AN ABSTRACT OF THE THESIS OF

CHUEN TIEN SHYU   for the degree of DOCTOR OF PHILOSOPHY

in   GEOPHYSICS   presented on   April 26, 1979

Title: NUMERICAL ANALYSIS OF CRITICAL FIELD FUNCTIONS

FOR THERMAL CONVECTION IN VERTICAL OR QUASI-

VERTICAL DARCY FLOW SLABS

Abstract approved:

The numerical analysis of thermal convection in porous media,
heated from below, and assuming Darcy flow conditions, involves the
solving of a set of non-linear equations for the temperature and flow
fields. The condition of criticality determining the onset of convec-
tion is obtained by linearization and the solving of an eigenvalue prob-
lem of the fourth order. The smallest eigenvalue represents the
critical Rayleigh number. The shape of the critical temperature and
flow fields is then obtained from the linear set. In most practical
cases, the problem setting is such that closed analytical solutions
cannot be derived.

The difficulties of solving the convection equations can be
overcome by using the Galerkin finite-element method. The method
allows the solution of both the linear set and also the more complete
non-linear set of equations at various boundary conditions and taking

variations in the material parameters into account.

In this thesis, the Galerkin method is used to solve the convection equations for infinitely long porous vertical or semi-vertical slabs with prescribed temperatures at the top and bottom surfaces. The first set of models investigated involve boundary walls that are impermeable to the fluid but perfectly conducting to heat.

The critical Rayleigh numbers and critical temperature and flow fields are obtained for such slabs with various aspect ratios. The results show that the critical number is raised by 200 to 400% as compared with published data for similar slabs with thermally non-conducting walls.

The results are generalized by investigating cases of slabs with (1) three types of vertically varying permeability, (2) by taking the temperature dependence of the fluid properties into account, (3) by including non-linear terms, and finally, (4) a few cases of slabs with boundary walls of finite thermal conductivity are investigated.

The results are applicable to a number of situations in geothermal areas. A brief discussion of two such cases is given, that is, (1) the estimating of the critical permeability profile for the East geothermal field in the Imperial Valley and (2) the computation of a temperature cross section in the Cumali geothermal field in Turkey.

Numerical Analysis of Critical Field Functions for
Thermal Convection in Vertical or Quasi-
Vertical Darcy Flow Slabs

by

Chuen Tien Shyu

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Completed April 1979

Commencement June 1979
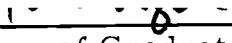
APPROVED:

Redacted for Privacy

Professor of Geophysics and Mathematics

in charge of major

Redacted for Privacy

Dean of School of Oceanography,

Redacted for Privacy

Dean of Graduate School

Date thesis is presented_____April 26, 1979_____

Typed by Clover Redfern for_____Chuen Tien Shyu_____

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NUMERICAL ANALYSIS OF CRITICAL FIELD FUNCTIONS
FOR THERMAL CONVECTION IN VERTICAL OR
QUASI-VERTICAL DARCY FLOW SLABS

I. INTRODUCTION

Thermal convection phenomena in porous media are of considerable interest in theoretical and applied geophysics. At a sufficiently high permeability, natural convection may take place in porous formations of almost any shape and dimensions. The fluid movements result in a convective transport of heat which may contribute to the natural heat flow and thus modify the local temperature field. A temperature field and heat flow theory based on purely conductive processes is inapplicable for such conditions.

The convective phenomena are of particular interest in situations of enhanced local heat flow such as in and around active geothermal systems and in areas of active volcanism. Convection may then contribute very significantly to the thermal phenomena and the outward flow of heat. As a result, the geothermal sciences take a very great interest in many theoretical and numerical aspects of the convective phenomena.

A problem setting of particular interest centers around the specific or critical conditions required to initiate thermal convection in a porous formation of given geometry, boundary conditions and material properties. Here we are not only interested in the critical

conditions that lead to convection but also in the shape of the numeri-

cal data on the resulting temperature and heat flow fields. As a mat-

ter of fact, the critical field functions have received very great

attention in the literature during the past two or three decades. The

problem setting that has been the focus of interest can be divided into

two main cases, that is, (1) instability analysis and (2) development

of time dependent models. Below we briefly review the literature in

these fields.

Classical Instability Analysis

Clasical linearized Rayleigh type convective instability analysis

(Rayleigh, 1916: Chandrasekhar, 1961) was applied by Horton and

Rogers (1945) and Lapwood (1948) to determine the critical Rayleigh

number (Rc) and flow pattern at the onset of convection in an

infinite horizontal layer of a saturated homogeneous and isotropic

porous Darcy type medium heated from below. Establishing the mass,

momentum and energy balances for this type of system, neglecting

the temperature and pressure dependence of the fluid, except where

they create buoyancy (Boussinesq approximation), and omitting all

nonlinear terms in the equations, Lapwood obtained for isothermal

and impermeable boundary surfaces, the value Rc = 39.5. Assum-

ing permeable surfaces the value is reduced to Rc = 27.1. Results

of experimental work by Katto and Masuoka (1967) show a satisfactory

agreement between theory and experiment. Using numerical tech-
niques, critical Rayleigh numbers for more general sets of boundary
conditions were obtained by Nield (1968). Analogue solutions for
similar systems including nonlinear terms have been given (Wooding,
1957: Wooding, 1958: Elder, 1967: Chan et al., 1970).

The procedure employed by Lapwood was extended by Beck
(1972) to investigate convection in porous boxes with insulated walls.
Zebib and Kassoy (1978) have used a two-term expansion of the tem-
perature and velocity vector fields to reinvestigate the box models
and conclude that when the Rayleigh number is just above the critical
value the two-dimensional convection mode transfers heat more
effectively than does the three-dimensional convection mode.
Recently, Zebib and Kassoy (1977) have reconsidered Beck's problem
by taking into account the effects of viscosity variation due to tem-
perature differences. They found that the general mode configuration
is the same as derived by Beck (1972) but, the critical Rayleigh num-
ber is significantly reduced as the thermal gradient increases. Straus
and Schubert (1977) have carried out a very complete analysis of the
effects of variable thermodynamic and transport properties of water
on the critical functions. They observed that the critical Rayleigh
number may be reduced very considerably below the values obtained
by Lapwood (1948). This implies that convection may occur at a
smaller temperature gradient than predicted by an analysis based on

the Boussinesq approximation.

Evolutionary Models

A few investigators have treated non-steady state or evolutionary models. Laboratory and numerical experiments on non-steady convective flows in porous slab systems have been reported by Elder (1967). A numerical and experimental analysis of convection in box geometries with insulated boundaries has been carried out by Holst and Aziz (1972). Mercer and others (Mercer and Pinder, 1973: Mercer, Pinder and Donaldson, 1975) have carried out a numerical simulation of the hydrothermal system at Wairakei, New Zealand. They assume a temperature dependent viscosity and thermal expansion coefficient. Mercer et al. (1975) point out, that their models are essentially two dimensional (horizontally). Cases involving multiphase fluids were investigated by Lasseter et al. (1975) and Faust and Mercer (1975). Sorey (1976) further investigated this problem in a vertical slab model, and demonstrated the importance of the variations in the physical properties of the fluid. However, Sorey (1976) did not treat the convection instability systematically and his analysis has been criticized by Straus and Schubert (1977).

Scope of the Present Research

The principal aim of the work by Beck (1972) and Zebib and

Kassoy (1977) quoted above has been to modify the earlier results so

that they could be made applicable to problems of convective stability

in specific geological structures such as individual fault blocks where

Darcy type flow conditions can be assumed. To reduce the computa-

tional effort, the walls bounding the blocks are assumed to be imper-

meable to the fluid and non-conductive to heat. Clearly, the latter

boundary condition is not entirely realistic and, as can be demon-

strated by a relatively simple argument, leads to an underestimate of

the critical Rayleigh Number. The magnitude of the error involved

cannot be estimated within the framework of the approach taken.

Turning to the other extreme of wall condition, it is evident

that the critical Rayleigh number of such blocks will be overesti-

mated by taking the walls to be perfectly conducting to heat. Assum-

ing this condition, we obtain an upper bound to the critical number.

Such data are therefore complementary to those of Beck (1972) and

Zebib and Kassoy (1977). The real critical Rayleigh number for

finitely conducting walls will be bounded by the two extremes. It is

therefore of considerable interest to provide data on the upper bound.

In the thesis, we will elaborate on this subject by carrying

through computations of the critical functions for very long vertical or

semi-vertical porous Darcy type slabs where we assume that the bounding walls are impermeable to the fluid but perfectly conducting to heat. In view of the geometry assumed, the flow and temperature fields will be assumed to be two-dimensional. To enhance the applicability of the results, several angles of tilt will be assumed for the slabs and the flow condition at the top bounding plane will be varied.

We will further generalize our work by investigating the implications of non-Boussinesq flows and temperature dependent fluid properties. A few specific cases of varying permeability will also be taken up for consideration.

Finally, a specific case of a slab embedded in rock having a realistic finite heat conductivity will be investigated.

It is of interest to note that the present work is an outgrowth of a joint research project of the Geophysics group at Oregon State University and the Department of Geophysical Sciences at Georgia Institute of Technology, Atlanta, Ga. The principal scope of this project was to investigate convective phenomena in geothermal systems. Both the OSU and GIT project were supported by the U.S. Geological Survey Extramural Geothermal Research Program. Dr. Robert P. Lowell served as the Principal Investigator at GIT.

II. NUMERICAL ANALYSIS

Various approaches to the mathematical formulation of describing the behavior of heat and mass transfer in fluid saturated porous materials have been given (see, for example, Lapwood, 1948; Wooding, 1957; Beck, 1972; Bear, 1972; Garg et al., 1975). Under the present working frame the common basic assumptions are,

1) the porous medium is isotropic;

2) the fluid is in liquid phase;

3) the flow is in the saturated laminar range;

4) no heat sources or sinks exist in the field;

5) heat conduction is assumed to occur in both the liquid and the solid material. Further, they are assumed to be in local (pointwise) equilibrium.

Under these conditions, the flow of heat and mass in the porous material may be described by the following set of equations.

Governing Equations

Flow Equation

The evolutionary form of Darcy's Law similar to that used by Wooding (1959) and Schowalter (1965) is taken to be

$$\rho_f[\frac{\partial \vec{V}}{\phi \partial t} + \frac{\vec{V}}{\phi} \cdot \nabla (\frac{\vec{V}}{\phi})] = -\nabla P - \frac{\mu}{K} \vec{V} + \rho_f \vec{g} \tag{2.1}$$

where $\vec{V}$ is the macroscopic average fluid velocity field, P the fluid pressure, ρ_f the fluid density, μ the fluid viscosity, K the permeability of the medium, ϕ the area porosity of the medium and $\vec{g}$ is the acceleration of gravity. The true (pore) fluid velocity is equal to $\vec{V}/\phi$. The inclusion of the intertia terms on the left hand side in Darcy's law is required to describe nonstationary flows. In the case of slowly varying laminar flows in porous media, these terms are generally negligible. Bear (1972) has discussed the range of validity of Darcy's law, and Bodvarsson (1970) has concluded that the inertia terms are of significance only in fractured rock with relatively wide openings (a few centimeters).

Continuity Equation

For a compressible fluid, we can write the continuity equation as

$$\frac{\partial (\phi \rho_f)}{\partial t} = -\nabla \cdot (\rho_f \vec{V}) \tag{2.2}$$

Energy Equation

Neglecting the viscous dissipation and applying the energy balance equation developed by Bird and others (1960, p. 313 (10.1-9)) we

obtain the following energy equation for the combined fluid and solid phases

$$[\phi\rho_f C_f + (1-\phi)\rho_s C_s] \frac{\partial T}{\partial t} - \alpha T \frac{DP}{Dt} = k_m \nabla^2 T - \rho_f C_f \vec{V} \cdot \nabla T \qquad (2.3)$$

where $\dfrac{DP}{Dt} = \dfrac{\partial P}{\partial t} + \vec{V} \cdot \nabla P$ is the rate of enthalpy increased by compression, ρ_s the density of the solid material, α the fluid thermal expansivity, k_m the thermal conductivity of the saturated medium, C_f the fluid specific heat at constant pressure, and C_s is the specific heat of the solid phase. For simplicity we take k_m

$$k_m = \phi k_f + (1-\phi)k_s \qquad (2.4)$$

where k_f is the thermal conductivity of the fluid and k_s the thermal conductivity of solid phase. In applying (2.4), we assume that the heat conduction through the fluid and solid phases is parallel but separate (Lagarde, 1965). In some cases the effective conductivity is slightly higher than the values derived from (2.4) because of the thermal dispersion. A discussion of this parameter based on experimental work has been given by Green (1963).

Equation of State

The physical properties of water are functions of both temperature and pressure. Relevant data are usually given in the form of

algebraic relations. The equations derived by Meyer et al. (1968) employed in this thesis are too complex to be quoted here. A subprogram PROPT (Appendix C) has been worked out to generate the density, thermal expansivity, compressibility, specific heat at constant pressure and viscosity of water for given temperature and pressure.

Galerkin Finite-Element Method

Our problem is to solve a couple set of nonlinear partial differential equations. In general, solutions of the equations cannot be derived in an analytical form, and consequently numerical techniques have to be resorted to. Furthermore, the problem setting is further complicating by spatial variations in the material properties such as permeability and conductivity. Moreover, irregular geometries and boundary conditions are often encountered. The selection of the Galerkin finite-element technique for solving our convective problem is based on the consideration of these difficulties.

Although the finite-element approach is often found to be more flexible than finite-difference methods the variational formulation is generally rather difficult. The Galerkin method offers an alternative way of formulating problems for finite element solutions without using direct variational principles. For many physical problems the methods of Galerkin and Ritz often lead to similar approximating

equations. The Galerkin method is, however, more universal and is applicable to equations of the elliptic, parabolic, and hyperbolic types. Kantorovich and Krylov (1964) and Forray (1968) present a detailed discussion of the relative merits of the Galerkin and Ritz methods. Douglas and Dupont (1970) give a comprehensive discussion of the extensive use of the methods in the petroleum industry in recent years.

The application of the Galerkin finite-element method to heat and mass transfer problems in geothermal fields was initiated only a few years ago (e.g., Thirriot et al., 1974; Mercer et al., 1975; Faust and Mercer, 1975). They find that this approach is of great value in the modeling of natural geothermal systems. On this method, the whole domain under consideration is divided into irregular sub-divisions as indicated by the physics and geometry of the underlying problem. The size of each subdomain (element) can be varied readily, and the approach yields good approximations of external and internal boundaries. Even inhomogeneties and anisotropicities are quite easily accommodated. Moreover, it is possible to represent coefficients of the partial differential equations which vary in space (e.g., permeability and density) as piece-wise function over each element (Pinder et al., 1973).

The primary disadvantage of the Galerkin method is the need for complicated computer programs and good computer facilities. To

derive efficient computer codes for the Galerkin method is a

formidable task; however, once the development is complete, the

codes may be applied to a wide range of similar problems without

modification. General references on the application of the Galerkin

method to field equations include Zienkiewicz (1971), Pinder and

Frind (1972), Segol et al. (1975), and Pinder and Gray (1977).

General Formulation Procedure

Equation (2.1), (2.2), and (2.3) can be written as

$$L(h) = f \quad \text{in} \quad D \tag{2.5}$$

where D is a bounded domain and the operator L acts on the

unknown field variable h to generate the known function f. To

solve (2.5) by the Galerkin finite-element method, we start out from

a trial solution of the following form

$$\hat{h}(P, t) = \sum_{i=1}^{n} a_i(t) N_i(P), \tag{2.6}$$

where $N_i(P)$ represent n basis (shape) functions of the field

point P forming a complete set for a linear subspace of dimension

n and chosen such that they satisfy the principal boundary conditions

imposed on (2.1), (2.2) and (2.3). The $a_i(t)$ are time-dependent

expansion coefficients to be determined. Substituting (2.6) into (2.5)

yields

$$L\left[\sum_{i=1}^{n} a_i(t)N_i(P)\right] - f = R \qquad (2.7)$$

where the residual R vanishes when the trial solution is an exact

solution. The Galerkin finite-element method is a special case of a

more general approach to the method of weighted residuals. The

basis functions $N_j(P)$ are weighting functions that are selected such

that the residual R is minimized relative to an appropriate norm.

In the present approach, this is equivalent to requiring the ortho-

gonalty of $L\left[\sum_{i=1}^{n} a_i(t)N_i(P)\right] - f$ to the basis function $N_j(P)$, i.e.

$$\int_{D}\left[L\sum_{i=1}^{n} a_i(t)N_i(P)-f\right]N_j(P)da = 0 \qquad (2.8)$$

$$j = 1, 2, \ldots, n$$

Since $N_j(P)$ belongs to a complete for a finite dimensional subspace

(dimension n) it is expected that as $n \rightarrow \infty$, the approximate

solution will tend to an exact one. Assuming that the above integra-

tions can be performed appropriately (2.8) represents a set of n

equations with n unknowns $(a_i(t), i = 1, 2, \ldots n)$. Often there is a

way to lower the order of the space derivatives in equation (2.8) and

to introduce natural boundary conditions on the basis of an integration by parts.

Element Basis Functions

Almost any two-dimensional domain of complex geometry can be divided into appropriate triangular elements. This approach will be used throughout this thesis. There is a trade-off between the number of the elements and the order of the basis functions. The same degree of accuracy can be achieved by reducing the number but increasing the order. In the field of petroleum engineering the emphasis has been on using fewer elements of high-order basis functions in relatively simple geometries while in structural engineering, the tendency has been toward using more elements of low-order basis functions. As shown in Chapter III, the analysis of convection stability does require elements of high-order basis functions for sufficiently fast convergence. Accordingly, a quadratic triangular element (Fig. 1) appears to be a reasonable choice for the problems to be investigated in the present thesis.

In each triangular element, the unknown function is given by the relation:

$$\hat{h} = \vec{N}^e \cdot \vec{a}^e \tag{2.9}$$

where

$$\vec{N}^e = (N_i, N_j, N_k, N_p, N_q, N_r)$$

$$\vec{a}_e^T = (a_i, a_j, a_k, a_p, a_q, a_r)$$

$$N_i = (2L_1^2 - L_1; \quad N_j = 2L_2^2 - L_2; \quad N_k = 2L_k^2 - L_k$$

$$N_p = 4L_1L_2; \quad N_q = 4L_2L_3; \quad N_r = 4L_3L_1$$

$$L_n = \text{local (natural or area) coordinate (see Fig. 1)} \quad n = 1, 2, 3$$

$$L_1 = \frac{1}{2\Delta} \left[(z_j x_k - z_k x_j) + (x_j - x_k)z + (z_k - z_j)x \right]$$

$$L_2 = \frac{1}{2\Delta} \left[(z_k x_i - z_i x_k) + (x_k - x_i)z + (z_i - z_k)x \right]$$

$$L_3 = \frac{1}{2\Delta} \left[(z_i x_j - z_j x_i) + (x_i - x_j)z + (z_j - z_i)x \right]$$

$$L_1 + L_2 + L_3 = 1$$

$\Delta = $ Area of the triangular element.
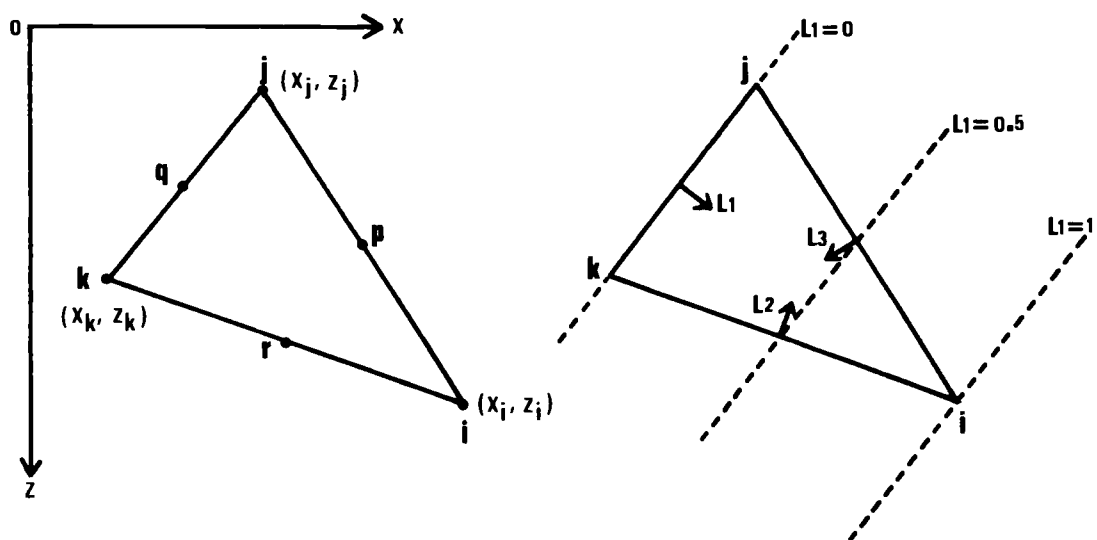
Figure 1. Quadratic triangular element and area coordinate system.

Treatment of Functional Coefficients

One of the most efficient ways of dealing with variable parameters in the governing equations is to assume that they are constant over the elements or vary in a similar manner as the basis functions. We may express these variables as linear functions

$$c_m(x, z) = L_1 c_i + L_2 c_j + L_3 c_k$$

where L_n are the linear triangular basis functions or local (area) coordinates and c_i, c_j, and c_k are coefficients. Of course, it is possible to express the c_m in terms of high-order basis functions (e.g., (2.9)), but this would only complicate the computation. It is a simple matter to incorporate the functional coefficients in conjunction with quadratic triangular elements and numerical integration. The above idea was first discussed by Desai and Abel (1972), and further extended by Pinder et al. (1973). This approach is applicable to both isotropic and anisotropic problems.

Differentiation and Integration

Before carrying out the integrations of the element equations in (2.8), we face the task of evaluating the partial derivatives, $\partial \hat{h}/\partial x$ and $\partial \hat{h}/\partial z$. Since $\hat{h}$ is expressed as a function of the local coordinates L_1, L_2, and L_3 as in (2.6), it is necessary to express

$\partial \hat{h} / \partial x$, $\partial \hat{h} / \partial z$, and dxdz in terms of L_1 and L_2 also. This can be done as follows.

From equation (2.6) we obtain

$$\frac{\partial \hat{h}}{\partial x} = \sum_{m=i}^{r} a_m \frac{\partial N_m}{\partial x} ; \quad \frac{\partial \hat{h}}{\partial z} = \sum_{m=i}^{r} a_m \frac{\partial N_m}{\partial z}$$

and hence we must express $\partial N_m / \partial x$ and $\partial N_m / \partial z$ in terms of L_1 and L_2. By the chain rule of differentiation we obtain

$$\left\{ \begin{array}{c} \dfrac{\partial N_m}{\partial x} \\[3mm] \dfrac{\partial N_m}{\partial z} \end{array} \right\} = [J]^{-1} \left\{ \begin{array}{c} \dfrac{\partial N_m}{\partial L_1} \\[3mm] \dfrac{\partial M_m}{\partial L_2} \end{array} \right\}$$

and similarly

$$dxdz = \det[J] dL_1 dL_2$$

where

$$[J] = \begin{bmatrix} \dfrac{\partial x}{\partial L_1} & \dfrac{\partial z}{\partial L_1} \\[3mm] \dfrac{\partial x}{\partial L_2} & \dfrac{\partial z}{\partial L_2} \end{bmatrix}$$

is the Jacobian matrix. It can be shown that if the mapping is acceptable, that is, one-to-one, then $[J]^{-1}$ exists (Zienkiewicz, 1971, p. 132). On the basis of these transformation integrals

expression such as (2.8) reduce to the form

$$\int_0^1 \int_0^{1-L_2} F(L_1, L_2, L_3) \, \det[J] dL_1 dL_2$$

where F is the transformed function $\left[L \sum_{m=i}^{r} a_m N_m - f^e \right] N_j$ at the

element level. Clearly, the transformed integrand F is not a

simple function that permits closed form integration. Taking into

account the fact that all the dependent variables are simple poly-

nomials numerical integration can, on the other hand, be carried out

to any degree of accuracy.

The accuracy of the numerical integration required to assure

convergence of the finite element method is of considerable impor-

tance. One of the most popular numerical integration schemes for

triangular areas of integration is similar to the Gauss method where

the integrand is evaluated at n discrete points. The integrals are

then calculated on the basis of the relation

$$\int_0^1 \int_0^{1-L_2} F(L_1, L_2, L_3) \det[J] dL_1 dL_2 = \sum_{k=1}^{n} w_k \, \det[J] F(L_1, L_2, L_3)$$

(2.10)

where the w_k are weighting factors which are uniquely associated

with each point k. Making use of n sampling points in relation

(2.10) exact integrals of all polynomial expressions of the form

$\Sigma L_1^\alpha L_2^\beta L_3^\gamma$ where $\alpha, \beta, \gamma \le n-1$ can be obtained. Corresponding weighting factors and stations may be found on the basis of Hammer's formulas (Hammer et al., 1956; Hammer and Stroud, 1958).

Treatment of Temporal Derivatives

Approximation solutions to the time-dependent equations can also be obtained on the basis of the Galerkin approach (e.g., Zienkiewicz and Parekh, 1970; Donea, 1974; Köhler and Pittr, 1974). Numerical experiments have, however, shown that, in general, none of the methods tested perform significantly better than the centered finite-difference procedure. This observation along with the simplicity of the latter method lead to the conclusion that the finite-difference scheme in time is the best overall choice in the majority of cases of time-dependent finite element analysis. It is convenient to combine the procedures by solving the transient equation using the Galerkin method, and then to discretize the time derivative using a finite difference scheme. For the problems at hand the backward difference may provide better results as has been found by Pinder (1973), Mercer (1973), and Segol et al. (1975). The finite-difference discretization of the time derivative is expanded in the following formula

$$\frac{1}{\Delta t} \int_{t}^{t+\Delta t} h(t')dt' \approx \varepsilon h_{t+\Delta t} + (1-\varepsilon)h_{t}$$

where $t \leq t' \leq t+\Delta t$. The case $\varepsilon = 0$ corresponds to an explicit scheme (forward difference scheme), $\varepsilon = 0.5$ is the centered implicit scheme (Crank-Nicholson scheme), and $\varepsilon = 1$ is the implicit scheme (backward difference scheme).

III. LINEARIZED STABILITY CRITERIA FOR NATURAL CONVECTION IN REGIONS BOUNDED BY HORIZONTAL PLANES

Consider a homogeneous and isotropic fluid saturated Darcy type porous system bounded above and below by horizontal planes, place a rectangular coordinate system with the origin in the upper plane and the z-axis vertically down. Hence, $z = 0$ in the upper plane and let the lower plane be at $z = H_z$. We assume that the system is initially at rest and that its initial state is characterized by the stationary temperature field T_0 and the pressure field P_0. Let this pressure field be hydrostatic, that is, $P_0 = \rho_f g z$ where ρ_f is the unperturbed density of the fluid. Moreover, we assume that the temperature in upper plane is constant and equal to T_1 whereas the lower plane has a constant temperature T_2 such that $T_0 = T_1 + (T_2 - T_1)z/H_z$.

Let the system be subjected to small convective perturbations, in the temperature θ, in the pressure p and to a velocity field $\vec{v}$ such that the resulting fields are

$$T = T_0 + \theta$$
$$P = P_0 + p$$

Following standard procedure, we assume that marginal stability is characterized by $\partial/\partial t = 0$ (Chandrasekhar, 1961). Substituting the

above perturbations θ, p, and $\vec{v}$ into (2.1), (2.2), (2.3), and by neglecting second-order terms (linearization) we obtain for marginal stability

$$\nabla \cdot (\rho_f \vec{v}) = 0 \qquad (3.1)$$

$$-\nabla p - (\frac{\mu}{K})\vec{v} + \rho_f' \vec{g} = 0 \qquad (3.2)$$

$$\rho_f C_f \vec{v} \cdot \nabla T_0 - \alpha T_0 \vec{v} \cdot \nabla p = k_m \nabla^2 \theta \qquad (3.3)$$

In the case of two-dimensional convection in the x-z plane where $\vec{v} = (u, w)$, these equations take the form

$$\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} + w(\beta \rho_f g - \alpha D) = 0 \qquad (3.4)$$

$$\frac{\partial p}{\partial x} + \frac{\mu}{K} u = 0 \qquad (3.5)$$

$$\frac{\partial p}{\partial z} + \frac{\mu}{K} w - (\beta \rho_f p - \alpha \rho_f \theta)g = 0 \qquad (3.6)$$

$$k_m \nabla^2 \theta + (\alpha T_0 \rho_f g - \rho_f C_f D)w = 0 \qquad (3.7)$$

where $D = (T_2 - T_1)/H_z$ is the unperturbed temperature gradient. To the first approximation, the parameters ρ_f, α, β, μ and K are assumed to be horizontally constant, but can vary vertically. The heat conductivity k_m is assumed constant. In the derivation of (3.6) we have used a linearized equation of state for the fluid such that the perturbed density is $\rho_f' = \beta \rho_f p - \alpha \rho_f \theta$. Eliminating p from (3.5) and (3.6) we obtain

$$\frac{\mu}{K}\frac{\partial w}{\partial x} + \frac{\mu}{K} g\beta\rho_f u + \alpha\rho_f g \frac{\partial\theta}{\partial x} - \frac{\mu}{K}\frac{\partial u}{\partial z} - u\frac{\partial}{\partial z}\left(\frac{\mu}{K}\right) = 0 \qquad (3.8)$$

Moreover, eliminating $\partial u/\partial x$ from (3.4) by making use of equation (3.8) we obtain

$$\frac{\mu}{K}\nabla^2_{x,z}w + \left[\frac{\mu}{K}(\beta\rho_f g - \alpha D) - g\beta\rho_f\frac{\mu}{K} + \frac{\partial}{\partial z}\left(\frac{\mu}{K}\right)\right]\frac{\partial w}{\partial z}$$

$$+ \left[\frac{\mu}{K}\frac{\partial}{\partial z}(\beta\rho_f g - \alpha D) - \beta^2\rho_f^2 g^2\frac{\mu}{K} + \alpha Dg\beta\rho_f\frac{\mu}{K} + \beta\rho_f g\frac{\partial}{\partial z}\left(\frac{\mu}{K}\right) - \alpha D\frac{\partial}{\partial z}\left(\frac{\mu}{K}\right)\right]w$$

$$+ \alpha\rho_f g\frac{\partial^2\theta}{\partial x^2} = 0 \qquad (3.9)$$

where

$$\nabla^2_{x,z} = \frac{\partial}{\partial x^2} + \frac{\partial}{\partial z^2}.$$

By introducing

$$z_0 = \frac{z}{H_z}, \quad x_0 = \frac{x}{H_z}, \quad \theta_0 = \frac{\theta}{DH_z}, \quad w_0 = \frac{\rho_{fs}C_{fs}H_z}{k_m}R^{-1/2}w$$

$$\rho_{f0} = \frac{\rho_f}{\rho_{fs}}, \quad \alpha_0 = \frac{\alpha}{\alpha_s}, \quad C_{f0} = \frac{C_f}{C_{fs}}, \quad \beta_0 = \frac{\beta}{\beta_s}$$

$$\mu_0 = \frac{\mu}{\mu_s}, \quad K_0 = \frac{k}{K_s}$$

where

$$R = \frac{g\alpha_s DH_z^2\rho_{fs}^2 C_{fs}K_s}{\mu_s k_m}$$

is defined as the Rayleigh number, and the subscript s refers to

the value of the parameter at the surface $z = 0$. For the sake of simplicity the subscript 0 will be dropped from the transformed quantities x_0, z_0, θ_0, w_0 hereafter. With this in mind, we may rewrite equations (3.7) and (3.9) with the dimensionless parameters included

$$R^{1/2} \nabla^2_{x,z} w + R^{1/2} \frac{K_0}{\mu_0} [\frac{\partial}{\partial z} (\frac{\mu_0}{K_0}) - \frac{\alpha_s \alpha_0 \mu_0 D H_z}{K_0}] \frac{\partial w}{\partial z}$$

$$+ R^{1/2} \frac{K_0}{\mu_0} \left\{ H_z \frac{\partial}{\partial z} [\frac{\mu_0}{K_0} (\beta_s \beta_0 \rho_{fs} \rho_{f0} g - \alpha_s \alpha_0 D)] \right.$$

$$\left. - \beta_s \beta_0 \rho_{fs} \rho_{f0} g H_z^2 (\beta_s \beta_0 \rho_{fs} \rho_{f0} g - \alpha_s \alpha_0 D) \frac{\mu_0}{K_0} \right\} w$$

$$+ \frac{\alpha_0 \rho_{f0} K_0}{\mu_0} R \frac{\partial^2 \theta}{\partial x^2} = 0$$

$$\nabla^2_{x,z} \theta + R^{1/2} \frac{\rho_{f0}}{DC_{fs}} (\alpha_s \alpha_0 H_z Dzg + \alpha_s \alpha_0 T_s g - C_{fs} C_{f0} D) w = 0.$$

This can be rewritten in a more compact form

$$\nabla^2_{x,z} w + E(z) \frac{\partial w}{\partial z} + F(z)w + G(z)R^{1/2} \frac{\partial^2 \theta}{\partial x^2} = 0 \qquad (3.10)$$

$$\nabla^2_{x,z} \theta - H(z)R^{1/2} w = 0 \qquad (3.11)$$

where

$$E(z) = \frac{K_0}{\mu_0} [\frac{\partial}{\partial z} (\frac{\mu_0}{K_0}) - \frac{\alpha_0 \alpha_s \mu_0 D H_z}{K_0}] \qquad (3.12)$$

$$F(z) = \frac{K_0}{\mu_0} \left\{ H_z \frac{\partial}{\partial z} \left[\frac{\mu_0}{K_0} (\beta_s \beta_0 \rho_{fs} \rho_{f0} g - \alpha_s \alpha_0 D) \right] \right.$$

$$\left. - \beta_s \beta_0 \rho_{fs} \rho_{f0} g H_z^2 (\beta_s \beta_0 \rho_{fs} \rho_{f0} g - \alpha_s \alpha_0 D) \frac{\mu_0}{K_0} \right\} \quad (3.13)$$

$$G(z) = \frac{\alpha_0 \rho_{f0} K_0}{\mu_0} \quad (3.14)$$

$$H(z) = \frac{\rho_{f0}}{DC_{fs}} (C_{fs} C_{f0} D - \alpha_s \alpha_0 g H_z Dz - \alpha_s \alpha_0 T_s g). \quad (3.15)$$

In the case of the Boussinesq approximation, the terms $E(z)$ and $F(z)$ are equal to zero, and $G(z)$ and $H(z)$ are equal to unity. Equations (3.10) and (3.11) together with given proper boundary conditions constitute an eigenvalue problem for the Rayleigh number R. In marginal stability analysis, we are only interested in the smallest positive eigenvalue of (3.10) and (3.11). This is the critical Rayleigh number Rc which determines the marginal stability of the perturbation and thereby the onset of convection.

Formulation

Following the general procedure of the Galerkin finite-element method discussed in Chapter II, the trial solutions are assumed to be of the form

$$w \approx \sum_{i=1}^{M} N_i w_i \qquad (3.16)$$

$$\theta \approx \sum_{i=1}^{M} N_i \theta_i \qquad (3.17)$$

where M is the total number of nodes in the finite element grid, w_i and θ_i are the finite element solutions for w and θ at the node i, and N_i are the quadratic basis functions defined over the triangles. Functional coefficients $E(z)$, $F(z)$, $G(z)$, and $H(z)$ are approximated by

$$E(z) \approx \sum_{j=1}^{M_c} L_j E_j, \qquad F(z) \approx \sum_{j=1}^{M_c} L_j F_j$$

$$(3.18)$$

$$G(z) \approx \sum_{j=1}^{M_c} L_j G_j, \qquad H(z) \approx \sum_{j=1}^{M_c} L_j H_j$$

where M_c is the number of corner nodes of the finite element grid and L_j are the linear triangular basis functions. Application of the Galerkin procedure on (3.10) and (3.11) with the substitution of (3.16) to (3.18) into (3.10) and (3.11) yields

$$\int_D \left[\nabla^2_{x,z} N^T \{w\} + L^T \{E\} \frac{\partial N^T}{\partial z} \{w\} + L^T \{F\} N^T \{w\} \right.$$

$$\left. + R^{1/2} L^T \{G\} \frac{\partial^2 N^T}{\partial x^2} \{\theta\} \right] N_k \, da = 0$$

$$\int_D [\nabla^2_{x,z} N^T \{\theta\} - R^{1/2} L^T \{H\} N^T \{w\}] N_k \, da = 0 \qquad k = 1, 2, \ldots, M$$

By making use of Green's theorem to remove the second derivatives from the integral expressions we obtain

$$\int_D \left(\frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} + \frac{\partial N_k}{\partial z} \cdot \frac{\partial N^T}{\partial z} \right) da \{w\} - \int_D L^T \{E\} N_k \frac{\partial N^T}{\partial z} da \{w\}$$

$$- \int_D L^T \{F\} N_k N^T da \{w\} + R^{1/2} \int_D L^T \{G\} \left(\frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} \right) da \{\theta\}$$

$$- \int_C N_k \left(\frac{\partial w}{\partial x} \ell_x + \frac{\partial w}{\partial z} \ell_z \right) ds - R^{1/2} \int_C L^T \{G\} N_k \frac{\partial \theta}{\partial x} \ell_x ds = 0 \qquad (3.19)$$

$$\int_D \left(\frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} + \frac{\partial N_k}{\partial z} \cdot \frac{\partial N^T}{\partial z} \right) da \{\theta\} + R^{1/2} \int_D L^T \{H\} N_k N^T da \{w\}$$

$$- \int_C N_k \left(\frac{\partial \theta}{\partial x} \ell_x + \frac{\partial \theta}{\partial z} \ell_z \right) ds = 0 \qquad (3.20)$$

where ℓ_x and ℓ_z are respectively the x and the z components of the unit normal to the boundary, and ds is a differential arc length along the boundary. The surface integral (boundary

residual) in equation (3.19) and (3.20) now enables us to introduce the boundary conditions. These equations with appropriate boundary conditions can be written in matrix form as

$$[A]\{w\} + R^{1/2}[B]\{\theta\} = 0 \qquad (3.21)$$

$$R^{1/2}[C]\{w\} + [D]\{\theta\} = 0 \qquad (3.22)$$

where

$$[A] = \int_D \left\{ \frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} + \frac{\partial N_k}{\partial z} \cdot \frac{\partial N^T}{\partial z} - L^T\{E\}N_k \frac{\partial N^T}{\partial z} - L^T\{F\}N_k N^T \right\} da$$

$$- \int_C N_k \left(\frac{\partial w}{\partial x} \ell_x + \frac{\partial w}{\partial z} \ell_z \right) ds$$

$$[B] = \int_D L^T\{G\} \left(\frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} \right) da - \int_C L^T\{G\}N_k \frac{\partial \theta}{\partial x} \ell_x ds$$

$$[C] = \int_D L^T\{H\}N_k N^T da$$

$$[D] = \int_D \left(\frac{\partial N_k}{\partial x} \cdot \frac{\partial N^T}{\partial x} + \frac{\partial N_k}{\partial z} \cdot \frac{\partial N^T}{\partial z} \right) da - \int_C N_k \left(\frac{\partial \theta}{\partial x} \ell_x + \frac{\partial \theta}{\partial z} \ell_z \right) ds$$

Equations (3.21) and (3.22) have non-trivial solutions if and only if

$$\det \begin{bmatrix} [A] & R^{1/2}[B] \\ [C]R^{1/2} & [D] \end{bmatrix} = 0$$

which can be further reduced to

$$\left|\,[D]^{-1}[C][A]^{-1}[B] - \frac{1}{R}\,[I]\,\right| = 0$$

This is an eigenvalue problem of a M x M matrix. The maximum value taken by the reciprocal eigenvalue 1/R of the system (3.21) and (3.22) represents the critical condition for the onset of convection. Matrices are formed in our program BAFORM and then solved for eigenvalues by program ORGAN. For the purpose of investigating convection instability problems, the program RAYLEI (including several developed subprograms) has the capability of dealing with various field conditions. Details of coding with the aid of the many supplementary statements are given throughout the program lists (Appendix C). Some of the important features include the following:

1) Two dimensional irregular geometry are subdivided into many triangular elements with quadratic basis functions.

2) The material properties or the coefficients of the partial differential equations are allowed to be continuous or discontinuous within each element.

3) Many types of realistic boundary conditions such as conducting boundaries and totally impermeable or permeable boundaries can be taken into consideration.

4) Coefficients of the matrices are stored in a band mode.

5) A subprogram GRID is available to generate elements and node numbers for slab geometries with different tilt angles.

The evaluation of the element equations in program BAFORM involves a large number of arithmetic calculations. These operations are listed in Appendix C.

An Example of Comparison with Existing Analytic Solutions

In order to demonstrate the validity of the above formulation, we will below compare numerical results obtained on the basis of our method to analytical results obtained for models with insulated walls developed by Sutton (1969) and Beck (1972). For the present purpose, we will limit Beck's three-dimensional model to the special two-dimensional case shown in Figure 2. The model involves a rectangular region in $-\infty < y < \infty$ filled with porous material and bounded by impermeable and insulated walls.
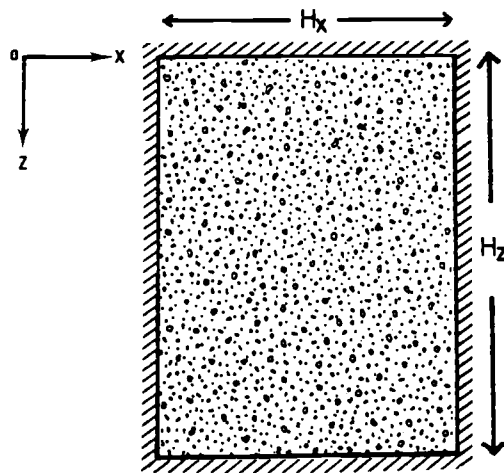
Figure 2. Schematic representation of a slab, Hx and Hz are in units of length.

In the Boussinesq approximation, the governing equations (3. 10)

and (3. 11) simplify to

$$\nabla^2_{x, z} w + R^{1/2} \frac{\partial^2 \theta}{\partial x^2} = 0 \qquad (3.23)$$

$$\nabla^2_{x, z} \theta - R^{1/2} w = 0 \qquad (3.24)$$

with the boundary conditions

$$u = 0 \quad \text{on} \quad x = 0, \ Hx/Hz$$

$$w = 0 \quad \text{on} \quad z = 0, \ 1$$

$$\frac{\partial \theta}{\partial x} = 0 \quad \text{on} \quad x = 0, \ Hx/Hz$$

$$\theta = 0 \quad \text{on} \quad z = 0, \ 1 \qquad (3.25)$$

The analytic solutions (Sutton, 1969; Beck, 1972) for the temperature

field and critical Rayleigh numbers Rc are given as

$$\theta = \frac{Hz}{\pi Hz} \cos \frac{\pi}{2} (1 + \frac{2Hzx}{Hx}) \cdot \sin \pi z$$

$$Rc = \pi^2 [(\frac{mHz}{Hx}) + (\frac{Hx}{mHz})]^2$$

where m = integeral part of $(1/2 + \frac{1}{2}\sqrt{1 + 4(Hx/Hz)}^2)$.

Figure 3 and 4 show the results of our numerical calculations

in comparison with the exact values obtained by Sutton (1969) and Beck

(1972). The results demonstrate a monotonic convergence of the

approximate critical Rayleigh number and field functions with mesh
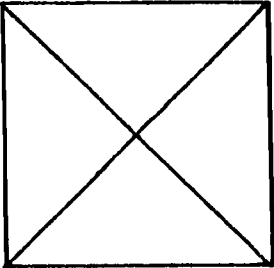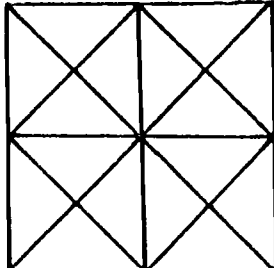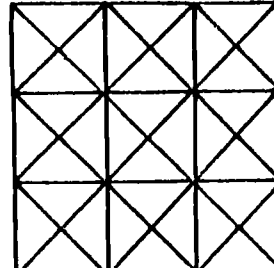
Mesh configuration	No. of elements	No. of nodes	Hx/Hz	Rc (exact)	Rc (finite element)	Error
			1	39.48	40.00	1.32%
	4	13	0.5	61.69	62.50	1.30%
			0.1	1006.80	1020.10	1.32%
			1	39.48	39.72	0.60%
	16	41	0.5	61.69	62.01	0.60%
			0.1	1006.80	1012.30	0.55%
			1	39.48	39.52	0.10%
	36	85	0.5	61.69	61.77	0.13%
			0.1	1006.80	1008.00	0.12%

Figure 3. Convergence of the critical Rayleigh number Rc versus the number of elements. Comparison between finite element and exact values of Rc for the box model in Figure 2.
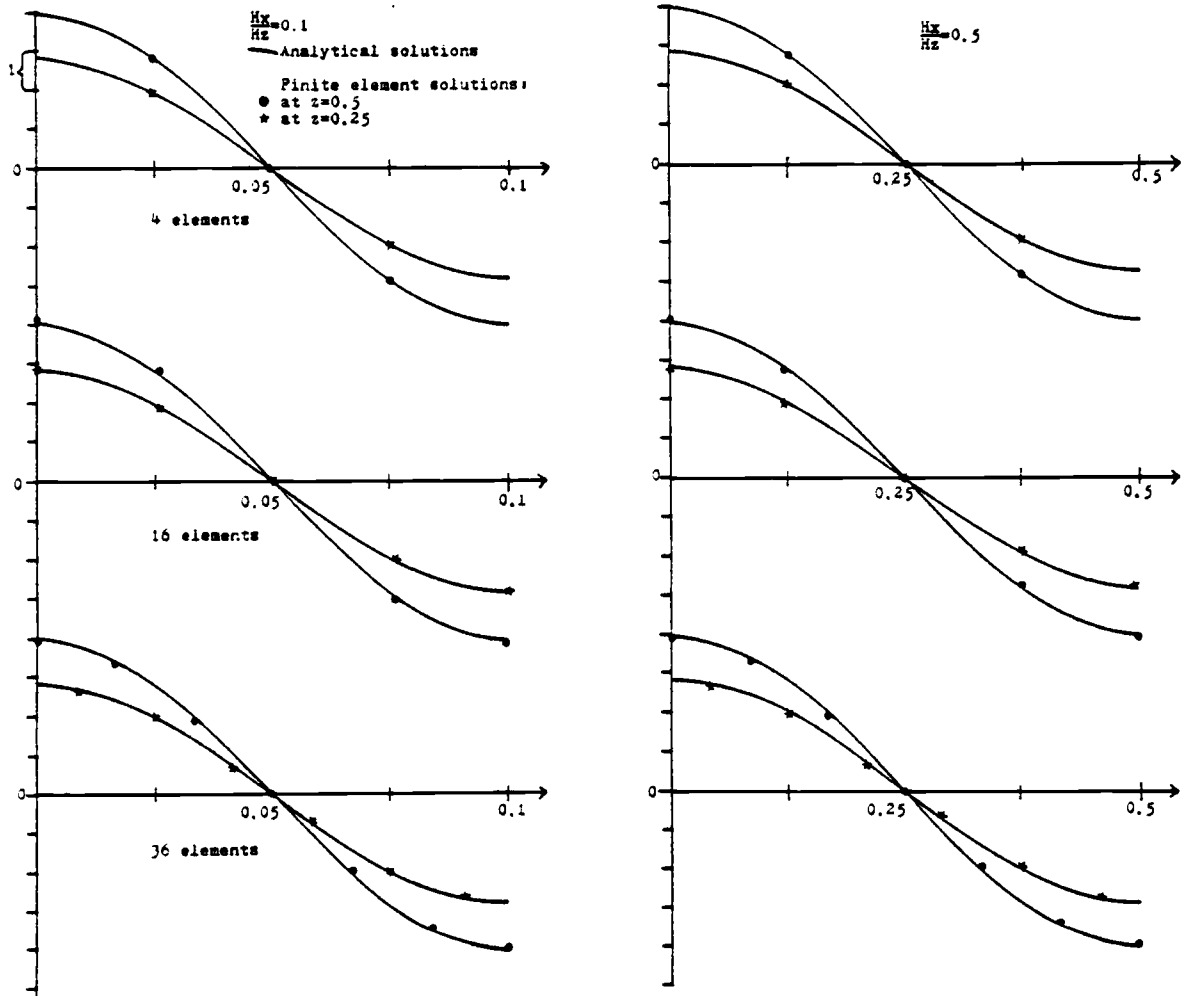
Figure 4. Comparison of the analytical solutions and the finite element solutions for the box model in Figure 2.

size. Even when only 4 elements are used, the numerical temperature results at the nodes agree within about 1% of the exact values.

Infinite Slab Models with Conducting Walls

As discussed in the introduction, we will now turn our attention to the basic task of the present research, that is the investigation of the critical field functions for models with perfectly conducting walls. Moreover, we will also investigate models with variable permeability, temperature and pressure dependent fluid properties and the implications of non-Boussinesq conditions will also be considered.

To carry out the computational work, the dimensions of the specific models under investigation have to be chosen with regard to applicability to realistic conditions. In our selection of slab dimensions, we have had the geometry of fault blocks in mind, in particular, the field conditions in the Basin and Range Province. There is a considerable amount of geothermal activity in this region, and most of it appears to be controlled by the master faults which are so conspicuous at the horst and graben structures. Although our numerical results are derived for models of specific dimensions, the data obtained can be applied to geometrically similar models of other dimensions with the help of simple dimensional analysis.

Conducting/Impermeable Boundaries in the Boussinesq Approximation

As shown in Figure 5 we commence by considering an infinitely long slab with a slanted parallelogram cross section in the x-z plane where the walls (1) and (2) make the angle ϕ with the horizontal. Considering only the 2-dimensional flow case in the x-z plane and assuming all boundaries are impermeable and perfectly conducting, the boundary conditions for the perturbation fields become

$$\theta = 0 \text{ on walls 1 and 2}$$

$$\theta = 0 \text{ on } z = 0, 1$$

$$w = 0 \text{ on } z = 0, 1$$

and from the flow condition on walls 1 and 2 (see Appendix A)

$$\frac{\partial w}{\partial x} = -G(z)R^{1/2}\frac{\partial \theta}{\partial x} - [I(z)w + \frac{\partial w}{\partial z}]\cot \phi$$

where

$$I(z) = \frac{K_0}{\mu_0}\frac{\partial}{\partial z}(\frac{\mu_0}{K_0}) - g\beta_s\beta_0\rho_{fs}\rho_{f0}Hz$$

All parameters are assumed constant except ρ_f' in (3.2) which is taken to be $\rho_f' = \rho_f[1-\alpha(T-T_s)]$.

Turning now to vertical slabs, with $\phi = 90°$ and using the Galerkin finite-element method, the set of equations (3.23) and (3.24) has been solved with the above boundary conditions. A comparison of

the eigenvalues derived for different aspect ratios Hx/Hz with those obtained by Sutton (1969) and Beck (1972) for the same type of flow but with insulated walls is presented in Table 1 and in Figure 6.
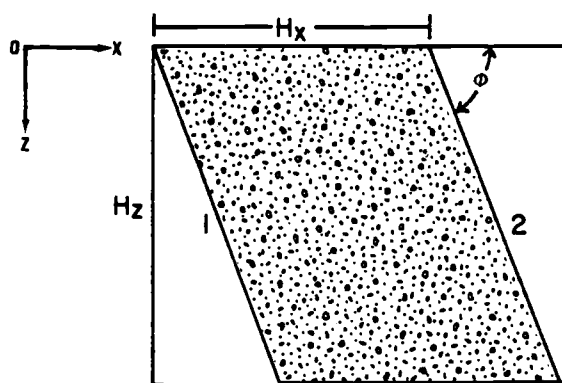
Figure 5. Tilted porous slab.

Table 1. Critical Rayleigh number at various aspect ratios for the onset of convection in infinitely long slabs (y-direction) with impermeable tops.

Hx/Hz	1.00	0.75	0.50	0.25
Conducting walls	80.10	109.40	199.20	681.40
Insulated walls	39.48	42.80	61.70	178.30
	0.10	0.05	0.01	
Conducting walls	4014.60	15336.60	297014.00	
Insulated walls	1006.80	3967.60	98715.70	
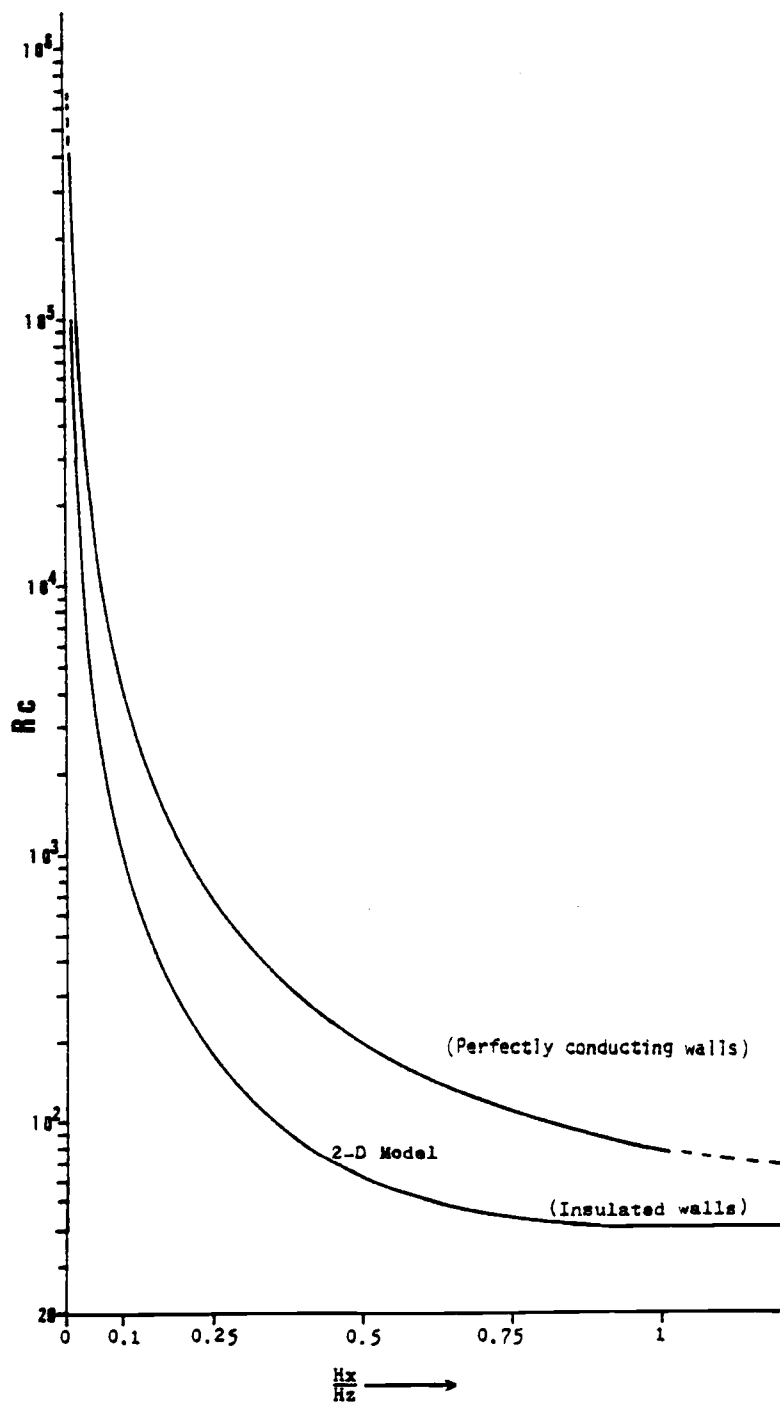
Figure 6. Comparison of critical Rayleigh numbers for vertical
           slabs at various aspect ratios and perfectly conducting
           walls with those of a similar model with insulated
           walls.

For a 3-dimensional model with insulated walls and extending over the entire y-axis, Beck (1972) derived for the limiting case of $H_y \rightarrow \infty$ a critical Rayleigh number of 29.48.[1] This value is considerably lower than the value calculated by Lowell and Shyu (1978) for the same model geometry with conducting walls. They came to the conclusion that for Hx/Hz $\leq$ 1 the critical Rayleigh number approaches to the above values for 2-dimensional models (Table 1 and Fig. 6). This suggests that the 2-dimensional circulation pattern is preferred when Hx/Hz $\leq$ 1. Even if the circulating motion in the case of the model with insulated walls is restricted to the x-z plane the critical Rayleigh numbers as derived by Sutton (1969) are still lower by a factor of 2 to 4 when compared to our present values. When Hx/Hz < 2, the heat transferred across the conducting walls becomes more important. Consequently, the onset of convection requires higher critical Rayleigh numbers. The dashed lines in Figure 6 implies that as the aspect ratio increases the effects of the conducting walls become less important and the critical Rayleigh number tends to the value 39.48 obtained by Lapwood (1948) for horizontal slabs of infinite extent.

Figure 7 illustrates the computed isotherms for cases of both insulated and conducting walls at a fixed aspect ratio Hx/Hz = 1.

---

[1] In the case of 3-d model with insulated walls the critical Rayleigh number $\rightarrow$ 39.48 as $H_y \rightarrow \infty$.

Rc = 39.48, $\frac{Hx}{Hz}$ = 1.
insulated walls
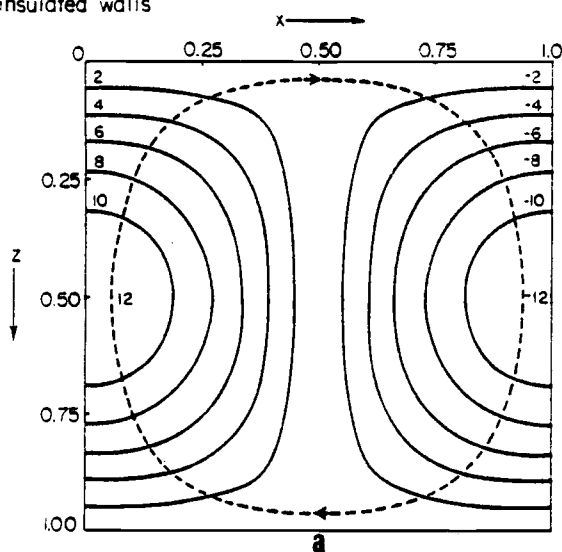
R = 83.5, $\frac{Hx}{Hz}$ = 1.
conducting walls
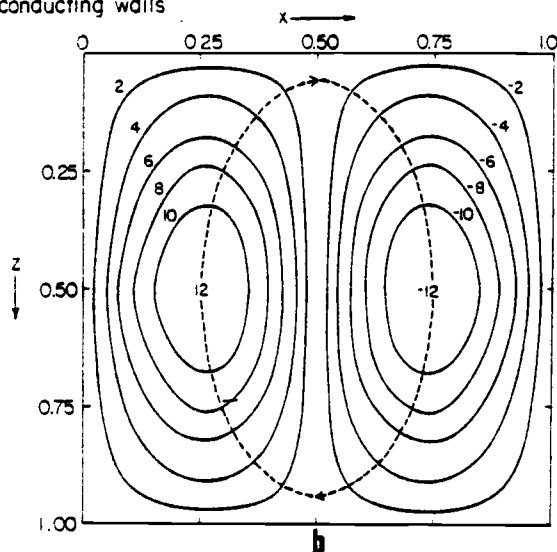
Rc = 80.1, $\frac{Hx}{Hz}$ = 1.
conducting walls
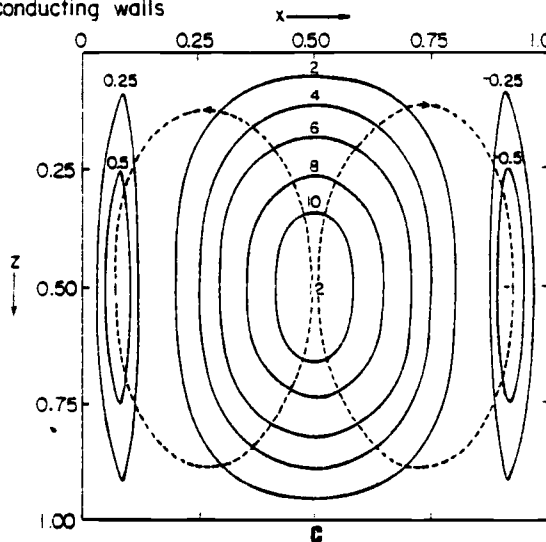
Figure 7. Critical convective patterns in vertical slabs as
affected by boundary conditions and Rayleigh numbers.
With R = 83.5, case b has a central convective cell
whereas at Rc = 80.1 case c has a double convective cell
with centrally ascending fluid. Solid contours are iso-
therms. Dashed lines are contours of convection cells.
R = Rayleigh number, Rc = critical Rayleigh number.

The signatures in Figure 7 will be used for the representation of results throughout this thesis. Isotherms are shown with solid lines contours of convection cells with dashed lines and the values of θ and the dip angle ϕ are indicated. For convenience the scale of the field amplitudes of the isotherms has been chosen such that the maximum amplitude is 12. Figure 7 indicates how the isotherms depend on the wall boundary conditions. The isotherms in Figure 7-a show clearly that no heat is transferred across the side walls, while in the case of conducting walls in Figure 7-b, heat is absorbed by one side wall and released by the other.

It is of interest to note that the asymetric single-node flow field obtained for conducting walls at a Rayleigh number of 83.1 and shown in Figure 7-b was derived on the basis of 2 x 4 x 4 = 32 computational elements. The symmetric two-node flow field obtained at a slightly lower Rayleigh number of 80.1 was, on the other hand, derived with the help of a more symmetric scheme of 3 x 4 x 3 = 36 computational elements. Analog results for a different aspect ratio are shown in Figure 8. Obviously, there is some inconsistency since we would expect the simpler mode to correspond to a lower Rayleigh number. We believe that we are here confronted with an artifact resulting from a "numerical resonance" between the computational scheme and the basic modes of the underlying model. The more symmetric computational scheme generates the symmetric mode
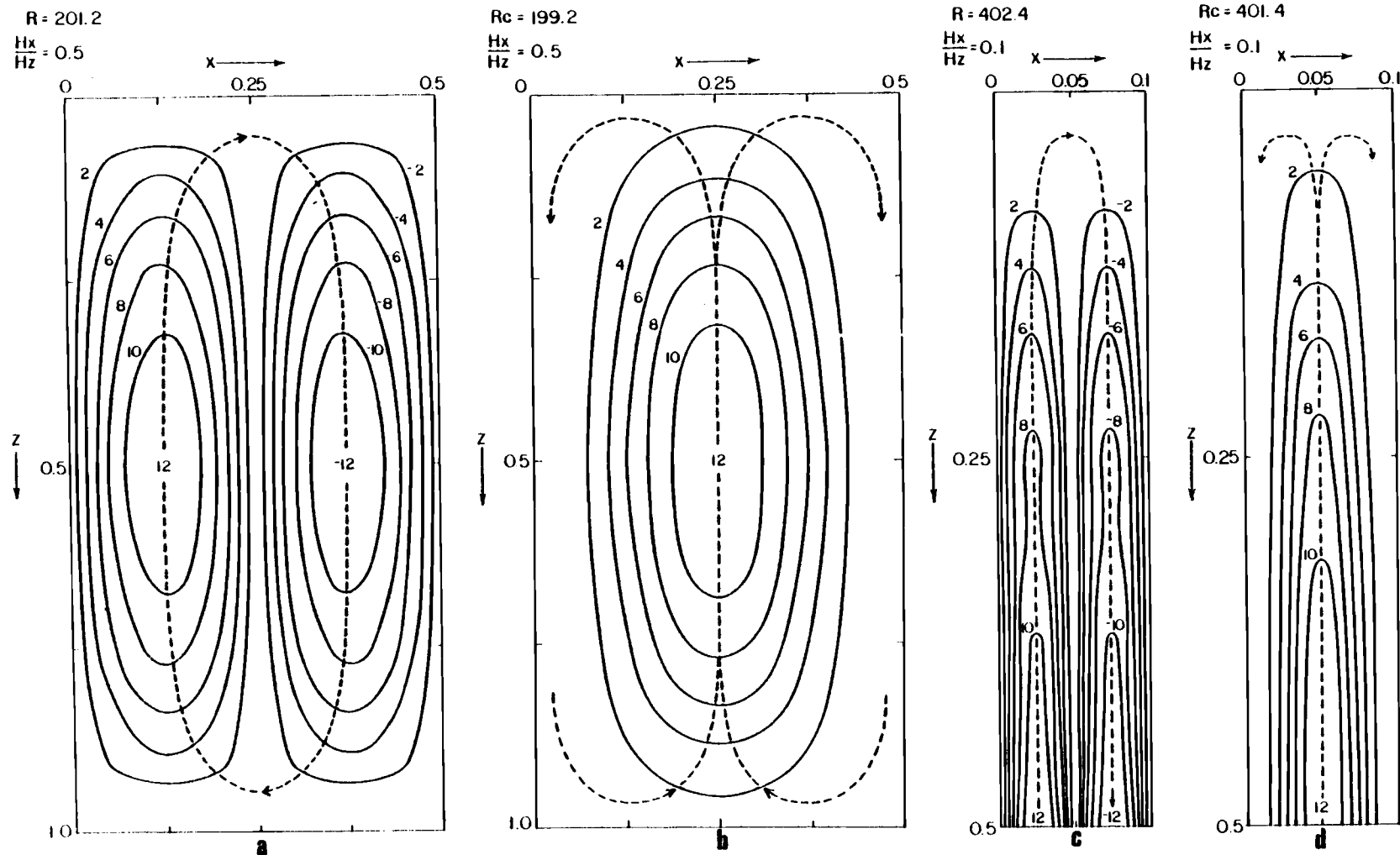
Figure 8. Critical isotherms and convection cells at various aspect ratios and Rayleigh numbers for infinite vertical slabs with impermeable tops. Due to scale limitation only the upper half of the convective patterns are shown in c and d where the aspect ratio Hx/Hz = 0.1.

rather than the more basic asymmetric mode. Introducing some

asymmetry into the model such as a slight tilt of the slab but still

using 36 computational elements results in the suppression of the

symmetric two-node flow field solutions and only the single-node

asymmetric field with $Rc \approx 80.1$ (see later in Table 2) is obtained.

In the following, we will therefore make the basic assumption that the

single-node flow field and the associated Rayleigh number represent

the critical situation.

Tilted Slabs with Conducting Walls and Impermeable Tops

Our next step is to investigate the effects of different dip angles

on the critical Rayleigh numbers and convection cell forms in

infinitely long slabs of the type investigated above. Using the same

computational procedure as above, our results for the critical

Rayleigh numbers at dip angles from $\phi = 90°$ to $\phi = 60°$ are

tabulated in Table 2 and their general trend with respect to decreas-

ing ϕ is sketched in Figure 9. The critical Raleigh number Rc

is not found to change much (within 10 percent) for $70° \leq \phi \leq 90°$.

This suggests that in estimating critical Rayleigh numbers, vertical

slab models can in many cases be used to approximate slightly tilted

structures. It is interesting to note as shown in Figure 9 that for

$Hx/Hz \leq 0.10$, the critical number decreases with the dip angle.

Therefore, a narrow $(Hx/Hz < 0.25)$ tilted slab may in many

Table 2. Critical Rayleigh number for the onset of convection in infinitely long slabs with impermeable top for various aspect ratios and tilt angles.

ϕ	Hx/Hz						
	1.00	0.75	0.50	0.25	0.10	0.05	0.01
90°	80.10	109.40	199.20	657.10	4014.60	15336.00	397014.00
85°	81.10	110.40	203.40	668.30	4019.60	15405.00	392051.00
80°	81.30	111.80	205.00	674.70	4078.40	15351.00	387997.00
75°	81.50	113.10	210.00	678.80	4017.80	15198.00	384706.00
70°	81.80	114.90	216.20	681.50	3973.00	14959.00	381456.00
65°	82.50	117.20	222.80	684.70	3894.20	14684.00	374781.00
60°	83.90	120.20	230.60	690.50	3812.50	14393.50	367996.00
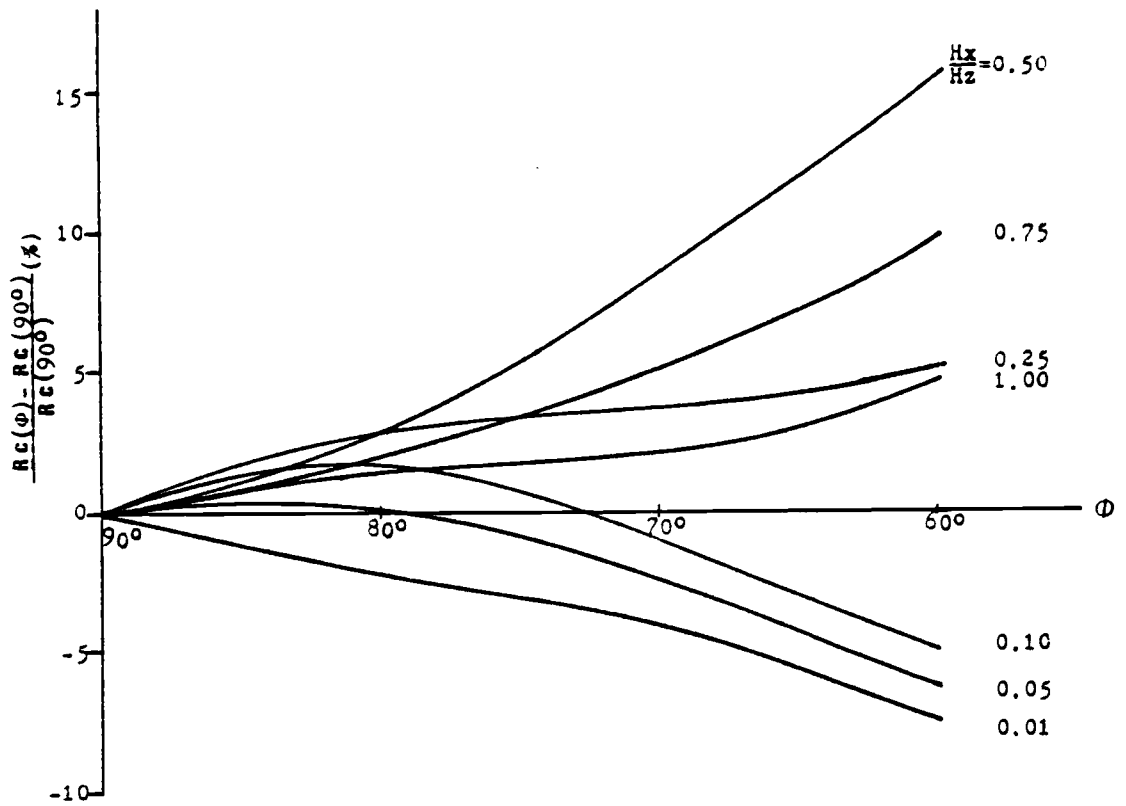
Figure 9. Variations of the critical Rayleigh number with the
tilt angle ϕ.

cases be more unstable than a vertical one with the same aspect ratio.

Numerical results on critical isotherms and vertical flow vectors in tilted infinite slabs are given in Figures 10 to 13. Again, the normalization of flow and temperature amplitude are arbitrary. The isotherms and vertical flow fields in these figures indicate a tendency of forming a main vertical cell in the central portion of the slab accompanied by two local subcells near the corners. As already stated, it is of interest to note that all critical flow solutions for tilted slabs turn out to be the asymmetric type.

Vertical Slabs with Conducting Walls and Permeable Tops

Flow or recharge through a permeable upper boundary may occur when either a standing liquid or a second porous medium (with a much larger permeability) overlies the porous layer of interest. The appropriate boundary condition is then one of constant pressure (Lapwood, 1948) which implies that there is no viscous interaction across the boundary $z = 0$. By considering the continuity equation we found for this case that, $\partial p/\partial x = 0$, $\partial p/\partial z = 0$, and therefore $\partial w/\partial z = 0$ at $z = 0$. The boundary conditions in (3.25) has to be replaced by $\partial w/\partial z = 0$ at $z = 0$ and $w = 0$ at $z = 1$.

Proceeding as in the previous section we can calculate the critical functions for models with a permeable top surface. The results given in Table 3 and Figure 14 show that a permeable top
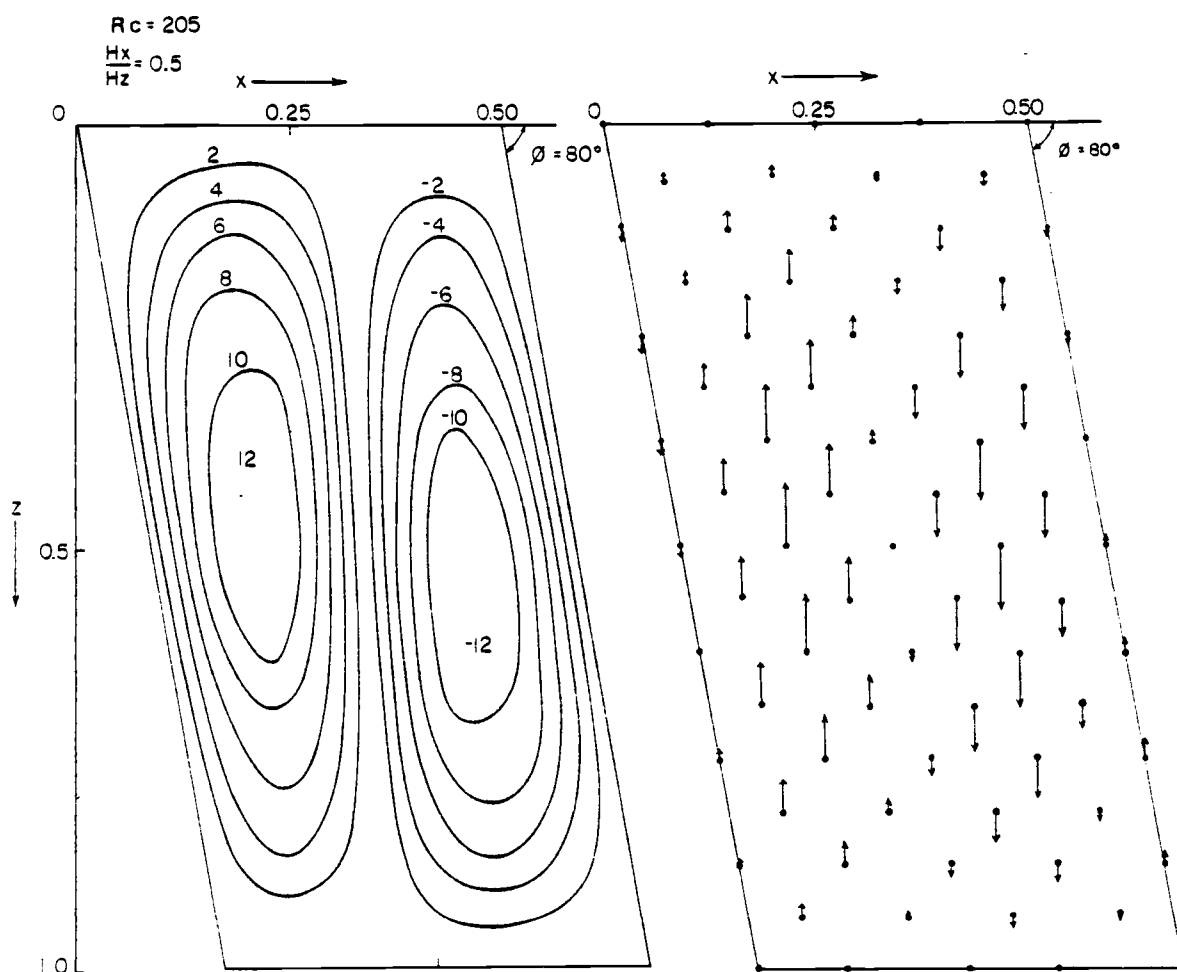
Figure 10. Critical isotherms and vertical flow vectors at an
aspect ratio Hx/Hz = 0.5, and a tile angle φ = 80°
--assuming an impermeable top surface.
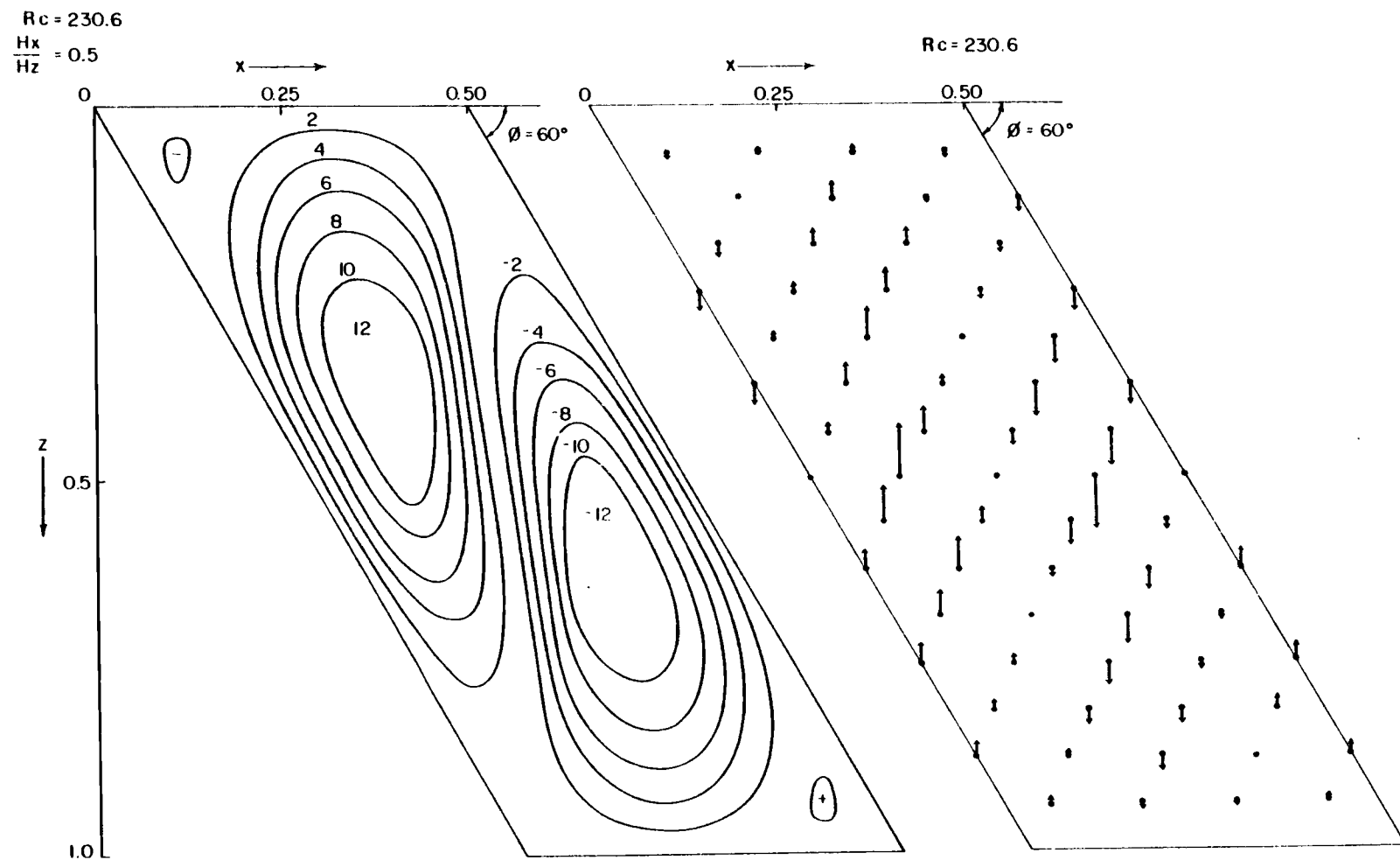
Figure 11. Critical isotherms and vertical flow vectors at an aspect ratio Hx/Hz = 0.5, and a tilt angle $\phi = 60°$ --assuming an impermeable top surface. Note the two positive and negative local temperature extremums near the corners. Their relative amplitude is very small.

Rc = 4078.4
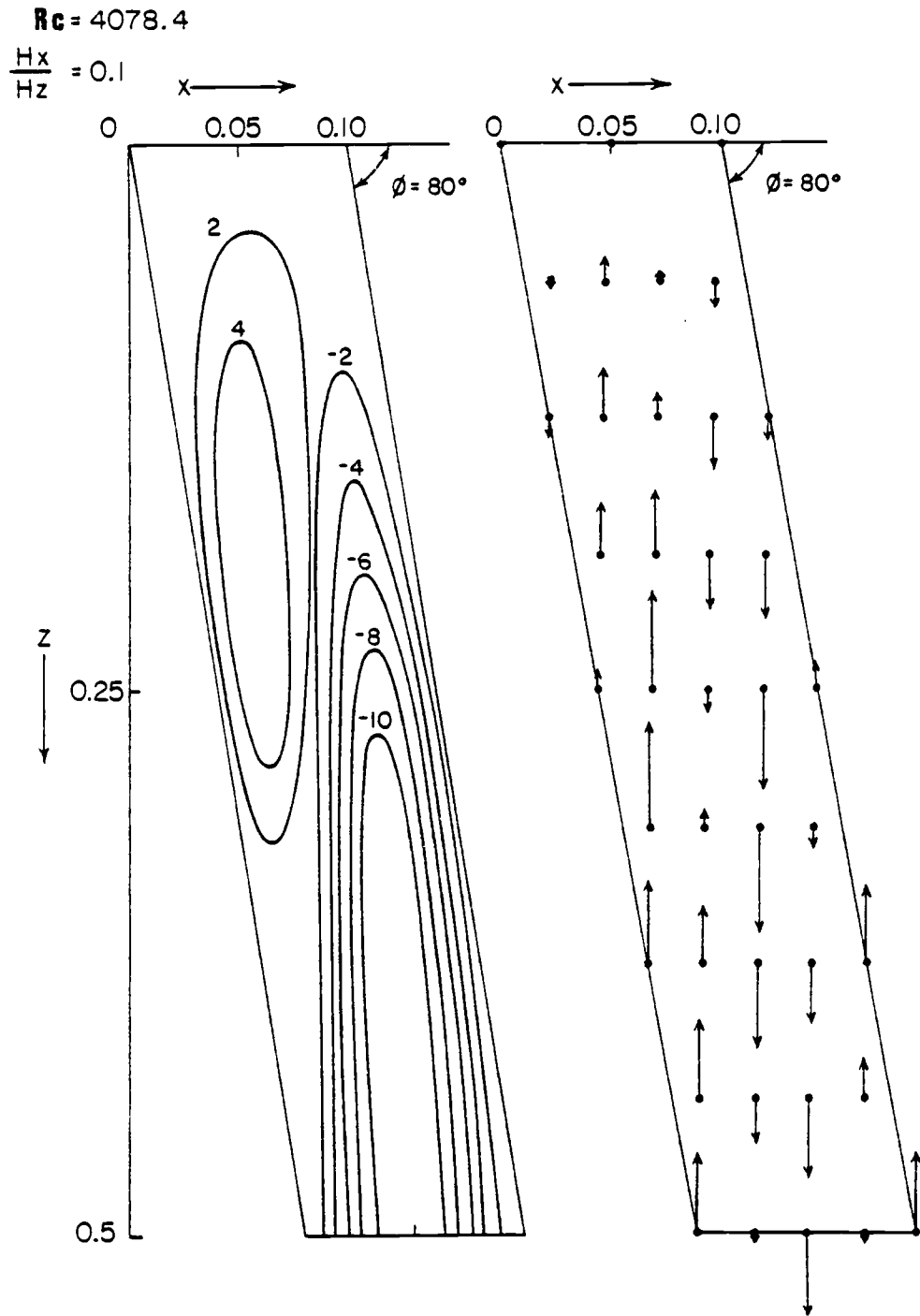
$\frac{Hx}{Hz} = 0.1$

Figure 12. Critical isotherms and vertical flow vectors at an
aspect ratio Hx/Hz = 0.1, and a tilt angle φ = 80°--
assuming an impermeable top surface. For con-
venience only the upper half of the critical fields is
shown.
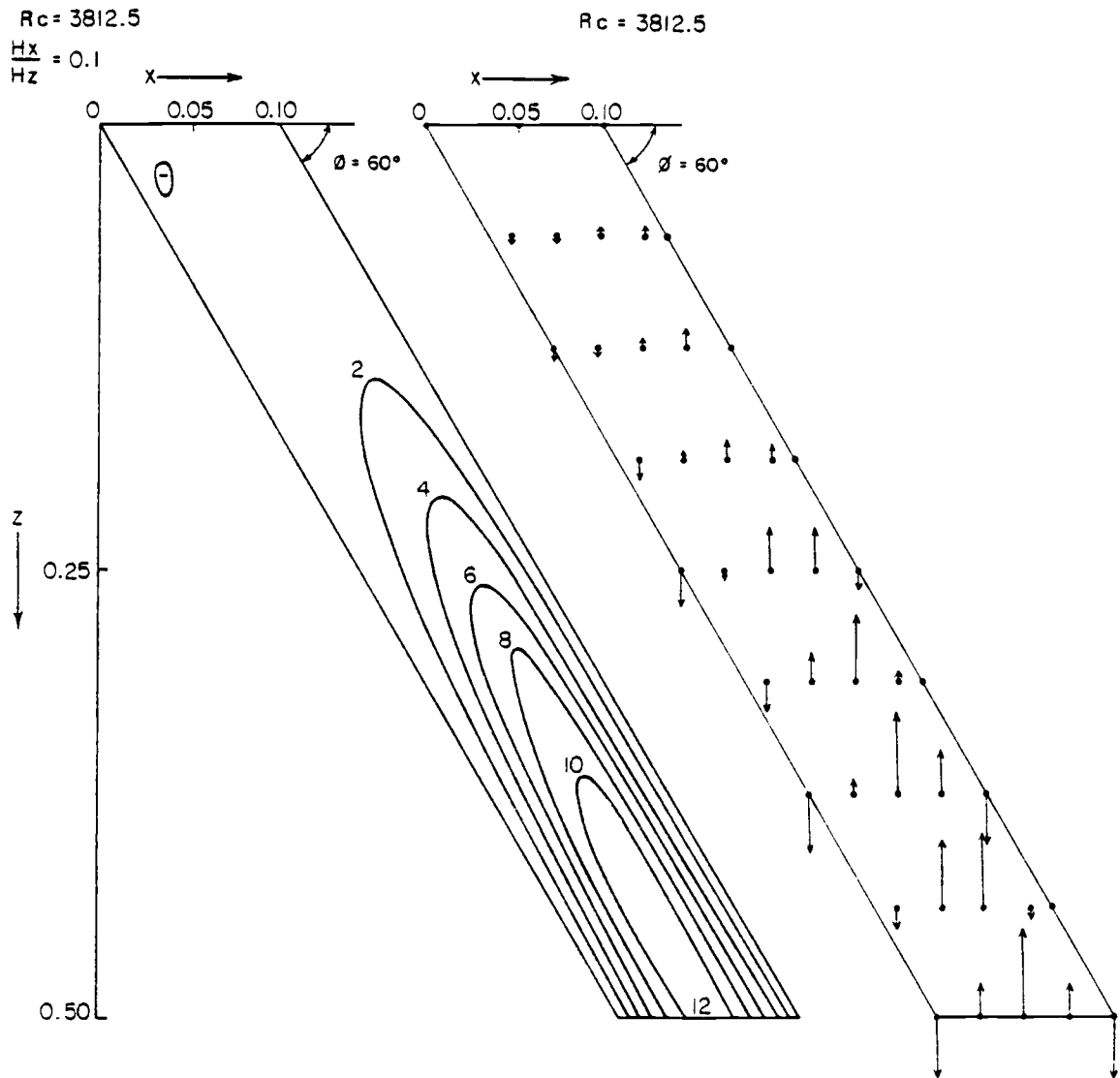
Figure 13. Critical isotherms and vertical flow vectors at an aspect ratio Hx/Hz = 0. 1, and a tilt angle φ = 60°-- assuming an impermeable top surface. For convenience only the upper half of the critical fields is shown.

Figure 14. Comparison of critical Rayleigh numbers for permeable and impermeable tops.

leads to a reduced value of the critical Rayleigh number. The differ-
ence depends highly on the aspect ratio. For example, when
$Hx/Hz = 1$, the critical number is reduced to 1/3 of the value for
the case of an impermeable top. However, at $Hx/Hz = 0.5$, this
difference becomes negligibly small. In other words, the critical
conditions are independent of the top condition when $Hx/Hz < 0.5$.

Table 3. Critical Rayleigh number for the onset of convection for
various aspect ratios--assuming permeable top surface.

Hx/Hz	1.00	0.75	0.50	0.25	0.10	0.05	0.01
Rc	66.40	100.20	189.00	654.80	4010.00	15438.00	297000.00

The isotherms are, however, significantly affected by the top
boundary condition. The vertical symmetry of the impermeable top
case is lost. In the permeable top case the double cell (see Figure
15-a) is shifted slightly upward (asymmetry about the lint $z = 0.5$).
This phenomenon can be understood with the help of the flow pattern
shown in Figure 15-b. Since the ascending and decending plumes can
cross the boundary, their flow pattern is less restricted and the
velocity therefore higher than in the case of an impermeable top.
This effect intensifies the positive and negative isotherms around the
top boundary. Although the critical Rayleigh number is little
affected by the top boundary condition when $Hx/Hz < 0.5$, the
circulation pattern may be basically different.
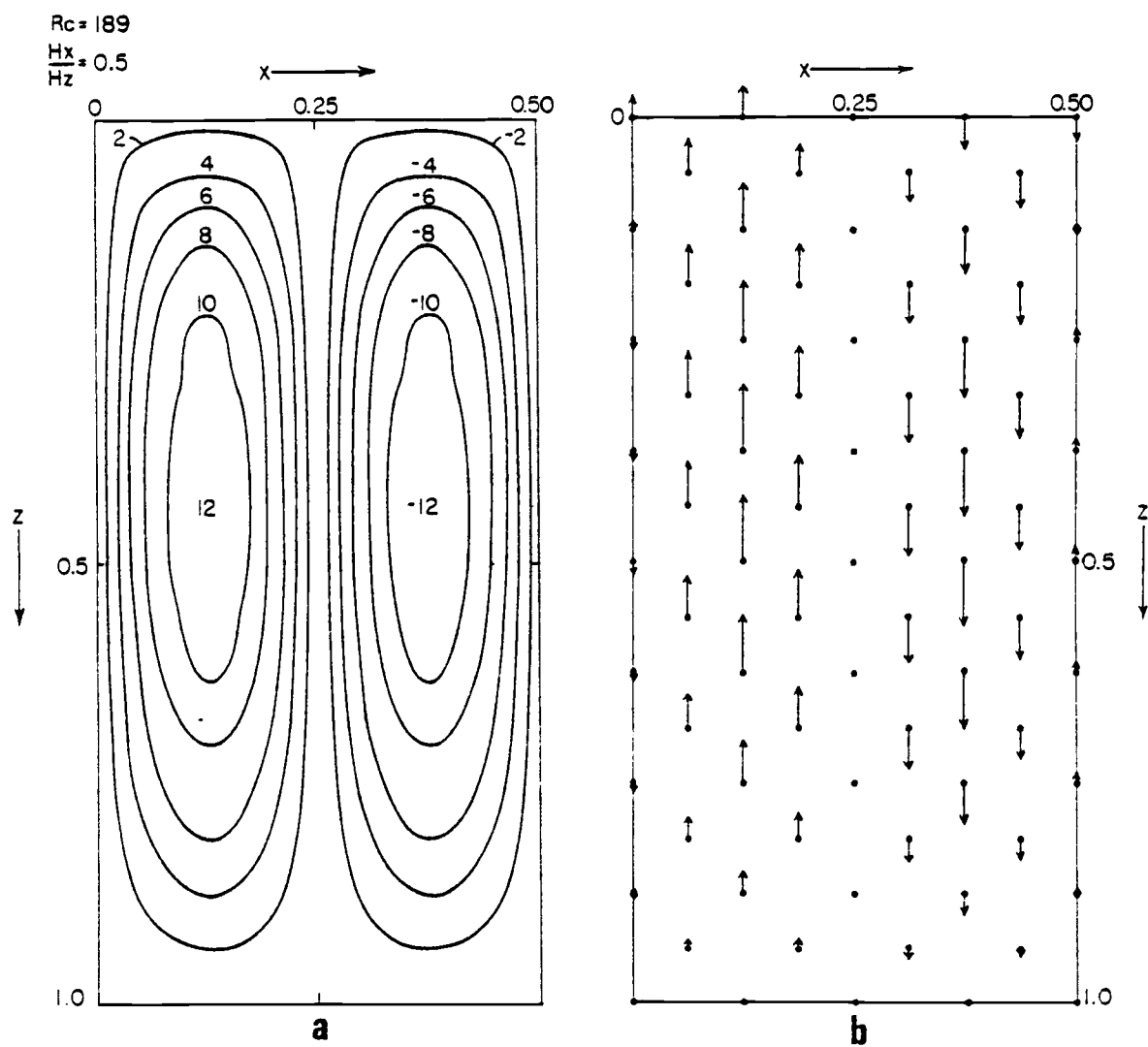
Figure 15. Critical isotherms and vertical flow vectors at an
aspect ratio of Hx/Hz = 0.5--assuming a permeable top
surface.

Tilted Slabs with Permeable Tops

Results of the computation of critical Rayleigh numbers for tilted slabs with permeable tops are given in Table 4 and critical flow patterns are shown in Figures 16 and 17. Taking a glance at the values in Table 4, two facts emerge. First, the values are not significantly smaller than in the case of impermeable tops, particularly in the range $Hx/Hz < 0.5$. Secondly, a sharp change in the magnitude of the critical Rayleigh number may occur at some dip angles. To illustrate this, the critical isotherms and vertical velocity patterns for normal slabs and those with a sharp change in the critical Rayleigh number are compared in Figure 16 and 17. The figures show that the abnormal behavior of the critical Rayleigh numbers is associated with the emergence of a three-cell pattern. A relatively large increase in the critical Rayleigh number is associated with the jump from double to triple cell convection. In general, a change in the topology of the cell pattern is associated with an abrupt change in the critical Rayleigh number.

Although, the criteria for the set up of multi-cell (> 2) convection is an interesting subject, we will not discuss it here. Particularly interesting cases can be explored separately with our program RAYLEI. At this stage, we would like to emphasize the fact emerging from these result, that in the case of permeable tops the

cell pattern is more dependent on the aspect ratio and the tilt angle

than in the case of impermeable tops.

Table 4. Critical Rayleigh numbers for the onset of convection at
various aspect ratios and tilt angles for slabs with a
permeable top surface.

ϕ	Hx/Hz						
	1.00	0.75	0.50	0.25	0.10	0.05	0.01
90°	66.40	100.20	189.00	654.80	4010.00	15438.00	397000.00
80°	66.10	97.00	192.40	674.20	4076.80	15508.00	408827.00
70°	67.23	101.50	208.10	816.00	3943.30	15240.00	378946.00
60°	72.40	114.60	309.10	645.90	3782.00	14682.00	364261.00
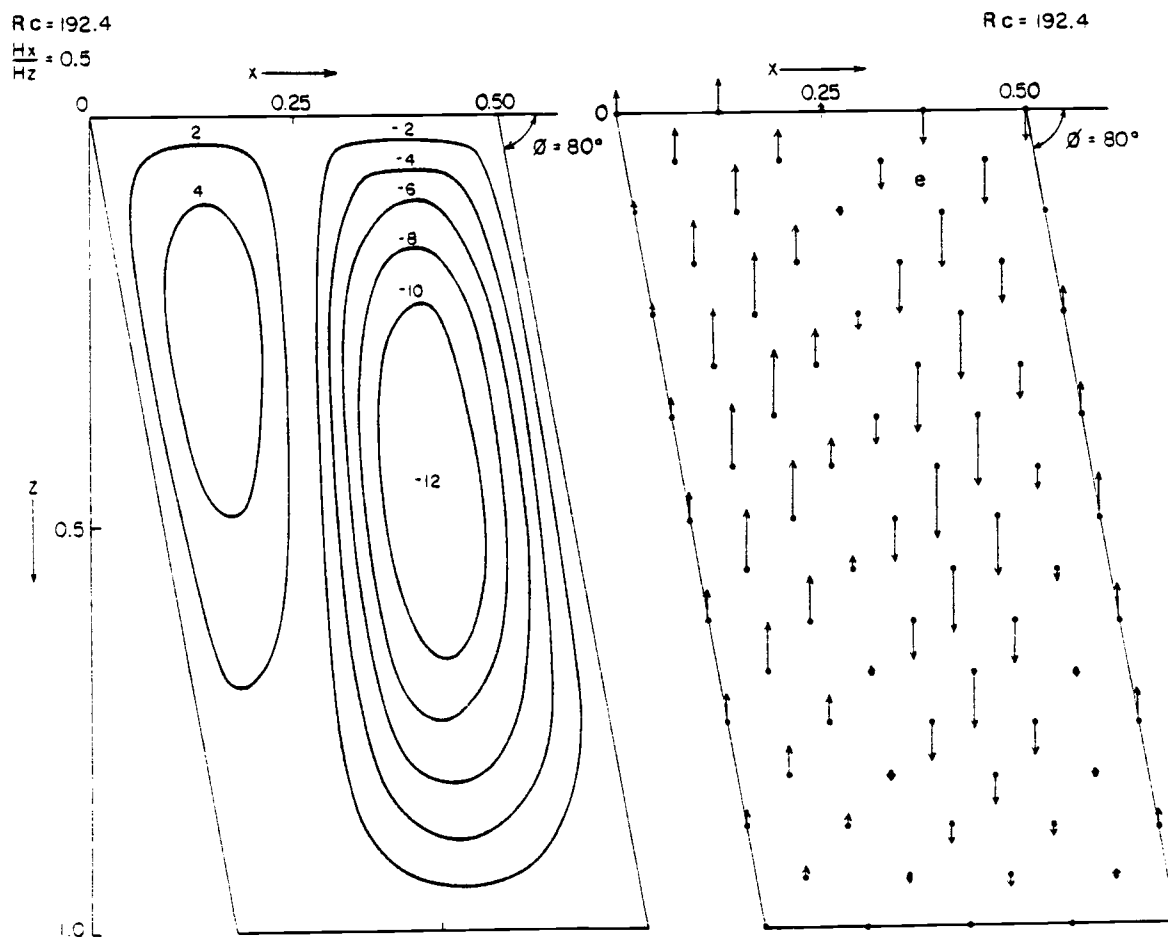
Figure 16. Critical isotherms and vertical flow vectors at an
aspect ratio of Hx/Hz = 0.5 and a tilt angle φ = 80°--
assuming a permeable top surface.
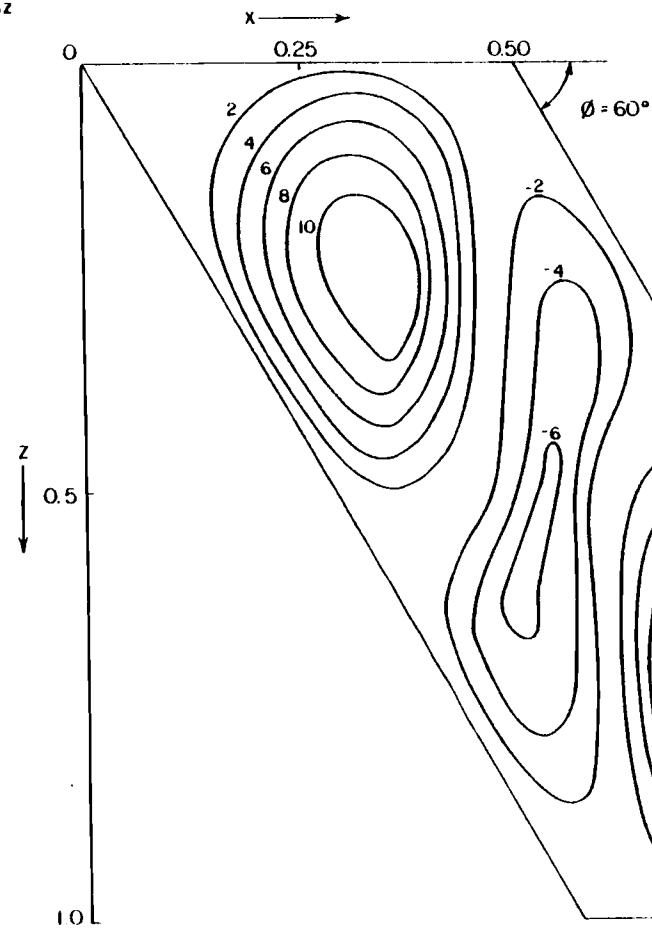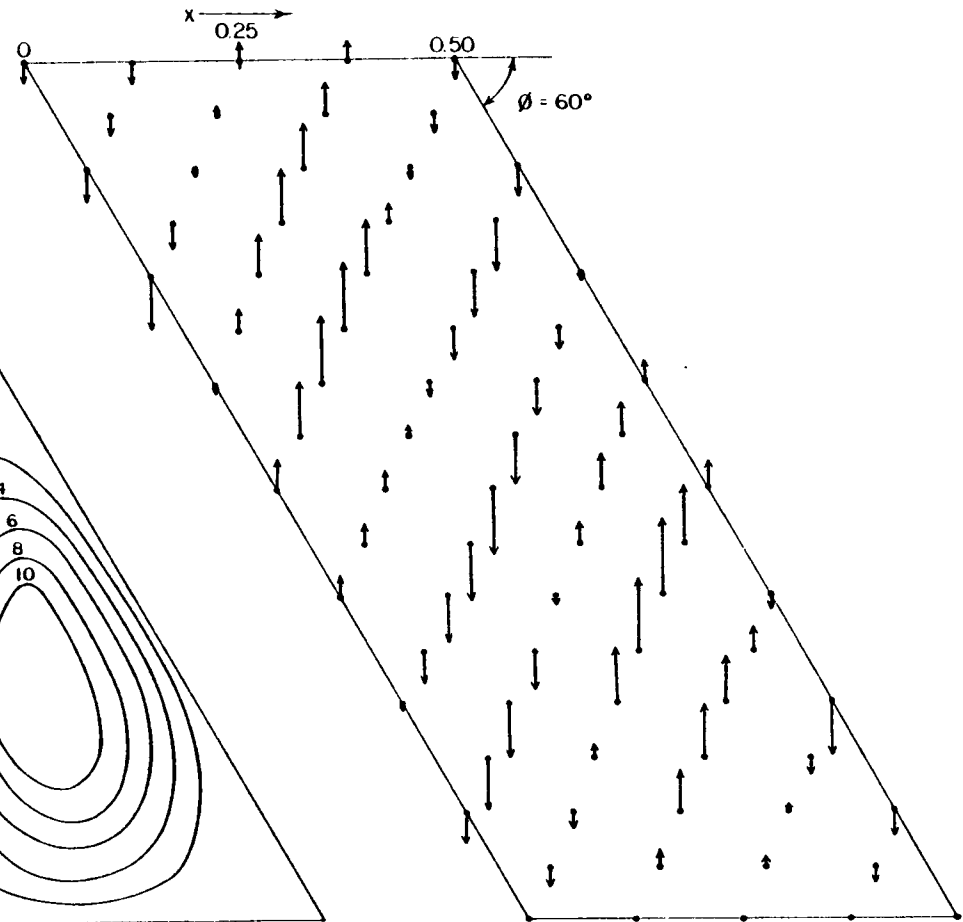
Figure 17. Critical isotherms and vertical flow vectors at an aspect ratio of Hx/Hz = 0.5 and a tilt angle of φ = 60°--assuming a permeable top surface.

IV. EFFECTS OF TEMPERATURE- AND PRESSURE-DEPENDENT THERMODYNAMICAL AND TRANSPORT PROPERTIES OF THE FLUID

Kassoy and Zebib (1975) have emphasized the importance of variable viscosity for the onset of free convection in porous media. The implications of variable thermodynamical and transport properties of water as well as of non-Boussinesq effects have been investigated by Straus and Schubert (1977). Their analysis is, however, limited to infinite horizontal layers with impermeable top and bottom boundaries. In this section we will employ our computational technique to study the effects of variable properties and non-Boussinesq effects on the critical Rayleigh numbers and flow modes for two-dimensional convection in infinite slabs of the same type as in our above models. Data on the thermodynamic properties of water are given in Appendix B.

With regard to the non-Boussinesq effects we have instead of setting $E(z) = F(z) = 0$ and $G(z) = H(z) = 1$ to solve the whole set of equations (3.10) to (3.15). The expressions $E(z)$, $F(z)$, $G(z)$, and $H(z)$ are then assumed to be known parameter functions to be applied in the solving of equations (3.10) and (3.11). A new computational program PROPT and a subprogram ROAP (Appendix C) have been developed to evaluate the parameter functions. No additional computational effort will be needed to take into account variations of

the material properties over individual element areas because during numerical integration the value of the parameter functions at the stations for integration are available on the basis of a one-dimensional linear interpolation between neighboring corner nodes. The introduction of the additional parameter terms in the integral equations increases the order of the highest order polynomials to be integrated from four to five (the unknown parameters θ and w are approximated by quadratic basis functions). Fortunately, fourth- and fifth-order polynomials require the same number of stations for integration, so that no additional work is required in the integration based on functional parameter representation.

Non-Boussinesq Slab Models with Conducting and Impermeable Boundaries

Assuming a surface temperature $T_s = 25\,°C$ and a surface pressure $P_s = 1$ bar, the stability problem formulated above has been solved using the shooting method. Because of the temperature dependent properties, we have in this case to prescribe the temperature gradient and geometry from which criticallity is to develop. In our analysis of a water saturated infinite slab with impermeable but conducting boundaries we assume the temperature gradients to be $D = 30,\ 50$ and $70\,°C/km$, depths $H_z \leq 5$ km and aspect ratios $H_x/H_z = 0.5,\ 0.25,\ 0.1$ and 0.01. The results are given below in

terms of a relative Rayleigh number which is the ratio between the actual Rayleigh number and the corresponding number for a model with constant properties equal to the state in the top surface. The relative Rayleigh number is displayed in Figure 18 as a function of Hz for 4 different aspect ratios with D as a parameter.

Clearly, the onset of convection in a water-saturated porous slab is substantially influenced by the variable properties and non-Boussinesq effects. As noted by Straus and Schubert (1977) for the case of an infinite horizontal porous layer, the water-saturated porous slab is considerably more unstable to thermal convection than an equivalent porous slab saturated with an ideal fluid having spatially constant thermodynamic properties. The results shown in Figures 18 and 19 indicate how the relative number r decreases with D and Hz when the other quantities are held fixed. To take an example, consider a slab of aspect ratio Hx/Hz = 0.25 with a moderate temperature gradient D = 30°C/km and being 1 km deep. This case leads to r = 0.5 (Figure 18). Furthermore, in the case of a slab with an aspect ratio Hx/Hz = 0.01, a temperature gradient D = 50°C/km, and depth of 5 km, the relative number r is reduced to only about 0.033.

It is interesting to consider the effects of variable fluid properties on the relative Rayleigh number as Hx/Hz tends to very small values assuming a fixed temperature gradient D. One may consider
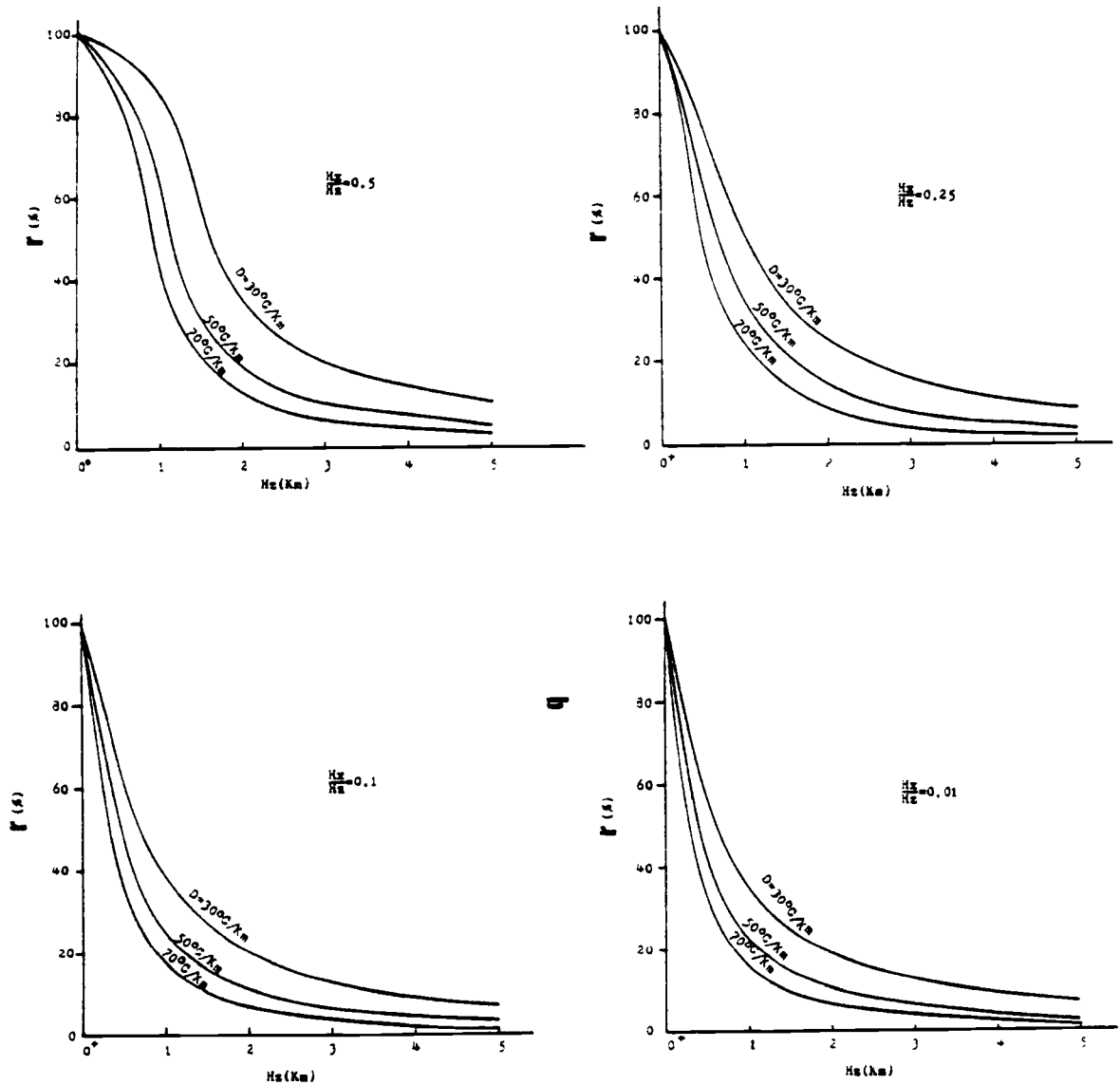
Figure 18. The relative Rayleigh number r as a function of the vertical dimension Hz and the initial temperature gradient D for two-dimensional convection in a vertical slab.

$$r = \frac{(\text{actual Rayleigh number})}{(\text{Rayleigh number for a model with constant properties})} \ .$$
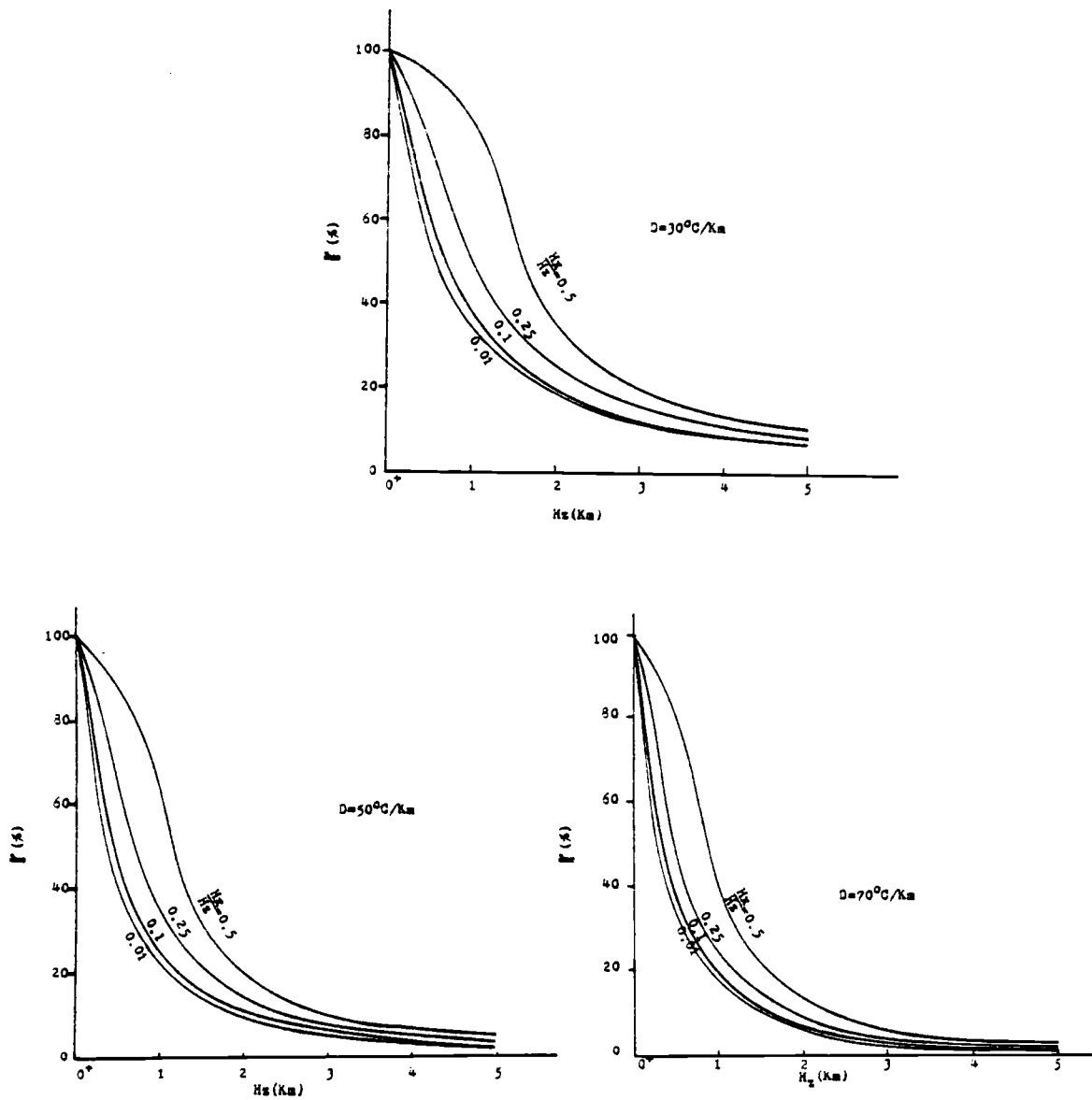
Figure 19. The relative Rayleigh number r (defined in Figure 18) as a function of the vertical dimension Hz with Hx/Hz as a parameter.

Hx/Hz as a parameter and arrive thereby at the graphs shown in Figure 19. An interesting fact is that the relative Rayleigh number decreases with the aspect ratio Hx/Hz until the latter reaches a value of 0.1. Below this value there is little change.

To elaborate on the above results, typical critical isotherms and vertical flow fields for infinite slabs with impermeable boundaries are shown in Figures 20 and 21. In contrast to our earlier results obtained with the Boussinesq approximation (e.g. the isotherms shown in Figure 8, which are symmetric about $z = 0.5 Hz$) the present results show a relatively high density of isotherms near the lower boundary. This implies that the hot fluid near the bottom boundary moves relatively fast in comparison with the cooler fluid near the top boundary. The associated vertical velocity fields given in Figures 20 and 21 provide further evidence for this phenomenon. A close inspection of the two figures indicates that as a result of the effects of temperature and pressure on the fluid properties only about 60 to 70% of the volume of the porous slabs is relatively strongly involved in the convection. The relative size of the active volume decreases with the aspect ratio.
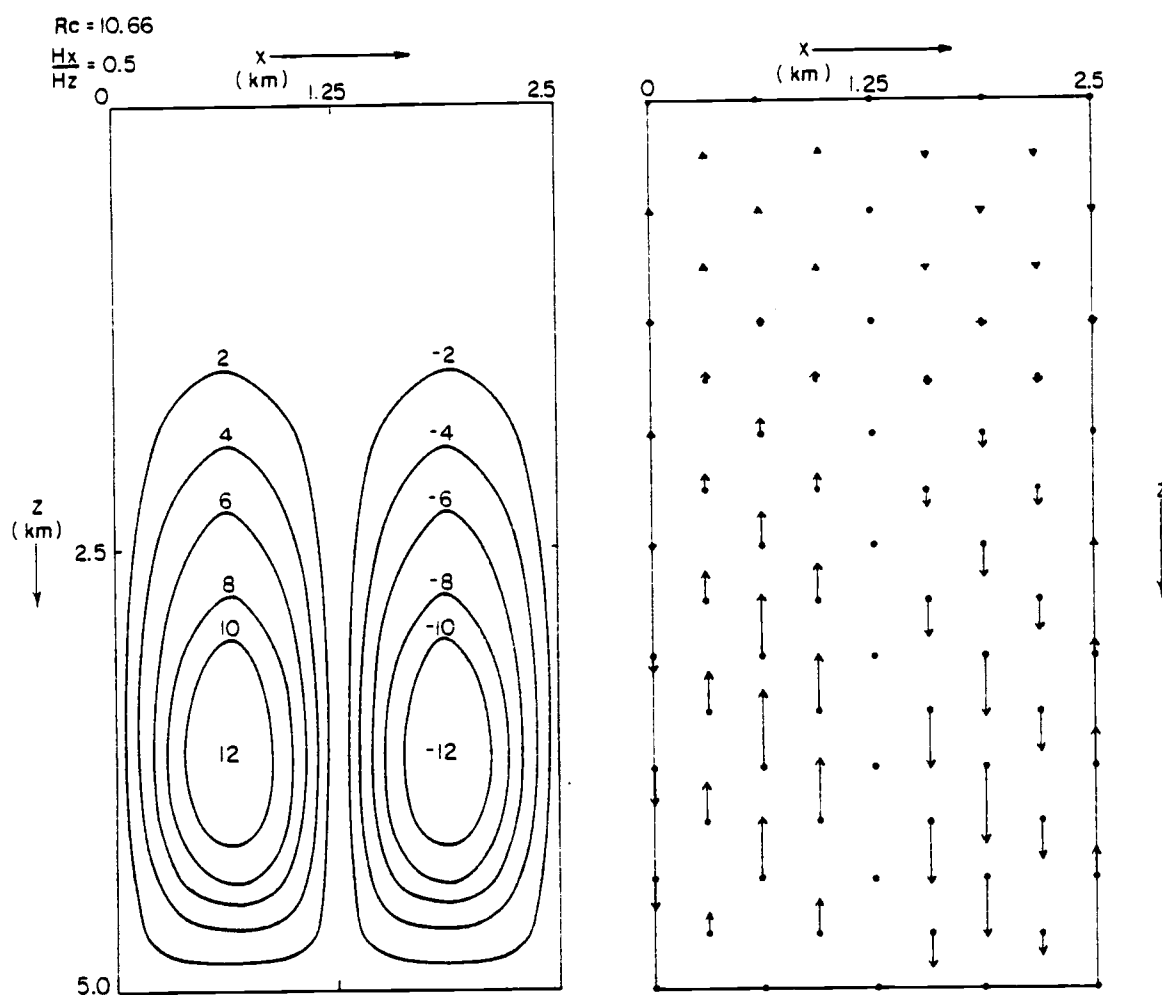
Figure 20. Critical isotherms and vertical flow vectors for the
non-Boussinesq case at an aspect ratio Hx/Hz = 0.5
assuming an impermeable top and a temperature
gradient D = 50°C/km.

Rc = 29.59

$\dfrac{Hx}{Hz}$ = 0.25
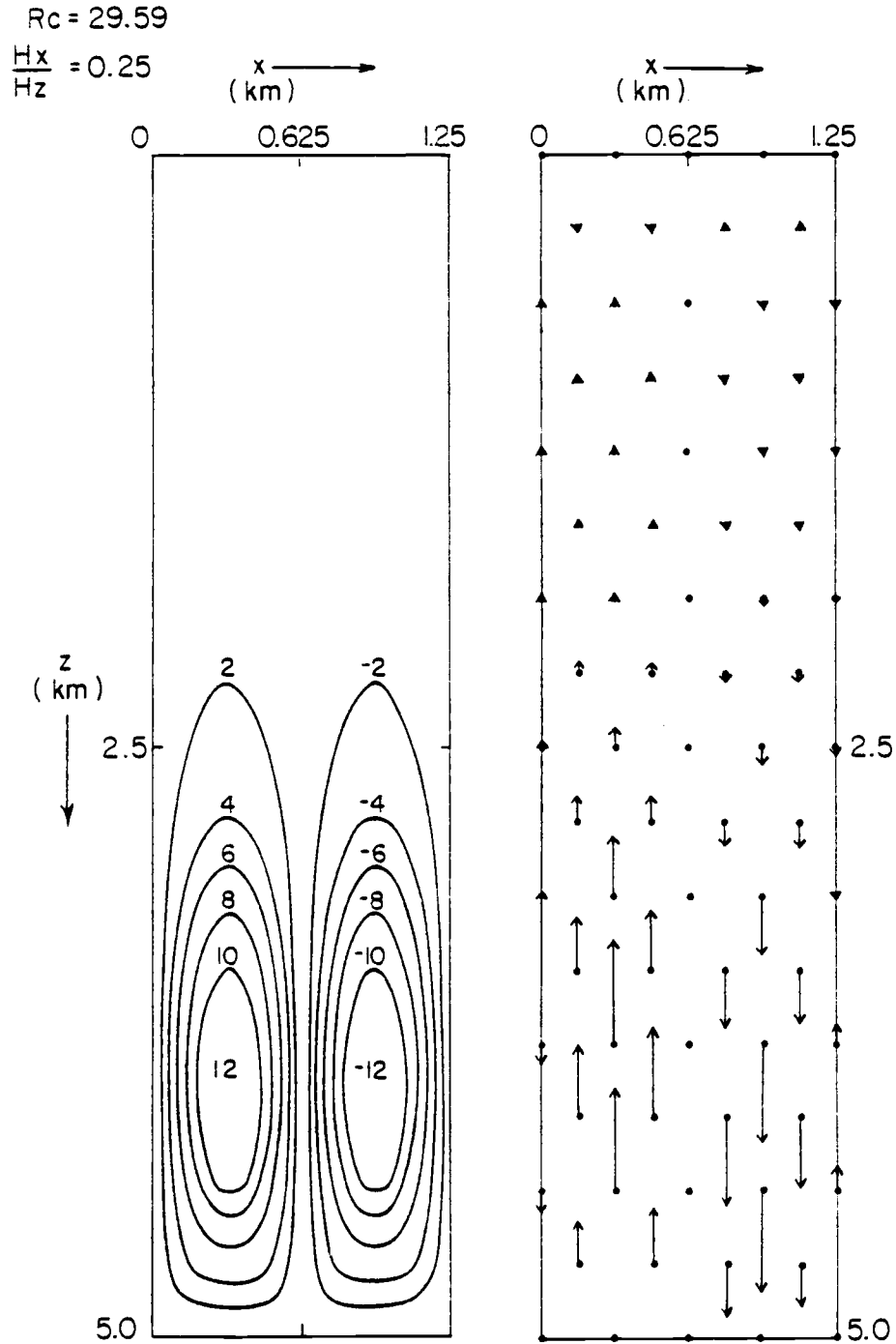
Figure 21. Critical isotherms and vertical flow vectors for the
non-Boussinesq case at an aspect ratio Hx/Hz = 0.5
--assuming an impermeable top and a temperature
gradient D = 50°C/km.

Non-Boussinesq Slab Models with Conducting
Boundaries and a Permeable Top

Since in the case of variable fluid properties discussed above

mainly the lower portion of the porous slabs is strongly involved in

the onset of convection, the upper top boundary conditions have little

influence on the critical Rayleigh number. Accordingly, one may

expect that in the non-Boussinesq case the critical Rayleigh number

for slabs with permeable tops should be very close to the values

those obtained for the case of impermeable tops. Several numerical

tests of this conjecture have been run with the help of our program

RAYLEI. In the work we have chosen $Hx = 0.5$ km, $Hz = 1$ km,

$D = 30°C/,m$, and obtained practically the same critical Rayleigh

number $Rc = 171.9$ for both cases of top condition. Therefore, the

data furnished in Figure 18 hold (except at near surface) equally well

for both types of top condition. The critical isotherms and velocity

vector solutions shown in Figure 22 are nearly the same for both

cases (Figure 20). The vertical velocity vectors shown at the top sur-

face in Figure 22 are extremely small but finite quantities.
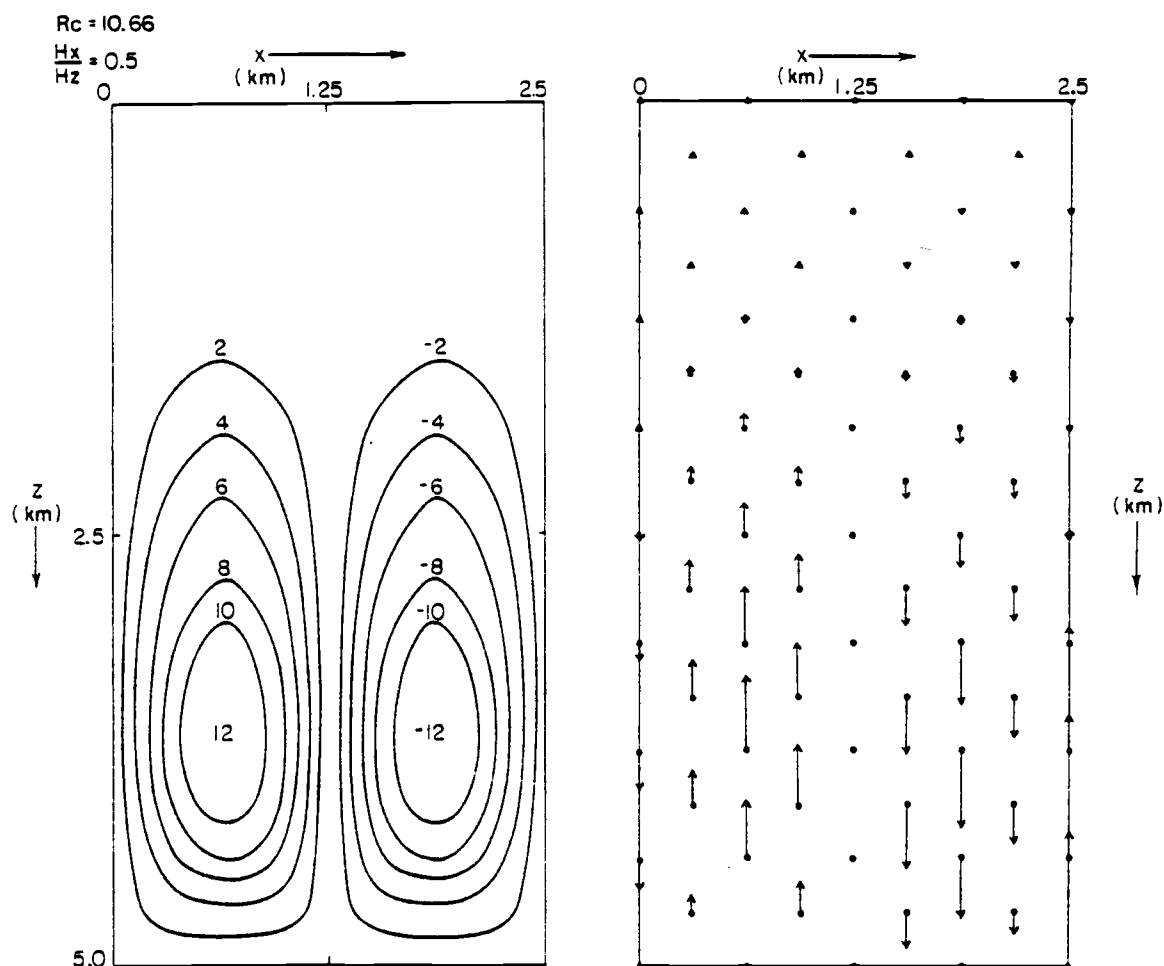
Figure 22. Critical isotherms and vertical flow vectors for the
non-Boussinesq case at an aspect ratio Hx/Hz = 0.5
--assuming a permeable top and a temperature gradi-
ent D = 50°C/km.

V. VARIABLE PERMEABILITY

In the natural setting, the formation permeability may vary within wide limits. In our modeling, we are therefore interested in being able to cope with large variations of this parameter. It is one of the advantages of the finite element formulation applied above that it does allow the material properties to vary from element to element, and therefore only little additional work is needed to be able to handle models with a complex permeability distribution. In order to look into the implications of a variable permeability, we present below results for four physically interesting cases: namely (a) a permeability constant with depth (Figure 23-a), (b) permeability decreasing linearly with depth (Figure 23-b), (c) permeability decreasing quadratically with depth but less than in the linear case (Figure 23-c) and (d) permeability decreasing quadratically with depth but greater than in the linear case (Figure 23-d). (To the best of our knowledge, the critical Rayleigh number of an infinite horizontal layer of variable permeability has not been determined.)

Since in the above finite element formulations, the derivatives of the material properties are included in equations (3.12) to (3.15), both gradual as well as abrupt changes from element to element are equally well within the capabilities of our program RAYLEI. For the present purpose we will consider the specific case of a vertical slab

Ks=Permeability at the top

Z(Km)

ⓒ ⓑ ⓓ ⓐ
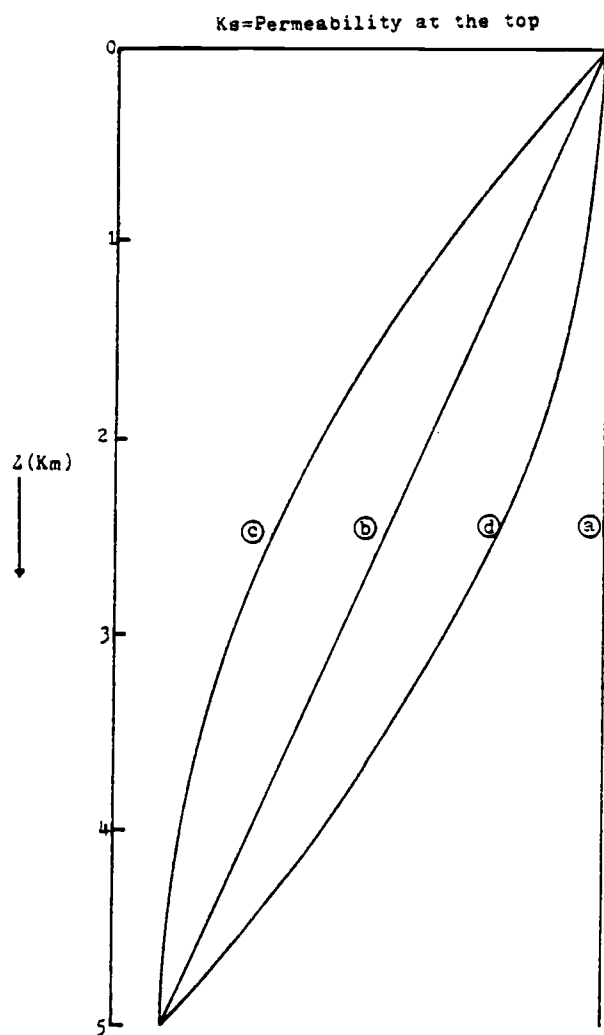
Figure 23. Four permeability models.
a = permeability constant with depth.
b = permeability decreasing linearly with depth.
c and d = two cases of permeability decreasing
quadratically with depth.

with an aspect ratio of Hx/Hz = 1/4 and a depth Hz = 5 km.
Except for case (a), the permeability varies as much as an order of
magnitude between top and bottom. The unperturbed temperature
gradient is assumed to be D = 50°C/km.

It is of some interest to discuss the effects of the surface
hydrodynamic boundary conditions before solving the above stated
problems. Making use of the results obtained in previous sections,
the critical Rayleigh number for the case Hx/Hz = 1/4 with a
permeable top is within 0.5% of that in the case of an impermeable
top (Table 2 and 3). The effects of temperature and pressure, which
shift the main convection zone downward, further reduce the influence
of the top boundary conditions. We may conclude that for
Hx/Hz = 1/4, the critical Rayleigh number is largely independent of
the hydrodynamic conditions at the top. This conclusion holds
although one might expect that the decrease of the permeability with
depth could offset the effects of temperature and pressure especially
when both the bottom permeability and temperature are low. To look
further into case (c) in Figure 23 with an unperturbed gradient
D = 30°C/km. As expected we find a critical Rayleigh number of
301.0 for an impermeable top, and a value of 300.9 for a permeable
top. This result implies that only one of the top boundary conditions
will be needed to analyze the rest of the cases. In Table 5 below,
we show further results for the critical Rayleigh number when

$Hx/Hz = 1/4$, $Hz = 5$ km, and $D = 50°C/km$. Except for the first

two rows which are quoted here from our previous results, we have

taken full account of the effects of temperature and pressure. As

expected the permeability decrease with depth tends to stabilize the

fluid within the porous material. In contrast to the influence of tem-

perature and pressure, these effects raise the values of the critical

Rayleigh number and thus suppress the onset of thermal convection.

In our earlier investigation, we have emphasized the importance of

variable hydrodynamic properties on the Rayleigh number. In the

present section, we conclude that the variable permeability is of

equal importance. This is not surprising since the parameter K

enters into equations (3.12) to (3.14) in the same manner as the

viscosity does.

Table 5. Critical Rayleigh numbers for various permeability situa-
tions assuming $Hx/Hz = 1/4$, $Hz = 5$ km and an unperturbed
gradient $D = 50°C/km$; Ks = permeability at top.

Permeability Functions	Rc	
(a) $K=Ks$	680.6	Imper. Top; Boussi. Approx.
(a) $K=Ks$	680.0	Per. Top; Boussi. Approx.
(a) $K=Ks$	29.6	Imper. Top; Non-Boussi.
(b) $K=(1-0.18z)Ks$	89.3	Imper. Top; Non-Boussi.
(c) $K=(1-0.36z+0.036z^2)Ks$	156.2	Imper. Top; Non-Boussi.
(d) $K=(1-0.036z^2)Ks$	62.3	Imper. Top; Non-Boussi.

The critical isotherms and vertical flow vectors corresponding to the cases in Table 5 are given in Figures 24 to 28. One interesting characteristic revealed in these figures is the shifting of the site of the main convection zone as we observe in the cases (a) to (d). Figures 20 to 22 reveal that the variable water properties force the main convections down to lower regions, whereas Figures 26 to 28 show that the permeability decrease with depth pushes it up to higher regions. This is physically reasonable since more buoyant water is created by higher temperatures at depth which is then dispersed in the more permeable sections of the porous medium. In these examples, the effect of the permeability decrease with depth nowhere completely offsets the effects of the variable water properties. The tendency of the main convection cells to move up or down as a result of these effects is a very useful piece of information which helps us to understand the physical behavior of natural convection systems.
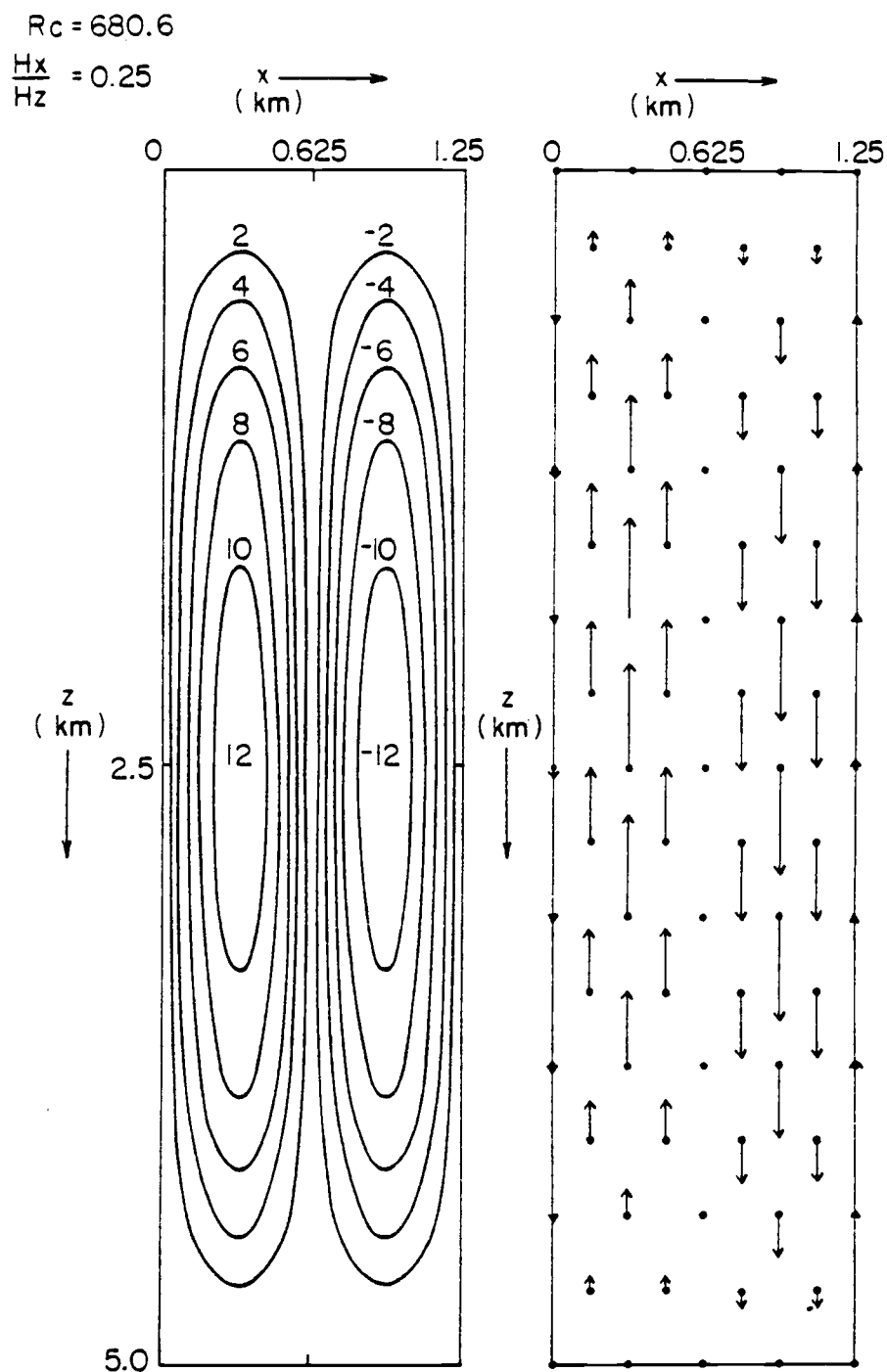
Figure 24. Critical isotherms and vertical flow vectors for an
impermeable top assuming Boussinesq approximation,
an unperturbed gradient D = 50°C/km and the
permeability function (a) in Table 5.

Rc = 680.0

$\frac{Hx}{Hz}$ = 0.25
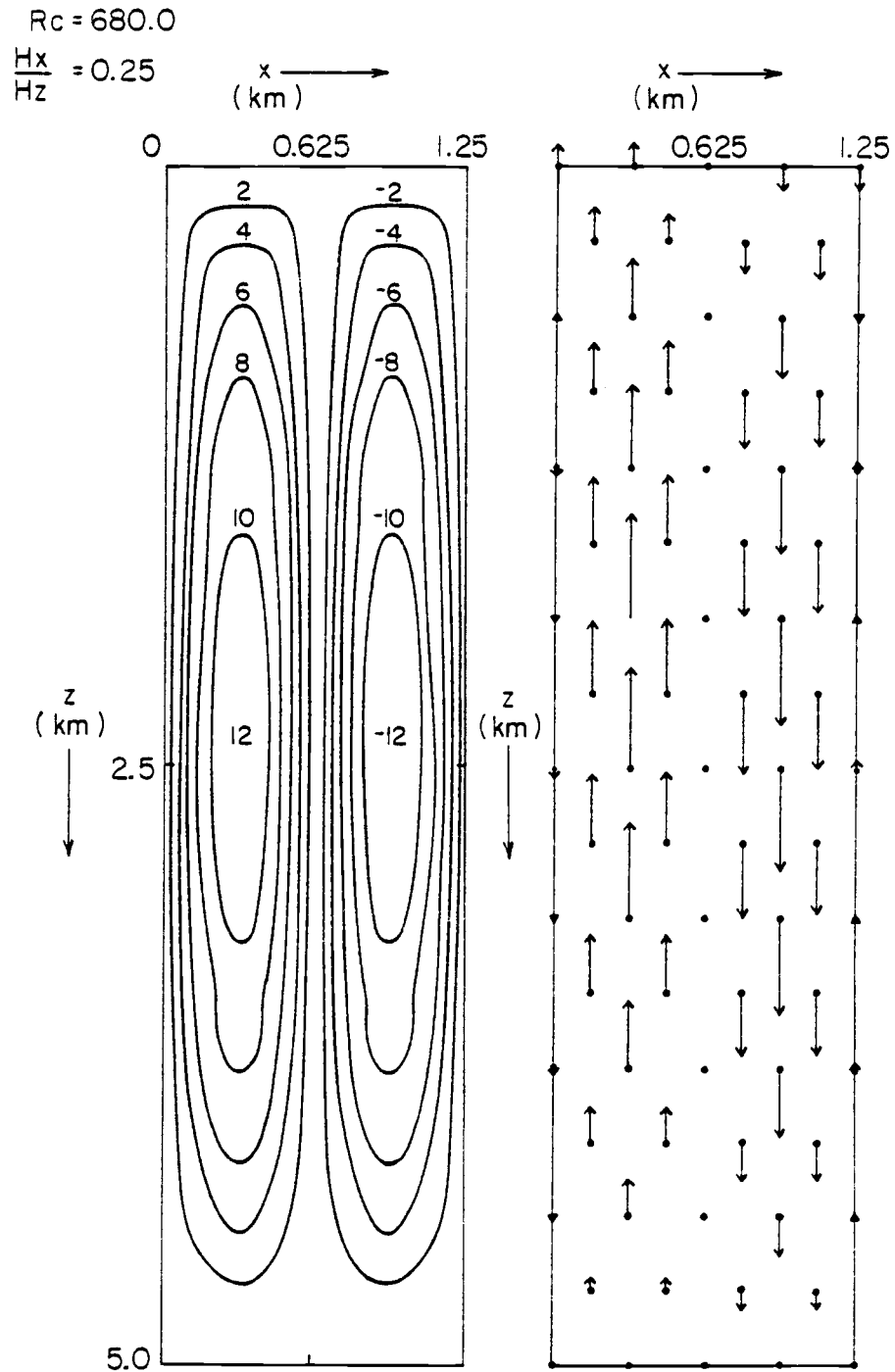
Figure 25. Critical isotherms and vertical flow vectors for a
permeable top assuming Boussinesq approximation,
an unperturbed gradient D = 50°C/km and the
permeability function (a) in Table 5.

Rc = 89.3

$\frac{Hx}{Hz}$ = 0.25
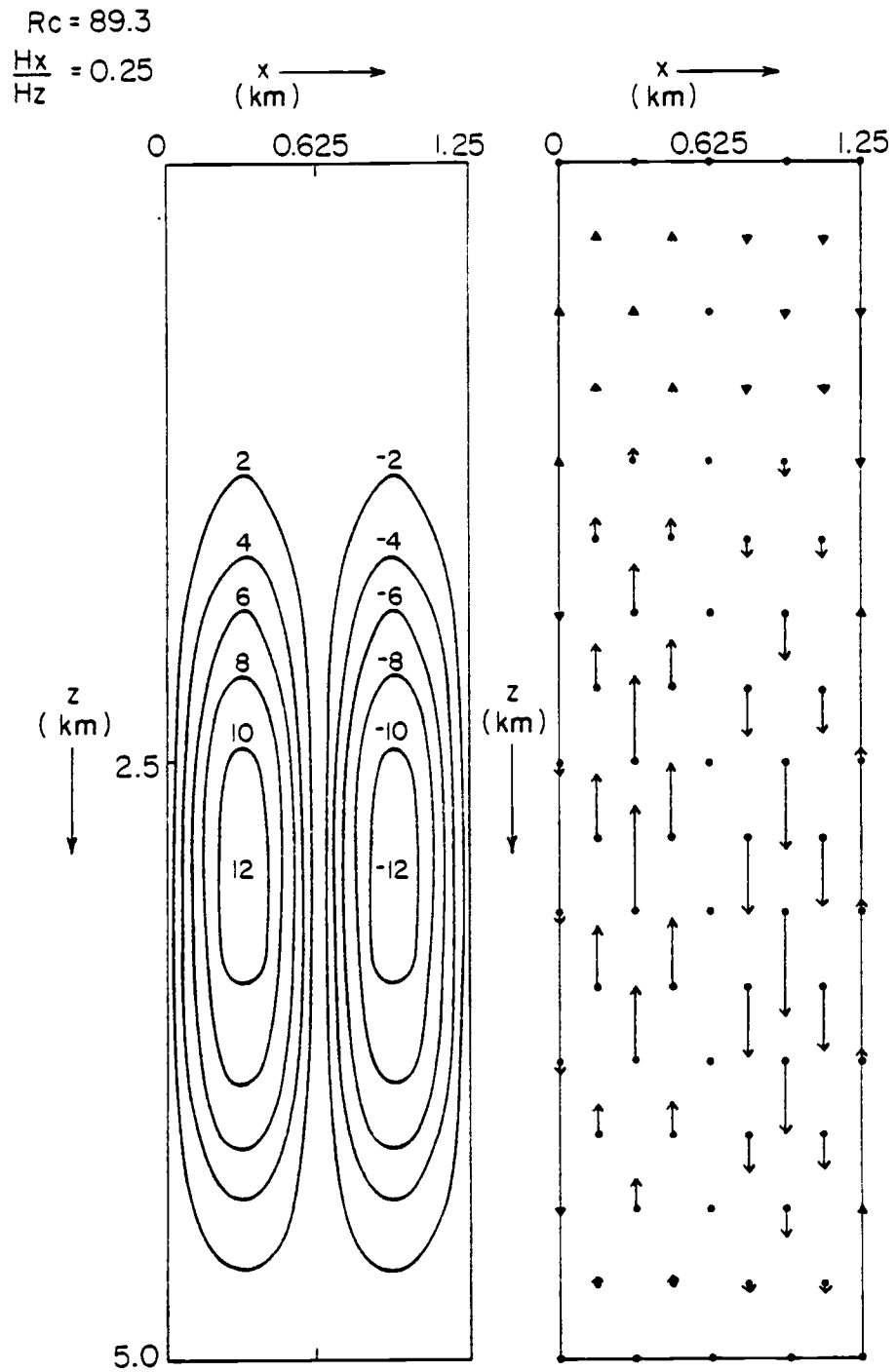
Figure 26. Critical isotherms and vertical flow vectors for an impermeable top assuming non-Boussinesq approximation, an unperturbed gradient D = 50 °C /km and the permeability function (b) in Table 5.

Rc = 156.2

$\dfrac{Hx}{Hz}$ = 0.25
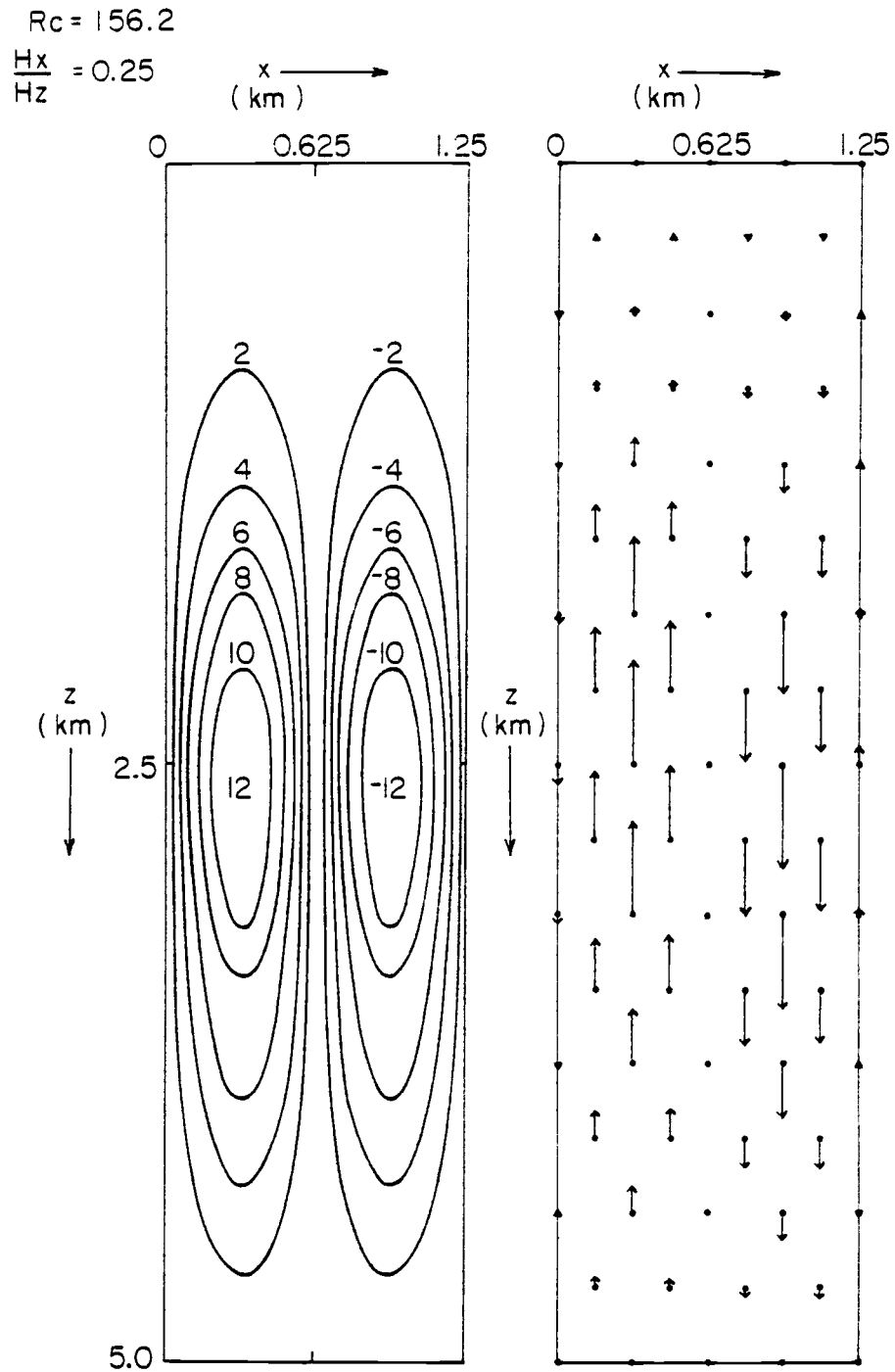
Figure 27. Critical isotherms and vertical flow vectors for an
impermeable top assuming non-Boussinesq approxi-
mation, an unperturbed gradient, D = 50°C/km and
the permeability function (c) in Table 5.

Rc = 62.3

$\dfrac{Hx}{Hz}$ = 0.25
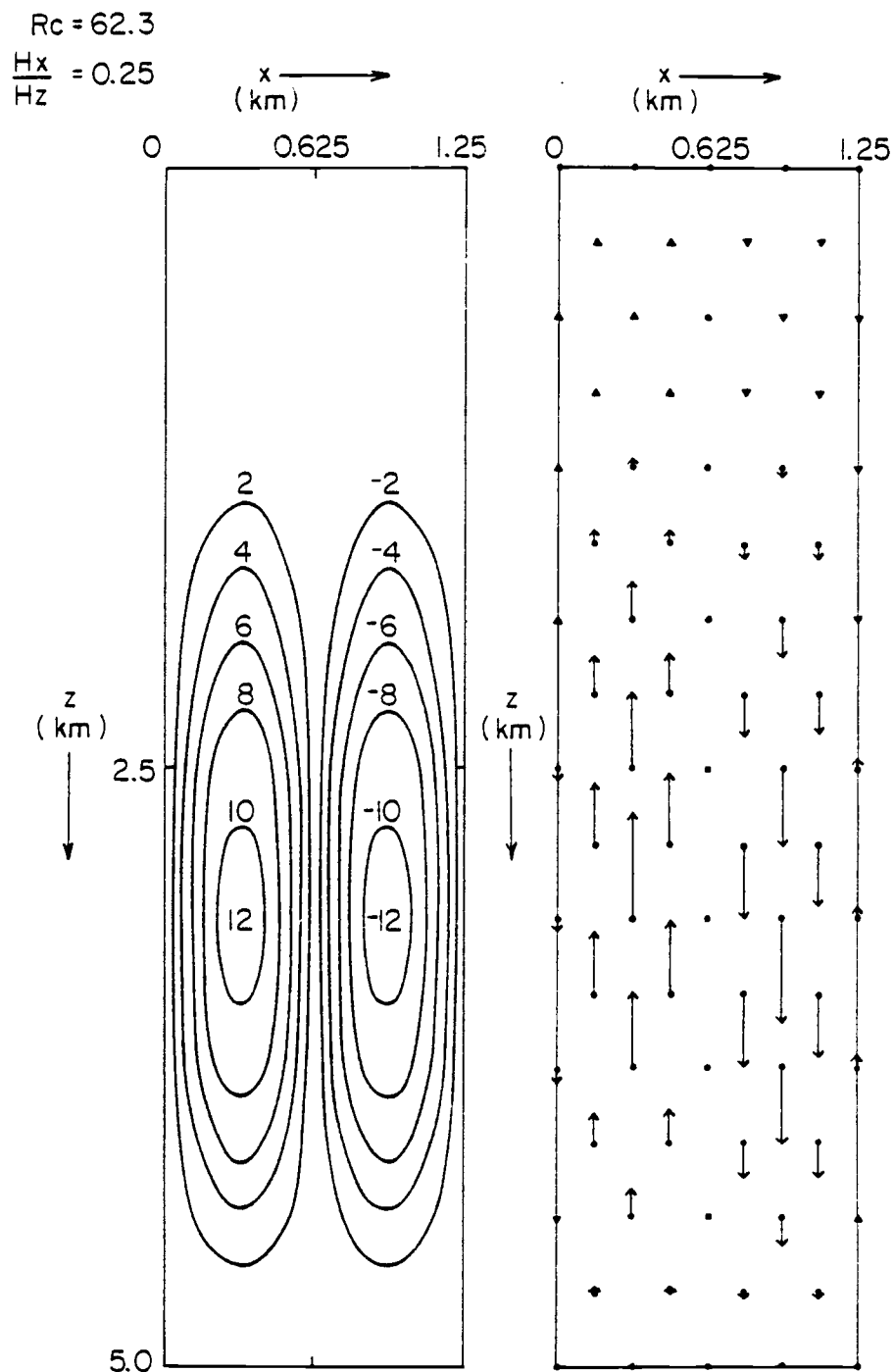
Figure 28. Critical isotherms and vertical flow vectors for an impermeable top assuming non-Boussinesq approximation, an unperturbed gradient D = 50°C/km and the permeability function (d) in Table 5.

77

VI. TIME DEPENDENT MODELS

In our analysis, we have so far concentrated on the critical functions such as the Rayleigh number and flow fields at the onset of convection. Our analysis has been based on the linear perturbation approximation of the governing equations and on the assumption of time independent temperature and flow boundary conditions. A number of interesting results concerning the onset of convection in our models have been obtained this way.

It is obviously of considerable interest to look briefly into a more general problem setting involving the evolutionary or time-dependent aspects of convection in our slab models when both flow non-linearities and heat conduction processes in the bounding media are taken into account. Below we will briefly present some computational results along these lines.

Governing Equations and Galerkin Formulation

The equations governing the heat and mass flow through porous media, based on Darcy's law and the principle of conservation of momentum and energy are listed earlier as (2.1), (2.2), and (2.3). For the present purpose we may again neglect the inertial forces as small compared with viscous forces in the porous medium. The left-hand side of (2.2) can then be expanded as

$$\frac{\partial \phi \rho_f}{\partial t} = \rho_f \frac{\partial \phi}{\partial P} \frac{\partial P}{\partial t} + \phi[(\frac{\partial \rho_f}{\partial P})_T \frac{\partial P}{\partial t} + (\frac{\partial \rho_f}{\partial T})_P \frac{\partial T}{\partial t}]$$

$$= -\rho_f \beta_m \frac{\partial P}{\partial t} + \rho_f \phi \beta \frac{\partial P}{\partial t} - \rho_f \phi \alpha \frac{\partial T}{\partial t}$$

where β_m is the compressibility of the wet porous medium. Thus, in the final form, our equations are

$$\vec{V} = \frac{K}{\mu} (-\nabla P + \rho_f \vec{g}) \tag{4.1}$$

$$\rho_f(\phi\beta - \beta_m) \frac{\partial P}{\partial t} = -\nabla \cdot (\rho_f \vec{V}) + \rho_f \phi \alpha \frac{\partial T}{\partial t} \tag{4.2}$$

$$[\rho_f \phi C_f + (1-\phi)\rho_s C_s] \frac{\partial T}{\partial t} = k_m \nabla^2 T - \rho_f C_f \vec{V} \cdot \nabla T + \alpha T \vec{V} \cdot \nabla P$$

$$+ \alpha T \frac{\partial P}{\partial t} . \tag{4.3}$$

We need to introduce the heat conduction equation to describe the temperature field in the solid material bounding the porous medium, that is,

$$\frac{\partial T}{\partial t} = a \nabla^2 T \tag{4.4}$$

where a is the thermal diffusivity of the solid wall material.

Although the basic problem is now time-dependent, the Galerkin finite-element procedure can be applied as in Chapter III. The conduction-convection phenomena involve a density dependence coupled with fluid flow which renders the problem nonlinear.

Moreover, some difficulties arise because of the explicit time-dependence of the temperature and pressure fields.

A convenient approach to the solving of the equations is to combine (4.1) and (4.2) by eliminating $\vec{V}$ and assuming that the last terms in (4.2) and (4.3) are negligible quantities. The combined equations can then be solved for the pressure P which is substituted into (4.1) to obtain the velocity field V. The velocity field is subsequently inserted in (4.3) for calculating the temperature field.

The above procedure leads to satisfactory results when the conductive heat transport is substantial. Computational difficulties (Meissner, 1973; Segol, et al., 1975; Dessai and Contractor, 1977) may, on the other hand, arise if the convective terms are very large. These results from the computation of the velocity field by (4.1) based on trial solutions for the pressure field only. Moreover, the computation of derivatives directly from trial solutions may in our case result in velocity discontinuities across the element boundaries. In view of this, it has been found necessary to include the components of the velocity as well as the pressure fields as primary nodal unknowns. This procedure has the advantage that the pressure and the velocity boundary conditions may be readily incorporated into the matrix equations. Accordingly, to insure an accurate velocity field in the mixed approach, the nodal unknowns were assumed to be temperature T, the velocity components u and w, and the

pressure P which are expressed as

$$T \approx \hat{T} = \sum_{i=1}^{M_t} N_i T_i(t), \quad P \approx \hat{P} = \sum_{i=1}^{M} N_i P_i(t)$$

$$u \approx \hat{u} = \sum_{i=1}^{M} N_i u_i(t), \quad w \approx \hat{w} = \sum_{i=1}^{M} N_i w_i(t)$$

Here, the set of basis functions N_i satisfy the essential boundary conditions of the partial differential equations and are assumed to be linearly independent. Moreover, $T_i(t)$, $u_i(t)$, $w_i(t)$, and $P_i(t)$ are unknown time dependent coefficients and M_t and M are the number of total temperature respectively the total flow nodes. In our mixed domains, M_t is much larger than M. In order to allow the parameters ρ_f, α, β, C_f, μ, and K to vary linearly in an element, we can use the functional representation for a parameter function f

$$f(t, x, z) = \sum_{j=1}^{M_c} L_j f_j(t)$$

Here again the parameters are function of time. The element equations were obtained by using the Galerkin residual procedure as in Chapter III. Only the relevant details are presented here. In this approach, the terms in equation (4. 1) can be expanded such that

$$\int_D \mu N^T N_k \, da \{u\} = -\int_D K \frac{\partial N^T}{\partial x} N_k \, da \{P\}$$

$$\int_D \mu N^T N_k \, da \{w\} = -\int_D K \frac{\partial N^T}{\partial z} N_k \, da \{P\} + \int_D K \rho_f g N_k \, da$$

and combined in a matrix form

$$[A]\{V\} + [B]\{P\} = \{F\}$$

where $\{V\}$, $\{P\}$, and $\{F\}$ are arrays of order $2M$ defined by

$$\{V\} = \begin{Bmatrix} u \\ w \end{Bmatrix}, \quad \{P\} = \begin{Bmatrix} P \\ P \end{Bmatrix}, \quad \{F\} = \int_D \begin{Bmatrix} 0 \\ K \rho_f g N_k \end{Bmatrix} da$$

$$[A] = \int_D \begin{bmatrix} \mu N^T N_k & 0 \\ 0 & \mu N^T N_k \end{bmatrix} da,$$

$$[B] = \int_D \begin{bmatrix} K \frac{\partial N^T}{\partial x} N_k & 0 \\ 0 & K \frac{\partial N^T}{\partial z} N_k \end{bmatrix} da.$$

Substituting (4.1) into (4.2) to eliminate $\vec{V}$, we find

$$\rho_f(\phi\beta - \beta_m) \frac{\partial P}{\partial t} = -\nabla \cdot \left[\frac{\rho_f K}{\mu} (-\nabla P + \rho_f \vec{g}) \right] + \phi \rho_f \alpha \frac{\partial T}{\partial t}$$

$$= \frac{\rho_f K}{\mu} \nabla^2 P - \nabla \left(\frac{\rho_f K}{\mu} \right) \cdot (-\nabla P + \rho_f \vec{g}) - \frac{\rho_f K g}{\mu} \frac{\partial \rho_f}{\partial z} + \phi \rho_f \alpha \frac{\partial T}{\partial t}$$

$$(4.5)$$

Applying Green's theorem to the Laplacian operator in (4.3), (4.4)

and (4.5) results in a system of matrix equations of the form

$$[A] \frac{\partial}{\partial t} \{x\} + [B]\{x\} = \{F\}$$

where

$$\{x\} = \{P\}$$

$$[A] = \int_D \rho_f \mu^2 (\phi\beta - \beta_m) N^T N_k \, da$$

$$[B] = \int_D [\rho_f \mu K (\frac{\partial N^T}{\partial x} \frac{\partial N_k}{\partial x} + \frac{\partial N^T}{\partial z} \frac{\partial N_k}{\partial z})] \, da$$

$$- \int_D \left\{ [\mu^2 K \frac{\partial}{\partial x} (\frac{\rho_f}{\mu})] \frac{\partial N^T}{\partial x} + [\mu^2 \frac{\partial}{\partial z} (\frac{\rho_f K}{\mu})] \frac{\partial N^T}{\partial z} \right\} N_k \, da$$

$$\{F\} = - \int_D g\rho_f \mu^2 \frac{\partial}{\partial z} (\frac{\rho_f K}{\mu}) N_k \, da + \int_D (\phi \rho_f \alpha \mu^2) N^T N_k \, da \frac{\partial T}{\partial t} + \int_C \mu \rho_f K \frac{\partial \hat{P}}{\partial n}$$

in (4.5),

$$\{x\} = \{T\}$$

$$[A] = \int_D [\phi \rho_f C_f + (1-\phi) \rho_s C_s] N^T N_k \, da$$

$$[B] = \int_D k_m (\frac{\partial N^T}{\partial x} \frac{\partial N_k}{\partial x} + \frac{\partial N^T}{\partial z} \frac{\partial N_k}{\partial z}) \, da$$

$$+ \int_D \rho_f C_f (u \frac{\partial N^T}{\partial x} + w \frac{\partial N^T}{\partial z}) N_k \, da$$

$$- \int_D [\alpha \rho_f g w - \frac{\alpha \mu}{K}(u^2 + w^2) + \alpha N^T \frac{\partial P}{\partial t}] N^T N_k \, da$$

$$\{F\} = \int_C k_m \frac{\partial \hat{T}}{\partial n} N_k \, ds$$

in (4.3), and

$$\{x\} = \{T\}$$

$$[A] = \int_{D(rock)} (\rho C) rock \cdot N^T N_k \, da$$

$$[B] = \int_{D(rock)} (k) rock \cdot (\frac{\partial N^T}{\partial x} \frac{\partial N_k}{\partial x} + \frac{\partial N^T}{\partial z} \frac{\partial N_k}{\partial z}) da$$

$$\{F\} = \int_{C(rock)} (k) rock \cdot \frac{\partial \hat{T}}{\partial n} \cdot N_k \, ds$$

in (4.4). Here, C is the boundary of the domain for each equation and $\partial \hat{P}/\partial n$, $\partial \hat{T}/\partial n$ are derivatives along the outward normal to C. The expressions F in (4.3) and (4.4) are assumed to cancel mutually at the boundary between the porous media and the adjacent rock. Physically, we assume that the heat flow is continuous at that boundary, that is, there is no accumulation of heat.

Details of Solution Algorithm

In general, an iterative approach is required to obtain a satis-factory solution to the above four coupled basic equations (4.1), (4.2),

(4.3) and (4.4). It is often advantageous to choose different time steps in solving the flow and energy equations. A time-lagged formulation procedure can allow considerable economy in the computation and still provide acceptable results (Sorey, 1976; Lippmann et al., 1976). Using this method the interrelated equations (4.3), (4.4) and (4.5) are solved alternatively by interlacing solutions in time. The flow equations are solved for P and $\vec{V}$ at a time $t+(\Delta t/2)$ by using data on $\partial T/\partial t$ and the fluid properties at t. Then the energy equations are used to obtain T at time $t+\Delta t$ with the help of data on $\vec{V}$, $\partial P/\partial t$ and the fluid properties at $t+(\Delta t/2)$. The values of the temperature and pressure dependent properties of the fluid are subsequently revised and then used to adjust the partial differential equations. From the physical point of view, the variation of the pressure in the flow equations is much faster than the variation of the temperature in energy equations. It is therefore justified to use much smaller time steps in the flow cycles than in the energy cycles. This technique which is illustrated in Figure 29 is essentially similar to the one suggested by Lippman et al. (1977). Our program CONVEC (with an example as illustration, is shown in Appendix D) has been developed to carry out these computations. To assure the convergence of the final steady state solutions, and to determine suitable time steps, several experiments were conducted with the proper boundary conditions and parameters. Since our model is designed to treat

evolutionary or transient problems, an arbitrary set of initial

conditions was assumed and the procedure was carried out through

time until a steady state was achieved. Our experience is that shorter

time steps in the energy equations require fewer time steps overall

in the flow equations and result in more accurate solutions. Longer

time steps in the energy equations require more time steps overall

in the flow equations, but the total computer time required to reach a

steady state is shortened. Short time steps are required in the

energy equation to preserve accuracy and stability at high Rayleigh

numbers.
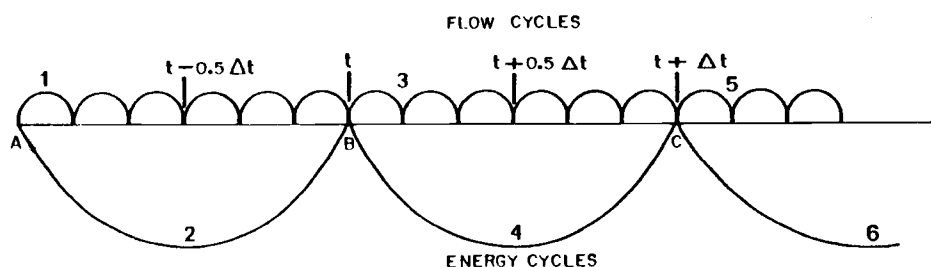
Figure 29. Marching scheme of interlacing energy and flow equations.

To illustrate the evolutionary behavior of models with heat

conducting boundary walls, a number of numerical examples are pre-

sented below. Except for individually specified cases, the material

properties assumed in all examples are listed in Table 6.

Table 6. Material properties used in evolutionary or transient models.

Property	
Density of solid phase	2.7×10^3 (kg/m^3)
Porosity	0.1
Thermal conductivity of solid phase	2.3 (joule/m, s, °C)
Heat capacity of solid phase	9.3×10^2 (joule/kg, °C)
Compressibility of porous media	2.5×10^{-10} (m^2/newton)
Thermal conductivity of fluid phase	0.65 (joule/m, s, °C)

Vertical Porous Slabs with Impermeable Caprock and Walls

Figure 30 shows the specific dimensions, boundary conditions, and initial temperature distribution chosen for the transient system model. The permeable slab C could represent a bounded frac-tured zone with two impermeable but thermally conducting walls. No internal heat sources are assumed for the model and the temperature of the lower boundary has been assumed on the basis of a regional temperature gradient of 30°C/km. The results of the stability analy-sis for the case of $Hx/Hz = 0.25$ and an unperturbed gradient $D = 30$°C/km, illustrated in Figure 18 show that a model of this geometry with perfectly conducting walls and a constant permeability has a critical Rayleigh number of 60.15.
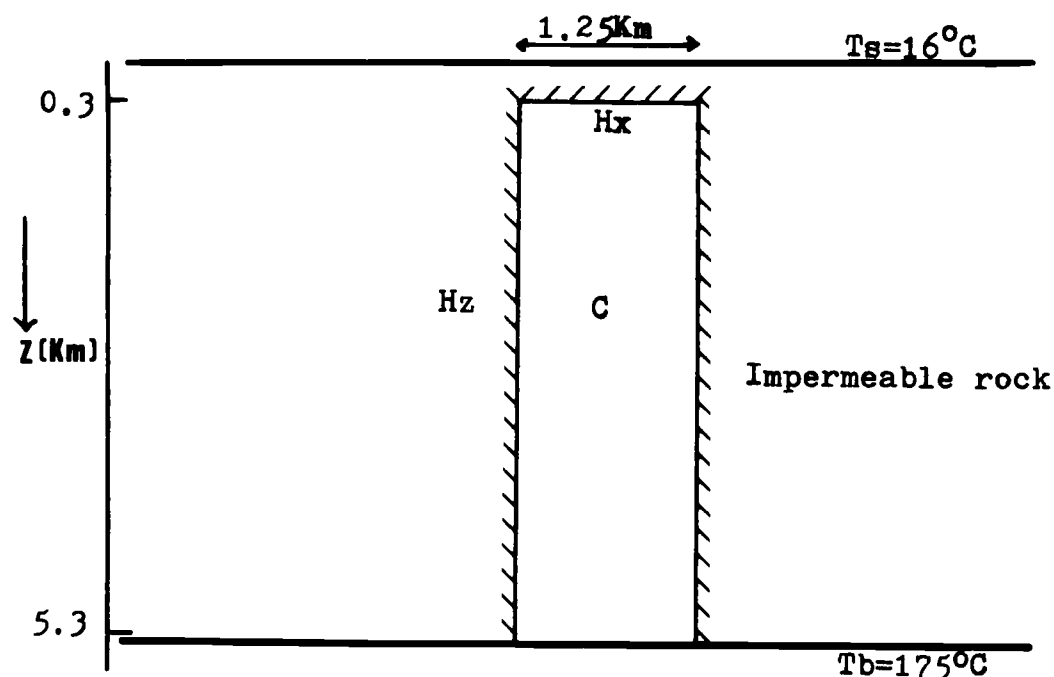
Figure 30. Geometry, initial temperature, and boundary conditions
of the transient system model.

For the present model, the fluid parameters are calculated on

the basis of empirical equations (e.g. Meyer et al., 1968; Mercer

et al., 1975) or with the help of our subprograms INIT and DENPRO

(Appendix D). The substituting of these parameters into the Rayleigh

number gives a corresponding permeability of $K = 1.19 \times 10^{-10} cm^2$

(11.9 millidarcy).

For a rapid approach to steady state in this example, the initial

disturbance was assumed on the basis of the results obtained from the

stability analysis in Chapter III. Other disturbances were also con-

sidered. In all cases, the steady state solutions were correctly found

to be independent of the disturbance initiating the flow. The run time

required to reach steady state decreases as the Rayleigh number

increases, and reaches a maximum at the critical Rayleigh number.

Temperature and Temperature Gradients

A series of computational operations for different Rayleigh numbers have been carried out for the above model. Assuming an initial unperturbed temperature gradient of $D = 30°C/km$ and computationally convenient pressure field perturbations, the temporal development of the temperature field along the central line of the convective channel has been investigated. The results are shown in Figure 31 and the final steady state isotherms are shown in Figure 32.

Figure 31 indicates that the time to reach a steady state decreases with increasing Rayleigh number. Systems that are almost critical require the maximum relaxation time. The final steady state solution is therefore adequate to describe convection in systems that are older than the relaxation time. For the system at hand a relaxation time of about 7000 years is obtained. As expected, the near-surface temperature gradients in the caprock are quite high whereas the temperature gradients in the convective zone are low (Figure 31). The final steady state temperature gradient in the convective channel depends on the initial Rayleigh number. Higher Rayleigh numbers result in lower steady state gradients. The graphs in Figure 31 show that the steady state temperature field is lower over a considerable part of the convecting cell. We can now define a generalized Rayleigh
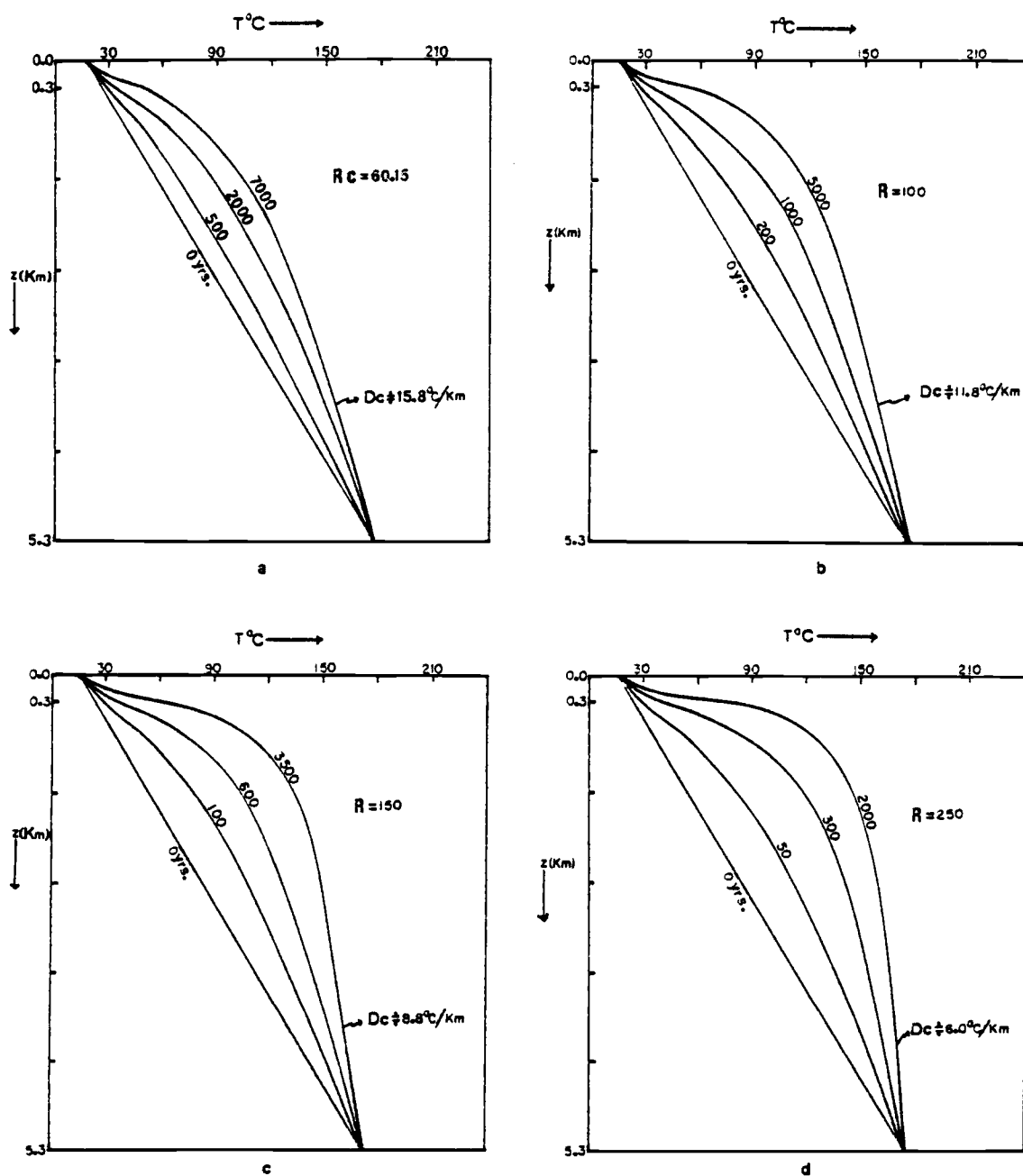
Figure 31. Temperature development at the center plane of the system shown in Figure 30.

Dc = approximate temperature gradient at the center plane of the convective zone.
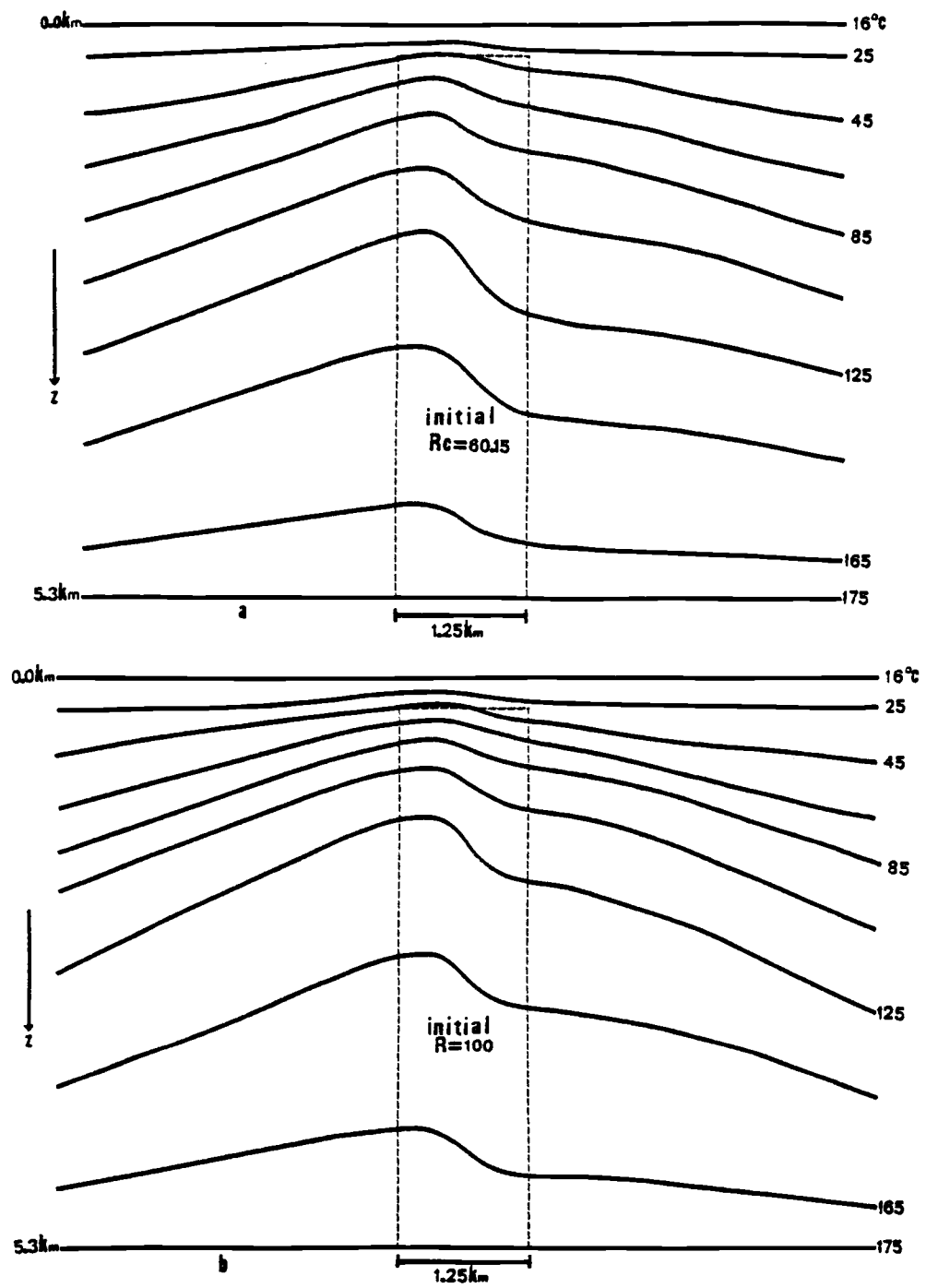
Figure 32. Steady state isotherms at different initial Rayleigh
numbers for the system in Figure 30.
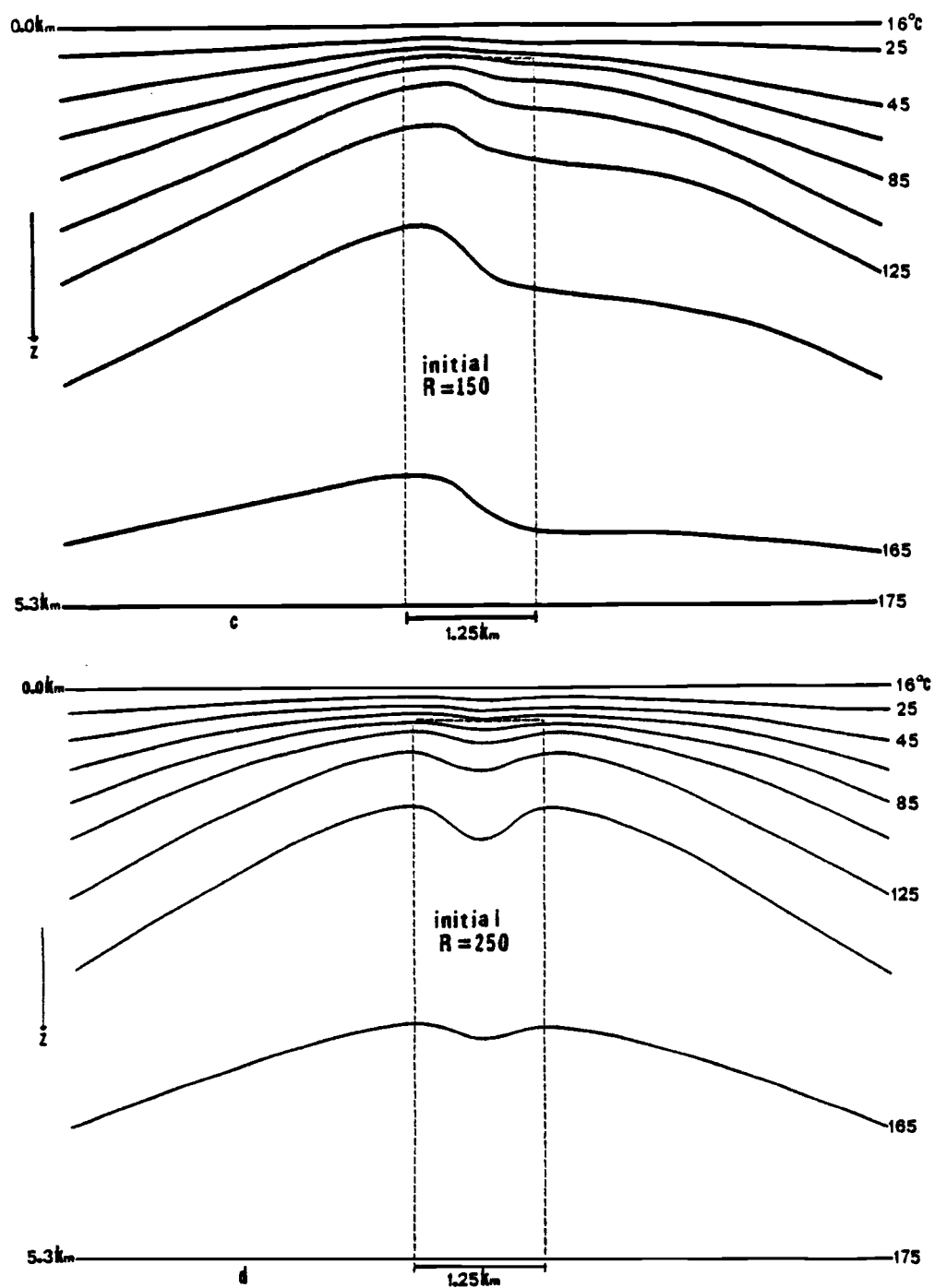
Figure 32.   Continued.

number on the basis of the constant gradient. In the case in Figure 31-a we obtain a value of the generalized number of 28.07. It is of interest to point out that the system is still convecting at this low number although the initial critical Rayleigh number was 60.15. This can be attributed to the different type of thermal boundary conditions (Figure 32). The isotherms across the boundary represent a thermal condition which lies in between the cases with fully conducting and fully insulated boundaries.

Heat Flow Calculation

The computation of the surface heat flow is of considerable interest for the above model. Here we face some difficulties because of discontinuities in the derivatives across the element boundaries. Various techniques can be used to obtain the gradient at node points. One procedure is to employ conjugate approximation theory (Oden and Brauchli, 1971). However, although work is being carried out in this field (Fischer, 1976; Larock and Herrmann, 1977) there is no definite averaging rule available for computation. An alternative approach is to use higher order interpolations over the elements, because differentiation will then yield gradients that are functions of the coordinates. The difficulties involved have, however, not often been emphasized sufficiently. For example, suppose $f(x)$ is given by discrete values at four points. A polynomial of third degree passing

through the four discrete points and approximating the function $f(x)$ in the interval may be written

$$f(x) \simeq \hat{f}(x) = \sum_{i=1}^{4} N_i(x) f_i$$

The approximation gives exact values at the points x_0 to x_3 (Figure 33). But the derivative of $\hat{f}(x)$ may approximate the derivative of $f(x)$ quite poorly.
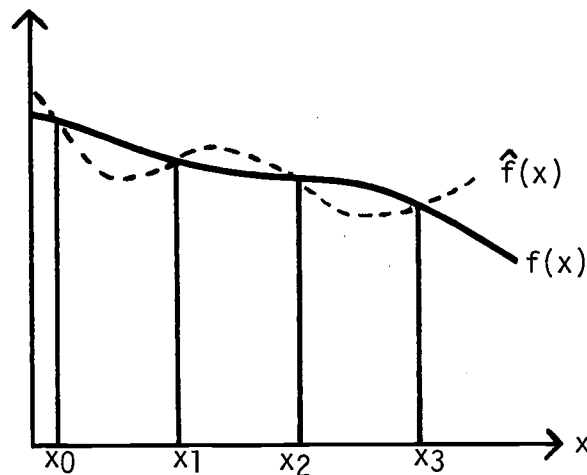
Figure 33. Given function f(x) and its approximate representation $\hat{f}(x) = \Sigma N_i f_i$.

To obtain the heat flow at the surface it is therefore better to apply a finite difference technique which acts as a smoothing devise. Using this method, our results for the surface gradients in Figure 31

are   Ds = 46.5, 56.5, 72.0, and 90.6°C/km   for the cases (a) to (d)

respectively.   The ratio of the calculated surface heat flow to the heat

flow with no convection is shown in Figure 34.   Clearly, the ratio

increases with the Rayleigh number.   On the other hand, the ratio of

the maximums to minimums decreases as the Rayleigh number

increases and in transitions from one mode to the other, the ratio is

further reduced (Figure 34-d to 34-e, Figure 34-f to 34-g).

The results in Figure 34 correspond to the cases of one, two

and three convection cells.   With practical applications in mind

further runs were made over a range of Rayleigh numbers with vari-

ous thicknesses of caprock.   The resulting surface temperature

gradients and temperatures at the top of convective zones, are shown

in Figure 35.   The data indicate a useful relation between the surface

gradient and the Rayleigh number which may in cases possibly be

applied to estimate the Rayleigh number from observational surface

temperature and heat flow data.
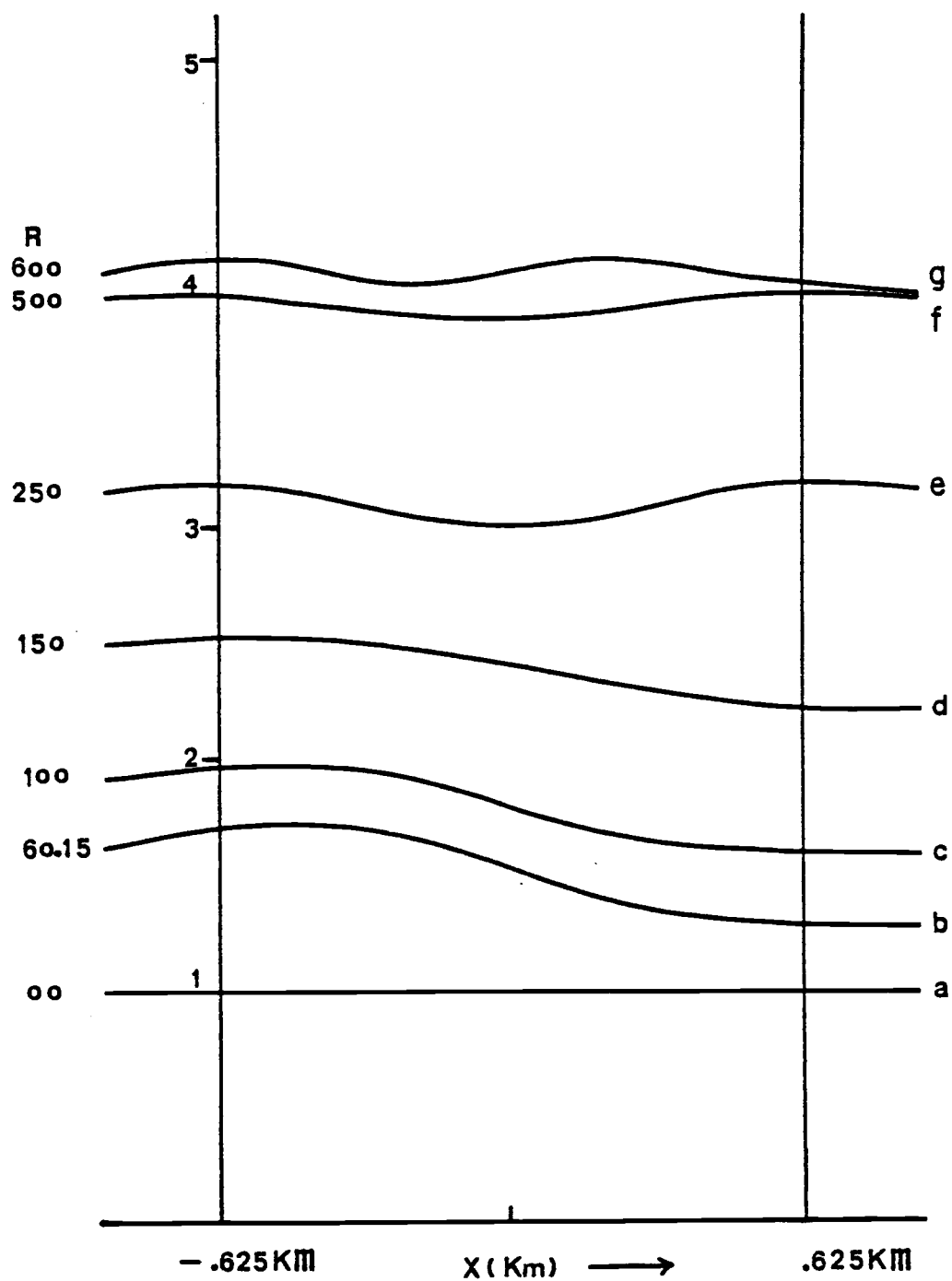
Figure 34. Ratio of the computed surface heat flow to the nonconvective heat flow for the model in Figure 30 with an impermeable cap rock.
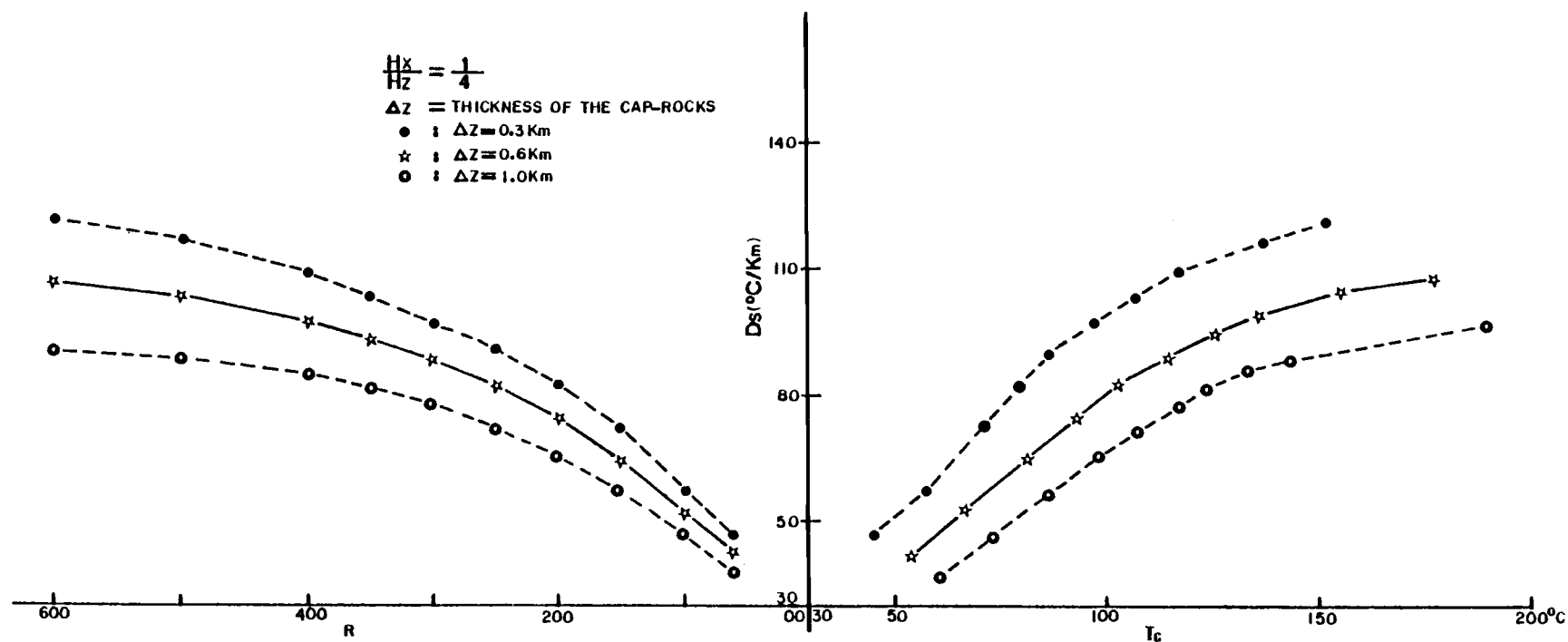
Figure 35. Temperature gradients Ds at the top center of the cap rock and the temperature Tc at the top center of the convection channel for the model in Figure 30 plotted as functions of the thickness of the cap rock and the Rayleigh number.

VII. CONCLUDING REMARKS ON APPLICATIONS

The purpose of the present project has been to develop computational codes and techniques to be applied in the numerical analysis of thermal convection in the natural environment, in particular, in geothermal areas. This involves numerical data on critical field functions and evolutionary processes at various types of field and boundary conditions. Powerful methods for this purpose have been derived and a number of results that have been displayed indicate the capabilities of the techniques.

On the other hand, the analysis of actual field cases was not contemplated. This would constitute a separate task that requires a considerable amount of work. However, to provide a more satisfactory conclusion to our theoretical efforts, we will in the following few paragraphs very briefly outline two cases of application of our results that appear to be of interest for the geothermal sciences.

Thermal Convection in the Imperial Valley, California

For our first case, we will turn to the Imperial Valley in California where there is a considerable amount of geothermal activity that has received attention in the literature. The East Mesa geothermal field that is of particular interest has been described by Swanberg (1976) and further data on conditions in the region and

mainly on the Salton Sea geothermal field are given by Schroeder (1976).

The sketch in Figure 36 shows the location of the "hot spots" or geothermal areas in the Imperial Valley north of the international border. The most relevant characteristics of the areas is that they are embedded in the sediments of the Salton Trough which is a structural depression that forms the northern extension of the Gulf of California and the East Pacific Rise. According to Biehler et al. (1964), the maximum thickness of the sediments is about 6.4 km. The sediments are generally believed to be of the type where Darcy's law can be applied to fluid movements.

The surface heat flow contours of the East Mesa field are shown in Figure 37 (Swanberg, 1976) and a few temperature-depth profiles are given in Figure 38. Over a considerable section of the anomaly the temperature at the depth of 2 km is 180°C to 200°C.

There is strong evidence that the geothermal "hot spots" in the Imperial Valley are caused by local magmatic intrusions into the basement below the floor of the Imperial Trough. Heat conducted from the intrusive bodies raises the temperature in the overlying sediments leading to convection in the entire vertical section above the hot bodies. In other words, the temperature flow-field is supercritical, that is, the Rayleigh number is above the critical value. We will apply this picture to the East Mesa geothermal field and
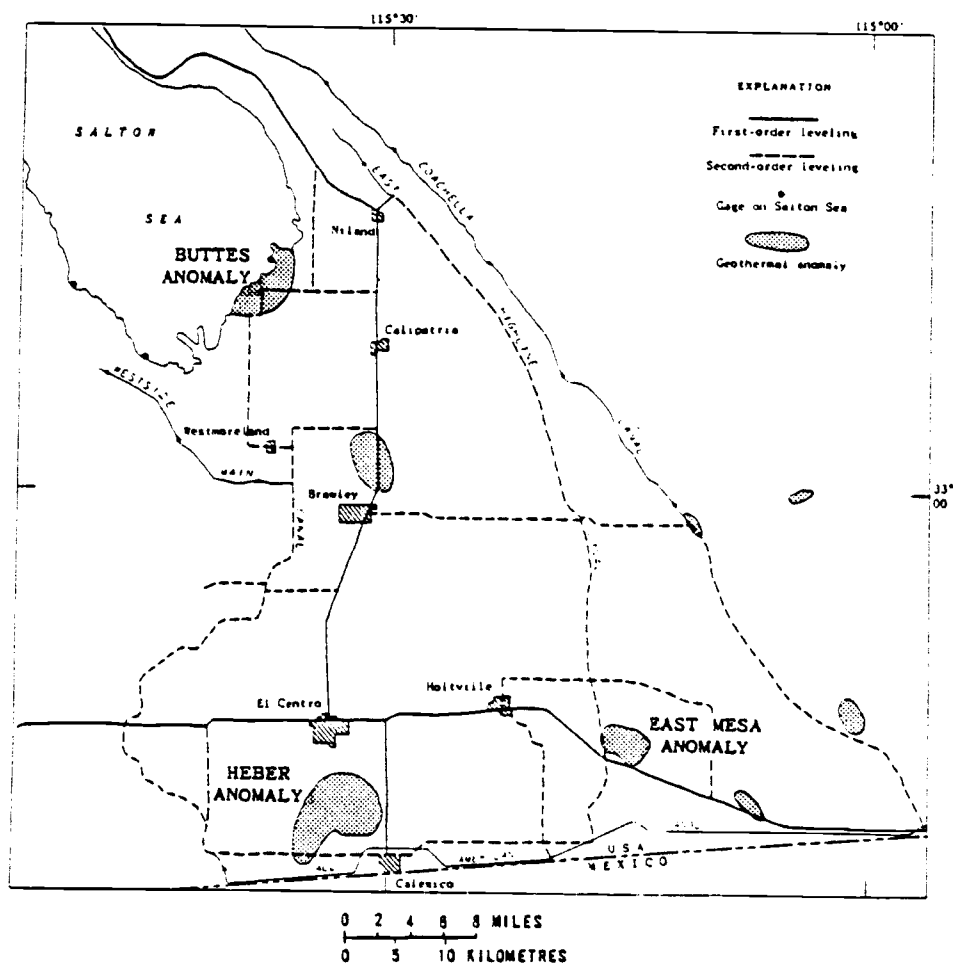
Figure 36.   Location of geothermal anomalies in Imperial Valley,
California (from Lofgren, 1974).

Figure 37. Distribution of heat flow over the Mesa anomaly and the border anomaly (southeast lobe). Data from Swanberg (1976).
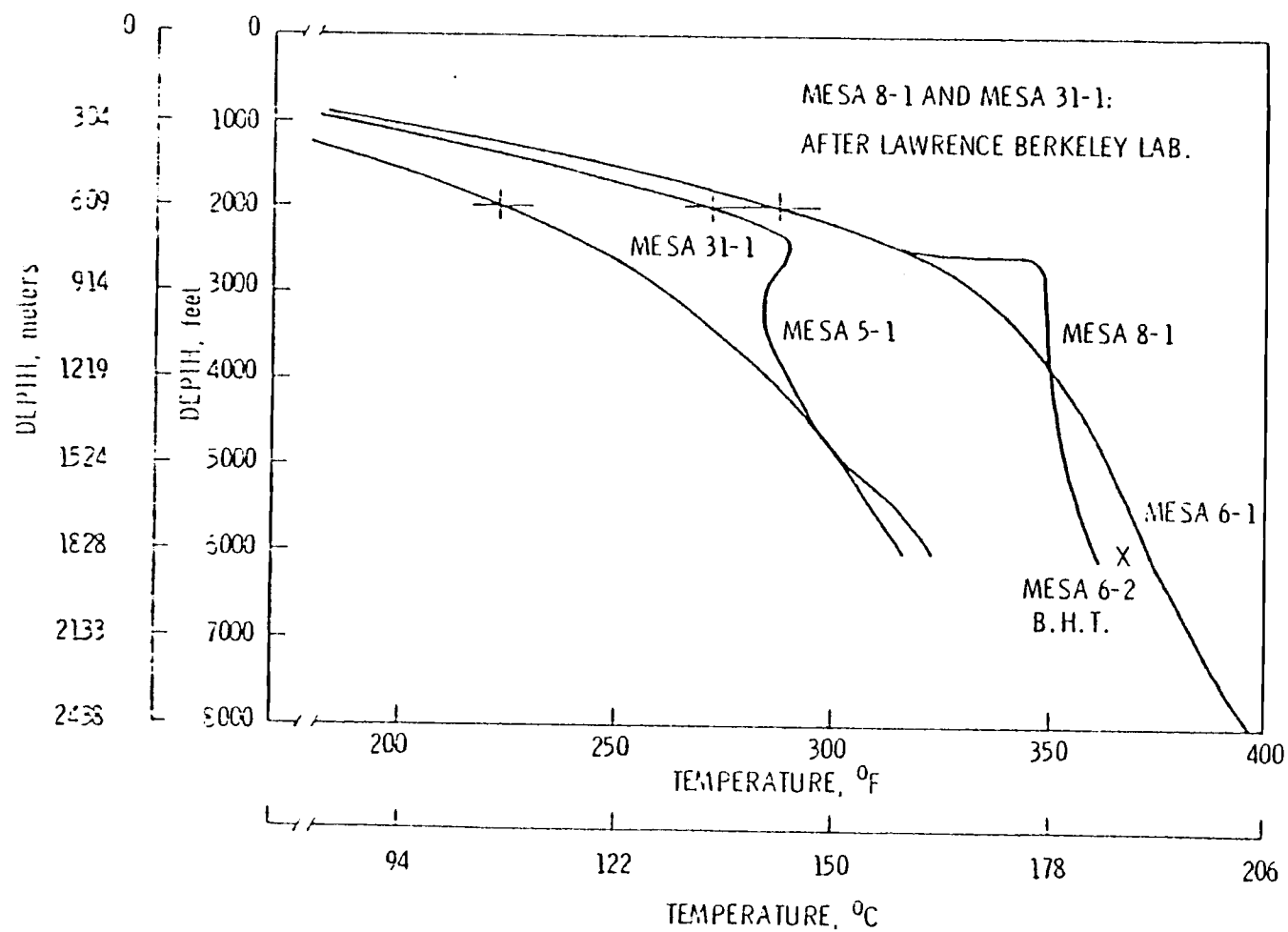
Figure 38. Equilibrium temperature profiles in East Mesa area (from Swanberg, 1976).

estimate the minimum permeability profile that will lead to a critical Rayleigh number when we assume a temperature of, for example, 280°C at the top of the basement. This is a plausible temperature at the top of a solidified silicic intrusion.

The heat flow contours in Figure 37 indicate quite strongly that the East Mesa field is controlled by the Combs-Hadley fault line and that there is convection in an elongated vertical slab along the fault. For the present purpose we assume that the slab has a thickness of about 1.25 km and that it bottoms out at 5 km such that the aspect ratio is 1/4. Moreover, we assume that the quadratic permeability profile (c) in Figure 23 can be taken to represent the conditions in the vertical sedimentary column.

On the basis of these assumptions, the data in Figure 27 are relevant to the present problem. The critical Rayleigh number is therefore 156 yielding an estimate of the minimum value of the surface permeability of 292 millidarcy. Based on the form assumed for the permeability profile the permeability at the bottom of the slab is then estimated at 29 millidarcy. The latter value appears somewhat high, but not implausible.

The Cumali Geothermal System in Turkey

Reviewing the geothermal literature, we find that the Cumali geothermal field in Turkey present a favorable setting where the

application of our above results is quite straightforward. This case has the additional advantage that a considerable number of boreholes have been drilled in the geothermal anomaly and that therefore good shallow temperature data are available. The area has been described by Eşder and Şimşek (1976).

A geolocal section through the area is given in Figure 39 which shows the general block structure and the location of some of the boreholes. Temperature data from some of the boreholes are shown in Figure 40.

Assuming that our results with regard to thermal convection in tilted slabs with heat conducting boundary walls (page 42) are applicable to the Cumali system, we have carried out a numerical analysis of the temperature field in the main convective block there. Omitting details of the calculation, we commence by deriving the temperature field corresponding to a few values of the Rayleigh number. Comparing the results with the actual borehole temperature data, we then select the most probable Rayleigh number and use this number to carry out our complete numerical computation of the temperature field.

The various temperature profiles corresponding to Rayleigh numbers of 40 to 60 are shown in Figure 40. The final temperature field results that are obtained on the basis of a Rayleigh number of $Ra = 50.4$ and a permeability of 10 millidarcy are shown in Figure 41. The end results appear quite plausible.
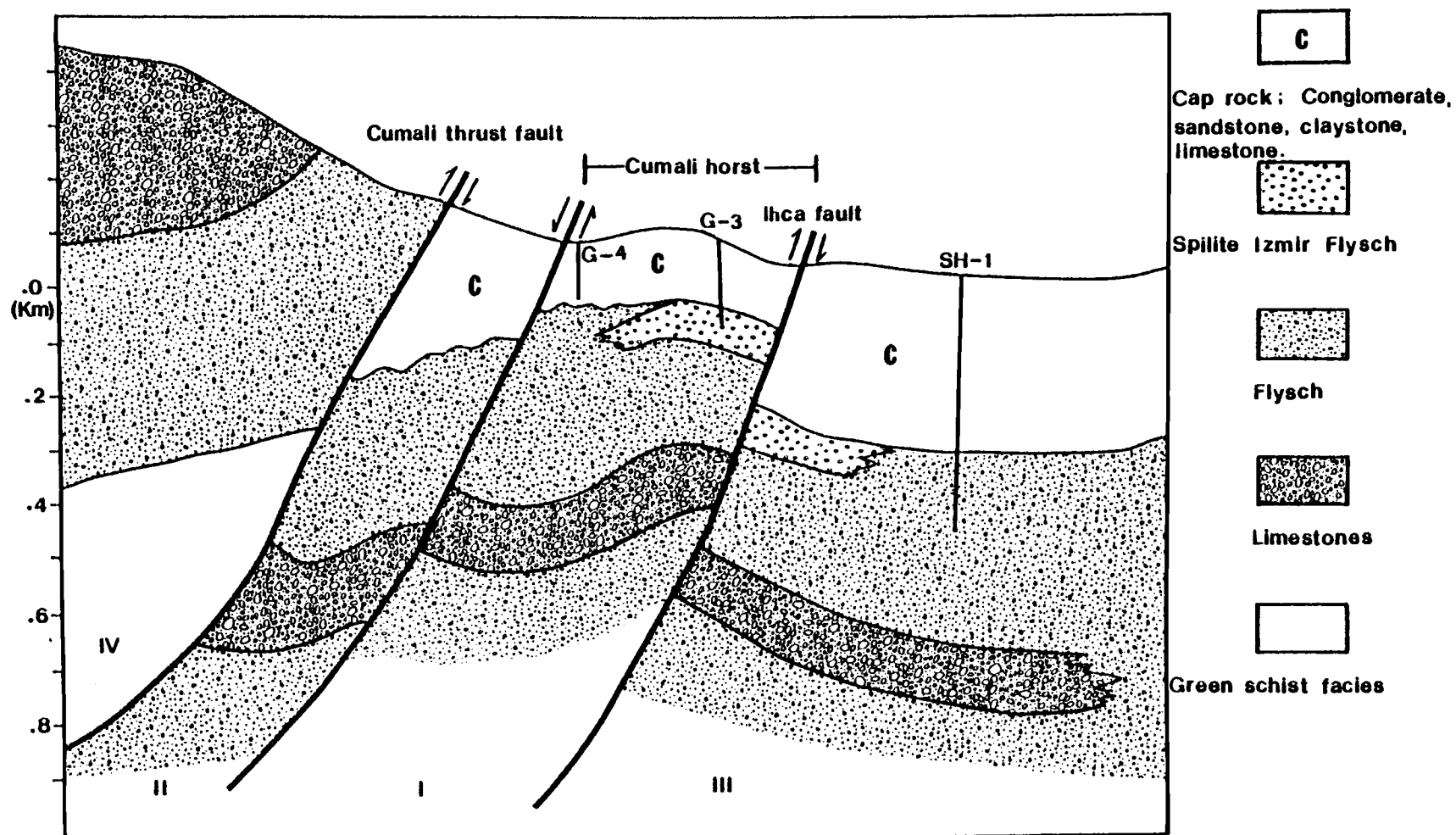
Figure 39. Geological crossection of Cumali geothermal field (only relevant details are portrayed from Eşder and Şimşek, 1975, Figure 6, page 354).
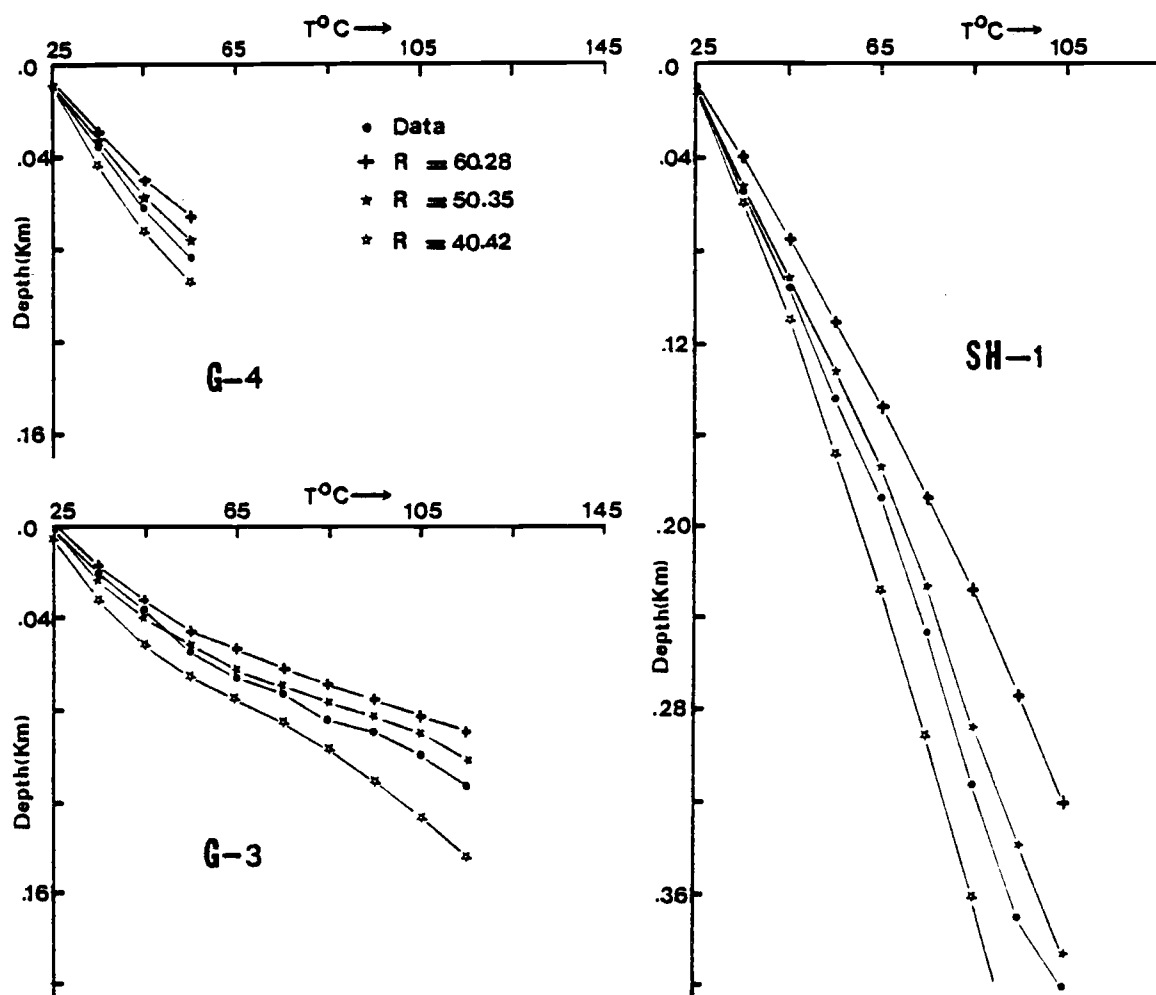
Figure 40. Comparison of the calculated temperature for various
Rayleigh numbers (+, ★, ✫) with data (•) from bore-
holes G-4, G-3, and SH-1.
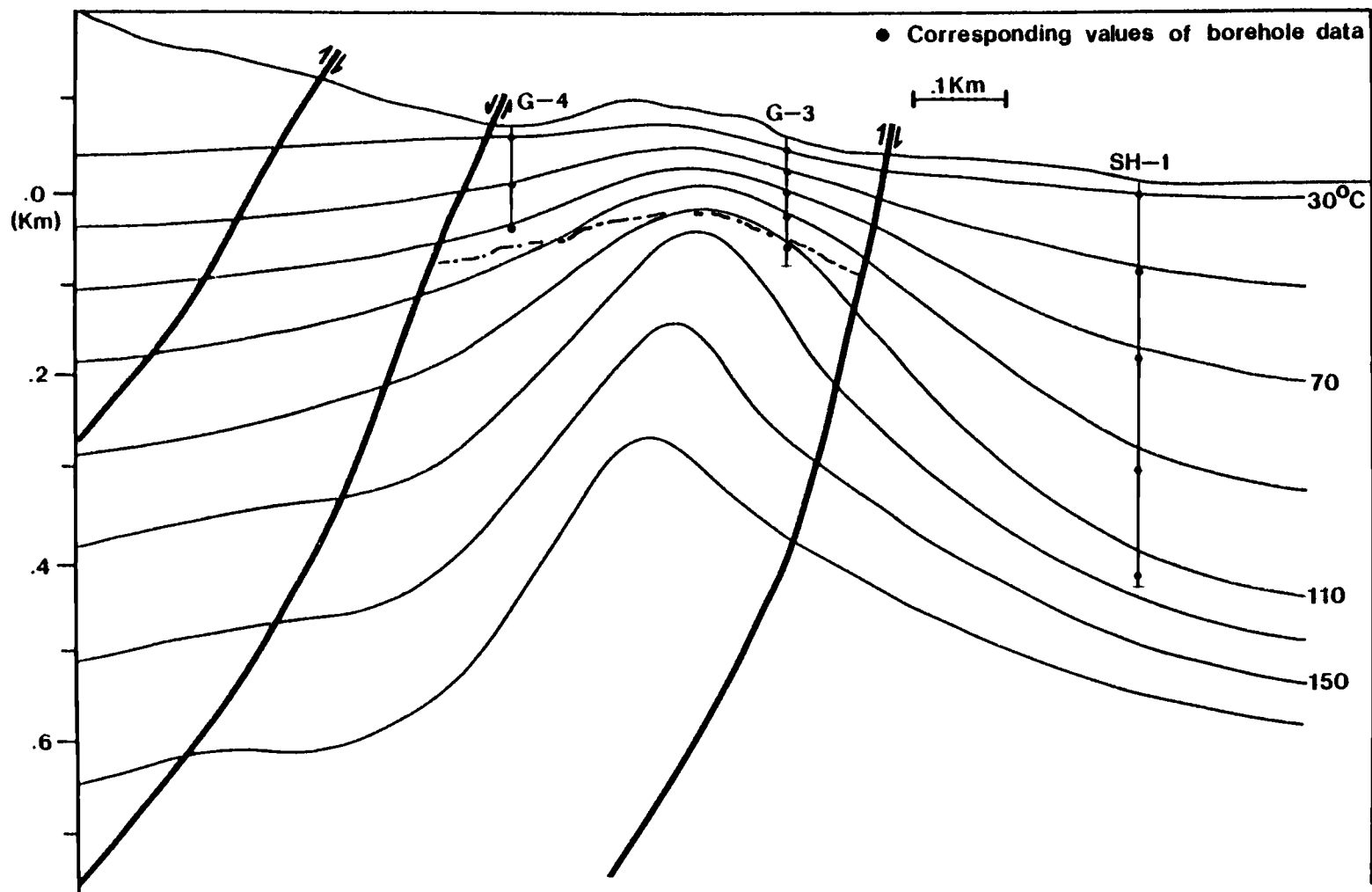
Figure 41. Calculated temperature field in Cumali district with the help of data information from G-4, G-3, and SH-1.

BIBLIOGRAPHY

Bear, J., Dynamics of Fluids in Porous Media, 764 pp., American Elsevier Publishing Co., Inc., New York, 1972.

Beck, J. L., Convection in a box of porous material saturated with fluid, The Physics of Fluids, 15:1377-1383, 1972.

Biehler, S., R. L. Kovach, and C. R. Allen, Geophysical framework of northern end of Gulf of California structural province, in Marin Geology of Gulf of California, ed. by T. V. Adel and G. Shor, Am. Assoc. Petroleum Geologists Mem., 3:126-143, 1964.

Bird, R. B., W. E. Stewart, and E. N. Lighfoot, Transport Phenomena, 780 pp., John Wiley and Sons, Inc., New York, 1960.

Bodvarsson, G., Confined fluids as strain meters, J. Geophys. Res., 75:2711-2718, 1970.

Chan, B. K., C. M. Ivey, and J. M. Barry, Natural convection in enclosed porous media with rectangular boundaries, J. Heat Transfer, 89:295-299, 1970.

Chandrasekhar, S., Hydrodynamic and Hydromagnetic Stability, 652 pp., Oxford Univ. Press, Amen House, London, 1961.

Desai, C. S., and J. F. Abel, Introduction to the Finite Element Method, 117 pp., Van Nostrand Reinhold, New York, 1972.

Desai, C. S., and D. N. Contractor, Flow, diffusion, and salt water intrusion in porous media, in Formulation and Computational Algorithms in Finite Element Analysis, ed. by K. J. Bath, T. Oden, and W. Wunderlich, Cambridge, 1977.

Donea, J., On the accuracy of finite element solutions to the transient heat-conduction equations, Int. J. for Numerical Methods in Engineering, 8:103-110, 1974.

Douglas, J., Jr., and T. Dupont, Galerkin methods for parabolic equations, SIAM J. Numer. Anal., 7:575-626, 1970.

Elder, J. W., Transient convection in a porous medium, J. Fluid Mech., 27:606-623, 1967.

Eşder, T., and Ş. Şimşek, Geology of İzmir-Seferihisar geothermal area, Western Anatolia of Turkey; determination of reservoirs by means of gradient drilling, in Proceedings of Second United Nationsl Symposium on the Development and use of Geothermal Resources, San Francisco, 1975.

Faust, C.R., and J.W. Mercer, Mathematical modeling of geothermal systems, in Proceedings of the Second United Nations Symposium on the Development and Use of Geothermal Resources, San Francisco, 1975.

Fischer, K., On the calculation of higher derivatives in finite elements, Computer Methods Appl. Mech. Eng., 7:323-330, 1976.

Forray, M.J., Variational Calculus in Science and Engineering, 193 pp., McGraw-Hill, New York, 1968.

Garg, S.K., J.W. Pritchett, and Jr. D.H. Brownell, Transport of mass and energy in porous media, in Proceedings of the Second United Nations Symposium on the Development and Use of Geothermal Resources, San Francisco, 1975.

Gray, W.G., and G.F. Pinder, Galerkin approximation of the time derivative in the finite element analysis of groundwater flow, Water Resources Res., 10:821-828, 1974.

Green, D.W., Heat Transfer with Flowing Fluid through Porous Media, Ph.D. thesis, Univ. of Oklahoma, 1963.

Hammer, P.C., O.J. Marlowe, and A.H. Stroud, Numerical integration over simplexes and cones, Math. Table Aids Comput., 10:130-137, 1956.

Hammer, P.C., and A.H. Stroud, Numerical evaluation of multiple integrals, Math. Tables Aids Comput., 12:272-280, 1958.

Holst, P.H., and K. Aziz, Transient three-dimensional natural convection in confined porous media, Int. J. Heat Mass Transfer, 15:73-90, 1972.

Horton, C.W., and F.T. Rogers, Jr., Convection currents in a porous medium, J. Appl. Phys., 16:367-370, 1945.

Kantorovich, L.V., and V.I. Krylov, Approximate Methods in Higher Analysis, 262 pp., translated by C.D. Benster, John Wiley-Interscience, New York, 1958.

Kassoy, D.R., and A. Zebib, Variable viscosity effects on the onset of convection in porous media, The Physics of Fluids, 18:1649-1651, 1975.

Katto, Y., and T. Masuoka, Criterion for the onset of convective flow in a fluid in a porous medium, Int. J. Heat Mass Transfer, 10:297-309, 1967.

Köhler, W., and J. Pitter, Calculation of transient temperature fields with finite elements in space and time dimensions, Int. J. for Numerical Methods in Engineering, 8:625-631, 1974.

Lagarde, A., Considerations sur le transfert de chaleur en milieu poreux, Revue Inst. Fr. Petrole, 20:384-446, 1965.

Lapwood, E.R., Convection of a fluid in a porous medium, Proceedings of the Cambridge Philosophical Society, 44:508-521, 1948.

Larock, B.E., and L.R. Herrmann, Improved flux prediction using low order finite elements, Finite Elements in Water Resources, ed. by W.G. Gray and G.F. Pinder, Pentech Press Limited, London, 1977.

Lasseter, T.J., P.A. Witherspoon and M.J. Lippmann, Multiphase multidimensional simulation of geothermal reservoirs, in Proceedings of the Second United Nations symposium on the Development and Use of Geothermal Resources, San Francisco, 1975.

Lippmann, M.J., T.N. Narasimhan, and P.A. Witherspoon, Numerical simulation of reservoir compaction in liquid dominated geothermal systems, presented at the Second Internal Symposium on Land and Subsidence, Anaheim, California, 1976.

Lippmann, M.J., C.F. Tsang, and P.A. Witherspoon, Analysis of the response of geothermal reservoirs under injection and production procedures, in 47th Annual California Regional Meeting of the Society of Petroleum Engineers of AIME., Bakersfield, California, 1977.

Lofgren, B. F. , Measuring ground movement in geothermal areas of Imperial Valley, California, in Proceedings Conference on Research for the Development of Geothermal Energy Resources, Pasadena, California, 1974.

Lowell, R. P. , and C. T. Shyu, On the onset of convection in a water-saturated porous box: effects of conducting walls, Letters in Heat and Mass Trans., 5:371-378, 1978.

Meissner, U. , A mixed finite element model for use in potential flow problems, Int. J. for Numerical Methods in Engineering, 6:467-473, 1973.

Mercer, J. W. , and G. F. Pinder, Galerkin-finite element simulation of a geothermal reservoir, Geothermics, 2:81-89, 1973.

Mercer, J. W. , G. F. Pinder, and I. G. Donaldson, A Galerkin-finite element analysis of the hydrothermal system at Wairakei, New Zealand, J. Geophys. Res., 80:2608-2621, 1975.

Meyer, C. A. , R. B. McClintock, G. J. Silverstri, and R. C. Spencer, 1967 ASME Steam Tables (2nd ed.), 329 pp. , The American Society of Mechanical Engineers, New York, 1968.

Nield, D. A. , Onset of thermohaline convection in a porous medium, Water Resources Res., 4:553-559, 1968.

Oden, J. T. , and H. J. Brauchli, On the calculation of consistent stress distributions in finite element approximations, Int. J. for Numerical Methods in Engineering, 3:317-325, 1971.

Pinder, G. F. , and E. O. Frind, Application of Galerkin's procedure to aquifer analysis, Water Resources Res., 8:108-120, 1972.

Pinder, G. F. , E. O. Frind, and S. S. Papadopulos, Functional coefficients in the analysis of groundwater flow using finite elements, Water Resources Res. , 9:222-226, 1973.

Pinder, G. F. , A Galerkin-finite element simulation of groundwater contamination on Long Island, New York, Water Resources Res., 9:1657-1669, 1973.

Pinder, G. F. , and W. G. Gray, Finite Element Simulation in Surface and Subsurface Hydrology, 295 pp. , Academic Press, New York, 1977.

Price, H.S., J.C. Cavendish, and R.S. Varga, Numerical methods of higher-order accuracy for diffusion-convection equations, Soc. Petroleum Engineers, J., 8:293-303, 1968.

Rayleigh L., On convection currents in a horizontal layer of fluid, when the higher temperature is on the under side, Phil. Mag., 32:529-546, 1916.

Schowalter, W.R., Stability criteria for miscible displacement of fluids from a porous medium, Amer. Inst. Chem. Engineers J., 11:99-105, 1965.

Schroeder, R.C., Reservoir Engineering Report for the Magma-SDG&E Geothermal Experimental Site Near the Sulton Sea, California, UCRL-52094(LBL), July 1976.

Segol, G., G.F. Pinder, and W.G. Gray, A Galerkin-finite element technique for calculating the transient position of the salt water front, Water Resources Res., 2:343-347, 1975.

Sorey, M.L., Numerical Modeling of Liquid Geothermal Systems, USGS Open File Report No. 75-613, Menlo Park, California, 1976.

Straus, J.M., and G. Schubert, Thermal convection of water in a porous medium: effects of temperature- and pressure-dependent thermodynamic and transport properties, J. Geophys. Res., 82:325-333, 1977.

Sutton, F.M., Stability of Flow in a Model of a Hydrothermal System, M.S. thesis, Victoria Univ., Wellington, New Zealand, 1969.

Swanberg, C.A., The Mesa geothermal anomaly, Imperial Valley, California, in A Comparison and Evaluation of Results Obtained from Surface Geophysics and Deep Resources, San Francisco, California, 1976.

Thirriot, C., R. Gaudu, F. Darbois, and D. Salomon, Use of finite elements in the analysis of a hydrothermal doublet, in Finite Element Method in Fluid Dynamic, ed. by R.H. Gallagher and others, Wiley, New York, 1974.

Wooding, R. A. , Steady state free thermal convection of fluid in a saturated permeable medium, J. Fluid Mech. , 2:273-285, 1957.

Wooding, R. A. , An experiment of free thermal convection of water in a saturated permeable medium, J. Fluid Mech. , 3:582-599, 1958.

Wooding, R. A. , The stability of a viscous liquid in a vertical tube containing porous material, in Proceedings of the Royal Society, London, 252:120-134, 1959.

Zebib, A. , and D. R. Kassoy, Onset of natural convection in a box of water-saturated porous media with large temperature variation, The Physics of Fluids, 20:4-9, 1977.

Zebib, A. , and D. R. Kassoy, Three-dimensional natural convection motion in a confined porous medium, The Physics of Fluids, 21:1-3, 1978.

Zienkiewicz, O. C. , and C. J. Parekh, Transient field problems: two dimensional and three dimensional analysis by Isoparametric finite elements, Int. J. for Numerical Methods in Engineering, 2:61-71, 1970.

Zienkiewicz, O. C. , The Finite Element Method in Engineering Science, 521 pp. , McGraw-Hill, New York, 1971.

APPENDICES

APPENDIX A

Imposed Boundary Conditions Due to the Elimination of Velocity u

To eliminate the velocity u by the mainpulation of equations (3.4), (3.5), and (3.6) resulting in equation (3.9), the natural boundary condition of u has to be incorporated with (3.7) and (3.9). Referring to the geometric configuration in Figure 5 for a porous slab with a dip angle ϕ, the boundary condition of u at wall 1 and 2 is

$$u = -w \cot \phi. \qquad (A.1)$$

Where u and w are the horizontal and the vertical component of the velocity. Eliminating pressure p from (3.5) and (3.6), then making use of (3.5) we have

$$\frac{\mu}{K} \frac{\partial w}{\partial x} + g\beta\rho_f \frac{\mu}{K} u + \alpha\rho_f g \frac{\partial \theta}{\partial x} - \frac{\mu}{K} \frac{\partial u}{\partial z} - u \frac{\partial}{\partial z} \left(\frac{\mu}{K} \right) = 0 \qquad (A.2)$$

At the boundary walls 1 and 2, (A.1) is substituted into (A.2)

$$\frac{\partial w}{\partial x} = - \frac{\alpha\rho_f g K}{\mu} \frac{\partial \theta}{\partial x} - \cot \phi \frac{\partial w}{\partial z} - \frac{Kw}{\mu} \cot \phi \frac{\partial}{\partial z} \left(\frac{\mu}{K} \right) + g\beta\rho_f w \cot \phi$$

The boundary condition can be reduced to dimensionless x_0, z_0, θ_0, w_0 and by the substitutions as in Chapter III,

$$\frac{\partial w_0}{\partial x_0} = -\left(\frac{\alpha_0 \rho_{f0} K_0}{\mu_0}\right) R^{1/2} \frac{\partial \theta_0}{\partial x_0}$$

$$- \cot \phi \left[\frac{\partial w_0}{\partial z_0} + \frac{K_0}{\mu_0} \frac{\partial}{\partial z_0}\left(\frac{\mu_0}{K_0}\right) w_0 - g\beta_0 \beta_s \rho_{f0} \rho_{fs} H_z w_0\right]$$

or

$$\frac{\partial w_0}{\partial x_0} = -G(z_0) R^{1/2} \frac{\partial \theta_0}{\partial x_0} - \cot \phi \cdot \left[I(z_0) w_0 + \frac{\partial w_0}{\partial z_0}\right]$$

where

$$G(z_0) = \frac{\alpha_0 \rho_{f0} K_0}{\mu_0}$$

$$I(z_0) = \frac{K_0}{\mu_0} \frac{\partial}{\partial z_0}\left(\frac{\mu_0}{K_0}\right) - g\beta_0 \beta_s \rho_{f0} \rho_{fs} H_z$$

$$R = \frac{g\alpha_s D H_z^2 \rho_{fs}^2 C_{fs} K_s}{\mu_s k_m} .$$

APPENDIX B

Thermodynamic Properties of Water

Convection of water in geothermal reservoirs and other struc-tures often involves temperatures as high as 350°C and pressures of the order of 500 bars. Kassoy and Zebib (1975), and Straus and Schubert (1977) have emphasized that the assumption of constant fluid parameters cannot be upheld under such conditions. In this Appendix we discuss briefly the effects of temperature and pressure on the properties of liquid water.

Viscosity

Although viscosity is both temperature and pressure dependent, it varies more strongly with temperature than it does with pressure. The following empirical equation for the viscosity has been suggested by Mercer et al. (1975),

$$\frac{1}{\mu} = 538 + 380A - 26A^2$$

$$A = (T-150)/100$$

where T is in °C and μ is in gm/cm, sec. This equation is 3% in error over the range T = 0°C to 300°C. In this range, water viscosity decreases by an order of magnitude between 25°C and 300°C. The effect is particularly important because it enhances convective

instability due to the reduction of the dissipative effects of viscosity. Figure 42-a gives data on the viscosity of water as a function of depth and the temperature gradient.

Density, Thermal Expansivity, Compressibility, Specific Heat and Adiabatic Gradient

The density of water decreases with temperature due to thermal expansion and increases with pressure due to compression. Several empirical equations are available for the density. A detailed discussion is given by Meyer et al. (1968) and Kinan (1968).

Figure 42-b to 42-f give data on the density, thermal expansivity, compressibility, specific heat and adiabatic gradient of water as a function of depth and the temperature gradient.
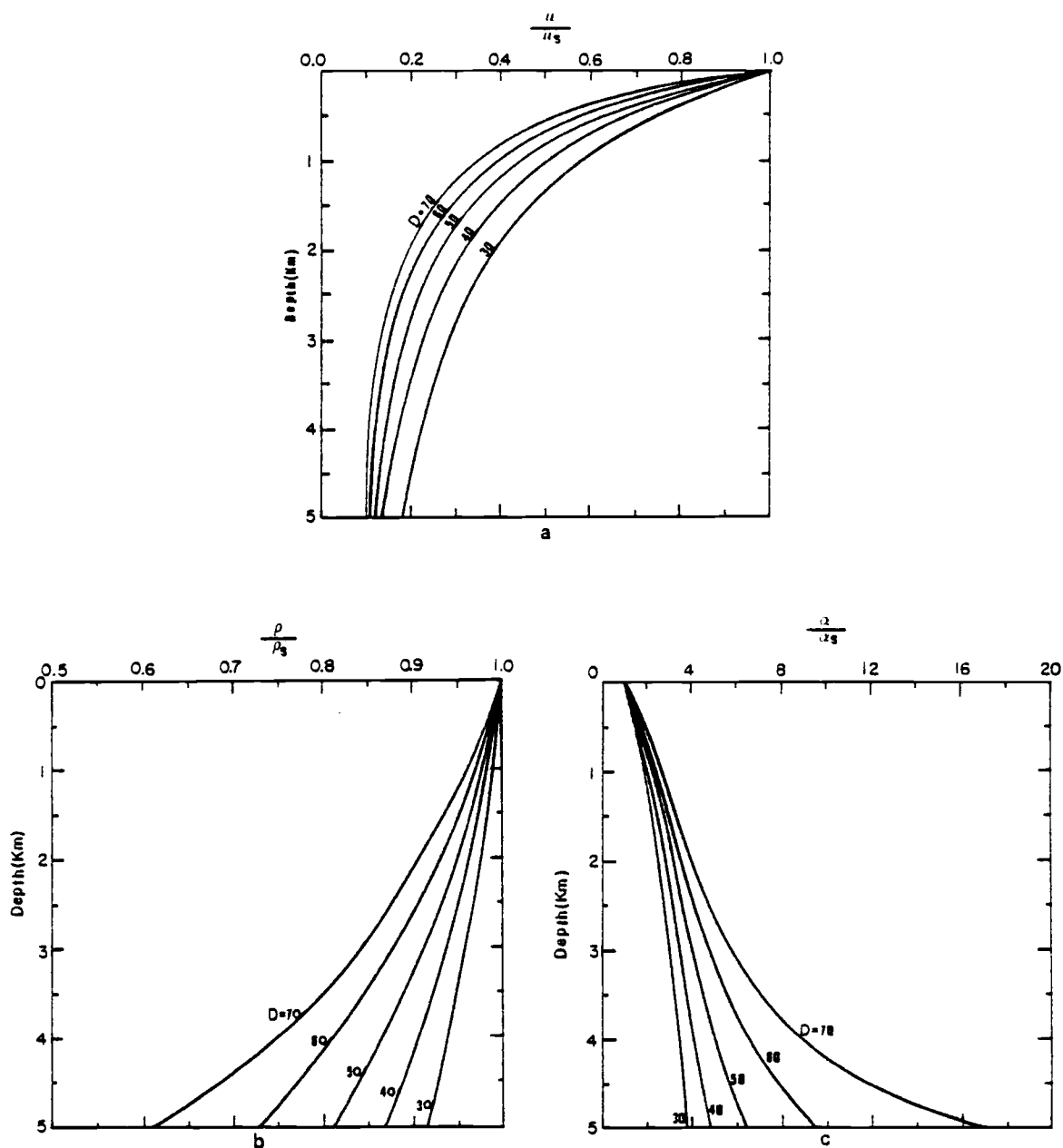
Figure 42. Water parameters as a function of depth at various temperature gradients D, assuming a surface temperature of Ts = 25°C.

μ = dynamic viscosity; ρ = density; α = thermal expansivity; β = compressibility; C_p = specific heat at constant pressure; g = gravity; T = temperature. The subscript s refers to values at the surface.
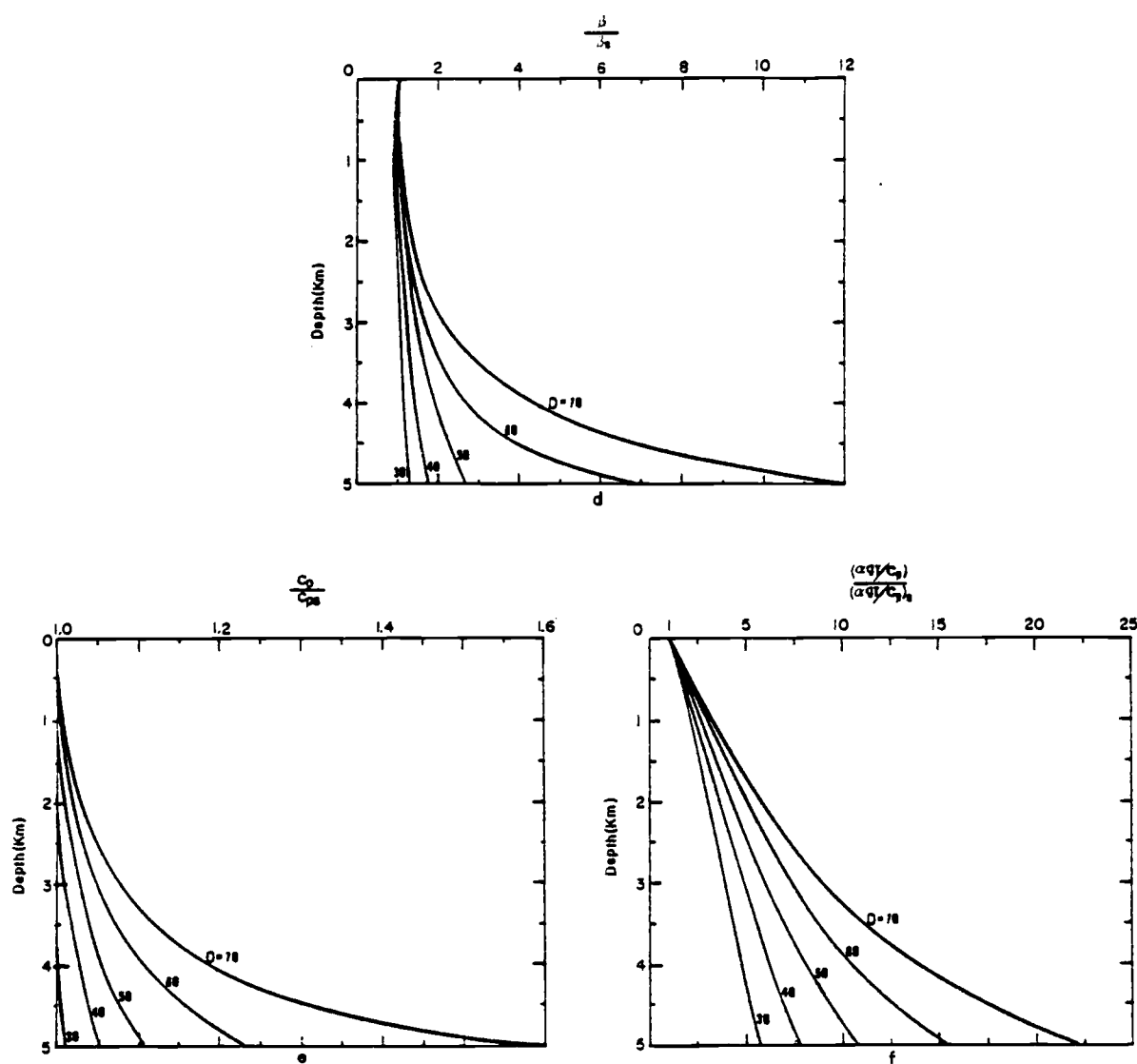
Figure 42. Continued.

APPENDIX C

Flow Charts and Program Listings for the Stability Analysis

THE OVERALL PROGRAM LOGIC

```
        ╭──────────────╮
        │    START     │
        ╰──────────────╯
               │
               ▼
        ┌──────────────┐
        │  RAYLEI      │
        │  Link BAFORM │
        │  and ORGAN   │
        └──────────────┘
               │
               ▼
     ┌──────────────────────┐
     │  Call BAFORM         │◄═══════════►( 1 )
     │  Form matrix equations│
     └──────────────────────┘
               │
               ▼
 ┌────────────────────────────────────────┐
 │  Call ORGAN                            │
 │  Form a matrix $[A] = [D]^{-1}[C][A]^{-1}[B]$,  │◄═══════════►( 2 )
 │  solve eigenvalues and eigenvectors for $[A]$  │
 └────────────────────────────────────────┘
               │
               ▼
        ┌──────────────┐
        │ Print results│
        └──────────────┘
```

```
                    ( 1 )
                      │
                      ▼
┌─────────────────────────────────────────────┐
│ Select boundary conditions, Boussinesq or non-│
│ Boussinesq approximation, permeability func-   │
│ tions, temperature gradient, Hx, Hz, dip angle │
│ etc.                                           │
└─────────────────────────────────────────────┘
                      │
                      ▼
┌─────────────────────────────────────────────┐
│ Call GRID                                      │
│ Generate quadratic element data,               │
│ locate the node numbers, subdivide             │
│ the region into elements                       │
└─────────────────────────────────────────────┘
                      │
                      ▼
┌─────────────────────────────────────────────┐
│ Call HAMER                                     │
│ Given Hammer's formula for                     │
│ numerical integration                          │
└─────────────────────────────────────────────┘
                      │
   ( 3 )──────────────┤
                      ▼
              ◇ Boussinesq                    ┌──────────────────────────┐
              ◇ approximation?  ──No──►       │ Call EFGHI               │
                      │                        │ Read coefficients of E(z),│
                    Yes                        │ F(z), G(z), H(z), I(z)   │
                      │                        └──────────────────────────┘
                      │◄──────────────────────────────┘
                      ▼
┌─────────────────────────────────────────────┐
│ Compute element matrices by using Hammer's     │
│ formula, function BA2 and function DXZ2         │
└─────────────────────────────────────────────┘
                      │
                      ▼
```

```
            ┌─────────────────────────┐
            │  Add element matrices   │
            │  to global matrices     │
            └─────────────────────────┘

                    Are
        No      all elements
   ( 3 )◄───    assembled ?

                    Yes

   ┌──────────────────────────────────────┐
   │ Modify system equations to account for│
   │ boundary conditions                    │
   └──────────────────────────────────────┘

   ┌──────────────────────────────────────┐
   │ Save information of total nodes, band │
   │ width, boundary points and all four   │
   │ global matrices on disk file TAPE2    │
   └──────────────────────────────────────┘
```

```
                    ( 2 )
                      │
                      ▼
  ┌──────────────────────────────────────┐
  │ Call READ                            │
  │ Obtain matrices [A], [B], [C],       │
  │ [D] from previous saved file         │
  │ TAPE2                                │
  └──────────────────────────────────────┘
                      │
                      ▼
┌──────────────────────────────────────────┐
│ Call LINV1F,  Call MULTI                 │
│ For matrix inversion and matrix multi-   │
│ plication until                          │
│ [A] = [D]⁻¹[C][A]⁻¹[B] if formed         │
└──────────────────────────────────────────┘
                      │
                      ▼
      ┌──────────────────────────────────┐
      │ Call EIGRF                       │
      │ Calculate eigenvalues and        │
      │ eigenvectors                     │
      └──────────────────────────────────┘
```

Call READ
Obtain matrices [A], [B], [C], [D] from previous saved file TAPE2

Call LINV1F, Call MULTI
For matrix inversion and matrix multiplication until
$[A] = [D]^{-1}[C][A]^{-1}[B]$ if formed

Call EIGRF
Calculate eigenvalues and eigenvectors

Definitions of Important Symbols and Explanations of
Program Listings

The following program names appear in the order in which they are called.   Parameters will not be defined if already defined in the previous programs unless they need to be redefined.   Numbers in parentheses represent the corresponding commentary locations in the original program listings.

Main program RAYLEI
Function: link subroutines BAFORM and ORGAN, solve eigenvalues and eigenvectors for the onset of natural convection in a porous slab with various conditions.

(1)   KNEW=1 repeat the calculations of critical Rayleigh number with different conditions.

                          KNEW≠1 end the execution of all programs.

Subroutine BAFORM
Function: form stiffness matrices [A], [B], [C], [D] in band model.

AR   area of an element.

ZN(i), WO(i)   coefficients and weights respectively of Hammer's formula for 1-D numerical integration (have not been used in the current programs).

ZS(i, j), W(i)   same as defined above except for 2-D.

PQR(i)   integral stations.

NG(i)   global node numbers.

XO(i), ZO(i),
X(i, j), Z(i, j)   coordinates.

NETR(i)   indexes of the most right elements.

NETL(i)	indexes of the most left elements.
Y(i, j)	global indexes of the 3 vertexes of the elements.
XC(i, j), ZC(i, j)	global coordinates of the 3 vertexes of the elements.
NT(i), NW(i)	global indexes of the boundary nodes for the temperatures and the vertical velocities respectively.
A(i, j), B(i, j), C(i, j), D(i, j)	matrices of $[A]$, $[B]$, $[C]$ and $[D]$ respectively.
E(i), F(i), G(i), H(i), RI(i)	for coefficients of $E(z)$, $F(z)$, $G(z)$, $H(z)$ and $I(z)$ respectively at 3 vertexes of an element, see also equations (3.12) to (3.15) and Appendix A.
(1)	IMP=1 for permeable top.
	IMP≠1 for impermeable top.
(2)	ITP=1 for non-Boussinesq approximation.
	ITP≠1 for Boussinesq approximation.
(3)	options for permeability functions and temperature gradients.
	IPER=1, 2, 3, 4 corresponding to permeability functions of (a), (b), (c), (d) in Table 5 respectively.
	GRAD=temperature gradient.
(4)	Hz=depth; Hx=horizontal dimensions; ZTA=dip angle.
(5)	see subroutine GRID.
IB2	maximum row dimension of a matrix.
(6)	see subroutine HAMER.
(7)	see subroutine EFGHI.

(8)   see function DXZ2.

(9)   see function BA2.

(10)   save NOD, IB2, NBCT, NBCW, NT(i), NW(i) and
                       matrices [A], [B], [C], [D] in disk file TAPE2.

Function BA2
Function: represent quadratic shape functions N_i (i=1,2...6) by
            area coordinate L_i (i=1,2,3) at stations P, Q, R.

IEA   index of shape function N_i (e.g. IEA=5 for N_5).

P, Q, R   stations for numerical integration.

Function DXZ2
Function: form derivatives of quadratic shape function N_i
            (i=1,2...6) in terms of area coordinate L_1 (i=1,2,3) at
            stations P, Q, R.

ND   ND=1 for $\partial N_i/\partial x$; ND=-1 for $\partial N_i/\partial z$.

Subroutine EFGHI
Function: obtain values of coefficients E(z), F(z), G(z), H(z) and
            I(z) at each node.

KLOP   number of intervals in Hz to approximate E(z),
                       F(z), G(z), H(z) and I(z).

(1)   file TAPE16 has been derived separately by pro-
                       gram EFGH, see program EFGH for details.

Subroutine HAMER
Function: provide stations and weights for numerical integrations.

AN(i), WO(i)   for 1-D element.

ZS(i), W(i)   for triangular element.

Subroutine GRID
Function: automatically generates element data for the system.

Y(i, j)   global indexes for vertexes of triangular elements.

XC(i, j), ZC(i, j)   global coordinates for vertexes of triangular elements.

NETR(k), NETL(K)   indexes of the most right and most left element respectively.

Subroutine ORGAN
Function: forms a matrix $[A] = [D]^{-1} [C] [A]^{-1} [B]$ and solve its eigenvalues and eigenvectors.   [A], [B], [C] and [D] are defined in Chapter III.

(1)   IJOB=1 for eigenvalues and eigenvectors.

                            IJOB≠1 for eigenvalues only.

(2)   see subroutine READ.

(3)   LINV1F is one of the IMSL library subroutines in CYBER 7300 system for matrix inversion, for details see IMSL manual.

(4)   see subroutine MULTI.

(5)   EIGRF is one of the IMSL library subroutines in CYBER 7300 system for eigenvalue and eigenvector solutions, for details see IMSL manual. Similar routines can be found in Matrix Eigensystem Routines - EISPAC guide (Smith et al., 1974).[2]

(6)   MW=sequence index of the maximum eigenvalue. ST=RC=critical Rayleigh number.

(7)   IVEL=1 for velocity calculations otherwise skipped.

---

[2] B. T. Smith and others, Matrix Eigensystem Routines - EISPACK Guide, 551 pp., Springer-Verlag, Berlin, 1974.

Subroutine READ
Function: read in from file TAPE2.

IRD   =1 for matrix [A].

                             =2 for matrix [B].

                             =3 for matrix [C].

                             =4 for matrix [D].

M, N   column and row dimensions respectively of the
                             matrix to be read in.

Subroutine MULTI
Function: multiplication of two matrices [A] and [B] i.e.
            [A]=[A] x [B].

MV, NV   column dimensions of [A] and [B] respectively.

MH, NH   row dimensions of [A] and [B] respectively.

Main program EFGH
Function: prepare data for functional coefficients E(z), F(z), G(z),
            H(z), and I(z) in program RAYLEI.

E(i), F(i), G(i),
H(i), RI(i)   magnitude of E(z), F(z), G(z), H(z) and I(z) at
                             z=(i)x(DEPTH)/(KLOP-1).

Subroutine PROPT
Function: water properties of ρ_f, α, β, C_f, and μ are derived from
            the empirical equations given by Meyer et al. (1968) and
            Mercer et al. (1975) if temperature and pressure are
            known.

T1   temperature (°C).

P1   pressure (bar),

(1)   RO=ρ_f, AP=α, BA=β, CP=C_f, UE=μ,
                             PER=permeability.

```
      RAYLEI                  3/ 2/79  17: 8:13     PAGE  1
 1;      PROGRAM RAYLEI(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE5,TAPE16)
 2;C     MAIN PROGRAM TO LINK SUBROUTINES "BAFORM" & "ORGAN"
 3;C     TO SOLVE CRITICAL RAYLEIGH NUMBERS FOR THE ONSET OF NATURAL
 4;C     CONVECTION IN A SLAB
 5;C     ***WRITTEN BY C. T. SHYU*** JAN. 1978***
 6;C
 7;C     ---TO FORM A BAND MATRIX FOR THE SYSTEM---
 8;    10 CALL BAFORM
 9;C     ---TO SOLVE CRITICAL RAYLEIGH NUMBERS(RAC)---
10;      CALL ORGAN
11;      WRITE 18
12;    18 FORMAT(1X,"SOLVING RAC FOR DIFFERENT CONDITIONS? (YES=1)")
13;      READ 19,KNEW        ⟵————————————————————————( 1 )
14;    19 FORMAT(I1)
15;      IF(KNEW.NE.1)GO TO 20
16;      GO TO 10
17;    20 CONTINUE
18;      END
```

```
         BAFORM               3/ 2/79  17: 8.39    PAGE  1
  1;        SUBROUTINE BAFORM
  2;C        CONVECTION IN A POROUS MEDIUM OF FAULT ZONES
  3;         COMMON AR,ZN(7),ZS(7,3),W(7),WD(7),PQR(4)
  4;         COMMON NG(6),X0(5),Z0(5),X(5,5),Z(5,5),NETR(6),NETL(6)
  5;         COMMON Y(36,3),XC(36,3),ZC(36,3),NT(38),NW(38)
  6;         COMMON A(85,53),B(85,53),C(85,53),D(85,53)
  7;         COMMON E(3),F(3),G(3),H(3),RI(3)
  8;         INTEGER Y
  9;C
 10;         WRITE 78
 11;    78 FORMAT(1X,"PERMEABLE TOP?(YES=1)")
 12;         READ 29,IMP                                            ( 1 )
 13;         WRITE 29,IMP
 14;         WRITE 68
 15;    68 FORMAT(1X,"CONSIDERING TEMPERATURE&PRESSURE EFFECTS?(YES=1)")
 16;         READ 29,ITP                                            ( 2 )
 17;    29 FORMAT(I1)
 18;         WRITE 29,ITP
 19;         IF(ITP.NE.1)GO TO 335
 20;         WRITE 58
 21;    58 FORMAT("PERM. CASES=? (1,2,3,OR 4),GRADIENT(OC/KM)=?(F5.0)")
 22;         READ 59,IPER,GRAD                                      ( 3 )
 23;    59 FORMAT(I1,F5.0)
 24;         WRITE 59,IPER,GRAD
 25;   335 WRITE 18
 26;    18 FORMAT(1X,"HZ,HX(KM),DIP(DEG.)=? (3F5.0)")
 27;         READ 19,HZ,HX,ZTA                                      ( 4 )
 28;    19 FORMAT(3F5.0)
 29;         WRITE 57,ITP,HZ,HX,ZTA
 30;         ZT=ZTA/180.*3.1415926536
 31;    57 FORMAT(1X,"ITP=",I1,1X,"HZ=",F6.3,1X,"HX=",F6.3,"ZTA=",F6.3)
 32;         HXHZ=HX/HZ
 33;         CALL GRID(IMP,IGD,IV,HXHZ,ZTA,NET,IB,NOD,NBCT,NBCW)    ( 5 )
 34;C
 35;         CALL HAMER                                             ( 6 )
 36;         IB2=2*IB-1
 37;         DO 585 I=1,NOD
 38;         DO 585 J=1,IB2
 39;   585 A(I,J)=B(I,J)=C(I,J)=D(I,J)=0.
 40;         KME=1
 41;         DO 1000 KE=1,NET
 42;         IZA=1
 43;         DO 73 I=1,3
 44;         NG(I)=Y(KE,I)
 45;         X0(I)=XC(KE,I)
 46;         Z0(I)=ZC(KE,I)
 47;         IF(ITP.NE.1)GO TO 73
 48;         ZHEW=Z0(I)
 49;         CALL EFGHI(GRAD,IPER,ZHEW,HZ,EE,FE,GE,HE,RIE,KME)      ( 7 )
 50;         KME=10
 51;         E(I)=EE
 52;         F(I)=FE
 53;         G(I)=GE
 54;         H(I)=HE
 55;         RI(I)=RIE
 56;    73 CONTINUE
 57;C    28 FORMAT(1X,I2,I2,I2,1X,F9.6,F9.6,F9.6,F9.6,F9.6,F9.6)
 58;C        CALCULATING THE ELEMENT STIFFNESS MATRIX IN ONE ELEMENT
```

```
       BAFORM                    3/ 2/79  17: 8.39      PAGE  2
59;      XO(4)=XO(1)
60;      XO(5)=XO(2)
61;      ZO(4)=ZO(1)
62;      ZO(5)=ZO(2)
63;      NG(4)=(NG(1)+NG(2))/2
64;      NG(5)=(NG(2)+NG(3))/2
65;      NG(6)=(NG(1)+NG(3))/2
66;      AR=(ZO(2)*XO(3)+ZO(1)*XO(2)+XO(1)*ZO(3)-XO(1)*ZO(2)-XO(2)*ZO(3)
67;    0-XO(3)*ZO(1))/2.
68;      AR=ABS(AR)
69;      DO 10 I=1,5
70;      DO 10 J=1,5
71;      X(I,J)=XO(I)-XO(J)
72;   10 Z(I,J)=ZO(I)-ZO(J)
73;      DJACO=Z(1,3)*X(2,3)-Z(2,3)*X(1,3)
74;      IF(ZTA.EQ.90.)GO TO 32
75;      DO 55 INC=1,6
76;      IF(KE.EQ.NETL(INC))IZA=2
77;   55 IF(KE.EQ.NETR(INC))IZA=3
78;   32 DO 25 I=1,6
79;      DO 15 J=1,6
80;      SM1=SM2=SM3=SM4=SM5=0.
81;      DO 16 KW=1,7
82;      P=ZS(KW,1)
83;      Q=ZS(KW,2)
84;      R=ZS(KW,3)
85;      STP1=0.
86;      STP2=0.
87;      STP3=1.
88;      STP4=1.
89;      STP5=0.
90;      IF(ITP.NE.1)GO TO 17
91;      STP1=P*E(1)+Q*E(2)+R*E(3)
92;      STP2=P*F(1)+Q*F(2)+R*F(3)
93;      STP3=P*G(1)+Q*G(2)+R*G(3)
94;      STP4=P*H(1)+Q*H(2)+R*H(3)
95;      STP5=P*RI(1)+Q*RI(2)+R*RI(3)
96;   17 SM1=DXZ2(1,I,P,Q,R)*DXZ2(1,J,P,Q,R)+DXZ2(-1,I,P,Q,R)
97;    0*DXZ2(-1,J,P,Q,R)          <──────  (8)
98;      SM2=-STP1*BA2(I,P,Q,R)*DXZ2(-1,J,P,Q,R)
99;    0-STP2*BA2(I,P,Q,R)*BA2(J,P,Q,R)   <──────  (9)
100;     SM1=SM1+(SM1+SM2)*W(KW)
101;     SM2=STP3*DXZ2(1,I,P,Q,R)*DXZ2(1,J,P,Q,R)*W(KW)+SM2
102;     SM3=STP4*BA2(I,P,Q,R)*BA2(J,P,Q,R)*W(KW)+SM3
103;     SM4=SM4+SM1*W(KW)
104;     GO TO(16,26,27),IZA
105;   26 SL1=SQRT(XO(1)**2+ZO(1)**2)
106;     SL2=SQRT(XO(2)**2+ZO(2)**2)
107;     SLD=SL2-SL1
108;     GO TO 24
109;   27 SL1=SQRT((XO(1)-HXHZ)**2+ZO(1)**2)
110;     SL2=SQRT((XO(3)-HXHZ)**2+ZO(3)**2)
111;     SLD=SL1-SL2
112;   24 SLP=SL1+SL2
113;     SK=0.5*(SLP+SLD*ZN(KW))
114;     SLI=(SLD-SK)/SLD
115;     SLJ=SLK=1.-SLI
116;     GO TO(36,36,37),IZA
```

```
        BAFORM                    3/ 2/79  17: 8.39     PAGE  3
117;    36 SLK=0.
118;       GO TO 33
119;    37 SLJ=0.
120;    33 SM5=SM5+STP5*BA2(J,SLI,SLJ,SLK)*BA2(I,SLI,SLJ,SLK)*WO(KW)
121;    16 CONTINUE
122;       IG=NG(I)
123;       JG=NG(J)
124;       JB=JG-IG+IB
125;       A(IG,JB)=SM1*DJACO+A(IG,JB)+0.5*SLD*SM5*(-1)**(IZA-1)*COS(ZT)
126;       B(IG,JB)=SM2*DJACO+B(IG,JB)
127;       C(IG,JB)=SM3*DJACO+C(IG,JB)
128;       D(IG,JB)=SM4*DJACO+D(IG,JB)
129;    15 CONTINUE
130;    25 CONTINUE
131; 1000 CONTINUE
132;C      STIFNESS MATRIX IS FORMED WITHOUT CONCERNING B.C. YET
133;C      **NOW TO INTRODUCE THE DIRICHLET B.C.***
134;C      **B.C. FOR TEMPERATURE**
135;       DO 105 I=1,NBCT
136;       DO 115 IO=1,NOD
137;       JN=NT(I)-IO+IB
138;       IF(JN.GT.IB2.OR.JN.LE.0)GO TO 115
139;       D(IO,JN)=B(IO,JN)=1.234567E20
140;   115 CONTINUE
141;       DO 116 JN=1,IB2
142;   116 C(NT(I),JN)=D(NT(I),JN)=1.234567E20
143;   105 CONTINUE
144;C      **B.C. FOR VERTICAL VELOCITY**
145;   202 DO 205 I=1,NBCW
146;       DO 215 IO=1,NOD
147;       JN=NW(I)-IO+IB
148;       IF(JN.GT.IB2.OR.JN.LE.0)GO TO 215
149;       C(IO,JN)=A(IO,JN)=1.234567E20
150;   215 CONTINUE
151;       DO 216 JN=1,IB2
152;   216 A(NW(I),JN)=B(NW(I),JN)=1.234567E20
153;   205 CONTINUE
154;       WRITE(2,23)NOD,IB2,NBCT,NBCW
155;    23 FORMAT(1X,I2,1X,I2,1X,I2,1X,I2)
156;       DO 20 I=1,NBCT
157;    20 WRITE(2,22)NT(I)
158;    22 FORMAT(1X,I2)
159;       DO 21 I=1,NBCW
160;    21 WRITE(2,22)NW(I)
161;       DO 510 I=1,NOD
162;       DO 510 J=1,IB2
163;       JO=J+I-IB
164;       IF(JO.LE.0.OR.JO.GT.NOD)GO TO 510
165;       SM1=ABS(A(I,J))+ABS(B(I,J))+ABS(C(I,J))+ABS(D(I,J))
166;       IF(SM1.EQ.4.*1.234567E20)GO TO 510
167;       IF(SM1.GT.1.0E-12)WRITE(2,528)I,J,A(I,J),B(I,J),C(I,J),D(I,J)
168;   510 CONTINUE
169;       I=J=0
170;       SM1=SM2=SM3=SM4=0.
171;       WRITE(2,528)I,J,\M1,SM2,SM3,SM4
172;   528 FORMAT(1X,I2,1X,I2,3X,E15.8,1X,E15.8,1X,E15.8,1X,E15.8)
173;       REWIND 2
174;       RETURN
```

```
        BAFORM                    3/ 2/79  17: 8:39    PAGE  4
175;        END
176;C       **********************************************
177;        FUNCTION BA2(IBA,P,Q,R)
178;C       BA2=SHAPE FUNCTIONS INTERMS OF AREA COORDINATES(P=L1;Q=L2;R=L3)
179;        COMMON AR,ZN(7),ZS(7,3),W(7),WO(7),PQR(4)
180;        PQR(1)=P
181;        PQR(2)=Q
182;        PQR(3)=R
183;        PQR(4)=P
184;        IF(IBA.GT.3)GO TO 20
185;        BA2=2.*PQR(IBA)*PQR(IBA)-PQR(IBA)
186;        GO TO 50
187;     20 KN=IBA-3
188;        BA2=4.*PQR(KN)*PQR(KN+1)
189;     50 RETURN
190;        END
191;C       **********************************************
192;        FUNCTION DXZ2(ND,M,P,Q,R)
193;C       DXZ2=DERIVATION OF THE SHAPE FUNCTION "N" WITH RESPECT TO
194;C       "X" OR "Z" IN TERMS OF AREA COORDINATES
195;C       ND=1 FOR D(N)/DX; ND=-1 FOR D(N)/DZ
196;        COMMON AR,ZN(7),ZS(7,3),W(7),WO(7),PQR(4)
197;        COMMON NG(6),X0(5),Z0(5),X(5,5),Z(5,5),NETR(6),NETL(6)
198;        IF(AR.LE.1.0E-10.OR.AR.GT.1.0E5)WRITE 18,AR
199;     18 FORMAT(1X,"AR",E12.5)
200;        PQR(1)=P
201;        PQR(2)=Q
202;        PQR(3)=R
203;        PQR(4)=P
204;        IF(ND.EQ.-1)GO TO 10
205;        IF(M.LE.3)DXZ2=(4.*PQR(M)-1.)*Z(M+2,M+1)/(2.*AR)
206;        KN=M-3
207;        IF(M.GT.3)DXZ2=2.*(PQR(KN+1)*Z(KN+2,KN+1)+PQR(KN)*Z(KN,KN+2))/AR
208;        GO TO 50
209;     10 IF(M.LE.3)DXZ2=(4.*PQR(M)-1.)*X(M+1,M+2)/(2.*AR)
210;        KN=M-3
211;        IF(M.GT.3)DXZ2=2.*(PQR(KN+1)*X(KN+1,KN+2)+PQR(KN)*X(KN+2,KN))/AR
212;     50 CONTINUE
213;        RETURN
214;        END
215;C       **********************************************
216;        SUBROUTINE EFGHI(GRAD,IPER,ZNEW,HZ,EE,FE,GE,HE,RIE,KE)
217;C       TO GENERATE COEFS. E,F,G,H,I FOR EACH ELEMENT
218;        DIMENSION ZOLD(11),E(11),F(11),G(11),H(11),RI(11)
219;        IF(KE.GE.2)GO TO 52
220;        KLOP=11
221;        READ(16,19)D,KPER
222;     19 FORMAT(F5.2,1X,I1)
223;        DO 51 K=1,KLOP
224;     51 READ(16,18)E(K),F(K),G(K),H(K),RI(K)     <---------------(1)
225;     18 FORMAT(1X,5(E12.5))
226;        IF(D.EQ.GRAD.AND.KPER.EQ.IPER)GO TO 52
227;     52 ZOD=ZNEW*HZ
228;        ZOLD(1)=0.
229;        DO 10 I=1,10
230;        ZOLD(I+1)=0.5*I
231;        Z=(ZOD-ZOLD(I))*(ZOD-ZOLD(I+1))
232;        IF(Z.LE.0.0)GO TO 20
```

```
          BAFORM                    3/ 2/79  17: 8:39      PAGE  5
233;      10 CONTINUE
234;      20 ZPOT=(ZOD-ZOLD(I))/(ZOLD(I+1)-ZOLD(I))
235;         EE=E(I)+(E(I+1)-E(I))*ZPOT
236;         FE=F(I)+(F(I+1)-F(I))*ZPOT
237;         GE=G(I)+(G(I+1)-G(I))*ZPOT
238;         HE=H(I)+(H(I+1)-H(I))*ZPOT
239;         RIE=RI(I)+(RI(I+1)-RI(I))*ZPOT
240;         RETURN
241;         END
242;C        *************************************************
243;         SUBROUTINE HAMER
244;C        HAMMER'S FORMULA FOR NUMERICAL INTEGRATION IN ONE-D & TRIANGULAR
245;C        ELEMENT
246;         COMMON AR,ZN(7),ZS(7,3),W(7),WO(7),PQR(4)
247;         ZN(1)=0.
248;         ZN(2)=0.4058451514
249;         ZN(3)=0.7415311856
250;         ZN(4)=0.94910791234
251;         ZN(5)=-ZN(2)
252;         ZN(6)=-ZN(3)
253;         ZN(7)=-ZN(4)
254;         WO(1)=0.41795918367
255;         WO(2)=WO(5)=0.38183005051
256;         WO(3)=WO(6)=0.27970539149
257;         WO(4)=WO(7)=0.129484966169
258;         S1=0.3333333333
259;         S2=0.79742699
260;         S3=0.10128651
261;         S4=0.05971587
262;         S5=0.47014206
263;         ZS(1,1)=ZS(1,2)=ZS(1,3)=S1
264;         ZS(2,1)=ZS(3,2)=ZS(4,3)=S2
265;         ZS(2,2)=ZS(2,3)=ZS(3,1)=ZS(3,3)=ZS(4,1)=ZS(4,2)=S3
266;         ZS(5,1)=ZS(6,2)=ZS(7,3)=S4
267;         ZS(5,2)=ZS(5,3)=ZS(6,1)=ZS(6,3)=ZS(7,1)=ZS(7,2)=S5
268;         W(1)=0.1125
269;         W(2)=W(3)=W(4)=0.06296959
270;         W(5)=W(6)=W(7)=0.06619788
271;         RETURN
272;         END
273;C        *************************************************
274;         SUBROUTINE GRID(IMP,IGD,IV,HXHZ,ZTA,NET,IB,NOD,NBCT,NBCW)
275;C        ---TO GENERATE ELEMENT DATA---
276;C        ---HXHZ=HX/HZ; NET=TOTAL ELDMENTS; IB=BAND WIDTH;
277;C        ---NOD=TOTAL NGDES; NBCT=TOTAL NODES AT B.C. FOR TEMPT.;
278;C        ---NBCW=TOTAL NODES AT B.C. FOR VELOCITY "W"----
279;C        ---INPUT: IMP,IGD,HXHZ,NET---OUTPUT: IV,IB,NOD,NBCT,NBCW---
280;         COMMON AR,ZN(7),ZS(7,3),W(7),WO(7),PQR(4)
281;         COMMON NG(6),XO(5),ZO(5),X(5,5),Z(5,5),NETR(6),NETL(6)
282;         COMMON Y(36,3),XC(36,3),ZC(36,3),NT(30),NU(30),NW(30)
283;         INTEGER Y,NT,NW
284;         ZT=ZTA/180.*3.1415926536
285;         WRITE 18
286;      18 FORMAT(1X,"GRID1,GRID2,GRID3 OR GRID4?(I1)")
287;         READ 19,IGD
288;         WRITE 19,IGD
289;      19 FORMAT(I1)
290;         WRITE 20
```

```
          BAFORM                    3/ 2/79  17: 8:39    PAGE  6
291;     28 FORMAT(1X,"VERTICAL?(YES=1)")
292;        READ 19,IV
293;        WRITE 19,IV
294;        WRITE 38
295;     38 FORMAT(1X,"HOW MANY ELEMENTS? (I2)")
296;        READ 39,NET
297;        WRITE 39,NET
298;     39 FORMAT(I2)
299;C
300;        IB=11+(IGD-1)*8
301;        NOD=NET*(10+8*IGD-8)/(4*IGD)+(2*IGD+1)
302;        IF(IV-1)13,12,13
303;     12 H=HXHZ/IGD
304;        V=4.*IGD/NET
305;        GO TO 14
306;     13 H=1./IGD
307;        V=HXHZ*4*IGD/NET
308;     14 NEN=NET-(4*IGD-1)
309;        NEP=NET*2/(4*IGD)+1
310;        NJP=4*IGD
311;        NI=0
312;        DO 210 I=1,NEN,NJP
313;        NI=NI+1
314;        DO 210 J=1,IGD
315;        I1=I+4*(J-1)
316;        IM=(NI-1)*(2+8*IGD)
317;        Y(I1,1)=Y(I1+1,1)=1+2*(J-1)+IM
318;        Y(I1,2)=Y(I1+2,2)=11+(IGD-1)*8+2*(J-1)+IM
319;        Y(I1,3)=Y(I1+1,2)=Y(I1+2,1)=Y(I1+3,2)=7+4*(IGD-1)+2*(J-1)+IM
320;        Y(I1+1,3)=Y(I1+3,1)=3+2*(J-1)+IM
321;        Y(I1+2,3)=Y(I1+3,3)=13+8*(IGD-1)+2*(J-1)+IM
322;        XC(I1,1)=XC(I1+1,1)=XC(I1,2)=XC(I1+2,2)=(J-1)*H
323;        XC(I1,3)=XC(I1+1,2)=XC(I1+2,1)=XC(I1+3,2)=H*(J-1)+0.5*H
324;        XC(I1+1,3)=XC(I1+3,1)=XC(I1+2,3)=XC(I1+3,3)=J*H
325;        ZC(I1,1)=ZC(I1+1,1)=ZC(I1+1,3)=ZC(I1+3,1)=(NI-1)*V
326;        ZC(I1,3)=ZC(I1+1,2)=ZC(I1+2,1)=ZC(I1+3,2)=(NI-1)*V+0.5*V
327;        ZC(I1,2)=ZC(I1+2,2)=ZC(I1+2,3)=ZC(I1+3,3)=NI*V
328;    210 CONTINUE
329;        IF(ZTA.EQ.90.)GO TO 212
330;        DO 211 I=1,NET
331;        DO 211 J=1,3
332;        XC(I,J)=ZC(I,J)/TAN(ZT)+XC(I,J)
333;    211 CONTINUE
334;C       B.C. NODES FOR TEMPERATURE
335;    212 NBCT=2*NEP+2*(2*IGD-1)
336;        DO 220 I=1,NEP
337;        NT(I)=1+(I-1)*(4*IGD+1)
338;    220 NT(NEP+I)=NT(I)+2*IGD
339;        IGF=2*IGD-1
340;        DO 221 I=1,IGF
341;        NT(2*NEP+I)=1+I
342;    221 NT(2*NEP+IGF+I)=NOD-IGF+I-1
343;        IF(IV.NE.1)GO TO 200
344;C       B.C. NODES FOR VELOCITY "W"
345;        NBCW=2*IGF+4
346;        NWB=NBCW/2
347;        IF(IMP.EQ.1)GO TO 223
348;        DO 222 I=1,NWB
```

```
        BAFORM                    3/ 2/79  17: 8:39     PAGE  7
349;        NW(I)=I
350;    222 NW(I+NNB)=NOD-NNB+I
351;        GO TO 226
352;    223 DO 225 I=1,NNB
353;    225 NW(I)=NOD-NNB+I
354;        NBCW=NNB
355;    226 IF(ZTA.EQ.90.)GO TO 999
356;        DO 50 I=1,6
357;        NETL(I)=1+(I-1)*4*IGD
358;     50 NETR(I)=I*4*IGD
359;        GO TO 999
360;    200 IF(IMP.EQ.1)GO TO 201
361;        NBCW=2*NEP
362;        DO 224 I=1,NEP
363;        NW(I)=1+(I-1)*(4*IGB+1)
364;    224 NW(NEP+I)=NW(I)+2*IGD
365;        GO TO 202
366;    201 DO 227 I=1,NEP
367;    227 NW(I)=1+(I-1)*(4*IGB+1)+2*IGD
368;        NBCW=NEP
369;    202 DO 230 I=1,NET
370;        DO 230 J=1,3
371;        XT=XC(I,J)
372;        XC(I,J)=ZC(I,J)
373;    230 ZC(I,J)=IGD*H-XT
374;    999 RETURN
375;        END
```

```
      ORGAN                    3/ 2/79  17:12:53     PAGE  1
 1;        SUBROUTINE ORGAN
 2;        DIMENSION B(88,88)
 3;        COMMON VEL(88,88),Z(88,88),W(88),WK(168)
 4;        COMMON A(88,88),F(88,88),E(88)
 5;        COMPLEX ZN,Z,W,WK
 6;        EQUIVALENCE (Z,B)
 7;        DO 3 I=1,88
 8;        E(I)=W(I)=WK(I)=0.
 9;        DO 3 J=1,88
10;        VEL(I,J)=A(I,J)=B(I,J)=F(I,J)=Z(I,J)=0.
11;      3 CONTINUE
12;        DO 2 I=81,168
13;      2 WK(I)=0.
14;C       **TO FORM A MATRIX "A"=1/D*C*(1/A)*B & SOLVE ITS EIGENVALUES
15;        WRITE 18
16;     18 FORMAT(1X,"SOLVING FOR EIGEN VECTORS? (YES=1)")
17;        READ 19,IJOB                              <--------( 1 )
18;     19 FORMAT(I1)
19;        DO 5 I=1,88
20;        E(I)=0.
21;        DO 5 J=1,88
22;      5 A(I,J)=B(I,J)=F(I,J)=0.
23;        CALL READ(1,MV,MH)                        <--------( 2 )
24;        CALL LINV1F(A,MV,88,B,0,E,IER1)           <----
25;        CALL READ(2,NV,NH)                            |
26;        DO 10 I=1,88                                  |
27;        DO 10 J=1,88                                  |---( 3 )
28;        S=B(I,J)
29;        B(I,J)=A(I,J)
30;     10 A(I,J)=S
31;        CALL MULTI(MV,MH,NV,NH)                   <--------( 4 )
32;        DO 20 I=1,88
33;        DO 20 J=1,88
34;        VEL(I,J)=A(I,J)
35;     20 B(I,J)=A(I,J)
36;        CALL READ(3,MCV,MCH)
37;        CALL MULTI(MCV,MCH,MV,NH)
38;        DO 30 I=1,MCV
39;        DO 30 J=1,NH
40;     30 F(I,J)=A(I,J)
41;        CALL READ(4,MDV,MDH)
42;        CALL LINV1F(A,MDV,88,B,0,E,IER2)
43;        DO 35 I=1,88
44;        DO 35 J=1,88
45;        A(I,J)=B(I,J)
46;     35 B(I,J)=F(I,J)
47;        CALL MULTI(MDV,MDH,MCV,NH)
48;        CALL EIGRF(A,MDV,88,IJOB,W,Z,88,WK,IER3)  <--------( 5 )
49;        ST=0.
50;        DO 52 I=1,MDV
51;        IF(REAL(W(I)).LE.ST)GO TO 52
52;        IF(ABS(AIMAG(W(I))).GT.5.0E-18)GO TO 52
53;        ST=REAL(W(I))
54;        MW=I
55;     52 CONTINUE
56;        ST=1./ST
57;        WRITE 38,MW,ST                            <--------( 6 )
58;        ZN=Z(1,MW)
```

```
      ORGAN                    3/ 2/79  17:12:53     PAGE  2
 59;   38 FORMAT(1X,"MU=",I3,3X,"RC=",E14.7)
 60;      IF(IJOB.NE.1)GO TO 281
 61;      DO 53 I=1,MDV,3
 62;      I1=I+1
 63;      I2=I+2
 64;      I3=I+3
 65;      Z(I,MU)=Z(I,MU)/ZN
 66;      Z(I1,MU)=Z(I1,MU)/ZN
 67;      Z(I2,MU)=Z(I2,MU)/ZN
 68;   53 WRITE 58,Z(I,MU),Z(I1,MU),Z(I2,MU)
 69;   58 FORMAT(3(E11.4,1X,E9.2))
 70;  281 WRITE 28
 71;   28 FORMAT(1X,"FOR VELOCITY? (YES=1)")
 72;      READ 19,IVEL
 73;      IF(IVEL.NE.1)GO TO 2000
 74;      DO 60 I=1,MV
 75;      WK(I)=0.
 76;      DO 61 J=1,MH
 77;   61 WK(I)=WK(I)+VEL(I,J)*Z(J,MU)
 78;   60 WK(I)=-SQRT(ST)*WK(I)
 79;      DO 63 I=1,MV,3
 80;      I1=I+1
 81;      I2=I+2
 82;   63 WRITE 58,WK(I),WK(I1),WK(I2)
 83;      DO 64 K=1,5
 84;      ST=1./REAL(W(K))
 85;      WRITE(5,38)K,ST
 86;      ZN=Z(1,K)
 87;      DO 65 I=1,MDV
 88;   65 Z(I,K)=Z(I,K)/ZN
 89;      WRITE(5,648)(Z(I,K),I=1,5)
 90;      WRITE(5,648)(Z(I,K),I=6,10)
 91;      WRITE(5,648)(Z(I,K),I=11,15)
 92;      WRITE(5,648)(Z(I,K),I=16,20)
 93;      WRITE(5,648)(Z(I,K),I=21,25)
 94;      WRITE(5,648)(Z(I,K),I=26,30)
 95;      WRITE(5,648)(Z(I,K),I=31,35)
 96;      WRITE(5,648)Z(36,K)
 97;  648 FORMAT(1X,5(E11.4,1X,F9.2,1X))
 98;   64 CONTINUE
 99; 2000 CONTINUE
100;      REWIND 2
101;      REWIND 3
102;      REWIND 5
103;      REWIND 16
104;      RETURN
105;      END
106;C    ********************************************************
107;      SUBROUTINE MULTI(MV,MH,NV,NH)
108;      COMMON A(80,80),B(80,80),F(80,80),E(80)
109;C    **"A"=(A*B),        MH=NV
110;      DO 10 I=1,MV
111;      DO 11 JM=1,MH
112;   11 E(JM)=A(I,JM)
113;      DO 10 J=1,NH
114;      A(I,J)=0.
115;      DO 10 K=1,NV
116;   10 A(I,J)=A(I,J)+E(K)*B(K,J)
```

```
        ORGAN                    3/ 2/79  17:12:53    PAGE  3
117;C      **END OF MULTIPLICATION***
118;       DO 20 I=1,NV
119;       E(I)=0.
120;       DO 20 J=1,NH
121;    20 B(I,J)=0.
122;       RETURN
123;       END
124;C      -------------------------------------------------
125;       SUBROUTINE READ(IRD,M,N)
126;       DIMENSION NW(50),NT(50),ABCD(4)
127;       COMMON A(80,80)
128;       DO 50 I=1,80
129;       DO 50 J=1,80
130;    50 A(I,J)=0.
131;       REWIND 2
132;       READ(2,20)NOD,IB2,NBCT,NBCW
133;    20 FORMAT(1X,I2,1X,I2,1X,I2,1X,I2)
134;       IB=(IB2+1)/2
135;       DO 10 I=1,NBCT
136;    10 READ(2,18)NT(I)
137;    18 FORMAT(1X,I2)
138;       DO 20 I=1,NBCW
139;    20 READ(2,18)NW(I)
140;       M=N=0
141;       DO 1000 IKK=1,10000
142;       READ(2,38)I,JN,ABCD(1),ABCD(2),ABCD(3),ABCD(4)
143;    38 FORMAT(1X,I2,1X,I2,3X,E15.8,1X,E15.8,1X,E15.8,1X,E15.8)
144;       IF(I.EQ.0)GO TO 1001
145;       IF(ABCD(IRD).EQ.1.234567E20)GO TO 1000
146;       J=JN+I-IB
147;       IS=JS=0
148;       GO TO (21,22,23,24),IRD
149;    21 DO 30 K=1,NBCW
150;       IF(I.GT.NW(K))IS=IS+1
151;    30 IF(J.GT.NW(K))JS=JS+1
152;       I=I-IS
153;       J=J-JS
154;       A(I,J)=ABCD(1)
155;       IF(I.GT.M)M=I
156;       IF(J.GT.N)N=J
157;       GO TO 1000
158;    22 DO 31 K=1,NBCW
159;    31 IF(I.GT.NW(K))IS=IS+1
160;       DO 32 K=1,NBCT
161;    32 IF(J.GT.NT(K))JS=JS+1
162;       A(I-IS,J-JS)=ABCD(2)
163;       IF(I-IS.GT.M)M=I-IS
164;       IF(J-JS.GT.N)N=J-JS
165;       GO TO 1000
166;    23 DO 33 K=1,NBCT
167;    33 IF(I.GT.NT(K))IS=IS+1
168;       DO 34 K=1,NBCW
169;    34 IF(J.GT.NW(K))JS=JS+1
170;       A(I-IS,J-JS)=ABCD(3)
171;       IF(I-IS.GT.M)M=I-IS
172;       IF(J-JS.GT.N)N=J-JS
173;       GO TO 1000
174;    24 DO 35 K=1,NBCT
```

```
        ORGAN                        3/ 2/79  17:12:53     PAGE  4
175;        IF(I.GT.NT(K))IS=IS+1
176;     35 IF(J.GT.NT(K))JS=JS+1
177;        A(I-IS,J-JS)=ABCD(4)
178;        IF(I-IS.GT.M)M=I-IS
179;        IF(J-JS.GT.N)N=J-JS
180; 1000 CONTINUE
181; 1001 REWIND 2
182;        RETURN
183;        END
```

```
C       PROGRAM EFGH
C       --PREPARE DATA FOR FUNCTIONAL COEFFICIENTS E(Z),F(Z),G(Z),H(Z),
C       --I(Z) IN PROGRAM "RAYLEI"
        DIMENSION FILEA(2)
        COMMON T1,P1,E(60),F(60),G(60),H(60),RI(60)
        COMMON RO,AP,BA,CP,UE,PER
        COMMON DUT,DBAT,DBAP,DROT,DROP,DAPT,DAPP
        DATA PEROS/1.0E-11/,GV/980./,HZ/500000./,TS/25./
C       --PEROS=1.0E-11 IS AN ARBITRARY ASSUMPTION THIS PARAMETER WILL
C       --BE CANCELLED ITSELF FINALLY
        DATA FILEA(1),FILEA(2)/5HEFGHF,4H SRC/
        CALL ENTER(1,FILEA)
        WRITE(4,109)
  109 FORMAT(1X,"BOTTOM DEPTH(KM)=? (F5.2)")
        READ(4,108)DEPTH
  108 FORMAT(F5.2)
        KLOP=11
        KDZ=51/(KLOP-1)
        WRITE(4,58)
   58 FORMAT(1X,"PERMEABILITY CASE=? (1,2,3,OR 4)")
        READ(4,59)IPER
   59 FORMAT(I1)
        WRITE(4,28)
   28 FORMAT(1X,"GRADIENT=?(OC/KM)")
        READ(4,29)D
        D=D/100000.
   29 FORMAT(F5.0)
        KPW=KDZ-1
        DZ=DEPTH*100000./50.
        PSU=1.0
        TSU=25.
        PRE=0.
        RO=0.
        P1=PSU
        DO 10 I=1,51
        K=0
        DEP=FLOAT(I-1)*DZ
        T1=TSU+D*DEP
        PO1=P1
   15 P1=PO1+PRE
        ROO=RO
        CALL PROPT(1,1)
        IF(I.NE.1)GO TO 14
        ROO=RO
        ROS=RO
        APS=AP
        BAS=BA
        CPS=CP
        UES=UE
        ADIAS=AP*980.*(T1+273.15)/CP
        PERS=PEROS
   14 CONTINUE
C------FOR PERMEABILITY PER------
        GO TO (51,52,53,54),IPER
   51 PER=PEROS
        DPERZ=0.
        GO TO 55
   52 PER=PEROS*(1.-0.18*DEP*1.0E-5)
```

```
      DPERZ=-0.18*1.0E-5*PEROS
      GO TO 55
   53 PER=PEROS*(1.-0.36*DEP*1.0E-5+0.036*DEP*DEP*1.0E-10)
      DPERZ=PEROS*(-0.36*1.0E-5+0.072*DEP*1.0E-10)
      GO TO 55
   54 PER=PEROS*(1.-0.036*DEP*DEP*1.0E-10)
      DPERZ=PEROS*(-0.072*DEP*1.0E-10)
   55 CONTINUE
C-----FOR E(ZN)-----
      E(I)=HZ*(D*DUT/UE-DPERZ/PER-AP*D)
C-----FOR F(ZN)-----
      F(I)=(BA*RO*GV-AP*D)*(D*DUT/UE-DPERZ/PER-BA*RO*GV)
      F(I)=(F(I)+RO*GV*(D*DBAT+RO*GV*DBAP))
      F(I)=F(I)+BA*GV*(D*DROT+RO*GV*DROP)
      F(I)=HZ*HZ*(F(I)-D*(D*DAPT+RO*GV*DAPP))
C-----FOR G(ZN)-----
      G(I)=AP*RO*UES*PER/(APS*ROS*UE*PEROS)
C-----FOR H(ZN)-----
      H(I)=RO*(CP*D-AP*D*DEP*GV-AP*TS*GV)/(D*CPS*ROS)
C-----FOR RI(ZN)-----
      RI(I)=HZ*(D*DUT/UE-BA*RO*GV-DPERZ/PER)
      K=K+1
      IF(I.EQ.1)GO TO 20
      PRE=980.*((RO+ROO)*0.5)*1000./100000.
      RTEST=ABS(ROO-RO)/RO
      IF(RTEST.LE.0.0001)GO TO 20
      GO TO 15
   20 IF(I.NE.1)GO TO 12
      ES=F(1)
      FS=F(1)
      GS=G(1)
      HS=H(1)
      RIS=RI(1)
      PERS=PER
   12 ROR=RO/ROS
      APR=AP/APS
      BAR=BA/BAS
      CPR=CP/CPS
      UER=UE/UES
      ADIA=AP*980.*(T1+273.15)/CP
      ADIAR=ADIA/ADIAS
      ER=E(I)/ES
      FR=F(I)/FS
      GR=G(I)/GS
      HR=H(I)/HS
      RIR=RI(I)/RIS
      DEP=DEP/100000.
      PRE=980.*((RO+ROO)*0.5)*1000./100000.
      KPW=KPW+1
      IF(KPW.NE.KDZ)GO TO 10
      WRITE(1,88)E(I),F(I),G(I),H(I),RI(I)
   88 FORMAT(2X,5(E12.5))
      KPW=0
   10 CONTINUE
    1 CONTINUE
      CALL CLOSE(1)
      END
```

```
      SUBROUTINE PROPT(NPROPT,NEFGHI)
C     NPROPT=1, FOR LIQUID PROPERTIES; NEFGHI=1, FOR E,F,G,H,I(20)
      DIMENSION A(22),C(12)
C     TO PROVIDE THE HYDRODYNAMIC PROPERTIES  EFFECTED BY TEMP. & PRESS.
C     FROM MEYER ET AL(1968)
C     RO=DENSITY,  AP=EXPANSION COEF.,  BA=COMPRESSIBILITY
C     CP=SPECIFIC HEAT AT CONST. PRESS.,  UE=DYNAMIC VISCOSITY
C     INPUT "P1(BAR),T1(DEGREE C)",OUTPUT "RO,AP,BA,CP IN C.G.S. "
      COMMON T1,P1,E(60),F(60),G(60),H(60),RI(60)
      COMMON RO,AP,BA,CP,UE,PER          (11)
      COMMON DUT,DBAT,DBAP,DROT,DROP,DAPT,DAPP
      DATA A/-5.422063673E2,-2.096666205E4,3.941286787E4,-6.733277739E4,
     :9.902381028E4,-1.093911774E5,8.590041667E4,-4.511168742E4,
     :1.418138926E4,-2.017271113E3,7.982692717,-2.616571843E-2,
     :1.522411790E-3,2.284279054E-2,2.421647003E2,1.269716088E-10,
     :2.074838320E-7,2.17402035E-8,1.105710496E-9,1.293441934E1,
     :1.308119072E-5,6.047626338E-14/,C/8.438375405E-1,5.362162162E-4,
     :1.72,7.342278489E-2,4.97585887E-2,6.5371543E-1,1.15E-6,
     :1.5108E-5,1.4188E-1,7.002753165,2.995284926E-4,2.04E-1/,
     :A0/6.824687741E3/
      S=(T1+273.15)/647.3
      B=P1/221.2
      YS=1.-C(1)*S*S-C(2)*S**(-6)
      Z=YS+SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
      DYS=-2.*C(1)*S+6.*C(2)*S**(-7)
      DYSS=-2.*C(1)-42.*C(2)*S**(-8)
      DZS=DYS+(C(3)*YS*DYS-C(4))/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
      SQ=SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
      DZSS=DYSS+(C(3)*DYS*DYS+C(3)*YS*DYSS)/SQ
      DZSS=DZSS-(C(3)*YS*DYS-C(4))**2/(SQ*SQ*SQ)
      DZB=C(5)/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
      DXS=-(5./17.)*A(11)*C(5)*(Z**(-22./17.))*DZS
      DXS=DXS+(A(13)+2.*A(14)*S-A(15)*10.*(C(6)-S)**9
     :-A(16)*19*S**18/(C(7)+S**19)**2)
      DXS=DXS+11.*(A(17)+2.*A(18)*B+3.*A(19)*B*B)*S**10/(C(8)+S**11)**2
      DXS=DXS-18.*A(20)*(S**17)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
      DXS=DXS-2.*A(20)*S**19*(-3./(C(10)+B)**4+C(11))-3*A(21)*B*B
     :-80.*A(22)*S**(-21)*A*B*B
      DXB=(-5./17.)*A(11)*C(5)*DZB*(Z**(-22./17.))
      DXB=DXB-(2.*A(18)+6.*A(19)*B)/(C(8)+S**11)
      DXB=DXB-A(20)*S**18*(C(9)+S*S)*12/(C(10)+B)**5
      DXB=DXB+6.*A(21)*(C(12)-S)*B+12.*A(22)*B*B/S**20
C     FOR DENSITY
      X1=A(11)*C(5)*(Z**(-5./17.))
      X1=X1+(A(12)+A(13)*S+A(14)*S*S+A(15)*(C(6)-S)**10+A(16)/
     :(C(7)+S**19))
      X1=X1-(A(17)+2.*A(18)*B+3.*A(19)*B*B)/(C(8)+S**11)
      X1=X1-A(20)*(S**18)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
      X1=X1+3.*A(21)*(C(12)-S)*B*B+4.*A(22)*(S**(-20))*B*B*B
      RO=1./(3.17*X1)
C------FOR EXPANSION COEF "AP" AND COMPRESIBILITY "BA"------
      AP=RO*3.17/647.3*DXS
      BA=-RO*3.17/221.2*DXB*1.0E-6
C------FOR SPECIFIC HEAT "CP" ------
      SI=420.*A(22)*(B**4)/S**22-A(20)*(S**16)*(306.*C(9)+380.*S*S)*
     :(1./(C(10)+B)**3+C(11)*B)
      SI=SI-(242*(S**20)/(C(8)+S**11)**3-110*(S**9)/(C(8)+S**11)**2)*
     :(A(17)*B+A(18)*B*B+A(19)*B*B*B)
      SI=SI+B*(2*A(14)+90*A(15)*(C(6)-S)**8+722*A(16)*(S**36)
     :/(C(7)+S**19)**3-342*A(16)*S**17/(C(7)+S**19)**2)
      SI=SI+A(11)*(17*DZSS/29-17*DYSS/12)*(Z**(12./17.))
```

```
      SI=SI+A(11)*(24*DZS/29-2*DYS)*DZS*(Z**(-5./17.))
      SI=SI+A(11)*(12*Z/29-YS)*(Z**(-5./17.)*DZSS-5*(Z**(-22./17.))*DZS
     *DZS/17.)
      SI=SI+2*A(3)+6*A(4)*S+12*A(5)*S**2+20*A(6)*S**3+30*A(7)*S**4
      SI=SI+42*A(8)*S**5+56*A(9)*S**6+72*A(10)*S**7-A0/S
      CP=-221.2*3.17*SI*S/647.3*1.0E6
C-----FOR VISCOSITY "UE"-----
C     UE=0.000001*(241.4*(10.**(247.8/(T1+133.15))))
      UEA=(T1-150.)/100.
      UED=538.+380.*UEA-26.*UEA**3
      UE=1./UED
      IF(NEFGHI.NE.1)GO TO 100
      DZBR=-C(5)*DZB/(Z-YS)**2
      DZSR=-C(5)*(DZS-DYS)/(Z-YS)**2
      DXSS=1680*A(22)*B**3/S**22
      DXSS=-(18*17*A(20)*S**16*(C(9)+S*S)+36*A(20)*S**18+38*A(20)*S**18)
     :*(C(11)-3/(C(10)+B)**4)+DXSS
      DXSS=DXSS+(-22*11*S**20/(C(8)+S**11)**3+110*S**9/(C(8)+S**11)**2)
     :*(A(17)+2*A(18)*B+3*A(19)*B*B)
      DXSS=2*A(14)+90*A(15)*(C(6)-S)**8+38*A(16)*S**18/(C(7)+S**19)**3
     :-(18*19)*A(16)*S**17/(C(7)+S**19)**2+DXSS
      DXSS=(110./289.)*A(11)*C(5)/Z**(39./17.)*DZS*DZS+DXSS
      DXSS=DXSS-(5./17.)*A(11)*C(5)*DZSS/Z**(22./17.)
      DXSB=-6*A(21)*B-240*A(22)*B*B/S**21
      DXSB=DXSB-(18*A(20)*S**17*(C(9)+S*S)+2*A(20)*S**19)
     :*(12/(C(10)+B)**5)
      DXSB=DXSB+11*(2*A(18)+6*A(19)*B)*S**10/(C(8)+S**11)**2
      DXSB=DXSB-(5./17.)*A(11)*C(5)*DZSB/Z**(22./17.)
      DXSB=DXSB+(110./289.)*A(11)*C(5)*DZB*DZS/Z**(39./17.)
      DXBB=24*A(22)*B/S**20+6*A(21)*(C(12)-S)
      DXBB=DXBB+60*A(20)*(C(9)+S*S)*S**18/(C(10)+B)**6
      DXBB=DXBB-6*A(19)/(C(8)+S**11)-(5./17.)*A(11)*C(5)*DZBB/
     :Z**(22./17.)
      DXBB=DXBB+(110./289.)*A(11)*C(5)*DZB*DZB/Z**(39./17.)
      DUT=-(3.8-0.79*UEA*UEA)/(UED*UED)
      DROT=-DXS/(647.3*3.17*X1*X1)
      DROP=-DXB/(3.17*2.212E8*X1*X1)
      DAPT=3.17*(-DXS*DXS/(3.17*X1*X1)+RO*DXSS)/(647.3*647.3)
      DAPP=3.17*(-DXB*DXS/(3.17*X1*X1)+RO*DXSB)/(647.3*2.212E8)
      DBAT=-3.17*(-DXS*DXB/(3.17*X1*X1)+RO*DXSB)/(647.3*2.212E8)
      DBAP=-3.17*(-DXB*DXB/(3.17*X1*X1)+RO*DXBB)/(4.892944E16)
100   CONTINUE
      RETURN
      END
```

APPENDIX D

Flow Charts and Program Listing for the
Time Dependent Model

The following programs are designed as an example to be structurally similar to that in Figure 30, Chapter VI. Considering the accuracy of the numerical integral algorithm (otherwise need to be integrated by using the characteristic relationship of area coordinates), we assumed both the hydrodynamic parameters and the permeability are constant within an element. The final element subdivision for the domain is shown in the figure on the following page.

Programs wrote in FORTRAN IV below are originally run on a ECLIPSE mini computer which has 32K 16-bit true work memory (exclude system memory). Due to storage limitation an "overlay technique" has been applied which can be found elsewhere in the manual along with most of the mini computers. The whole programs of the system are located in several overlay areas, only limited files are loaded in the core during run time, although the whole program listings become lengthy. Some programs need to be renamed in different overlays because of the recognition of the programs during load.
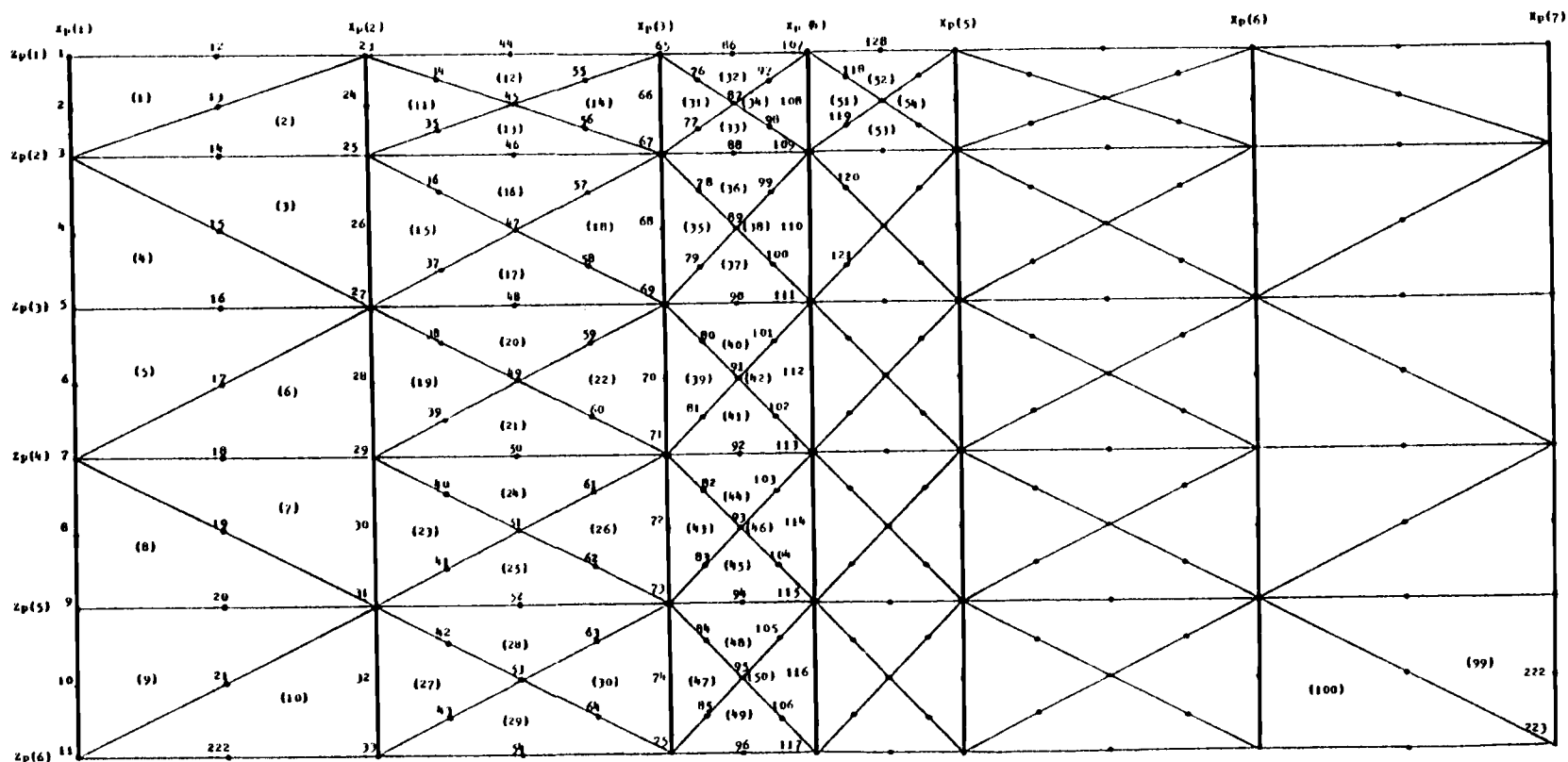
Figure 43. Element and node numbering for the domain (scale is not proportional to the real geometry as in Figure 30).

THE OVERALL PROGRAM LOGIC

```
                    ┌─────────────┐
                    │    START    │
                    └─────────────┘
                           │   (root program)
                           ▼
            ┌───────────────────────────────┐
            │ CONVEC                         │
            │ Link overlays BASIC, START,    │
            │ REVISE, MAIN, MAIN2, SOLVE     │
            └───────────────────────────────┘
                           │   (overlay BASIC)
                           ▼
  ┌──────────────────────────────────────────────────┐
  │ Call GRID                                          │
  │ Generate element data, locate node numbers, par-  │
  │ tial element data are read in from file GRIDIN     │
  └──────────────────────────────────────────────────┘
                           │   (overlay BASIC)
                           ▼
            ┌───────────────────────────────┐
            │ Call HAMER                     │
            │ Input stations and weights for │
            │ numerical integrations         │
            └───────────────────────────────┘
                           │   (overlay START)
                           ▼
  ┌──────────────────────────────────────────────────┐
  │ Call INIT                                          │
  │ Input parameters of porous media and rocks,        │
  │ boundary conditions, initial conditions, function  │
  │ DENPRO has been called to provide hydrodynamic     │
  │ parameters                                         │
  └──────────────────────────────────────────────────┘
      ┌───┐                │
      │ 1 │────────────────▶│
      └───┘                 │   (overlay REVISE)
                            ▼
            ┌───────────────────────────────┐
            │ Call PROPT, set IDP=0          │
            │ Revise hydrodynamic parameters │
            │ for update temperature and     │
            │ pressure                       │
            └───────────────────────────────┘
                            │
                            ▼
```

```
                        ┌──────────────────┐
                        │     IDP>1 ?      │────── Yes ──────┐
                        └──────────────────┘                 │
                                │                            │
                               No                            │
                                │         (overlay MAIN2)    │
        ┌───────────────────────────────────────────┐        │
        │ Call STIFP                                 │        │
        │ BA2P, DXZ2P, COMPP, COMPB have             │        │
        │ been used to form finite difference (in    │        │
        │ time) equations                            │        │
        │ [A]P^{n+1}=[B]P^{n}+F                       │        │
        │ Save [A], [B], F on disk file AXAY         │        │
        └───────────────────────────────────────────┘        │
                                │◄───────────────────────────┘
                                │         (overlay SOLVE)
        ┌───────────────────────────────────────────┐
        │ Call EQSO, IDP=IDP+1                        │
        │ Read in from file AXAY and from            │
        │ [A]P^{n+1}=[B]P^{n}+F=F'                    │
        │ Solve for P^{n+1}, update P                │
        └───────────────────────────────────────────┘
```

Call STIFP

BA2P, DXZ2P, COMPP, COMPB have been used to form finite difference (in time) equations

$$[A]\vec{P}^{n+1}=[B]\vec{P}^{n}+\vec{F}$$

Save [A], [B], $\vec{F}$ on disk file AXAY

(overlay SOLVE)

Call EQSO, IDP=IDP+1

Read in from file AXAY and from

$$[A]\vec{P}^{n+1}=[B]\vec{P}^{n}+\vec{F}=\vec{F}'$$

Solve for $\vec{P}^{n+1}$, update $\vec{P}$

No ◄─── IDP≥MCY ?

Yes

(overlay MAIN)

Call STIFF, ID=1

BA2, DXZ2, COMPAC have been used to form $[A]\vec{u}=\vec{F}$

(overlay SOLVE)

Call EQSO

Solve for $\vec{u}$

(overlay MAIN)

Call STIFF, ID=2
BA2, DXZ2, COMPAC have been used
to form $[A]\vec{w}=\vec{F}$

(overlay SOLVE)

Call EQSO
Solve for $\vec{w}$
Set IDP=0

(overlay MAIN)

Call STIFF, ID=4
BA2, DXZ2, COMPAC have been used
to form $[A]\vec{T}^{n+1}=\vec{F}$

(overlay SOLVE)

Call EQSO
Solve for $\vec{T}$ at $t+\Delta t$
Update $\vec{T}$

Print results on disk file SHCON if
necessary

KTIME=KEGY? —No→ ①

Yes

END

MCY = number of flow cycles per energy cycle.
n, n+1 = time levels.
F = forcing functions or known vectors.
KEGY = total time levels required.

Flow chart for subroutine STIFF
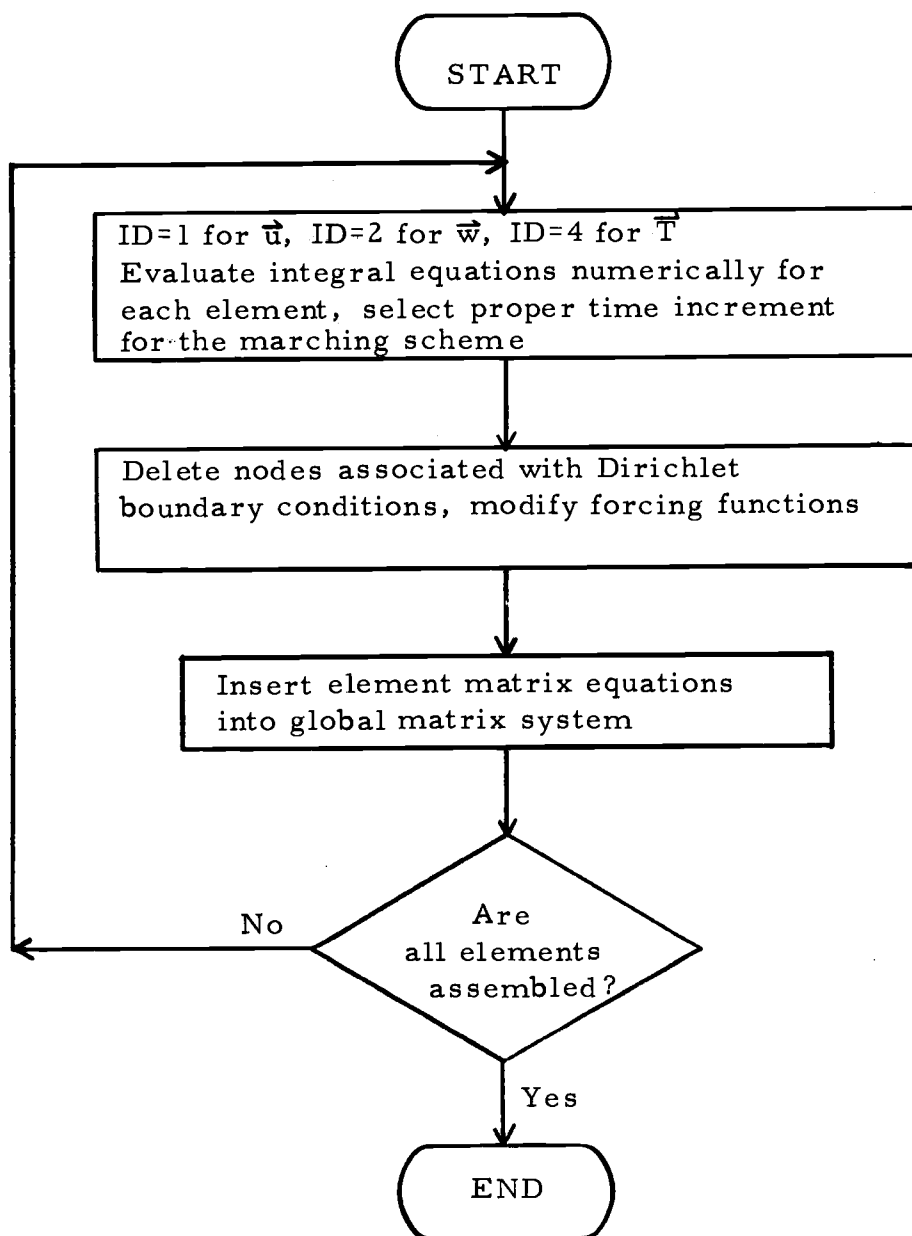
```
                        ╭──────────────╮
                        │    START     │
                        ╰──────────────╯
                               │
                               ▼
   ┌───────────────────────────────────────────────────────┐
   │ ID=1 for u⃗, ID=2 for w⃗, ID=4 for T⃗                    │
   │ Evaluate integral equations numerically for           │
   │ each element, select proper time increment            │
   │ for the marching scheme                               │
   └───────────────────────────────────────────────────────┘
                               │
                               ▼
   ┌───────────────────────────────────────────────────────┐
   │ Delete nodes associated with Dirichlet                │
   │ boundary conditions, modify forcing functions         │
   └───────────────────────────────────────────────────────┘
                               │
                               ▼
   ┌───────────────────────────────────────────────────────┐
   │ Insert element matrix equations                       │
   │ into global matrix system                             │
   └───────────────────────────────────────────────────────┘
                               │
                               ▼
                         ╱───────────╲
          No            ╱    Are      ╲
      ◄────────────────╱ all elements  ╲
                       ╲  assembled?   ╱
                        ╲─────────────╱
                               │
                              Yes
                               │
                               ▼
                        ╭──────────────╮
                        │     END      │
                        ╰──────────────╯
```

Definitions of Important Symbols and Explanations of
Program Listings

The following program names appear in the order in which they are called. Parameters will not be defined if already defined in the previous programs unless they need to be redefined. Numbers in parentheses represent the corresponding commentary locations in the original program listings.

Main program CONVEC
Function: link programs in overlay BASIC, START, REVISE, MAIN, MAIN2 and SOLVE, evaluate temperature, flow velocity and pressure solutions for time dependent model of natural convection in a porous slab with various conditions.

SHGOB	disk file for the element data storage.
(1)	see subroutine GRID.
(2)	see subroutine HAMER.
(3)	see subroutine INIT.
(4)	see subroutine PROPT.
SHCON	disk file for output of solutions.
SHGAB	disk file for scrap use.
AXAY	disk file for temporary storage of flow equations in matrix form.
DTP, DT	time increments in the flow and in the energy equations respectively.
(5)	see subroutine STIFP.
(6)	see subroutine EQSO.

NP(i)	node indexes for pressure.
P1(i), P2(i)	pressure at time t and t+Δt respectively.
YY(i)	forcing functions in input, solutions in output.
(7)	see subroutine STIFF.
NU(i)	node indexes for velocity u.
U(i)	horizontal velocity u.
NW(i)	node indexes for velocity w.
W(i)	vertical velocity w.
NT(i)	node indexes for temperature T.
T1(i), T2(i)	temperature at time t and t+Δt respectively.

Subroutine GRID
Function: generate the element data for the structure.

GRIDIN	disk file, provide partly node and element numbers, specify partly node coordinates.
IMD, IGD	grid intervals in depth and the horizontal respectively.
XP(i), ZP(i)	node coordinates in the horizontal and depth respectively.
ZTA	dip angle of the slab.
NET	total elements.
NOD	total nodes.
Y(i, j)	global node numbers, i=element number, j=vertex number in a triangular element.
NF(i)	node indexes in the domain of the porous slab.

XC(i,j), ZC(i,j)   the X and Z coordinates of the nodes, i=element number, j=vertex number in a triangular element.

Subroutine HAMER
Function: provide stations and weights for numerical integrations.

ID   =1 for 1-D element.

                    =2 for triangular element.

ZN(i)   stations for 1-D element.

ZS(i,j)   stations for triangular element.

WE(i)   weights.

Subroutine INIT
Function: introduce material parameters, boundary conditions, initial conditions.

FAI   porosity.

BFAI   vertical compressibility (cm^2/dyne).

ROS   density of solid phase in the porous medium (g/cm^3).

CS   specific heat of solid phase in the porous medium (erg/g, °C).

COND   conductivity of the porous medium (erg/cm, s, °C).

RORCK   rock density (g/cm^3).

CSRCK   rock specific heat (erg/g, °C).

CONRCK   conductivity of the rock (erg/cm, s, °C).

PERS   permeability at the surface (cm^2).

TS   temperature at the surface (°C).

DTDZ   temperature gradient (°C/km).

GRAV $=980$ (cm/s^2).

(1)   NT(i)=555 if temperature at node i is a known value (Dirichlet conditions), so as NU(i) to velocity u, NW(i) to velocity w, NP(i) to pressure P.

PER   permeability (cm^2).

PER1, PER2   permeability coefficients associated with the first order and the second order terms respectively in Table 6.

NG(i)   global node indexes in an element.

XO(i), ZO(i)   node coordinates in an element

RAYLE   Rayleigh number.

PERS1   =PERS.

Function DENPRO

Function: water properties of ρ_f, α, β, C_f, and μ are derived from the empirical equations given by Meyer et al. (1968) and Mercer et al. (1975) if temperature and pressure are known.

IDT   =1 for ρ_f (g/cm^3).

                      =2 for α (1/°C).

                      =3 for β (cm^2/dyne).

                      =4 for C_f (erg/g, °C).

                      = 5 for μ (poise).

TEMP   temperature (°C).

PRESS   pressure (dyne/cm^2).

AO, A(i), C(i)   coefficients for the empirical equations.

Subroutine PROPT
Function: as stated in function DENPRO.

RO(i), AP(i), BA(i),
CP(i), UE(i), PER(i) ρ_f, α, β, C_f, μ, and permeability at node i
                         respectively.

Subroutine STIFP
Function: form matrix equations in a partial packing storage scheme
            (this scheme was suggested by Key,[3] 1973) to solve for
            pressure.

ISCH   =1 for backward difference scheme.

                         =2 for Crank-Nicholson scheme.

(1)   will form a system as $[AX]\vec{P}^{n+1} = [AY]\vec{P}^{n} + \vec{F}$.

MEQ   total equations to be solved.

MROWX   maximum row dimension of AX(i, j).

MROWY   maximum row dimension of AY(i, j).

AR   area of an element.

DJACO   Jacobian determinant for coordinate transforma-
                         tion.

(2)   average values of material properties in an
                         element.

P, Q, R   temporary storage of ZS(i, j).

(3)   see function BA2P and function DXZ2P.

AXE(i, j)   local (for one element) matrix of AX(i, j).

AYE(i, j)   local (for one element) matrix of AY(i, j).

------

[3] J.E. Key, Computer program for solution of large, sparse, unsymmetric systems of linear equations, Int. J. for numerical methods in engineering, 6: 497-509, 1973.

(4)   see subroutine COMPP and subroutine COMPB.

(5)   write down information of $YY(i)$, $AX(i,j)$, $NCOAX(i,j)$, $AY(i,j)$ and $NCOAY(i,j)$ on disk file AXAY.

Function BA2P

Function: represent quadratic shape functions $Ni(i=1,2,\ldots 6)$ by area coordinates Li $(i=1,2,3)$ at stations P, Q, R.

IBA   index of shape function Ni (e.g. IBA=5 for N5).

P, Q, R   stations for numerical integrations.

Function DXZ2P

Function: form derivatives of quadratic shape functions Ni $(i=1,2,\ldots 6)$ in terms of area coordinates Li $(i=1,2,3)$ at stations P, Q, R.

ND $=1$ for $\partial N_i /\partial x$

                    $=-1$ for $\partial N_i /\partial z$

M   index of shape function Ni.

Subroutine COMPP

Function: insert a local matrix $AE(i,j)$ into a global matrix $A(i,j)$ in a partial packing storage scheme.

$AE(i,j)$ $=AXE(i,j)$ defined in STIFP.

$A(i,j)$ $=AX(i,j)$ defined in STIFP.

$NCOL(i,j)$ $=NCOAX(i,j)$, column indices of $A(i,j)$.

Subroutine COMPB

Function: insert a local matrix $AE2(i,j)$ into a global matrix $A2(i,j)$ in a partial packing storage scheme.

$AE2(i,j)$ $=AYE(i,j)$ defined in STIFP.

A2(i, j)   =AY(i, j) defined in STIFP.

NCOL2(i, j)   =NCOAY(i, j), column indices of A2(i, j).

Subroutine EQSO
Function: solve matrix system equations $[A]\vec{X}=\vec{B}$, $[A]$=partial
             packed matrix, $\vec{B}$=known vector. Details of the program
             was discussed by Gupta and Tanji (1977). [4]

NPT   total equations to be solved.

MROW   maximum row dimension of A(i, j) in calling pro-
                          gram.

NNN   maximum row dimension of A(i, j) in EQSO,
                          NNN$\geq$MROW.

ZTEST   value below which element made equal to zero.

B(i)   forcing functions in input, solutions in output.

IBANDW   number of nonzero coefficients in each row.

NCOL(i, j)   matrix containing indices of nonzero coefficients
                          of A(i, j).

NNCOL(i)   one dimension array for pivotal row indices.

NPIV(i)   one dimension array to store pivotal column.

AA(i)   one dimension array used for pivotal row elements.

Subroutine STIFF
Function: form matrix equations in a partial packing storage
             scheme.

ID   =1 for velocity u.

                          = 2 for velocity w.

---

[4]S. K. Gupta and K. K. Tanji, Computer program for solution of large, sparse, unsymmetric systems of linear equations, Int. J. Numerical Methods in Engineering, 11:1251-1259, 1977.

=4 for temperature T.

(1) see function BA2 and function DXZ2.

UKNO velocity u at previous time.

WKNO velocity w at previous time.

PKNO pressure P at previous time.

(2) see subroutine COMPAC.

Function BA2
Function: as stated in function BA2P.

Function DXZ2
Function: as stated in function DXZ2P.

Subroutine COMPAC
Function: as stated in subroutine COMPP.

```
     CONVEC                    3/ 2/79   9: 4: 3     PAGE  1
 1;C      POGRAM CONVEC              ---C. T. SHYU---MAY, 1978---
 2;C      A TRANSCIENT MODEL FOR CONVECTION IN POROUS MATERIAL CONFINED IN A SLAB
 3;C      ****** WROTE BY C. T. SHYU, MAY, 1978 ******
 4;       COMMON SGW1(22),SGW2(12),SGW3
 5;       COMMON FAI,8FAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 6;       COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,HET,NOD,ZTO,XTO,IMD,IGD,ZTA
 7;       COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
 8;       COMMON NT(223),NU(77),NW(77),NP(77)
 9;       COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
10;       COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
11;       COMMON AXE(6,6),AX(179,25),YY(223),NCOAX(179,25)
12;       EXTERNAL BASIC,START,MAIN,MAIN2,SOLVE,REVISE
13;C      88MATERIAL PROPERTIES ARE INPUT THROUGH SUBROUTINE "INIT"
14;C      90WATER PROPERTIES ARE PROVIDED THROUGH SUBROUTINE "PROPT"
15;       CALL FOPEN(35,"SHG08")
16;       CALL OVOPN(21,"CONVEC.0L",IER)
17;C-----GENERATE FINITE GRID POINTS FROM GIVEN GEOMETRY---
18;       CALL OVLOD(21,BASIC,0,IER8)
19;       CALL GRID  <------------------------------------------( 1 )
20;C-----INTRODUCE HAMER'S FORMULA FOR NUMERICAL INTEGRATION---( 2 )
21;       CALL HAMER(2)  <--------------------------------------
22;C-----INITIALIZE SYSTEMS AND PROVIDE INITIAL CONDITIONS---
23;       CALL OVLOD(21,START,0,IERST)
24;       CALL INIT  <------------------------------------------( 3 )
25;       CALL OVLOD(21,REVISE,0,IERS)
26;       CALL PROPT  <-----------------------------------------( 4 )
27;C------------------------------------------------------------
28;       CALL FOPEN(30,"SHCON")
29;       CALL FOPEN(31,"SHGA8")
30;       CALL FOPEN(33,"AXAY")
31;C-----CALCULATE NEW P AT T+DT-----
32;       ACCEPT"HOW MANY FLOW CYCLES PER ENEREGY CYCLE?",NCY
33;       ACCEPT"TOTAL ENERGY STEPS=?",KEGY
34;       KTIME=0
35;  300  KTIME=KTIME+1
36;       IDP=0
37;       DTP=DT/NCY
38;  100  IDP=IDP+1
39;       IF(IDP.GT.1)GO TO 6
40;       CALL OVLOD(21,MAIN2,0,IERM2)
41;       CALL STIFP(1,NEQ,NROWX,NROWY)  <----------------------( 5 )
42;    6  CALL OVLOD(21,SOLVE,0,IERS)
43;       CALL EQSO(NEQ,NROWX,25,0,0)  <------------------------( 6 )
44;       IND=0
45;       DO 30 I=1,77
46;       IF(NP(I).EQ.555)GO TO 30
47;       IND=IND+1
48;       P1(I)=P2(I)
49;       P2(I)=YY(IND)
50;   30  CONTINUE
51;       IF(IDP.GE.NCY)GO TO 101
52;       NHC=(NCY+1)/2
53;       IF(KTIME.EQ.1.AND.IDP.EQ.NHC)GO TO 101
54;       GO TO 100
55;C-----CALCULATE U---
56;  101  CALL OVLOD(21,MAIN,0,IERM)
57;       CALL STIFF(0,1,NEQ,NROW)  <---------------------------( 7 )
58;       CALL OVLOD(21,SOLVE,0,IERS)
```

```
      CONVEC                    3/ 2/79    9: 4: 3     PAGE  2
 59;        CALL EQSO(MEQ,MROW,25,0,0)
 60;        IND=0
 61;        DO 10 I=1,77
 62;        IF(NU(I).EQ.555)GO TO 10
 63;        IND=IND+1
 64;        U(I)=YY(IND)
 65;   10 CONTINUE
 66;C-----CALCULATE W---
 67;        CALL OVLOD(21,MAIN,0,IERM)
 68;        CALL STIFF(0,2,MEQ,MROW)
 69;        CALL OVLOD(21,SOLVE,0,IERS)
 70;        CALL EQSO(MEQ,MROW,25,0,0)
 71;        IND=0
 72;        DO 20 I=1,77
 73;        IF(NW(I).EQ.555)GO TO 20
 74;        IND=IND+1
 75;        W(I)=YY(IND)
 76;   20 CONTINUE
 77;C-----TEMPERATURE AT T+DT---
 78;        CALL OVLOD(21,MAIN,0,IERM)
 79;        CALL STIFF(1,4,MEQ,MROW)
 80;        CALL OVLOD(21,SOLVE,0,IERS)
 81;        CALL EQSO(MEQ,MROW,25,0,0)
 82;        IND=0
 83;        DO 40 I=1,223
 84;        IF(NT(I).EQ.555)GO TO 40
 85;        IND=IND+1
 86;        T1(I)=T2(I)
 87;        T2(I)=YY(IND)
 88;   40 CONTINUE
 89;        CALL OVLOD(21,REVISE,0,IERR)
 90;        CALL PROPT
 91;        DT=1.2*DT
 92;        KTIME=KTIME+1
 93;        WRITE(30,18)KTIME
 94;   18 FORMAT(1X,"KTIME=",I3)
 95;        DO 50 I=1,7
 96;        J2=11*I
 97;        J1=J2-10
 98;   50 WRITE(30,58)(U(J),J=J1,J2)
 99;   58 FORMAT(1X,11(E11.4,1X))
100;        DO 51 I=1,7
101;        J2=11*I
102;        J1=J2-10
103;   51 WRITE(30,58)(W(J),J=J1,J2)
104;        DO 52 I=1,20
105;        J2=11*I
106;        J1=J2-10
107;   52 WRITE(30,58)(T2(J),J=J1,J2)
108;        WRITE(30,59)(T2(J),J=221,223)
109;   59 FORMAT(1X,3(E11.4,1X))
110;        IF(KTIME.GE.KEGY)GO TO 1000
111;        GO TO 300
112; 1000 CALL RESET
113;        END
```

```
      GRID                        3/ 2/79   9:57:16     PAGE  1
 1;        OVERLAY BASIC
 2;        SUBROUTINE GRID
 3;C    ---TO GENERATE ELEMENT DATA---GRID SCALE=M*N=IMD*IGD---
 4;C    ---NET=TOTAL ELEMENTS: NOD=TOTAL NODS: IB=BAND WIDTH: ZTA=DIP ANGLE---
 5;C    ---ZTO=BOTTOM DEPTN: XTO=HORIZONTAL WIDTH---
 6;C    ---OUTPUT=IMD,IGD,IB,ZTO,XTO,Y,NET,NOD,XC,ZC---
 7;        INTEGER Y
 8;        DIMENSION Y(100,3),XC(100,3),ZC(100,3)
 9;        COMMON SGW1(22),SGW2(12),SGW3
10;        COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
11;        COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
12;        COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
13;        COMMON NT(223),NU(77),NW(77),XP(7),ZP(6)
14;        CALL FOPEN(36,'GRIDIN')
15;        WRITE(10,18)
16;     18 FORMAT(1X,'GRID DIMENSION(M*N), M,N=? (I1,1X,I1)')
17;        READ(36,19)IMD,IGD
18;        TYPE IMD,IGD
19;     19 FORMAT(I1,1X,I1)
20;        WRITE(10,28)
21;     28 FORMAT(1X,'XP1,XP2,XP3,XP4,XP5,XP6,XP7=? 7(F7.3,1X)')
22;        READ(36,29)XP(1),XP(2),XP(3),XP(4),XP(5),XP(6),XP(7)
23;        WRITE(10,29)XP(1),XP(2),XP(3),XP(4),XP(5),XP(6),XP(7)
24;     29 FORMAT(7(F7.3,1X))
25;        WRITE(10,38)
26;     38 FORMAT(1X,'ZP1,ZP2,ZP3,ZP4,ZP5,ZP6=? 6(F7.3,1X)')
27;        READ(36,29)ZP(1),ZP(2),ZP(3),ZP(4),ZP(5),ZP(6)
28;        WRITE(10,29)ZP(1),ZP(2),ZP(3),ZP(4),ZP(5),ZP(6)
29;        WRITE(10,58)
30;     58 FORMAT(1X,'DIP ANGLE(DEGREE)=? (F5.0)')
31;        READ(36,59)ZTA
32;     59 FORMAT(F5.0)
33;        WRITE(10,59)ZTA
34;        ZT=ZTA/180.*3.1415926536
35;        ZTO=ZP(6)
36;        XTO=XP(7)
37;        NET=100
38;        NOD=223
39;        DO 9 I=1,100
40;        DO 9 J=1,3
41;      9 Y(I,J)=0
42;        DO 20 I=1,223
43;     20 NF(I)=0
44;        DO 31 KM=1,5
45;        DO 30 I=1,9
46;        NF(66+(KM-1)*21+I)=(KM-1)*17+I
47;        IF(KM.EQ.5.OR.I.EQ.9)GO TO 30
48;        NF(77+(KM-1)*21+I)=(KM-1)*17+I+9
49;     30 CONTINUE
50;     31 CONTINUE
51;        DO 33 I=1,10
52;     33 READ(36,78)(Y(I,J),J=1,3)
53;     78 FORMAT(I3,1X,I3,1X,I3)
54;        DO 34 I=91,100
55;     34 READ(36,78)(Y(I,J),J=1,3)
56;        CALL FCLOS(36)
57;        DO 12 K=1,11,2
58;        K1=K+22
```

```
      GRID                    3/ 1/79  15:38:51    PAGE  2
59;         KH=K/2
60;         DO 13 I=1,10
61;         DO 13 J=1,3
62;         IF(Y(I,J).EQ.K)XC(I,J)=XP(1)
63;         IF(Y(I,J).EQ.K)ZC(I,J)=ZP(1+KH)
64;         IF(Y(I,J).EQ.K1)XC(I,J)=XP(2)
65;      13 IF(Y(I,J).EQ.K1)ZC(I,J)=ZP(1+KH)
66;      12 CONTINUE
67;         DO 14 K=191,201,2
68;         K1=K+22
69;         KH=(K-190)/2
70;         DO 15 I=91,100
71;         DO 15 J=1,3
72;         IF(Y(I,J).EQ.K)XC(I,J)=XP(6)
73;         IF(Y(I,J).EQ.K)ZC(I,J)=ZP(1+KH)
74;         IF(Y(I,J).EQ.K1)XC(I,J)=XP(7)
75;      15 IF(Y(I,J).EQ.K1)ZC(I,J)=ZP(1+KH)
76;      14 CONTINUE
77;         DO 210 J=1,4
78;         NI=0
79;         DO 210 I=1,5
80;         NI=NI+1
81;         I1=11+4*(I-1)+(J-1)*20
82;         IM=42*(J-1)+23+2*(I-1)
83;         Y(I1,1)=IM
84;         Y(I1+1,1)=Y(I1,1)
85;         Y(I1,2)=IM+2
86;         Y(I1+2,2)=Y(I1,2)
87;         Y(I1,3)=IM+2+20
88;         Y(I1+1,2)=Y(I1,3)
89;         Y(I1+2,1)=Y(I1,3)
90;         Y(I1+3,2)=Y(I1,3)
91;         Y(I1+1,3)=IM+2+40
92;         Y(I1+3,1)=Y(I1+1,3)
93;         Y(I1+2,3)=IM+2+40+2
94;         Y(I1+3,3)=Y(I1+2,3)
95;         XC(I1,1)=XP(J+1)
96;         XC(I1+1,1)=XC(I1,1)
97;         XC(I1,2)=XC(I1,1)
98;         XC(I1+2,2)=XC(I1,1)
99;         XC(I1,3)=XP(J+1)+0.5*(XP(J+2)-XP(J+1))
100;        XC(I1+1,2)=XC(I1,3)
101;        XC(I1+2,1)=XC(I1,3)
102;        XC(I1+3,2)=XC(I1,3)
103;        XC(I1+1,3)=XP(J+2)
104;        XC(I1+3,1)=XC(I1+1,3)
105;        XC(I1+2,3)=XC(I1+1,3)
106;        XC(I1+3,3)=XC(I1+1,3)
107;        ZC(I1,1)=ZP(NI)
108;        ZC(I1+1,1)=ZC(I1,1)
109;        ZC(I1+1,3)=ZC(I1,1)
110;        ZC(I1+3,1)=ZC(I1,1)
111;        ZC(I1,3)=ZP(NI)+0.5*(ZP(NI+1)-ZP(NI))
112;        ZC(I1+1,2)=ZC(I1,3)
113;        ZC(I1+2,1)=ZC(I1,3)
114;        ZC(I1+3,2)=ZC(I1,3)
115;        ZC(I1,2)=ZP(NI+1)
116;        ZC(I1+2,2)=ZC(I1,2)
```

```
        GRID                      3/ 1/79  15:30:51     PAGE  3
117;        ZC(I1+2,3)=ZC(I1,2)
118;        ZC(I1+3,3)=ZC(I1,2)
119;    210 CONTINUE
120;        IF(ZTA.EQ.90.)GO TO 212
121;        DO 211 I=1,NET
122;        DO 211 J=1,3
123;        XC(I,J)=ZC(I,J)/TAN(ZT)+XC(I,J)
124;    211 CONTINUE
125;    212 DO 220 I=1,NET
126;        DO 220 J=1,3
127;        WRITE(35,68)Y(I,J),XC(I,J),ZC(I,J)
128;    220 CONTINUE
129;     68 FORMAT(2X,I3,2X,F8.4,2X,F8.4)
130;        REWIND 35
131;        RETURN
132;        END
```

File "GRIDIN"
5 6
0.00000 5.00000 12.0000 12.6250 13.2500 20.2500 25.2500
0.00000 0.30000 1.55000 2.80000 4.05000 5.30000 0.00000
90.00
001 003 023
023 003 025
003 027 025
003 005 027
005 007 027
027 007 029
007 031 029
007 009 031
009 011 031
031 011 033
191 213 213
191 193 215
193 195 215
215 195 217
195 219 217
195 197 219
197 199 219
219 199 221
199 223 221
199 201 223

```
      HAMER                    3/ 1/79  15:33:31     PAGE  1
 1;        SUBROUTINE HAMER(ID)
 2;C       HAMMER'S FORMULA FOR NUMERICAL INTEGRATION IN ONE-D & TRIANGULAR
 3;C       DOMAIN.  E.G.  ID=1 FOR 1-D; ID=2 FOR TRIANGULAR
 4;        COMMON SGW1(22),SGW2(12),SGW3
 5;        COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 6;        COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4)
 7;        IF(ID.EQ.2)GO TO 20
 8;        ZN(1)=-0.86113631-0.1594853E-8
 9;        ZN(2)=-0.33998104-0.3584856E-8
10;        ZN(3)=-ZN(2)
11;        ZN(4)=-ZN(1)
12;        WE(1)=0.34785484+0.5137454E-8
13;        WE(2)=0.65214515+0.4862546E-8
14;        WE(3)=WE(2)
15;        WE(4)=WE(1)
16;        GO TO 50
17;    20 S1=0.3333333333
18;        S2=0.79742699
19;        S3=0.10128651
20;        S4=0.05971587
21;        S5=0.47014206
22;        ZS(1,1)=S1
23;        ZS(1,2)=S1
24;        ZS(1,3)=S1
25;        ZS(2,1)=S2
26;        ZS(3,2)=S2
27;        ZS(4,3)=S2
28;        ZS(2,2)=S3
29;        ZS(2,3)=S3
30;        ZS(3,1)=S3
31;        ZS(3,3)=S3
32;        ZS(4,1)=S3
33;        ZS(4,2)=S3
34;        ZS(5,1)=S4
35;        ZS(6,2)=S4
36;        ZS(7,3)=S4
37;        ZS(5,2)=S5
38;        ZS(5,3)=S5
39;        ZS(6,1)=S5
40;        ZS(6,3)=S5
41;        ZS(7,1)=S5
42;        ZS(7,2)=S5
43;        WE(1)=0.1125
44;        WE(2)=0.06296959
45;        WE(3)=WE(2)
46;        WE(4)=WE(2)
47;        WE(5)=0.06619708
48;        WE(6)=WE(5)
49;        WE(7)=WE(5)
50;    50 RETURN
51;        END
```

```
      INIT                    3/ 1/79  15:35:46     PAGE  1
  1;      OVERLAY START
  2;      SUBROUTINE INIT
  3;C     GRID DIMENSION=M*N=IMD*IGD, ZTO=BOTTOM DEPTH, XTO=HORIZON. WIDTH
  4;C     INTRO. PARAMETERS OF POROUS MEDIUM AND ROCKS, AND BOUNDARY CONDI.
  5;      COMMON SGW1(22),SGW2(12),SGW3
  6;      COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
  7;      COMMON AR,ZM(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
  8;      COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
  9;      COMMON NT(223),NU(77),NW(77),NP(77)
 10;      COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
 11;      COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
 12;      COMMON AXE(6,6),ZP(1001),PP(1001),AX(1940)
 13;      EXTERNAL DENPRO
 14;C------------------------------------------------------------------
 15;C---FAI=POROSITY--BFAI=VERTICAL COMPRESSIBILITY OF POROUS MEDIUM
 16;C---ROS=ROCK DENSITY(IN PORO.)--CS=ROCK SPECIFIC HEAT(IN PORO.)
 17;C---COND=CONDUCTIVITY OF PORO. MEDIUM--RORCK=ROCK DENSITY
 18;C---CSRC=RORCK SPECIFIC HEAT--CONRCK=CONDUCTIVITY OF ROCK
 19;C---PERS=PERMEABILITY AT SURFACE--DT=TIME INCREDIMENT--TS=SURFACE TEMP.
 20;C---GRAV=GRAVITY--DTDZ=TEMPERATURE GRADIENT AT INITIAL TIME
 21;C------------------------------------------------------------------
 22;      FAI=0.1
 23;      BFAI=2.5E-11
 24;      ROS=2.7
 25;      CS=0.93
 26;      COND=5.1*4.185*10000.
 27;      RORCK=2.7
 28;      CSRCK=0.93
 29;      CONRCK=5.5*4.185*10000.
 30;      ACCEPT"INITIAL DT FOR ENERGY CYCLES=?",DT
 31;      ACCEPT"SURFACE TEMPERATURE(DEG.)=?",TS
 32;      ACCEPT"TEMPERATURE GRADIENT(DEG./KM)=?",DTDZ
 33;      GRAV=980.
 34;C------------------------------------------------------------------
 35;      ACCEPT"CRITICAL RAYLEIGH NUMBER=?",RAC
 36;      TYPE"RAYLEIGH NUMBER=",RAC
 37;      DZ=2TO/1000.
 38;      DO 11 I=1,223
 39;   11 NT(I)=0
 40;      DO 12 I=1,77
 41;      NU(I)=0
 42;   12 NW(I)=0
 43;C-----B. C. NODES FOR TEMPERATURE---
 44;      I2=2*IMD+1
 45;      DO 10 I=1,I2   <--
 46;      NT(I)=555
 47;   10 NT(212+I)=555
 48;      NT(12)=555
 49;      NT(22)=555
 50;      NT(202)=555
 51;      NT(212)=555
 52;      DO 15 I=1,9
 53;      NT(20*I+2+I)=555
 54;   15 NT(20*I+I+2+10)=555
 55;C-----B. C. NODES FOR VELOCITY U---
 56;      DO 20 I=1,9
 57;      NU(NF(66+I))=555
 58;   20 NU(NF(150+I))=555  <--
```

}(1)

```
      INIT                    3/ 1/79  15:35:46    PAGE  2
59;C-----8.C. NODES FOR VELOCITY W---
60;       DO 30 I=1,5   ←
61;       NW(NF(67+21*(I-1)))=555
62;   30 NW(NF(75+21*(I-1)))=555
63;C-----8. C. NODES FOR PRESSURE P---
64;       DO 60 I=1,5
65;   60 NP(NF(21*(I-1)+67))=555   ←
66;       KIMP=0
67;   18 FORMAT(1X,"PERMEABILITY COEFFICIENTS=? 2(F5.0)")
68;   19 FORMAT(2F5.0)
69;C-----INPUT VALUES AT 8. C. FOR T, U, W, P---
70;C-----CALCULATE PRESSURE WITH RESPECT TO DEPTH-------
71;       P=1.
72;       T=TS
73;       ZP(1)=0.
74;       PP(1)=P
75;       DO 510 K=2,1001
76;       ZP(K)=DZ*(K-1)
77;       PO=P
78;       DEN=DENPRO(1,1,T,P)
79;       P=P+DEN*GRAV*DZ/10.
80;       T=T+DZ*DTDZ
81;       DENN=DENPRO(1,1,T,P)
82;       DP=(DEN+DENN)/2.*DZ*GRAV/10.
83;       P=PO+DP
84;       PP(K)=P
85;   510 CONTINUE
86;   123 KIMP=KIMP+1
87;       ACCEPT"PERMEABILITY AT SURFACE=?",PERS
88;       TYPE PERS
89;       WRITE(10,18)
90;       READ(11,19)PERM1,PERM2
91;C-----------------------------------------------------
92;       DO 1000 KE=1,NET
93;       DO 25 I=1,3
94;       READ(35,69)NG(I),XO(I),ZO(I)
95;   25 CONTINUE
96;   69 FORMAT(1X,I3,2X,F8.4,2X,F8.4)
97;       NG(4)=(NG(1)+NG(2))/2
98;       NG(5)=(NG(2)+NG(3))/2
99;       NG(6)=(NG(1)+NG(3))/2
100;       XO(4)=(XO(1)+XO(2))/2
101;       XO(5)=(XO(2)+XO(3))/2
102;       XO(6)=(XO(1)+XO(3))/2
103;       ZO(4)=(ZO(1)+ZO(2))/2
104;       ZO(5)=(ZO(2)+ZO(3))/2
105;       ZO(6)=(ZO(1)+ZO(3))/2
106;       IF(KIMP.GE.2)GO TO 35
107;C-----INPUT TEMPERATURE AT 8. C. ---
108;       DO 100 I=1,6
109;       T2(NG(I))=TS+ZO(I)*DTDZ
110;   100 T1(NG(I))=T2(NG(I))
111;C-----INPUT VELOCITY U,W AT 8. C. ---
112;       DO 200 I=1,6
113;       IF(NF(NG(I)).EQ.0)GO TO 200
114;       IF(NU(NF(NG(I))).EQ.555)U(NF(NG(I)))=0.
115;   200 CONTINUE
116;       DO 300 I=1,6
```

$\left.\begin{array}{l} \\ \\ \\ \end{array}\right\}$ (1)

```
117;        IF(NF(NG(I)).EQ.0)GO TO 300
118;        IF(NW(NF(NG(I))).EQ.555)W(NF(NG(I)))=0.
119;  300 CONTINUE
120;C-----INPUT PRESSURE AT B. C. ---
121;        DO 520 I=1,6
122;        J=NF(NG(I))
123;        IF(J.EQ.0)GO TO 520
124;        DO 531 K=1,1000
125;        SGN=(ZO(I)-ZP(K))*(ZO(I)-ZP(K+1))/ZP(K+1)
126;        IF(SGN.GT.1.0E-6)GO TO 531
127;        P2(J)=(PP(K)+(ZO(I)-ZP(K))/(ZP(K+1)-ZP(K))*(PP(K+1)-PP(K)))*1000000.
128;        P1(NF(NG(I)))=P2(NF(NG(I)))
129;  531 CONTINUE
130;  520 CONTINUE
131;C-----INPUT PERMEABILITY---
132;   35 DO 530 I=1,6
133;        IF(NF(NG(I)).EQ.0)GO TO 530
134;        PER(NF(NG(I)))=PERS*(1.+PERM1*ZO(I)+PERM2*ZO(I)*ZO(I))
135;  530 CONTINUE
136; 1000 CONTINUE
137;        REWIND 35
138;        T=T2(109)
139;        P=P2(35)
140;        RO3=DENPRO(1,0,T,P)
141;        AP3=DENPRO(2,0,T,P)
142;        CP3=DENPRO(4,0,T,P)
143;        UE3=DENPRO(5,0,T,P)
144;        TYPE"RO,AP,CP,UE,COND",RO3,AP3,CP3,UE3,COND
145;        RAYLE=GRAV*AP3*DTDZ*(ZTO)**2*(RO3**2)*PERS*CP3*100000./(UE3*COND)
146;        TYPE"CALCULATED RAYLEIGH NUMBER=",RAYLE
147;        PERS1=RAC*PERS/RAYLE
148;        TYPE"PERMEABILITY AT SURFACE SHULD BE >=",PERS1
149;        IF(KIMP.GE.2)GO TO 121
150;        GO TO 123
151;C-----INTRODUCING DISTURBANT PRESSURE P ---
152;  121 DO 50 I=1,77
153;        U(I)=0.
154;   50 W(I)=0.
155;        P2(NF(89))=1.05*P1(NF(89))
156;        P2(NF(131))=0.95*P1(NF(131))
157;        P2(NF(90))=1.1*P1(NF(90))
158;        P2(NF(132))=0.9*P1(NF(132))
159;        P2(NF(91))=1.15*P1(NF(91))
160;        P2(NF(133))=0.85*P1(NF(133))
161;        P2(NF(92))=1.2*P1(NF(92))
162;        P2(NF(134))=0.8*P1(NF(134))
163;        P2(NF(93))=1.25*P1(NF(93))
164;        P2(NF(135))=0.75*P1(NF(135))
165;        P2(NF(94))=1.2*P1(NF(94))
166;        P2(NF(136))=0.8*P1(NF(136))
167;        P2(NF(95))=1.1*P1(NF(95))
168;        P2(NF(137))=0.9*P1(NF(137))
169;        RETURN
170;        END
171;
```

```
    DENPRO                        3/ 1/79  15:48:21     PAGE  1
 1;       FUNCTION DENPRO(IPT,IBAR,TEMP,PRESS)
 2;C       TO PROVIDE THE HYDRODYNAMIC PROPERTIES  EFFECTED BY TEMP. & PRESS.
 3;C       RO=DENSITY, AP=EXPANSION COEF., BA=COMPRESSIBILITY
 4;C       CP=SPECIFIC HEAT AT CONST. PRESS., UE=DYNAMIC VISCOSITY
 5;C       IPT=1 FOR RO, 2 FOR AP, 3 FOR BA, 4 FOR CP, 5 FOR UE
 6;C       INPUT"TEMP(DEG. C), PRESS(C.G.S.) IF IBAR=0, (BAR) IF IBAR=1
 7;C       OUTPUT "RO,AP,BA,CP,UE IN C.G.S."
 8;       COMMON A(22),C(12),A0
 9;       COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
10;       COMMON AR,ZN(7),ZS(7,3),UE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
11;       COMMON NG(6),XO(6),ZO(6),XV(5,5),ZV(5,5),NF(263)
12;       COMMON NT(263),NU(77),NW(77),NP(77)
13;       COMMON U(77),W(77),T1(263),T2(263),P1(77),P2(77)
14;       COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
15;       A(1)=-5.422863673E2
16;       A(2)=-2.096666205E4
17;       A(3)=3.941296787E4
18;       A(4)=-6.733277739E4
19;       A(5)=9.902381028E4
20;       A(6)=-1.093911774E5
21;       A(7)=8.590841667E4
22;       A(8)=-4.511168742E4
23;       A(9)=1.418138926E4
24;       A(10)=-2.017271113E3
25;       A(11)=7.982692717
26;       A(12)=-2.616571843E-2
27;       A(13)=1.522411790E-3
28;       A(14)=2.284279054E-2
29;       A(15)=2.421647003E2
30;       A(16)=1.269716088E-10
31;       A(17)=2.074838328E-7
32;       A(18)=2.17402835E-9
33;       A(19)=1.105718498E-9
34;       A(20)=1.293441934E1
35;       A(21)=1.308119072E-5
36;       A(22)=6.847626338E-14
37;       C(1)=8.438375485E-1
38;       C(2)=5.362162162E-4
39;       C(3)=1.72
40;       C(4)=7.342278489E-2
41;       C(5)=4.97585887E-2
42;       C(6)=6.5371543E-1
43;       C(7)=1.15E-6
44;       C(8)=1.5188E-5
45;       C(9)=1.4188E-1
46;       C(10)=7.002753165
47;       C(11)=2.995284926E-4
48;       C(12)=2.04E-1
49;       A0=6.824687741E3
50;       S=(TEMP+273.15)/647.3
51;       IF(IBAR.EQ.0)PRESS=PRESS/1000000.
52;       B=PRESS/221.2
53;       YS=1.-C(1)*S*S-C(2)*S**(-6)
54;       Z=YS+SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
55;       DYS=-2.*C(1)*S+6.*C(2)*S**(-7)
56;       DYSS=-2.*C(1)-42.*C(2)*S**(-8)
57;       DZS=DYS+(C(3)*YS*DYS-C(4))/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
58;       SQ=SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
```

```
       DENPRO                    3/ 1/79  15:48:21    PAGE  2
 59;         DZSS=DYSS+(C(3)*DYS*DYS+C(3)*YS*DYSS)/SQ
 60;         DZSS=DZSS-(C(3)*YS*DYS-C(4))**2/(SQ*SQ*SQ)
 61;         DZB=C(5)/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
 62;         DXS=-(5./17.)*A(11)*C(5)*(Z**(-22./17.))*DZS
 63;         DXS=DXS+(A(13)+2.*A(14)*S-A(15)*18.*(C(6)-S)**9
 64;        :-A(16)*19*S**18/(C(7)+S**19)**2)
 65;         DXS=DXS+11.*(A(17)+2.*A(18)*B+3.*A(19)*B*B)*S**18/(C(8)+S**11)**2
 66;         DXS=DXS-18.*A(20)*(S**17)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
 67;         DXS=DXS-2.*A(20)*S**19*(-3./(C(10)+B)**4+C(11))-3*A(21)*B*B
 68;        :-80.*A(22)*S**(-21)*B*B*B
 69;         DXB=(-5./17.)*A(11)*C(5)*DZB*(Z**(-22./17.))
 70;         DXB=DXB-(2.*A(18)+6.*A(19)*B)/(C(8)+S**11)
 71;         DXB=DXB-A(20)*S**18*(C(9)+S*S)*12/(C(10)+B)**5
 72;         DXB=DXB+6.*A(21)*(C(12)-S)*B+12.*A(22)*B*B/S**20
 73;C-----FOR DENSITY------
 74;         X1=A(11)*C(5)*(Z**(-5./17.))
 75;         X1=X1+(A(12)+A(13)*S+A(14)*S*S+A(15)*(C(6)-S)**18+A(16)/
 76;        :(C(7)+S**19))
 77;         X1=X1-(A(17)+2.*A(18)*B+3.*A(19)*B*B)/(C(8)+S**11)
 78;         X1=X1-A(20)*(S**18)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
 79;         X1=X1+3.*A(21)*(C(12)-S)*B*B+4.*A(22)*(S**(-20))*B*B*B
 80;         DENSITY=1./(3.17*X1)
 81;         DENPRO=DENSITY
 82;         IF(IPT.LE.1)GO TO 50
 83;C-----FOR EXPANSION COEF "AP" AND COMPRESIBILITY "BA"------
 84;         DENPRO=DENSITY*3.17/647.3*DXS
 85;         IF(IPT.LE.2)GO TO 50
 86;         DENPRO=-DENSITY*3.17/221.2*DXB*1.8E-6
 87;         IF(IPT.LE.3)GO TO 50
 88;C-----FOR SPECIFIC HEAT "CP" ------
 89;         SI=420.*A(22)*(B**4)/S**22-A(20)*(S**16)*(386.*C(9)+380.*S*S)*
 90;        :(1./(C(10)+B)**3+C(11)*B)
 91;         SI=SI-(242*(S**20)/(C(8)+S**11)**3-110*(S**9)/(C(8)+S**11)**2*
 92;        :(A(17)*B+A(18)*B*B+A(19)*B*B*B)
 93;         SI=SI+B*(2*A(14)+90*A(15)*(C(6)-S)**8+722*A(16)*(S**36)
 94;        :/(C(7)+S**19)**3-342*A(16)*S**17/(C(7)+S**19)**2)
 95;         SI=SI+A(11)*(17*DZSS/29-17*DYSS/12)*(Z**(12./17.))
 96;         SI=SI+A(11)*(24*DZS/29-2*DYS)*DZS*(Z**(-5./17.))
 97;         SI=SI+A(11)*(12*Z/29-YS)*(Z**(-5./17.)*DZSS-5*(Z**(-22./17.))*DZS
 98;        :*DZS/17)
 99;         SI=SI+2*A(3)+6*A(4)*S+12*A(5)*S**2+20*A(6)*S**3+30*A(7)*S**4
100;         SI=SI+42*A(8)*S**5+56*A(9)*S**6+72*A(10)*S**7-A0/S
101;         DENPRO=-221.2*3.17*SI*S/647.3*1.0E6
102;         IF(IPT.LE.4)GO TO 50
103;C-----FOR VISCOSITY "UE"------
104;C        UE(NFIND)=0.000001*(241.4*(10.**(247.8/(T2(IND)+133.15))))
105;         UEA=(TEMP-150.)/100.
106;         UED=530.+300.*UEA-26.*UEA**3
107;         DENPRO=1./UED
108;  50 RETURN
109;         END
```

```
      PROPT                        3/ 1/79  15:42:45      PAGE  1
 1;          OVERLAY REVISE
 2;          SUBROUTINE PROPT
 3;C         TO PROVIDE THE HYDRODYNAMIC PROPERTIES  EFFECTED BY TEMP. & PRESS.
 4;C         RO=DENSITY, AP=EXPANSION COEF., BA=COMPRESSIBILITY
 5;C         CP=SPECIFIC HEAT AT CONST. PRESS., UE=DYNAMIC VISCOSITY
 6;C         INPUT "P2(C.G.S.),T2(DEGREE C)",OUTPUT "RO,AP,BA,CP IN C.G.S."
 7;          COMMON A(22),C(12),A0
 8;          COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 9;          COMMON AR,ZN(7),ZS(7,3),UE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
10;          COMMON NG(6),XO(6),ZO(6),XV(5,5),ZV(5,5),NF(223)
11;          COMMON NT(223),NU(77),NW(77),NP(77)
12;          COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
13;          COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
14;          DO 100 IND=1,NOD
15;          NFIND=NF(IND)
16;          IF(NFIND.EQ.0)GO TO 100
17;          S=(T2(IND)+273.15)/647.3
18;          B=P2(NF(IND))/(221.2*1000000.)
19;          YS=1.-C(1)*S*S-C(2)*S**(-6)
20;          Z=YS+SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
21;          DYS=-2.*C(1)*S+6.*C(2)*S**(-7)
22;          DYSS=-2.*C(1)-42.*C(2)*S**(-8)
23;          DZS=DYS+(C(3)*YS*DYS-C(4))/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
24;          SQ=SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
25;          DZSS=DYSS+(C(3)*DYS*DYS+C(3)*YS*DYSS)/SQ
26;          DZSS=DZSS-(C(3)*YS*DYS-C(4))**2/(SQ*SQ*SQ)
27;          DZB=C(5)/SQRT(C(3)*YS*YS-2.*C(4)*S+2.*C(5)*B)
28;          DXS=-(5./17.)*A(11)*C(5)*(Z**(-22./17.))*DZS
29;          DXS=DXS+(A(13)+2.*A(14)*S-A(15)*18.*(C(6)-S)**9
30;          :-A(16)*19*S**18/(C(7)+S**19)**2)
31;          DXS=DXS+11.*(A(17)+2.*A(18)*B+3.*A(19)*B*B)*S**10/(C(8)+S**11)**2
32;          DXS=DXS-18.*A(20)*(S**17)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
33;          DXS=DXS-2.*A(20)*S**19*(-3./(C(10)+B)**4+C(11))-3*A(21)*B*B
34;          :-80.*A(22)*S**(-21)*B*B*B
35;          DXB=(-5./17.)*A(11)*C(5)*DZB*(Z**(-22./17.))
36;          DXB=DXB-(2.*A(18)+6.*A(19)*B)/(C(8)+S**11)
37;          DXB=DXB-A(20)*S**18*(C(9)+S*S)*12/(C(10)+B)**5
38;          DXB=DXB+6.*A(21)*(C(12)-S)*B+12.*A(22)*B*B/S**20
39;C-----FOR DENSITY------
40;          X1=A(11)*C(5)*(Z**(-5./17.))
41;          X1=X1+(A(12)+A(13)*S+A(14)*S*S+A(15)*(C(6)-S)**10+A(16)/
42;          :(C(7)+S**19))
43;          X1=X1-(A(17)+2.*A(18)*B+3.*A(19)*B*B)/(C(8)+S**11)
44;          X1=X1-A(20)*(S**18)*(C(9)+S*S)*(-3./(C(10)+B)**4+C(11))
45;          X1=X1+3.*A(21)*(C(12)-S)*B*B+4.*A(22)*(S**(-20))*B*B*B
46;          RO(NFIND)=1./(3.17*X1)
47;C-----FOR EXPANSION COEF "AP" AND COMPRESIBILITY "BA"------
48;          AP(NFIND)=RO(NFIND)*3.17/647.3*DXS
49;          BA(NFIND)=-RO(NFIND)*3.17/221.2*DXB*1.0E-6
50;C-----FOR SPECIFIC HEAT "CP" ------
51;          SI=420.*A(22)*(B**4)/S**22-A(20)*(S**16)*(306.*C(9)+380.*S*S)*
52;          :(1./(C(10)+B)**3+C(11)*B)
53;          SI=SI-(242*(S**20)/(C(8)+S**11)**3-110*(S**9)/(C(8)+S**11)**2)*
54;          :(A(17)*B+A(18)*B*B+A(19)*B*B*B)
55;          SI=SI+B*(2*A(14)+90*A(15)*(C(6)-S)**8+722*A(16)*(S**36)
56;          :/(C(7)+S**19)**3-342*A(16)*S**17/(C(7)+S**19)**2)
57;          SI=SI+A(11)*(17*DZSS/29-17*DYSS/12)*(Z**(12./17.))
58;          SI=SI+A(11)*(24*DZS/29-2*DYS)*DZS*(Z**(-5./17.))
```

```
      PROPT                    3/ 1/79  15:42:45    PAGE  2
59;       SI=SI+A(11)*(12*Z/29-YS)*(Z**(-5./17.)*DZSS-5*(Z**(-22./17.))*DZS
60;      .*DZS/17)
61;       SI=SI+2*A(3)+6*A(4)*S+12*A(5)*S**2+20*A(6)*S**3+30*A(7)*S**4
62;       SI=SI+42*A(8)*S**5+56*A(9)*S**6+72*A(10)*S**7-A0/S
63;       CP(NFIND)=-221.2*3.17*SI*S/647.3*1.0E6
64;C-----FOR VISCOSITY "UE"------
65;C     UE(NFIND)=0.000001*(241.4*(10.**(247.8/(T2(IND)+133.15))))
66;       UEA=(T2(IND)-150.)/100.
67;       UED=538.+300.*UEA-26.*UEA**3
68;       UE(NFIND)=1./UED
69;  100 CONTINUE
70;       RETURN
71;       END
```

```
        STIFP                    3/ 1/79  15:49:48     PAGE  1
  1;         OVERLAY MAIN2
  2;         SUBROUTINE STIFP(ISCH,NEQ,NROWX,NROWY)
  3;C     CONVECTION IN A POROUS MEDIUM OF FAULT ZONES
  4;C     TO FORM STIFFNESS MATRICES FOR PRESSURE "P" SOLUTIONS
  5;C     ISCH=1 FOR BACK DIFF. SCHEME; ISCH=2 FOR CRANK-NICHOLSON SCHEME
  6;C     NEQ=TOTAL EQNS. TO BE SOLVED; NROW=MAX. DIMENSOION OF ROWS--
  7;         COMMON SGU1(22),SGU2(12),SGU
  8;         COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
  9;         COMMON AR,ZN(7),ZS(7,3),UE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
 10;         COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
 11;         COMMON NT(223),NU(77),NW(77),NP(77)
 12;         COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
 13;         COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
 14;         COMMON AXE(6,6),AX(77,25),NCOAX(77,25),YY(77)
 15;         COMMON AYE(6,6),AY(77,25),NCOAY(77,25)
 16;         EXTERNAL BA2P,DXZ2P,COMPP,COMPB
 17;C
 18;         DT1=1./DTP
 19;         DT2=1./DT
 20;         DO 505 I=1,77
 21;         YY(I)=0.
 22;         DO 504 J=1,25
 23;         AX(I,J)=0.
 24;         NCOAX(I,J)=0
 25;         AY(I,J)=0.
 26;         NCOAY(I,J)=0
 27;   504 CONTINUE
 28;   505 CONTINUE
 29;         DO 1000 KE=1,NET
 30;         DO 25 I=1,3
 31;    25 READ(35,69)NG(I),XO(I),ZO(I)
 32;    69 FORMAT(1X,I3,2X,F8.4,2X,F8.4)
 33;         IF(KE.GE.51.AND.KE.LE.54)GO TO 1000
 34;         IF(KE.LE.34.OR.KE.GE.71)GO TO 1000
 35;C     CALCULATING THE ELEMENT STIFFNESS MATRIX IN ONE ELEMENT
 36;         XO(4)=XO(1)
 37;         XO(5)=XO(2)
 38;         ZO(4)=ZO(1)
 39;         ZO(5)=ZO(2)
 40;         NG(4)=(NG(1)+NG(2))/2
 41;         NG(5)=(NG(2)+NG(3))/2
 42;         NG(6)=(NG(1)+NG(3))/2
 43;         AR=(ZO(2)*XO(3)+ZO(1)*XO(2)+XO(1)*ZO(3)-XO(1)*ZO(2)-XO(2)*ZO(3)
 44;        8-XO(3)*ZO(1))/2.
 45;         AR=ABS(AR)
 46;         DO 10 I=1,5
 47;         DO 10 J=1,5
 48;         X(I,J)=XO(I)-XO(J)
 49;    10 Z(I,J)=ZO(I)-ZO(J)
 50;         DJACO=Z(1,3)*X(2,3)-Z(2,3)*X(1,3)
 51;         RAB1=0.3333333*(RO(NF(NG(1)))+RO(NF(NG(2)))+RO(NF(NG(3))))
 52;         RAB2=0.3333333*(AP(NF(NG(1)))+AP(NF(NG(2)))+AP(NF(NG(3))))
 53;         RAB3=0.3333333*(BA(NF(NG(1)))+BA(NF(NG(2)))+BA(NF(NG(3))))
 54;         RAB5=0.3333333*(UE(NF(NG(1)))+UE(NF(NG(2)))+UE(NF(NG(3))))
 55;         RAB6=0.3333333*(PER(NF(NG(1)))+PER(NF(NG(2)))+PER(NF(NG(3))))
 56;         DO 15 I=1,6
 57;         YNG=0.
 58;         DO 17 J=1,6
```

(1)

(2)

```
     STIFP                          3/ 1/79  15:49:40     PAGE  2
59;        SI=0.
60;        SAX=0.
61;        SAY=0.
62;        SYY=0.
63;        SYYBORROW=0.
64;        DO 16 KW=1,7
65;        P=ZS(KW,1)
66;        Q=ZS(KW,2)
67;        R=ZS(KW,3)
68;        SAX=SAX+WE(KW)*8A2P(J)*8A2P(I)
69;        SAY=SAY-WE(KW)*(DXZ2P(1,J)*DXZ2P(1,I)+DXZ2P(-1,J)*DXZ2P(-1,I))
70;        SYYBORROW=SYYBORROW+WE(KW)*8A2P(J)*8A2P(I)
71;     16 CONTINUE
72;        SAXT=SAX*(-8FAI+FAI*RA83)*RA81*RA85*RA85
73;        SAYT=SAY*RA81*RA85*RA86
74;        SAX=DT1*ISCH*SAXT-SAYT
75;        SAY=DT1*ISCH*SAXT+(ISCH-1)*SAYT
76;        DTDT=(T2(NG(J))-T1(NG(J)))*DT2
77;        YNG=YNG+FAI*RA81*RA82*RA85*RA85*SYYBORROW*DTDT
78;        AXE(I,J)=SAX*DJACO
79;        AYE(I,J)=SAY*DJACO
80;        YY(NF(NG(I)))=YY(NF(NG(I)))+YNG*DJACO
81;     17 CONTINUE
82;     15 CONTINUE
83;        CALL COMPP
84;        CALL COMPB
85;   1000 CONTINUE
86;        REWIND 35
87;C        ---MODIFY THE EQUATIONS AND INTRODUCE B.C. VALUES---
88;        DO 30 I=1,77
89;        IF(NP(I).NE.555)GO TO 30
90;        DO 31 J=1,25
91;        IF(NCOAX(I,J).NE.I)GO TO 31
92;        YY(I)=AX(I,J)*P2(I)*1.0E30
93;        AX(I,J)=1.0E30*AX(I,J)
94;        GO TO 30
95;     31 CONTINUE
96;     30 CONTINUE
97;        NEQ=77
98;        NROWX=0
99;        DO 50 I=1,NEQ
100;        NR=0
101;        DO 51 J=1,25
102;        IF(NCOAX(I,J).EQ.0)GO TO 51
103;        NR=NR+1
104;     51 CONTINUE
105;        IF(NR.GT.NROWX)NROWX=NR
106;     50 CONTINUE
107;        NROWY=0
108;        DO 52 I=1,NEQ
109;        NR=0
110;        DO 53 J=1,25
111;        IF(NCOAY(I,J).EQ.0)GO TO 53
112;        NR=NR+1
113;     53 CONTINUE
114;        IF(NR.GT.NROWY)NROWY=NR
115;     52 CONTINUE
116;        NROW=NROWX
```

(3)

(4)

```
      STIFP                    3/ 1/79  15:49:40     PAGE  3
117;      IF(MROWX.LT.MROWY)MROW=MROWY
118;      DO 60 I=1,NEQ  ⟵─────────────────────────────────────┐
119;      WRITE(33,38)YY(I)                                     │
120;      DO 60 J=1,MROW                                        │ }( 5 )
121;      WRITE(33,39)AX(I,J),NCOAX(I,J),AY(I,J),NCOAY(I,J)     │
122;   60 CONTINUE  ⟵──────────────────────────────────────────┘
123;   38 FORMAT(1X,E13.6)
124;   39 FORMAT(1X,E13.6,1X,I2,1X,E13.6,1X,I2)
125;      REWIND 33
126;      RETURN
127;      END
```

```
     BA2P                      3/ 1/79  15:51:11     PAGE  1
  1;      FUNCTION BA2P(IBA)
  2;C      BA2P=SHAPE FUNCTIONS INTERMS OF AREA COORDINATES(P=L1;Q=L2;R=L3)
  3;       COMMON SGU1(22),SGU2(12),SGU3
  4;       COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
  5;       COMMON AR,ZH(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
  6;       PQR(1)=P
  7;       PQR(2)=Q
  8;       PQR(3)=R
  9;       PQR(4)=P
 10;       IF(IBA.GT.3)GO TO 20
 11;       BA2P=2.*PQR(IBA)*PQR(IBA)-PQR(IBA)
 12;       GO TO 50
 13;  20 KN=IBA-3
 14;       BA2P=4.*PQR(KN)*PQR(KN+1)
 15;  50 RETURN
 16;       END
```

```
    DXZ2P                      3/ 1/79  15:52: 5      PAGE  1
 1;       FUNCTION DXZ2P(ND,M)
 2;C       DXZ2P=DERIVATION OF THE SHAPE FUNCTION "N" WITH RESPECT TO
 3;C       "X" OR "Z" IN TERMS OF AREA COORDINATES
 4;C       ND=1 FOR D(N)/DX; ND=-1 FOR D(N)/DZ
 5;       COMMON SGW1(22),SGW2(12),SGW3
 6;       COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 7;       COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
 8;       COMMON NG(6),X0(6),Z0(6),X(5,5),Z(5,5)
 9;       IF(AR.LE.1.0E-10.OR.AR.GT.1.0E5)WRITE(25,18)AR
10;    18 FORMAT(1X,"AR",E12.5)
11;       PQR(1)=P
12;       PQR(2)=Q
13;       PQR(3)=R
14;       PQR(4)=P
15;       IF(ND.EQ.-1)GO TO 10
16;       IF(M.LE.3)DXZ2P=(4.*PQR(M)-1.)*Z(M+2,M+1)/(2.*AR)
17;       KN=M-3
18;       IF(M.GT.3)DXZ2P=2.*(PQR(KN+1)*Z(KN+2,KN+1)+PQR(KN)*Z(KN,KN+2))/AR
19;       GO TO 50
20;    10 IF(M.LE.3)DXZ2P=(4.*PQR(M)-1.)*X(M+1,M+2)/(2.*AR)
21;       KN=M-3
22;       IF(M.GT.3)DXZ2P=2.*(PQR(KN+1)*X(KN+1,KN+2)+PQR(KN)*X(KN+2,KN))/AR
23;    50 CONTINUE
24;       RETURN
25;       END
```

```
      COMPP                  3/ 1/79  15:53:14    PAGE  1
 1;        SUBROUTINE COMPP
 2;C        INSERT AE(6,6) IN TO A(IM,17)
 3;C        "NCOL"=COLUMN INDEX OF "A"
 4;C        "NG"=GLOBAL NODE INDEX
 5;        COMMON SGW1(22),SGW2(12),SGW3
 6;        COMMON FAI,BFAI,ROS,CS,COMB,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 7;        COMMON AR,ZM(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
 8;        COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
 9;        COMMON NT(223),NU(77),NW(77),NP(77)
10;        COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
11;        COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
12;        COMMON AE(6,6),A(77,25),NCOL(77,25),YY(77)
13;        COMMON AE2(6,6),A2(77,25),NCOL2(77,25)
14;        SBF=0.
15;        DO 70 I=1,6
16;        DO 70 J=1,6
17;        IF(ABS(AE(I,J)).GT.SBF)SBF=ABS(AE(I,J))
18;     70 CONTINUE
19;        DO 71 I=1,6
20;        DO 71 J=1,6
21;        S=AE(I,J)/SBF
22;        IF(ABS(S).LE.3.0E-6)AE(I,J)=0.
23;     71 CONTINUE
24;        DO 10 K=1,6
25;        IM=NF(NG(K))
26;        IF(IM.LE.0)GO TO 10
27;        DO 11 L=1,6
28;        IF(ABS(AE(K,L)).LE.3.0E-50)GO TO 11
29;        JM=NF(NG(L))
30;        IF(JM.LE.0)GO TO 11
31;        DO 16 JC=1,18
32;        IF(NCOL(IM,JC).EQ.0)GO TO 17
33;        IF(JM.LT.NCOL(IM,JC))GO TO 18
34;        IF(JM.NE.NCOL(IM,JC))GO TO 16
35;        A(IM,JC)=A(IM,JC)+AE(K,L)
36;        GO TO 11
37;     16 CONTINUE
38;        JC=JC-1
39;     17 A(IM,JC)=AE(K,L)
40;        NCOL(IM,JC)=JM
41;        GO TO 11
42;     18 LA=NCOL(IM,JC)
43;        ALA=A(IM,JC)
44;        DO 19 ML=JC,17
45;        NX=NCOL(IM,ML+1)
46;        ANX=A(IM,ML+1)
47;        NCOL(IM,ML+1)=LA
48;        A(IM,ML+1)=ALA
49;        IF(NX.EQ.0)GO TO 20
50;        LA=NX
51;     19 ALA=ANX
52;     20 A(IM,JC)=AE(K,L)
53;        NCOL(IM,JC)=JM
54;     11 CONTINUE
55;     10 CONTINUE
56;        RETURN
57;        END
```

```
 1;       SUBROUTINE COMPB
 2;C       INSERT AE2(6,6) IN TO A2(IM,17)
 3;C       "NCOL"=COLUMN INDEX OF "A"
 4;C       "NG"=GLOBAL NODE INDEX
 5;       COMMON SGW1(22),SGW2(12),SGW3
 6;       COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 7;       COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
 8;       COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
 9;       COMMON NT(223),NU(77),NW(77),NP(77)
10;       COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
11;       COMMON RO(77),AP(77),BA2(77),CP(77),UE(77),PER(77)
12;       COMMON AE(6,6),A(77,25),NCOL(77,25),YY(77)
13;       COMMON AE2(6,6),A2(77,25),NCOL2(77,25)
14;       SDF=0.
15;       DO 78 I=1,6
16;       DO 78 J=1,6
17;       IF(ABS(AE2(I,J)).GT.SDF)SDF=ABS(AE2(I,J))
18;    78 CONTINUE
19;       DO 71 I=1,6
20;       DO 71 J=1,6
21;       S=AE2(I,J)/SDF
22;       IF(ABS(S).LE.3.0E-6)AE2(I,J)=0.
23;    71 CONTINUE
24;       DO 10 K=1,6
25;       IM=NF(NG(K))
26;       IF(IM.LE.0)GO TO 10
27;       DO 11 L=1,6
28;       IF(ABS(AE2(K,L)).LE.5.0E-50)GO TO 11
29;       JM=NF(NG(L))
30;       IF(JM.LE.0)GO TO 11
31;       DO 16 JC=1,25
32;       IF(NCOL2(IM,JC).EQ.0)GO TO 17
33;       IF(JM.LT.NCOL2(IM,JC))GO TO 18
34;       IF(JM.NE.NCOL2(IM,JC))GO TO 16
35;       A2(IM,JC)=A2(IM,JC)+AE2(K,L)
36;       GO TO 11
37;    16 CONTINUE
38;       JC=JC-1
39;    17 A2(IM,JC)=AE2(K,L)
40;       NCOL2(IM,JC)=JM
41;       GO TO 11
42;    18 LA=NCOL2(IM,JC)
43;       ALA=A2(IM,JC)
44;       DO 19 ML=JC,24
45;       NX=NCOL2(IM,ML+1)
46;       ANX=A2(IM,ML+1)
47;       NCOL2(IM,ML+1)=LA
48;       A2(IM,ML+1)=ALA
49;       IF(NX.EQ.0)GO TO 20
50;       LA=NX
51;    19 ALA=ANX
52;    20 A2(IM,JC)=AE2(K,L)
53;       NCOL2(IM,JC)=JM
54;    11 CONTINUE
55;    10 CONTINUE
56;       RETURN
57;       END
```

```
        EQSO                    3/ 1/79  16: 1:16      PAGE  1
  1;        OVERLAY SOLVE
  2;        SUBROUTINE EQSO(NPT,NROW,NNN,ZTEST)
  3;        COMMON SGW1(22),SGW2(12),SGW3
  4;        COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
  5;        COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
  6;        COMMON NG(6),XO(6),ZO(6),XEO(5,5),Z(5,5),NF(223)
  7;        COMMON NT(223),NU(77),NW(77),NPP(77)
  8;        COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
  9;        COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
 10;        COMMON AXE(6,6),A(179,25),B(223),NCOL(179,25)
 11;        DIMENSION NPIV(179),IBANDW(179),NNCOL(179),AA(179)
 12;C----------------------------------------------------------------------
 13;C SOLVES "A"X=B
 14;C A=PARTIALLY PACKED MATRIX OF NON-ZERO COEFF.---(INPUT)
 15;C B=KNOWN VECTOR---(INPUT),SOLUTIONS(OUTPUT)
 16;C AA=SINGLE DIMENSIONED ARRAY FOR PIVOTAL ROW ELEMENTS
 17;C IBANDW=NO. OF NONZERO COEFF. IN EACH ROW
 18;C NCOL=INDICES MATRIX OF NONZERO COEFF. OF "A"---(INPUT)
 19;C NNCOL=ONE COLUMN ARRAY FOR PIVOTAL ROW INDICES
 20;C NNN=COLUMN DIMENSION FOR A & NCOL IN MAIN PROGRAM---(INPUT)
 21;C NPIV=ONE COLUMN ARRAY TO STORE PIVOTAL COLUMN
 22;C NPT=TOTAL EQUATIONS, TOTAL UNKNOWNS
 23;C ZTEST=VALUE BELOW WHICH ELEMENT MADE EQUAL TO ZERO
 24;C----------------------------------------------------------------------
 25;C@@@@@@INITIALIZE THE BANDTH COUNTER @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 26;        IF(NPT.NE.77)GO TO 100
 27;        DO 101 I=1,179
 28;        B(I)=0.
 29;        AA(I)=0.
 30;        NPIV(I)=0
 31;        DO 101 J=1,25
 32;        A(I,J)=0.
 33;        NCOL(I,J)=0
 34;   101 CONTINUE
 35;C      CALL FOPEN(33,"AXAY")
 36;        DO 200 I=1,77
 37;        READ(33,308)B(I)
 38;        DO 201 J=1,NROW
 39;        READ(33,309)A(I,J),NCOL(I,J),AA(J),NPIV(J)
 40;   201 CONTINUE
 41;        SUM=0.
 42;        DO 202 J=1,NROW
 43;        IF(NPIV(J).EQ.0)GO TO 203
 44;   202 SUM=SUM+AA(J)*P2(NPIV(J))
 45;   203 B(I)=B(I)+SUM
 46;   200 CONTINUE
 47;   308 FORMAT(E13.6)
 48;   309 FORMAT(E13.6,1X,I2,1X,E13.6,1X,I2)
 49;        REWIND 33
 50;C      --------------------------------
 51;   100 DO 972 I=1,179
 52;        AA(I)=0.
 53;        NPIV(I)=0
 54;        IBANDW(I)=0
 55;        NNCOL(I)=0
 56;   972 CONTINUE
 57;        IF(ABS(ZTEST).GT.0.0001)ZTEST=0.0
 58;        NSTOP=0
```

```
      EQSO                    3/ 1/79  16: 1:16     PAGE  2
59;        MAXWID=0
60;        DO 1 I=1,NPT
61;      1 IBANDW(I)=0
62;        DO 5 I=1,NPT
63;        DO 2 J=1,NWN
64;        NC=NCOL(I,J)
65;        IF(NC.EQ.0)GO TO 3
66;        IBANDW(I)=J
67;      2 CONTINUE
68;      3 IF(MAXWID.LT.J)MAXWID=J
69;        IF(J.NE.1)GO TO 5
70;        WRITE(31,18)I
71;     18 FORMAT(1X,"ALL ELEMENTS IN ROW",I4,"EQUAL TO ZERO,SINGULARITY")
72;        NSTOP=1
73;      5 CONTINUE
74;        IF(NSTOP.EQ.1)STOP
75;        NPT1=NPT-1
76;        DO 23 LL=1,NPT1
77;C00000 FINDING THE ROW WITH MINI. BANDWIDTH 000000000000000000000
78;        KK=10000
79;        DO 6 I=LL,NPT
80;        IC=IBANDW(I)
81;        IF(IC.LE.0)GO TO 6
82;        IF(IC.GE.KK)GO TO 6
83;        MINROW=I
84;        KK=IC
85;      6 CONTINUE
86;C00000 INTERCHANGE ROWS WITH LL 00000000000000000000000000000000000000000
87;        LM=IBANDW(LL)
88;        M=MINROW
89;        DO 7 I=1,LM
90;        NNCOL(I)=NCOL(M,I)
91;        NCOL(M,I)=NCOL(LL,I)
92;        AA(I)=A(M,I)
93;        A(M,I)=A(LL,I)
94;      7 CONTINUE
95;        SAVE=B(LL)
96;        B(LL)=B(M)
97;        B(M)=SAVE
98;        IBANDW(LL)=IBANDW(M)
99;        IBANDW(M)=LM
100;C00000 FINDING BIG A IN MINROW 000000000000000000000000000000000000000000
101;        NC=IBANDW(LL)
102;        NNCOL(NC+1)=0
103;        MINROW=LL
104;        A1=0.0
105;        I=MINROW
106;        DO 8 J=1,NC
107;        AAA=ABS(AA(J))
108;        IF(AAA.LT.A1)GO TO 8
109;        A1=AAA
110;        IY=J
111;      8 CONTINUE
112;        MAXCOL=NNCOL(IY)
113;        NPIV(LL)=MAXCOL
114;C00000 NORMALIZE THE MINROW 000000000000000000000000000000000000000000000
115;        X=AA(IY)
116;        DO 9 J=1,NC
```

```
       EQSO                 3/ 1/79  16: 1:16    PAGE  3
117;      AA(J)=AA(J)/X
118;      NCOL(MINROW,J)=NNCOL(J)
119;    9 A(MINROW,J)=AA(J)
120;      B(MINROW)=B(MINROW)/X
121;      A(MINROW,IY)=1.0
122;      AA(IY)=1.0
123;C00000 FINDING THE ROWS WHICH CONTAIN MAXCOL 00000000000000000000000000000
124;      LL1=LL+1
125;      DO 22 I=LL1,NPT
126;      IF(IBANDW(I).EQ.0)GO TO 22
127;      NC=IBANDW(I)
128;      DO 21 J=1,NC
129;      IF(NCOL(I,J)-MAXCOL)21,10,22
130;C00000 IF NCOL(I,J) IS EQUAL TO MAXCOL THEN ROW CONTAINS THE VARIABLE
131;C00000 NOPROW=THE ROW BEING OPERATED 000000000000000000000000000000000000
132;   10 NOPROW=I
133;      JKOP=1
134;      JKPI=1
135;      C=-A(NOPROW,J)
136;      B(NOPROW)=B(MINROW)*C+B(NOPROW)
137;   11 CONTINUE
138;      IF(NNCOL(JKPI).EQ.0)GO TO 22
139;      IF(NCOL(NOPROW,JKOP).EQ.0)GO TO 12
140;      IF(NNCOL(JKPI)-NCOL(NOPROW,JKOP))12,14,20
141;   12 IBANDW(I)=IBANDW(I)+1
142;      IF(MAXWID.LT.IBANDW(I))MAXWID=IBANDW(I)
143;      IF(MAXWID.GT.NNN)GO TO 31
144;      II=IBANDW(I)
145;      JKL=JKOP+1
146;   13 IX=II-1
147;      A(NOPROW,II)=A(NOPROW,IX)
148;      NCOL(NOPROW,II)=NCOL(NOPROW,IX)
149;      II=IX
150;      IF(IX.GE.JKL)GO TO 13
151;      A(NOPROW,JKOP)=AA(JKPI)*C
152;      NCOL(NOPROW,JKOP)=NNCOL(JKPI)
153;      IX=NCOL(NOPROW,JKOP)
154;      GO TO 19
155;C00000MINROW AND THE ROW BEING CONSIDERED CONTAIN THIS ELEMENT SHIFTING
156;C00000 OF BOTH ROWS IS DONE AND NOPROW IS OPERATED 00000000000000000000
157;   14 IX=NCOL(NOPROW,JKOP)
158;      IF(IX.EQ.MAXCOL)GO TO 15
159;      X=AA(JKPI)*C+A(NOPROW,JKOP)
160;      A(NOPROW,JKOP)=X
161;C00000 TESTING TO SEE IF ANY OTHER ELEMENTS WERE ELIMINATED OTHER THAN
162;C20000 MAXCOL IN THE NOPROW 0000000000000000000000000000000000000000000
163;      ATEST=ABS(X)-ZTEST
164;      IF(ATEST.GT.0.)GO TO 19
165;   15 IBANDW(NOPROW)=IBANDW(NOPROW)-1
166;      IF(IBANDW(NOPROW))16,16,17
167;   16 WRITE(31,28)MINROW,MAXCOL,NOPROW
168;   28 FORMAT("MATRIX IS SINGULAR","MINROW=",I4,"MAXCOL=",I4,"ROW
169;      0OPERATED",I4)
170;      STOP
171;   17 IX=IBANDW(NOPROW)
172;      DO 36 NK=JKOP,IX
173;      A(I,NK)=A(I,NK+1)
174;   36 NCOL(I,NK)=NCOL(I,NK+1)
```

```
        EQSO                       3/ 1/79  16: 1:16      PAGE  4
175;        IX=IX+1
176;        NCOL(I,IX)=0
177;        A(I,IX)=0.0
178;        JKPI=JKPI+1
179;        GO TO 11
180;    19 JKPI=JKPI+1
181;C00000 MINROW DOES NOT CONTAIN THIS ELEMENT SHIFT NOPROW AND CONTINUE
182;    20 JKOP=JKOP+1
183;        GO TO 11
184;    21 CONTINUE
185;    22 CONTINUE
186;    23 CONTINUE
187;C00000 ELIMINATE MINROW AND MAXCOL FROM BEING CONSIDERED AGAIN 0000
188;        NPIV(NPT)=NCOL(NPT,1)
189;        WRITE(31,38)MAXWID
190;    38 FORMAT(1X,"MAXWID=",I4,"ROW DIMENSIONS OF A & NCOL")
191;C00000 BACK SUBSTITUTION 0000000000000000000000000000000000000000
192;        DO 24 I=1,NPT
193;    24 AA(I)=0.0
194;        DO 27 I=1,NPT
195;        II=NPT-I+1
196;        LM=IBANDW(II)
197;        NP=NPIV(II)
198;        IF(NP.EQ.0)GO TO 27
199;        DO 26 J=1,LM
200;        NN=NCOL(II,J)
201;        IF(NN.EQ.NP)GO TO 25
202;        B(II)=B(II)-AA(NN)*A(II,J)
203;        GO TO 26
204;    25 IJ=J
205;    26 CONTINUE
206;        AA(NP)=B(II)/A(II,IJ)
207;    27 CONTINUE
208;C00000 STORE THE SOLUTION IN "B" VECTOR 0000000000000000000000000000
209;        DO 37 I=1,NPT
210;    37 B(I)=AA(I)
211;        GO TO 33
212;    31 WRITE(31,58)I
213;        WRITE(31,68)LL
214;        STOP
215;    58 FORMAT(1X,"COLUMN DIMENSIONS OF NCOL & A EXCEEDED IN ROW",I4)
216;    68 FORMAT(1X,"WHILE THE NUMBER OF ROWS OPERATED WERE",I4)
217;    33 RETURN
218;        END
```

```
      STIFF                    3/ 1/79  15:56:56     PAGE  1
 1;        OVERLAY MAIN
 2;        SUBROUTINE STIFF(ISCH,ID,MEQ,MROW)
 3;C       CONVECTION IN A POROUS MEDIUM OF FAULT ZONES
 4;C       TO FORM STIFFNESS MATRICES, ID=1 FOR "U", ID=2 FOR "V", ID=4 FOR "T"
 5;C       ISCH=1 FOR BACK DIFF. SCHEME; ISCH=2 FOR CRANK-NICHOLSON SCHEME
 6;C       MEQ=TOTAL EQNS. TO BE SOLVED; MROW=MAX. DIMENSOION OF ROWS--
 7;        COMMON SGW1(22),SGW2(12),SGW
 8;        COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 9;        COMMON AR,ZM(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IMD,IGD,ZTA
10;        COMMON NG(6),XO(6),ZO(6),X(5,5),Z(5,5),NF(223)
11;        COMMON NT(223),NU(77),NW(77),NP(77)
12;        COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
13;        COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
14;        COMMON AXE(6,6),AX(179,25),YY(223),NCOAX(179,25)
15;        EXTERNAL BA2,DXZ2
16;C
17;        DT1=1./DTP
18;        DT2=1./DT
19;        DO 505 I=1,179
20;        YY(I)=0.
21;        DO 504 J=1,25
22;        AX(I,J)=0.
23;        NCOAX(I,J)=0
24;    504 CONTINUE
25;    505 CONTINUE
26;        DO 1000 KE=1,NET
27;        NTOP=0
28;        DO 25 I=1,3
29;     25 READ(35,69)NG(I),XO(I),ZO(I)
30;     69 FORMAT(1X,I3,2X,F8.4,2X,F8.4)
31;        IF(KE.GE.51.AND.KE.LE.54)NTOP=100
32;        IF(ID.NE.4.AND.KE.LE.34)GO TO 1000
33;        IF(ID.NE.4.AND.KE.GE.71)GO TO 1000
34;        IF(ID.NE.4.AND.NTOP.EQ.100)GO TO 1000
35;C       CALCULATING THE ELEMENT STIFFNESS MATRIX IN ONE ELEMENT
36;        XO(4)=XO(1)
37;        XO(5)=XO(2)
38;        ZO(4)=ZO(1)
39;        ZO(5)=ZO(2)
40;        NG(4)=(NG(1)+NG(2))/2
41;        NG(5)=(NG(2)+NG(3))/2
42;        NG(6)=(NG(1)+NG(3))/2
43;        AR=(ZO(2)*XO(3)+ZO(1)*XO(2)+XO(1)*ZO(3)-XO(1)*ZO(2)-XO(2)*ZO(3)
44;       0-XO(3)*ZO(1))/2.
45;        AR=ABS(AR)
46;        DO 10 I=1,5
47;        DO 10 J=1,5
48;        X(I,J)=XO(I)-XO(J)
49;     10 Z(I,J)=ZO(I)-ZO(J)
50;        DJACO=Z(1,3)*X(2,3)-Z(2,3)*X(1,3)
51;        IF(KE.LE.34.OR.KE.GE.71)GO TO 20
52;        IF(NTOP.EQ.100)GO TO 20
53;        RAB1=0.3333333*(RO(NF(NG(1)))+RO(NF(NG(2)))+RO(NF(NG(3))))
54;        RAB2=0.3333333*(AP(NF(NG(1)))+AP(NF(NG(2)))+AP(NF(NG(3))))
55;        RAB3=0.3333333*(BA(NF(NG(1)))+BA(NF(NG(2)))+BA(NF(NG(3))))
56;        RAB4=0.3333333*(CP(NF(NG(1)))+CP(NF(NG(2)))+CP(NF(NG(3))))
57;        RAB5=0.3333333*(UE(NF(NG(1)))+UE(NF(NG(2)))+UE(NF(NG(3))))
58;        RAB6=0.3333333*(PER(NF(NG(1)))+PER(NF(NG(2)))+PER(NF(NG(3))))
```

```
      STIFF                      3/ 1/79  15:56:56     PAGE  2
59;      RA812=0.3333333*(1./PER(NF(NG(1)))+1./PER(NF(NG(2)))+1./PER(NF(NG(3))))
60;   20 DO 15 I=1,6
61;      YNG=0.
62;      DO 17 J=1,6
63;      SI=0.
64;      SAX=0.
65;      SAY=0.
66;      SYY=0.
67;      SYYBORROW=0.
68;      DO 16 KW=1,7
69;      P=ZS(KW,1)
70;      Q=ZS(KW,2)
71;      R=ZS(KW,3)
72;      GO TO (1,2,3,4),ID
73;    1 SAX=SAX+WE(KW)*BA2(I)*BA2(J)  <========================>( I )
74;      SAY=SAY-WE(KW)*DXZ2(1,J)*BA2(I)  <====
75;      GO TO 16
76;    2 SAX=SAX+WE(KW)*BA2(I)*BA2(J)
77;      SAY=SAY-WE(KW)*DXZ2(-1,J)*BA2(I)
78;      SYY=SYY+WE(KW)*BA2(I)
79;      GO TO 16
80;    3 SAX=SAX+WE(KW)*BA2(J)*BA2(I)
81;      SAY=SAY-WE(KW)*(DXZ2(1,J)*DXZ2(1,I)+DXZ2(-1,J)*DXZ2(-1,I))
82;      SYYBORROW=SYYBORROW+WE(KW)*BA2(J)*BA2(I)
83;      GO TO 16
84;    4 IF(KE.LE.34.OR.KE.GE.71) GO TO 5
85;      IF(NTOP.EQ.100)GO TO 5
86;      SAX=SAX+WE(KW)*(FAI*RA81*RA84+(1.-FAI)*ROS*CS)*BA2(J)*BA2(I)
87;      UKNO=0.
88;      WKNO=0.
89;      PKNO=0.
90;      DO 12 KNO=1,6
91;      UKNO=UKNO+BA2(KNO)*U(NF(NG(KNO)))
92;      WKNO=WKNO+BA2(KNO)*W(NF(NG(KNO)))
93;   12 PKNO=PKNO+BA2(KNO)*(P2(NF(NG(KNO)))-P1(NF(NG(KNO))))*DT1
94;      SAY=SAY-WE(KW)*COND*(DXZ2(1,J)*DXZ2(1,I)+DXZ2(-1,J)*DXZ2(-1,I))
95;      SAY=SAY-WE(KW)*RA81*RA84*(UKNO*DXZ2(1,J)+WKNO*DXZ2(-1,J))*BA2(I)
96;      SAY=SAY+WE(KW)*RA82*(RA81*GRAY*WKNO-RA85*RA812*(WKNO**2+UKNO**2))*
97;     .BA2(J)*BA2(I)
98;      SAY=SAY+WE(KW)*RA82*PKNO*BA2(J)*BA2(I)
99;      GO TO 16
100;   5 SAX=SAX+WE(KW)*RORCK*CSRCK*BA2(J)*BA2(I)
101;     SAY=SAY-WE(KW)*CONRCK*(DXZ2(1,J)*DXZ2(1,I)+DXZ2(-1,J)*DXZ2(-1,I))
102;  16 CONTINUE
103;     GO TO (51,51,53,54),ID
104;  51 YNG=YNG+SAY*P2(NF(NG(J)))
105;     AXE(I,J)=SAX*DJACO*RA85
106;     GO TO 17
107;  53 SAXT=SAX*(-BFAI+FAI*RA83)*RA81*RA85*RA85
108;     SAYT=SAY*RA81*RA85*RA86
109;     SAX=DT1*ISCH*SAXT-SAYT
110;     SAY=DT1*ISCH*SAXT+(ISCH-1)*SAYT
111;     DTDT=(T2(NG(J))-T1(NG(J)))*DT2
112;     YNG=YNG+SAY*P2(NF(NG(J)))+FAI*RA81*RA82*RA85*RA85*SYYBORROW*DTDT
113;     AXE(I,J)=SAX*DJACO
114;     GO TO 17
115;  54 SAXT=SAX
116;     SAYT=SAY
```

```
       STIFF                      3/ 1/79  15:56:56     PAGE  3
117;       SAX=DT2*ISCH*SAXT-SAYT
118;       SAY=DT2*ISCH*SAXT+(ISCH-1)*SAYT
119;       YNG=YNG+SAY*T2(NG(J))
120;       AXE(I,J)=SAX*DJACO
121;   17 CONTINUE
122;       GO TO(61,62,63,64),ID
123;   61 YY(NF(NG(I)))=YY(NF(NG(I)))+YNG*RAB6*DJACO
124;       GO TO 15
125;   62 YY(NF(NG(I)))=YY(NF(NG(I)))+(YNG+SYY*RAB1*GRAY)*RAB6*DJACO
126;       GO TO 15
127;   63 YY(NF(NG(I)))=YY(NF(NG(I)))+YNG*DJACO
128;       GO TO 15
129;   64 YY(NG(I))=YY(NG(I))+YNG*DJACO
130;   15 CONTINUE
131;C     ---DELETE BOUNDARY NODES AND MODIFY FORCE FUNCTIONS---
132;       DO 38 I=1,6
133;       GO TO (31,32,33,34),ID
134;   31 IF(NU(NF(NG(I))).NE.555)GO TO 38
135;       YY(NF(NG(I)))=U(NF(NG(I)))
136;       GO TO 37
137;   32 IF(NV(NF(NG(I))).NE.555)GO TO 38
138;       YY(NF(NG(I)))=V(NF(NG(I)))
139;       GO TO 37
140;   33 IF(NP(NF(NG(I))).NE.555)GO TO 38
141;       YY(NF(NG(I)))=P2(NF(NG(I)))
142;       GO TO 37
143;   34 IF(NT(NG(I)).NE.555)GO TO 38
144;       YY(NG(I))=T2(NG(I))
145;   37 DO 36 J=1,6
146;       IF(ID.EQ.4)GO TO 35
147;       YY(NF(NG(J)))=YY(NF(NG(J)))-AXE(J,I)*YY(NF(NG(I)))
148;       GO TO 36
149;   35 YY(NG(J))=YY(NG(J))-AXE(J,I)*YY(NG(I))
150;   36 CONTINUE
151;       IF(ID.EQ.4)GO TO 39
152;       YY(NF(NG(I)))=1.234567E20
153;       GO TO 60
154;   39 YY(NG(I))=1.234567E20
155;   60 DO 65 J=1,6
156;       AXE(I,J)=0.
157;   65 AXE(J,I)=0.
158;   38 CONTINUE
159;       CALL COMPAC(ID) ─────────────────────────────(2)
160; 1000 CONTINUE
161;       REWIND 35
162;C     TO OBTAIN INFORMATION OF NO. OF EQNS.(NEQ), & MAX. ROW DIMENSIONS(NROW)-
163;       DO 70 I=1,179
164;       IF(NCOAX(I,1).EQ.0)GO TO 75
165;       NEQ=I
166;   70 CONTINUE
167;       GO TO 74
168;   75 NEQ=I-1
169;   74 NROW=0
170;       DO 77 I=1,NEQ
171;       KNO=0
172;       DO 79 J=1,18
173;       IF(NCOAX(I,J).EQ.0)GO TO 78
174;       KNO=KNO+1
```

```
      STIFF                     3/ 1/79  15:56:56     PAGE  4
175;  79 CONTINUE
176;  78 IF(KNO.GT.MROW)MROW=KNO
177;  77 CONTINUE
178;     I=0
179;  66 I=I+1
180;     IF(I.GE.223)GO TO 68
181;     IF(YY(I).NE.1.234567E20)GO TO 66
182;     DO 67 IP=I,222
183;  67 YY(IP)=YY(IP+1)
184;     I=I-1
185;     GO TO 66
186;     REWIND 32
187;  68 RETURN
188;     END
```

```
       9A2                      3/ 1/79  15:58:11     PAGE   1
 1;        FUNCTION 9A2(IBA)
 2;C        9A2=SHAPE FUNCTIONS INTERMS OF AREA COORDINATES(P=L1;Q=L2;R=L3)
 3;        COMMON SGW1(22),SGW2(12),SGW3
 4;        COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
 5;        COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
 6;        PQR(1)=P
 7;        PQR(2)=Q
 8;        PQR(3)=R
 9;        PQR(4)=P
10;        IF(IBA.GT.3)GO TO 20
11;        9A2=2.*PQR(IBA)*PQR(IBA)-PQR(IBA)
12;        GO TO 50
13;   20 KN=IBA-3
14;        9A2=4.*PQR(KN)*PQR(KN+1)
15;   50 RETURN
16;        END
```

```
      DXZ2                       3/ 1/79  15:58:31     PAGE  1
  1;      FUNCTION DXZ2(ND,N)
  2;C      DXZ2=DERIVATION OF THE SHAPE FUNCTION "N" WITH RESPECT TO
  3;C      "X" OR "Z" IN TERMS OF AREA COORDINATES
  4;C      ND=1 FOR D(N)/DX; ND=-1 FOR D(N)/DZ
  5;      COMMON SGW1(22),SGW2(12),SGW3
  6;      COMMON FAI,BFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAV,DTDZ
  7;      COMMON AR,ZN(7),ZS(7,3),WE(7),PQR(4),P,Q,R,NET,NOD,ZTO,XTO,IND,IGD,ZTA
  8;      COMMON NG(6),X0(6),Z0(6),X(5,5),Z(5,5)
  9;      IF(AR.LE.1.0E-10.OR.AR.GT.1.0E5)WRITE(25,10)AR
 10;   10 FORMAT(1X,"AR",E12.5)
 11;      PQR(1)=P
 12;      PQR(2)=Q
 13;      PQR(3)=R
 14;      PQR(4)=P
 15;      IF(ND.EQ.-1)GO TO 10
 16;      IF(N.LE.3)DXZ2=(4.*PQR(N)-1.)*Z(N+2,N+1)/(2.*AR)
 17;      KN=N-3
 18;      IF(N.GT.3)DXZ2=2.*(PQR(KN+1)*Z(KN+2,KN+1)+PQR(KN)*Z(KN,KN+2))/AR
 19;      GO TO 50
 20;   10 IF(N.LE.3)DXZ2=(4.*PQR(N)-1.)*X(N+1,N+2)/(2.*AR)
 21;      KN=N-3
 22;      IF(N.GT.3)DXZ2=2.*(PQR(KN+1)*X(KN+1,KN+2)+PQR(KN)*X(KN+2,KN))/AR
 23;   50 CONTINUE
 24;      RETURN
 25;      END
```

```
      COMPAC                     3/ 1/79  15:58:52     PAGE  1
  1;      SUBROUTINE COMPAC(ID)
  2;C     INSERT AE(6,6) IN TO A(IM,17)
  3;C     "NCOL"=COLUMN INDEX OF "A"
  4;C     "NG"=GLOBAL NODE INDEX
  5;      COMMON SGW1(22),SGW2(12),SGW3
  6;      COMMON FAI,OFAI,ROS,CS,COND,RORCK,CSRCK,CONRCK,PERS,DT,DTP,TS,GRAY,DTDZ
  7;      COMMON AR,ZW(7),ZS(7,3),WE(7),PQR(4),P,Q,R,WET,NOD,ZTO,XTO,IMD,IGD,ZTA
  8;      COMMON NG(6),XG(6),ZO(6),X(5,5),Z(5,5),NF(223)
  9;      COMMON NT(223),NU(77),NW(77),NP(77)
 10;      COMMON U(77),W(77),T1(223),T2(223),P1(77),P2(77)
 11;      COMMON RO(77),AP(77),BA(77),CP(77),UE(77),PER(77)
 12;      COMMON AE(6,6),A(179,25),YY(223),NCOL(179,25)
 13;      SDF=0.
 14;      DO 70 I=1,6
 15;      DO 70 J=1,6
 16;      IF(ABS(AE(I,J)).GT.SDF)SDF=ABS(AE(I,J))
 17;   70 CONTINUE
 18;      DO 71 I=1,6
 19;      DO 71 J=1,6
 20;      S=AE(I,J)/SDF
 21;      IF(ABS(S).LE.3.0E-6)AE(I,J)=0.
 22;   71 CONTINUE
 23;      DO 10 K=1,6
 24;      IM=NF(NG(K))
 25;      IF(ID.NE.4)GO TO 30
 26;      IM=NG(K)
 27;   30 KIS=0
 28;      DO 50 IS=1,IM
 29;      GO TO (51,52,53,54,53),ID
 30;   51 IF(NU(IS).NE.555)GO TO 50
 31;      GO TO 56
 32;   52 IF(NW(IS).NE.555)GO TO 50
 33;      GO TO 56
 34;   53 IF(NP(IS).NE.555)GO TO 50
 35;      GO TO 56
 36;   54 IF(NT(IS).NE.555)GO TO 50
 37;   56 KIS=KIS+1
 38;   50 CONTINUE
 39;      IM=IM-KIS
 40;      IF(IM.LE.0)GO TO 10
 41;      DO 11 L=1,6
 42;      IF(ABS(AE(K,L)).LE.5.0E-20)GO TO 11
 43;      JM=NF(NG(L))
 44;      IF(ID.NE.4)GO TO 60
 45;      JM=NG(L)
 46;   60 KIS=0
 47;      DO 55 IS=1,JM
 48;      GO TO(61,62,63,64,63),ID
 49;   61 IF(NU(IS).NE.555)GO TO 55
 50;      GO TO 66
 51;   62 IF(NW(IS).NE.555)GO TO 55
 52;      GO TO 66
 53;   63 IF(NP(IS).NE.555)GO TO 55
 54;      GO TO 66
 55;   64 IF(NT(IS).NE.555)GO TO 55
 56;   66 KIS=KIS+1
 57;   55 CONTINUE
 58;      JM=JM-KIS
```

```
       COMPAC                    3/ 1/79  15:58:52     PAGE  2
59;        IF(JM.LE.0)GO TO 11
60;        DO 16 JC=1,25
61;        IF(NCOL(IM,JC).EQ.0)GO TO 17
62;        IF(JM.LT.NCOL(IM,JC))GO TO 18
63;        IF(JM.NE.NCOL(IM,JC))GO TO 16
64;        A(IM,JC)=A(IM,JC)+AE(K,L)
65;        GO TO 11
66;     16 CONTINUE
67;        JC=JC-1
68;     17 A(IM,JC)=AE(K,L)
69;        NCOL(IM,JC)=JM
70;        GO TO 11
71;     18 LA=NCOL(IM,JC)
72;        ALA=A(IM,JC)
73;        DO 19 ML=JC,24
74;        NX=NCOL(IM,ML+1)
75;        ANX=A(IM,ML+1)
76;        NCOL(IM,ML+1)=LA
77;        A(IM,ML+1)=ALA
78;        IF(NX.EQ.0)GO TO 20
79;        LA=NX
80;     19 ALA=ANX
81;     20 A(IM,JC)=AE(K,L)
82;        NCOL(IM,JC)=JM
83;     11 CONTINUE
84;     10 CONTINUE
85;        RETURN
86;        END
```