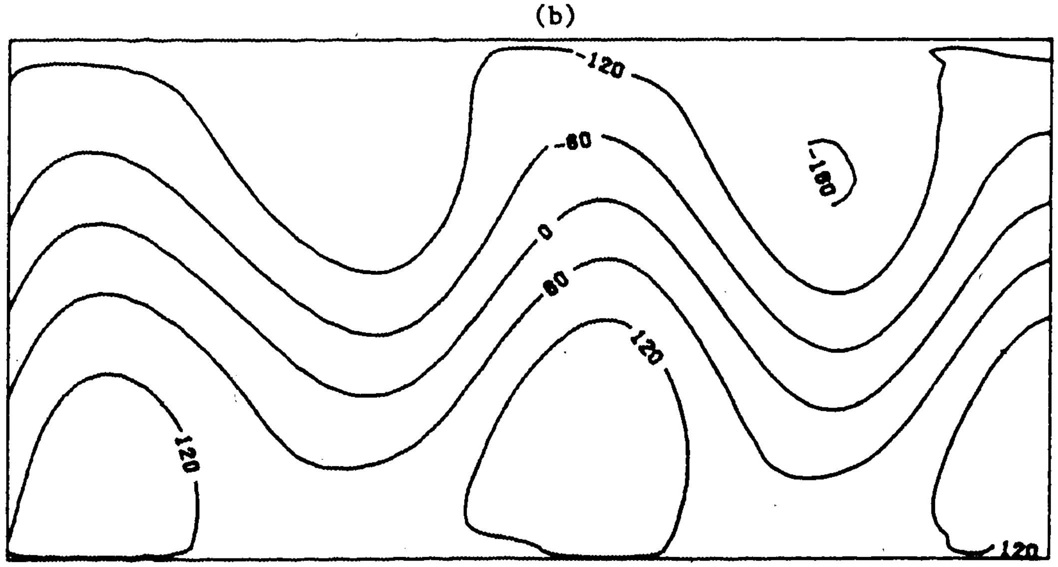
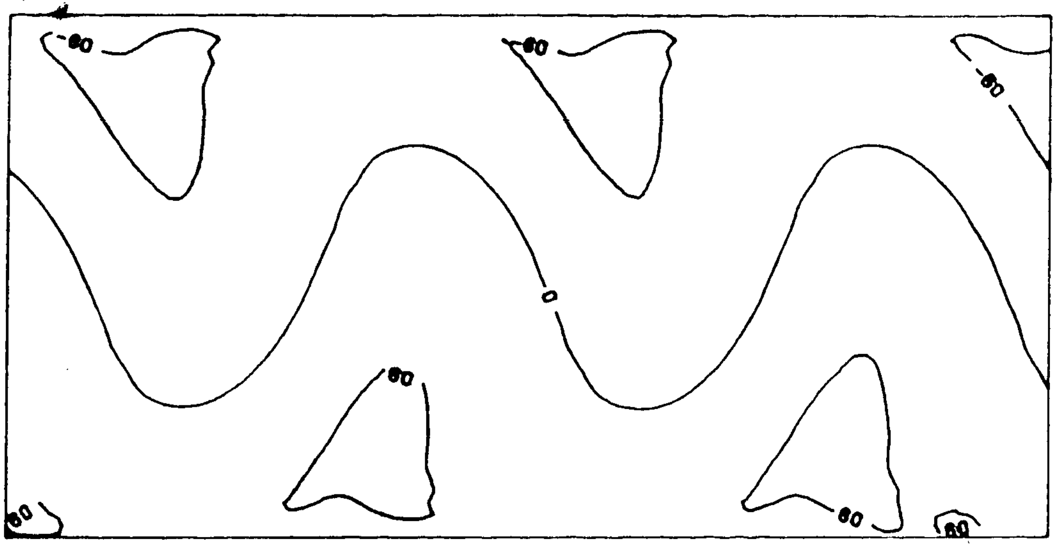


Fig. 5.19 The 48-hour finite-element solution for the mean height field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

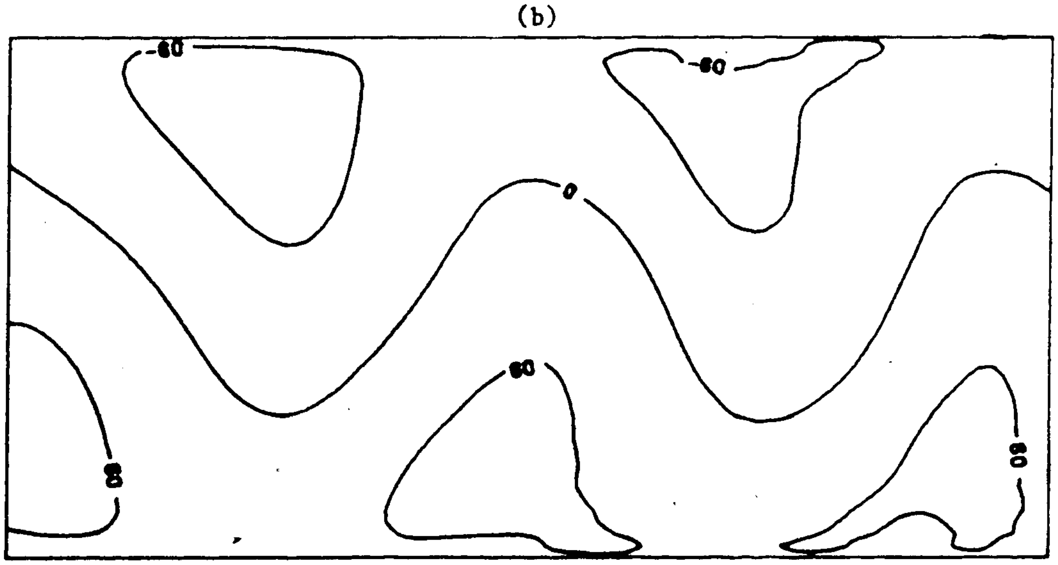
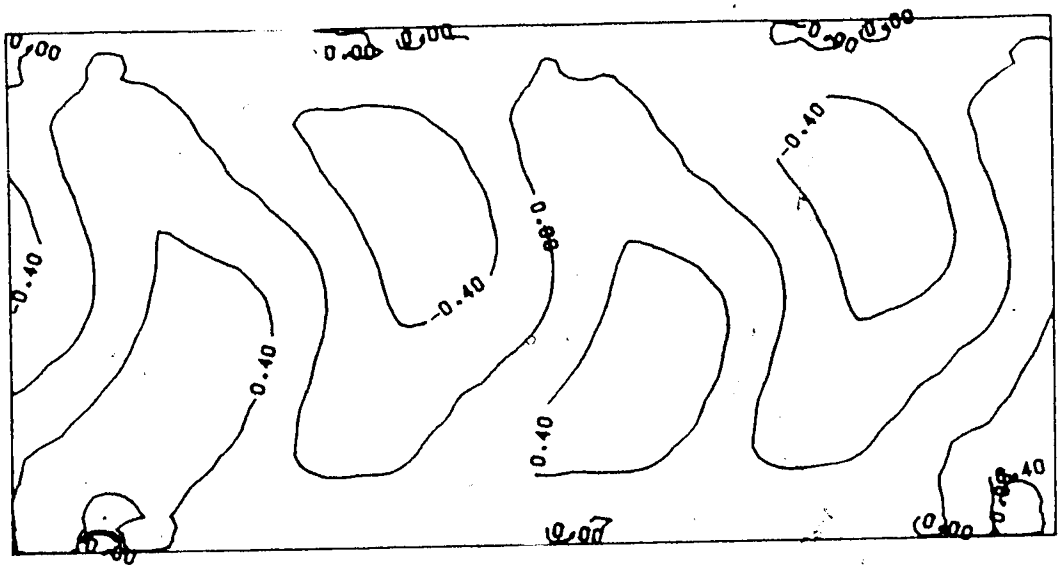


Fig. 5.20 The 48-hour finite-element solution for the thickness field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

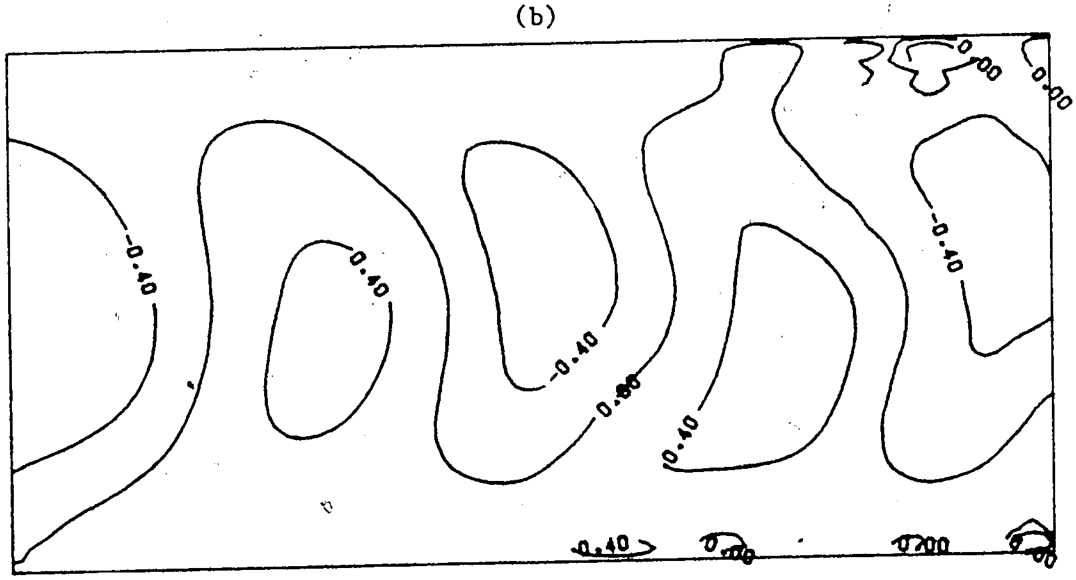
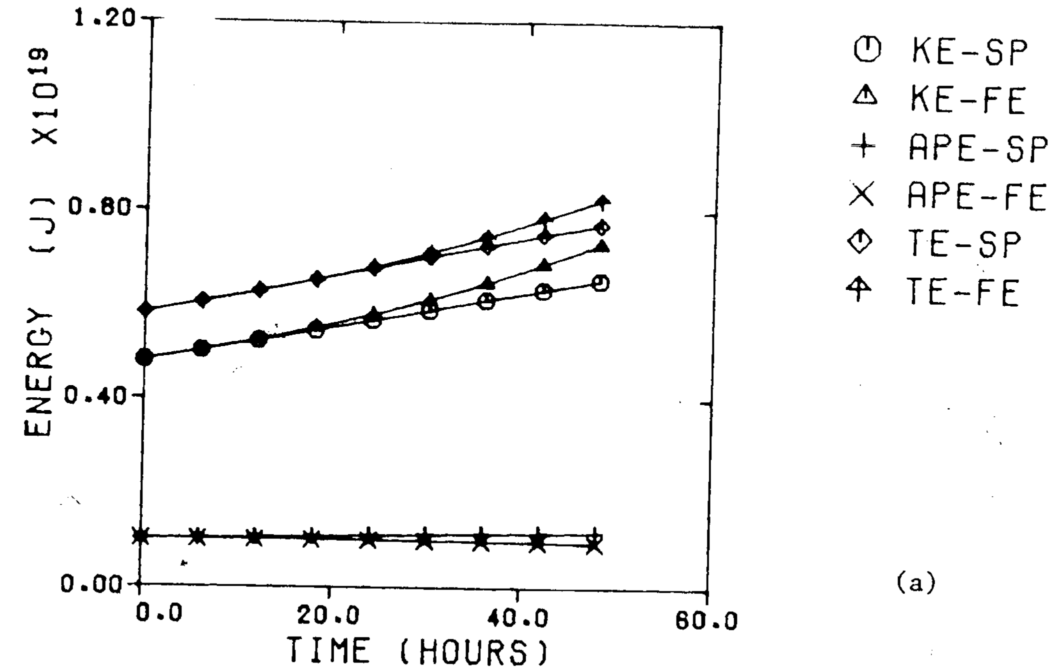


Fig. 5.21 The 48-hour finite-element solution for the omega field in μbar/sec for Case II in (a) area B and (b) area A.



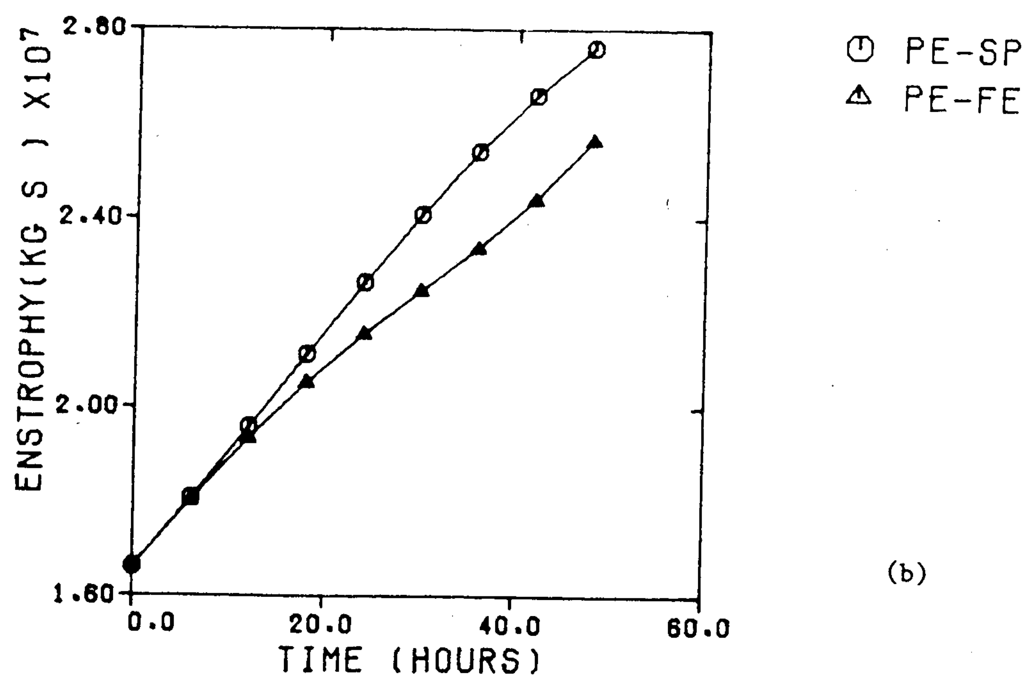
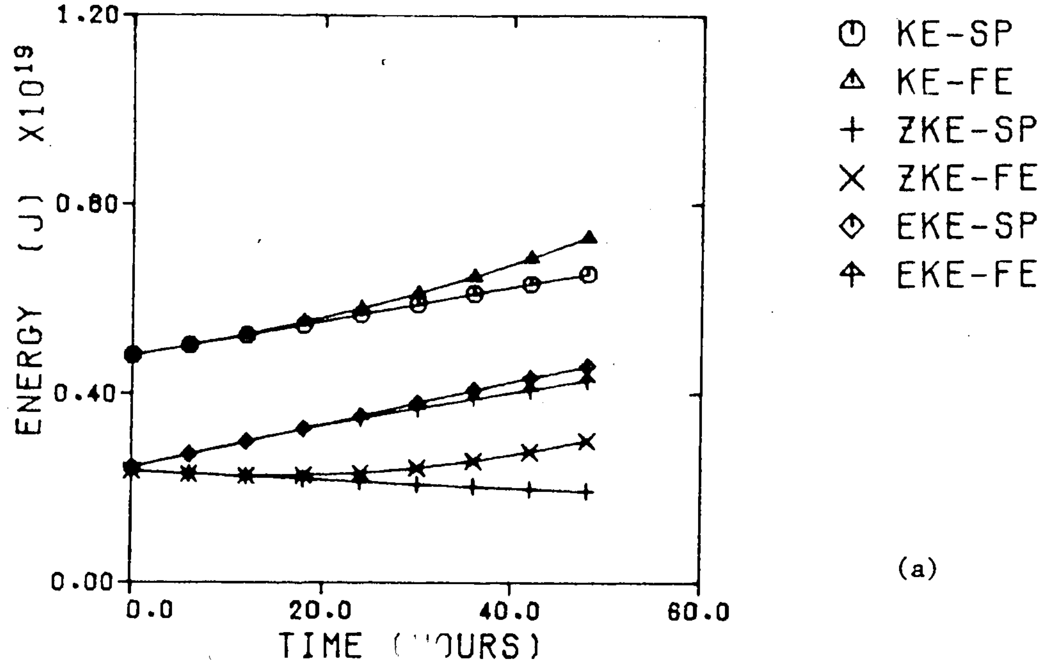
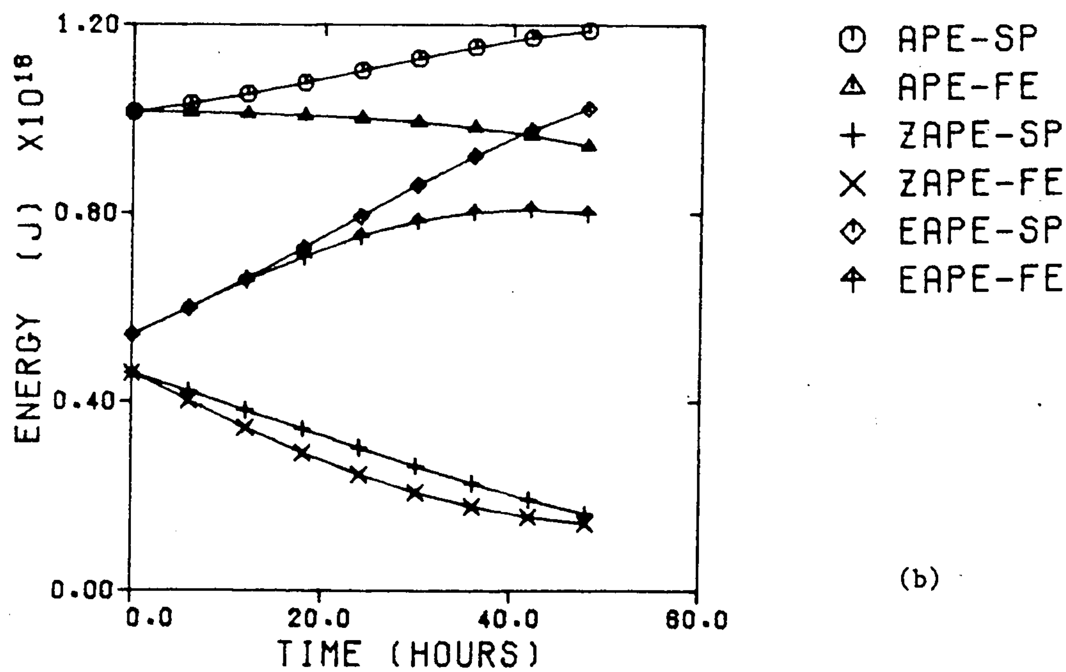


Fig. 5.22 The (a) energies and (b) potential enstrophy of Case II for the two solutions in the verification area.

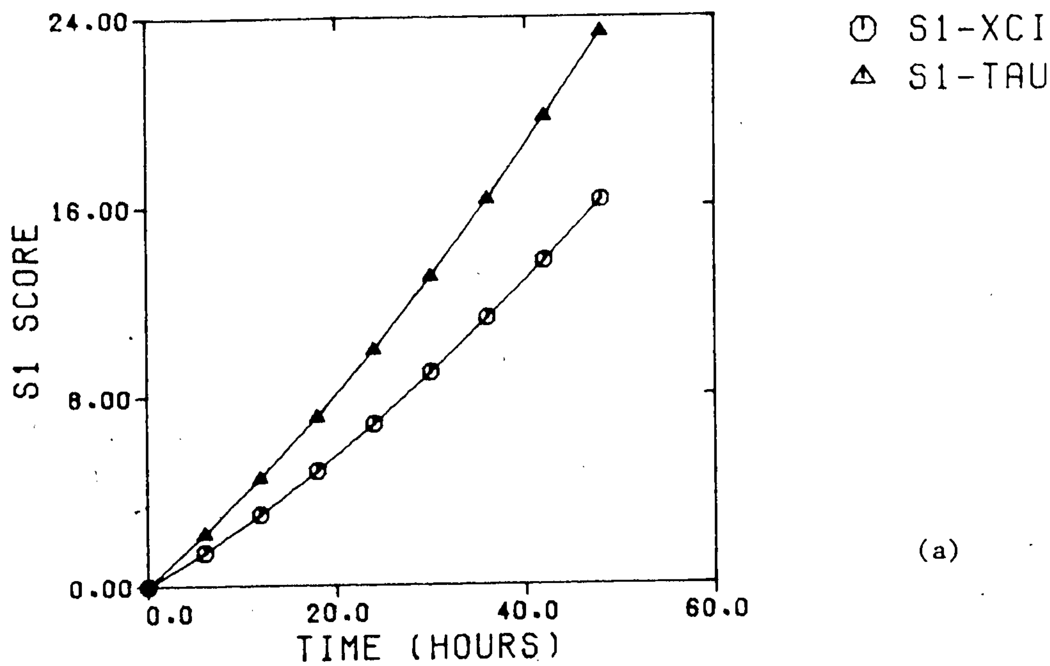


(a)



(b)

Fig. 5.23 The (a) kinetic energies and (b) potential energies of Case II for the two solutions in the verification area.



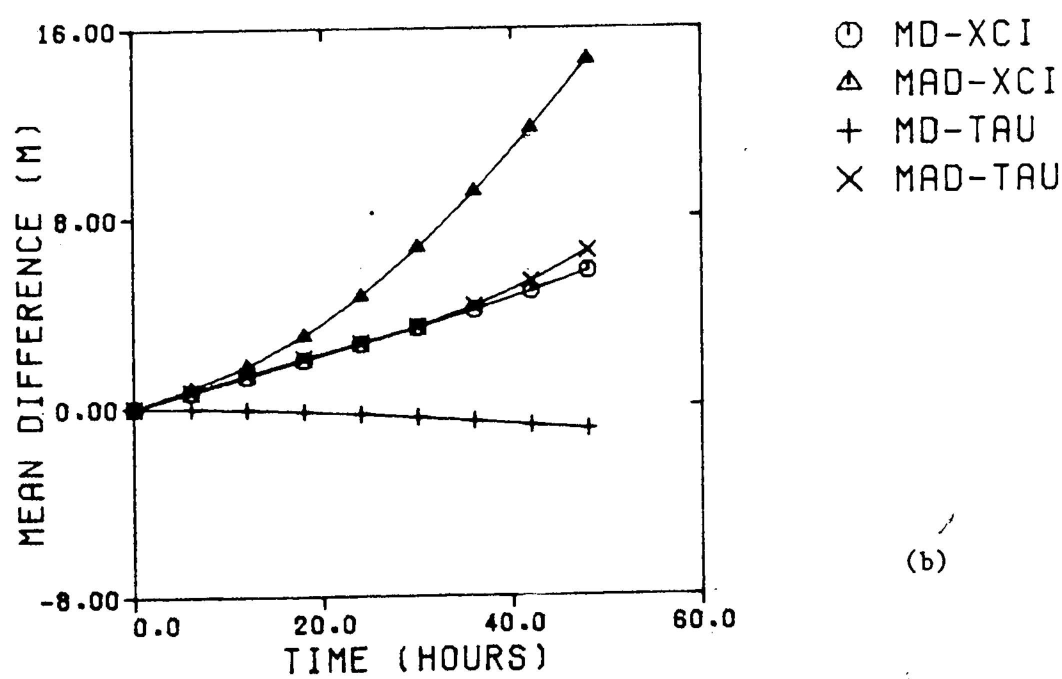


Fig. 5.24 The (a) S1 scores and (b) MD and MAD curves of Case II for the two solutions in the verification area.

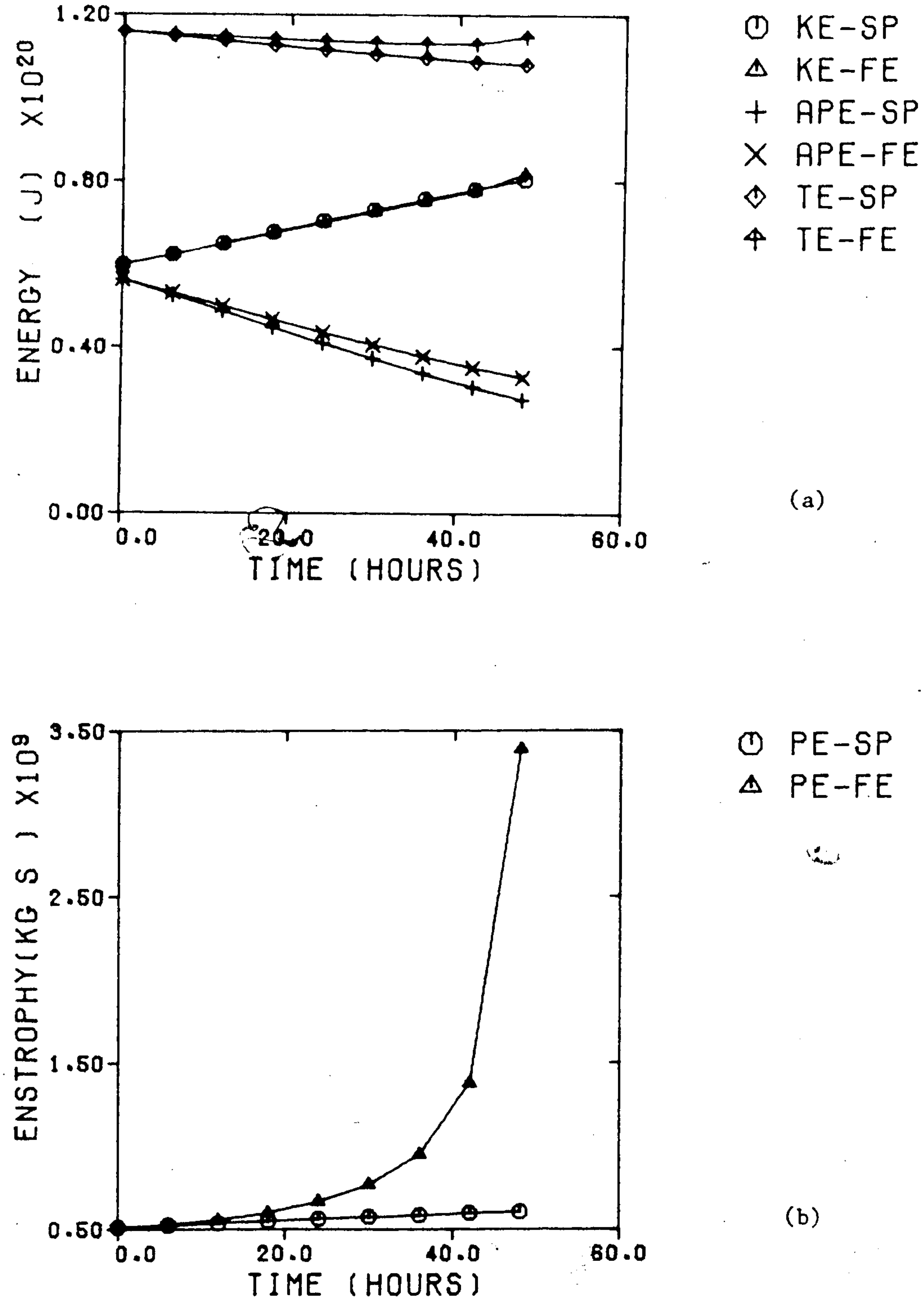
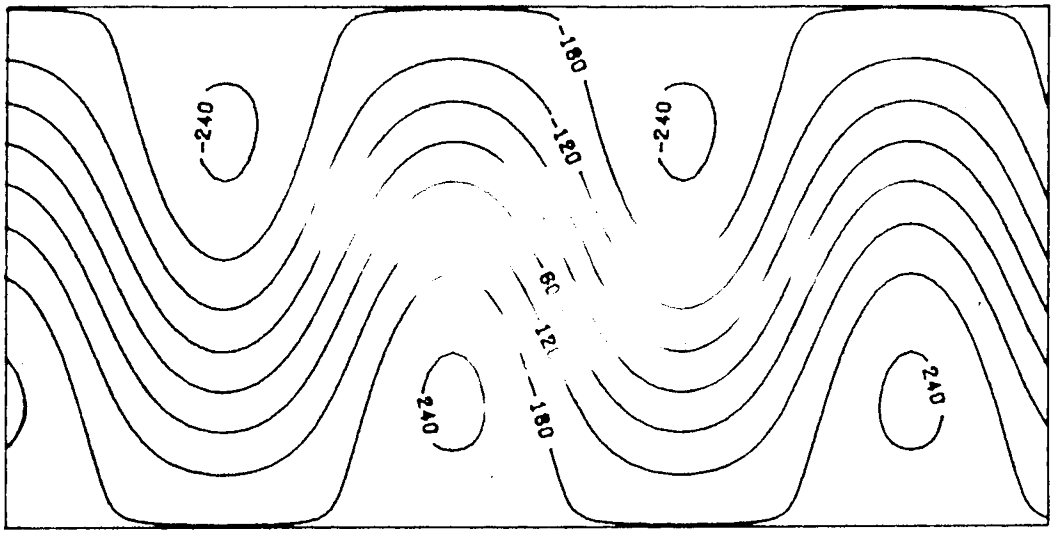


Fig. 5.25 The (a) energies and (b) potential enstrophy of Case II for the two solutions in the entire domain.



(a)

(b)

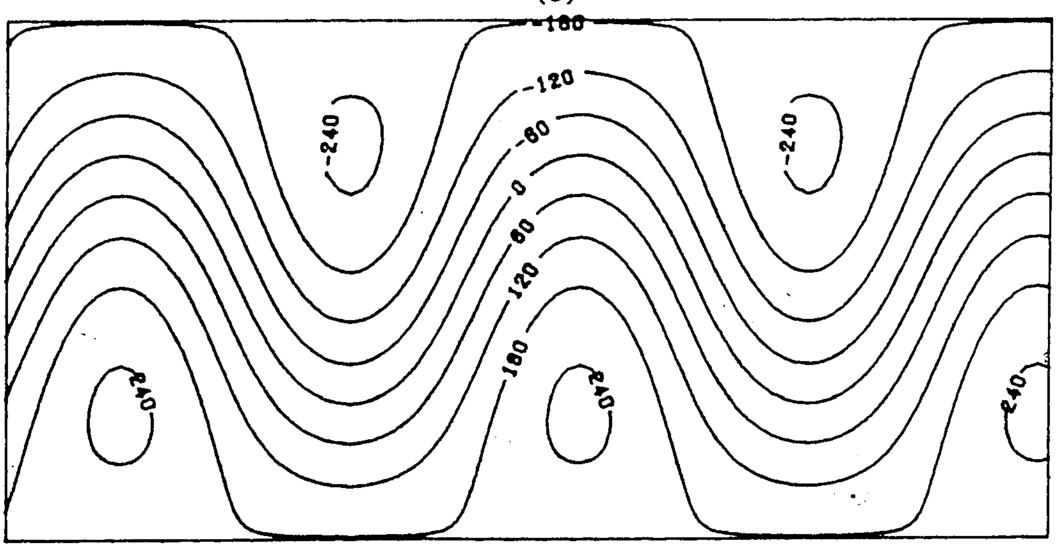
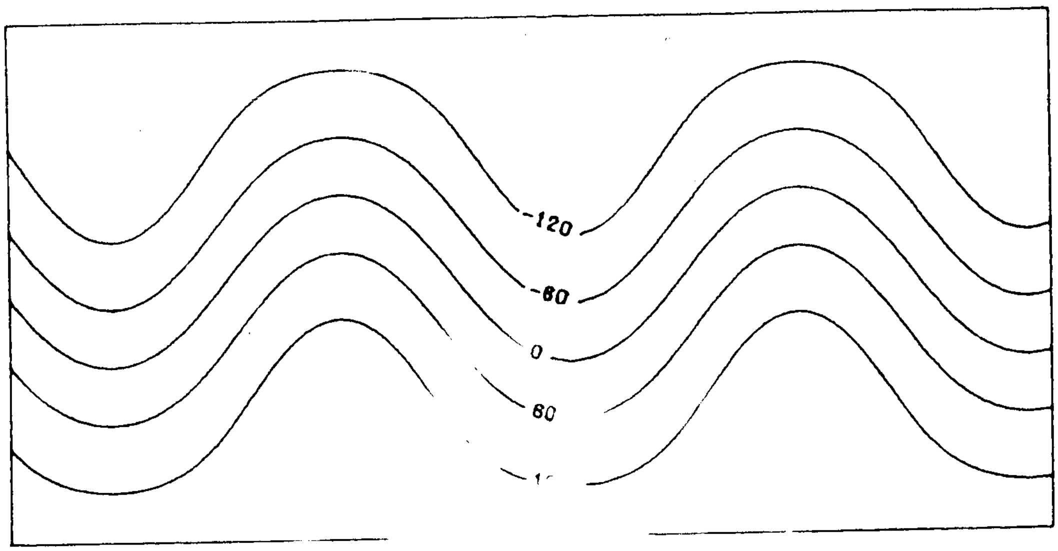


Fig. 5.26 Initial mean height field in metres for Case III in (a) area B and (b) area A.



(b)

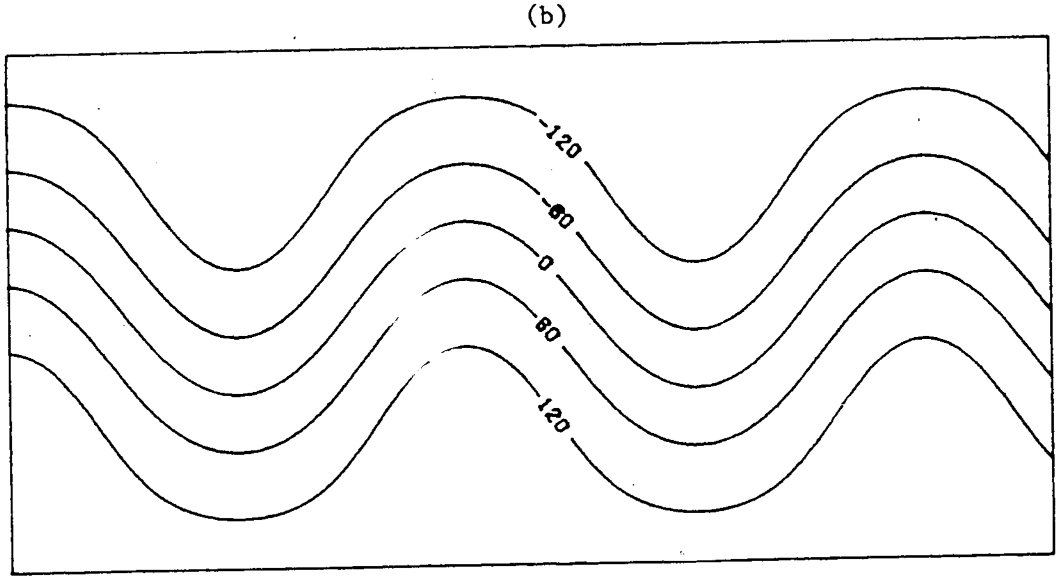
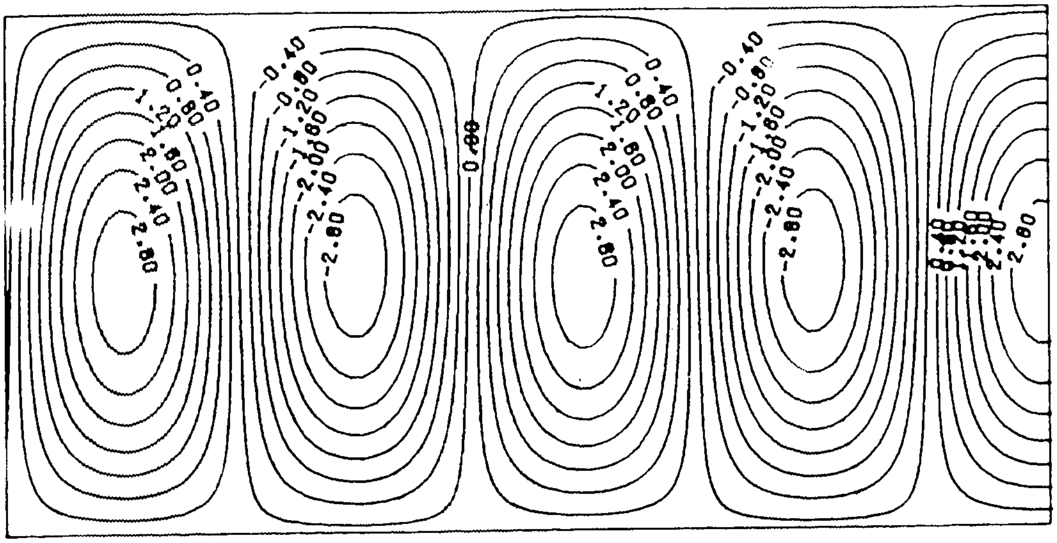


Fig. 5.27 Initial thickness field in metres for Case III in (a) area B and (b) area A.



(a)

(b)

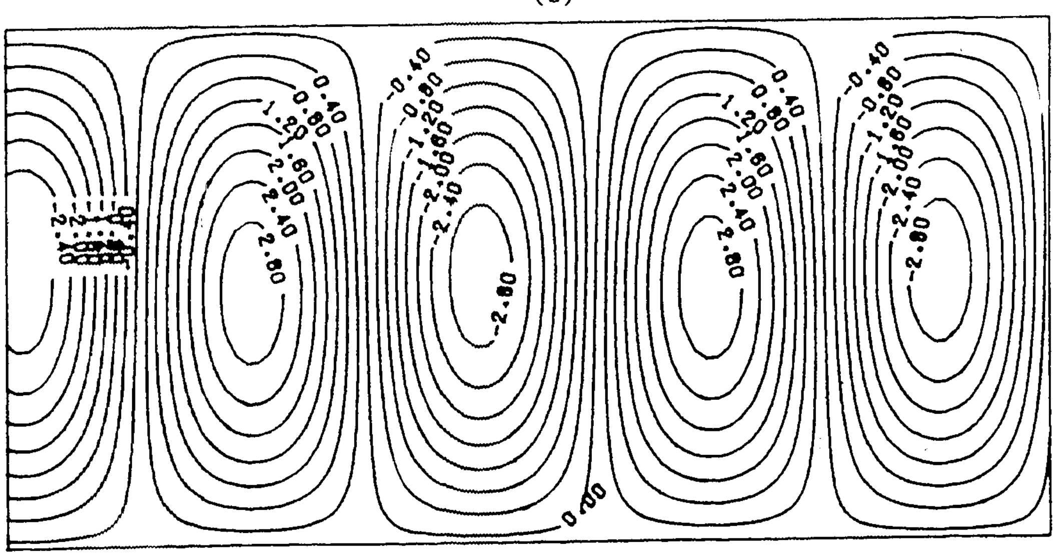
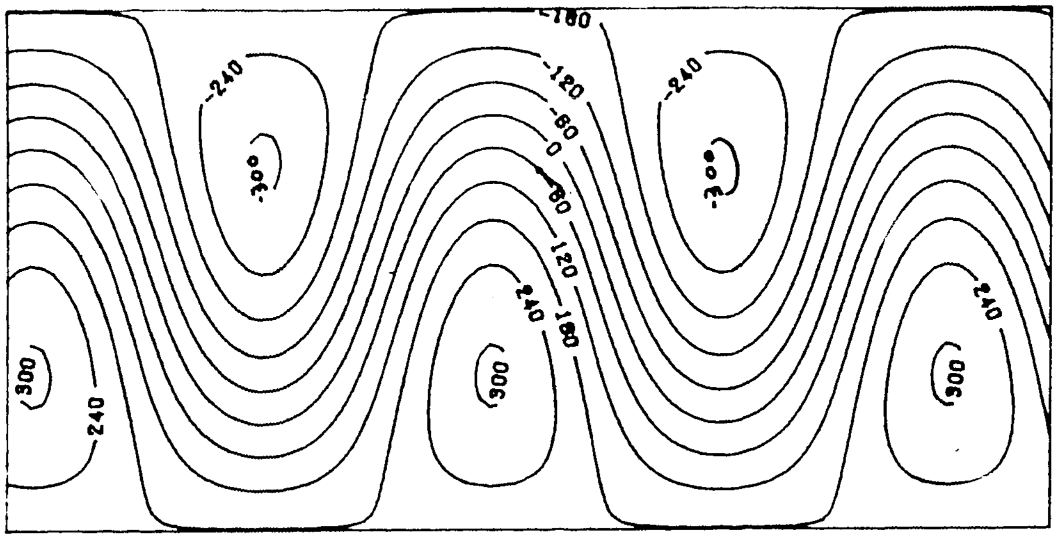


Fig. 5.28 Initial omega field in μbar/sec for Case III in (a) area B and (b) area A.



(a)

(b)

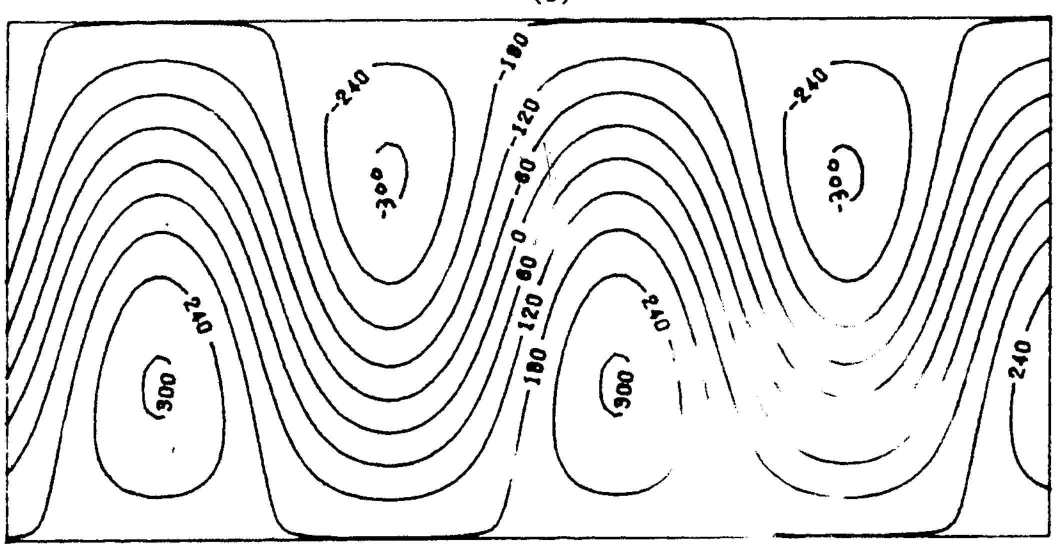
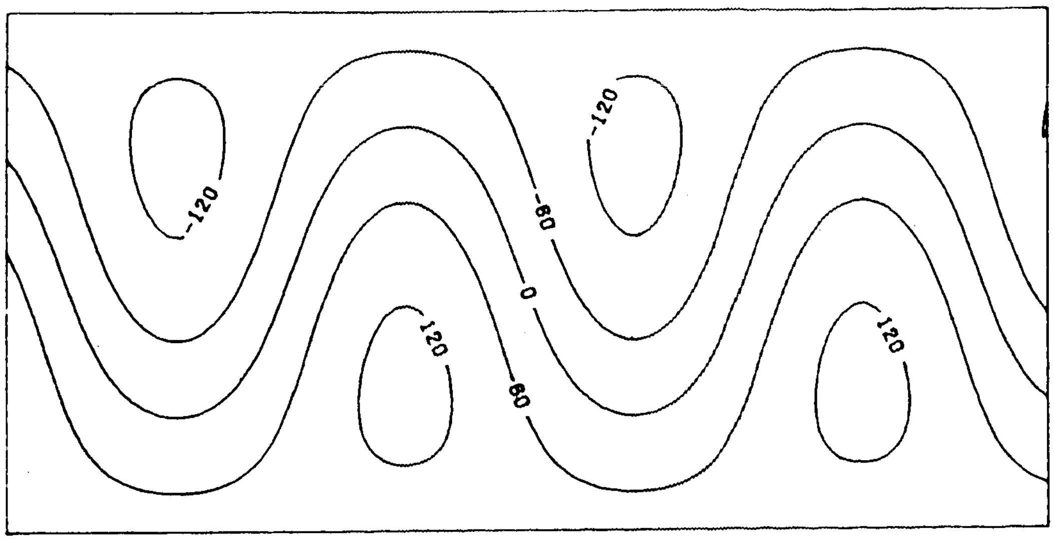


Fig. 5.29 The 24-hour spectral solution for the mean height field in metres for Case III in (a) area B and (b) area A.



(a)

(b)

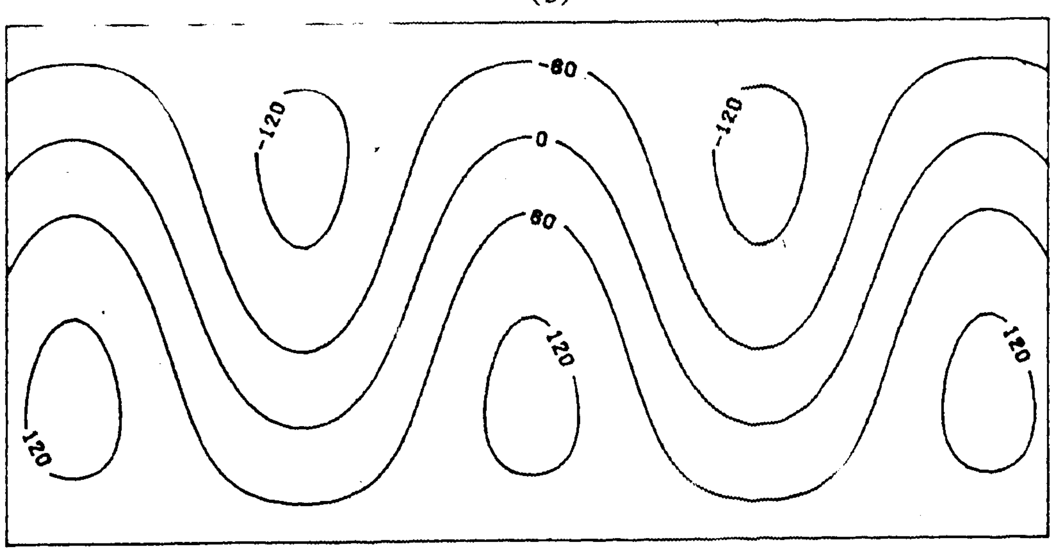
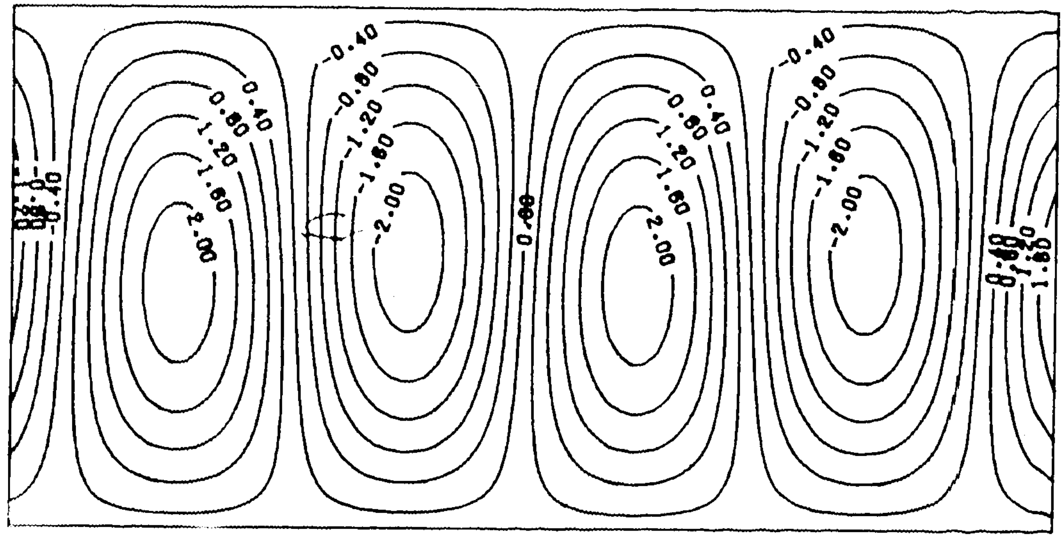


Fig. 5.30 The 24-hour spectral solution for the thickness field in metres for case III in (a) area B and (b) area A.



(a)

(b)

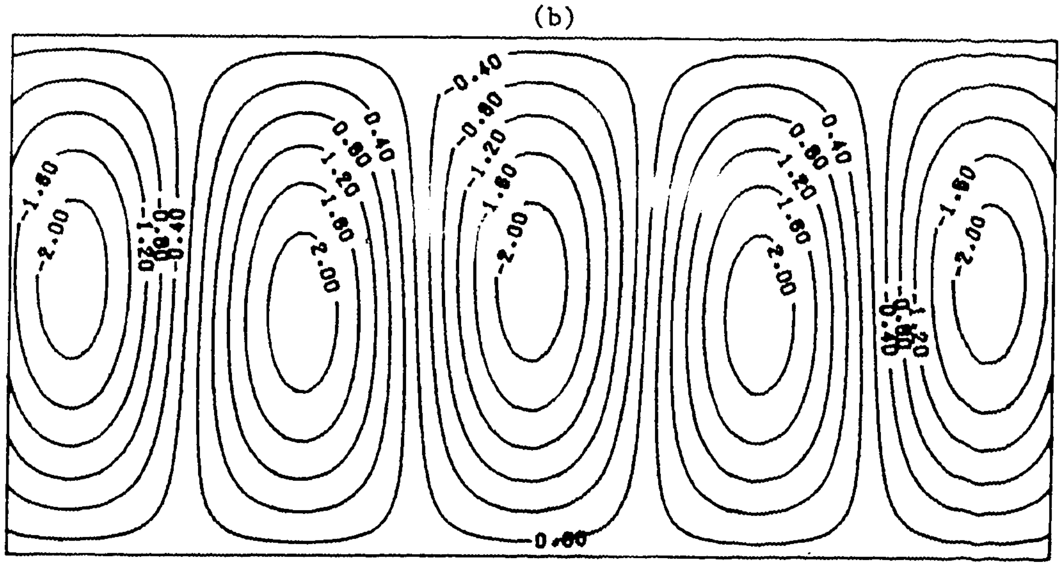
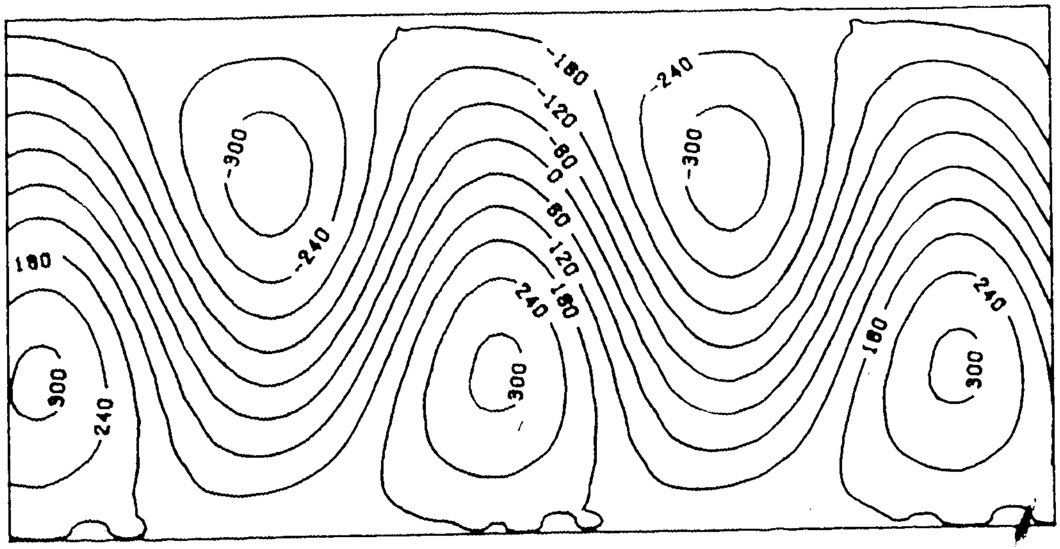


Fig. 5.31 The 24-hour spectral solution for the omega field in μbar/sec for Case III in (a) area B and (b) area A.



(a)

(b)

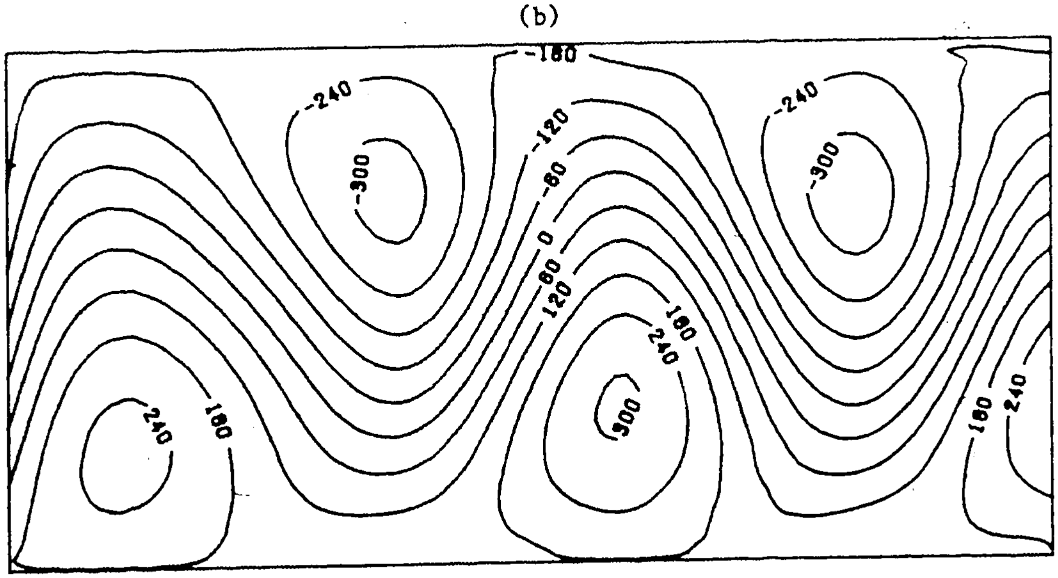
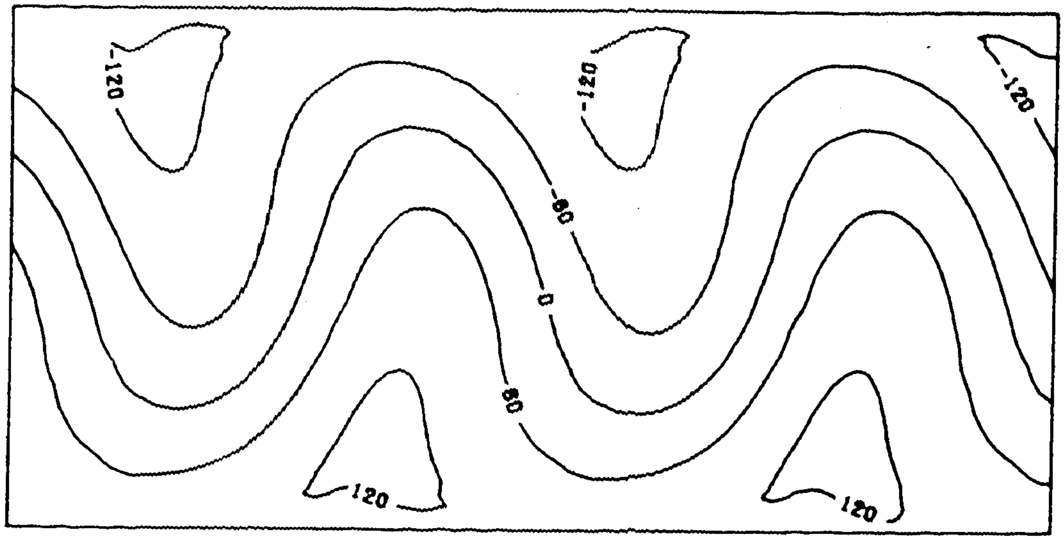


Fig. 5.32 The 24-hour finite-element solution for the mean height field in metres for Case II in (a) area B and (b) area A.



(a)

(b)

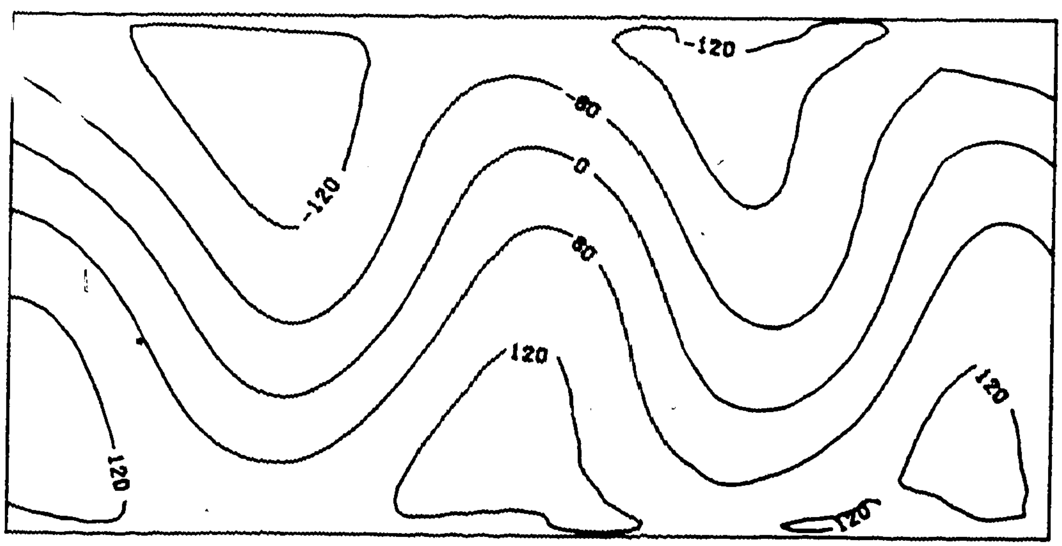
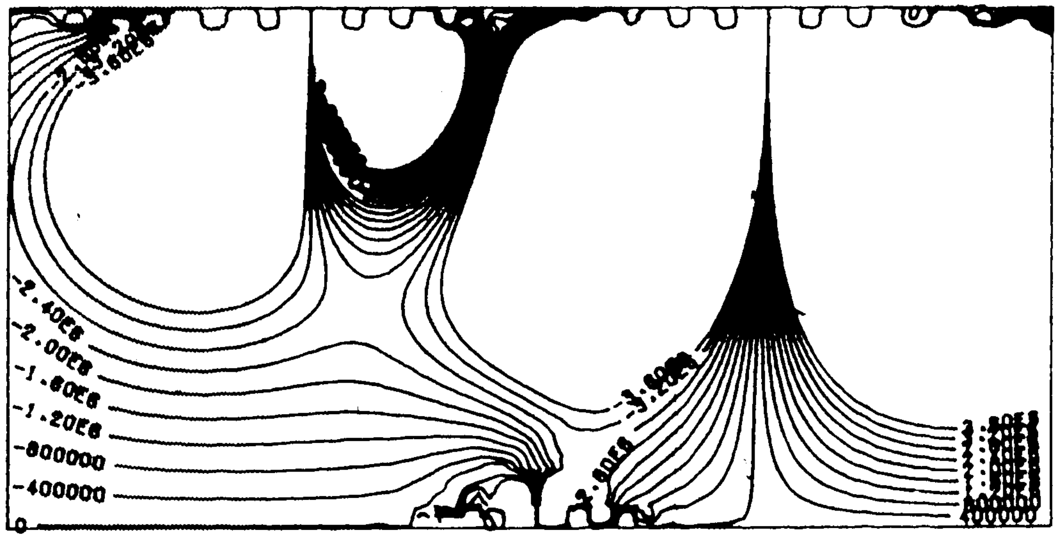


Fig. 5.33 The 24-hour finite-element solution for the thickness field in metres for Case III in (a) area B and (b) area A.



(a)

(b)

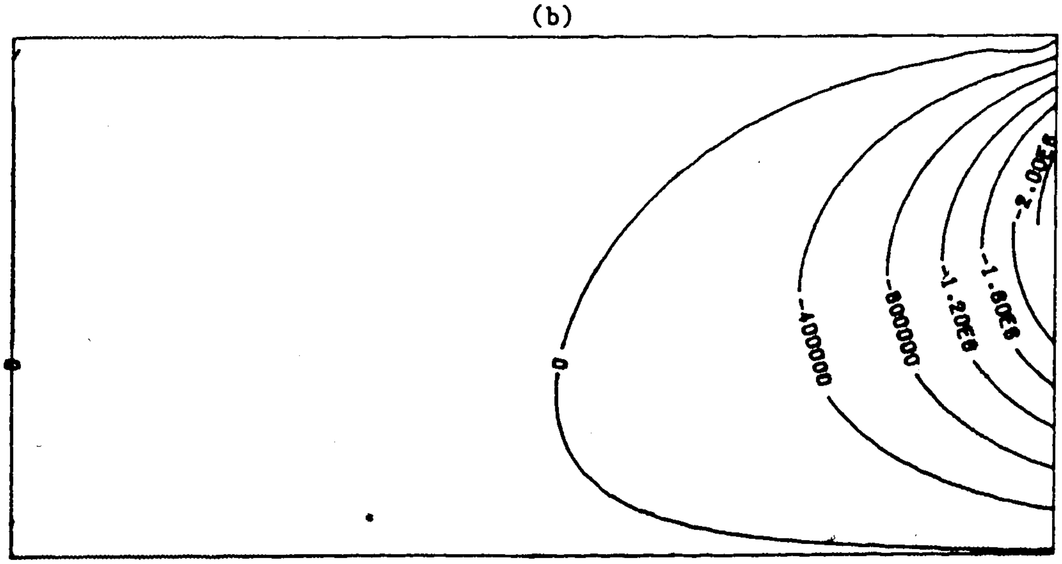
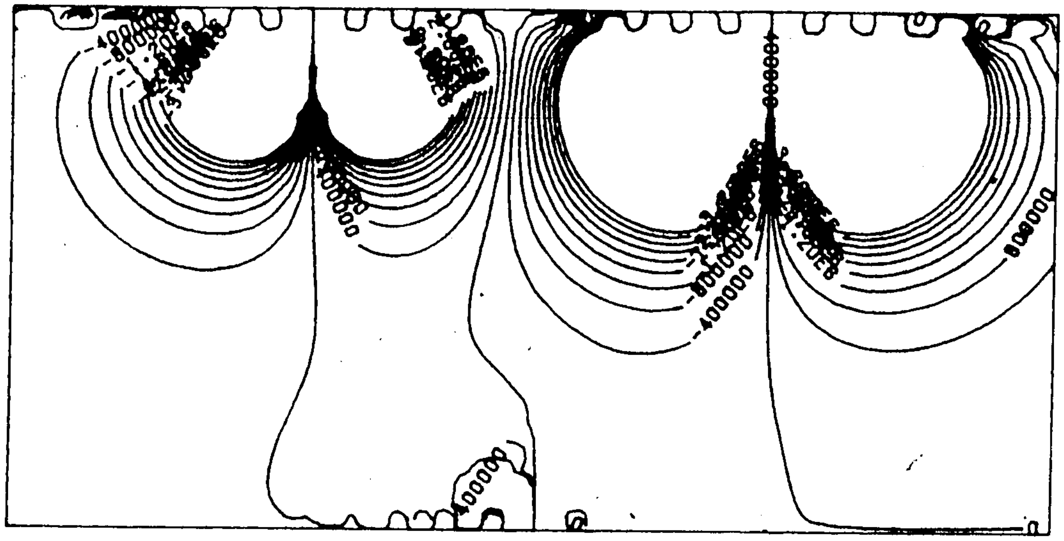


Fig. 5.34 The 30-hour finite-element solution for the mean height field in metres for Case III in (a) area B and (b) area A.



(a)

(b)

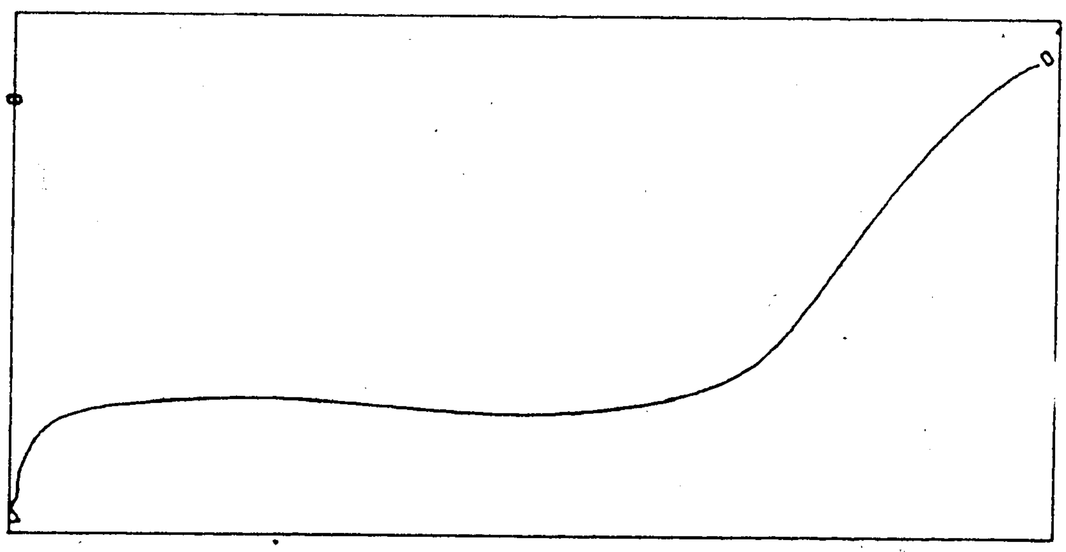


Fig. 5.35 The 30-hour finite-element solution for the thickness field in metres for Case III in (a) area B and (b) area A.

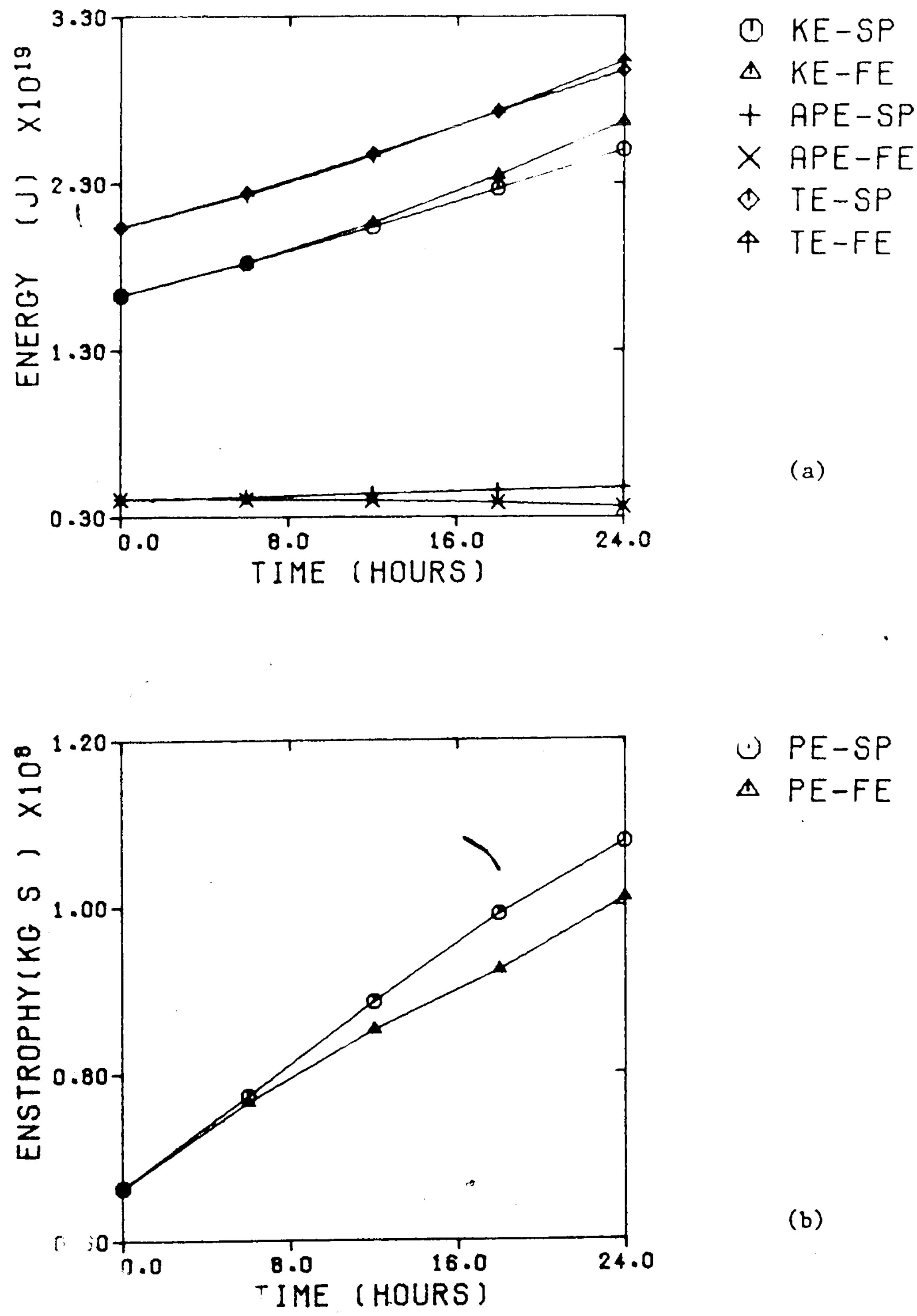


Fig. 5.36 The (a) energies and (b) potential enstrophy of Case III for the two solutions in the verification area.

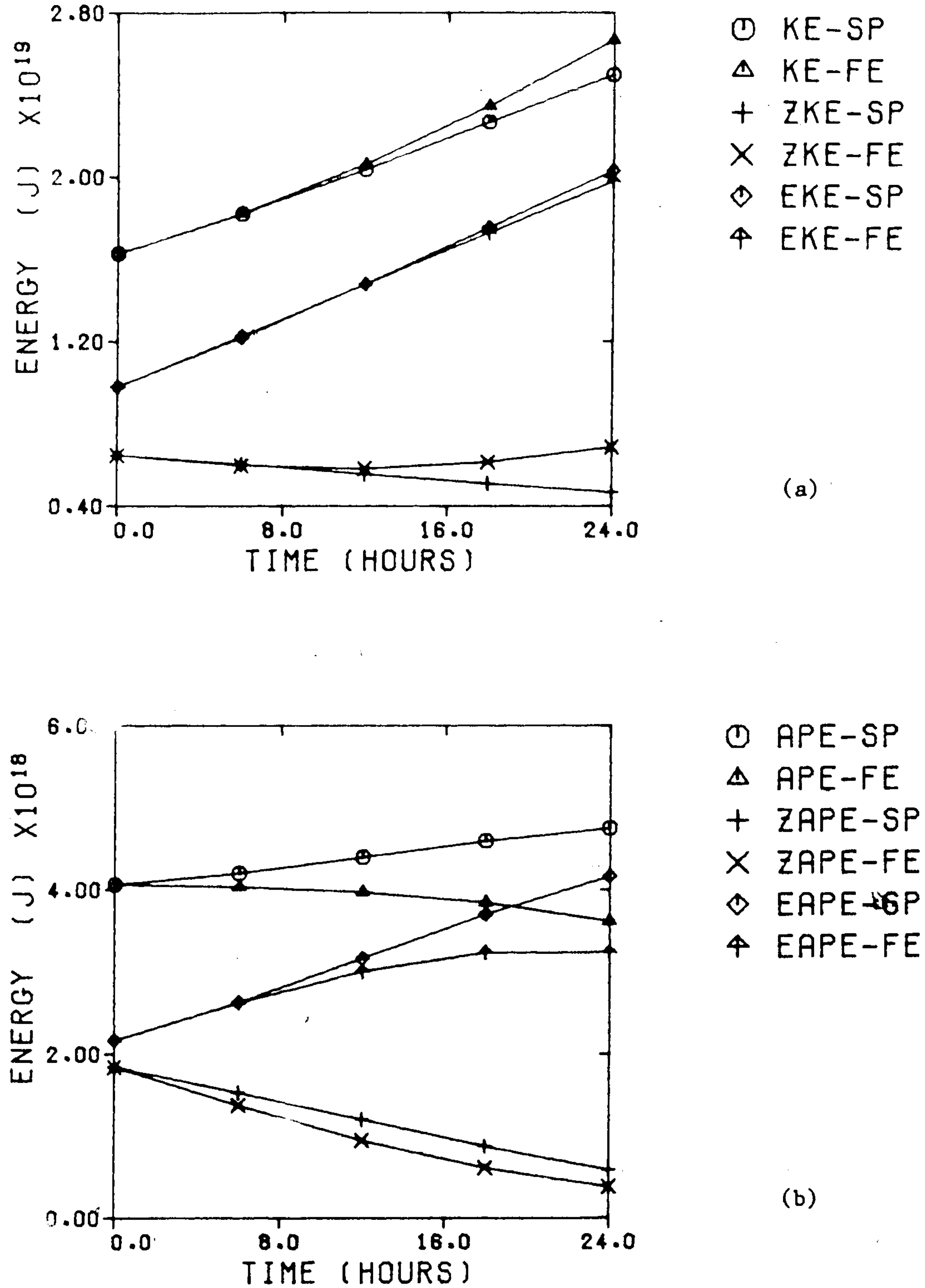


Fig. 5.37 **The (a) kinetic energies and (b) potential energies of Case III for the two solutions in the verification area.**

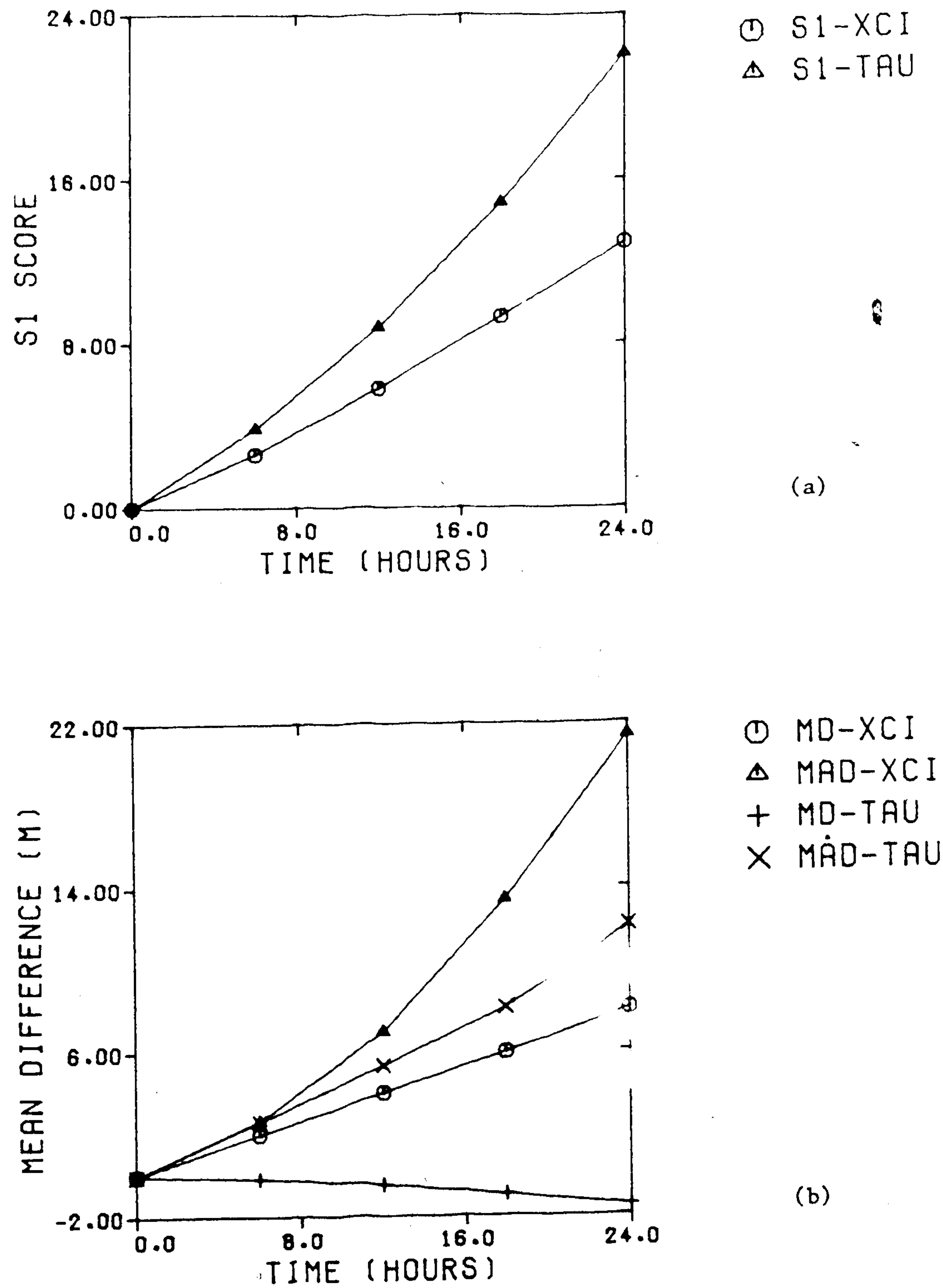
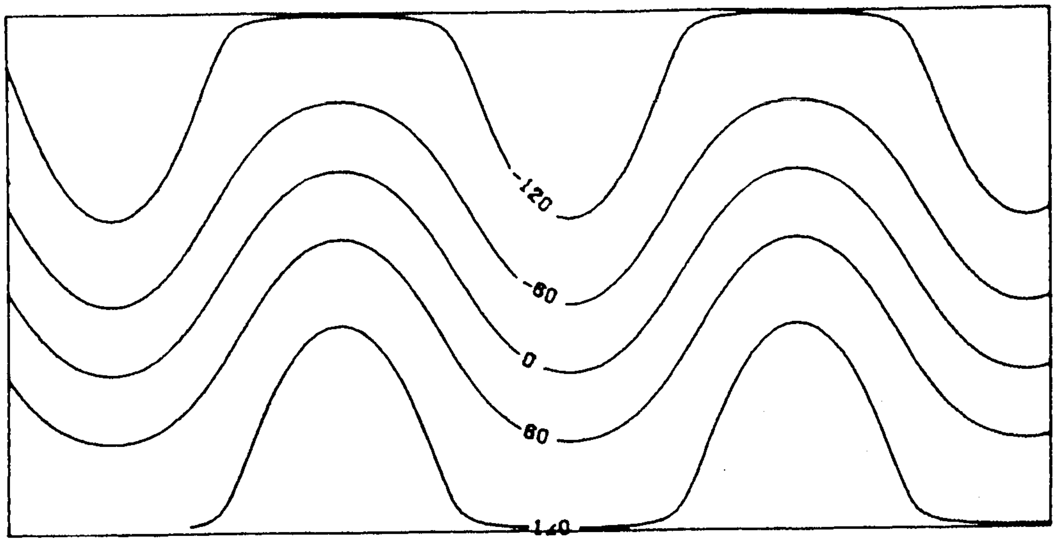


Fig. 5.38 The (a) S1 scores and (b) MD and MAD curves for Case III for the two solutions in the verification area.



(a)

(b)

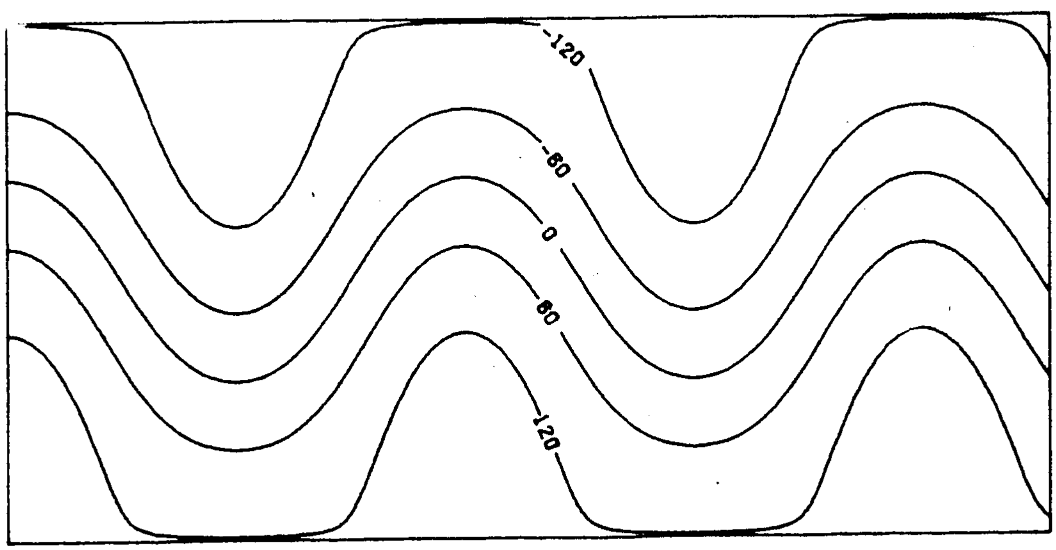
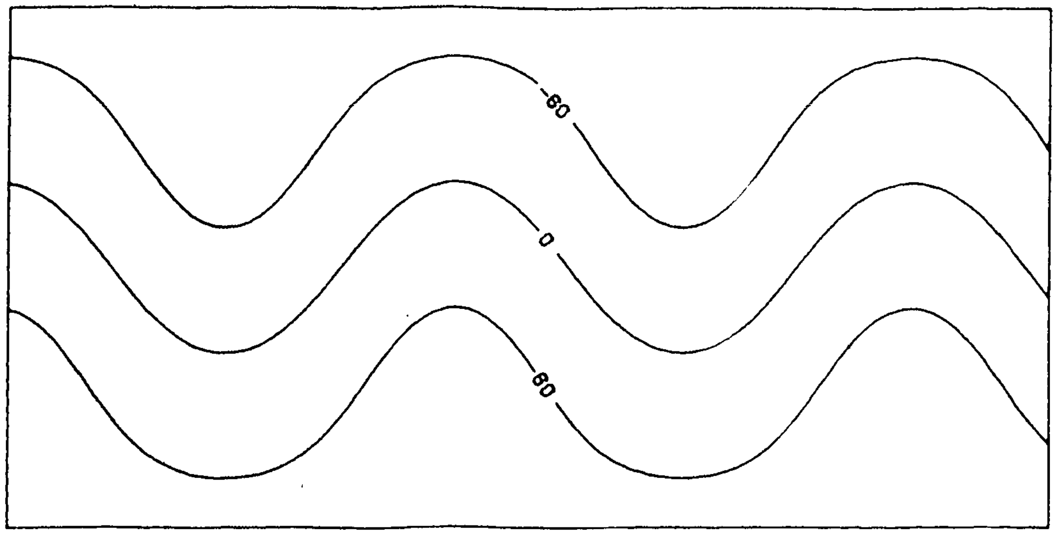


Fig. 5.39 Initial mean height field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

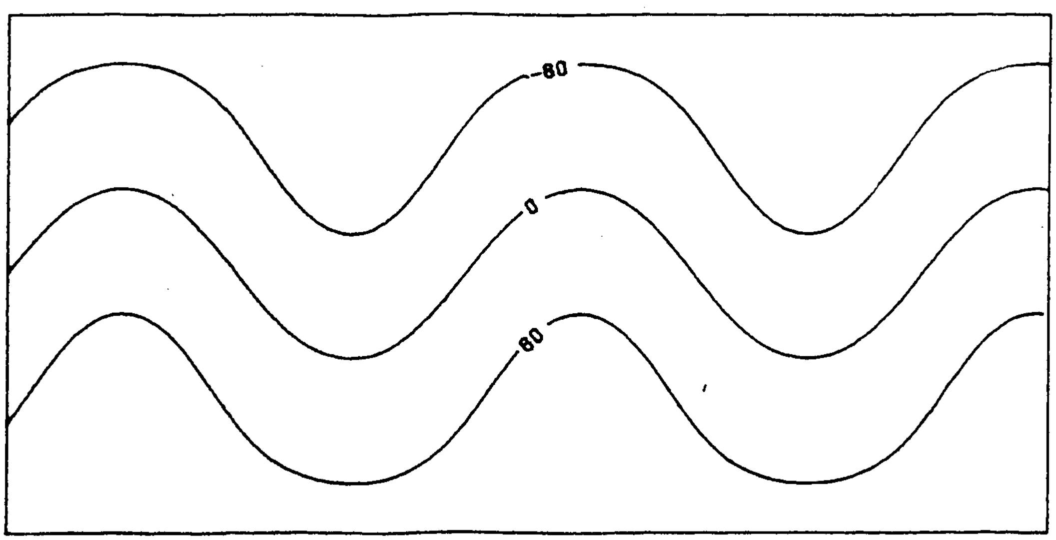
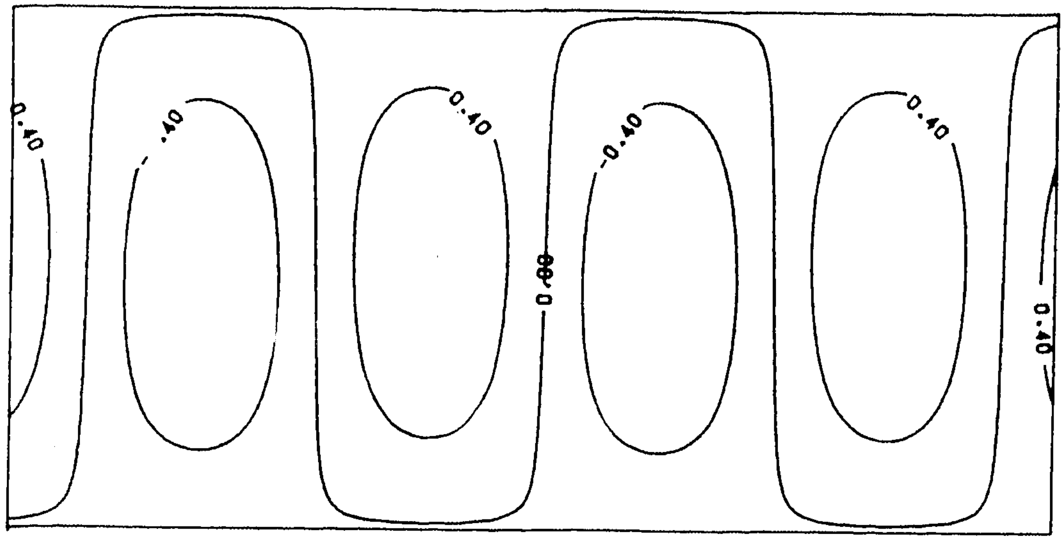


Fig. 5.40 Initial thickness field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

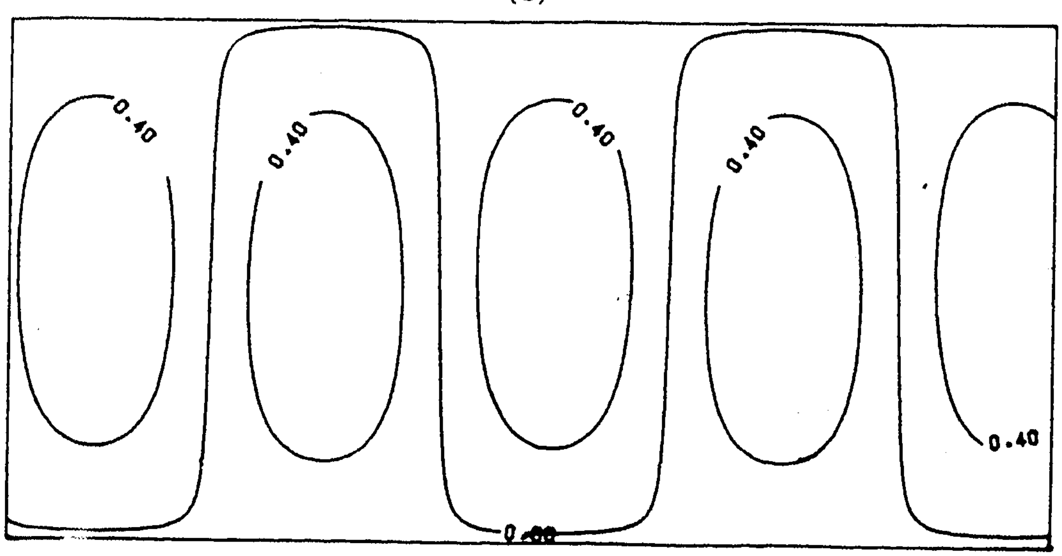
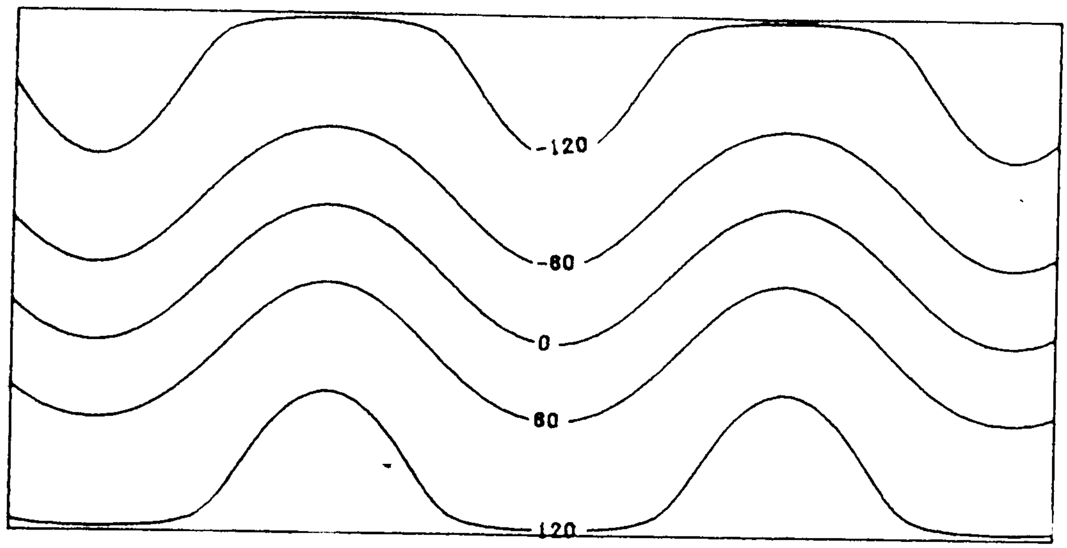


Fig. 5.41 Initial omega field in μbar/sec for Case IV in (a) area B and (b) area A.



(a)

(b)

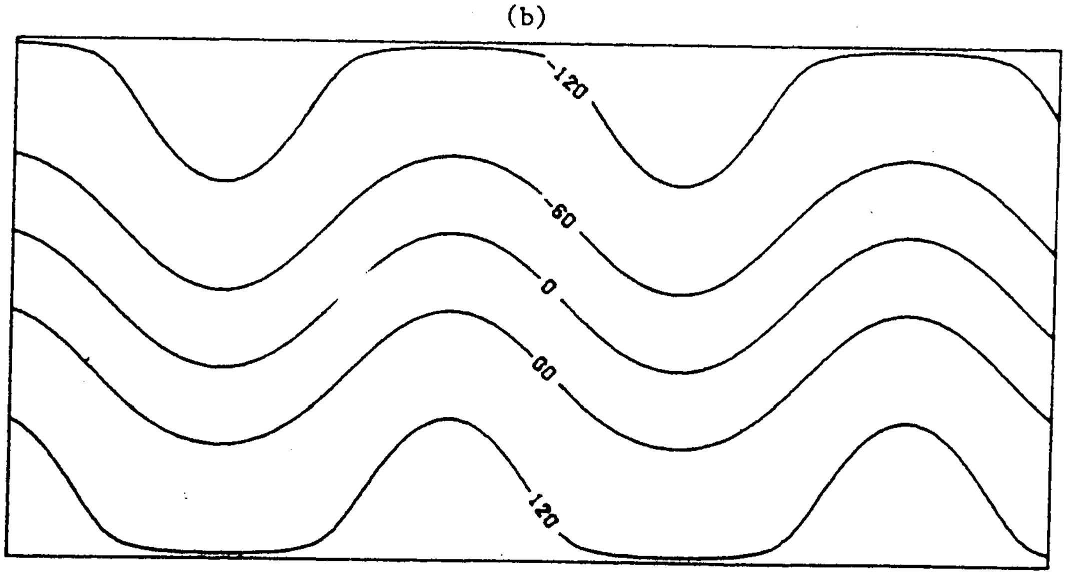
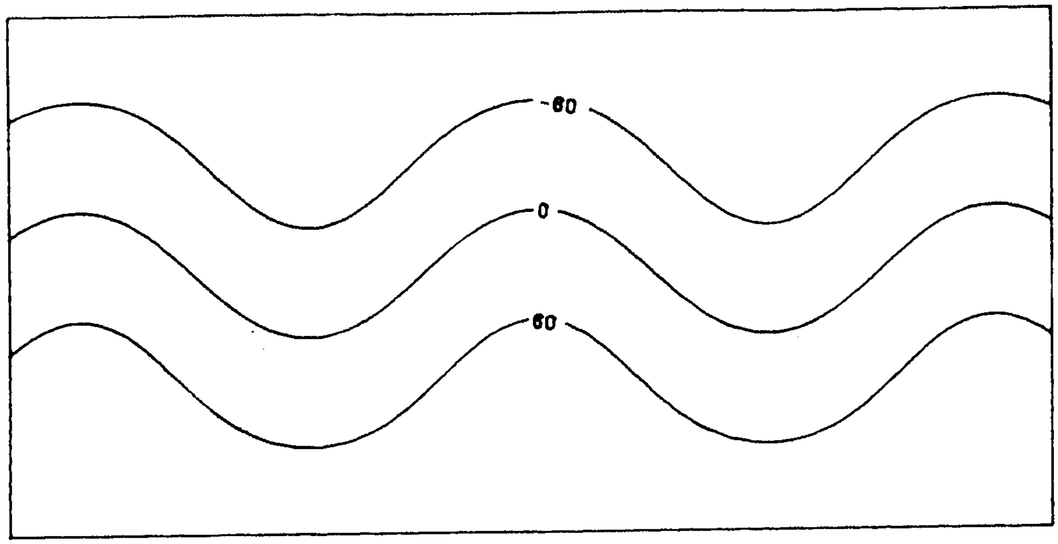


Fig. 5.42 The 48-hour spectral solution for the mean height field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

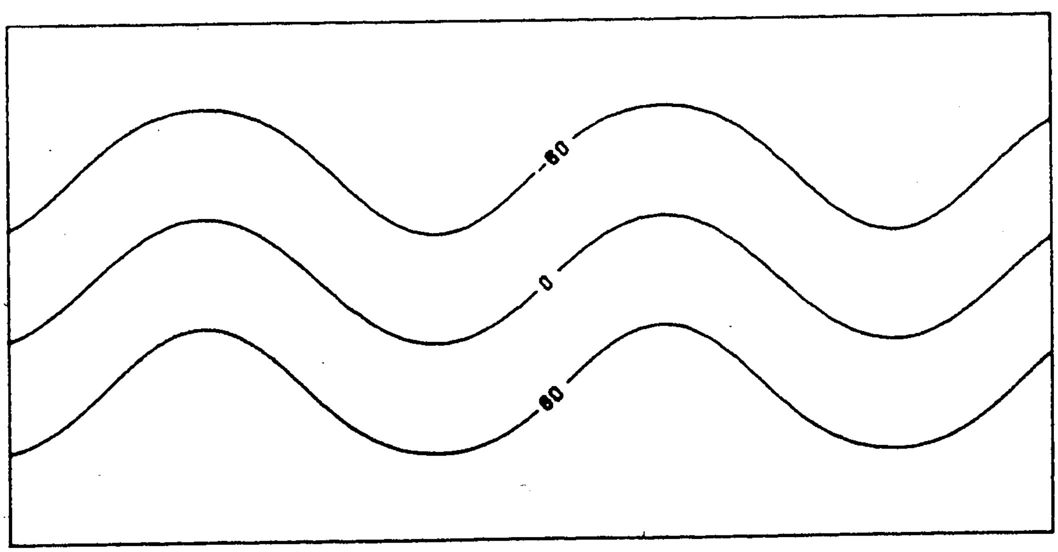
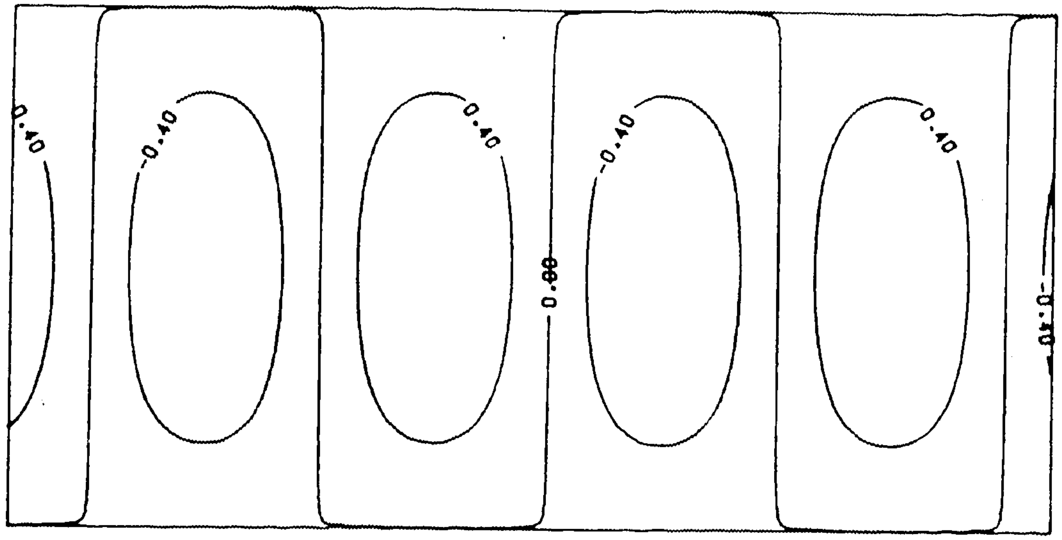


Fig. 5.43 The 48-hour spectral solution for the thickness field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

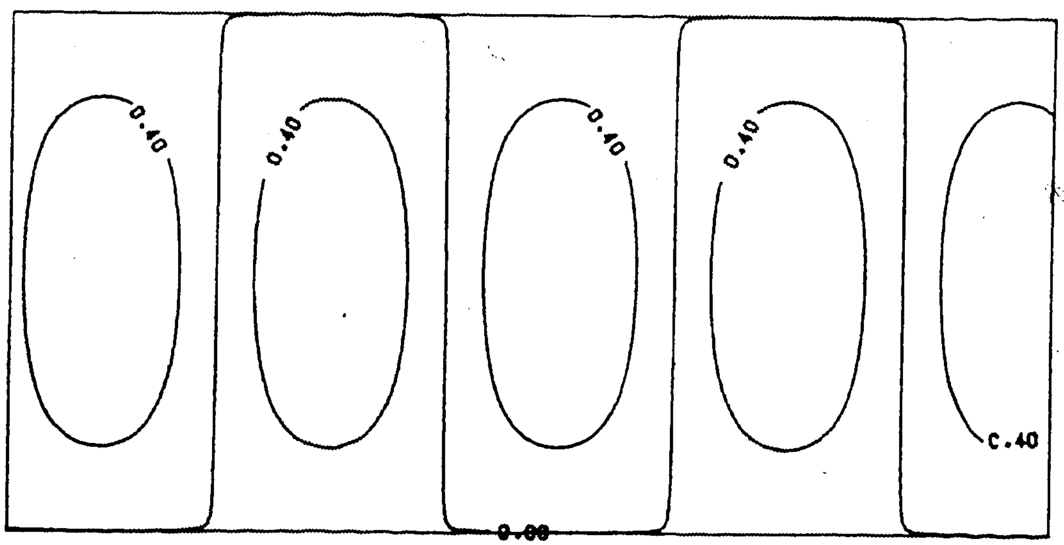
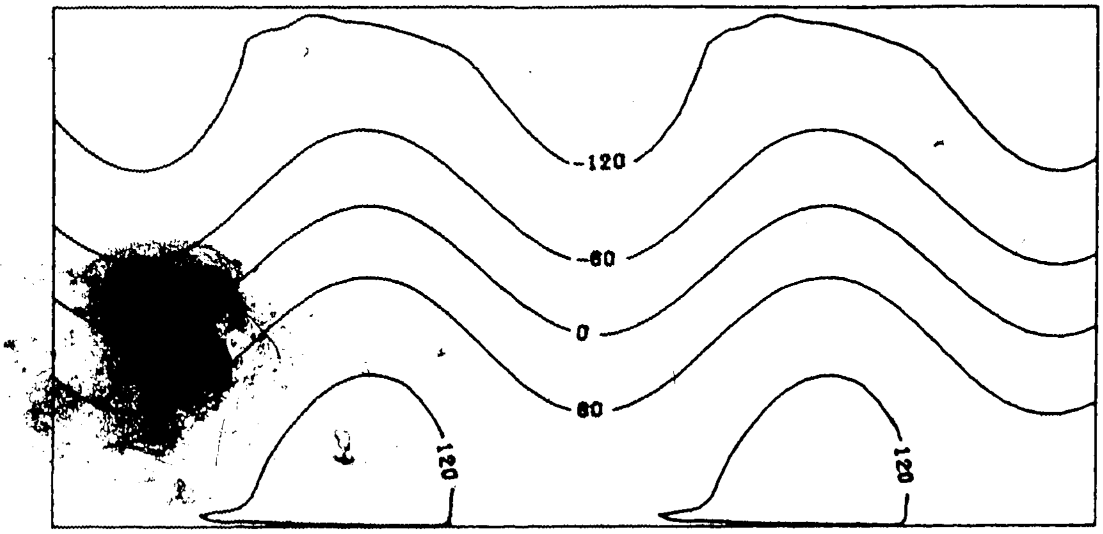


Fig. 5.44 The 48-hour spectral solution for the omega field in μbar/sec for Case IV in (a) area B and (b) area A.



(a)

(b)

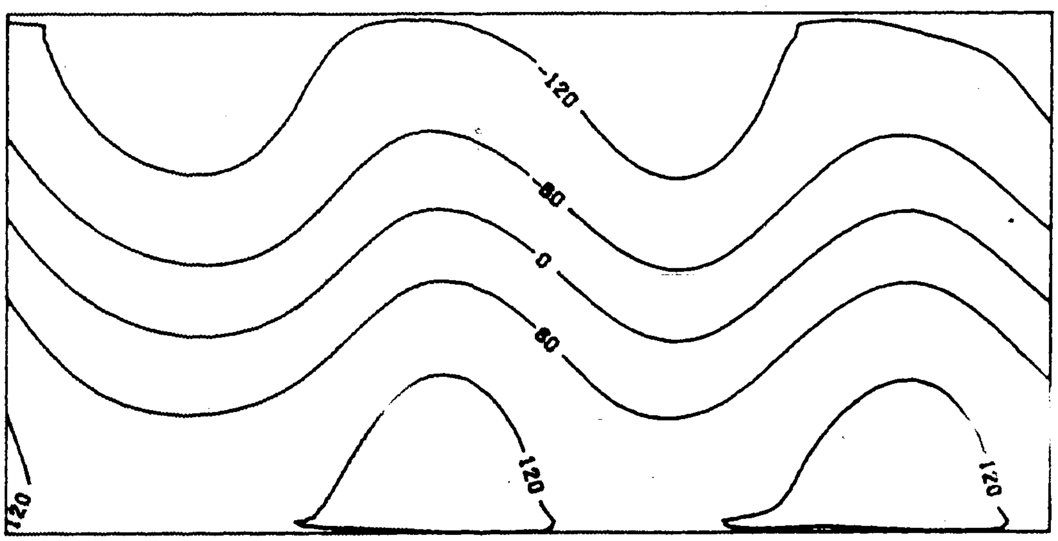
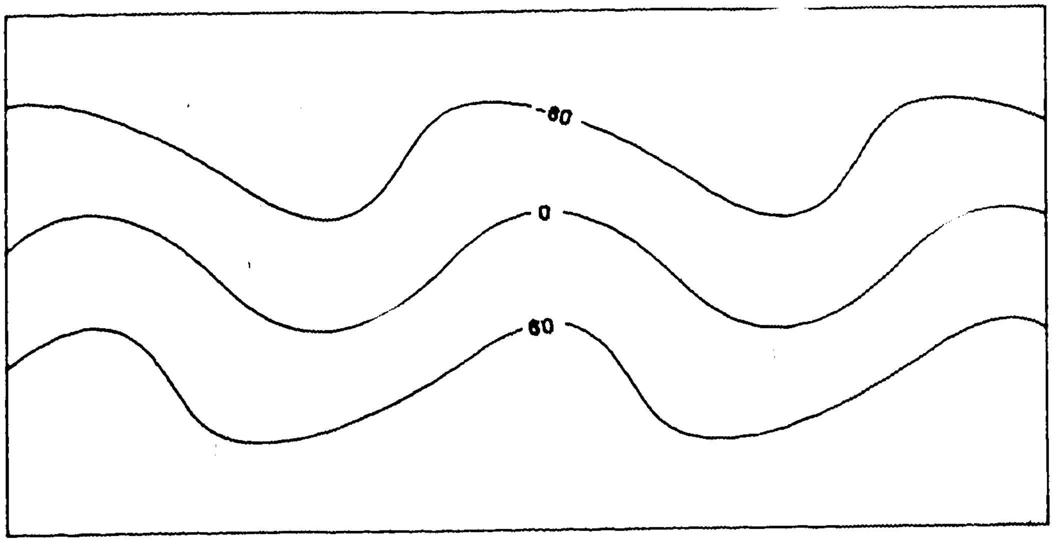


Fig. 5.45 The 48-hour finite-element solution for the mean height field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

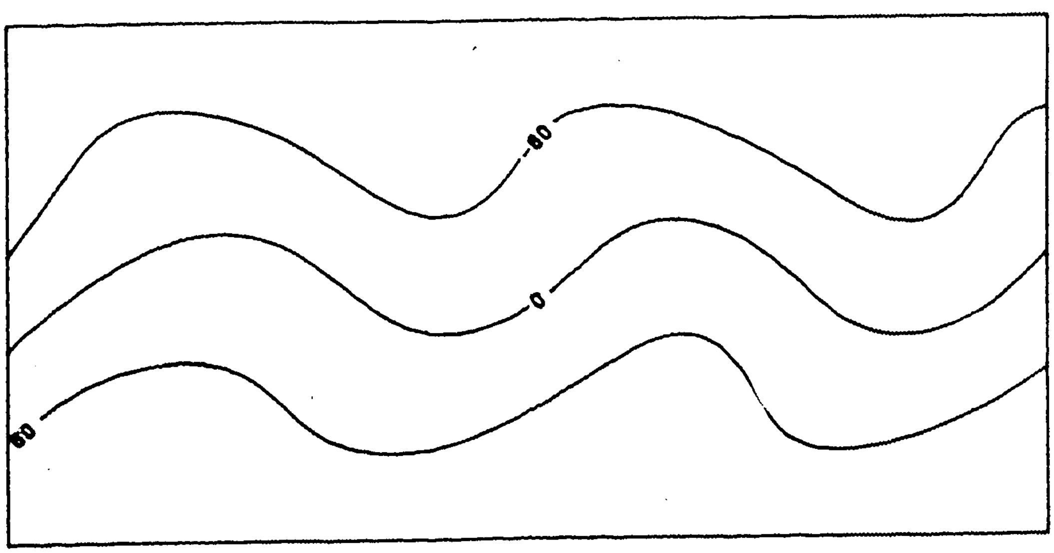
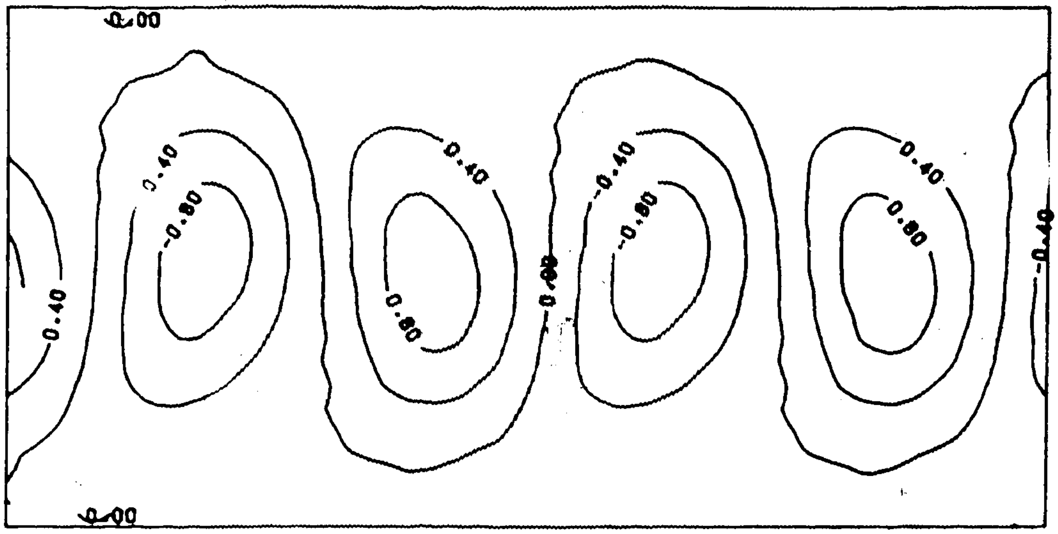


Fig. 5.46 The 48-hour finite-element solution for the thickness field in metres for Case IV in (a) area B and (b) area A.



(a)

(b)

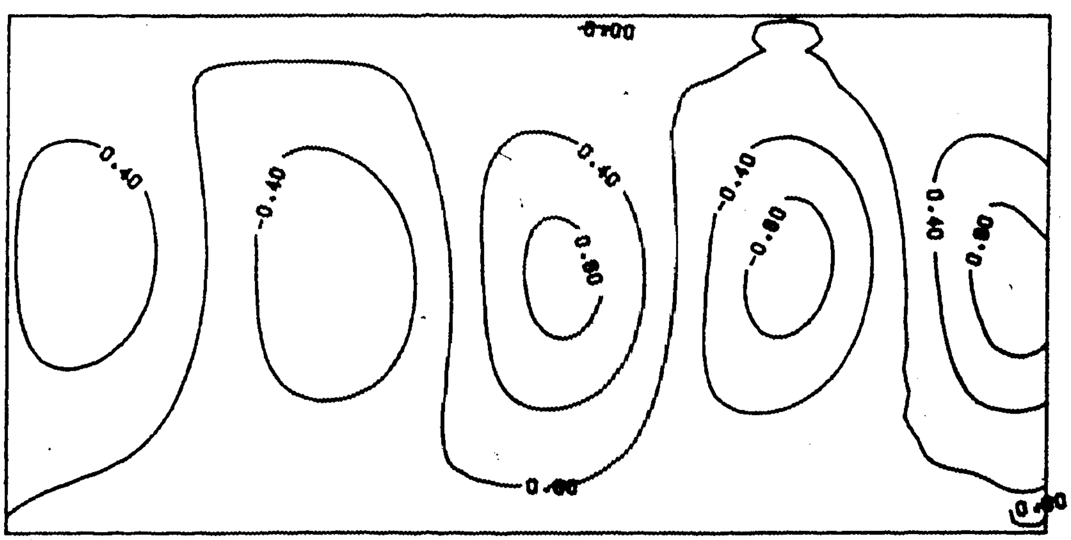


Fig. 5.47 The 48-hour finite-element solution for the omega field in μbar/sec for Case IV in (a) area B and (b) area A.

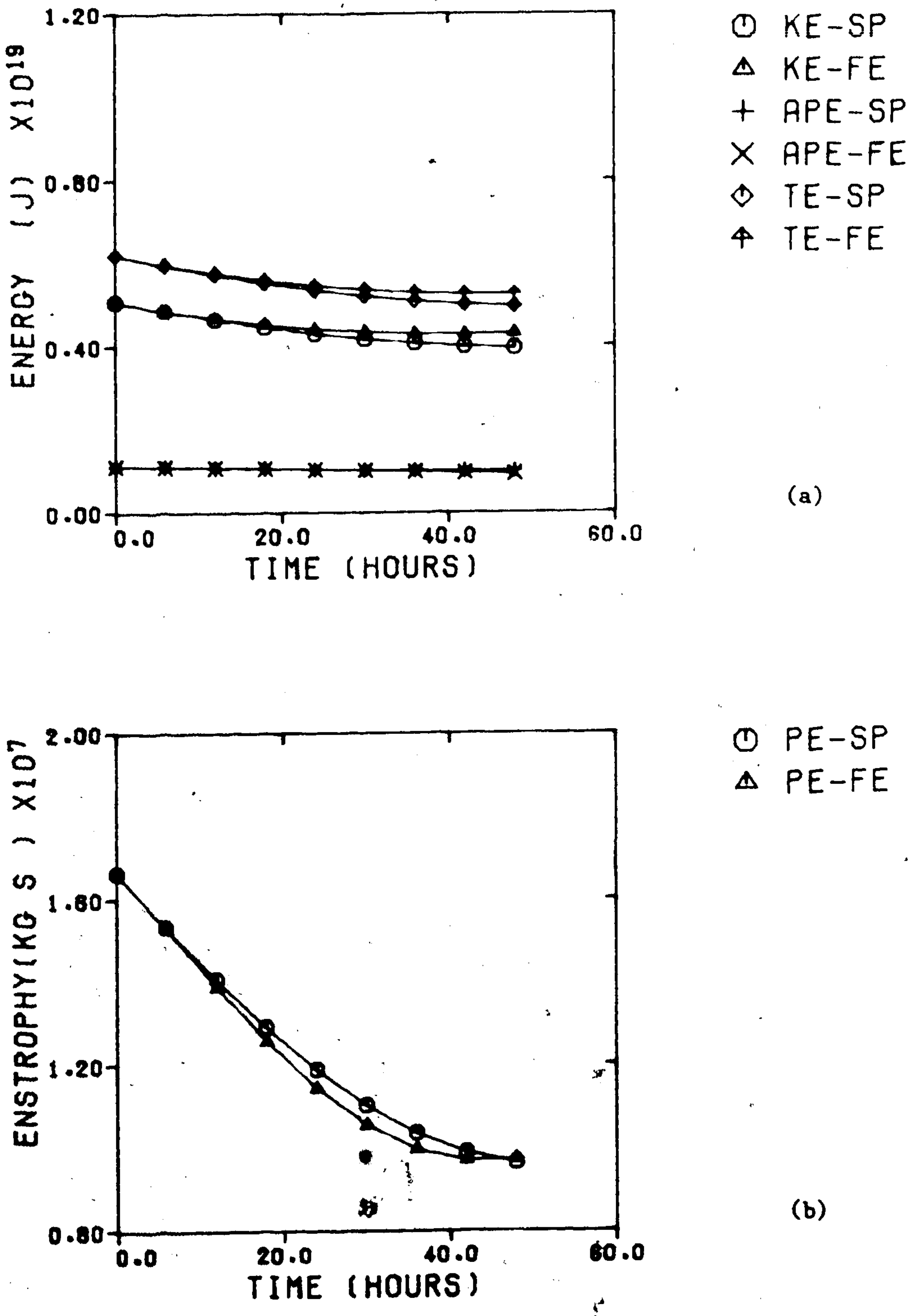


Fig. 5.48 The (a) energies and (b) potential enstrophy of Case IV for the two solutions in the verification area.

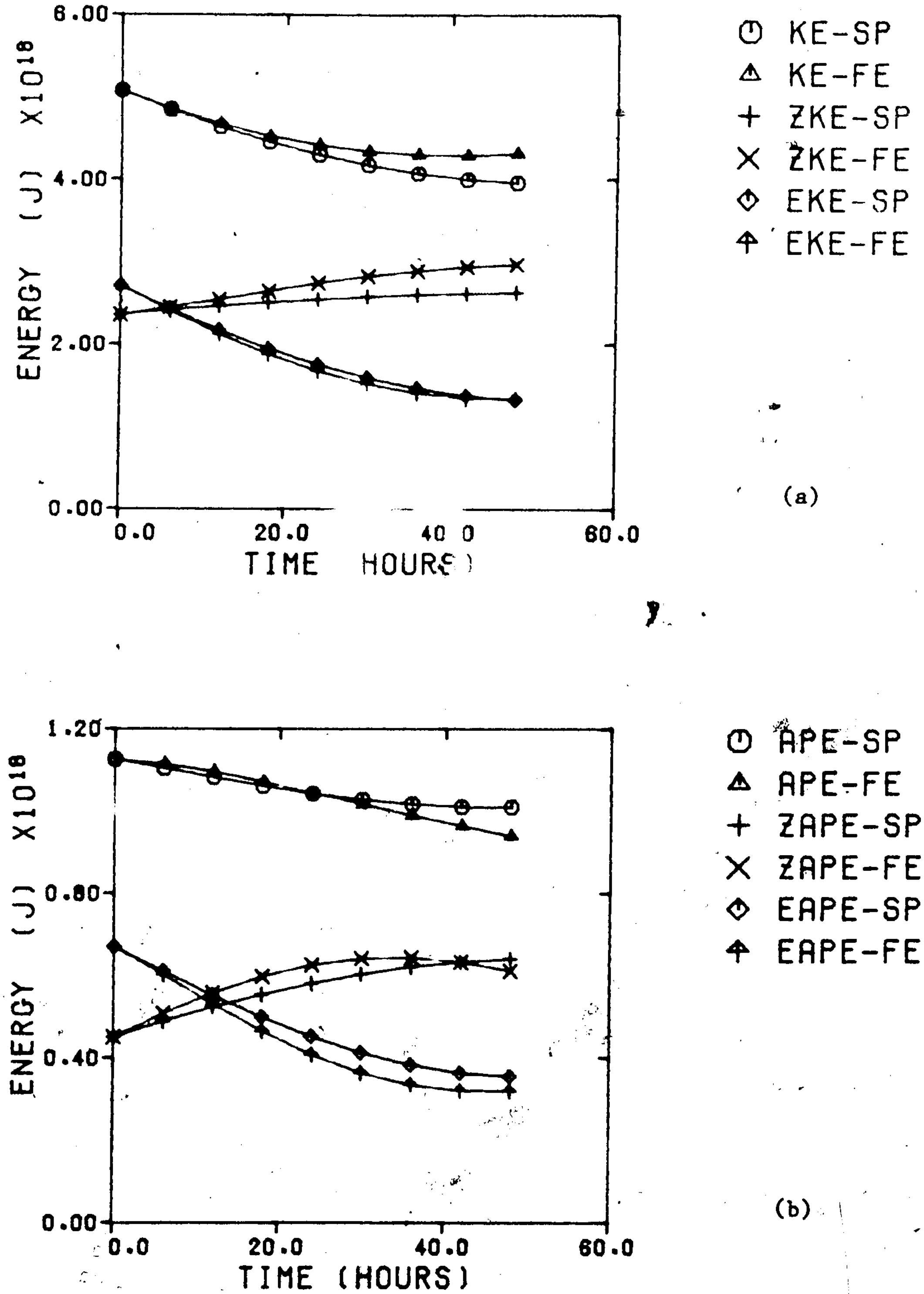
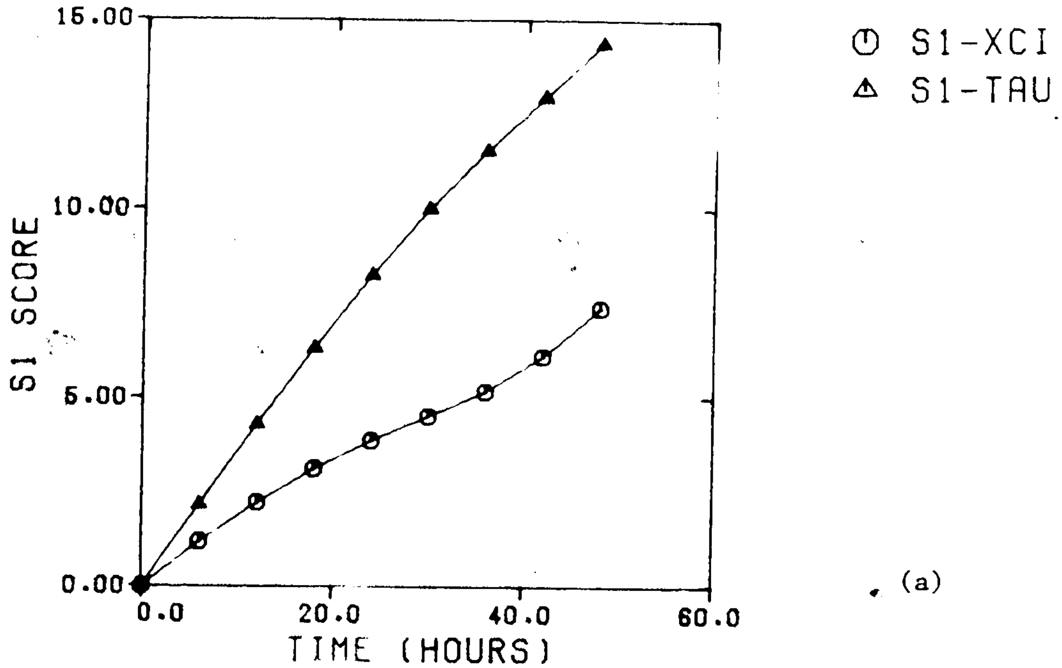


Fig. 5.49 The (a) kinetic energies and (b) potential energies of Case IV for the two solutions in the verification area.



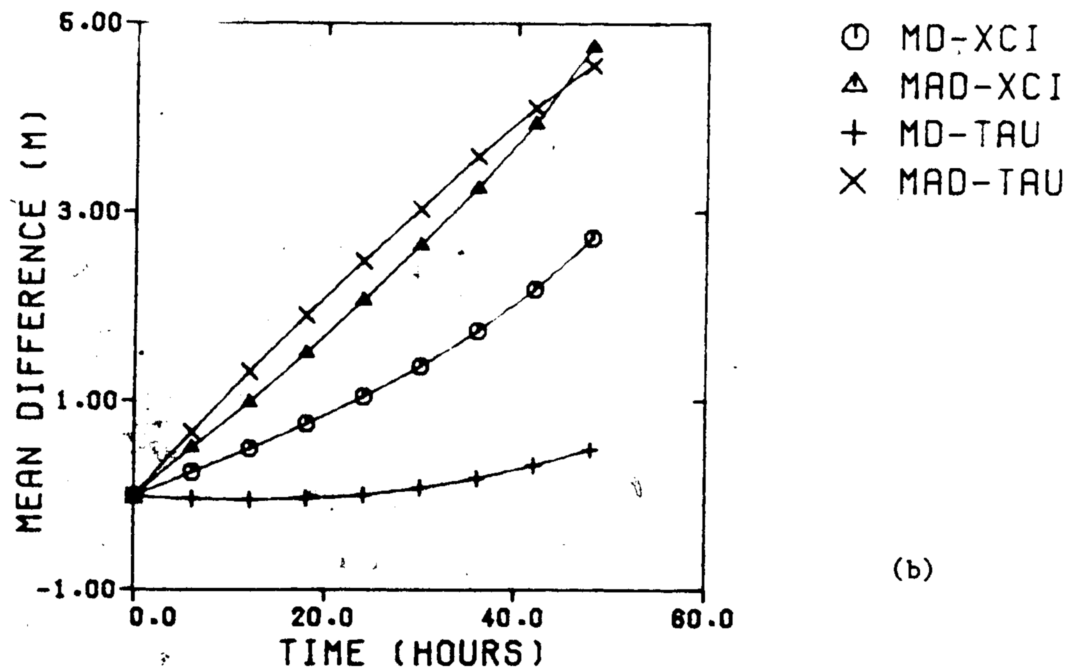
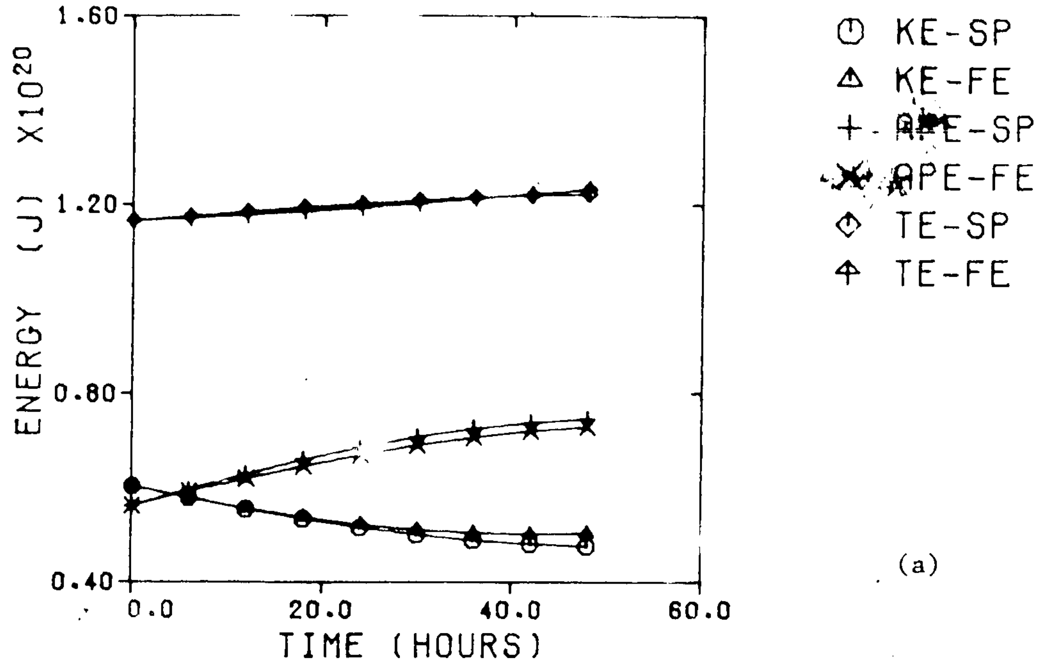
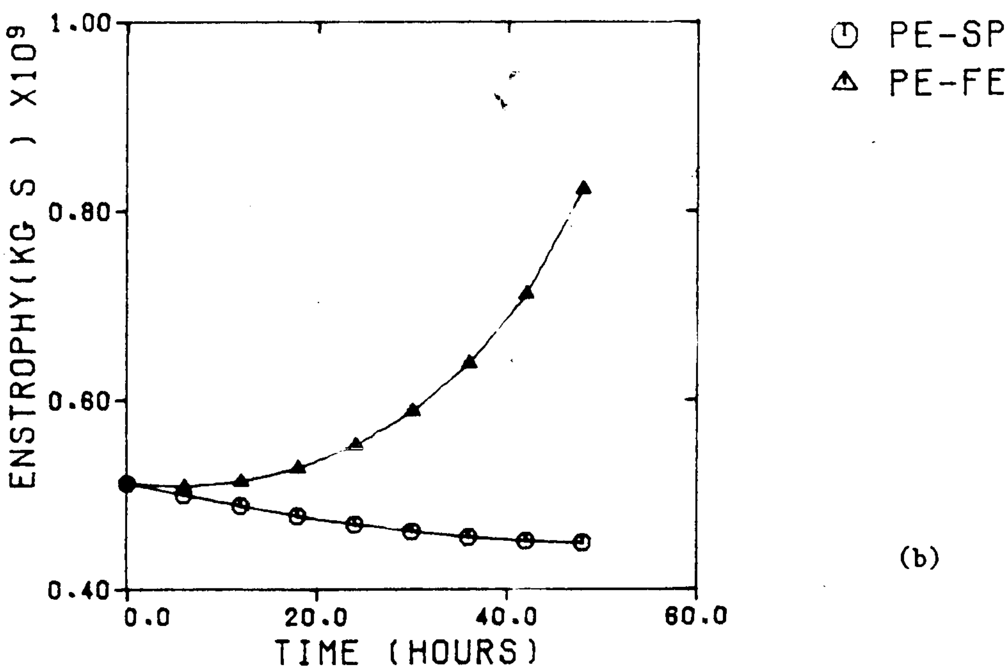


Fig. 5.50 The (a) S1 scores and (b) MD and MAD curves of Case IV for the two solutions in the verification area.



(a)



(b)

Fig. 5.51 The (a) energies and (b) potential enstrophy of Case IV for the two solutions in the entire domain.

REFERENCES

Argyris, J.H., 1960: Energy Theorems and Structural Analysis. Butterworth, England. (Reprinted from Aircraft Eng., 1954-1955).

Baer, F., 1970: Analytical solutions to low-order spectral systems. Archiv. Met. Geoph. Biokl., Ser. A, 19, 255-282.

Baer, F., and F.N. Alyea, 1971: Effects of spectral truncation on general circulation and long range prediction. J. Atmos. Sci., 29, 768-772.

Bjerknes, V., 1904: Das problem der wettervorhersage, betracht von standpunke der mechanik and der physik. Meteor. Z., 21, 1-7.

Bourke, W., 1972: An efficient, one level, primitive equation spectral model. Mon. Wea. Rev., 100, 683-689.

Bourke, W., 1974: A multi-level spectral model, 1, formulation and hemispheric integrations. Mon. Wea. Rev., 102, 687-701.

Charney, J.G., R. Fjortoft, and J. von Neumann, 1950: Numerical integration of the barotropic vorticity equation. Tellus, 2, 237-254.

Clough, R.W., 1960: The finite-element method in plane stress analysis. Proc. Am. Soc. Civil Engineers, 87, 345-378.

Clough, R.W., 1965: The finite-element method in structural mechanics. Chapter 7 of Stress Analysis (eds. O.C. Zienkiewicz and G.S. Holister), Wiley.

Cressman, G.P., 1958: Barotropic divergence and very long atmospheric waves. Mon. Wea. Rev., 87, 367-374.

Cressman, G.P., 1963: A three-level model suitable for daily numerical forecasting. Tech. Memo. 22, National Meteorological Center, NWS/NOAA 25 pp.

Cullen, M.J.P., 1973: A simple finite-element method for meteorological problems. J. Inst. Math. Applic., 11, 15-31.

Cullen, M.J.P., 1974a: A finite-element method for a nonlinear initial value problem. J. Inst. Math. Applic., 13, 233-247.

Cullen, M.J.P., 1974b: Integrations of the primitive equations on a sphere using the finite element method. Quart. J. Roy. Meteor. Soc., 100, 555-562.

Cullen, M.J.P., 1976: On the use of artificial smoothing in Galerkin and finite difference solutions of the primitive equations. Quart. J. Roy. Meteor. Soc., 102, 77-93.

Cullen, M.J.P., and C.D. Hall, 1979: Forecasting and general circulation results from finite-element models. Quart. J. Roy. Meteor. Soc., 105, 571-592.

Daley, R., C. Girard, J. Henderson, and I. Simmonds, 1976: Short-term forecasting with a multi-level spectral primitive equation model, Part I, model formulation. Atmosphere, 14, 98-134.

Davies, H.C., 1976: A lateral boundary formulation for multi-level prediction models. Quart. J. Roy. Meteor. Soc., 102, 405-418.

Eliasen, E., B. Machenhauer, and E. Rasmussen, 1970: On a numerical method for integration of the hydrodynamical equations with a spectral representation of the horizontal fields. Institute of Theoretical Meteorology, University of Copenhagen, Report No. 2.

Ellsaesser, H.W., 1966: Evaluation of spectral versus grid methods of hemispheric numerical weather prediction. J. Appl. Meteor., 5, 246-262.

Galerkin, B.G., 1915: Rods and plates. Series occurring in various questions concerning the elastic equilibrium of rods and plates. Engineers Bulletin, 19, 897-908.

Grammeltvedt, A., 1969: A survey of finite difference schemes for the primitive equations for a barotropic fluid. Mon. Wea. Rev., 97, 384-404.

Haltiner, G.J., 1971: Numerical Weather Prediction. John Wiley and Sons, Inc., New York, 317 pp.

Held, I.M., 1975: Momentum transport by quasi-geostrophic eddies. J. Atmos. Sci., 32, 1494-1497.

Holton, J.R., 1972: An Introduction to Dynamic Meteorology. Academic Press, New York, 319 pp.

Hornstein, R.A., 1978: Weather and Why. Atmospheric Environment Service, Environment Canada, 61 pp.

Hoskins, B.J., and A.J. Simmons, 1975: A multi-layer spectral model and the semi-implicit method. Quart. J. Roy. Meteor. Soc., 101, 637-656.

Howcroft, J., 1971: Local forecast model: present status and preliminary verification. Office Note No. 50, National Meteorological Center, National Weather Service, Washington, D.C., 22 pp.

Jeffreys, H., 1922: On the dynamics of wind. Quart. J. Roy. Meteor. Soc., 48, 29-47.

Kreider, D.L., R.G. Kulier, D.R. Ostberg, and F.W. Perkins, 1966: An Introduction to Linear Analysis. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 773 pp.

Lilly, D.K., 1965: On the computational stability of numerical solutions of time-dependent nonlinear geophysical fluid dynamics problems. Mon. Wea. Rev., 93, 11-26.

Lorenz, E.N., 1955: Available potential energy and the maintenance of the general circulation. Tellus, 7, 157-167.

Lorenz, E.N., 1960: Maximum simplification of the dynamic equations. Tellus, 12, 243-254.

Machenhauer, B., and R. Daley, 1972: A baroclinic primitive equation model with a spectral representation in three dimensions. Institute of Theoretical Meteorology, University of Copenhagen, Report No. 4.

McHenry, D., 1943: A lattice analogy for the solution of plane stress problems. J. Inst. Civ. Eng., 21, 59-82.

Newmark, N.M., 1949: Numerical methods of analysis in bars, plates and elastic bodies in "Numerical Methods in Analysis in Engineering". (Ed. L.E. Grinter). Macmillan, 1949.

Norrie, D.H., and G. De Vries, 1976: Finite Element Bibliography. IFI/ Plenum, New York, 686 pp.

Orszag, S.A., 1970: Transform method for calculation of vector-coupled sums: application to the spectral form of the vorticity equation. J. Atmos. Sci., 27, 329-335.

Pedlosky, J., 1979: Geophysical Fluid Dynamics. Springer-Verlag, New York, 624 pp.

Perkey, D.J., and C.W. Kreitzberg, 1976: A time-dependent lateral boundary scheme for limited-area primitive equation models. Mon. Wea. Rev., 104, 744-755.

Phillips, N.A., 1951: A simple three-dimensional model for the study of large-scale extratropical flow patterns. J. Meteor., 8, 381-394.

Phillips, N.A., 1954: Energy transformations and meridional circulations associated with simple baroclinic waves in a two-level quasi-geostrophic model. Tellus, 6, 273-286.

Phillips, N.A., and J. Shukla, 1973: On the strategy of combining coarse and fine grid meshes in numerical weather prediction. J. Appl. Meteor., 12, 1102-1113.

Ralston, A., 1965: A First Course in Numerical Analysis. McGraw-Hill Inc., New York, 578 pp.

Raylegh, Lord (J.W. Strutt), 1870: On the theory of resonance. Trans. Roy. Soc. (London), A161, 77-118.

Richardson, L.F., 1921: Weather Prediction by Numerical Process. Cambridge Univ. Press, London. Reprinted by Dover.

Ritz, W., 1909: Uber eine neue methode zur losung gewissen variationsprobleme der mathematischen physik. J. Reine. Angew. Math., 135, 1-61.

Robert, A.J., 1966: The integration of a low-order spectral form of the primitive meteorological equations. J. Meteor. Soc. Japan, 44, 237-245.

Robert, A.J., 1970: Forecast experiments with a spectral model. Proc. Eigth Stanstead Seminar, Publications in Meteorology, No. 97, McGill University, Montreal.

Shuman, F.G., and J.B. Hovermale, 1968: An operational six-layer primitive equation model. J. Appl. Meteor., 7, 525-547.

Staniforth, A.N., and R.W. Daley, 1977: A finite-element formulation for the vertical discretization of sigma-coordinate primitive equation models. Mon. Wea. Rev., 105, 1108-1118.

Staniforth, A.N., and R.W. Daley, 1978: Formulation of the DRPN baroclinic finite-element primitive equations model. Notes Scientifiques et Techniques de RPN, Atmospheric Environment Service, Dorval, 66 pp.

Staniforth, A.N., and R.W. Daley, 1979: A baroclinic finite-element model for regional forecasting with the primitive equations. Mon. Wea. Rev., 107, 107-121.

Staniforth, A.N., and H. Mitchell, 1977: A semi-implicit finite-element barotropic model. Mon. Wea. Rev., 105, 154-169.

Staniforth, A.N., and H. Mitchell, 1978: A variable-resolution finite-element technique for regional forecasting with the primitive equations. Mon. Wea. Rev., 106, 439-447.

Stone, P.H., 1974: The meridional variation of the eddy heat fluxes by baroclinic waves and their parameterization. J. Atmos. Sci. 31, 444-456.

Strang, G., and G.J. Fix, 1973: An Analysis of the Finite Element Method. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Szmelter, J., 1959: The energy methods of networks of arbitrary shape in problems of the theory of elasticity. Proc. I.U.T.A.M. Symposium on Non-Homogeneity in Elasticity and Plasticity (ed. W. Olszak), Pergamom Press.

Teweles, S. (Jr.), and H.B. Wobus, 1954: Verification of prognostic charts. Bull. Amer. Met. Soc., 35, 455-463.

Tong, P., and J.N. Rossettos, 1977: Finite-Element Method, Basic Techniques and Implementation. MIT Press, Cambridge, Massachusetts, 332 pp.

Wang, H.H., P. Halpen, J. Douglas, and T. Dupont, 1972: Numerical solutions of the one-dimensional primitive equations using Galerkin approximations with localized basis functions. Mon. Wea. Rev., 100, 738-746.

Young, D.M., 1971: Iterative Solution of Large Linear Systems. Academic Press, New York, 570 pp.

Zienkiewicz, O.C., 1977: The Finite Element Method. McGraw-Hill Book Company (UK) Limited, Berkshire, England, 787 pp.

APPENDIX A

COMPUTER PROGRAM LISTINGS

This appendix contains a listing of the computer programs used and developed for this thesis. In addition, some plotting routines available on the computer were used. All calculations were done using double-precision variables.

```
C
C
C  FILE:     FEMAIN
C  FILES:
C     OBJECT  -  FEMAIN.O + INTERP.O + DIFF.O + PROJ.O + PROD.O
C     INPUT
C            2 = RELDAT
C            3 = FEMIN
C            4 = MESHD
C            5 = VMESH
C     OUTPUT
C            6 = *PRINT*
C            7 = GARBAGE
C            8 = FEDIAG
C
C   PROPER DIMENSIONS
C    X(NIU)    ,Y(NJU)    ,XI(NIP)    ,YI(NJP)
C    XCI(NI,NJ)    ,TAU(NI,NJ)    ,OMEG(NI,NJ)
C     WK1-WK7 DIM. MAX(NI,NJ)
C
C  THIS IS THE MAIN ROUTINE FOR THE FINITE ELEMENT MODEL.
C
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION X(281),  Y(45),  XI(63),  YI(27)
      DIMENSION XCI(63,27),  TAU(63,27),  OMEG(63,27)
      DIMENSION FX(63,27),FY(63,27),FXY(63,27)
      DIMENSION D(4,281,45),WORK(45,8)
      DIMENSION SPAR(6)
      DIMENSION SPARX(6)
      DIMENSION WKA(63,27)
      DIMENSION WKB(63,27)
      DIMENSION WKC(63,27)
      DIMENSION WKD(63,27)
      DIMENSION WKE(63,27)
      DIMENSION WKF(63,27)
      DIMENSION WKG(63,27)
      DIMENSION WKH(63,27)
      DIMENSION WKI(63,27)
      DIMENSION WKJ(63,27)
      DIMENSION WKK(63,27)
      DIMENSION WKL(63,27)
      DIMENSION OUT(63,27),XCI1(63,27),TAU1(63,27)
      DIMENSION XCT2(63,27),TAT2(63,27)
      REAL*8 L,LX,LY,L2,L3
      REAL*8 LYPR,LXPR
      COMMON / WKS1D1 / WK1(63)
      COMMON / WKS1D2 / WK2(63)
      COMMON / WKS1D3 / WK3(63)
      COMMON / WKS1D4 / WK4(63)
      COMMON / WKS1D5 / WK5(63)
      COMMON / WKS1D6 / WK6(63)
      COMMON / WKS1D7 / WK7(63)
      COMMON / WKS1D8 / WK8(63)
      COMMON / WKS1D9 / WK9(63)
      COMMON / WKS110 / WK10(63)
      COMMON / WKS1X1 / WK1X(63)
      COMMON / WKS1X2 / WK2X(63)
      COMMON / WKS1X3 / WK3X(63)
      COMMON / WKS1X4 / WK4X(63)
      COMMON / WKS1X5 / WK5X(63)
      COMMON / HXMESH / HX(63)
```

```
      COMMON / HYMESH / HY(27)
      COMMON / HXMES1 / HX1(63)
      COMMON / HYMES1 / HY1(27)
      COMMON / RATIO1 / RAT1(63,27)
      COMMON / RATII1 / RATI1(63,27)
      COMMON / RATIO4 / RAT4(63,27)
      COMMON / RATIO0 / RAT0(63,27)
      COMMON / RATIO2 / RAT2(63,27)
      COMMON / RATIO3 / RAT3(63,27)
      COMMON / RATIO5 / RAT5(63,27)
      COMMON / CONSTA / C4,C5
      LOGICAL FOURTH
      LOGICAL*1 LFMT(1)/'*'/
      DATA SPAR/.30D0,.50D0,.30D0,.15D0,.08D0,.02D0/
      DATA SPARX/.10D0,.15D0,.10D0,.05D0,.01D0,.005D0/
      ICNT=0
      CALL PLOTS
      CALL ORGEP(1.0,1.0,1.0)
C
C  SET CONSTANTS
C
      ALP=-12.0D0/5.0D0
      LX=2.8D07
      LY=8.8D06
      PI=3.141592654D0
      LYPR=2.0D0*PI/LY
      LXPR=2.0D0*PI/LX
      FO=1.0312D-4
      DP=5.0D04
      BETA=1.62D-11
      SIG=2.8D-6
      G=9.806D0
      L=LX/2.0D0/PI
      L2=L*L
      L3=1.0D0/L2
      ALPHA=LX/LY
      ALP2=1.0/ALPHA/ALPHA
      C4=L*L*FO*FO/G
      C5=DP*FO*10.
      BETAO=-BETA/FO
      SIGS=DP*DP/2.0D0/FO/FO/L/L*SIG
      SIGI=1.0D0/SIGS
      SIGJ=L2*SIGI
      OMCT=2.0D0*L*L*FO*FO/(SIG*DP*DP)
      OMCO=OMCT*L*L
      BEPR=2.0D0*L*L*FO*BETA/(SIG*DP*DP)
      HELMCF=2.0D0*FO*FO/(SIG*DP*DP)
C
C  SET FOURTH = .TRUE. FOR FOURTH ORDER SOLUTION AND
C      FOURTH = .FALSE. FOR SECOND ORDER SOLUTION.
C
      FOURTH =.TRUE.
C
C  READ NO. OF POINTS IN X AND Y DIRNS OF UNDERLYING GRID.
C
      READ(4,LFMT) NIU,NJU
C
C  READ NO. OF PTS IN X AND Y DIRNS. OF NON UNIFORM GRID
C
      READ(4,LFMT) NI,NJ
      NIP=NI
      NJP=NJ
C
C READ NO. OF PTS IN X AND Y DIRNS OF UNIFORM PART OF NON-UNIFORM GRID
C
      READ(4,LFMT) NILA,NJLA
```

```
C
C  READ X AND Y MESH LENGTHS OF UNDERLYING UDIFORM GRID
C
      READ (4,LFMT) HXU,HYU
C
C  READ X AND Y MESH LENGTHS OF UNIFORM PART OF NON-UNIFORM GRID
C
      READ(4,LFMT) HXN,HYN
C
C  READ X-POSN'S OF UNDERLYING GRID
C
      READ(5,LFMT)(X(I),I=1,NIU)
C
C  READ Y-POSN'S OF UNDERLYING GRID
C
      READ(5,LFMT)(Y(I),I=1,NJU)
C
C READ X-POSN'S OF NON UNIFORM GRID
C
      READ(5,LFMT)(XI(I),I=1,NIP)
C
C READ Y-POSN'S OF NON UNIFORM GRID
C
      READ(5,LFMT)(YI(I),I=1,NJP)
C
C  READ NO. OF WAVES, NO. OF HOURS FOR PROG., TIMESTEP, NO. OF
C  HOURS BETWEEN OUTPUT OF MAPS.
C
      READ(3,LFMT) N,M,DT,IDT2
C
C READ INITIAL VALUES OF HEIGHT, THICKNESS AND OMEGA FIELDS
C
      READ(3,LFMT) XCII1,XCI2,XCI3,TAUU1,TAU2,TAU3,OMEG1,OMEG2,OMEG3
C
C  READ INITIAL HEIGHT CHANGE GUESSES TO SAVE CPU TIME
C
      READ(3,LFMT)DXC1,DXC2,DXC3,DTA1,DTA2,DTA3
C
C  READ RELAXATION PARAMETERS.
C
      READ(2,LFMT) RELF,XTOL
      READ(2,LFMT) RELFH,XTOLH
      DT1=DT*.5ODO
      NIPM=NIP-1
      NJPM=NJP-1
      NIM=NI-1
      NJM=NJ-1
C
C  CALCULATE GRID LENGTHS OF NON UNIFORM GRID
C
      CALL CALH(HX1,XI,NIP)
      CALL CALH(HY1,YI,NJP)
      HX1(NIP)=HX1(NIPM)
      HY1(NJP)=HY1(NJPM)
      DO 60 I=1,NI
   60 HX(I)=HX1(I)
      DO 65 J=1,NJ
   65 HY(J)=HY1(J)
C
C  FORM RATIO MATRICES FOR SOLUTION
C
      HELP=5.ODO*HELMCF/12.ODO
      DO 848 J=1,NJ
      J1=J-1
      HY2=HY(J)
      IF(J.EQ.1) GO TO 849
```

```
      HYJ=HY2+HY(J1)
  849 DO 848 I=1,NI
      I1=I-1
      HX2=HX(I)
      RAT1(I,J)=HY2/HX2
      RATI1(I,J)=1.0D0/RAT1(I,J)
      IF(I.LT.2.OR.J.LT.2) GO TO 848
      HXI=HX2+HX(I1)
      RAT0(I,J)=HY2*HX2
      RAT2(I,J)=HY2*HXI
      RAT3(I,J)=HYJ*HX2
      RAT4(I,J)=RAT1(I,J)+RAT1(I1,J)+RAT1(I,J1)+RAT1(I1,J1)
      RAT4(I,J)=RAT4(I,J)+RATI1(I,J)+RATI1(I1,J)+RATI1(I,J1)
      RAT4(I,J)=RAT4(I,J)+RATI1(I1,J1)
      RAT5(I,J)=RAT4(I,J)+HELP*HXI*HYJ
      RAT5(I,J)=1.0D0/RAT5(I,J)
      RAT4(I,J)=1.0D0/RAT4(I,J)
  848 CONTINUE
C
C  CALCULATE INITIAL VALUES AT GRID POINTS
C
      XCII1=XCII1*DSQRT(2.0D0)
      XCI2=XCI2*2.0D0
      XCI3=XCI3*2.0D0
      DXC1=DXC1*DSQRT(2.0D0)
      DXC2=DXC2*2.0D0
      DXC3=DXC3*2.0D0
      TAUU1=TAUU1*DSQRT(2.0D0)
      TAU2=TAU2*2.0D0
      TAU3=TAU3*2.0D0
      DTA1=DTA1*DSQRT(2.0D0)
      DTA2=DTA2*2.0D0
      DTA3=DTA3*2.0D0
      OMEG1=OMEG1*DSQRT(2.0D0)
      OMEG2=OMEG2*2.0D0
      OMEG3=OMEG3*2.0D0
      CX=2.0D0*PI*DFLOAT(N)/LX
      CY=2.0D0*PI/LY
      SY=0.0D0
      DO 100 J=1,NJ
      J1=J-1
      IF(J.EQ.1) GO TO 110
      SY=SY+HY(J1)
  110 Y2=SY*CY
      SIY=DSIN(Y2)
      COY=DCOS(Y2)
      SX=0.0D0
      DO 100 I=1,NI
      I1=I-1
      IF(I.EQ.1) GO TO 120
      SX=SX+HX(I1)
  120 X2=SX*CX
      SIX=DSIN(X2)
      COX=DCOS(X2)
      S=SIY*SIX
      C=SIY*COX
      XCI(I,J)=XCII1*COY+XCI2*S+XCI3*C
      TAU(I,J)=TAUU1*COY+TAU2*S+TAU3*C
      OMEG(I,J)=OMEG1*COY+OMEG2*S+OMEG3*C
      XCI1(I,J)=DXC1*COY+DCX2*S+DXC3*C
      TAU1(I,J)=DTA1*COY+DTA2*S+DTA3*C
  100 CONTINUE
C
C  BEGINNING OF THE TIME LOOP
C
      DO 1000 II=1,M
```

```
      I2=(II-1)/IDT2*IDT2
      I3=I2-(II-1)
      IF(I3.EQ.0) ICNT=ICNT+1
      I4=(ICNT-1)/4*4-(ICNT-1)
C
C  SET ARRAYS TO 0.
C
C
C  FORM X-DERIV. OF XCI FIELD.
C
      CALL DXDYDS(WKI,XCI,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKA,WKI,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
C
C  FORM Y-DERIV. OF XCI FIELD.
C
      CALL DXDYDS(WKI,XCI,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKB,WKI,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
C
C  FORM X-DERIV. OF TAU FIELD.
C
      CALL DXDYDS(WKI,TAU,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKC,WKI,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
C
C  FORM Y-DERIV. OF TAU FIELD.
C
      CALL DXDYDS(WKI,TAU,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKD,WKI,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
C
C  ADD TERM 5 TO WKG, THE R.H.S. OF EQN. 1
C
      CALL GDADGD(WKG,0.0D0,WKG,BETA0,WKA,NI,NJ)
C
C  ADD TERM 6 TO WKH, THE R.H.S. OF EQN. 2
C
      CALL GDADGD(WKH,0.0D0,WKH,BETA0,WKC,NI,NJ)
      CALL GDADGD(WKL,0.0D0,WKL,[illegible]R,WKC,NI,NJ)
C
C  FORM LAPLACIAN OF XCI
C
      CALL D2XYS(WKI,XCI,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKE,WKI,NI,NJ,.TRUE.,.FALSE.,FOURTH)
      CALL D2XYS(WKI,XCI,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKF,WKI,NI,NJ,.FALSE.,.TRUE.,FOURTH)
      CALL GDADGD(WKE,1.0D0,WKE,1.0D0,WKF,NI,NJ)
      CALL SMY(WKE,NI,NJ,SPARY,6,WKK)
      CALL SMX(WKE,NI,NJ,SPARX,6,WKK)
C
C  FORM Y-DERIV. OF LAPLAC OF XCI.
C
      CALL DXDYDS(WKI,WKE,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKF,WKI,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
C
C  FORM TERM 1 OF EQN. 1
C
      CALL NLLOOP(WKI,WKA,WKF,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 1  (I.E. WKG)
C
      CALL GDADGD(WKG,1.0D0,WKG,-L2,WKJ,NI,NJ)
C
C  FORM TERM 1 OF EQN. 2
C
      CALL NLLOOP(WKI,WKC,WKF,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
```

```
C  ADD TO R.H.S. OF EQN. 2  (I.E. WKH)
C
      CALL GDADGD(WKH,1.0D0,WKH,-L2,WKJ,NI,NJ)
      CALL GDADGD(WKL,1.0D0,WKL,-OMCO,WKJ,NI,NJ)
C
C  FORM X-DERIV. OF LAPLACIAN OF XCI
C
      CALL DXDYDS(WKI,WKE,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKE,WKI,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
C
C  FORM TERM 2 OF EQN 1.
C
      CALL NLLOOP(WKF,WKB,WKE,NI,NJ)
      CALL PSOLVE(WKI,    ,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 1
C
      CALL GDADGD(WKG,1.0D0,WKG,L2,WKI,NI,NJ)
C
C  FORM TERM 2 OF EQN. 2.
C
      CALL NLLOOP(WKF,WKD,WKE,NI,N
      CALL PSOLVE(WKI,WKF,NI,NJ,.T            ,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 2.
C
      CALL GDADGD(WKH,1.0D0,WKH,L2,WKI,NI,NJ)
      CALL GDADGD(WKL,1.0D0,WKL,OMCO,WKI,NI,NJ)
C
C  FORM LAPLACIAN OF TAU
C
      CALL D2XYS(WKE,TAU,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKI,WKE,NI,NJ,.TRUE.,.FALSE.,FOURTH)
      CALL D2XYS(WKF,TAU,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKJ,WKF,NI,NJ,.FALSE.,.TRUE.,FOURTH)
      CALL GDADGD(WKE,1.0D0,WKI,1.0D0,WKJ,NI,NJ)
      CALL SMY(WKE,NI,NJ,SPAR,6,WKK)
      CALL SMX(WKE,NI,NJ,SPARX,6,WKK)
C
C  FORM Y-DERIV. OF LAPLACIAN OF TAU
C
      CALL DXDYDS(WKF,WKE,1.0D0,.FAL.   .TRUE.,NI,NJ)
      CALL PSOLVE(WKK,WKF,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
C
C  FORM TERM3 OF EQN. 1
C
      CALL NLLOOP(WKI,WKC,WKK,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 1
C
      CALL GDADGD(WKG,1.0D0,WKG,-L2,WKJ,NI,NJ)
C
C  FORM TERM 3 OF EQN. 2
C
      CALL NLLOOP(WKI,WKA,WKK,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRU      SE.)
C
C  ADD TO R.H.S. OF EQN. 2
C
      CALL GDADGD(WKH,1.0D0,WKH,-L2,WKJ,NI,NJ)
      CALL GDADGD(WKL,1.0D0,WKL,-OMCO,WKJ,NI,NJ)
C
C  FORM X-DERIV. OF LAPLACIAN OF TAU
C
      CALL DXDYDS(WKF,WKE,1.0D0,.TRUE.,.FALSE.,NI,NJ)
```

```
      CALL PSOLVE(WKJ,WKF,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
C
C  FORM TERM 4 OF EQN. 1
C
      CALL NLLOOP(WKI,WKD,WKJ,NI,NJ)
      CALL PSOLVE(WKK,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 1
C
      CALL GDADGD(WKG,1.ODO,WKG,L2,WKK,NI,NJ)
C
C  FORM TERM 4 OF EQN. 2
C
      CALL NLLOOP(WKI,WKB,WKJ,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  ADD TO R.H.S. OF EQN. 2
C
      CALL GDADGD(WKH,1.ODO,WKH,L2,WKJ,NI,NJ)
      CALL GDADGD(WKL,1.ODO,WKL,OMCO,WKJ,NI,NJ)
C
C  FIND VERTICAL VELOCITY
C
      CALL NLLOOP(WKI,WKA,WKD,NI,NJ)
      CALL PSOLVE(WKJ,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL NLLOOP(WKI,WKB,WKC,NI,NJ)
      CALL PSOLVE(WKK,WKI,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL GDADGD(WKF,1.ODO,WKJ,-1.ODO,WKK,NI,NJ)
      IF(II.EQ.1) GO TO 748
      IF(I3.NE.O.OR.I4.NE.O) GO TO 748
      CALL D2XYS(WKE,WKF,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKI,WKE,NI,NJ,.TRUE.,.FALSE.,FOURTH)
      CALL D2XYS(WKJ,WKF,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKK,WKJ,NI,NJ,.FALSE.,.TRUE.,FOURTH)
      CALL GDADGD(WKE,1.ODO,WKI,1.ODO,WKK,NI,NJ)
      CALL GDADGD(WKL,1.ODO,WKL,OMCO,WKE,NI,NJ)
      CALL PROJN(WKI,WKL,FOURTH,NI,NJ,.TRUE.,.TRUE.)
      DO 226 J=1,NJ
      DO 226 I=1,NI
  226 WKI(I,J)=ALP*WKI(I,J)
      CALL RELHEL(OMEG,WKI,NI,NJ,LX,HELMCF,RELFH,XTOLH)
  748 CONTINUE
C
C  ADD TERM 6 TO R.H.S. OF EQN. 2
C
      CALL GDADGD(WKH,1.ODO,WKH,OMCT,WKF,NI,NJ)
C
      WRITE(7,999) II
  999 FORMAT(I6)
      IF(I4.EQ.O.AND.I3.EQ.O) CALL OUTPUT(II,XCI,TAU,OMEG,XI,
     1 YI,X,Y,NI,NJ,NIU,NJU,FX,FY,FXY,D,WORK,NIP,NJP,OUT,DT)
      IF(I3.EQ.O) WRITE(8) XCI
      IF(I3.EQ.O) WRITE(8) TAU
      IF(II.EQ.M) GO TO 1000
C
C  THE R.H.S. OF EQN. 1 AND 2 HAVE NOW BEEN FORMED AND MAY PROCEED
C  WITH THE SOLUTION.
C
C
      CALL PROJN(WKI,WKG,FOURTH,NI,NJ,.TRUE.,.TRUE.)
      CALL PROJN(WKF,WKH,FOURTH,NI,NJ,.TRUE.,.TRUE.)
C
C  SOLVE EQN. 1
C
      D[illegible] J=1,NJ
      D[illegible] I=1,NI
```

```
      WKI(I,J)=ALP*WKI(I,J)
  225 WKF(I,J)=ALP*WKF(I,J)
      CALL REL(XCI1,WKI,NI,NJ,LX,RELF,XTOL)
C
C  SOLVE EQN  2
C
      CALL RELHEL(TAU1,WKF,NI,NJ,LX,HELMCF,RELFH,XTOLH)
      IF(II.EQ.1) GO TO 240
C
C  EXTRAPOLATE IN TIME USING ADAMS-BASHFORTH METHOD.
C
      DO 250 J=1,NJ
      DO 250 I=1,NI
      XCI(I,J)=XCI(I,J)+DT1*(3.ODO*XCI1(I,J)-XCT2(I,J))
      TAU(I,J)=TAU(I,J)+DT1*(3.ODO*TAU1(I,J)-TAT2(I,J))
  250 CONTINUE
      GO TO 260
C
C  EXTRAPOLATE FIRST TIME STEP USING FORWARD DIFFERENCE.
C
  240 DO 200 J=1,NJ
      DO 200 I=1,NI
      XCI(I,J)=XCI(I,J)+DT*XCI1(I,J)
      TAU(I,J)=TAU(I,J)+DT*TAU1(I,J)
  200 CONTINUE
  260 DO 220 J=1,NJ
      DO 220 I=1,NI
      XCT2(I,J)=XCI1(I,J)
      TAT2(I,J)=TAU1(I,J)
  220 CONTINUE
 1000 CONTINUE
      CALL PLOT(0.0,0.0,999)
      STOP
      END
      SUBROUTINE SMY(F,NI,NJ,SPAR,K,FSM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(NI,NJ),FSM(NI,NJ),SPAR(K)
      COMMON / HYMESH / HY(1)
C
C  SIMPLE 3 PT. SMOOTHER IN Y-DIRECTION
C
C  SPAR=SMOOTHING PARAMETERS FOR 6 GRID POINTS
C
      NJM=NJ-1
      S2=SPAR(1)
      S1=1.ODO-S2
      DO 10 I=1,NI
      FSM(I,1)=S1*F(I,1)+S2*F(I,2)
   10 FSM(I,NJ)=S1*F(I,NJ)+S2*F(I,NJM)
      DO 20 J=2,6
      S1=1.ODO-SPAR(J)
      J1=J+1
      J2=J-1
      DO 20 I=1,NI
   20 FSM(I,J)=S1*F(I,J)+SPAR(J)*(F(I,J1)+F(I,J2))
      DO 30 J=2,6
      JP=NJ-J+1
      S1=1.ODO-SPAR(J)
      J1=JP+1
      J2=JP-1
      DO 30 I=1,NI
   30 FSM(I,JP)=S1*F(I,JP)+SPAR(JP)*(F(I,J1)+F(I,J2))
      NJ5=NJ-5
      DO 60 J=1,6
      DO 60 I=1,NI
   60 F(I,J)=FSM(I,J)
```

```
      DO 70 J=NJ5,NJ
      DO 70 I=1,NI
   70 F(I,J)=FSM(I,J)
      RETURN
      END
      SUBROUTINE SMX(F,NI,NJ,SPAR,K,FSM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(NI,NJ),FSM(NI,NJ),SPAR(K)
      COMMON / HXMESH / HX(1)
C
C  SIMPLE 3-POINT SMOOTHER IN X-DIRECTION.
C
      NIM=NI-1
      DO 10 J=1,6
      S2=SPAR(J)
      S1=1.ODO-SPAR(J)
      FSM(NI,J)=S1*F(NI,J)+S2*F(NIM,J)
      FSM(1,J)=S1*F(1,J)+S2*F(2,J)
      JP=NJ-J+1
      FSM(1,JP)=S1*F(1,JP)+S2*F(2,JP)
   10 FSM(NI,JP)=S1*F(NI,JP)+S2*F(NIM,JP)
      DO 20 I=2,NIM
      I1=I-1
      I2=I+1
      DO 20 J=1,6
      S1=1.ODO-SPAR(J)
      FSM(I,J)=S1*F(I,J)+SPAR(J)*(F(I2,J)+F(I1,J))
      JP=NJ-J+1
      FSM(I,JP)=S1*F(I,JP)+SPAR(J)*(F(I2,JP)+F(I1,JP))
   20 CONTINUE
      NJ5=NJ-5
      DO 60 J=1,6
      DO 60 I=1,NI
   60 F(I,J)=FSM(I,J)
      DO 70 J=NJ5,NJ
      DO 70 I=1,NI
   70 F(I,J)=FSM(I,J)
      RETURN
      END
END OF FILE
```

```
C
C
C  FILE:  SPEC
C
C  FILES:
C     OBJECT  - SPEC.O
C             - SPLATM.O  (USES FOR OUTPUT)
C     INPUT
C              3 = VMESH
C              4 = MESHD
C              5 = SPECD   - INITIAL AMPLITUDES AND CONTROL DATA
C     OUTPUT
C              6 = *PRINT* - OUTPUT OF MAPS AND GRAPHS
C              7 = *PRINT* - OUTPUT OF AMPLITUDES
C              8 = *PRINT* - ENERGY AND ENSTROPHY
C             12 = FEMIN   - INITIAL DATA FOR FEM MODEL
C             10 = *PRINT* - THEORETICAL CALCULATIONS
C          11 = SPDIAG - BINARY OUTPUT FOR DIAGNOSTICS
C
C     SUBROUTINES:
C                - MAIN
C                - OUTPT
C
C  MAIN ROUTINE FOR THE SPECTRAL MODEL.
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(200,9),EKE(200),APE(200),TEN(200),PE(200)
      DIMENSION PHIH(200),PHIT(200),DPHI(200)
C
C  PROPER DIMENSIONS FOR ARRAYS:
C
C  DIM X(NIU),Y(NJU),XCIB(NIU,NJU),TAUB(NIU,NJU),OMEGB(NIU,NJU)
C  DIM XI(NI),YI(NI)
C
       DIMENSION X(281),Y(45),XCIB(281,45),TAUB(281,45),OMEGB(281,45)
      DIMENSION XI(63),YI(27),HX1(63),HY1(27)
      DIMENSION F(63,27),HX(63),HY(27)
      REAL*8 L,LX,LY,LXPR,LYPR
      COMMON TAU1(200),TAU2(200),TAU3(200),XCI1(200),XCI2(200)
      COMMON XCI3(200),OMEG1(200),OMEG2(200),OMEG3(200)
      COMMON C4,C5,DT
      LOGICAL*1 LFMT(1)/'*'/
C
C     READ INITIAL VALUES.
C
      READ(5,LFMT) XCI1(1),XCI2(1),XCI3(1),TAU1(1),TAU2(1),TAU3(1)
C
C     N=NUMBER OF WAVES, M=NUMBER OF TIME STEPS, DT=TIME STEP
C     IDT2 = NO. OF HOURS BETWEEN OUTPUT OF MAPS.
C
      READ(5,LFMT) N,M,DT,IDT2
      WRITE(7,22) N
   22 FORMAT('    NUMBER OF WAVES IS  ',I4)
C
C     SET VALUE OF CONSTANTS
C
      FN=DFLOAT(N)
      ICNT=0
      CALL PLOTS
      CALL ORGEP(1.0,1.0,1.0)
      PI=3.1415927D0
      FO=1.0312D-4
```

```
      BETA=1.62D-11
      DP=1.00D4
      SIG=2.8D-6
      LY=8.80D6
      LX=2.80D7
      L=LX/2.0D0/PI
      LXPR=2.0D0*PI*FN/LX
      LYPR=2.*PI/LY
      ALPHA=LX/LY
      A2=ALPHA*ALPHA
      BSTAR=L*BETA/FO
      SIGST=DP*DP*SIG/(2.0D0*FO*FO*L*L)
      G=9.806D0
      GAM=1.2004D0
      ITIME=0
      GAN=GAM*ALPHA*FN
      TO=A2+FN*FN
      T1=1.0D0/A2+SIGST
      T2=SIGST+1.0D0/TO
      T3=A2+1.0D0/SIGST
      T4=TO+1.0D0/SIGST
      A2=ALPHA*ALPHA
      C1=2.0D0*PI*PI*DP*FO*FO*L**4/(G*ALPHA*SIGST)
      C6=L**6*PI**4*DP*FO*FO*8.0D0/(G*ALPHA)
      C3=DP*2.0D0*PI*L*L/G/ALPHA
      C4=D*L*FO*FO/G
      C5=DP*FO*10.0D0
      M1=M+1
      Q2=2.0D0*FO*FO/(SIG*DP*DP)
      C7=4.0D0*DSQRT(2.0D0)*L*L*FO/LY
      C8=2.0D0*PI*FN/LX
      CR=BETA/C8/C8
      C9=2.0D0*(1      +Q2/C8/C8)
      WRITE(12,474    ,M,DT,IDT2
  474 FORMAT(2I6,4X,E12.6,I5)
C
C  GET PARAMETERS FOR F.E.M. MODEL TO MAKE OUPUT CONSISTENT
C
      READ(4,LFMT) NIU,NJU
      READ(4,LFMT) NI,NJ
      READ(4,LFMT) NILA,NJLA
      READ(4,LFMT) HXU,HYU
      READ(3,LFMT)(X(I),I=1,NIU)
      READ(3,LFMT)(Y(I),I=1,NJU)
      READ(3,LFMT)(XI(I),I=1,NI)
      READ(3,LFMT)(YI(I),I=1,NJ)
      CALL CALH(HX1,XI,NI)
      CALL CALH(HY1,YI,NJ)
      NIM=NI-1
      NJM=NJ-1
      HX1(NI)=HX1(NIM)
      HY1(NI)=HY1(NJM)
      NIP=NI
      NJP=NJ
      NIM=NI-1
      NJM=NJ-1
      DO 60 I=1,NIP
   60 HX(I)=HX1(I)
      DO 65 J=1,NJP
   65 HY(J)=HY1(J)
      WRITE(10,475) CR
  475 FORMAT('  ROSSBY PHASE SPEED =  ', E12.4,//)
      WRITE(10,470)
  470 FORMAT(4X,'ITIME',3X,'THERMAL WIND      MEAN WIND ',10X,
     1'REAL PHASE SPEEDS',6X,'IMAGINARY SPEED',4X,'E-TIME')
      WRITE(8,12)
```

```
   12  FORMAT(3X,'TIME',10X,'APE',13X,'KE',14X,'TEN',13X,'PE')
       WRITE(7,13)
   13  FORMAT('    TIME',6X,'XCI1',9X,'XCI2',9X,'XCI3',9X,
      1  'TAU1',9X,'TAU2',9X,'TAU3',9X,'OMEG1',8X,'OMEG2',8X,'OMEG3')
       DO 500 I=1,M
        ITIME=I-1
C
C      FIND THE VERTICAL VELOCITIES
C
       OMEG1(I)=-GAN/T1*(TAU2(I)*XCI3(I)-TAU3(I)*XCI2(I))
       OMEG2(I)=(((TAU1(I)*XCI3(I)+TAU3(I)*XCI1(I))*FN*FN/TO
      1  -(TAU3(I)*XCI1(I)-TAU1(I)*XCI3(I)))*GAN
      2  -BSTAR*FN*TAU3(I)/TO)/T2
       OMEG3(I)=(((TAU2(I)*XCI1(I)-TAU1(I)*XCI2(I))
      1  -(TAU1(I)*XCI2(I)+TAU2(I)*XCI1(I))*FN*FN/TO)*GAN
      2  +BSTAR*FN*TAU2(I)/TO)/T2
       IF(I.EQ.1)  WRITE(12,1000) XCI1(1),XCI2(1),XCI3(1),TAU1(1),
      1 TAU2(1),TAU3(1),OMEG1(1),OMEG2(1),OMEG3(1)
 1000  FORMAT(9(1X,E12.4))
       WRITE(7,10) ITIME,XCI1(I),XCI2(I),XCI3(I),TAU1(I),TAU2(I),
      1 TAU3(I),OMEG1(I),OMEG2(I),OMEG3(I)
   10  FORMAT(3X,I4,9(1X,E12.4))
C
C   CALCULATE PHASES
C
       PHIH(I)=DATAN2(XCI2(I),XCI3(I))*180.0D0/PI
       PHIT(I)=DATAN2(TAU2(I),TAU3(I))*180.0D0/PI
       DPHI(I)=PHIH(I)-PHIT(I)
C
C   CALCULATE THEORETICAL PARAMETERS
C
       UT=C7*TAU1(I)
       UST=C7*XCI1(I)
       D=(Q2*CR/C8/C8)**2+4.0D0*(1.0D0-(Q2/C8/C8)**2)*UT*UT
       CTEMP=-(2.0D0+Q2/C8/C8)*CR
       IF(D.GE.0.0D0) GO TO 400
C
C   IMAGINARY PHASE VELOCITY
C
       CREA2=0.0D00
       CIM=DSQRT(-D)/C9
       CREA1=CTEMP/C9
       ETAU=1.0D0/CIM/C8
       GO TO 450
  400  CREA1=(CTEMP+DSQRT(D))/C9
       CREA2=(CTEMP-DSQRT(D))/C9
       CIM=0.0D0
       ETAU=0.0D0
  450  WRITE(10,460) I,UT,UST,CREA1,CREA2,CIM,ETAU
  460  FORMAT(3X,I4,6(4X,E12.4))
C
C  CALCULATE ENERGIES.
C
       APE(I)=TAU1(I)*TAU1(I)+TAU2(I)*TAU2(I)+TAU3(I)*TAU3(I)
       APE(I)=APE(I)*C1
       EKE(I)=XCI2(I)*XCI2(I)+XCI3(I)*XCI3(I)+TAU2(I)*TAU2(I)
      1  +TAU3(I)*TAU3(I)
       EKE(I)=(EKE(I)*(1.0D0/LY/LY+FN*FN/LX/LX)+(XCI1(I)*XCI1(I)
      1  +TAU1(I)*TAU1(I))/LY/LY)
       EKE(I)=EKE(I)*C6
       TEN(I)=APE(I)+EKE(I)
C
C  CALCULATE POTENTIAL ENSTROPHY.
C
       Q1=(DABS(A2*XCI1(I)+T3*TAU1(I)))**2
      2 +(DABS(TO*XCI2(I)+T4*TAU2(I)))**2
```

```
      1 +(DABS(TO*XCI3(I)+T4*TAU3(I)))**2
       Q1=LX*LY*Q1/2.ODO
       Q3=(DABS(A2*XCI1(I)-T3*TAU1(I)))**2
      2 +(DABS(TO*XCI2(I)-T4*TAU2(I)))**2
      1 +(DABS(TO*XCI3(I)-T4*TAU3(I)))**2
       Q3=LX*LY*Q3/2.ODO
       PE(I)=(Q1+Q3)*C3
       WRITE(8,11)ITIME, APE(I),EKE(I),TEN(I),PE(I)
  11   FORMAT(3X,I4,4(4X,E12.4))
       I2=(I-1)/IDT2*IDT2
       I3=I2-(I-1)
       IF(I3.EQ.O) ICNT=ICNT+1
       I4=(ICNT-1)/4*4-(ICNT-1)
       IF(I4.EQ.O.AND.I3.EQ.O) CALL OUTPT(I,X,Y,NIU,NJU,LXPR,
      1 LYPR,HXU,HYU,XCIB,TAUB,OMEGB,DT)
       IF(I3.EQ.O) CALL FOUT(F,NI,NJ,HX,HY,XCI1(I),XCI2(I),XCI3(I),
      1 LXPR,LYPR)
       IF(I3.EQ.O) CALL FOUT(F,NI,NJ,HX,HY,TAU1(I),TAU2(I),TAU3(I),
      1 LXPR,LYPR)
C
C      BEGIN INTEGRATION
C
       TIME=I*DT
       IF(ITIME.NE.1) GO TO 100
C
C      FIRST STEP:  USE EULER
       TAU1(2)=TAU1(1)-OMEG1(1)*DT/ALPHA/ALPHA
       T1BAR=(TAU1(1)+TAU1(2))/2.ODO
       TAU2(2)=TAU2(1)+((T1BAR*XCI3(1)+TAU3(1)*XCI1(1))*FN*FN*GAN/TO
      1   -BSTAR*FN*TAU3(1)/TO-OMEG2(1)/TO)*DT
       T2BAR=(TAU2(1)+TAU2(2))/2.ODO
       TAU3(2)=TAU3(1)+(-(T1BAR*XCI2(1)+T2BAR*XCI1(1))*FN*FN*GAN/TO
      1   +BSTAR*FN*T2BAR/TO-OMEG3(1)/TO)*DT
       XCI1(2)=XCI1(1)
       T3BAR=(TAU3(1)+TAU3(2))/2.ODO
       XCI2(2)=XCI2(1)+((XCI1(1)*XCI3(1)+T1BAR*T3BAR)
      1   *GAN*FN*FN/TO-BSTAR*FN*XCI3(1)/TO)*DT
       X2BAR=(XCI2(1)+XCI2(2))/2.ODO
       XCI3(2)=XCI3(1)+(-(XCI1(1)*X2BAR+T1BAR*T2BAR)
      1   *GAN*FN*FN/TO+BSTAR*FN*X2BAR/TO)*DT
       GO TO 500
 100   CONTINUE
       I1=I+1
       I2=I-1
C
C      USE ADAMS-BASHFORTH METHOD FOR SUBSEQUENT TIME STEPS
C
       XCI1(I1)=XCI1(I)
       TAU1(I1)=TAU1(I)-(3.ODO*OMEG1(I)/ALPHA/ALPHA/2.ODO
      1   -OMEG1(I2)/ALPHA/ALPHA/2.)*DT
       C=(TAU1(I)*XCI3(I)+TAU3(I)*XCI1(I))*GAN*FN*FN/TO
      1   -BSTAR*FN*TAU3(I)/TO-OMEG2(I)/TO
       C2=(TAU1(I2)*XCI3(I2)+TAU3(I2)*XCI1(I2))*GAN*FN*FN/TO
      1   -BSTAR*FN*TAU3(I2)/TO-OMEG2(I2)/TO
       TAU2(I1)=TAU2(I)+(3.ODO*C/2.ODO-C2/2.ODO)*DT
       C=-(TAU1(I)*XCI2(I)+TAU2(I)*XCI1(I))*GAN*FN*FN/TO
      1   +BSTAR*FN*TAU2(I)/TO-OMEG3(I)/TO
       C2=-(TAU1(I2)*XCI2(I2)+TAU2(I2)*XCI1(I2))*GAN*FN*FN/TO
      1   +BSTAR*FN*TAU2(I2)/TO-OMEG3(I2)/TO
       TAU3(I1)=TAU3(I)+(3.ODO*C/2.ODO-C2/2.ODO)*DT
       C=XCI1(I)*XCI3(I)*GAN*FN*FN/TO-BSTAR*FN*XCI3(I)/TO
       C2=XCI1(I2)*XCI3(I2)*GAN*FN*FN/TO-BSTAR*FN*XCI3(I2)/TO
       T=(TAU1(I)*TAU3(I)+TAU1(I1)*TAU3(I1))*GAN*FN*FN/TO/2.
       XCI2(I1)=XCI2(I)+(3.ODO*C/2.ODO-C2/2.ODO+T)*DT
C      FROM EQN. 12
       C=XCI1(I)*XCI2(I)+XCI1(I1)*XCI2(I1)
```

```
     1   +TAU1(I)*TAU2(I)+TAU1(I1)*TAU2(I1)
      C2=XCI2(I)+XCI2(I1)
      XCI3(I1)=XCI3(I)-C*GAN*FN*FN/TO*DT/2.0D0+
     1 C2*BSTAR*FN/TO*DT/2.0D0
  500 CONTINUE
      DXC1=(XCI1(2)-XCI1(1))/DT
      DXC2=(XCI2(2)-XCI2(1))/DT
      DXC3=(XCI3(2)-XCI3(1))/DT
      DTA1=(TAU1(2)-TAU1(1))/DT
      DTA2=(TAU2(2)-TAU2(1))/DT
      DTA3=(TAU3(2)-TAU3(1))/DT
      WRITE(12,1000) DXC1,DXC2,DXC3,DTA1,DTA2,DTA3
C
C  PLOT OUTPUT PARAMETERS
C
      DO 600 I=1,M
      A(I,1)=XCI1(I)
      A(I,2)=XCI2(I)
      A(I,3)=XCI3(I)
      A(I,4)=TAU1(I)
      A(I,5)=TAU2(I)
      A(I,6)=TAU3(I)
      A(I,7)=OMEG1(I)
      A(I,8)=OMEG2(I)
      A(I,9)=OMEG3(I)
  600 CONTINUE
      CALL SPLATM(A,200,9,9,M1,2,-.5D-1,.5D-1)
C
C  PLOT ENERGYS
C
      DO 620 I=1,M
      A(I,1)=EKE(I)
      A(I,2)=APE(I)
      A(I,3)=TEN(I)
  620 CONTINUE
      CALL SPLATM(A,200,9,3,M1,2,0.0D0,0.0D0)
C
C  PLOT PHASES
C
      DO 640 I=1,M1
      A(I,1)=PHIH(I)
      A(I,2)=PHIT(I)
      A(I,3)=DPHI(I)
  640 CONTINUE
      CALL SPLATM(A,200,9,3,M1,2,-180.0D0,180.0D0)
      CALL PLOT(0.0,0.0,999)
      STOP
      END
      SUBROUTINE OUTPT(I,X,Y,NIU,NJU,LXPR,LYPR,HXU,HYU,XCIB,TAUB,
     1 OMEGB,DT)
C
C  SET UP FIELDS TO BE PLOTTED IN MANNER CONSISTENT WITH FEM
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XCIB(NIU,NJU),TAUB(NIU,NJU),OMEGB(NIU,NJU)
      DIMENSION X(NIU),Y(NJU)
      REAL*8 LXPR,LYPR
      COMMON TAU1(200),TAU2(200),TAU3(200),XCI1(200),XCI2(200)
      COMMON XCI3(200),OMEG1(200),OMEG2(200),OMEG3(200)
      COMMON C4,C5
      RT2=DSQRT(2.0D0)
      DO 20 KI=1,NIU
      XX=LXPR*(X(KI)-HXU)
      CX=DCOS(XX)
      SX=DSIN(XX)
      DO 20,KJ=1,NJU
```

```
      YY=LYPR*(Y(KJ)-HYU)
      CY=DCOS(YY)
      SY=DSIN(YY)
      S=SX*SY
      C=CX*SY
      XCIB(KI,KJ)=RT2*CY*XCI1(I)+2.ODO*S*XCI2(I)+2.ODO*C*XCI3(I)
      TAUB(KI,KJ)=RT2*CY*TAU1(I)+2.ODO*S*TAU2(I)+2.ODO*C*TAU3(I)
      OMEGB(KI,KJ)=RT2*CY*OMEG1(I)+2.ODO*S*OMEG2(I)+2.ODO*C*OMEG3(I)
      XCIB(KI,KJ)=C4*XCIB(KI,KJ)
      TAUB(KI,KJ)=C4*TAUB(KI,KJ)
      OMEGB(KI,KJ)=C5*OMEGB(KI,KJ)
   20 CONTINUE
      CALL FPLOT(XCIB,NIU,NJU,1)
      CALL FPLOT(TAUB,NIU,NJU,1)
      CALL FPLOT(OMEGB,NIU,NJU,O)
      RETURN
      END
      SUBROUTINE FOUT(F,NI,NJ,HX,HY,F1,F2,F3,LXPR,LYPR)
      IMPLICIT REAL*8 (A-H,O-Z)
C
C  OUTPUT FIELDS FOR DIAGNOSTIC PROGRAM
C
      DIMENSION F(NI,NJ),HX(NI),HY(NJ)
      REAL*8 LXPR,LYPR
      F11=F1*DSQRT(2.ODO)
      F22=F2*2.ODO
      F33=F3*2.ODO
      SY=O.ODO
      DO 100 J=1,NJ
      J1=J-1
      IF(J1.EQ.O) GO TO 110
      SY=SY+HY(J1)
  110 Y2=SY*LYPR
      SIY=DSIN(Y2)
      COY=DCOS(Y2)
      SX=O.ODO
      DO 100 I=1,NI
      I1=I-1
      IF(I1.EQ.O) GO TO 120
      SX=SX+HX(I1)
  120 X2=SX*LXPR
      SIX=DSIN(X2)
      COX=DCOS(X2)
      S=SIY*SIX
      C=SIY*COX
      F(I,J)=F11*COY+F22*S+F33*C
  100 CONTINUE
      WRITE(11) F
      RETURN
      END
END OF FILE
```

```
C      FILE:  DIFF
C      OBJECT FILE:  DIFF.O
C      SUBROUTINES:
C              - DXDYDS
C              - CALH
C              - SETD2
C              - D2XYS
C
C
C
       SUBROUTINE DXDYDS(R,U,CON,ALONGX,ALONGY,NI,NJ)
C
C   S/R : DXDYDS - CALCULATES A DIFFERENCE IN X AND Y DIRECTIONS
C
C   AUTHOR : ANDREW STANIFORTH
C
C   REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C   ARGUMENTS :
C      OUT  - R          - RESULT
C      IN   - U          - FIELD TO BE DIFFERENCED
C           - CON        - MULTIPLICATIVE CONSTANT (MULTIPLIES DERIVATIVE)
C           - ALONGX     - IF .TRUE. DIFFERENCE IN X DIRECTION
C           - ALONGY     - IF .TRUE. DIFFERENCE IN Y DIRECTION
C           - NI         - X DIMENSION
C           - NJ         - Y DIMENSION
C
C
C
       IMPLICIT REAL*8 (A-H,O-Z)
       DIMENSION R(NI,NJ),U(NI,NJ)
       LOGICAL ALONGX,ALONGY
       CONX=.50D0*CON
       NIM=NI-1
       NJM=NJ-1
       IF(ALONGX) GO TO 100
       IF(ALONGY) GO TO 200
       RETURN
C
C    X - DERIVATIVE
C
   100 CONTINUE
       IT=NI-2
       DO 150 J=1,NJ
       DO 140 I=1,IT
       I1=I+1
       I2=I+2
   140 R(I1,J)=CONX*(U(I2,J)-U(I,J))
       R(1,J)=CONX*(U(2,J)-U(1,J))
       R(NI,J)=CONX*(U(NI,J)-U(NIM,J))
   150 CONTINUE
       RETURN
C
C   Y - DERIVATIVE
C
   200 CONTINUE
       IT=NJ-2
       DO 250 I=1,NI
       DO 240 J=1,IT
       J1=J+1
       J2=J+2
   240 R(I,J1)=CONX*(U(I,J2)-U(I,J))
       R(I,1)=CONX*(U(I,2)-U(I,1))
```

```
      R(I,NJ)=CONX*(U(I,NJ)-U(I,NJM))
  250 CONTINUE
      RETURN
      END
      SUBROUTINE CALH(H,X,M)
C
C  S/R: CALH - CALCULATION OF GRID LENGTHS FROM POINTS X.
C                H(I)=X(I+1)-X(I), I=1,M-1
C
C  AUTHOR:  A. STANIFORTH  -  1973
C
C  REVISION 001: A. STANIFORTH - C. THIBEAULT  JAN 79 DOCUMENTATION
C  REVISION 002: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  CALL:   CALL CALH(H,X,M)
C
C  ARGUMENTS:
C    OUT    - H - GRID-LENGHTS
C    IN     - X - ARRAY OF POINTS IN ASCENDING ORDER
C           - M - LENGHT OF ARRAYS H AND X
C
C  NOTES:  - H(M) IS NOT CALCULATED
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION X(M),H(M)
      DO 10 I=2,M
      I1=I-1
      H(I1)=X(I)-X(I1)
   10 CONTINUE
      RETURN
      END
      SUBROUTINE SETD2(A,B,C,CON,H,N)
C
C  S/R : SETD2  - SET-UP ELEMENTS OF TRI-DIAGONAL MATRIX USED IN S/R
C                 D2XYS FOR TAKING SECOND DERIVATIVES.
C
C  AUTHOR - ANDREW STANIFORTH - JAN 79
C
C  REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  ARGUMENTS:
C    OUT    - A    - LOWER-DIAGONAL ELEMENTS
C           - B    - DIAGONAL ELEMENTS
C           - C    - UPPER-DIAGONAL ELEMENTS
C           - CON  - MULTIPLICATIVE FACTOR (CON*SECOND DERIVATIVE)
C           - H    - MESH-LENGTHS
C           - N    - NUMBER OF NODAL POINTS
C
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(N),B(N),C(N),H(N)
      HR=CON/H(1)
      B(1)=0.0D0
      NM=N-1
      DO 100 I=1,NM
      I1=I+1
      HR=CON/H(I)
      A(I1)=HR
      B(I)=B(I)-HR
      B(I1)=-HR
  100 C(I)=HR
      RETURN
      END
      SUBROUTINE D2XYS(R,U,CON,ALONGX,ALONGY,NI,NJ)
```

```
C
C  S/R: D2XYS - CALCULATES 2ND DERIVATIVE IN X- OR Y-
C               DIRECTION DEPENDING UPON WHETHER ALONGX OR
C               ALONGY IS TRUE.
C
C  AUTHOR:  A. STANIFORTH - OCTOBER 1977
C
C  REVISION 001:  A. STANIFORTH - C. THIBEAULT JAN 79 DOCUMENTATION
C  REVISION 002:  T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  CALL:  CALL D2XYS(R,U,CON,ALONGX,ALONGY,NI,NJ)
C
C  ARGUMENTS:
C   OUT   - R       - RESULT
C   IN    - U       - INPUT FIELD TO BE TWICE DIFFERENCIATED
C         - CON     - CONSTANT
C         - ALONGX  - IF TRUE DIFFERENTIATE IN X-DIRECTION
C         - ALONGY  - IF TRUE DIFFERENTIATE IN Y-DIRECTION
C         - NI      - X-DIMENSION
C         - NJ      - Y-DIMENSION
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION R(NI,NJ),U(NI,NJ)
      COMMON / WKS1D4 / WK4(1)
      COMMON / WKS1D5 / A(1)
      COMMON / WKS1D6 / B(1)
      COMMON / WKS1D7 / C(1)
      COMMON / HXMESH / HX(1)
      COMMON / HYMESH / HY(1)
      LOGICAL ALONGX,ALONGY
C
C
C
      NIM=NI-1
      NJM=NJ-1
      IF(ALONGX) GO TO 100
      IF(ALONGY) GO TO 200
      RETURN
C
C  X-DIRECTION
C
  100 CONTINUE
      CALL SETD2(A,B,C,CON,HX,NI)
      DO 130 J=1,NJ
      WK4(NI)=B(NI)*U(NI,J)
      DO 120 I=1,NIM
      I1=I+1
  120 WK4(I)=C(I)*U(I1,J)+B(I)*U(I,J)
      DO 125 I=1,NIM
      I1=I+1
  125 R(I1,J)=A(I1)*U(I,J)+WK4(I1)
      R(1,J)=WK4(1)
  130 CONTINUE
      RETURN
C
C  Y-DIRECTION
C
  200 CONTINUE
      CALL SETD2(A,B,C,CON,HY,NJ)
      DO 230 I=1,NI
      WK4(NJ)=B(NJ)*U(I,NJ)
      DO 220 J=1,NJM
      J1=J+1
  220 WK4(J)=C(J)*U(I,J1)+B(J)*U(I,J)
      DO 225 J=1,NJM
```

```
      J1=J+1
  225 R(I,J1)=A(J1)*U(I,J)+WK4(J1)
      R(I,1)=WK4(1)
  230 CONTINUE
      RETURN
      END
      SUBROUTINE GDADGD(R,CONG,G,CONH,H,NI,NJ)
C
C  S/R:  GDADGD -  R(I,J)=CONG*G(I,J)+CONH*H(I,J)
C
C  AUTHOR: A. STANIFORTH DEC 78
C
C  REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  ARGUMENTS:
C   OUT   - R     - RESULTING GRID
C   IN    - CONG  - MULTIPLICATIVE CONSTANT FOR FIRST INPUT GRID
C         - G     - FIRST INPUT GRID
C         - CONH  - MULTIPLICATIVE CONSTANT FOR SECOND INPUT GRID
C         - H     - SECOND INPUT GRID
C         - NI    - DIMENSION OF X-DIRN
C         - NJ    - DIMENSION OF Y-DIRN
C
C  - R AND G MAY SHARE THE SAME SPACE IF G NOT REQUIRED SUBSEQUENTLY
C  - R AND H MAY SHARE THE SAME SPACE IF H NOT REQUIRED SUBSEQUENTLY
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION R(NI,NJ),G(NI,NJ),H(NI,NJ)
      DO 20 J=1,NJ
      DO 20 I=1,NI
      R(I,J)=CONG*G(I,J)+CONH*H(I,J)
   20 CONTINUE
      RETURN
      END
END OF FILE
```

```
C
C
C   FILE   INTERP
C   OBJECT FILE   INTERP O
C   SUBROUTINES
C         - INTRPT
C         - DINT
C         - ROSSR3
C         - SPD
C           FD1
C         - FDM
C
C
C
      SUBROUTINE INTRPT(FI,IFI,JFI,F,IF,JF,XI,YI,X,Y,FX,FY,FXY,
     1                  HX,HY,KL,P,S,C,A,D,WORK,KDER)
C
C ORIGINALLY WRITTEN AT DRPN, DORVAL P.Q.
C
C REVISION 001   T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C INTRPT DOES 2-DIMENSIONAL SPLINE INTERPOLATION.  THE INPUT F(IF,JF) IS
C DEFINED ON THE PTS ((X(I),Y(J),I=1,IF),J=1,JF)  THE OUTPUT
C FI(4,IFI,JFI) IS CALCULATED AT PTS ((XI(I),YI(J),I=1,IFI),J=1,JFI)
C USING BI-CUBIC SPLINES IN TERMS OF THE PARTIAL DERIVATIVES OF F,FX,FY,
C FXY.  THESE P.D.S ARE CALCULATED BY SPD.  (IF KL.EQ.1, FXY=DX(DY(F)),
C WHILE IF KL.NE.1, FXY-DY(DX(F))).
C FD1 AND FDM RETURN END PT DERIVATIVES USED BY SPD.
C THE VECTORS XI,YI,X,Y,(HX(I)=X(I+1)-X(I),I=1,IF-1),
C (HY(J)=Y(J+1)-Y(J),J=1,JF-1), ARE ADDITIONAL INPUT.
C THE ACTUAL INTERPOLATION IS DONE BY THE ROUTINE DINT, AFTER THE
C EVALUATION OF FX,FY,FXY.
C IF KDER=FALSE, THEN ONLY INTERPOLATED VALUES OF THE FUNCTION ARE
C RETURNED.
C IF KDER=TRUE, THE VALUES OF THE X,Y AND XY DERIVATIVES AT THE
C INTERPOLATED PTS WILL ALSO BE RETURNED.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FI(4,IFI,JFI)
      DIMENSION F (IF,JF)
      DIMENSION FX(IF,JF)
      DIMENSION FY(IF,JF)
      DIMENSION FXY(IF,JF)
C
C THE PROPER DIMENSIONS HERE ARE IFI,JFI,IF,JF,IF,JF
C
      DIMENSION XI(IFI)
      DIMENSION YI(JFI)
      DIMENSION X(IF)
      DIMENSION Y(JF)
      DIMENSION HX(IF)
      DIMENSION HY(JF)
C
C WORKING STORAGE FOR SPD,ROSSR3 (LENGTH MAXO(IF,JF) IS SUFFICIENT)
C
      DIMENSION P(IF)
      DIMENSION S(IF)
      DIMENSION A(IF)
      DIMENSION C(IF)
      DIMENSION D(IF)
C
C WORKING STORAGE FOR DINT
C
```

```
      DIMENSION WORK(JFI,8)
      LOGICAL KOER
C
C CHECK TO ENSURE EXTRAPOLATION IS NOT BEING ATTEMPTED A MACHINE
C DEPENDENT INCREMENT IS ADDED IN TEST TO ACCOUNT FOR MACHINE ROUND-OFF
C
      X1=XI(1)+DABS(XI(1)*1.D-12      1)
      IF(X1.LT.O.) WRITE(6,900      1),
      IF(X1.LT.O.) STOP
      X1=YI(1)+DABS(YI(1)*1.D
      IF(X1.LT.O.) WRITE(6,90
      IF(X1.LT.O.) STOP
      X1= X(1)+DABS( X(1)*1.D-
      X1=X(IF)+DABS(X(IF)*1.D-12)
      IF(X1.LT.O.) WRITE(6,902) IF,X(IF),IFI,XI(IFI)
      IF(X1.LT.O.) STOP
      X1=Y(JF)+DABS(Y(JF)*1.D-12)-YI(JFI)
      IF(X1.LT.O.) WRITE(6,903) JF,Y(JF),JFI,YI(JFI)
      IF(X1.LT.O.) STOP
  900 FORMAT(1H1,' EXTRAPOLATION ATTEMPTED, DETECTED BY INTRPT  XI(1)=',
     1E16.8,'LT X(1)=',E16.8)
  901 FORMAT(1H1,' EXTRAPOLATION ATTEMPTED, DETECTED BY INTRPT  YI(1)=',
     1E16.8,'LT Y(1)=',E16.8)
  902 FORMAT(1H1,' EXTRAPOLATION ATTEMPTED, DETECTED BY INTRPT  X(',I5
     1')=',E16.8,'LT XI(',I5,')=',E16.8)
  903 FORMAT(1H1,' EXTRAPOLATION ATTEMPTED, DETECTED BY INTRPT  Y(',I
     1')= ,E16.8,'LT YI(',I5,')=',E16.8)
      CMU1=0.0D0
      CLMDAM = 0.0D0
      IF(KL.NE.1) GO TO 150
   10 DO 100 I=1,IF
      DO 50 J=1,JF
      S(J)=F(I,J)
   50 CONTINUE
      C  =FD1(S,HY,JF)
      CM = FDM(S,HY,JF)
      CALL SPD (P,S,JF HY,CMU1,C1,CLMDAM,CM,A,C,D)
      DO 60 J=1,JF
      FY(I,J)=P( )
   60 CONTINUE
      IF(KL.EQ.1) GO TO 100
      DO 70 J=1,JF
      S(J)=FX(I,J)
   70 CONTINUE
      C1=FD1(S,HY,JF)
      CM=FDM(S,HY,JF)
      CALL SPD(P,S,JF,HY,CMU1,C1,CLMDAM,CM,A,C,D)
      DO 75 J=1,JF
      FXY(I,J)=P(J)
   75 CONTINUE
  100 CONTINUE
      IF(KL.NE.1) GO TO 600
  150 DO 500 J=1,JF
      DO 200 I=1,IF
      S(I)=F(I,J)
  200 CONTINUE
      C1=FD1(S,HX,IF)
      CM=FDM(S,HX,IF)
      CALL SPD(P,S,IF,HX,CMU1,C1,CLMDAM,CM,A,C,D)
      DO 210 I=1,IF
      FX(I,J)=P(I)
  210 CONTINUE
      IF(KL.NE.1) GO TO 500
      DO 260 I=1,IF
      S(I)=FY(I,J)
  260 CONTINUE
```

```
      C1=FD1(S,HX,IF)
      CM=FDM(S,HX,IF)
      CALL SPD(P,S,IF,HX,CMU1,C1,CLMDAM,CM,A,C,D)
      DO 280 I=1,IF
      FXY(I,J)=P(I)
  280 CONTINUE
  500 CONTINUE
      IF(KL.NE.1) GO TO 10
  600 CALL DINT(FI,IFI,JFI,F,IF,JF,FX,FY,FXY,XI,YI,X,Y,HX,HY,
     1    WORK(1,1),WORK(1,2),WORK(1,3),WORK(1,4),WORK(1,5),WORK(1,6),
     2    WORK(1,7),WORK(1,8),KDER)
      RETURN
      END
      SUBROUTINE DINT(FI,IFI,JFI,F,IF,JF,FX,FY,FXY,XI,YI,X,Y,HX,HY,
     1    ZA,ZB,ZC,ZD,ZAY,ZBY,ZCY,ZDY,KDER)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FI(4,IFI,JFI)
      DIMENSION F(IF,JF)
      DIMENSION FX(IF,JF)
      DIMENSION FY(IF,JF)
      DIMENSION FXY(IF,JF)
      DIMENSION XI(IFI)
      DIMENSION YI(JFI)
      DIMENSION X(IF)
      DIMENSION Y(JF)
      DIMENSION HX(IF)
      DIMENSION HY(JF)
      DIMENSION ZA(JFI),ZB(JFI),ZC(JFI),ZD(JFI)
      DIMENSION ZAY(JFI),ZBY(JFI),ZCY(JFI),ZDY(JFI)
      LOGICAL KDER
C
C ORIGINALLY WRITTEN AT DRPN, DORVAL P.Q.
C REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL    U OF A
C
C THE DISCRETE FN F(I,J) IS ASSUMED KNOWN AT THE PTS WHOSE COORDS ARE
C (X(I),Y(J)), WHERE I=1,...IF   ,J=1,...JF.
C IT IS ASSUMED THAT X(I).LT.X(I+1) AND Y(J).LT.Y(J+1).
C HX(I) MUST BE PREVIOUSLY DEFINED AS HX(I)=X(I+1)-X(I) FOR I=1,...(IF-1).
C SIM. FOR HY(J).
C IF KDER=TRUE  THEN
C THIS SUBROUTINE RETURNS IN FI(K,I,J) THE INTERPOLATED VALUES OF THE
C FUNCTION AND ITS X,Y,XY DERIVATIVES AT PTS (XI(I),YI(J)),
C WHERE I=1,.....IFI   ,J=1,...JFI.
C IT IS ASSUMED THAT XI(I).LT.XI(I+1) AND YI(J).LT.YI(J+1).
C HERE K=1 REFERS TO THE FUNCTION,
C      K=2 REFERS TO THE X-DER OF THE FUNCTION,
C      K=3 REFERS TO THE Y-DER OF THE FUNCTION,
C      K=4 REFERS TO THE XY-DER OF THE FUNCTION.
C NOTE THAT X(1).LE.XI(1).LE.XI(IFI).LE.XI(IF).
C SIM. FOR Y AND YI.
C IF KDER=FALSE THEN ONLY THE INTERPOLATED VALUES OF THE FUNCTION ARE
C RETURNED. (I.E. CALCULATIONS ARE PERFORMED FOR K=1 ONLY).
C THE APPROXIMATION USED IS A BICUBIC SPLINE OF INTERPOLATION IN
C TERMS OF THE PARTIAL DERIVATIVES FX(I,J),FY(I,J) AND FXY(I,J)
C WHERE I=1,...IF   ,J=1,...JF.
C THESE DERIVATIVES MUST BE CALCULATED OUTSIDE OF THE ROUTINE.
C E.G. BY USING SUBROUTINE SPD IN A SUITABLE MANNER - SUBROUTINE INTD
C DOES THIS AND CALLS THIS SUBROUTINE. (INTD IS CALLABLE FROM INTRPT).
C WORKING STORAGE ARRAYS ZA,ZB,ZC,ZD HAVE DIMENSION JFI.
C
C
     LL=2
     DO 15 J=1,JFI
     DO 5 L=LL,JF
     IF(YI(J).LE.Y(L)) GO TO 8
   5 CONTINUE
```

```
      L=JF
    8 L1=L-1
      LL=L
      WN=YI(J)-Y(L1)
      WD=1.0D0/HY(L1)
      WE=WN*WD
      WE1=1.0D0-WE
      WE2=WE1*WE1
      WW=2.0D0*WE
      ZA(J)=WE2*WN
      ZB(J)=WE1*WN*WE
      ZC(J)=WE2*(1.0D0+WW)
      ZD(J)=WE*WE*(3.0D0-WW)
      ZAY(J)=-WW*WE1+WE2
      ZBY(J)=-WE*WE+WW*WE1
      ZCY(J)=-6.0D0*WD*WE1*WE
      ZDY(J)=-ZCY(J)
   15 CONTINUE
      KK=2
      DO 500 I=1,IFI
      DO 40 K=KK,IF
      IF(XI(I).LE.X(K)) GO TO 45
   40 CONTINUE
      K=IF
   45 K1=K-1
      KK=K
      WM=XI(I)-X(K1)
      WDD=1.0D0/HX(K1)
      WZ=WM*WDD
      WZ1=1.0D0-WZ
      WZ2=WZ1*WZ1
      ZZ=2.0D0*WZ
      ZE=WZ2*WM
      ZF=WZ1*WM*WZ
      ZG=WZ2*(1.0D0+ZZ)
      ZL=WZ*WZ*(3.0D0-ZZ)
      IF(.NOT.KDER) GO TO 46
      ZEX=-ZZ*WZ1+WZ2
      ZFX=-WZ*WZ+ZZ*WZ1
      ZGX=-6.0D0*WDD*WZ1*WZ
      ZLX=-ZGX
   46 LL=2
      DO 400 J=1,JFI
      DO 420 L=LL,JF
      IF(YI(J).LE.Y(L)) GO TO 450
  420 CONTINUE
      L=JF
  450 L1=L-1
      LL=L
      Z1=ZE*FXY(K1,L1)-ZF*FXY(K,L1)+ZG*FY(K1,L1)+ZL*FY(K,L1)
      Z2=ZE*FXY(K1,L)-ZF*FXY(K,L)+ZG*FY(K1,L)+ZL*FY(K,L)
      Z3=ZE*FX(K1,L1)-ZF*FX(K,L1)+ZG*F(K1,L1)+ZL*F(K,L1)
      Z4=ZE*FX(K1,L)-ZF*FX(K,L)+ZG*F(K1,L)+ZL*F(K,L)
      FI(1,I,J)=ZA(J)*Z1-ZB(J)*Z2+ZC(J)*Z3+ZD(J)*Z4
      IF(.NOT.KDER) GO TO 400
      Z1X=ZEX*FXY(K1,L1)-ZFX*FXY(K,L1)+ZGX*FY(K1,L1)+ZLX*FY(K,L1)
      Z2X=ZEX*FXY(K1,L)-ZFX*FXY(K,L)+ZGX*FY(K1,L)+ZLX*FY(K,L)
      Z3X=ZEX*FX(K1,L1)-ZFX*FX(K,L1)+ZGX*F(K1,L1)+ZLX*F(K,L1)
      Z4X=ZEX*FX(K1,L)-ZFX*FX(K,L)+ZGX*F(K1,L)+ZLX*F(K,L)
      FI(2,I,J)=ZA(J)*Z1X-ZB(J)*Z2X+ZC(J)*Z3X+ZD(J)*Z4X
      FI(3,I,J)=ZAY(J)*Z1-ZBY(J)*Z2+ZCY(J)*Z3+ZDY(J)*Z4
      FI(4,I,J)=ZAY(J)*Z1X-ZBY(J)*Z2X+ZCY(J)*Z3X+ZDY(J)*Z4X
  400 CONTINUE
  500 CONTINUE
      RETURN
      END
```

```
      FUNCTION FDM(F,H,M)
C
C  FUNCTION  FDM - THIS FUNCTION RETURNS IN FDM THE DERIVATIVE
C                  OF F AT THE PT X(M) IN TERMS OF
C                  F(M-3),F(M-2),F(M-1),F(M),H(M-3),H(M-2),H(M-1)
C                  WHERE H(I)=X(I+1)-X(I) FOR I=M-3,M-2,M-1
C
C  AUTHOR:  A. STANIFORTH  -  1973
C
C  REVISION 001: A  STANIFORTH - C. THIBEAULT  JAN 79 DOCUMENTATION
C  REVISION 002: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C FUNCTION :   CALL FDM(F,H,M)
C
C  ARGUMENTS:
C     IN     - F - FIELD OF VALUES
C            - H - MESH-SPACING
C            - M - NO  OF POINTS
C
C  NOTES: - THE APPROXIMATION USED IS THE DIFFERENTIATED FORM OF
C           LAGRANGES CUBIC INTERPOLATION FORMULA FOR NON-UNIFORM
C           GRIDS.
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(M),H(M)
      H1=H(M-1)
      H2=H(M-2)
      H3=H(M-3)
      X1=H1+H2
      X2=H2+H3
      X3=X1+H3
      FDM=-X1*H1*F(M-3)/(H3*X2*X3)+X3*H1*F(M-2)/(H3*H2*X1)
      FDM=FDM-X3*X1*F(M-1)/(X2*H2*H1)+(1.0D0/X3+1.0D0/X1+1.0D0/H1)*F(M)
      RETURN
      END
      FUNCTION FD1(F,H,M)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(M),H(M)
C
C ORIGINALLY WRITTEN AT DRPN, DORVAL P.Q.
C REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C
C  LET F(1),F(2),...... BE THE VALUES OF A FUNCTION F DEFINED AT
C  SUCCESSIVE PTS X(1),X(2),..... OF A NON UNIFORM GRID.
C  THIS FUNCTION ROUTINE RETURNS IN FD1 THE DERIVATIVE OF F AT THE PT
C  X(1) IN TERMS OF F(1),F(2),F(3),F(4),H(1),H(2),H(3).
C  THE APPROXIMATION USED IS THE DIFFERENTIATED FORM OF LAGRANGES CUBIC
C  INTERPOLATION FORMULA FOR NON UNIFORM GRIDS.
C
C
      H1=H(1)
      H2=H(2)
      H3=H(3)
      X1=H1+H2
      X2=H2+H3
      X3=X1+H3
      FD1=-(1.0D0/H1+1.0D0/X1+1.0D0/X3)*F(1)+X1*X3*F(2)/(H1*H2*X2)
      FD1=FD1-H1*X3*F(3)/(X1*H2*H3)+H1*X1*F(4)/(X3*X2*H3)
      RETURN
      END
      SUBROUTINE ROSSR3(P,A,DELTA,C,D,M)
C
C ORIGINALLY WRITTEN AT DRPN, DORVAL P.Q.
C REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
```

```
C
C   VERSION 3
C   ANY TRI-DIAGONAL MATRIX SAY Q IN EQUATION QP=D MAY BE
C   NORMALIZED WITH 1'S ON DIAGONAL AND C(I), I=1,M-1 FOR UPPER DIAGONAL
C   AND A(I),I=2,M FOR LOWER DIAGONAL.
C
C   DELTA  IS A WORKING STORAGE ARRAY OF DIMENSION M
C   IF THE VECTOR D IS NOT REQUIRED SUBSEQUENTLY THEN THE CALL STATEMENT
C   CALL ROSSR3(P,A,D,C,D,M)
C   WILL USE THE ARRAY D AS WORKING STORAGE AND REDUCE THE OVERALL STORAGE
C   REQUIRED.
C   IF THE ARRAY C IS NOT REQUIRED SUBSEQUENTLY THEN THE CALL STATEMENT
C   CALL ROSSR3(C,A,DELTA,C,D,M)
C   WILL REDUCE CORE STORAGE REQUIREMENTS.
C   IF BOTH C AND D ARE NOT REQUIRED SUBSEQUENTLY THEN THE CALL STATEMENT
C   CALL ROSSR3(C,A,D,C,D,M)
C   WILL FURTHER REDUCE THE CORE REQUIREMENTS.
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(M),A(M),DELTA(M),C(M),D(M)
      C(M)=0.0D0
      P(1)=-C(1)
      DELTA(1)=D(1)
      DO 1 I=2,M
      I1=I-1
      AI=A(I)
      X=1.0D0/(1.0D0+AI*P(I1))
      P(I)=-C(I)*X
    1 DELTA(I)=(D(I)-AI*DELTA(I1))*X
      P(M)=DELTA(M)
      DO 2 I=2,M
      II=M-I+1
      II1=II+1
    2 P(II)=P(II)*P(II1)+DELTA(II)
      RETURN
      END
      SUBROUTINE SPD(P,S,M,H,CMU1,C1,CLMDAM,CM,A,C,D)
C
C
C   THIS SUBROUTINE RETURNS IN P THE DERIVATIVES OF THE FUNCTION S DEFINED
C   AT M CONSECUTIVE DISCRETE DATA POINTS X(1),X(2),.....X(M).
C   THESE DATA POINTS DEFINE A NON-UNIFORM GRID H(I)=X(I+1)-X(I) WHERE I
C   RUNS FROM 1 TO (M-1)
C   IT USES SUBROUTINE ROSSR3.
C   THE APPROXIMATION USED IS A CUBIC SPLINE FIT TO ALL THE DATA POINTS.
C   IT REQUIRES BOUNDARY CONDITIONS TO BE SPECIFIED OF THE FORM
C     P(1)+CMU1*P(2)=C1
C     CLMDAM*P(M-1)+P(M)=CM
C   FOR A NATURAL SPLINE (THIS IS LEAST RESTRICTIVE CONDITION AND IMPLIES
C   ZERO CURVATURE AT THE END POINTS), SDD(1)=SDD(M)=0, AND WE HAVE
C   CMU1=CLMDAM=0.5, C1=1.5*(S(2)-S(1))/H(1), CM=1.5*(S(M)-S(M-1))/H(M-1)
C
C   FOR SPECIFIED SLOPES (I.E. SD(1) AND SD(M) GIVEN) WE HAVE
C   CMU1=CLMDAM=0.   C1=SD(1)    CM=SD(M)
C
C   FOR SPECIFIED SECOND DERIVATIVES (I.E. SDD(1) AND SDD(M) GIVEN) WE HAVE
C   CMU1=CLMDAM=0.5,  C1=1.5*(S(2)-S(1))/H(1)-H(1)*SDD(1)*0.25
C                     CM=1.5*(S(M)-S(M-1))/H(M-1)+H(M-1)*SDD(M)*0.25
C
C   NOTE THAT IF THE GRID SIZES H(I) ARE FIXED THEN THE CALCULATION OF A
C   AND C MAY BE MADE OUTSIDE THE SUBROUTINE AND PASSED VIA A COMMON BLOCK.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(M),S(M),A(M),C(M),D(M),H(M)
      M1=M-1
      DO 1 I=2,M1
      I1=I-1
```

```
      HI=H(I)
      HI1=H(I1)
      X1=0.5DO/(HI+HI1)
      A(I)=HI*X1
    1 C(I)=HI1*X1
      DO 2 I=2,M1
      I1=I-1
      I2=I+1
      HI=H(I)
      HI1=H(I1)
      SI=S(I)
      X1=H(I)/H(I-1)
    2 D(I)=1.5DO*((SI-S(I1))*X1+(S(I2)-SI)/X1)/(HI+HI1)
      C(1)=CMU1
      A(M)=CLMDAM
      D(1)=C1
      D(M)=CM
      CALL ROSSR3(P,A,D,C,D,M)
      RETURN
      END
END OF FILE
```

```
C         FILE:  PROD
C         OBJECT FILE.  PROD.O
C         SUBROUTINES:
C                 NLINX
C                 NLLOOP
C
C
C
          SUBROUTINE NLINX(R,U,V,H6,NI)
C
C  ADAPTED FROM DRPN DORVAL P.Q.
C  REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
          IMPLICIT REAL*8 (A-H,O-Z)
          DIMENSION R(NI),U(NI),V(NI)
C
C  H6(I)=H(I)/6. MUST BE PRESET BEFORE ROUTINE
C
          DIMENSION H6(NI)
          NIM=NI-1
          WK50=U(1)*V(1)
          R(1)=0.0D0
          DO 300 I=1,NIM
          I1=I+1
          H6I=H6(I)
          WK4=(U(I)+U(I1))*(V(I)+V(I1))*.5D0
          R(I)=R(I)+(WK50+WK4)*H6I
          WK50=U(I1)*V(I1)
      300 R(I1)=(WK50+WK4)*H6I
          RETURN
          END
          SUBROUTINE NLLOOP(R,U,V,NI,NJ)
C
C  S/R : NLLOOP - COMPUTE THE HORIZONTAL PRODUCT R=U*V
C
C  AUTHOR : A. STANIFORTH - REVISED DEC 78 TO REMOVE BANK CONFLICTS
C  REVISION 001: T. GOOS 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  ARGUMENTS:
C    OUT   - R   - RESULTING PRODUCT
C    IN    - U   - INPUT FIELD
C          - V   - INPUT FIELD
C          - NI  - X DIMENSION
C          - NJ  - Y DIMENSION
C
C
          IMPLICIT REAL*8 (A-H,O-Z)
          DIMENSION R(NI,NJ),U(NI,NJ),V(NI,NJ)
          COMMON / HXMESH / HX(1)
          COMMON / HYMESH / HY(1)
          COMMON / WKS1D1 / HX6(1)
          COMMON / WKS1D2 / UO(1)
          COMMON / WKS1D3 / VO(1)
          COMMON / WKS1D6 / RO(1)
          COMMON / WKS1D7 / WK7(1)
          COMMON / WKS1D8 / WK8(1)
          COMMON / WKS1D9 / WK9(1)
          COMMON / WKS110 / HOLD(1)
          TWELFT=1.0D0/12.0D0
          SIXTH=1.0D0/6.0D0
          NIM=NI-1
          NJM=NJ-1
C
```

```
C  COMPUTE H6(I)=H(I)/6
C  AND SET RO TO O
C
      DO 50 I=1,NIM
      UO(I)=U(I,1)
      VO(I)=V(I,1)
      RO(I)=0.0D0
   50 HX6(I)=HX(I)/6.0D0
      RO(NI)=0.0D0
      UO(NI)=U(NI,1)
      VO(NI)= V(NI,1)
      J=1
C
C  COMPUTE ROW PRODUCT U*V ON FIRST ROW
C
      CALL NLINX(WK7,UO,VO,HX6,NI)
C
C  LOOP OVER ROWS
C
      DO 100 J=1,NJM
      J1=J+1
      HYJ1=HY(J)*SIXTH
      HYJ2=HY(J)*TWELFT
      DO 75 I=1,NI
C
C  ADD FIRST CONTRIBUTION TO J TH ROW
C
      HOLD(I)=HYJ1*WK7(I)+RO(I)
C
C  COMPUTE 2.*U AND 2.*V ON COLLOCATION ROW
C
      WK8(I)=UO(I)+U(I,J1)
   75 WK9(I)=VO(I)+V(I,J1)
C
C  COMPUTE ROW PRODUCT HY(J)*U*V/3. ON COLLOCATION ROW
C
      CALL NLINX(WK7,WK8,WK9,HX6,NI)
      DO 90 I=1,NI
      WK8(I)=HYJ2*WK7(I)
C
C  ADD OTHER CONTRIBUTION TO J TH ROW
C
      RO(I)=HOLD(I)+WK8(I)
C
C  MOVE RESULT RO INTO R(1,J)
C
      R(I,J)=RO(I)
C
C  ADD FIRST CONTRIBUTION TO (J+1)ST ROW
C
   90 HOLD(I)=WK8(I)
C
C COMPUTE ROW PRODUCT U*V ON (J+1)ST ROW
C
      CALL NLINX(WK7,U(1,J1),V(1,J1),HX6,NI)
C
C  ADD OTHER CONTRIBUTION TO (J+1)ST LEVEL
C
      DO 120 I=1,NI
      RO(I)=HYJ1*WK7(I)+HOLD(I)
C
C  MOVE U(I,J+1),V(I,J+1) INTO UO,VO
C
      UO(I)=U(I,J1)
  120 VO(I)=V(I,J1)
  100 CONTINUE
```

```
C
C  MOVE RESULT RO INTO LAST ROW OF R
C
      DO 150 I=1,NI
  150 R(I,NJ)=RO(I)
      RETURN
      END
END OF FILE
```

```
C
C
C  FILE:  PROJ
C  OBJECT FILE:  PROJ.O
C  SUBROUTINES:
C      - PROJN
C      - SETABC
C      - SETTRI
C      - SOLTRI
C      - PSOLVE
C
C
      SUBROUTINE PROJN(R,RHS,FOURTH,NI,NJ,ALONGX,ALONGY)
C
C  S/R: PROJN - THE FINITE-ELEMENT PROJECTION OPERATOR (ALONG ANY
C               OR ALL DIRECTIONS, AS DECIDED BY THE LOGICAL VARIABLES
C               ALONGX,ALONGY)
C
C               R = PROJECTION (RHS)
C
C  AUTHOR   A. STANIFORTH - SPRING 1977
C
C  REVISION 001: A. STANIFORTH - JAN 1979  VECTORIZED
C  REVISION 002: A. STANIFORTH - C. THIBEAULT  FEB 79 DOCUMENTATION
C  REVISION 003: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  CALL:  CALL PROJN(R,RHS,FOURTH,NI,NJ,ALONGX,ALONGY)
C
C  ARGUMENTS:
C     OUT   - R      - RESULT OF PROJECTION
C     IN    - RHS    - RIGHT-HAND-SIDE TO BE PROJECTED
C           - FOURTH - LOGICAL SWITCH - TRUE  = FOURTH ORDER PROJECTION
C                                               IN HORIZONTAL
C                                     - FALSE = SECOND ORDER PROJECTION
C                                               IN HORIZONTAL
C           - NI     - X-DIMENSION
C           - NJ     - Y-DIMENSION
C           - ALONGX - IF .TRUE. PROJECT IN X DIRECTION
            - ALONGY - IF .TRUE. PROJECT IN Y DIRECTION
C
C
C  NOTES:   - AT PRESENT THE CODE HANDLES SECOND AND FOURTH ORDER IN
C             BOTH DIMENSIONS.
C           - IF ALONGX AND ALONGY ARE ALL .FALSE., A COPY OF RHS IS
C             RETURNED IN R.
C
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION R(NI,NJ), RHS(NI,NJ)
      COMMON / HXMESH / HX(1)
      COMMON / HYMESH / HY(1)
      COMMON / WKS1D5 / A(1)
      COMMON / WKS1D6 / B(1)
      COMMON / WKS1D7 / C(1)
      LOGICAL ALONGX,ALONGY
      LOGICAL FOURTH
C
C
C  SET UP WEIGHTS FOR PROJECTION OPERATOR
C  WT = 2. CORRESPONDS TO USUAL PROJECTION, WHILST
C  WT = 5. CORRESPONDS TO PROJECTION USED IN THE FOURTH-ORDER SOLUTION
C          .OF ELLIPTIC BOUNDARY-VALUE PROBLEMS.
```

```
C
      WTX=2.0D0
      WTY=2.0D0
      IF(FOURTH) WTX=5.0D0
      IF(FOURTH) WTY=5.0D0
C
C  CODE FOR PROJECTION ALONG X DIRECTION
C
      IF(.NOT.ALONGX) GO TO 200
      CALL SETABC(A,B,C,HX,WTX,NI)
      NIM=NI-1
      DO 130 J=1,NJ
      DO 150 I=1,NIM
      I1=I+1
      I2=I+2
      R(I1,J)=A(I1)*RHS(I,J)+B(I1)*RHS(I1,J)
      IF(I.LT.NIM) R(I1,J)=R(I1,J)+C(I1)*RHS(I2,J)
  150 CONTINUE
      R(1,J)=C(1)*RHS(2,J)+B(1)*RHS(1,J)
  130 CONTINUE
      DO 135 J=1,NJ
      DO 135 I=1,NI
      RHS(I,J)=R(I,J)
  135 CONTINUE
C
  200 CONTINUE
C
C  CODE FOR PROJECTION ALONG Y DIRECTION
C
      IF(.NOT.ALONGY) RETURN
      CALL SETABC(A,B,C,HY,WTY,NJ)
      NJM=NJ-1
      DO 230 I=1,NI
      DO 250 J=1,NJM
      J1=J+1
      J2=J+2
      R(I,J1)=A(J1)*RHS(I,J)+B(J1)*RHS(I,J1)
      IF(J.LT.NJM) R(I,J1)=R(I,J1)+C(J1)*RHS(I,J2)
  250 CONTINUE
      R(I,1)=C(1)*RHS(I,2)+B(1)*RHS(I,1)
  230 CONTINUE
      RETURN
      END
      SUBROUTINE SETABC(A,B,C,H,WT,N)
C
C  S/R: SETABC - SETS THE TRI-DIAGONAL ELEMENTS OF THE MATRIX
C                ASSOCIATED WITH THE PROJECTION OPERATOR.
C
C  AUTHOR: A. STANIFORTH - SPRING 1977
C
C  REVISION 001: A. STANIFORTH - C. THIBEAULT FEB 79 DOCUMENTATION
C  REVISION 002: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  CALL:  CALL SETABC(A,B,C,H,WT,N)
C
C  ARGUMENTS:
C      OUT      - A  - LOWER-DIAGONAL ELEMENTS
C               - B  - DIAGONAL ELEMENTS
C               - C  - UPPER-DIAGONAL ELEMENTS
C       IN      - H  - MESH-SPACING
C               - WT - USUALLY WT=2., BUT WT=5. WHEN SOLVING EBV PROBLEMS
C                      TO FOURTH ORDER
C               - N  - NO. OF POINTS
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
```

```
      DIMENSION A(N).B(N).C(N)
      DIMENSION H(N)
      NM=N-1
      C1=0.5D0/(1.0D0+WT)
      C2=WT*C1
      A(1)=0.0D0
      B(1)=H(1)*C2
      C(1)=H(1)*C1
      DO 10 I=2,NM
      HI1=H(I-1)
      HI=H(I)
      A(I)=HI1*C1
      B(I)=(HI1+HI)*C2
      C(I)=HI*C1
   10 CONTINUE
      A(N)=H(NM)*C1
      B(N)=H(NM)*C2
      C(N)=0.0D0
      RETURN
      END
      SUBROUTINE SETTRI(BIGE,BIGC,BIGA,A,B,C,N)
C
C  S/R: SETTRI - SETTRI DOES THE PREPROCESSING PASS TO SOLVE A
C                TRI-DIAGONAL SYSTEM OF LINEAR EQUATIONS.
C
C  AUTHOR: D. ROBERTSON - MARCH 1977
C
C  REVISION 001: A. STANIFORTH - C. THIBEAULT FEB 79 DOCUMENTATION
C  REVISION 002: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C  CALL:  CALL  SETTRI(BIGE,BIGC,BIGA,A,B,C,N)
C
C  ALGORITHM:
C             - SOLVES        M*P=D
C
C               WHERE M IN THE N BY N TRI-DIAGONAL MATRIX
C
C               ****                                 ****
C               *B(1),C(1),0,0....                      *
C               *A(2),B(2),C(2),0,0....                 *
C               *                                       *
C               *                                       *
C               *                                       *
C               *  ... 0,A(I),B(I),C(I),0...            *
C               *                                       *
C               *                                       *
C               *                      ...0,0,A(N),B(N)*
C               ****                                 ****
C
C             - A(1) AND C(N) ARE NOT DEFINED BY THE MATRIX, BUT ARRAYS A,C
C               SHOULD BE OF FULL SIZE, N.  FOR CONVENIENCE THIS ROUTINE
C               ZEROES C(N).  THE METHOD IS GAUSSIAN ELIMINATION WITHOUT
C               PIVOTING, FOLLOWED BY BACK SUBSTITUTION.
C
C ARGUMENTS
C   OUT      - BIGE  - PROCESSED ARRAY TO BE USED IN S/R SOLTRI
C            - BIGC  - PROCESSED ARRAY TO BE USED IN S/R SOLTRI
C            - BIGA  - PROCESSED ARRAY TO BE USED IN S/R SOLTRI
C   IN       - A     - LOWER-DIAGONAL ELEMENTS OF TRI-DIAGONAL MATRIX
C            - B     - DIAGONAL ELEMENTS OF TRI-DIAGONAL MATRIX
C   IN       - C     - UPPER-DIAGONAL ELEMENTS OF TRI-DIAGONAL MATRIX
C            - N     - NO. OF POINTS
C
C NOTES:  - SETTRI IS CALLED TO COMPUTE BIGE,BIGC,BIGA FROM A,B,C
C           AFTER THAT A CALL TO SOLTRI WILL COMPLETE THE SOLUTION FOR
C           A GIVEN RHS D.  THIS METHOD IS EFFICIENT WHEN SOLVING WITH
```

```
C            SEVERAL DIFFERENT RHS BUT THE SAME MATRIX M, AS ALL
C            DIVISIONS ARE DONE ONLY ONCE, AND SOLTRI PERFORMS N
C            VECTORIZABLE MULTIPLICATIONS FOLLOWED BY TWO SIMILAR
C            RECURSIVELY DEFINED, NON-VECTORIZABLE , LOOPS, EACH HAVING
C            N MULTIPLIES AND N SUBTRACTIONS, PER RHS.  EXAMINATION OF
C            SOLTRI SHOWS BIGA(1), BIGC(N) ARE NOT USED.
C
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BIGE(N),BIGC(N),BIGA(N),A(N),B(N),C(N)
      BIGE(1)=1.0D0/B(1)
      BIGC(1)=C(1)*BIGE(1)
      C(N)=0.0D0
      DO 50 I=2,N
      I1=I-1
      AI=A(I)
      BIGE(I)=1.0/(B(I)-AI*BIGC(I1))
      BIGC(I)=BIGE(I)*C(I)
      BIGA(I)=AI*BIGE(I)
   50 CONTINUE
      RETURN
      END
      SUBROUTINE SOLTRI(N,P,D,BIGD,BIGE,BIGC,BIGA)
C
C S/R:  SOLTRI - USED AFTER SETTRI TO COMPLETE THE SOLUTION TO THE
C                TRI-DIAGONAL MATRIX PROBLEM.  SEE SETTRI FOR DETAILS.
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(N),D(N),BIGD(N),BIGE(N),BIGC(N),BIGA(N)
      DO 20 I=1,N
      BIGD(I)=D(I)*BIGE(I)
   20 CONTINUE
      DO 50 I=2,N
      I1=I-1
      BIGD(I)=BIGD(I)-BIGA(I)*BIGD(I1)
   50 CONTINUE
      P(N)=BIGD(N)
      DO 100 I=2,N
      IREV=N-I+1
      IREV1=IREV+1
      P(IREV)=BIGD(IREV)-BIGC(IREV)*P(IREV1)
  100 CONTINUE
      RETURN
      END
      SUBROUTINE PSOLVE(R,RHS,NI,NJ,ALONGX,ALONGY,FOURTH)

C S/R: PSOLVE : SOLUTION OF THE MATRIX PROBLEMS ASSOCIATED WITH THE
C               PROJECTION OPERATOR (ALONG ANY OR ALL DIRECTIONS, AS
C               DECIDED BY THE LOGICAL VARIABLES ALONGX,ALONGY)
C
C                      R = INVERSE PROJECTION ( RHS )
C
C AUTHOR: A. STANIFORTH - REVISED DEC 78 TO EXECUTE FASTER
C REVISION 001: T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A
C
C ARGUMENTS:
C   OUT    - R      - RESULT
C   IN     - RHS    - RIGHT-HAND SIDE
C          - NI     - X DIMENSION
C          - NJ     - Y DIMENSION
C          - ALONGX - IF .TRUE. , INVERSE PROJECT IN X DIRECTION
C          - ALONGY - IF .TRUE. , INVERSE PROJECT IN Y DIRECTION
C
C
```

```
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION R(NI,NJ),RHS(NI,NJ)
      COMMON / HXMESH / HX(1)
      COMMON / HYMESH / HY(1)
      COMMON / WKS1D1 / A(1)
      COMMON / WKS1D3 / B(1)
      COMMON / WKS1D4 / C(1)
      COMMON / WKS1D5 / CF1(1)
      COMMON / WKS1D6 / CF2(1)
      COMMON / WKS1D7 / CF3(1)
      COMMON / WKS1D9 / TEMP2(1)
      LOGICAL ALONGX,ALONGY,FOURTH
C
C  SET UP WTS ASSOCIATED WITH PROJN OPERATOR.
C  WT=2. CORRESPONDS TO NORMAL USAGE.
C
      WTX=2.0D0
      WTY=2.0D0
      IF(FOURTH) WTX=5.0D0
      IF(FOURTH) WTY=5.0D0
C
C  CODE FOR SOLVING ALONG X DIRECTION
C
      IF(.NOT.ALONGX) GO TO 200
      CALL SETABC(A,B,C,HX,WTX,NI)
      CALL SETTRI(CF1,CF2,CF3,A,B,C,NI)
      DO 130 J=1,NJ
      CALL SOLTRI(NI,R(1,J),RHS(1,J),TEMP2,CF1,CF2,CF3)
  130 CONTINUE
      GO TO 250
  200 CONTINUE
      DO 450 J=1,NJ
      DO 450 I=1,NI
  450 R(I,J)=RHS(I,J)
C
C  CODE FOR SOLVING ALONG Y DIRECTION
C
  250 IF(.NOT.ALONGY) RETURN
      CALL SETABC(A,B,C,HY,WTY,NJ)
      CALL SETTRI(CF1,CF2,CF3,A,B,C,NJ)
      DO 460 I=1,NI
      R(I,1)=CF1(1)*R(I,1)
  460 CONTINUE
      DO 215 J=2,NJ
      J1=J-1
      CF3J=CF3(J)
      CF1J=CF1(J)
      DO 215 I=1,NI
      R(I,J)=-CF3J*R(I,J1)+CF1J*R(I,J)
  215 CONTINUE
      DO 225 J=2,NJ
      JREV=NJ-J+1
      JREV1=JREV+1
      CF2J=CF2(JREV)
      DO 225 I=1,NI
      R(I,JREV)=-CF2J*R(I,JREV1)+R(I,JREV)
  225 CONTINUE
      RETURN
      END
END OF FILE
```

```
      SUBROUTINE OUTPUT(II,XCI,TAU,OMEG,XI,YI,X,Y,NI,NJ,NIU,NJU,
     1 FX,FY,FXY,D,WORK,NIP,NJP,OUT,DT)
C
C  ROUTINE TO PRODUCE MAPS OF FORECAST FIELDS
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XCI(NI,NJ),TAU(NI,NJ),OMEG(NI,NJ)
      DIMENSION XI(NIP),YI(NJP),X(NIU),Y(NJU)
      DIMENSION D(4,NIU,NJU)
      DIMENSION FX(NIP,NJP),FY(NIP,NJP),FXY(NIP,NJP),WORK(NJU,8)
      DIMENSION OUT(NIP,NJP)
      DIMENSION E(281,45)
      COMMON / HXMES1 / HX1(1)
      COMMON / HYMES1 / HY1(1)
      COMMON / WKS1X1 / WK1(1)
      COMMON / WKS1X2 / WK2(1)
      COMMON / WKS1X3 / WK3(1)
      COMMON / WKS1X4 / WK4(1)
      COMMON / WKS1X5 / WK5(1)
      COMMON / CONSTA / C4,C5
      LOGICAL KDER
C
C  USES ROUTINE INTRPT TO PRODUCE VALUES OF FIELDS ON UNIFORM GRID
C
      KDER=.FALSE.
      KL=2
      ITIME=DFLOAT(II-1)*DT/.3700D+.5D0
      CALL INTRPT(D,NIU,NJU,XCI,NIP,NJP,X,Y,XI,YI,FX,
     1  FY,FXY,HX1,HY1,KL,WK1,WK2,WK3,WK4,WK5,WORK,KDER)
      DO 100 J=1,NJU
      DO 100 I=1,NIU
  100 E(I,J)=C4*D(1,I,J)
      CALL FPLOT(E,NIU,NJU,1)
      CALL INTRPT(D,NIU,NJU,TAU,NIP,NJP,X,Y,XI,YI,FX,
     1 FY,FXY,HX1,HY1,KL,WK1,WK2,WK3,WK4,WK5,WORK,KDER)
      DO 300 J=1,NJU
      DO 300 I=1,NIU
  300 E(I,J)=C4*D(1,I,J)
      CALL FPLOT(E,NIU,NJU,1)
      CALL INTRPT(D,NIU,NJU,OMEG,NIP,NJP,X,Y,XI,YI,FX,
     1 FY,FXY,HX1,HY1,KL,WK1,WK2,WK3,WK4,WK5,WORK,KDER)
      DO 475 J=1,NJU
      DO 475 I=1,NIU
  475 E(I,J)=C5*D(1,I,J)
      CALL FPLOT(E,NIU,NJU,0)
      RETURN
      END
      SUBROUTINE MSHFLT(FIELD,WKX,WKY,CON,NI,NJ,IPOW)
C
C  S/R: MSHFLT - REDUCES AMPLITUDE OF FIELD (IN-PLACE) FOR REGIONS OF
C                POOR HORIZONTAL RESOLUTION
C                  FIELD(I,J) = FAC*(FIELD(I,J)-CON)+CON, WHERE
C                FAC = (HMIN(I,J)/H(I,J))**IPOW,
C                H(I,J) = SQRT(HX(I)*HX(I) + HY(J)*HY(J)),
C                HMIN(I,J)= SMALLEST(H(I,J)).
C
C  AUTHOR:  ROGER DALEY - SUMMER 1977
C
C REVISION 001:  A. STANIFORTH - C. THIBEAULT  JAN 79 DOCUMENTATION
C REVISION 002:  T. GOOS - 1979-80 ADAPTED FOR AMDAHL AT U OF A.
C
C  CALL:  CALL MSHFLT(FIELD,WKX,WKY,CON,NI,NJ,IPOW)
C
```

```
C   ARGUMENST
C    IN-OUT - FIELD - FIELD USED FOR IN-PLACE AMPLITUDE REDUCTION
C    IN      - XI     - X-SPECIFICATION OF GRID
C            - YI     - Y-SPECIFICATION OF GRID
C            - WKX    - WORK(NI)
C           - WKY     - WORK(NJ)
C            - CON    - PARAMETER CONTROLLING MEAN OF FIELD
C           - NI     - X-DIMENSION
C            - NJ     - Y-DIMENSION
C           - IPOW    - PARAMETER OF AMPLITUDE REDUCTION
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FIELD(NI,NJ)
      DIMENSION WKX(NI),WKY(NJ)
      FMIN=1.D12
      DO 50 J=1,NJ
      DO 50 I=1,NI
      SX = DSQRT(WKX(I)*WKX(I) + WKY(J)*WKY(J))
      IF(SX.LE.FMIN) FMIN=SX
   50 CONTINUE
      DO 100 J=1,NJ
      DO 100 I=1,NI
      SX = DSQRT(WKX(I)*WKX(I) + WKY(J)*WKY(J))
      FAC = (FMIN/SX)**IPOW
  100 FIELD(I,J)=FAC*(FIELD(I,J)-CON)+CON
      RETURN
      END
      SUBROUTINE REL(PHIG,PG,NI,NJ,ALX,RELF,XTOL)
      IMPLICIT REAL *8(A-H,O-Z)
C
C  PERFORMS RELAXATION SOLUTION FOR POISSONS EQUATION.
C
      DIMENSION PG(NI,NJ),PHIG(NI,NJ)
      COMMON / HXMESH / HX(1)
      COMMON / RATIO1 / RAT1(63,27)
      COMMON / RATII1 / RATI1(63,27)
      COMMON / RATIO4 / RAT4(63,27)
      FIVE=5.0D0
      NIM=NI-1
      NJM=NJ-1
      NI2=NI-2
      NJ2=NJ-2
      NJ4=NJ-4
      DNI3=2.0D0*ALX
      DNI3=1.0D0/DNI3
      RELFP=1.0D0-RELF
      DO 1000 K=1,140
      PHI=0.0D0
      SUM=0.0
      DO 20 I=2,NIM
      I1=I-1
      H=HX(I)+HX(I1)
   20 SUM=SUM+H*PHIG(I,2)
      H=HX(1)+HX(NIM)
      SUM=SUM+PHIG(1,2)*H
      SUM=SUM*DNI3
      DO 25 I=1,NI
   25 PHIG(I,1)=SUM
      DO 300 J=2,NJM
      J1=J-1
      SUP=(PHIG(1,J1)+PHIG(NI,J1))/2.0D0
      PHIG(1,J1)=SUP
      PHIG(NI,J1)=SUP
      J2=J+1
      DO 300 I=2,NIM
```

```
      Z=REL[illegible]PHIG(I,J)
      I1=I-1
      I2=I+1
      RA11=RAT1(I1,J1)
      RAO1=RAT1(I,J1)
      RA1O=RAT1(I1,J)
      RA=RAT1(I,J)
      RAI11=RATI1(I1,J1)
      RAIO1=RATI1(I,J1)
      RAI1O=RATI1(I1,J)
      RAI=RATI1(I,J)
      SUM=(RA11+RAI11)*PHIG(I1,J1)
      SUM=SUM+(RAO1+RAIO1)*PHIG(I2,J1)
      SUM=SUM+(RA1O+RAI1O)*PHIG(I1,J2)
      SUM=SUM+(RA+RAI)*PHIG(I2,J2)
      SUM=SUM+(FIVE*(RAI+RAI1O)-RA-RA1O)*PHIG(I,J2)
      SUM=SUM+(FIVE*(RAI11+RAIO1)-RA11-RAO1)*PHIG(I,J1)
      SUM=SUM+(FIVE*(RA1O+RA11)-RAI1O-RAI11)*PHIG(I1,J)
      SUM=SUM+(FIVE*(RA+RAO1)-RAI-RAIO1)*PHIG(I2,J)
      SUM=SUM/FIVE
      SUM=(SUM+PG(I,J))*RAT4(I,J)
C     WRITE(7,806) SUM,RAT4(I,J),PG(I,J),PHIG(I,J)
      PHIO=PHIG(I,J)
      PHIG(I,J)=Z+RELF*SUM
      IF(I.LT.18.OR.I.GT.46) GO TO 805
      IF(J.LE.5) GO TO 805
      IF(J.GE.NJ4) GO TO 805
      IF(PHIO.GT.-1.OD-50.AND.PHIO.LT.1.OD-50) GO TO 805
      PHIO=DABS((PHIG(I,J)-PHIO)/PHIO)
      PHI=DMAX1(PHIO,PHI)
  805 CONTINUE
  806 FORMAT(4(2X,E12.6))
  300 CONTINUE
      SUM=0.ODO
      DO 40 I=2,NIM
      I1=I-1
      H=HX(I)+HX(I1)
   40 SUM=SUM+H*PHIG(I,NJM)
      H=HX(1)+HX(NIM)
      SUM=SUM+PHIG(1,NJM)*H
      SUM=SUM*DNI3
      DO 45 I=1,NI
   45 PHIG(I,NJ)=SUM
      IF(K.EQ.1) GO TO 1000
      IF(PHI.LT.XTOL) GO TO 1505
 1000 CONTINUE
 1505 CONTINUE
      WRITE(7,1049) K,PHI
 1049 FORMAT('    K = ', I5,4X,E12.6)
      RETURN
      END
      SUBROUTINE TRANS(OUT,XCI,NI,NJ,NIP,NJP)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION OUT(NIP,NJP),XCI(NI,NJ)
      NIM=NI-1
      NJM=NJ-1
      DO 50 J=2,NJM
      J1=J-1
      DO 50 I=2,NIM
      I1=I-1
   50 OUT(I1,J1)=XCI(I,J)
      RETURN
      END
      SUBROUTINE RELHEL(PHIG,PG,NI,NJ,ALX,ALAM,RELF,XTOL)
      IMPLICIT REAL *8(A-H,O-Z)
C
```

```
C   PERFORMS RELAXATION SOLUTION FOR HELMHOLZ PROBLEM
C
      DIMENSION PG(NI,NJ),PHIG(NI,NJ)
      COMMON / HXMESH / HX(1)
      COMMON / RATIO1 / RAT1(63,27)
      COMMON / RATII1 / RATI1(63,27)
      COMMON / RATIO4 / RAT4(63,27)
      COMMON / RATIO0 / RAT0(63,27)
      COMMON / RATIO2 / RAT2(63,27)
      COMMON / RATIO3 / RAT3(63,27)
      COMMON / RATIO5 / RAT5(63,27)
      FIVE=5.0D0
      ALAMP=ALAM/60.D0
      NIM=NI-1
      NJM=NJ-1
      NI2=NI-2
      NJ2=NJ-2
      NJ4=NJ-4
      DNI3=2.0D0*ALX
      DNI3=1.0D0/DNI3
      RELFP=1.0D0-RELF
      DO 1000 K=1,140
      PHI=0.0D0
      SUM=0.0
      DO 20 I=2,NIM
      I1=I-1
      H=HX(I)+HX(I1)
   20 SUM=SUM+H*PHIG(I,2)
      H=HX(1)+HX(NIM)
      SUM=SUM+PHIG(1,2)*H
      SUM=SUM*DNI3
      DO 25 I=1,NI
   25 PHIG(I,1)=SUM
      DO 300 J=2,NJM
      J1=J-1
      SUP=(PHIG(1,J1)+PHIG(NI,J1))/2.0D0
      PHIG(1,J1)=SUP
      PHIG(NI,J1)=SUP
      J2=J+1
      DO 300 I=2,NIM
  828 CONTINUE
      Z=RELFP*PHIG(I,J)
      I1=I-1
      I2=I+1
      P11=PHIG(I1,J1)
      P21=PHIG(I2,J1)
      P12=PHIG(I1,J2)
      P22=PHIG(I2,J2)
      P02=PHIG(I,J2)
      P01=PHIG(I,J1)
      P10=PHIG(I1,J)
      P20=PHIG(I2,J)
      RA11=RAT1(I1,J1)
      RA01=RAT1(I,J1)
      RA10=RAT1(I1,J)
      RA=RAT1(I,J)
      RAI11=RATI1(I1,J1)
      RAI01=RATI1(I,J1)
      RAI10=RATI1(I1,J)
      RAI=RATI1(I,J)
      SUM=(RA11+RAI11)*P11
      SUM=SUM+(RA01+RAI01)*P21
      SUM=SUM+(RA10+RAI10)*P12
      SUM=SUM+(RA+RAI)*P22
      SUM=SUM+(FIVE*(RAI+RAI10)-RA-RA10)*P02
      SUM=SUM+(FIVE*(RAI11+RAI01)-RA11-RA01)*P01
```

```
      SUM=SUM+(FIVE*(RA10+RA11)-RAI10-RAI11)*P10
      SUM=SUM+(FIVE*(RA+RAO1)-RAI-RAIO1)*P20
      SUM=SUM/5.0D0
      SUM5=RAT2(I,J)*PO2  +RAT3(I,J)*P20
      SUM5=SUM5+RAT3(I1,J)*P10 + RAT2(I,J1)*PO1
      SUM5=SUM5+FIVE+RATO(I,J)*P22+RATO(I1,J)*P12
      SUM5=SUM5+RATO(I1,J1)*P11+RATO(I,J1)*P21
      SUM5=SUM5*ALAMP
      SUM=SUM-SUM5
  210 SUM=(SUM+PG(I,J))*RAT5(I,J)
C     WRITE(7,600) SUM,SUM5,RAT5(I,J),PG(I,J),PHIG(I,J)
      PHIO=PHIG(I,J)
      PHIG(I,J)=Z+RELF*SUM
      IF(I.LT.8.OR.I.GT.46) GO TO 805
      IF(J.LE.3) GO TO 805
      IF(J.GE.NJ4) GO TO 805
      IF(PHIO.GT.-1.OD-50.AND.PHIO.LT.1.OD-50) GO TO 805
      PHIO=DABS((PHIG(I,J)-PHIO)/PHIO)
      PHI=DMAX1(PHIO,PHI)
  805 CONTINUE
  827 FORMAT(3I6,E12.6,3X,E12.6)
  300 CONTINUE
      SUM=0.0D0
      DO 40 I=2,NIM
      I1=I-1
      H=HX(I)+HX(I1)
   40 SUM=SUM+H*PHIG(I,NJM)
      H=HX(1)+HX(NIM)
      SUM=SUM+PHIG(1,NJM)*H
      SUM=SUM*DNI3
      DO 45 I=1,NI
   45 PHIG(I,NJ)=SUM
      IF(K.EQ.1) GO TO 1000
      IF(PHI.LT.XTOL) GO TO 1505
 1000 CONTINUE
 1505 CONTINUE
      WRITE(7,1049) K,PHI
 1049 FORMAT('  HELM.  K=',I5,4X,E12.6)
  610 FORMAT(1X,8(1X,E12.6))
  600 FORMAT(8(2X,E12.6))
      RETURN
      END
      SUBROUTINE FPLOT(Z,NIU,NJU,IFLAG)
      REAL*8 Z(NIU,NJU)
C
C  PLOTS FIELDS IN 2 SECTIONS; ONE OVER THE NON-UNIFORM PORTION
C  OF THE GRID AND ONE INCLUDING THE UNIFORM PORTION OF THE
   GRID.

      DIMENSION F(92,45)
      DIMENSION CVAL1(19),CVAL2(19),VOP(8),IOP(8)

    NTOUR VALUES FOR HEIGHT AND THICKNESS FIELDS.

      ATA CVAL1/540.0,480.0,420.0,360.0,300.0,240.0,180.0,
       20.0,60.0,0    -60.0,-120.0,-180.0,-240.0,-300.0,
       60.   20.0, 180.0,-540.0/

          IES FOR OMEGA FIELD.

           2/3.6,3.  2.8,2.4,2.0,1.6,1.2,.8,.4,0.0,
              2,-    -2.0,-2.4,-2.8,-3.2,-3.6/
       TA VOP/  ,0.0.    0.0,-1.0,0.0,0.0,0.0/
      
      OP
      IO
```

```
      IOP(4)=1
      IOP(6)=0
      IOP(7)=1
      IOP(8)=0
      IOP(5)=IFLAG
      NC=19
      CALL ORIGIN(999.6.5,3.5,1.0,1.0)
C
C  PLOT VARIABLE PORTION
C
      DO 100 J=1,45
      DO 100 I=1,92
  100 F(I,J)=Z(I,J)
      IF(IFLAG.EQ.0) CALL CONTUR(5.5,2.75,F,92,92,45,CVAL2,NC,IOP,VOP)
      IF(IFLAG.EQ.1) CALL CONTUR(5.5,2.75,F,92,92,45,CVAL1,NC,IOP,VOP)
      CALL ORIGIN(999.6.5,3.5,1.0,1.0)
C
C  PLOT UNIFORM PORTION
C
      DO 150 J= 1,45
      DO 150 I=92,183
      I1=I-91
  150 F(I1,J)=Z(I,J)
      IF(IFLAG.EQ.0) CALL CONTUR(5.5,2.75,F,92,92,45,CVAL2,NC,IOP,VOP)
      IF(IFLAG.EQ.1) CALL CONTUR(5.5,2.75,F,92,92,45,CVAL1,NC,IOP,VOP)
      RETURN
      END
END OF FILE
```

```
C
C
C  FILE   DIAG
C
C  FILES
C     OBJECT - DIAG
C
C      INPUT
C            1 = FEDIAG
C            2 = SPDIAG
C            3 = VMESH
C            4 = MESHD
C            5 = SPECD
C     OUTPUT
C            6 = ENERGIES AND ENSTROPHY
C            7 = ERRORS IN MAP FORM
C            8 = MEAN ERRORS AND S1 SCORES
C            9 = -PDIAG
C
C  COMPUTES DIAGNOSTIC VALUE   'OM THE SPECTRAL AND FEM FORECASTS.
C  THESE VALUES INCLUDE:
C        KINETIC ENERGIES
C        AVAILABLE POTENTIAL ENERGIES
C        POTENTIAL ENSTROPHYS
C        MEAN DIFFERENCE BETWEEN SOLUTIONS
C        MEAN ABSOLUTE DIFFERENCE BETWEEN SOLUTIONS
C        S1 SCORES
C
C  USES SYSTEM ROUTINES TO PLOT GRAPHS OF THESE VALUES.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XCSP(63,27),TASP(63,27),XCFE(63,27),TAFE(63,27)
      DIMENSION XCSPO(63,27),TASPO(63,27),XCFEO(63,27),TAFEO(63,27)
      DIMENSION X(281),Y(45),XI(63),YI(27)
      DIMENSION HX1(63),HY1(27)
      DIMENSION APE(9),EKE(9),TOT(9),PE(9)
      DIMENSION APEF(9),EKEF(9),TOTF(9),PEF(9)
      DIMENSION DEXC(9),ADEXC(9),DETA(9),ADETA(9)
      DIMENSION S1XC(9),S1TA(9)
      REAL ZZ(11,9),T(11)
      DIMENSION ZKESP(9),EKESP(9),ZAPESP(9),EAPESP(9)
      DIMENSION ZKEFE(9),EKEFE(9),ZAPEFE(9),EAPEFE(9)
      DIMENSION APET(9), APETF(9),EKET(9),EKETF(9)
      DIMENSION TOTT(9),TOTTF(9),PET(9),PETF(9)
      REAL TI1(3),TI2(4),TI3(2),TI4(5)
      LOGICAL*1 LFMT(1) /'*'/
      DATA T/0.0,6.0,12.0,18.0,24.0,30.0,36.0,42.0,48.0,0.0,0.0/
      DATA TI1/'ENER','GY  ','(J) '/
      DATA TI2/'ENST','ROPH','Y(KG',' S )'/
      DATA TI3/'S1 S','CORE'/
      DATA TI4/'MEAN',' DIF','FERE','NCE ','(M) '/
      COMMON / HXMESH / HX(63)
      COMMON / HYMESH / HY(27)
      COMMON / WKS1D1 / WK1(63)
      COMMON / WKS1D2 / WK2(63)
      COMMON / WKS1D3 / WK3(63)
      COMMON / WKS1D4 / WK4(63)
      COMMON / WKS1D5 / WK5(63)
      COMMON / WKS1D6 / WK6(63)
      COMMON / WKS1D7 / WK7(63)
      COMMON / WKS1D8 / WK8(63)
      COMMON / WKS1D9 / WK9(63)
      COMMON / WKS110 / WK10(63)
      COMMON / WORKA / WKA(63,27)
```

```
      COMMON / WORKB / WKB(63,27)
      COMMON / WORKC / WKC(63,27)
      COMMON / WORKD / WKD(63,27)
      COMMON / WORKE / WKE(63,27)
      COMMON / WORKF / WKF(63,27)
C
C SET CONSTANTS
C
C
C CXCI=L*L*FO*FO/G
C
      CXCI=2.15357266D4
C
C CPE=2.0D0*FO*FO/(DP*DP*SIG)
C
      CPE=3.0382098D-12
      CEKE=2.1383390D22
      CAPE=6.4967228D10
      CCPE=1.0691695E22
C
C READ PARAMETERS OF GRID.
C
      READ(4,LFMT) NIU,NJU
      READ(4,LFMT) NI,NJ
      READ(3,LFMT)(X(I),I=1,NIU)
      READ(3,LFMT)(Y(J),J=1,NJU)
      READ(3,LFMT)(XI(I),I=1,NI)
      READ(3,LFMT)(YI(J),J=1,NJ)
      NIP=NI
      NJP=NJ
C
C CALCULATE GRID LENGTHS.
C
      CALL CALH(HX1,XI,NIP)
      CALL CALH(HY1,YI,NJP)
      NJPM=NJP-1
      NIPM=NIP-1
      HX1(NIP)=HX1(NIPM)
      HY1(NJP)=HY1(NJPM)
      NIM=NI-1
      NJM=NJ-1
      DO 60 I=1,NIP
   60 HX(I)=HX1(I)
      DO 65 J=1,NJP
   65 HY(J)=HY1(J)
C
C READ INITIAL AMPLITUDES
C
      READ(5,LFMT) X1,X2,X3,T1,T2,T3
C
C N   NO. OF WAVES, M = NO. OF TIME STEPS
C       TIMESTEP, IPT2 = NO. OF HOURS BETWEEN MAPS
C
      READ(5,LFMT) N,M,DT,IDT2
C
C CALCULATE NO. OF RECORDS ON FILE
C
      NR=(M-1)/IDT2+1
C
C SET BOUNDARIES OF COMPARISON GRID(S)
C
      IL1=23
      IR1=40
      JT1=18
      JB1=10
C
```

```
C  MAIN LOOP
C
      H=HX(30)
      DO 1000 II=1,NR
      S1XC(II)=0.000
      S1TA(II)=0.000
C
C  READ SPECTRAL FIELDS
C
      READ(2) XCSP
      READ(2) TASP
C
C  READ FEM FIELDS
C
      READ(1) XCFE
      READ(1) TAFE
      IF(II.NE.1) GO TO 200
      DO 180 J=1,NJ
      DO 180 I=1,NI
      XCSPO(I,J)=XCSP(I,J)
      TASPO(I,J)=TASP(I,J)
      XCFEO(I,J)=XCFE(I,J)
      TAFEO(I,J)=TAFE(I,J)
  180 CONTINUE
  200 CONTINUE
C
C  CALCULATE ENERGIES AND ENSTROPHIES
C
      CALL APOTEN(APE(II),IL1,IR1,JT1,JB1,TASP,NI,NJ,
     1 H,APET(II),1)
      CALL APOTEN(APEF(II),IL1,IR1,JT1,JB1,TAFE,NI,NJ,
     1 H,APETF(II),1)
      CALL EKINEN(XCSP,TASP,NI,NJ,H,EKE(II),IL1,IR1,JT1,
     1 JB1,EKET(II),1)
      CALL EKINEN(XCFE,TAFE,NI,NJ,H,EKEF(II),IL1,IR1,JT1,
     1 JB1,EKETF(II),1)
      CALL POTENS(XCSP,TASP,NI,NJ,H,PE(II),IL1,IR1,JT1,
     1 JB1,CPE,PET(II),1)
      CALL POTENS(XCFE,TAFE,NI,NJ,H,PEF(II),IL1,IR1,JT1,
     1 JB1,CPE,PETF(II),1)
      PE(II)=PE(II)*CCPE
      APE(II)=APE(II)*CAPE
      EKE(II)=EKE(II)*CEKE
      TOT(II)=APE(II)+EKE(II)
      PEF(II)=PEF(II)*CCPE
      APEF(II)=APEF(II)*CAPE
      EKEF(II)=EKEF(II)*CEKE
      TOTF(II)=APEF(II)+EKEF(II)
      PET(II)=PET(II)*CCPE
      APET(II)=APET(II)*CAPE
      EKET(II)=EKET(II)*CEKE
      TOTT(II)=APET(II)+EKET(II)
      PETF(II)=PETF(II)*CCPE
      APETF(II)=APETF(II)*CAPE
      EKETF(II)=EKETF(II)*CEKE
      TOTTF(II)=APETF(II)+EKETF(II)
      IF(II.EQ.1) GO TO 1001
C
C  CALCULATE DIFFERENCE IN LAST (II-1) OUTPUTS
C
C  TRUE DIFF. FOR XCI FIELD
C
      CALL GDADGD(WKA,1.000,XCSP,-1.000,XCSPO,NI,NJ)
C
C  FORECAST DIFF. FOR XCI FIELD
C
```

```
      CALL GDADGD(WKB,1.ODO,XCFE,-1.ODO,XCFEO,NI,NJ)
C
C   FIND ERROR IN FORECAST
C
      CALL GDADGD(WKC,1.ODO,WKA,-1.ODO,WKB,NI,NJ)
      CALL MEAN(WKC,NI,NJ,DEXC(II),IL1,IR1,JT1,JB1)
      CALL AMEAN(WKC,NI,NJ,ADEXC(II),IL1,IR1,JT1,JB1)
C
C   CALCULATE DIFFERENCE IN LAST (II-1) OUTPUTS
C
C   TRUE DIFF. FOR TAU FIELD.
C
      CALL GDADGD(WKA,1.ODO,TASP,-1.ODO,TASPO,NI,NJ)
C
C   FORECAST DIFF. FOR TAU FIELD
C
      CALL GDADGD(WKB,1.ODO,TAFE,-1.ODO,TAFEO,NI,NJ)
C
C   FIND ERROR IN FORECAST
C
      CALL GDADGD(WKC,1.ODO,WKA,-1.ODO,WKB,NI,NJ)
      CALL MEAN(WKC,NI,NJ,DETA(II),IL1,IR1,JT1,JB1)
      CALL AMEAN(WKC,NI,NJ,ADETA(II),IL1,IR1,JT1,JB1)
C
C   CALCULATE S1 SCORES
C
C   FOR XCI
C
      CALL DXDYDS(WKB,XCSP,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKA,WKB,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL DXDYDS(WKC,XCSP,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKB,WKC,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
      CALL DXDYDS(WKD,XCFE,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKC,WKD,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL DXDYDS(WKE,XCFE,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKD,WKE,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
C
C   ERROR IN X-PRESS. GRAD - WKE
C
      CALL GDADGD(WKE,1.ODO,WKC,-1.ODO,WKA,NI,NJ)
C
C   ERROR IN Y-PRESS. GRAD - WKF
C
      CALL GDADGD(WKF,1.ODO,WKD,-1.ODO,WKB,NI,NJ)
C
C   FORM SUM OF ABS OF ERRORS
C
      CALL SUM(WKE,NI,NJ,SUM1,IL1,IR1,JT1,JB1)
      CALL SUM(WKF,NI,NJ,SUM2,IL1,IR1,JT1,JB1)
      SUM3=SUM1+SUM2
      CALL SUMB(WKC,WKA,NI,NJ,SUM1,IL1,IR1,JT1,JB1)
      CALL SUMB(WKD,WKB,NI,NJ,SUM2,IL1,IR1,JT1,[illegible])
      SUM4=SUM1+SUM2
      IF(DABS(SUM4).LT.1.OD-50) GO TO 435
      S1XC(II)=100.ODO*SUM3/SUM4
C
C   CALCULATE S1 SCORE FOR TAU.
C
  435 CALL DXDYDS(WKB,TASP,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKA,WKB,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL DXDYDS(WKC,TASP,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKB,WKC,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
      CALL DXDYDS(WKD,TAFE,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKC,WKD,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL DXDYDS(WKE,TAFE,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKD,WKE,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
```

```
C
C  ERROR IN X-PRESS  GRAD - WKE
C
      CAL' GDADGD(WKE,1.0D0,WKC,-1.0D0,WKA,NI,NJ)
C
C  ERROR  Y-PRESS  GRAD - WKF
C
      CALL GDADGD(WKF,1.0D0,WKD,-1.0D0,WKB,NI,NJ)
C
C  FORM SUM OF ABS  OF ERRORS
C
      CALL SUM(WKE,NI,NJ,SUM1,IL1,IR1,JT1,JB1)
      CALL SUM(WKF,NI,NJ,SUM2,IL1,IR1,JT1,JB1)
      SUM3=SUM1+SUM2
      CALL SUMB(WKC,WKA,NI,NJ,SUM1,IL1,IR1,JT1,JB1)
      CALL SUMB(WKD,WKB,NI,NJ,SUM2,IL1,IR1,JT1,JB1)
      SUM4=SUM1+SUM2
      IF(DABS(SUM4).LT.1.0D-50) GO TO 436
      SITA(II)=100.0D0*SUM3/SUM4
  436 CONTINUE
 1001 CONTINUE
C
C  CALCULATE ZONAL AND EDDY ENERGIES
C
      CALL ZEENE(XCSP,TASP,NI,NJ,JT1,JB1,IL1,IR1,H,
     1 ZKESP(II),EKESP(II),ZAPESP(II),EAPESP(II))
      CALL ZEENE(XCFE,TAFE,NI,NJ,JT1,JB1,IL1,IR1,H,
     1 ZKEFE(II),EKEFE(II),ZAPEFE(II),EAPEFE(II))
 1000 CONTINUE
C
C  INITIALIZE PLOTTING PROGRAM.
C
      CALL PLOTS
      CALL ORGEP(1.0,'      0)
C
C  PLOT ENERGIES
C
      DO 650 IT=1,NR
      ZZ(IT,1)=EKE(IT)
      ZZ(IT,2)=EKEF(IT)
      ZZ(IT,3)=APE(IT)
      ZZ(IT,4)=APEF(IT)
      ZZ(IT,5)=TOT(IT)
      ZZ(IT,6)=TOTF(IT)
  650 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,6,3.0,3.0,TI1,3)
C
C  PLOT KINETIC ENERGIES.
C
      DO 690 IT=1,NR
      ZZ(IT,1)=EKE(IT)
      ZZ(IT,2)=EKEF(IT)
      ZZ(IT,3)=ZKESP(IT)*CEKE
      ZZ(IT,4)=ZKEFE(IT)*CEKE
      ZZ(IT,5)=EKESP(IT)*CEKE
      ZZ(IT,6)=EKEFE(IT)*CEKE
  690 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,6,3.0,3.0,TI1,3)
C
C  PLOT POTENTIAL ENERGIES.
C
      DO 695 IT=1,NR
      ZZ(IT,1)=APE(IT)
      ZZ(IT,2)=APEF(IT)
      ZZ(IT,3)=ZAPESP(IT)*CAPE
      ZZ(IT,4)=ZAPEFE(   )*CAPE
```

```
      ZZ(IT,5)=EAPESP(IT)*CAPE
      ZZ(IT,6)=EAPEFE(IT)*CAPE
  695 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,6,3.0,3.0,TI1,3)
C
C  PLOT ENSTROPHY
C
      DO 655 IT=1,NR
      ZZ(IT,1)=PE(IT)
      ZZ(IT,2)=PEF(IT)
  655 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,2,3.0,3.0,TI2,4)
C
C  PLOT S1 SCORES
C
      DO 660 IT=1,NR
      ZZ(IT,1)=S1XC(IT)
      ZZ(IT,2)=S1TA(IT)
  660 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,2,3.0,3.0,TI3,2)
C
C  PLOT MEAN ENERGIES
C
      DO 665 IT=1,NR
      ZZ(IT,1)=DEXC(IT)*CXCI
      ZZ(IT,2)=ADEXC(IT)*CXCI
      ZZ(IT,3)=DETA(IT)*CXCI
      ZZ(IT,4)=ADETA(IT)*CXCI
  665 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,4,3.0,3.0,TI4,5)
C
C  PLOT TOTALS
C
      DO 830 IT=1,NR
      ZZ(IT,1)=EKET(IT)
      ZZ(IT,2)=EKETF(IT)
      ZZ(IT,3)=APET(IT)
      ZZ(IT,4)=APETF(IT)
      ZZ(IT,5)= TOTT(IT)
      ZZ(IT,6)=TOTTF(IT)
  830 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,6,3.0,3.0,TI1,3)
      DO 840 IT=1,NR
      ZZ(IT,1)=PET(IT)
      ZZ(IT,2)=PETF(IT)
  840 CONTINUE
      CALL LPLOT(ZZ,T,NR+2,2,3.0,3.0,TI2,4)
      CALL PLOT(0.0,0.0,
      STOP
      END
      SUBROUTINE FINT(F,NI,NJ,H,SUM,IL,IR,JT,JB)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(NI,NJ)
C
C  PERFORMS INTEGRATION OF F OVER THE SQUARE
C  IL<=X<=IR  :  JB<=Y<=JT
C  ASSUMING A UNIFORM MESH LENGTH H IN BOTH DIRECTIONS.
C
      SUM=0.0
C
C  DO EDGES.
C
      IL1=IL+1
      IR1=IR-1
      DO 300 I=IL1,IR1
  300 SUM=SUM+F(I,JT)
```

```
      DO 310 I=IL1,IR1
  310 SUM=SUM+F(I,JB)
      JT1=JT-1
      JB1=JB+1
      DO 320 J=JB1,JT1
  320 SUM=SUM+F(IL,J)+F(IR,J)
      SUM=SUM*.50D0
C
C  DO CORNERS.
C
      SUM1=F(IL,JT)+F(IR,JT)+F(IL,JB)+F(IR,JB)
      SUM=SUM+.25D0*SUM1
C
C  DO INTERIOR.
C
      SUM1=0.0
      DO 350 J=JB1,JT1
      DO 350 I=IL1,IR1
  350 SUM1=SUM1+F(I,J)
      SUM=SUM1+SUM
      SUM=SUM*H*H
      RETURN
      END
      SUBROUTINE EKINEN(XCSP,TASP,NI,NJ,H,EKE,IL1,IR1,
     1 JT1,JB1,EKET,IFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XCSP(63,27),TASP(63,27)
      COMMON / HXMESH / HX(63)
      COMMON / HYMESH / HY(27)
      COMMON / WKS1D1 / WK1(63)
      COMMON / WKS1D2 / WK2(63)
      COMMON / WKS1D3 / WK3(63)
      COMMON / WKS1D4 / WK4(63)
      COMMON / WKS1D5 / WK5(63)
      COMMON / WKS1D6 / WK6(63)
      COMMON / WKS1D7 / WK7(63)
      COMMON / WKS1D8 / WK8(63)
      COMMON / WKS1D9 / WK9(63)
      COMMON / WKS110 / WK10(63)
      COMMON / WORKA / WKA(63,27)
      COMMON / WORKB / WKB(63,27)
      COMMON / WORKC / WKC(63,27)
      COMMON / WORKD / WKD(63,27)
C
C  FIND KINETIC ENERGY.
C
      CALL DXDYDS(WKA,XCSP,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL NLLOOP(WKA,WKB,WKB,NI,NJ)
      CALL PSOLVE(WKD,WKA,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL DXDYDS(WKA,XCSP,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
      CALL NLLOOP(WKA,WKB,WKB,NI,NJ)
      CALL PSOLVE(WKC,WKA,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL GDADGD(WKD,1.0D0,WKD,1.0D0,WKC,NI,NJ)
      CALL DXDYDS(WKA,TASP,1.0D0,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.TRUE.,.FALSE.,.FALSE.)
      CALL NLLOOP(WKA,WKB,WKB,NI,NJ)
      CALL PSOLVE(WKC,WKA,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL GDADGD(WKD,1.0D0,WKD,1.0D0,WKC,NI,NJ)
      CALL DXDYDS(WKA,TASP,1.0D0,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.FALSE.,.TRUE.,.FALSE.)
      CALL NLLOOP(WKA,WKB,WKB,NI,NJ)
      CALL PSOLVE(WKC,WKA,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL GDADGD(WKD,1.0D0,WKD,1.0D0,WKC,NI,NJ)
      CALL FINT(WKD,NI,NJ,H,EKE,IL1,IR1,JT1,JB1)
```

```
      IF(IFLAG.EQ.1) CALL TINT(WKD,EKET,NI,NJ)
      RETURN
      END
      SUBROUTINE POTENS(XCSP,TASP,NI,NJ,H,PE,IL1,IR1,
     1 JT1,JB1,CPE,PET,IFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XCSP(NI,NJ),TASP(NI,NJ)
      COMMON / HXMESH / HX(63)
      COMMON / HYMESH / HY(27)
      COMMON / WKS1D1 / WK1(63)
      COMMON / WKS1D2 / WK2(63)
      COMMON / WKS1D3 / WK3(63)
      COMMON / WKS1D4 / WK4(63)
      COMMON / WKS1D5 / WK5(63)
      COMMON / WKS1D6 / WK6(63)
      COMMON / WKS1D7 / WK7(63)
      COMMON / WKS1D8 / WK8(63)
      COMMON / WKS1D9 / WK9(63)
      COMMON / WKS110 / WK10(63)
      COMMON / WORKA / WKA(63,27)
      COMMON / WORKB / WKB(63,27)
      COMMON / WORKC / WKC(63,27)
      COMMON / WORKD / WKD(63,27)
C
C  FORM LAPLACE OF XCI
C
      CALL D2XYS(WKA,XCSP,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.TRUE.,.FALSE.,.TRUE.)
      CALL D2XYS(WKA,XCSP,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKC,WKA,NI,NJ,.FALSE.,.TRUE.,.TRUE.)
      CALL GDADGD(WKD,1.ODO,WKB,1.ODO,WKC,NI,NJ)
C
C  FORM LAPLACE OF TAU
C
      CALL D2XYS(WKA,TASP,1.ODO,.TRUE.,.FALSE.,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.TRUE.,.FALSE.,.TRUE.)
      CALL D2XYS(WKA,TASP,1.ODO,.FALSE.,.TRUE.,NI,NJ)
      CALL PSOLVE(WKC,WKA,NI,NJ,.FALSE.,.TRUE.,.TRUE.)
      CALL GDADGD(WKC,1.ODO,WKC,1.ODO,WKB,NI,NJ)
C
C  DO SUMS.
C
      CALL GDADGD(WKA,1.ODO,WKD,1.ODO,WKC,NI,NJ)
      CALL GDADGD(WKA,1.ODO,WKA,-CPE,TASP,NI,NJ)
      CALL GDADGD(WKB,1.ODO,WKD,-1.ODO,WKC,NI,NJ)
      CALL GDADGD(WKB,1.ODO,WKB,CPE,TASP,NI,NJ)
C
C  FORM SQUARES.
C
      CALL NLLOOP(WKC,WKA,WKA,NI,NJ)
      CALL PSOLVE(WKA,WKC,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL NLLOOP(WKC,WKB,WKB,NI,NJ)
      CALL PSOLVE(WKB,WKC,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
C
C  FORM SUM.
C
      CALL GDADGD(WKC,1.ODO,WKA,1.ODO,WKB,NI,NJ)
C
C  INTEGRATE.
C
      CALL FINT(WKC,NI,NJ,H,PE,IL1,IR1,JT1,JB1)
      IF(IFLAG.EQ.1) CALL TINT(WKC,PET,NI,NJ)
      RETURN
      END
      SUBROUTINE MEAN(FIE,NI,NJ,AVE,IL,IR,JT,JB)
C
```

```
C  FIND ARITHMETIC MEAN OF FIELD.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FIE(NI,NJ)
      SUM=0.0D0
      ISUM=0
      DO 100 J=JB,JT
      DO 100 I=IL,IR
      ISUM=ISUM+1
  100 SUM=SUM+FIE(I,J)
      AVE=SUM/DFLOAT(ISUM)
      RETURN
      END
      SUBROUTINE AMEAN(FIE,NI,NJ,AVE,IL,IR,JT,JB)
C
C  FINDS ARITHMETIC MEAN OF [illegible]UTE VALUE OF FIELD
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FIE(NI,NJ)
      SUM=0.0D0
      ISUM=0
      DO 100 J=JB,JT
      DO 100 I=IL,IR
      ISUM=ISUM+1
  100 SUM=SUM+DABS(FIE(I,J))
      AVE=SUM/DFLOAT(ISUM)
      RETURN
      END
      SUBROUTINE SUM(FIE,NI,NJ,SUM1,IL,IR,JT,JB)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FIE(NI,NJ)
      SUM1=0.0D0
      DO 100 J=JB,JT
      DO 100 I=IL,IR
  100 SUM1=SUM1+DABS(FIE(I,J))
      RETURN
      END
      SUBROUTINE SUMB(FIE1,FIE2,NI,NJ,SUM1,IL,IR,JT,JB)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FIE1(NI,NJ),FIE2(NI,NJ)
      SUM1=0.0D0
      DO 100 J=JB,JT
      DO 100 I=IL,IR
      F1=DABS(FIE1(I,J))
      F2=DABS(FIE2(I,J))
  100 SUM1=SUM1+DMAX1(F1,F2)
      RETURN
      END
      SUBROUTINE LPLOT(PL,T,NT,IFLAG,XSIZE,YSIZE,TITLE,NTITLE)
      DIMENSION PL(NT,6),T(NT),SCA(4)
      REAL TITLE(NTITLE)
      NUM=4*NTITLE
      NT2=NT-2
      NT1=NT-1
C
C  SCALE HORIZONTAL AXIS
C
      CALL SCALE(T,XSIZE,NT2,1)
C
C  SCALE VERTICAL AXIS
C
      XZ1=XSIZE+.8
      YZ1=YSIZE+.8
      CALL ORIGIN(999,XZ1,YZ1,1.0,1.0)
      SC=0.0
      OR=1.0E55
```

```
      DO 100 I=1,IFLAG
      DO 100 J=1,NT2
      SC=AMAX1(SC,PL(J,I))
      OR=AMIN1(OR,PL(J,I))
  100 CONTINUE
      SCA(1)=OR
      SCA(2)=SC
      CALL SCALE(SCA,YSIZE,2,1)
      DO 125 I=1,IFLAG
      PL(NT,I)=SCA(4)
  125 PL(NT1,I)=SCA(3)
C
C  PLOT VERTICAL AXIS
C
      CALL AX2EP(1.0,3,2,0,.85)
      CALL AXIS2(0.0,0.0,TITLE,NUM,YSIZE,90.0,SCA(3),SCA(4),-1.0)
      CALL AXIS2(XSIZE,0.0,' ',1,-YSIZE,90.0,SCA(3),SCA(4),1.0)
C
C  PLOT HORIZONTAL AXIS
C
      CALL AX2EP(1.0,3,1,0,.85)
      CALL AXIS2(0.0,0.0,'TIME (HOURS)',-12,XSIZE,0.0,T(NT1),T(NT),1.)
      CALL AXIS2(0.0,YSIZE,' ',-1,-XSIZE,0.0,T(NT1),T(NT),1.)
C
C  PLOT CURVES
C
      DO 150 I=1,IFLAG
      CALL LINE(T,PL(1,I),NT2,1,1,I)
  150 CONTINUE
      RETURN
      END
      SUBROUTINE APOTEN(APE,IL1,IR1,JT1,JB1,F,NI,NJ,
     1 H,APET,IFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(NI,NJ)
      COMMON / HXMESH / HX(63)
      COMMON / HYMESH / HY(27)
      COMMON / WORKA / WKA(63,27)
      COMMON / WORKB / WKB(63,27)
      CALL NLLOOP(WKA,F,F,NI,NJ)
      CALL PSOLVE(WKB,WKA,NI,NJ,.TRUE.,.TRUE.,.FALSE.)
      CALL FINT(WKB,NI,NJ,H,APE,IL1,IR1,JT1,JB1)
      IF(IFLAG.EQ.1) CALL TINT(WKB,APET,NI,NJ)
      RETURN
      END
      SUBROUTINE ZEENE(F1,F2,NI,NJ,JT,JB,IL,IR,H,ZKE,EKE,ZAPE,EAPE)
C
C  SUBROUTINE TO CALCULATE ZONAL AND EDDY ENERGIES.
C
C  F1 = XCI
C  F2 = TAU
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F1(NI,NJ),F2(NI,NJ)
      COMMON / HXMESH / HX(63)
      COMMON / HYMESH / HY(27)
      COMMON / WKS1D1 / WK1(63)
      COMMON / WKS1D2 / WK2(63)
      COMMON / WKS1D3 / WK3(63)
      COMMON / WKS1D4 / WK4(63)
      COMMON / WKS1D5 / WK5(63)
      COMMON / WKS1D6 / WK6(63)
      COMMON / WKS1D7 / WK7(63)
      COMMON / WKS1D8 / WK8(63)
      COMMON / WKS1D9 / WK9(63)
      COMMON / WKS110 / WK10(63)
      COMMON / WORKA / WKA(63,27)
```

```
      COMMON / WORKB / WKB(63,27)
      COMMON / WORKC / WKC(63,27)
      COMMON / WORKD / WKD(63,27)
      COMMON / WORKE / WKE(63,27)
      COMMON / WORKF / WKF(63,27)
C
C  FORM AVERAGE IN Y-DIRECTION.
C
      ANI=DFLOAT(IR-IL+1)
      DO 100 J=1,NJ
      SUM=0.0D0
      SUM1=0.0D0
      DO 105 I=IL,IR
  105 SUM=SUM+F1(I,J)
      DO 106 I=IL,IR
  106 SUM1=SUM1+F2(I,J)
      SUM=SUM/ANI
      SUM1=SUM1/ANI
      DO 110 I=1,NI
  110 WKE(I,J)=SUM
      DO 115 I=1,NI
  115 WKF(I,J)=SUM1
  100 CONTINUE
C
C  CALCULATE ZKE
C
      CALL EKINEN(WKE,WKF,NI,NJ,H,ZKE,IL,IR,
     1 JT,JB,SUM,0)
C
C  CALCULATE ZAPE
C
      CALL APOTEN(ZAPE,IL,IR,JT,JB,WKF,NI,NJ,
     1 H,SUM,0)
      DO 140 J=1,NJ
      DO 140 I=1,NI
      WKE(I,J)=F1(I,J)-WKE(I,J)
      WKF(I,J)=F2(I,J)-WKF(I,J)
  140 CONTINUE
C
C  CALCULATE EKE
C
      CALL EKINEN(WKE,WKF,NI,NJ,H,EKE,IL,IR,
     1 JT,JB,SUM,0)
      CALL APOTEN(EAPE,IL,IR,JT,JB,WKF,NI,NJ,
     1 H,SUM,0)
      RETURN
      END
      SUBROUTINE TINT(F,SUM,NI,NJ)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F(NI,NJ)
      COMMON / HXMESH / HX(1)
      COMMON / HYMESH / HY(1)
      NJM=NJ-1
      NIM=NI-1
      SUM=0.0D0
      DO 100 J=2,NJM
      J1=J-1
      HYB=(HY(J1)+HY(J))/2.0D0
      DO 100 I=2,NIM
      I1=I-1
      HXB=(HX(I1)+HX(I))/2.0D0
      SUM=SUM+F(I,J)*HYB*HXB
  100 CONTINUE
C
C  TOP AND BOTTOM
C
```

```
      HYB=HY(1)/2.0D0
      HYB2=HY(NJM)/2.0D0
      DO 150 I=2,NIM
      I1=I-1
      HXB=(HX(I1)+HX(I))/2.0D0
      SUM=SUM+F(I,1)*HYB*HXB
      SUM=SUM+F(I,NJ)*HYB2*HXB
  150 CONTINUE
C
C  ENDS
C
      HXB=(HX(1)+HX(NIM))/2.0D0
      DO 200 J=2,NJM
      J1=J-1
      HYB=(HY(J1)+HY(J))/2.0D0
      SUM=SUM+F(1,J)*HYB*HXB
  200 CONTINUE
C
C  CORNERS
C
      HXB=(HX(1)+HX(NIM))/2.0D0
      SUM=SUM+F(1,1)*HXB*HY(1)/2.0D0
      SUM=SUM+F(1,NJ)*HXB*HY(NJM)/2.0D0
      RETURN
      END
END OF FILE
```