Lawrence Radiation Laboratory
UNIVERSITY OF CALIFORNIA
LIVERMORE

UCRL-50820

THE OGRE CODE
A TWO-DIMENSIONAL NUMERICAL MODEL OF THE
TRANSIENT FLOW OF ONE OR TWO COMPRESSIBLE FLUIDS
THROUGH CONFINED POROUS MEDIA

John A. Korver

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in this notice, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

P5909

Contents

Abstract	1
Introduction	1
Theory	2
Governing Equations	2
Fluid Viscosity, Compressibility and Solubility	3
Effective Permeability	4
Capillary Pressure	5
Saturation	5
Critical Saturation	6
Process Equations	6
Boundary Conditions	7
Impermeable Boundaries	7
Permeable Boundaries	8
Initial Conditions	8
Pressure and Saturation Distributions	8
Computational Method	9
Fluid Representation	9
Outline of Computational Scheme	9
Modified Process Equations	14
One-Phase or Two-Phase Option	17
Finite Difference Approximations	17
Time Increments	23
Pressure Equation—Matrix Solutions	24
Mass Conservation Checks	27
The Computer Program	28
Program Organization	28
Input	29
Output	30
Error Exit Routines	30
References	31
Additional References	32
Appendix A—Definition of Symbols	33
Appendix B—Code Symbols	36
Appendix C—Input Card Format	45
Appendix D—Data Output Format	47
Appendix E—Fortran Code Listing	64
Appendix F—Code Flow Diagrams	126

THE OGRE CODE A TWO-DIMENSIONAL NUMERICAL MODEL OF THE TRANSIENT FLOW OF ONE OR TWO COMPRESSIBLE FLUIDS THROUGH CONFINED POROUS MEDIA

Abstract

The OGRE code was originally developed to evaluate the effects of nuclear stimulation of oil and gas reservoirs. As a result, the mathematical model follows closely some of the two-dimensional models used by the petroleum industry. The nomenclature also follows that used by the industry but, since the processes being analyzed are the same as those which take place in the viscous flow of any fluid through porous media, the model can be used for a more general class of problems.

Analysis is carried out by the solution of finite difference approximations to the differential equations describing the flow of one or two compressible fluids in the vertical plane. The medium may be homogeneous or heterogeneous and, in the heterogeneous case, several standard geometries or a random distribution of permeability and porosity may be used.

Like many working codes, OGRE is in a continuing process of modification, and thus the presentation here represents its current form.

Introduction

The OGRE code was originally developed to evaluate the effects of nuclear explosions on the producing characteristics of oil or gas reservoirs. As a result, the mathematical model follows closely some of the two-dimensional models used by the petroleum industry.

However, there is an important difference. Whereas the petroleum industry ordinarily evaluates the performance of entire reservoirs involving many wells, the OGRE code examines the performance of a single well. In the former case, the analysis is carried out in the horizontal or $x-y$ plane, and in the latter case, in the vertical or $x-z$ plane.

Since analysis in the vertical plane allows for examining the effects of the gravity potential and, in the case of two-phase flow the effects of capillarity, the OGRE code can be used to examine a more general class of problems. In addition, the code has been designed to investigate either the flow of a single compressible fluid or the simultaneous flow of two compressible fluids, one a wetting phase and the other a nonwetting

phase. Finally, flow can take place linearly, cylindrically or spherically, and the gravity vector can operate in any direction in the vertical plane.

Since the mathematical model was originally developed in terms of oil and gas as the two compressible phases, the derivations of the governing equations will be given in these terms. In this case, oil is the wetting phase and gas is the nonwetting phase. The same mathematics would apply if, for example, water were the wetting phase and air were the nonwetting phase.

Theory

The process being studied is the simultaneous isothermal flow of two compressible phases (oil and gas) in a confined reservoir under the influence of a pressure gradient and the force of gravity. The effect of the pressure gradient is to drive the oil in the direction of the pressure decline and to reduce the solution gas content in the same direction.

Definitions of the symbols used in the following treatment are given in Appendix A.

GOVERNING EQUATIONS

The mass of oil (water) contained in a unit volume of reservoir rock is

$$\frac{\phi \gamma_o G \sigma_o}{\beta_o}.$$

The mass of free gas (air) in this unit volume is

$$\phi \gamma_g \sigma_g G \beta_g.$$

The mass of gas (air) dissolved in the oil (water) is

$$\frac{\phi R_s \gamma_g G \sigma_o}{\beta_o}.$$

The total mass is then

$$G \phi \left(\frac{\gamma_o \sigma_o}{\beta_o} + \frac{R_s \gamma_g \sigma_o}{\beta_o} + \gamma_g \sigma_g \beta_g \right).$$

It is assumed that there are no phase changes as a function of pressure, i. e., there are fixed amounts of oil and gas in the system for all time.

The continuity equations for the oil (water) and gas (air) are

$$\nabla \cdot \left(\frac{\gamma_o G}{\beta_o} v_o \right) = - \phi \frac{\partial}{\partial t} \left(\frac{\gamma_o G}{\beta_o} \sigma_o \right) \quad (1)$$

for the oil (water), and

$$\nabla \cdot \left(\gamma_g G \beta_g v_g + R_s \frac{\gamma_g G}{\beta_o} v_o \right) = - \phi \frac{\partial}{\partial t} \left(\gamma_g G \beta_g \sigma_g + R_s \frac{\gamma_g G}{\beta_o} \sigma_o \right) \quad (2)$$

for the gas (air), both free and that dissolved in the oil (water).

The equations of motion are

$$v_o = - \frac{k_o}{\mu_o} (\nabla p_o + \nabla \gamma_o G) \quad (3)$$

for the oil (water), including the dissolved gas (air), and

$$v_g = - \frac{k_g}{\mu_g} (\nabla p_g + \nabla \gamma_g G) \quad (4)$$

for the free gas (air).

It is to be understood here that

$$k_o = k_o(\sigma_o), \quad \mu_o = \mu_o(p_o),$$

$$k_g = k_g(\sigma_g), \quad \mu_g = \mu_g(p_g).$$

The use of these equations constitutes a working hypothesis that Darcy's Law for viscous flow can be applied to the separate phases of multiphase flow. The hypothesis seems to be sufficiently well borne out by experiment.

By using cylindrical coordinates in two dimensions and writing Eqs. (3) and (4) in terms of the r and z components, we obtain

$$v_{or} = - \frac{k_o}{\mu_o} \left[\frac{\partial p_o}{\partial r} + \left(\frac{\gamma_o}{\beta_o} + R_s \frac{\gamma_g}{\beta_o} \right) G \sin \alpha \right], \quad (5)$$

$$v_{oz} = - \frac{k_o}{\mu_o} \left[\frac{\partial p_o}{\partial z} + \left(\frac{\gamma_o}{\beta_o} + R_s \frac{\gamma_g}{\beta_o} \right) G \cos \alpha \right], \quad (6)$$

$$v_{gr} = - \frac{k_g}{\mu_g} \left(\frac{\partial p_g}{\partial r} + \gamma_g G \beta_g \sin \alpha \right), \quad (7)$$

$$v_{gz} = - \frac{k_g}{\mu_g} \left(\frac{\partial p_g}{\partial z} + \gamma_g G \beta_g \cos \alpha \right). \quad (8)$$

FLUID VISCOSITY, COMPRESSIBILITY AND SOLUBILITY

The P-V-T relationships for the gas and oil are contained in the compressibility factors β_o and β_g and in the gas solubility factor R_s . These relationships, as well as the mathematics throughout the remainder of this report, apply also to air and water.

The viscosities, compressibilities and solubilities of the fluids are considered to be functions of the pressure of each fluid. These functions will be unique for each problem, and often cannot be expressed in an explicit analytical form. To solve this difficulty, numerical models often are based on tables of laboratory data, and interpolation is used to obtain the necessary numerical values.

During the development of the OGRE code, however, it was decided to express these values as linear functions of the pressure, recognizing that, for specific problems of interest, reference to appropriate tables would be more accurate. In the present form of the program linear functions are still used, but tables can be easily substituted. The functions presently being used are:

$$\mu_o = \frac{\partial \mu_o}{\partial p_o} p_o + C_{vo} \quad (9)$$

$$\mu_g = \frac{\partial \mu_g}{\partial p_g} p_g + C_{vg} \quad (10)$$

$$\beta_o = \frac{\partial \beta_o}{\partial p_o} p_o + C_{Bo} \quad (11)$$

$$\beta_g = \frac{\partial \beta_g}{\partial p_g} p_g + C_{Bg} \quad (12)$$

$$R_s = \frac{\partial R_s}{\partial p_o} p_o + C_{Rs} \quad (13)$$

where the slopes $\partial \mu_o / \partial p_o$, $\partial \mu_g / \partial p_g$, $\partial R_o / \partial p_o$, $\partial \beta_g / \partial p_g$, $\partial R_s / \partial p_o$ and the intercepts C_{vo} , C_{vg} , C_{Bo} , C_{Bg} , C_{Rs} are constant input values.

EFFECTIVE PERMEABILITY

The subscripted permeabilities k_o and k_g in Eqs. (5) through (8) are effective permeabilities for the oil and gas. In two-phase flow, the permeability to each phase is less than the absolute permeability which would obtain if only one phase completely occupied the pore spaces. In fact, $k_o + k_g < k_a$ where k_a is the absolute permeability. The effective permeability to each phase has been found to be primarily a function of the saturation of that phase.

The rigorously developed analytic expression has not been derived to date. Wyckoff and Botset¹ published experimental data showing permeability of the wetting phase as a function of saturation. Cardwell and Parsons² claimed to have fit the data with $k_w = k_a \sigma_w^{3.5}$, and $k_w = k_a \sigma_w^2$ respectively. Here k_w is the wetting phase permeability, k_a is the absolute permeability, and σ_w is the wetting phase saturation.

The discrepancy in the exponent raises some doubts as to the correctness of either formula. Moreover, both of these expressions were developed from gravity drainage experiments in columns, and no attempt was made to determine the permeability of the nonwetting phase.

More recently, Pirson,³ reporting on work done by Archie and others,⁴ presents expressions for both the wetting and nonwetting phases in oil- or water-wet systems. These expressions are derived from laboratory experiments on the electrical resistivity of saturated cores. These expressions are:

$$k_o = \left(\frac{\sigma_o - \sigma_{oi}}{1 - \sigma_{oi}} \right)^{1/2} \sigma_o^3 k_a \quad (14)$$

for the wetting phase, and

$$k_g = \left(1 - \frac{\sigma_o - \sigma_{oi}}{1 - \sigma_{oi}} \right) \left[1 - \left(\frac{\sigma_o - \sigma_{oi}}{1 - \sigma_{oi}} \right)^{1/4} \sigma_o^{1/2} \right]^2 k_a \quad (15)$$

for the nonwetting phase. The expression for k_g , incidentally, is that used when the nonwetting phase is displacing the wetting phase, the process being called drainage. The permeability-saturation curve for the nonwetting phase shows a hysteresis, the curve being different depending on whether the process is drainage or imbibition.

CAPILLARY PRESSURE

Equations (5) through (8) contain both oil and gas phase pressures. These may be related by the equation

$$p_c = p_g - p_o \quad (16)$$

If, then, an expression can be found for p_c in other terms, either p_o or p_g can be eliminated from the equations. Rose⁵ has developed a relationship, based on the early work of Kozeny⁶ and Carman,⁷ which can be rearranged to give

$$p_c = \psi \left(\frac{\sigma_o \phi}{k_c k_o} \right)^{1/2} \quad (17)$$

where k_c is the Kozeny constant (≈ 5) and ψ is the surface tension.

SATURATION

The saturations of the oil and gas phases are related by the equation

$$\sigma_o + \sigma_g = 1.0 \quad (18)$$

or, if connate water is included,

$$\sigma_o + \sigma_g + \sigma_w = 1.0. \quad (19)$$

CRITICAL SATURATION

The published data on the effective permeabilities of oil and gas as a function of their saturations show that the permeability of each phase goes to zero below a certain saturation of that phase.

Another implication here relates to the concept of a free surface. If the capillary pressure is zero, the interfacial curvature must also be zero. This means the pressure is in equilibrium, with a flat interface, or free surface. However, for any given porous medium, there is a lower limit to the amount of the interfacial curvature. For this reason, then, there is no interface in the system across which the pressure differential is zero.

The hypothetical free surface, however, may be determined as the distance below the interface at which the capillary pressure goes to zero.

PROCESS EQUATIONS

By using (16) and (18) to eliminate p_g and σ_g from (2) and (4), we are left with four independent equations, (1), (2), (3), and (4); and four dependent variables, p_o , σ_o , v_o and v_g .

Solving (1) and (2) for the time derivatives of the oil saturation, in cylindrical coordinates, gives

$$\frac{\partial \sigma_o}{\partial t} = \frac{1}{\beta_o} \frac{\partial \beta_o}{\partial p_o} \left(\sigma_o \frac{\partial p_o}{\partial t} + \frac{v_{or}}{\phi} \frac{\partial p_o}{\partial r} + \frac{v_{oz}}{\phi} \frac{\partial p_o}{\partial z} \right) - \frac{1}{\phi} \left(\frac{v_{or}}{r} + \frac{\partial v_{or}}{\partial r} + \frac{\partial v_{oz}}{\partial z} \right) \quad (20)$$

and

$$\begin{aligned} \frac{\partial \sigma_o}{\partial t} = & - \left(\frac{\beta_g}{1 - \beta_o - \beta_g} \right) \left\{ \frac{1}{\phi} \left[\left(\frac{v_{gr}}{r} + \frac{\partial v_{gr}}{\partial r} + \frac{\partial v_{gz}}{\partial z} \right) \beta_g + \left(\frac{v_{or}}{r} + \frac{\partial v_{or}}{\partial r} + \frac{\partial v_{oz}}{\partial z} \right) \frac{R_s}{\beta_o} \right. \right. \\ & + \left(v_{gr} \left(\frac{\partial p_o}{\partial t} + \frac{\partial p_c}{\partial t} \right) + v_{gz} \left(\frac{\partial p_o}{\partial t} + \frac{\partial p_c}{\partial t} \right) \right) \frac{\partial \beta_g}{\partial p_g} \\ & + \left(v_{or} \frac{\partial p_o}{\partial r} + v_{oz} \frac{\partial p_o}{\partial z} \right) \left(\frac{1}{\beta_o} \frac{\partial R_s}{\partial p_o} - \frac{R_s}{\beta_o^2} \frac{\partial \beta_o}{\partial p_o} \right) \\ & \left. \left. - \left[(1 - \sigma_o) \frac{\partial \beta_g}{\partial p_g} \left(\frac{\partial p_o}{\partial t} + \frac{\partial p_c}{\partial t} \right) + \left(\frac{\sigma_o}{\beta_o} \frac{\partial R_s}{\partial p_o} - \sigma_o \frac{R_s}{\beta_o^2} \frac{\partial \beta_o}{\partial p_o} \right) \frac{\partial p_o}{\partial t} \right] \right\} \quad (21) \end{aligned}$$

Equating (20) and (21) and solving for the time derivative of the oil pressure results in

$$\begin{aligned} \frac{\partial p_o}{\partial t} = & - \frac{1}{c\phi} \left\{ \eta \left(\frac{v_{or}}{r} + \frac{\partial v_{or}}{\partial r} + \frac{\partial v_{oz}}{\partial z} \right) - \zeta \left(\frac{v_{gr}}{r} + \frac{\partial v_{gr}}{\partial r} + \frac{\partial v_{gz}}{\partial z} \right) \right. \\ & - \tau \left(v_{or} \frac{\partial p_o}{\partial r} + v_{oz} \frac{\partial p_o}{\partial z} \right) \\ & \left. - \theta \left[v_{gr} \left(\frac{\partial p_o}{\partial r} + \frac{\partial p_c}{\partial r} \right) + v_{gz} \left(\frac{\partial p_o}{\partial z} + \frac{\partial p_c}{\partial z} \right) + (1 - \sigma_o) \phi \frac{\partial p_c}{\partial t} \right] \right\} \end{aligned} \quad (22)$$

where

$$\begin{aligned} \eta &= 1 - \frac{R_s}{R_s - \beta_o \beta_g}, & \theta &= \frac{\zeta}{\beta_g} \frac{\partial \beta_g}{\partial p_g}, \\ \zeta &= \frac{\beta_o \beta_g}{R_s - \beta_o \beta_g}, & \tau &= \eta \frac{1}{\beta_o} \frac{\partial \beta_o}{\partial p_o} + \frac{1}{R_s - \beta_o \beta_g} \frac{\partial R_s}{\partial p_o}, \\ c &= \sigma_o \tau + (1 - \sigma_o) \theta. \end{aligned}$$

Equations (20) and (22), together with (5), (6), (7), (8), (14), (15), (16), (17), and (18), plus suitable boundary and initial conditions, constitute a system which is amenable to solution by finite-difference methods.

Boundary Conditions

IMPERMEABLE BOUNDARIES

Along the impervious walls of the top and bottom of the reservoir, on the surface defined by the outer limit of the drainage radius, and along the nonproducing segments of the well bore, the above equations are modified to meet the following conditions:

- No mass transport occurs across these surfaces; i. e., the normal components of the velocity vanish.
- The surfaces are free slip surfaces; i. e., the tangential components of the velocity are unrestricted.

On the horizontal boundaries, then, (6) and (8) become

$$v_{oz} = 0, \quad (23)$$

$$v_{gz} = 0. \quad (24)$$

On the vertical boundaries, (5) and (7) become

$$v_{or} = 0, \quad (25)$$

$$v_{gr} = 0. \quad (26)$$

At impermeable boundary intersections, both components of the velocity are normal to one or the other boundary so that both components vanish. Equations (3) and (4) become

$$v_o = 0, \quad (27)$$

$$v_g = 0. \quad (28)$$

With these restrictions on the boundary velocities, no modifications are necessary for Eqs. (20) and (22).

PERMEABLE BOUNDARIES

Along the producing segment(s) of the well bore, the oil pressure is either constant or is a function of some chosen flow rate. In the first case

$$P_o \text{ (well bore)} = \text{constant}, \quad (29)$$

or in the second case

$$P_o \text{ (well bore)} = f(Q_o)$$

where Q_o is the oil production rate.

Initial Conditions

PRESSURE AND SATURATION DISTRIBUTIONS

Initially, the system is in hydrostatic equilibrium; that is, the velocities are everywhere equal to zero.

The initial conditions are generated in the code by using the input value of the bottom hole oil pressure (BHP), setting Eq. (6) equal to zero and solving for the oil pressure at each succeeding level; thus

$$-\frac{k_o}{\mu_o} \left[\frac{\partial p_o}{\partial z} + (\gamma_o + R_s \gamma_g) \frac{G}{\beta_o} \cos \alpha \right] = 0$$

or

$$\frac{\partial p_o}{\partial z} = -(\gamma_o + R_s \gamma_g) \frac{G}{\beta_o} \cos \alpha. \quad (30)$$

Since R_s and β_o are both functions of p_o , an iterative scheme is used to predict and correct the pressure at each level. If the problem contains two phases, initially the saturation of each phase must be a function of the solubility of the gas (nonwetting phase) in the oil (wetting phase). The solubility is, in turn, a function of the oil pressure. Because of the difficulty in determining this function, most problems are initialized at pressures which assure that only one phase exists (any gas present is dissolved in the oil).

The following expression has been used for determining the initial saturation distribution in a two-phase system, but its validity has only been empirically demonstrated.

$$\frac{\partial \sigma_o}{\partial z} = \frac{\partial R_s}{\partial p_o} \frac{\partial p_o}{\partial z} / \phi \beta_g \quad (31)$$

Computational Method

FLUID REPRESENTATION

Since the numerical solutions are obtained by finite difference approximations to the differential equations, a Eulerian rectilinear grid of cells in two dimensions is used to represent the region of interest. Figure 1 illustrates the grid system and shows the points of definition, relative to the cells, of the field-variable quantities describing the flow regime. The pressures (p_o) and the saturations (σ_o) are computed at the points shown in Fig. 1, and then interpolated or extrapolated to the intersections of the cell boundaries and the secondary grid. The result is shown in Fig. 2. Pressures and saturations are thus defined at all intersections of the two grids. Those variables which are functions of the pressure or the saturation or both can then be explicitly solved for at each point. Specifically, μ_o , μ_g , k_o , k_g , β_o , β_g , R_s , p_c , and p_g are solved for at each point.

Figure 3 shows the complete computational grid. It may be noted from this figure that the saturations are computed only at interior grid points. Thus there is no need for boundary equations for the saturation.

The top, bottom and right sides of the grid are impermeable boundaries. The left side of the grid represents the radius of the well bore. Each cell along this side may be permeable or impermeable, depending on the type of problem being investigated.

OUTLINE OF THE COMPUTATIONAL SCHEME

The solutions to the finite-difference approximations are obtained by advancing the problem in time in a stepwise manner. The solutions at the end of each time cycle become the initial conditions for the next cycle.

Fig. 1. OGRE grid system—positions of computed field variables.

Each cycle consists of the following operations:

1. Step forward in time ($t^{n+3} = t^{n+0} + 3\Delta t^{n+3}$).
2. Compute new oil pressures along the well bore ($P_{0,1,k}^{n+3}$). These pressures may be fixed, or may be a function of time or a specified flux rate.
3. Compute new formation factors, solubilities and fluid viscosities throughout the system ($\beta_{o,j,k}^{n+3}$, $\beta_{g,j,k}^{n+3}$, $P_{s,j,k}^{n+3}$, $\mu_{o,j,k}^{n+3}$, $\mu_{g,j,k}^{n+3}$).
4. Compute flux rates along well bore and cumulative fluxes into or out of the system (Q_{ow}^{n+3} , Q_{ogw}^{n+3} , Q_{gw}^{n+3} , Q_{ocum}^{n+3} , Q_{gcum}^{n+3}).

- $\triangle$ Computed pressures, interpolated saturations
- $\triangle$ Computed saturations, interpolated pressures
- $\triangle$ Interpolated pressures, interpolated saturations

Fig. 2. OGRE grid system—positions of computed and interpolated field variables.

5. Compute current mass in the system. Compare change in mass with cumulative flux into or out of the system.
6. Compute new oil pressures in the horizontal (u) direction $(p_{o,j,k}^{n+3})$.
7. Linearly interpolate new oil pressures to all grid intersections.
8. Quadratically interpolate new oil pressures to the grid intersections in the first two columns of cells.
9. Step backward in time ($t^{n+2} = t^{n+3} - \Delta t^{n+3}$).
10. Time-interpolate new oil pressures $[p_{o,j,k}^{n+2} = (p_{o,j,k}^{n+3} + p_{o,j,k}^{n+1})/2]$.

Fig. 3. OGRE grid system showing permeable and impermeable boundaries.

11. Compute formation factors, solubilities and fluid viscosities for $t = t^{n+2}$ ($\beta_{o,j,k}^{n+2}$, $\beta_{g,j,k}^{n+2}$, $R_{s,j,k}^{n+2}$, $\mu_{o,j,k}^{n+2}$, $\mu_{g,j,k}^{n+2}$).
12. Compute fluid velocities and their space derivatives.
13. Compute new oil saturations ($\sigma_{o,j,k}^{n+2}$).
14. Linearly interpolate and extrapolate new saturations to all grid intersections.
15. Quadratically extrapolate new saturations to the well bore grid points.
16. Compute new permeabilities ($k_{o,j,k}^{n+2}$, $k_{g,j,k}^{n+2}$).
17. Compute new capillary pressures ($p_{c,j,k}^{n+2}$).
18. Compute new gas pressures ($p_{g,j,k}^{n+2}$).
19. Step forward in time ($t^{n+5} = t^{n+2} + 3\Delta t^{n+3}$).
20. Recompute new formation factors, solubilities and fluid viscosities ($\beta_{o,j,k}^{n+5}$, $\beta_{g,j,k}^{n+5}$, $R_{s,j,k}^{n+5}$, $\mu_{o,j,k}^{n+5}$, $\mu_{g,j,k}^{n+5}$).

21. Compute new oil pressures in vertical (z) direction $p_{0,j,k}^{n+5}$.
22. After repeating steps 7 through 18, the cycle is completed.

It can be seen that the solutions to the saturation equation are centered in time between two known solutions to the pressure equation. The reason for this is that the saturation equation is critically dependent on the pressures, while the pressure equation is only mildly dependent on the saturations. Figure 4 illustrates the time/cycle sequencing presently used, along with another sequencing which is being considered.

$P_{0_i} \sigma_{0_i}$ - Initial conditions

$P_{0(u)}$ - u-direction sweep

$P_{0(z)}$ - z-direction sweep

Fig. 4. OGRE computational sequencing.

MODIFIED PROCESS EQUATIONS

Before forming the finite difference approximations, which are used to obtain the numerical solutions, the differential equations are modified in the following manner:

- An implicit, or backward, difference method is used to solve the pressure equation (22), since this method is computationally more stable than explicit methods, and it allows for larger time steps. To take full advantage of this method, the velocities in Eq. (22) are expressed as functions of the pressure. The saturation equation (20) is solved explicitly; therefore, the velocities and their space derivatives are not expressed as functions of the pressure. Moreover, these velocities are computed explicitly only at the points where the saturations are computed (see Fig. 1).
- Distances in the horizontal direction are expressed in terms of a new function, u , as follows:

$$u = \begin{cases} r/r_w & \text{(Linear horizontal scaling)} \\ \ln(r/r_w) & \text{(Logarithmic horizontal scaling).} \end{cases} \quad (32)$$

- Constant multipliers are added to appropriate terms in both Eqs. (20) and (22). These constants are used to specify the type of flow (linear, cylindrical or spherical) and the number of dimensions (1 or 2).
- The saturation equation (20) is transformed into an exponential function to reduce the generation of computational noise.

The resulting new equations in u - z space for the pressure equation are:

$$\begin{aligned} & \left[c - \phi (1 - \sigma_o) \right] \frac{\partial p_o}{\partial t} \\ &= \zeta \left(\rho \left[\delta \epsilon E_{16} \left[\epsilon \left(\frac{\partial p_o}{\partial u} + \frac{\partial p_c}{\partial u} \right) + E_5 \right] + E_{16} \left[\epsilon \left(\frac{\partial^2 p_o}{\partial u^2} - \frac{\partial p_o}{\partial u} + \frac{\partial^2 p_c}{\partial u^2} - \frac{\partial p_c}{\partial u} \right) + E_9 \right] \right. \right. \\ & \quad \left. \left. + \left[\epsilon \left(\frac{\partial p_o}{\partial u} + \frac{\partial p_c}{\partial u} \right) + E_5 \right] E_1 \right] \right. \\ & \quad \left. + \lambda \left[E_{16} \left(\frac{\partial^2 p_o}{\partial z^2} + \frac{\partial^2 p_c}{\partial z^2} + E_{10} \right) + \left(\frac{\partial p_o}{\partial z} + \frac{\partial p_c}{\partial z} + E_6 \right) E_3 \right] \right) \\ & - \eta \left(\rho \left[\delta \epsilon E_{15} \left(\frac{\partial p_o}{\partial u} + E_7 \right) + E_{15} \left[\epsilon \left(\frac{\partial^2 p_o}{\partial u^2} - \frac{\partial p_o}{\partial u} \right) - E_{11} \frac{\partial p_o}{\partial u} + E_{16} \frac{\partial p_c}{\partial u} \right] \right. \right. \\ & \quad \left. \left. + \left(\epsilon \frac{\partial p_o}{\partial u} + E_7 \right) E_2 \right] \right) \end{aligned}$$

$$\begin{aligned}
& + \lambda \left[E_{15} \left(\frac{\partial^2 p_o}{\partial z^2} - E_{13} \frac{\partial p_o}{\partial z} + E_{14} \frac{\partial p_o}{\partial z} \right) + \left(\frac{\partial p_o}{\partial z} + E_6 \right) E_4 \right] \\
& + \theta \left(\rho \left[E_{16} \left(\epsilon \left(\frac{\partial p_o}{\partial u} + \frac{\partial p_c}{\partial u} \right) \right) \left(\epsilon \left(\frac{\partial p_o}{\partial u} + \frac{\partial p_c}{\partial u} \right) + E_5 \right) \right] \right. \\
& \quad \left. + \lambda \left[E_{16} \left(\frac{\partial p_o}{\partial z} + \frac{\partial p_c}{\partial z} \right) \left(\frac{\partial p_o}{\partial z} + \frac{\partial p_c}{\partial z} + E_6 \right) \right] + (1 - \sigma_o) \phi \frac{\partial p_c}{\partial t} \right) \\
& + \tau \left(\rho \left[E_{15} \epsilon \frac{\partial p_o}{\partial u} \left(\epsilon \frac{\partial p_o}{\partial u} + E_7 \right) \right] \right. \\
& \quad \left. + \lambda \left[E_{15} \frac{\partial p_o}{\partial z} \left(\frac{\partial p_o}{\partial z} + E_8 \right) \right] \right) \tag{32}
\end{aligned}$$

where ξ , η , θ , τ , and c are as defined before, ρ and λ are either 0 or 1, depending on whether or not the horizontal or vertical space derivatives are operable, and

$$\delta = \begin{cases} 0 & \text{(linear flow)} \\ 1 & \text{(cylindrical flow)} \\ 2 & \text{(spherical flow)}, \end{cases}$$

$$\epsilon = \begin{cases} \frac{1}{r_w} & \text{(linear flow, linear horizontal scaling)} \\ \frac{1}{ur_w} & \text{(cylindrical flow, linear horizontal scaling)} \\ \frac{1}{r_w e^u} & \text{(cylindrical or spherical flow, logarithmic horizontal scaling)}, \end{cases}$$

$$E_1 = \epsilon \left(\frac{1}{\mu_g} \frac{\partial k_g}{\partial u} - \frac{k_g}{\mu_g} \frac{\partial \mu_g}{\partial u} \right), \tag{34}$$

$$E_2 = \epsilon \left(\frac{1}{\mu_o} \frac{\partial k_o}{\partial u} - \frac{k_o}{\mu_o} \frac{\partial \mu_o}{\partial u} \right), \tag{35}$$

$$E_3 = \left(\frac{1}{\mu_g} \frac{\partial k_g}{\partial z} - \frac{k_g}{\mu_g} \frac{\partial \mu_g}{\partial z} \right), \tag{36}$$

$$E_4 = \left(\frac{1}{\mu_o} \frac{\partial k_o}{\partial z} - \frac{k_o}{\mu_o} \frac{\partial \mu_o}{\partial z} \right), \tag{37}$$

$$E_5 = \gamma_g G \beta_g \sin \alpha, \tag{38}$$

$$E_6 = \gamma_g G \beta_g \cos \alpha, \tag{39}$$

$$E_7 = (\gamma_o + R_s \gamma_g) \frac{G}{\beta_o} \sin \alpha, \quad (40)$$

$$E_8 = (\gamma_o + R_s \gamma_g) \frac{G}{\beta_o} \cos \alpha, \quad (41)$$

$$E_9 = \epsilon \gamma_g G \sin \alpha \frac{\partial \beta_g}{\partial p_g} \frac{\partial p_g}{\partial u}, \quad (42)$$

$$E_{10} = \gamma_g G \cos \alpha \frac{\partial \beta_g}{\partial p_g} \frac{\partial p_g}{\partial z}, \quad (43)$$

$$E_{11} = \epsilon (\gamma_o + R_s \gamma_g) \frac{G}{\beta_o^2} \sin \alpha \frac{\partial \beta_o}{\partial p_o}, \quad (44)$$

$$E_{12} = \epsilon \gamma_g \frac{G}{\beta_o} \sin \alpha \frac{\partial R_s}{\partial p_o}, \quad (45)$$

$$E_{13} = (\gamma_o + R_s \gamma_g) \frac{G}{\beta_o^2} \cos \alpha \frac{\partial \beta_o}{\partial p_o}, \quad (46)$$

$$E_{14} = \gamma_g \frac{G}{\beta_o} \cos \alpha \frac{\partial R_s}{\partial p_o}, \quad (47)$$

$$E_{15} = \frac{k_o}{\mu_o}, \quad (48)$$

$$E_{16} = \frac{k_g}{\mu_g}, \quad (49)$$

and for the saturation equation,

$$\frac{\partial \sigma_o}{\partial t} = \sigma_o e^{c_1} + c_4 \quad (50)$$

where

$$c_1 = \frac{1}{\beta_o} \frac{\partial \beta_o}{\partial p_o} \frac{\partial p_o}{\partial t}, \quad (51)$$

$$c_4 = c_2 + c_3, \quad (52)$$

$$c_2 = \frac{1}{\beta_o} \frac{\partial \beta_o}{\partial p_o} \left(\epsilon v_{ou} \frac{\partial p_o}{\partial u} + v_{oz} \frac{\partial p_o}{\partial z} \right) \frac{1}{\phi}, \quad (53)$$

$$c_3 = -\frac{1}{\phi} \left(\delta \epsilon v_{ou} + \frac{\partial v_{ou}}{\partial u} + \frac{\partial v_{oz}}{\partial z} \right). \quad (54)$$

ONE-PHASE OR TWO-PHASE OPTION

For problems involving the simultaneous flow of two fluids, the full set of equations is used. For problems involving only one fluid, this fluid is assumed to be the wetting phase. This condition would prevail, for example, in an oil reservoir where the oil pressure is always higher than that pressure at which any gas in solution could become free gas. The same condition would prevail in a natural gas reservoir or in a confined aquifer with no trapped air.

In running problems of this type, the nonwetting or gas phase is eliminated by setting its density (γ_g) to zero. With only one fluid present, then, the saturation equation (50) is equal to 1 for all time.

FINITE DIFFERENCE APPROXIMATIONS

The finite difference analog to Eq. (33) is

$$\begin{aligned}
 & \left[c_{j,k}^n - \theta_{j,k}^n \phi_{j,k} \left(1 - \sigma_{o,j,k}^{n+1} \right) \right] \left[p_{o,j,k}^{n+2} - p_{o,j,k}^n \right] \frac{1}{2\Delta x^{n+2}} \\
 &= \epsilon_{j,k}^n \left\{ \rho \left[\delta \epsilon_{j,k} E_{16,j,k}^n \left\{ \epsilon_{j,k} \left[\left(p_{o,j+2,k}^{n+2} - p_{o,j-2,k}^{n+2} \right) + \left(p_{c,j+2,k}^n - p_{c,j-2,k}^n \right) \right] \right. \right. \right. \\
 &\quad \times \frac{1}{4\Delta u} + E_{5,j,k}^n \left. \right\} + E_{16,j,k}^n \left(\epsilon_{j,k} \left[\left(p_{o,j+2,k}^{n+2} - 2p_{o,j,k}^{n+2} + p_{o,j-2,k}^{n+2} \right) \frac{1}{4\Delta u^2} \right. \right. \\
 &\quad + \left(p_{c,j+2,k}^n - 2p_{c,j,k}^n + p_{c,j-2,k}^n \right) \frac{1}{4\Delta u^2} - cs \left[\left(p_{o,j+2,k}^{n+2} - p_{o,j-2,k}^{n+2} \right) \frac{1}{4\Delta u} \right. \right. \\
 &\quad \left. \left. \left. - \left(p_{c,j+2,k}^n - p_{c,j-2,k}^n \right) \frac{1}{4\Delta u} \right] \right] + E_{9,j,k}^n \right\} + \left\{ \epsilon_{j,k} \left[\left(p_{o,j+2,k}^{n+2} - p_{o,j-2,k}^{n+2} \right) \right. \right. \right. \\
 &\quad \left. \left. \left. + \left(p_{c,j+2,k}^n - p_{c,j-2,k}^n \right) \right] \frac{1}{4\Delta u} + E_{5,j,k}^n \right\} E_{1,j,k}^n \right\} \\
 &+ \lambda \left[E_{16,j,k}^n \left[\left(p_{o,j,k+2}^n - 2p_{o,j,k}^n + p_{o,j,k-2}^n \right) \frac{1}{4\Delta z^2} \right. \right. \right. \\
 &\quad + \left(p_{c,j,k+2}^n - 2p_{c,j,k}^n + p_{c,j,k-2}^n \right) \frac{1}{4\Delta z^2} + E_{10,j,k}^n \left. \right] + \left[\left(p_{o,j,k+2}^n - p_{o,j,k-2}^n \right) \right. \\
 &\quad \left. \left. \times \frac{1}{4\Delta z} + \left(p_{c,j,k+2}^n - 2p_{c,j,k}^n + p_{c,j,k-2}^n \right) \frac{1}{4\Delta z} + E_{6,j,k}^n \right] E_{3,j,k}^n \right\} \left. \right\}
 \end{aligned}$$

$$\begin{aligned}
& - \eta_{j,k}^n \left\{ \rho \left[\epsilon_{j,k} E_{15,j,k}^n \left[\epsilon_{j,k} \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} + E_{7,j,k}^n \right] + E_{15,j,k}^n \right. \right. \\
& \quad \times \left. \left[\epsilon_{j,k} \left[\left(p_{o_{j+2,k}}^{n+2} - 2p_{o_{j,k}}^{n+2} + p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u^2} \right. \right. \right. \\
& \quad - \left. \left. \left. \cos \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} \right] - E_{11,j,k}^n \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} \right. \right. \right. \\
& \quad + \left. \left. \left. E_{12,j,k}^n \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} \right] \right. \right. \\
& \quad + \left. \left[\epsilon_{j,k} \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} + E_{7,j,k}^n \right] E_{2,j,k}^n \right] \\
& \quad + \lambda \left[E_{15,j,k}^n \left[\left(p_{o_{j,k+2}}^n - 2p_{o_{j,k}}^n + p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z^2} - E_{13,j,k}^n \left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \right. \right. \\
& \quad \times \left. \left. \frac{1}{4\Delta z} + E_{14,j,k}^n \left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z} \right] + \left[\left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \right. \right. \\
& \quad \times \left. \left. \frac{1}{4\Delta z} + E_8 \right] E_4 \right] \left. \right\} \\
& + \theta_{j,k}^n \left\{ \rho \left[E_{16,j,k}^n \left[\epsilon_{j,k} \left[\left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} + \left(p_{c_{j+2,k}}^n - p_{c_{j-2,k}}^n \right) \frac{1}{4\Delta u} \right] \right] \right. \right. \\
& \quad \times \left. \left[\epsilon_{j,k} \left[\left(p_{o_{j+2,k}}^n - p_{o_{j-2,k}}^n \right) \frac{1}{4\Delta u} + \left(p_{c_{j+2,k}}^n - p_{c_{j-2,k}}^n \right) \frac{1}{4\Delta u} \right] + E_{5,j,k}^n \right] \right. \\
& \quad + \lambda \left[E_{16,j,k}^n \left[\left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z} + \left(p_{c_{j,k+2}}^n - p_{c_{j,k-2}}^n \right) \frac{1}{4\Delta z} \right] \right. \\
& \quad \times \left. \left[\left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z} + \left(p_{c_{j,k+2}}^n - p_{c_{j,k-2}}^n \right) \frac{1}{4\Delta z} + E_{6,j,k}^n \right] \right. \\
& \quad + \left. \left. \left. \left(1 - \sigma_{o_{j,k}}^{n+1} \right) \phi_{j,k} \left(p_{c_{j,k}}^{n+1} - p_{c_{j,k}}^{n-1} \right) \frac{1}{2\Delta t^{n+2}} \right] \right. \right\} \\
& + \tau_{j,k}^n \left\{ \rho \left[E_{15,j,k}^n \epsilon_{j,k} \left(p_{o_{j+2,k}}^{n+2} - p_{o_{j-2,k}}^{n+2} \right) \frac{1}{4\Delta u} \left[\epsilon_{j,k} \left(p_{o_{j+2,k}}^n - p_{o_{j-2,k}}^n \right) \frac{1}{4\Delta u} + E_{7,j,k}^n \right] \right. \right. \\
& \quad + \lambda \left[E_{15,j,k}^n \left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z} \left[\left(p_{o_{j,k+2}}^n - p_{o_{j,k-2}}^n \right) \frac{1}{4\Delta z} + E_{8,j,k}^n \right] \right] \left. \right\} \quad (55)
\end{aligned}$$

where cs is either 1 or 0, depending on whether the horizontal scaling is linear or logarithmic. Thus

$cs = 0$ (linear horizontal scaling)

$cs = 1$ (logarithmic horizontal scaling).

$$\zeta_{j,k}^n = \frac{\beta_{o,j,k}^n \beta_{g,j,k}^n}{(R_{s,j,k}^n - \beta_{o,j,k}^n \beta_{g,j,k}^n)} \quad (56)$$

$$\eta_{j,k}^n = 1 - \frac{R_{s,j,k}^n}{(R_{s,j,k}^n - \beta_{o,j,k}^n \beta_{g,j,k}^n)} \quad (57)$$

$$\theta_{j,k}^n = \frac{\zeta_{j,k}^n}{\beta_{g,j,k}^n} \Delta \beta_g \quad (58)$$

$$\tau_{j,k}^n = \frac{\eta_{j,k}^n}{\beta_{o,j,k}^n} \Delta \beta_o + \frac{\Delta R_s}{(R_{s,j,k}^n - \beta_{o,j,k}^n \beta_{g,j,k}^n)} \quad (59)$$

$$c_{j,k}^n = \sigma_{o,j,k}^{n+1} \tau_{j,k}^n + \theta_{j,k}^n (1 - \sigma_{o,j,k}^{n+1}) \quad (60)$$

$$E_{1,j,k}^n = \epsilon_{j,k} \left[\frac{1}{\mu_{g,j,k}^n} (k_{g,j+1,k}^n - k_{g,j-1,k}^n) - \frac{k_{g,j,k}^n}{\mu_{g,j,k}^n} (\mu_{g,j+1,k}^n - \mu_{g,j-1,k}^n) \right] \frac{1}{2\Delta u} \quad (61)$$

$$E_{2,j,k}^n = \epsilon_{j,k} \left[\frac{1}{\mu_{o,j,k}^n} (k_{o,j+1,k}^n - k_{o,j-1,k}^n) - \frac{k_{o,j,k}^n}{\mu_{o,j,k}^n} (\mu_{o,j+1,k}^n - \mu_{o,j-1,k}^n) \right] \frac{1}{2\Delta u} \quad (62)$$

$$E_{3,j,k}^n = \left[\frac{1}{\mu_{g,j,k}^n} (k_{g,j,k+1}^n - k_{g,j,k-1}^n) - \frac{k_{g,j,k}^n}{\mu_{g,j,k}^n} (\mu_{g,j,k+1}^n - \mu_{g,j,k-1}^n) \right] \frac{1}{2\Delta z} \quad (63)$$

$$E_{4,j,k}^n = \left[\frac{1}{\mu_{o,j,k}^n} (k_{o,j,k+1}^n - k_{o,j,k-1}^n) - \frac{k_{o,j,k}^n}{\mu_{o,j,k}^n} (\mu_{o,j,k+1}^n - \mu_{o,j,k-1}^n) \right] \frac{1}{2\Delta z} \quad (64)$$

$$E_{5,j,k}^n = \gamma_g G \beta_{g,j,k}^n \sin \alpha \quad (65)$$

$$E_{6,j,k}^n = \gamma_g G \beta_{g,j,k}^n \cos \alpha \quad (66)$$

$$E_{7,j,k}^n = \left(\gamma_o + R_{s,j,k}^n \gamma_g \right) \frac{G}{\beta_{o,j,k}^n} \sin \alpha \quad (67)$$

$$E_{8,j,k}^n = \left(\gamma_o + R_{s,j,k}^n \gamma_g \right) \frac{G}{\beta_{o,j,k}^n} \cos \alpha \quad (68)$$

$$E_{9,j,k}^n = \epsilon_{j,k} \gamma_g G \sin \alpha \Delta \beta_g \left(p_{g,j+1,k}^n - p_{g,j,k-1}^n \right) \frac{1}{2\Delta u} \quad (69)$$

$$E_{10,j,k}^n = \gamma_g G \cos \alpha \Delta \beta_g \left(p_{g,j,k+1}^n - p_{g,j,k-1}^n \right) \frac{1}{2\Delta z} \quad (70)$$

$$E_{11,j,k}^n = \epsilon_{j,k} \left[\left(\gamma_o + R_{s,j,k}^n \gamma_g \right) \frac{G}{(\beta_{o,j,k}^n)^2} \sin \alpha \Delta \beta_o \right] \quad (71)$$

$$E_{12,j,k}^n = \epsilon_{j,k} \gamma_g \frac{G}{\beta_{o,j,k}^n} \sin \alpha \Delta R_s \quad (72)$$

$$E_{13,j,k}^n = \left(\gamma_o + R_{s,j,k}^n \gamma_g \right) \frac{G}{(\beta_{o,j,k}^n)^2} \cos \alpha \Delta \beta_o \quad (73)$$

$$E_{14,j,k}^n = \gamma_g \frac{G}{\beta_{o,j,k}^n} \cos \alpha \Delta R_s \quad (74)$$

$$E_{15,j,k}^n = \frac{k_{o,j,k}^n}{\mu_{o,j,k}^n} \quad (75)$$

$$E_{16,j,k}^n = \frac{k_{g,j,k}^n}{\mu_{g,j,k}^n} \quad (76)$$

and

$$\Delta \beta_g = \frac{\partial \beta_g}{\partial p_g} = \text{constant}$$

$$\Delta\beta_o = \frac{\partial\beta_o}{\partial p_o} = \text{constant}$$

$$\Delta R_s = \frac{\partial R_s}{\partial p_o} = \text{constant.}$$

The finite difference analog to Eq. (50) is

$$\sigma_{o,j,k}^{n+1} = \sigma_{o,j,k}^{n-1} e^{\left(2\Delta t^{n+2} C_{1,j,k}^{n+1}\right)} + 2\Delta t^{n+2} C_{4,j,k}^{n+1} \quad (77)$$

where

$$C_{1,j,k}^{n+1} = \Delta\beta_{o,j,k}^{n+1} \left(p_{o,j,k}^{n+2} - p_{o,j,k}^n\right) / \left(2\beta_{o,j,k}^{n+1} \Delta t^{n+2}\right) \quad (78)$$

$$C_{4,j,k}^{n+1} = \frac{\Delta\beta_{o,j,k}^{n+1}}{\phi_{j,k}\beta_{o,j,k}^{n+1}} \left[\epsilon_{j,k} v_{ou,j,k}^{n+1} \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1}\right) \frac{1}{2\Delta u} \right. \\ \left. + v_{oz,j,k}^{n+1} \left(p_{o,j,k+1}^{n+1} - p_{o,j,k-1}^{n+1}\right) \frac{1}{2\Delta z} \right] \\ - \frac{1}{\phi_{j,k}} \left[\delta\epsilon_{j,k} v_{ou,j,k}^{n+1} + \Delta v_{oz,j,k}^{n+1} \right]. \quad (79)$$

The finite-difference approximations for the velocities [Eqs. (5)-(8)] are:

$$v_{ou,j,k}^{n+1} = - \frac{k_{o,j,k}^{n+1}}{\mu_{o,j,k}^{n+1}} \left[\epsilon_{j,k} \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1}\right) \frac{1}{2\Delta u} + \left(\gamma_o + R_{s,j,k}^{n+1}\right) \frac{G}{\beta_{o,j,k}^{n+1}} \sin \alpha \right] \quad (80)$$

$$v_{oz,j,k}^{n+1} = - \frac{k_{o,j,k}^{n+1}}{\mu_{o,j,k}^{n+1}} \left[\left(p_{o,j,k+1}^{n+1} - p_{o,j,k-1}^{n+1}\right) \frac{1}{2\Delta z} + \left(\gamma_o + R_{s,j,k}^{n+1}\right) \frac{G}{\beta_{o,j,k}^{n+1}} \cos \alpha \right] \quad (81)$$

$$v_{gu,j,k}^{n+1} = - \frac{k_{g,j,k}^{n+1}}{\mu_{g,j,k}^{n+1}} \left[\epsilon_{j,k} \left(p_{g,j+1,k}^{n+1} - p_{g,j-1,k}^{n+1}\right) \frac{1}{2\Delta u} + \gamma_g G \beta_{g,j,k}^{n+1} \sin \alpha \right] \quad (82)$$

$$v_{gz,j,k}^{n+1} = - \frac{k_{g,j,k}^{n+1}}{\mu_{g,j,k}^{n+1}} \left[\left(p_{g,j,k+1}^{n+1} - p_{g,j,k-1}^{n+1}\right) \frac{1}{2\Delta z} + \gamma_g G \beta_{g,j,k}^{n+1} \cos \alpha \right] \quad (83)$$

The space derivatives of the velocities can be approximated by either of two methods. One method commonly used is, for example:

$$\frac{\Delta v_{ou,j,k}^{n+1}}{\Delta u} = \epsilon_{j,k} \left(v_{ou,j+1,k}^{n+1} - v_{ou,j-1,k}^{n+1} \right) \frac{1}{2\Delta u}.$$

The other method, which is the method employed here, is to first differentiate the velocity equations, and then difference the resulting new equations. These are as follows:

$$\begin{aligned} \Delta v_{ou,j,k}^{n+1} = & -E_{15,j,k}^{n+1} \left[\epsilon_{j,k} \left\{ \left(p_{o,j+1,k}^{n+1} - 2p_{o,j,k}^{n+1} + p_{o,j-1,k}^{n+1} \right) \frac{1}{\Delta u^2} - cs \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} \right\} \right. \\ & \left. - E_{11,j,k}^{n+1} \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} + E_{12,j,k}^{n+1} \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} \right] \\ & - \left[\epsilon_{j,k} \left(p_{o,j+1,k}^{n+1} - p_{o,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} + E_{7,j,k}^{n+1} \right] E_{2,j,k}^{n+1} \end{aligned} \quad (84)$$

$$\begin{aligned} \Delta v_{oz,j,k}^{n+1} = & -E_{15,j,k}^{n+1} \left[\left(p_{o,j,k+1}^{n+1} - 2p_{o,j,k}^{n+1} + p_{o,j,k-1}^{n+1} \right) \frac{1}{\Delta z^2} - E_{13,j,k}^{n+1} \left(p_{o,j,k+1}^{n+1} - p_{o,j,k-1}^{n+1} \right) \frac{1}{2\Delta z} \right. \\ & \left. + E_{14,j,k}^{n+1} \left(p_{o,j,k+1}^{n+1} - p_{o,j,k-1}^{n+1} \right) \frac{1}{2\Delta z} \right] - \left[\left(p_{o,j,k+1}^{n+1} - p_{o,j,k-1}^{n+1} \right) \frac{1}{2\Delta z} \right. \\ & \left. + E_{8,j,k}^{n+1} \right] E_{4,j,k}^{n+1} \end{aligned} \quad (85)$$

$$\begin{aligned} \Delta v_{gu,j,k}^{n+1} = & -E_{16,j,k}^{n+1} \left[\epsilon_{j,k} \left\{ \left(p_{g,j+1,k}^{n+1} - 2p_{g,j,k}^{n+1} + p_{g,j-1,k}^{n+1} \right) \frac{1}{\Delta u^2} \right. \right. \\ & \left. \left. - cs \left(p_{g,j+1,k}^{n+1} - p_{g,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} \right\} + \gamma_g G \Delta \beta_g \sin \alpha \right. \\ & \left. \times \left(p_{g,j+1,k}^{n+1} - p_{g,j-1,k}^{n+1} \right) \frac{1}{2\Delta u} \right] - \left[\epsilon_{j,k} \left(p_{g,j+1,k}^{n+1} - p_{g,j-1,k}^{n+1} \right) \right. \\ & \left. \times \frac{1}{2\Delta u} + E_{5,j,k}^{n+1} \right] E_{1,j,k}^{n+1} \end{aligned} \quad (86)$$

$$\begin{aligned} \Delta v_{gz,j,k}^{n+1} = & -E_{16,j,k}^{n+1} \left[\left(p_{g,j,k+1}^{n+1} - 2p_{g,j,k}^{n+1} + p_{g,j,k-1}^{n+1} \right) \frac{1}{\Delta z^2} \right. \\ & + \gamma_g G \Delta \beta_g \cos \alpha \left(p_{g,j,k+1}^{n+1} - p_{g,j,k-1}^{n+1} \right) \frac{1}{2\Delta z} \left. \right] - \left[\left(p_{g,j,k+1}^{n+1} - p_{g,j,k-1}^{n+1} \right) \right. \\ & \times \frac{1}{2\Delta z} + E_{6,j,k}^{n+1} \left. \right] E_{4,j,k}^{n+1}. \end{aligned} \quad (87)$$

TIME INCREMENTS

Numerical instability problems are generally inherent in any finite-difference approximation scheme. The restraints imposed to minimize these problems are largely dependent on the method of solution of the finite-difference approximations. Implicit schemes have been shown to be less restrictive, in this sense, than explicit schemes.

Because of the nature of the two predictive equations (55) and (77), the OGRE code employs both implicit and explicit schemes for their solution. The pressure equation (55) is solved implicitly and the saturation equation (77) is solved explicitly. The criteria for numerical stability, naturally, are different for each scheme.

For the explicit solution to the saturation equation the requirement on the time increment is

$$\Delta t_c^n \leq \frac{\phi_{j,k} \bar{C}_{j,k}^n \Delta x^2}{2M_{j,k}^n \left[1 + \left(\frac{\Delta x}{\Delta z} \right)^2 \right]} \quad (88)$$

where

$$\bar{C}_{j,k}^n = \Delta \beta_o \sigma_{o,j,k}^n + \Delta \beta_g (1 - \sigma_{o,j,k}^n)$$

$$M_{j,k}^n = \frac{k_{o,j,k}^n}{\mu_{o,j,k}^n} + \frac{k_{g,j,k}^n}{\mu_{g,j,k}^n}$$

$$\Delta x = \begin{cases} r_w \Delta u & \text{(linear scaling)} \\ r_w e^{u_j \Delta u} & \text{(logarithmic scaling).} \end{cases}$$

It has been found empirically that time increments on the order of 50 times Δt_c are reasonable as upper limits for use in the implicit solution of the pressure equation. For those problems, then, where only one fluid phase exits, an initial arbitrary time

increment less than or equal to Δt_c is chosen and then increased periodically as the problem progresses. If, of course, the second phase appears during the progress of the problem, the time increments must also satisfy Eq. (88).

An additional constraint on the time increment exists whenever a problem includes a heterogeneous permeability distribution. In this case,

$$\Delta t_c^n \leq \left| \frac{\phi_{j,k} \mu_{j,k}^n \Delta u^2}{2\epsilon_{j,k}^n B_{1,j,k}^n k_{o,j,k}^n} \right| \quad (89)$$

and

$$\Delta t_c^n \leq \left| \frac{\phi_{j,k} \Delta u}{\sqrt{2\epsilon_{j,k}^n (B_{2,j,k}^n + B_{3,j,k}^n + B_{4,j,k}^n)}} \right| \quad (90)$$

where

$$B_{1,j,k}^n = - \frac{\beta_{o,j,k}^n (\beta_{g,j,k}^n)^2}{\Delta \beta_g}$$

$$B_{2,j,k}^n = v_{ou,j,k}^n$$

$$B_{3,j,k}^n = \frac{\epsilon_{j,k} B_{1,j,k}^n k_{o,j,k}^n}{\mu_{o,j,k}^n}$$

$$B_{4,j,k}^n = \frac{\epsilon_{j,k} B_{1,j,k}^n (k_{o,j+1,k}^n - k_{o,j-1,k}^n)}{2\Delta u \mu_{o,j,k}^n} .$$

For any problem, then, the time increment is the minimum value of Δt determined from all these criteria.

PRESSURE EQUATION—MATRIX SOLUTIONS

The implicit finite difference solutions to the pressure equation (55) are arrived at by the simultaneous solutions of sets of linearized algebraic equations. Since the system is two-dimensional, two sets of solutions are necessary to advance the problem through one time cycle; one set in the u, or horizontal direction, and one set in the z, or vertical direction. This technique is commonly referred to as iterative ADIP (Alternating Direction Implicit Program).

The unknown quantities in the u-direction are $p_{o_{j-2,k}}^{n+2}$, $p_{o_{j,k}}^{n+2}$ and $p_{o_{j+2,k}}^{n+2}$. In the z-direction the unknowns are $p_{o_{j,k-2}}^{n+4}$, $p_{o_{j,k}}^{n+4}$ and $p_{o_{j,k+2}}^{n+4}$. After collecting the appropriate terms in Eq. (55), the u- and z-direction equations have the following form:

$$U_{j-2,k}^n p_{o_{j-2,k}}^{n+2} + U_{j,k}^n p_{o_{j,k}}^{n+2} + U_{j+2,k}^n p_{o_{j+2,k}}^{n+2} = X_{j,k}^n \quad (91)$$

for the u-direction, and

$$V_{j,k-2}^{n+2} p_{o_{j,k-2}}^{n+4} + V_{j,k}^{n+2} p_{o_{j,k}}^{n+4} + V_{j,k+2}^{n+2} p_{o_{j,k+2}}^{n+4} = Y_{j,k}^{n+2} \quad (92)$$

where U, V, X, and Y are the matrix coefficients. The boundary equations (23) through (29) are written in the same form. For a system of five grid points in the u-direction and five in the z-direction, the full set of equations would be:

$$\begin{aligned} U_{1,1,k}^n p_{o_{1,k}}^{n+2} + U_{1,2,k}^n p_{o_{2,k}}^{n+2} &= X_{1,k}^n \\ U_{2,1,k}^n p_{o_{1,k}}^{n+2} + U_{2,2,k}^n p_{o_{2,k}}^{n+2} + U_{2,3,k}^n p_{o_{3,k}}^{n+2} &= X_{2,k}^n \\ U_{3,2,k}^n p_{o_{2,k}}^{n+2} + U_{3,3,k}^n p_{o_{3,k}}^{n+2} + U_{3,4,k}^n p_{o_{4,k}}^{n+2} &= X_{3,k}^n \quad (93) \\ U_{4,3,k}^n p_{o_{3,k}}^{n+2} + U_{4,4,k}^n p_{o_{4,k}}^{n+2} + U_{4,5,k}^n p_{o_{5,k}}^{n+2} &= X_{4,k}^n \\ U_{5,4,k}^n p_{o_{4,k}}^{n+2} + U_{5,5,k}^n p_{o_{5,k}}^{n+2} &= X_{5,k}^n \end{aligned}$$

for the u-direction, where $1 \leq k \leq 5$, and

$$\begin{aligned} V_{1,j,1}^{n+2} p_{o_{j,1}}^{n+4} + V_{1,j,2}^{n+2} p_{o_{j,2}}^{n+4} &= Y_{j,k}^{n+2} \\ V_{2,j,1}^{n+2} p_{o_{j,1}}^{n+4} + V_{2,j,2}^{n+2} p_{o_{j,2}}^{n+4} + V_{2,j,3}^{n+2} p_{o_{j,3}}^{n+4} &= Y_{j,2}^{n+2} \\ V_{3,j,2}^{n+2} p_{o_{j,2}}^{n+4} + V_{3,j,3}^{n+2} p_{o_{j,3}}^{n+4} + V_{3,j,4}^{n+2} p_{o_{j,4}}^{n+4} &= Y_{j,3}^{n+2} \quad (94) \\ V_{4,j,3}^{n+2} p_{o_{j,3}}^{n+4} + V_{4,j,4}^{n+2} p_{o_{j,4}}^{n+4} + V_{4,j,5}^{n+2} p_{o_{j,5}}^{n+4} &= Y_{j,4}^{n+2} \\ V_{5,j,4}^{n+2} p_{o_{j,4}}^{n+4} + V_{5,j,5}^{n+2} p_{o_{j,5}}^{n+4} &= Y_{j,5}^{n+2} \end{aligned}$$

for the z-direction where $1 \leq j \leq 5$.

Since the equations have been linearized, Eqs. (93) and (94) may be written in matrix form, thus:

$$|U^n| |p_o^{n+2}| = |X^n| \quad (95)$$

and

$$|V^{n+2}| |p_o^{n+4}| = |Y^{n+2}| \quad (96)$$

where

$$|U^n| = \begin{vmatrix} U_{1,1,k}^n & U_{1,2,k}^n & 0 & 0 & 0 \\ U_{2,1,k}^n & U_{2,2,k}^n & U_{2,3,k}^n & 0 & 0 \\ 0 & U_{3,2,k}^n & U_{3,3,k}^n & U_{3,4,k}^n & 0 \\ 0 & 0 & U_{4,3,k}^n & U_{4,4,k}^n & U_{4,5,k}^n \\ 0 & 0 & 0 & U_{5,4,k}^n & U_{5,5,k}^n \end{vmatrix} \quad (97)$$

$$|V^{n+2}| = \begin{vmatrix} V_{1,j,1} & V_{1,j,2} & 0 & 0 & 0 \\ V_{2,j,1} & V_{2,j,2} & V_{2,j,3} & 0 & 0 \\ 0 & V_{3,j,2} & V_{3,j,3} & V_{3,j,4} & 0 \\ 0 & 0 & V_{4,j,3} & V_{4,j,4} & V_{4,j,5} \\ 0 & 0 & 0 & V_{5,j,4} & V_{5,j,5} \end{vmatrix} \quad (98)$$

$$|p_c^{n+2}| = \begin{vmatrix} p_{o,1,k}^{n+2} \\ p_{o,2,k}^{n+2} \\ p_{o,3,k}^{n+2} \\ p_{o,4,k}^{n+2} \\ p_{o,5,k}^{n+2} \end{vmatrix} \quad (99) \quad |p_o^{n+4}| = \begin{vmatrix} p_{o,j,1}^{n+4} \\ p_{o,j,2}^{n+4} \\ p_{o,j,3}^{n+4} \\ p_{o,j,4}^{n+4} \\ p_{o,j,5}^{n+4} \end{vmatrix} \quad (100) \quad |X^n| = \begin{vmatrix} X_{1,k}^n \\ X_{2,k}^n \\ X_{3,k}^n \\ X_{4,k}^n \\ X_{5,k}^n \end{vmatrix} \quad (101) \quad |Y^{n+2}| = \begin{vmatrix} Y_{j,1}^{n+2} \\ Y_{j,2}^{n+2} \\ Y_{j,3}^{n+2} \\ Y_{j,4}^{n+2} \\ Y_{j,5}^{n+2} \end{vmatrix} \quad (102)$$

Solutions are then obtained by inverting matrices U^n and V^{n+2} :

$$|P_o^{n+2}| = |U^n|^{-1} |X^n| \quad (103)$$

$$|P_o^{n+4}| = |V^{n+2}|^{-1} |Y^{n+2}| \quad (104)$$

MASS CONSERVATION CHECKS

At the end of each computation cycle, the current mass of oil and gas (if present) is computed. The differences between the initial masses of oil and gas and the current masses are compared with the computed total mass flux of each phase into or out of the system. The difference between these quantities is a measure of the mass conservation error.

The error for each phase is computed in the following manner:

$$E_o = 1 - \frac{\Delta M_o}{Q_{o,cum}} \quad (105)$$

$$E_g = 1 - \frac{\Delta M_g}{Q_{g,cum}} \quad (106)$$

where

$$\Delta M_o = M_o \text{ (initial)} - M_o \text{ (current)}$$

$$\Delta M_g = M_g \text{ (initial)} - M_g \text{ (current)}$$

and

$$M_o = 2\pi\gamma_o G \sum_{k=2}^{jk-1} \Delta z \sum_{j=2}^{jj-1} \frac{\sigma_{o,j,k} \phi_{j,k}}{\beta_{o,j,k}} r_w^2 \omega$$

$$M_g = 2\pi\gamma_g G \sum_{k=2}^{jk-1} \Delta z \sum_{j=2}^{jj-1} \phi_{j,k} \left[(1 - \sigma_{o,j,k}) \beta_{g,j,k} + \sigma_{o,j,k} \frac{R_{s,j,k}}{\beta_{o,j,k}} \right] r_w^2 \omega$$

and

$$\omega = \begin{Bmatrix} e^{2u_{j+1}} - e^{2u_{j-1}} \\ u_{j+1}^2 - u_{j-1}^2 \end{Bmatrix}$$

depending on whether the horizontal scaling is linear or logarithmic.

The Computer Program

PROGRAM ORGANIZATION

The code consists of a main program and 21 subroutines, written in a combination of Fortran II and Fortran IV for the CDC 6600 system at LRL. The relationships among the segments are shown in Fig. 5. Their functions are as follows:

Fig. 5. OGRE code logic.

MAIN	This routine reads most of the input data and generates the initial conditions.
OUTPUT	Generates printed output at specified intervals. This output includes the current time, the pressure and saturation arrays, current flux rates, cumulative fluxes, and other field variable arrays.
PQCON	This is the main operating routine. Most of the other subroutines are called from this subroutine. The main functions performed are: <ul style="list-style-type: none"> ● Keeps track of current number of cycles and current time. ● Computes formation factors, viscosities, permeabilities, fluid velocities and their derivatives, saturations, gas pressures and capillary pressures. ● Computes well bore boundary oil pressures. ● Linearly interpolates oil pressures and saturations. ● Computes time steps.
COEFU	Computes u-direction coefficients for oil pressure matrix equations.
COEFZ	Computes z-direction coefficients for oil pressure matrix equations.
MATRIX	Solves u- and z-direction pressure equations by matrix inversion.
QUAD	Quadratically interpolates oil pressures near the well bore.
SOQUAD	Quadratically extrapolates oil saturations to the well bore.
K-TIME	Computes time steps for problems involving a heterogeneous permeability distribution.
OGBAL	Computes current mass in system. Computes error in mass conservation.
MOBAL	Cathode-ray tube (CRT) routine. Plots oil and gas mass conservation errors as a function of time.
QGXT	CRT routine. Plots gas flux rates as a function of time.
QOGXT	CRT routine. Plots oil flux rates as a function of time.
QOGCUM	CRT routine. Plots cumulative oil and gas fluxes as a function of time.
POUZ	CRT routine. Plots oil pressure profile at specified times.
SOUZ	CRT routine. Plots oil saturation profile at specified times.
PGUZ	CRT routine. Plots gas pressure profile at specified times.
VOUZ	CRT routine. Plots oil velocity field at specified times.
VGUZ	CRT routine. Plots gas velocity field at specified times.
OILSAT	CRT routine. Plots oil saturation profile at specified times. Uses a point density variation routine to indicate relative saturation.
SMART	Generates a "CLICHE" called STUFF consisting of "COMMON" and "DIMENSION" statements for inclusion in all other subroutines.

Appendix E is a Fortran listing of the OGRE code, and Appendix F presents detailed logical flow diagrams of each segment of the code.

INPUT

Input to the code is with punched cards. For most problems, 11 cards containing 62 variables are necessary. For cases where heterogeneous permeability and/or

porosity exist, additional punch cards are necessary which define the permeability and/or porosity distribution. The first 10 cards, plus any permeability and/or porosity data, are read by the main code; the last card of the original 11 is read by PQCON.

A routine in the main subroutine (OGRE) also permits the porosity and/or absolute permeability distributions to be generated as a function of several different geometries. For either card input or any of these geometric distributions, the input variable, JZ, must be nonzero. Then the input value of IG determines which of these distributions is to be used. The values of IG and the corresponding distributions are shown below.

- IG =
- 1—Two discrete zones with radial symmetry.
 - 2—Two discrete zones with spherical symmetry.
 - 3—Card input.
 - 4—Exponentially decaying spherical zone.

The format for the input cards is shown in Appendix C. The variable symbols are defined in Appendix B.

OUTPUT

Output for the OGRE code is of two types—that generated at specified output intervals and consisting of the pertinent results of the code computations, and that generated as the result of an error condition. The latter consists of intermediate results which are intended to aid in the diagnosis of the error. Some of these intermediate results may also be generated by the appropriate input parameters. This option may be useful in short trial runs of new problems. Output of the first type is produced both on the high-speed printer and the cathode-ray tube. Output of the second type is produced only on the high-speed printer. Appendix D presents examples of the output format for a normal two-phase problem.

Not shown are the initial condition oil pressure and saturation arrays, which are listed after the input data listing. Also not included are examples of error exit routine output.

It will be noted that, since the code is in a continual state of evolution, some output is not adequately labeled for identification. All output can be identified from the code listing, although this is admittedly an arduous process. As time permits, however, labels will be added so that all output is properly identified.

ERROR EXIT ROUTINES

Since there are a number of places in the code where computational troubles may occur because of input errors or unsuitable input conditions, error exit procedures are inserted at many of these places. These are included to prevent unnecessary loss of machine time, and to provide diagnostic output for determining the cause of the trouble.

In all cases, a complete output cycle and memory dump is executed. In some cases, additional relevant output is produced. In addition, there are places where the code will respond with an output cycle and memory dump to Sense Switch 1.

These error exit procedures occur in the following subroutines:

- PQCON - Occurs when the number of iterations to find the proper well bore pressure gradient to match a specified flow rate exceeds 100.
- PQCON - Sense Switch 1. Occurs in the routine which tests for proper production rate.
- PQCON - Occurs when any value in the oil saturation array exceeds 1.1.
- PQCON - Sense Switch 1. Occurs at the end of the output cycle.
- MATRIX - Occurs when X(I) is either negative or zero. The X, Y, RES, and A arrays are also produced. Since OMBB is now set equal to 1, the current values of COAO, COBO, COCO, and the arrays of PON and DPRO in QUAD are also produced.

References

1. R. D. Wyckoff and H. G. Botset, "The Flow of Gas-Liquid Mixtures through Unconsolidated Sands," Physics, 7 (1936).
2. W. T. Cardwell and R. L. Parsons, Gravity Drainage Theory (AIME Technical Publication No. 2464, November 1948).
3. S. J. Pirson, Oil Reservoir Engineering, (McGraw-Hill Book Co., New York, 1958).
4. G. E. Archie, "Electrical Resistivity Log as an Aid in Determining Some Reservoir Characteristics," Trans. AIME 146 (1942).
5. W. Rose, "Theoretical Generalizations Leading to the Evaluation of Relative Permeability," Trans. AIME (May 1949).
6. J. Kozeny, "Über-Kapillare Leitung des Wassers in Boden," Sitzb. Akad. Wiss. Wien., Math-nature. Kl., 136, 27-306 (1927).
7. P. C. Carman, "Capillary Rise and Capillary Movement of Moisture in Fine Sands," Soil Sci., 52, 1-14 (1941).

Additional References

Since a complete listing of the references consulted in the development of the OGRE code would be inordinately long, only those which played a major role in its development are presented.

N. J. Clark, "Theoretical Aspects of Oil and Gas Equilibrium Calculations," J. Petrol. Tech. (April 1962).

B. C. Craft and M. F. Hawkins, Applied Petroleum Reservoir Engineering (Prentiss-Hall, Inc., Englewood Cliffs, N. J., 1959).

J. Douglas, Jr., "On The Numerical Integration of $\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2 = \partial u / \partial t$ by Implicit Methods," J. Soc. Indust. Applied Math., Vol. 3, No. 1 (March 1955).

J. Douglas, Jr. and D. W. Peaceman, "Numerical Solution of Two-Dimensional Heat-Flow Problems," A. I. Ch. E. Journal Vol. 1, No. 4 (Dec. 1955).

I. Fatt and H. Dykstra, "Relative Permeability Studies," Trans. AIME, 192 (1951).

H. N. Hall, "Analysis of Gravity Drainage," J. Petrol. Technol. (Sept. 1961).

R. G. Hawthorne, "Two-Phase Flow in Two-Dimensional Systems-Effects of Rate, Viscosity and Density of Fluid Displacement in Porous Media," Trans. AIME, 219 (1960).

M. Muskat, The Flow of Homogeneous Fluids Through Porous Media (McGraw-Hill Book Co., New York, 1937).

M. Muskat, Physical Principles of Oil Production (McGraw-Hill Book Co., New York, 1949).

M. Muskat and M. W. Meres, "The Flow of Heterogeneous Fluids Through Porous Media," Physics, 7 (1936).

D. W. Peaceman and H. H. Rachford, "The Numerical Solution of Parabolic and Elliptic Differential Equations," J. Soc. Indust. Appl. Math., Vol. 3, No. 1 (March 1955).

D. W. Peaceman and H. H. Rachford, "Numerical Calculation of Multidimensional Miscible Displacement," J. Soc. Petrol. Engr. (Dec. 1962).

L. A. Rapaport and W. J. Leas, "Relative Permeability to Liquid in Liquid-Gas Systems," Trans. AIME, 192 (1951).

W. Rose and W. A. Bruce, "Evaluation of Capillary Character in Petroleum Reservoir Rock," Trans. AIME (May 1949).

H. L. Stone and A. O. Garder, "Analysis of Gas-Cap or Dissolved Gas Drive Reservoirs," J. Soc. Petrol. Engr. (June 1961).

P. L. Terwilliger, et al., "An Experimental and Theoretical Investigation of Gravity Drainage Performance," Trans. AIME, 192 (1951).

W. J. West, W. W. Garvin and J. W. Sheldon, "Solution of the Equations of Unsteady State Two-Phase Flow in Oil Reservoirs," Trans. AIME, 201 (1964).

Appendix A

Definition of Symbols

SYMBOL	DEFINITION	UNITS
B_1	Defined by Eq. (90)	
B_2	Defined by Eq. (90)	ft/sec
B_3	Defined by Eq. (90)	ft ² -sec/lb
B_4	Defined by Eq. (90)	ft-sec ² /lb
c	Defined by Eq. (60)	ft-sec ² /lb
c_1	Defined by Eq. (51)	1/sec
c_2	Defined by Eq. (53)	1/sec
c_3	Defined by Eq. (54)	1/sec
c_4	Defined by Eq. (52)	1/sec
C_{Bg}	Constant in Eq. (12)	
C_{Bo}	Constant in Eq. (11)	
C_{Rs}	Constant in Eq. (13)	
cs	Scaling parameter in Eqs. (55), (84), (86)	
C_{vg}	Constant in Eq. (10)	lb/sec-ft
C_{vo}	Constant in Eq. (9)	lb/sec-ft
E_1	Defined by Eq. (34)	sec-ft ² /lb
E_2	Defined by Eq. (35)	sec-ft ² /lb
E_3	Defined by Eq. (36)	sec-ft ² /lb
E_4	Defined by Eq. (37)	sec-ft ² /lb
E_5	Defined by Eq. (38)	lb/ft ³
E_6	Defined by Eq. (39)	lb/ft ³
E_7	Defined by Eq. (40)	lb/ft ³
E_8	Defined by Eq. (41)	lb/ft ³
E_9	Defined by Eq. (42)	lb/ft ⁴
E_{10}	Defined by Eq. (43)	lb/ft ⁴
E_{11}	Defined by Eq. (44)	sec ² /ft ³
E_{12}	Defined by Eq. (45)	sec ² /ft ³
E_{13}	Defined by Eq. (46)	sec ² /ft ²
E_{14}	Defined by Eq. (47)	sec ² /ft ²
E_{15}	Defined by Eq. (48)	sec-ft ³ /lb
E_{16}	Defined by Eq. (49)	sec-ft ³ /lb
G	Gravitational constant	ft/sec ²
j	Grid index in horizontal direction	
jj	Total grid points in horizontal direction	
k	Grid index in vertical direction	
jk	Total grid points in vertical direction	

k_a	Fully saturated (absolute) permeability	ft^2
k_c	Kozeny constant	
k_g	Gas permeability	ft^2
k_o	Oil permeability	ft^2
M	Defined by Eq. (88)	$\text{ft}^3\text{-sec/lb}$
n	Time index	
p_c	Capillary pressure	lb/ft-sec^2
p_g	Gas pressure	lb/ft-sec^2
p_o	Oil pressure	lb/ft-sec^2
p_{oi}	Initial condition oil pressure	lb/ft-sec^2
Q_g	Free Gas flux rate	lb/sec
Q_o	Oil flux rate	lb/sec
Q_{og}	Dissolved gas flux rate	lb/sec
$Q_{o\text{cum}}$	Cumulative oil flux	lb
$Q_{g\text{cum}}$	Cumulative gas flux	lb
r	Horizontal distance from well centerline	ft
R_s	Gas solubility factor	
t	Time	sec
u	Defined by Eq. (32)	
v_g	Gas velocity	ft/sec
v_{gr}	Gas velocity in the r direction	ft/sec
v_{gu}	Gas velocity in the u direction	ft/sec
v_{gz}	Gas velocity in the z direction	ft/sec
v_o	Oil velocity	ft/sec
v_{or}	Oil velocity in the r direction	ft/sec
v_{ou}	Oil velocity in the u direction	ft/sec
v_{oz}	Oil velocity in the z direction	ft/sec
z	Vertical distance above datum	ft
α	Angle measured from horizontal	radius
β_g	Gas formation factor	
β_o	Oil formation factor	
δ	Geometric flow parameter [Eq. (33)]	
ϕ	Formation porosity	
ϵ	Defined by Eq. (33)	$1/\text{ft}$
ψ	Surface tension	
η	Defined by Eq. (22)	
θ	Defined by Eq. (22)	$\text{ft-sec}^2/\text{lb}$
ζ	Defined by Eq. (22)	
τ	Defined by Eq. (22)	$\text{ft-sec}^2/\text{lb}$
ρ	Defined by Eq. (33)	
λ	Defined by Eq. (33)	

γ_g	Gas mass	$\text{lb-sec}^2/\text{ft}^4$
γ_o	Oil mass	$\text{lb-sec}^2/\text{ft}^4$
μ_g	Gas viscosity	$\text{lb}/\text{ft-sec}$
μ_o	Oil viscosity	$\text{lb}/\text{ft-sec}$
σ_g	Gas saturation	
σ_o	Oil saturation	
σ_{oi}	Irreducible oil saturation	
σ_w	Water saturation	
Δt	Time increment	sec
Δu	Increment in u	
Δz	Increment in z	ft
ΔR_s	$\partial R_s / \partial P_o$	$\text{ft-sec}^2/\text{lb}$
$\Delta \beta_g$	$\partial \beta_g / \partial P_g$ (constant)	$\text{ft-sec}^2/\text{lb}$
$\Delta \beta_o$	$\partial \beta_o / \partial P_o$ (constant)	$\text{ft-sec}^2/\text{lb}$
Δv_{gu}	Difference form of $\partial v_{gu} / \partial u$ [Eq. (86)]	ft/sec
Δv_{gz}	Difference form of $\partial v_{gz} / \partial z$ [Eq. (87)]	ft/sec
Δv_{ou}	Difference form of $\partial v_{ou} / \partial u$ [Eq. (84)]	ft/sec
Δv_{oz}	Difference form of $\partial v_{oz} / \partial z$ [Eq. (85)]	ft/sec
Δt_c	Critical time increment [Eq. (38)]	sec

Appendix B

Code Symbols

NAME	DEFINITION	EQUIVALENCE
A	Matrix routine array	A
AA	Unused	
AK	Saturated permeability	k_a
ALPHA	Angle measured from either the r or z-axis	α
ASON	SON-1.0	
AT	Term in exponential saturation equation	
AVGU	Absolute value of VGU	$ v_{gu} $
AVGZ	Absolute value of VGZ	$ v_{gz} $
AVOU	Absolute value of VOU	$ v_{ou} $
AVOZ	Absolute value of VOZ	$ v_{oz} $
AXE	TS(j + 1)	
AXS	TS(j)	
AYF	QOS(j + 1)	
AYS	QOS(j)	
B	Matrix routine array	
BAT	Term in Eq. (90)	B_3
BB	Unused	
BBT	Unused	
BG	Gas formation factor	β_g
BGA	Factor in mass computation equation	
BIT	Term in Eq. (90)	B_2
BO	Oil formation factor	β_o
BOA	Factor in mass computation equation	
BOI	Initial value of oil formation factor	
BOT	Term in Eq. (90)	B_4
BP	Oil bubble point	
BT	Term in exponential saturation equation	
BUT	Term in Eq. (90)	B_1
C	Defined by Eqs. (22) and (60)	c
CC	Unused	
CCT	Unused	
CK	Kozeny constant	k_c
COAG	Summation of quadratic solutions in QUAD	
COAO	Summation of quadratic solutions in QUAD	

COAS	Summation of quadratic solutions in SOQUAD	
COBG	Summation of quadratic solutions in QUAD	
COBO	Summation of quadratic solutions in QUAD	
COBS	Summation of quadratic solutions in SOQUAD	
COCG	Summation of quadratic solutions in QUAD	
COCO	Summation of quadratic solutions in QUAD	
COCs	Summation of quadratic solutions in SOQUAD	
CS	See ICS CS=ICS	
CT	Term in exponential saturation equation	
C4	Unused	
D	Intermediate solution in MATRIX	D
DBG	Constant gradient of gas formation factor	$\partial \beta_g / \partial p_g$
DBO	Constant gradient of oil formation factor	$\partial \beta_o / \partial p_o$
DD	Unused	
DELMG	Deviation in gas mass conservation	
DELMO	Deviation in oil mass conservation	
DELTA	Defined by Eq. (33)	δ
DEVG	Fractional error in gas mass conservation	
DEVO	Fractional error in oil mass conservation	
DKA	Permeability zoning parameter in MAIN	
DMGS	Time summary of gas mass balance error	
DMOS	Time summary of oil mass balance error	
DP	Increment in oil pressure	Δp_o
DPRG	Horizontal space derivative of gas pressure	$\partial p_g / \partial u$
DPRO	Horizontal space derivative of oil pressure	$\partial p_o / \partial u$
DQT	Increment in input flow rate	
DR	Increment in r-direction	r
DRS	Constant gradient of gas saturation	$\partial R_g / \partial p_o$
DT	Increment in time	Δt
DU	Increment in u-direction	Δu
DVGU	Horizontal space derivative of gas velocity	$\partial v_{gu} / \partial u$
DVGZ	Vertical space derivative of gas velocity	$\partial v_{gz} / \partial z$
DVOU	Horizontal space derivative of oil velocity	$\partial v_{ou} / \partial u$
DVOZ	Vertical space derivative of oil velocity	$\partial v_{oz} / \partial z$
DZ	Increment in z-direction	Δz
E	Coefficient array in COEFU and COEFZ	
EE	Unused	
EFA	Quadratic solution to SO in SOQUAD	
EFB	Quadratic solution to SO in SOQUAD	
EFC	Quadratic solution to SO in SOQUAD	
EFD	Quadratic solution to SO in SOQUAD	
EFE	Quadratic solution to SO in SOQUAD	

EFF	Quadratic solution to SO in SOQUAD	
EFG	Quadratic solution to SO in SOQUAD	
EFH	Quadratic solution to SO in SOQUAD	
EFI	Quadratic solution to SO in SOQUAD	
EFM	Quadratic solution to SO in QUAD	
EFN	Quadratic solution to SO in QUAD	
EFO	Quadratic solution to SO in QUAD	
EFP	Quadratic solution to SO in QUAD	
EFQ	Quadratic solution to SO in QUAD	
EFR	Quadratic solution to SO in QUAD	
EFS	Quadratic solution to SO in QUAD	
EFT	Quadratic solution to SO in QUAD	
EFU	Quadratic solution to SO in QUAD	
EMAX	Maximum value of SG	
EPSIL	Maximum allowable error in constant flow option	
ET	Unused	
ETA	Defined by Eqs. (22) and (57)	η
EU	Defined by Eq. (33) (in general) [*]	ϵ
E1	Defined by Eqs. (34) and (61)	E_1
E2	Defined by Eqs. (35) and (62)	E_2
E3	Defined by Eqs. (36) and (63)	E_3
E4	Defined by Eqs. (37) and (64)	E_4
E5	Defined by Eqs. (38) and (65)	E_5
E6	Defined by Eqs. (39) and (66)	E_6
E7	Defined by Eqs. (40) and (67)	E_7
E8	Defined by Eqs. (41) and (68)	E_8
E9	Defined by Eqs. (42) and (69)	E_9
E10	Defined by Eqs. (43) and (70)	E_{10}
E11	Defined by Eqs. (44) and (71)	E_{11}
E12	Defined by Eqs. (45) and (72)	E_{12}
E13	Defined by Eqs. (46) and (73)	E_{13}
E14	Defined by Eqs. (47) and (74)	E_{14}
E15	Defined by Eqs. (48) and (75)	E_{15}
E16	Defined by Eqs. (49) and (76)	E_{16}
F	Unused	
FF	Unused	
FM	AK/OVCON used in PQCON	

^{*}In some loops EU has a specific definition.

FPERM	Permeability multiplier	
FPOR	Porosity multiplier	
FT	Unused	
G	Gravitational constant	G
GA	Equivalent to $\Delta\beta_o BP + 1.04$	
GAMG	Gas mass	γ_g
GAMD	Oil mass	γ_o
GB	Unused	
GC	Unused	
GD	Unused	
GE	Unused	
GF	Unused	
GG	Unused	
GGPZ	Factor in mass computation equation	
GH	Unused	
GM	Initial mass of gas in system	M_g
GMBA	Initial time increment	Δt
GMBB	Time limit for lowering well pressure	
GMBC	Input name for EPSIL	
GMBD	Input name for BP	
GMCA	Input name for BOI	
GMCB	Unused	
GMCC	Unused	
GMCD	Unused	
GMI	Equivalent to GMT	
GMR	GM-GMI	
GMT	Current total mass of gas in system	
GOPZ	Factor in mass computation equation	
GOR	Gas/oil ratio	
GORS	Time summary of GOR	
GVCOEf	Gas viscosity coefficient	
GVCON	Gas viscosity constant	
H	Coefficient array in COEFU and COEFZ	
I	Loop index in COEFU and COEFZ	
ICON	Number of contours to be plotted	
ICOUNT	Current size of matrix array	
ICS	Test for u-direction scaling	
ICT	Test for card or tape input	
IG	Test for zoning geometry	
II	Unused	
IM	Unused	
INJ	Horizontal index in initial saturation loop	

INK	Vertical index in initial saturation loop	
IO	Flag for well-bore saturation computation	
IOPT	Control parameter for boundary condition	
IT	Number of cycles for constant DT	
ITER	Matrix iteration counter	
IUZ	Test for matrix coefficient entry point	
IZ	Permeability zoning parameter in MAIN	
J	Horizontal grid index	j
JA	Unused	
JB	Boundary index for space zoning	
JC	JB+1	
JJ	JK/2	
JJ	Maximum number of horizontal grid points	
JK	Maximum number of vertical grid points	
JL	JJ/2	
JM	JJ-1	
JN	JK-1	
JO	JJ-2	
JP	JK-2	
JPP	JS-1	
JS	Time summary index for JT	
JT	Test value for time summary printout	
JXK	Numerical value of JJ x JK	
JZ	Test for space zoning	
K	Vertical grid index	k
KK	Unused	
KKK	Unused	
KKL	Test on DT increment	
KL	Cycle counter	
KM	Test for storing time summaries	
KR	Constant in OILSAT	
KZ	Unused	
K2	ICON+1	
L	Printout cycle counter test	
LA	MO-2	
LB	MM+2	
LJ	Index of u-direction grid point of lowest DT in KTIME	
LK	Index of z-direction grid point of lowest DT in KTIME	
LN	Limit of interaction loop for initial hydrostatic head	

LSPIN	Test for terminating QO iterations	
LT	Iteration index for DT increase	
LX	Test in incrementing ΔDT	
MI	Loop index in MATRIX	
MF	Constant in QOGXT	
MG	Constant in QOGXT	
MI	Constant 2 in PQCON	
MM	Upper well boundary index	
MN	Loop index in COEFU and COEFZ	
MO	Lower well boundary index	
MX	Control parameter for KTIME printout	
MZ	Index of maximum value of SG	
N	Output cycle index	
NN	Loop index in COEFU and COEFZ	
NO	Unused	
NP	Number of cycles between printout	
NQ	Unused	
NT	Number of cycles between storing time summaries	
ODT	Previous value of DT	
OM	Initial mass of oil in system	
OMBA	Temporary location for DELTA	
OMBB	Test for matrix coefficient printout	
OMBC	Input value of DP	
OMBD	Unused	
OMCA	Equivalent to RSi	
OMCB	Unused	
OMCC	Input value of SBO	
OMCD	Unused	
OMI	Equivalent to OMT	
OMR	OM-OMI	
OMT	Current total mass of oil in system	
ORS	Previous value of RS	R_s
OVCOEF	Oil viscosity coefficient	
OVCON	Oil viscosity constant	
P	Scaling factor in plotting routines	
PC	Capillary pressure	P_c
PCMAX	Maximum value of PC array	
PCMIN	Minimum value of PC array	
PCO	Previous value of PC	P_c
PDIF	Pressure difference test factor (POLA)	
PERM	Space dependent permeability	

PG	Gas pressure	p_g
PGM	Unused	
PGMAX	Maximum value of PG array	
PGMIN	Minimum value of PG array	
PHI	Formation porosity	ϕ
PI	Constant	π
PKG	Gas permeability	k_g
PKO	Oil permeability	k_o
PN	Intermedial oil pressures	p_o^{n+1}
PO	Oil pressure (old)	p_o^n
PIO	Initial formation pressure	p_o
POMAX	Maximum value of PO array	
POMIN	Minimum value of PO array	
PON	Oil pressure (new)	p_o^{n+2}
POR	Space dependent formation porosity	
PSI	Oil surface tension	ψ
PTA	Equivalent to DBG	
PTB	Term in option 3 of PQCON	
1 TC	Term in option 3 of PQCON	
PW	Initial bottom hole oil well pressure	
Q	Scaling factor in plotting routines	
QD	QT/ZD	
QG	Gas mass flow rate	Q_g
QGCS	Unused	
QO	Oil mass flow rate	Q_o
QOCS	Unused	
QOCUM	Cumulative oil production	$Q_{o\text{cum}}$
QOG	Dissolved gas mass flow rate	Q_{og}
QOGS	Time summary of QOG	Q_{og}
QOS	Time summary of QO	Q_o
QOSC	Oil volume flow rate	
QOSTB	Unused	
QP	QT $\times$ TIME/GMBB	
QQ	Test value of QO in PQCON option 2	
QT	Total mass flow rate	
QTS	Time summary of QT	
QX	Summation of oil mass flow rate	
QY	Summation of gas mass flow rate	
QZ	Summary of dissolved gas mass flow rate	
R	Unused	r

RATIO	Term in linear oil pressure interpolations	
RES	Matrix residuals	
RHO	u-direction parameter	ρ
RJ	Difference of the squares of the radii	
RJJ	Unused	
RPRW	Equivalent to RWx U(J) in MAIN	
RS	Gas solubility factor	R_s
RSI	Initial gas saturation	
RW	Well radius	r_w
S	Intermediate solution of matrix equations	s
SBO	Single phase oil formation factor	s_{bo}
SG	Gas saturation	s_g
SGM	Initial gas saturation	s_{gm}
SO	Oil saturation	s_o
SOI	Irreducible oil saturation	
SOM	Initial oil saturation	
SOMAX	Maximum value of SO array	
SOMIN	Minimum value of SO array	
SON	New oil saturation	s_o^{n+2}
SRS	Single-phase gas saturation	
STEST	Test on constant in QOGXT	
TAU	Defined by Eqs. (22) and (59)	τ
TEMP	Intermediate solution in MATRIX	
TERM	Term in Eqs. (14) and (15)	
THETA	Defined by Eqs. (22) and (58)	θ
TIME	Elapsed real time	
TKI	Solution to Eq. (89)	Δt_c
TK2	Solution to Eq. (90)	Δt_c
TOLER	EPSIL x TIME/GMBS	
TS	Time summary of time	
TSS	Pressure transient transit time to outer boundary	
U	Horizontal coordinate	u
UM	Equivalent to U(JJ)	
UMA	Equivalent to U(1)	
V	Unused	
VB	Unused	
VGU	Radial gas velocity	v_{gu}
VGZ	Vertical gas velocity	v_{gz}
VISG	Gas viscosity	μ_{sg}
VISO	Oil viscosity	μ_{so}
VMAX	Maximum velocity	

VOU	Radial oil velocity	v_{ou}
VOZ	Vertical oil velocity	v_{oz}
W	Array of contour levels	
X	Matrix value of oil pressure	X
XAX	Constant in SOQUAD	
XB	Equivalent to U(J) in VGUZ and VOUZ	
XE	Ratio of horizontal velocity to VMAX	
XI	Equivalent to DU or DZ in VGUZ and VOUZ	
XTM	Equivalent to TS(JS)	
XTO	Equivalent to TS(1)	
X1	U(J-1) in OILSAT	
X2	U(J+1) in OILSAT	
X3	U(J+1) in OILSAT	
X4	U(J-1) in OILSAT	
Y	Right-hand side of matrix equation	Y
YB	Equivalent to Z(K) in VGVZ and VOVZ	
YDA	Maximum and minimum values of QOS and QOGS	
YDMN	Minimum value of YDA	
YDMX	Maximum value of YDA	
YE	Ratio of vertical velocity to VMAX	
Y1	Z(k-1) in OILSAT	
Y2	Z(k-1) in OILSAT	
Y3	Z(k+1) in OILSAT	
Y4	Z(k+1) in OILSAT	
Z	Vertical coordinate	z
ZA	Permeability zoning parameter in MAIN	
ZD	JK/2	
ZETA	Defined by Eqs. (22) and (56)	ζ
ZLAM	z-direction parameter	
ZM	Equivalent to Z(JK)	
ZMA	Equivalent to Z(1)	

Appendix C

Input Card Format

IOPT	ICT	NP	NT	JZ	JT	IG	CARD NO.
1	0	100	1	0	0	1	1-11

IUZ	ICS	IT	GMBB	OMBC	GMBC	OMBD	CARD NO.
1	1	1	0.000E+00	0.000E+00	0.000E+00	0.000E+00	1-11

RHO	ZLAM	DELTA	ICON	GMBA	OMBB	PDIF	CARD NO.
1.0	1.0	1.0	0.0	0.000E+00	0.000E+00	0.000E+00	1-11

CK	PHI	AK	GAMO	GAMG	POI	CARD NO.
0.000E+00	1.000E-01	1.000E-01	1.000E-01	1.000E+00	1.000E+00	1-11

SOI	DU	DZ	DBO	DBG	DRS	CARD NO.
1.000E-01	0.000E-01	0.000E-01	0.000E+00	0.000E+00	0.000E+00	1-11

QT	G	PI	PSI	RW	CARD NO.
0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	1-11

OIL AND GAS RESERVOIR ENGINEERING CODE

KL = 5

PM = CONSTANT = 3.160E+05

L = 5 BT = 2.00E-01 GISC = 2.763E-05 GT = 4.945E-03 GOR = 6.792E+02

TIME = 2.000E+00 INITIAL OIL = 4.366E+09 INITIAL GAS = 3.102E+08

OIL PRODUCED = 2.053E-03 GAS PRODUCED = 1.482E-04 OIL MASS DIF. = 1.185E+07

GAS MASS DIF. = -1.423E+06 OIL MASS DEV. = -5.772E+09 GAS MASS DEV. = 9.603E+09

Z	OF	OG	OG
1.0000E+00	1.3993E-04	1.0102E-05	3.1812E-44
3.0000E+00	1.3993E-04	1.0099E-05	3.1810E-44
5.0000E+00	1.3993E-04	1.0095E-05	9.0567E-20
7.0000E+00	1.3988E-04	1.0091E-05	9.0512E-20
9.0000E+00	1.3985E-04	1.0088E-05	9.0466E-20
1.1000E+01	1.3981E-04	1.0084E-05	9.0398E-20
1.3000E+01	1.3980E-04	1.0081E-05	2.6562E-19
1.5000E+01	1.3978E-04	1.0077E-05	2.6459E-19
1.7000E+01	1.4056E-04	1.0172E-05	1.3092E-19
1.9000E+01	1.4100E-04	1.0162E-05	9.7140E-20

Appendix D

Data Output Format

OIL AND GAS RESERVOIR ENGINEERING CODE

INPUT DATA

INTY	ICT	NP	NT	JZ	JT	IG	
1	0	50	1	0	47	1	
INZ	ICS	IT	GMBB	BP	EPSILN	OMBD	
0	1	1000	100.0	1.00E+00	6.30E-03	0.	
WOB	ZLAW	DELTA	GMBR	ICDH	GVGB	GWCC	
1.00	1.00	1.00	0.20	-21	0.	-1.21E-07	
CK	PHI	KK	GMB	GMB5	PGI		
5.000E+00	2.500E-01	1.000E-13	1.563E+00	9.320E-04	3.880E+05		
SST	DU	DZ	DBB	DB5	DBS		
1.000E-01	2.500E-01	1.000E+00	1.480E-06	5.000E-04	3.13E-04		
QT	G	PI	PSI	TM	SPM		
5.000E-03	3.220E+01	3.142E+00	1.575E-01	1.000E+00	1.000E+00		
SEM	PEM	FPERM	FPFR	PM	BDI		
1.100E+00	1.000E-04	0.	0.	3.140E+05	1.591E+00		
JB	MF	MM	JJ	JK			
0	1	21	23	21			
TIME	ALPHA	OSCUH	OSCUH	BP	PSI		
0.	0.	0.	0.	3.874E+05	1.200E+02		
ZS	LT	DOT	SVCHN	SVCHN	SVCHIF	SVCHIF	TSS
0	25	1.10	1.700E-03	2.440E-07	-2.550E-11	4.440E-13	0.

SOM	SGM	PGM	FPERM	FPOR	PV	CARD NO.
1.000E+00	1.000E+00	0.000E+00	0.000E+00	0.000E+00	1.000E+00	2-11

JB	MO	MM	JJ	JK	CARD NO.
0	1	21	22	21	2-11

TIME	ALPHA	GMBD	OMCA	GMCA	OMCC	CARD NO.
0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	2-11

OVCON	GVCON	OVCOEF	GVCOEF	CARD NO.
0.000E+00	0.000E+00	0.000E+00	0.000E+00	2-11

IO	LT	DQT	CARD NO.
1	13	1.10	2-11

NOTE: In the following arrays, all values in the u-direction are given and alternate values in the z-direction are given.

OIL PRESSURE ARRAY

	0.	2.00	4.00	6.00	8.00	10.00	12.00	14.00	16.00	18.00	20.00
0.	386415.00	386344.14	386282.24	386215.43	386148.57	386081.72	386014.86	385948.01	385881.15	385814.29	385747.38
0.25	387245.31	387178.46	387111.61	387044.75	386977.90	386911.05	386844.19	386777.34	386710.51	386643.73	386576.91
0.50	387980.59	387913.74	387846.89	387780.03	387713.18	387646.33	387579.48	387512.63	387445.82	387379.02	387312.17
0.75	387940.85	387873.99	387807.14	387740.29	387673.44	387606.59	387539.73	387472.88	387406.04	387339.22	387272.47
1.00	387944.94	387878.08	387811.22	387744.37	387677.51	387610.65	387543.80	387476.94	387410.09	387343.24	387276.38
1.25	387949.03	387882.17	387815.31	387748.46	387681.60	387614.75	387547.89	387481.04	387414.19	387347.34	387280.48
1.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
1.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
2.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
2.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
2.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
2.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
3.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
3.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
3.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
3.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
4.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
4.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
4.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
4.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
5.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
5.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
5.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
5.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
6.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
6.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
6.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
6.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
7.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
7.25	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
7.50	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
7.75	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43
8.00	388000.00	387933.14	387866.29	387799.43	387732.57	387665.72	387598.86	387532.01	387465.15	387398.29	387331.43

NATURAL OIL VELOCITY

	1.00	3.00	5.00	7.00	9.00	11.00	13.00	15.00	17.00	19.00
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.25	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07	3.441E-07
0.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.75	2.1307E-04	2.124E-04	2.126E-04	2.127E-04	2.128E-04	2.129E-04	2.130E-04	2.131E-04	2.132E-04	2.133E-04
1.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.25	3.4248E-12	3.4220E-12	3.4192E-12	3.4164E-12	3.4136E-12	3.4108E-12	3.4080E-12	3.4052E-12	3.4024E-12	3.4000E-12
1.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.75	2.2181E-15	2.2160E-15	2.2138E-15	2.2117E-15	2.2095E-15	2.2073E-15	2.2052E-15	2.2031E-15	2.2010E-15	2.2000E-15
2.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.25	4.4441E-14	4.4477E-14	4.4513E-14	4.4549E-14	4.4585E-14	4.4621E-14	4.4657E-14	4.4693E-14	4.4729E-14	4.4765E-14
2.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.75	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20	3.2644E-20
3.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.25	0.	4.036E-20	2.042E-20	2.042E-20	1.0208E-19	8.1647E-20	0.	2.0402E-20	2.0397E-20	2.0392E-20
3.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.75	2.4741E-20	2.4785E-20	0.	-2.4779E-20	-2.4767E-20	-1.2320E-20	0.	-2.4749E-20	-4.9435E-21	-1.2368E-20
4.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.25	0.	7.516E-21	2.2391E-20	2.2391E-20	7.5104E-21	7.5090E-21	0.	7.5094E-21	2.2511E-20	5.0004E-20
4.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.75	0.	0.	0.	4.553E-21	4.553E-21	4.5594E-21	1.3680E-20	4.5511E-21	4.5511E-21	4.5511E-21
5.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.25	2.7638E-21	0.	-2.7644E-21	-5.5275E-21	0.	-2.7624E-21	-8.2652E-21	0.	2.7604E-21	0.
5.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.75	0.	-1.6771E-21	-1.6767E-21	-1.6763E-21	-3.3518E-21	-1.6739E-21	-1.6731E-21	-1.6747E-21	-1.6743E-21	-1.6739E-21
6.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.25	1.0175E-21	1.0172E-21	0.	0.	-1.0165E-21	-2.0325E-21	-1.0320E-21	-1.0157E-21	-5.0465E-21	-1.0619E-21
6.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.75	0.	6.1647E-22	0.	6.1648E-22	1.844E-21	6.1632E-22	6.1625E-22	6.1603E-22	-1.2319E-21	-2.4631E-21
7.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.25	2.7480E-22	0.	3.7412E-22	3.7403E-22	0.	0.	0.	0.	0.	0.
7.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.75	0.	0.	0.	-6.8059E-22	-6.8042E-22	0.	0.	4.5323E-22	2.2659E-21	2.0383E-21
8.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

OIL PERMEABILITY SURGE

[illegible]

VERTICAL OIL VELOCITY

[illegible]

RADIAL STD. VELOCITY DERIVATIVES

	1.00	3.00	5.00	7.00	9.00	11.00	13.00	15.00	17.00	19.00
0.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.25	5.0708E-07	5.0697E-07	5.0682E-07	5.0670E-07	5.0659E-07	5.0647E-07	5.0634E-07	5.0621E-07	5.0604E-07	5.0591E-07
0.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.75	5.1728E-09	5.1713E-09	5.1697E-09	5.1682E-09	5.1667E-09	5.1652E-09	5.1636E-09	5.1621E-09	5.1604E-09	5.1587E-09
1.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.25	5.0472E-14	5.0451E-14	5.0436E-14	5.0420E-14	5.0404E-14	5.0388E-14	5.0372E-14	5.0356E-14	5.0340E-14	5.0324E-14
1.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.75	1.1357E-17	9.7977E-18	1.1804E-17	1.0251E-17	1.0689E-17	1.0612E-17	9.7852E-18	9.7267E-18	1.4625E-17	4.7066E-17
2.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.25	4.4441E-14	3.5538E-20	4.4472E-14	4.4421E-14	1.0945E-18	3.3241E-14	4.4377E-14	4.4278E-14	4.4300E-14	5.5502E-14
2.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.75	5.7280E-14	5.0324E-14	2.3574E-19	2.0202E-14	4.0343E-14	1.3441E-14	0.	-3.3814E-14	2.4603E-14	2.3514E-14
3.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.25	4.4047E-14	5.3122E-14	3.0640E-14	1.8374E-14	4.2873E-14	8.1647E-20	0.	-3.4683E-14	1.9357E-14	3.4665E-14
3.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.75	7.4372E-20	2.7263E-14	4.4118E-20	2.4773E-20	2.7243E-14	1.8570E-14	0.	-7.4254E-20	1.4346E-14	8.6573E-20
4.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.25	4.0145E-20	5.2614E-20	3.7572E-20	1.4274E-14	6.7598E-20	6.7591E-20	0.	-4.7542E-20	1.4257E-14	1.5004E-20
4.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.75	3.6480E-20	0.	0.	-4.1010E-20	6.8332E-20	6.8317E-20	1.3660E-20	5.1940E-20	2.1553E-20	1.0001E-21
5.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.25	1.4360E-20	0.	-4.1466E-20	3.2105E-20	4.3373E-20	4.3373E-20	3.6049E-20	4.1777E-20	4.6437E-20	0.
5.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.75	0.	-2.5157E-20	1.1737E-20	2.5144E-20	3.6870E-20	2.5132E-20	2.6476E-20	2.5125E-20	1.1710E-20	1.6734E-21
6.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.25	1.0173E-21	-4.1350E-21	1.6272E-20	2.4440E-20	7.1194E-21	2.2857E-20	2.2352E-20	7.1102E-21	1.4246E-20	1.2183E-21
6.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.75	4.4370E-21	-1.0484E-20	0.	-3.5501E-21	1.1714E-20	-3.9474E-21	-4.3134E-21	1.0473E-20	1.3550E-20	2.4631E-21
7.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.25	2.6201E-21	-2.4437E-21	3.6011E-21	4.3384E-21	9.4746E-21	0.	0.	-2.4644E-21	4.4236E-21	8.4639E-21
7.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.75	3.6324E-21	-1.8158E-21	1.8133E-21	1.1343E-21	1.1340E-21	0.	-4.0674E-22	2.2664E-21	4.5316E-22	3.5511E-21
8.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

VERTICAL STD. VELOCITY DERIVATIVES

	1.00	3.00	5.00	7.00	9.00	11.00	13.00	15.00	17.00	19.00
0.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.25	-1.3244E-13	-1.3242E-13	-1.2408E-13	-1.2406E-13	-1.3272E-13	-1.3265E-13	-8.4460E-13	-8.4414E-13	3.2147E-10	3.1943E-10
0.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.75	1.8331E-13	1.8347E-13	1.2638E-13	1.2634E-13	1.8331E-13	1.8328E-13	7.0453E-13	7.0421E-13	2.5704E-10	2.5640E-10
1.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
1.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
2.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
2.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.75	1.1748E-13	1.1745E-13	1.1742E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
3.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
3.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
4.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
4.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
5.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
5.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
6.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.25	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
6.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
7.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.25	1.1748E-13	1.1745E-13	1.1742E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
7.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.75	1.1748E-13	1.1745E-13	1.1741E-13	1.1738E-13	1.1735E-13	1.1732E-13	1.1729E-13	1.1726E-13	1.1744E-13	1.1741E-13
8.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

[illegible]

1.00 17.00 19.00

[illegible]

GAS PRESSURE ARRAY

	0.	2.00	4.00	6.00	8.00	10.00	12.00	14.00	16.00	18.00	20.00
0.	347553.02	347448.18	347343.34	347238.50	347133.66	347028.82	346923.98	346819.14	346714.30	346609.46	346504.62
0.25	347552.30	347447.45	347342.61	347237.77	347132.93	347028.09	346923.25	346818.41	346713.57	346608.73	346503.89
0.50	347551.58	347446.73	347341.89	347237.05	347132.21	347027.37	346922.53	346817.69	346712.85	346608.01	346503.17
0.75	347550.86	347446.01	347341.17	347236.33	347131.49	347026.65	346921.81	346816.97	346712.13	346607.29	346502.45
1.00	347550.14	347445.29	347340.45	347235.61	347130.77	347025.93	346921.09	346816.25	346711.41	346606.57	346501.73
1.25	347549.42	347444.57	347339.73	347234.89	347130.05	347025.21	346920.37	346815.53	346710.69	346605.85	346501.01
1.50	347548.70	347443.85	347339.01	347234.17	347129.33	347024.49	346919.65	346814.81	346709.97	346605.13	346500.29
1.75	347547.98	347443.13	347338.29	347233.45	347128.61	347023.77	346918.93	346814.09	346709.25	346604.41	346499.57
2.00	347547.26	347442.41	347337.57	347232.73	347127.89	347023.05	346918.21	346813.37	346708.53	346603.69	346498.85
2.25	347546.54	347441.69	347336.85	347232.01	347127.17	347022.33	346917.49	346812.65	346707.81	346602.97	346498.13
2.50	347545.82	347440.97	347336.13	347231.29	347126.45	347021.61	346916.77	346811.93	346707.09	346602.25	346497.41
2.75	347545.10	347440.25	347335.41	347230.57	347125.73	347020.89	346916.05	346811.21	346706.37	346601.53	346496.69
3.00	347544.38	347439.53	347334.69	347229.85	347125.01	347020.17	346915.33	346810.49	346705.65	346600.81	346495.97
3.25	347543.66	347438.81	347333.97	347229.13	347124.29	347019.45	346914.61	346809.77	346704.93	346599.99	346495.25
3.50	347542.94	347438.09	347333.25	347228.41	347123.57	347018.73	346913.89	346809.05	346704.21	346599.27	346494.53
3.75	347542.22	347437.37	347332.53	347227.69	347122.85	347018.01	346913.17	346808.33	346703.49	346598.55	346493.81
4.00	347541.50	347436.65	347331.81	347226.97	347122.13	347017.29	346912.45	346807.61	346702.77	346597.83	346493.09
4.25	347540.78	347435.93	347331.09	347226.25	347121.41	347016.57	346911.73	346806.89	346702.05	346597.11	346492.37
4.50	347540.06	347435.21	347330.37	347225.53	347120.69	347015.85	346911.01	346806.17	346701.33	346596.39	346491.65
4.75	347539.34	347434.49	347329.65	347224.81	347119.97	347015.13	346910.29	346805.45	346700.61	346595.67	346490.93
5.00	347538.62	347433.77	347328.93	347224.09	347119.25	347014.41	346909.57	346804.73	346699.89	346594.95	346490.21
5.25	347537.90	347433.05	347328.21	347223.37	347118.53	347013.69	346908.85	346804.01	346699.17	346594.23	346489.49
5.50	347537.18	347432.33	347327.49	347222.65	347117.81	347012.97	346908.13	346803.29	346698.45	346593.51	346488.77
5.75	347536.46	347431.61	347326.77	347221.93	347117.09	347012.25	346907.41	346802.57	346697.73	346592.79	346488.05
6.00	347535.74	347430.89	347326.05	347221.21	347116.37	347011.53	346906.69	346801.85	346697.01	346592.07	346487.33
6.25	347535.02	347430.17	347325.33	347220.49	347115.65	347010.81	346905.97	346801.13	346696.29	346591.35	346486.61
6.50	347534.30	347429.45	347324.61	347219.77	347114.93	347010.09	346905.25	346800.41	346695.57	346590.63	346485.89
6.75	347533.58	347428.73	347323.89	347219.05	347114.21	347009.37	346904.53	346799.69	346694.85	346589.91	346485.17
7.00	347532.86	347428.01	347323.17	347218.33	347113.49	347008.65	346903.81	346798.97	346694.13	346589.19	346484.45
7.25	347532.14	347427.29	347322.45	347217.61	347112.77	347007.93	346903.09	346798.25	346693.41	346588.47	346483.73
7.50	347531.42	347426.57	347321.73	347216.89	347112.05	347007.21	346902.37	346797.53	346692.69	346587.75	346483.01
7.75	347530.70	347425.85	347321.01	347216.17	347111.33	347006.49	346901.65	346796.81	346691.97	346587.03	346482.29
8.00	347530.00	347425.13	347320.29	347215.45	347110.61	347005.77	346900.93	346796.09	346691.25	346586.31	346481.57

CAPILLARY PRESSURE ARRAY

	0.	2.00	4.00	6.00	8.00	10.00	12.00	14.00	16.00	18.00	20.00
0.	11137.02	11137.02	11137.02	11137.13	11137.13	11137.13	11137.16	11137.16	11137.26	11137.26	11137.26
0.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
0.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
0.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
1.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
1.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
1.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
1.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
2.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
2.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
2.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
2.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
3.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
3.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
3.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
3.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
4.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
4.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
4.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
4.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
5.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
5.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
5.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
5.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
6.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
6.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
6.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
6.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
7.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
7.25	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
7.50	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
7.75	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02
8.00	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02	11137.02

VERTICAL GAS VELOCITY

VERTICAL GAS VELOCITY

GAS PERMEABILITY ARRAY

[illegible]

GAS SOLUBILITY ARMY

[illegible]

RADIAL GAS VELOCITY DERIVATIVES

	1.00	3.00	5.00	7.00	9.00	11.00	13.00	15.00	17.00	19.00
0.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.25	1.2822E-45	1.2823E-45	4.2472E-21	4.2453E-21	4.2434E-21	4.2414E-21	1.2465E-20	1.2418E-20	7.0594E-21	4.1041E-21
0.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.75	0.	0.	1.1492E-24	1.1484E-24	1.1476E-24	1.1468E-24	3.2467E-24	3.2493E-24	4.4553E-25	4.1150E-24
1.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	2.1848E-27
1.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	2.5353E-43
2.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	-8.9714E-46
3.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.5278E-50
3.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	-1.6511E-49
4.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	-4.6230E-50
4.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	-3.0114E-51
5.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	4.8591E-51
5.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	8.9379E-52
6.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	-6.5053E-51
6.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	-1.3152E-51
7.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	-4.7864E-51
7.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
8.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	-2.0563E-51

VERTICAL GAS VELOCITY DERIVATIVES

	1.00	3.00	5.00	7.00	9.00	11.00	13.00	15.00	17.00	19.00
0.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.25	0.	3.6916E-25	2.5919E-24	1.2494E-27	-1.2852E-27	2.3534E-24	3.3553E-24	-2.0691E-24	9.4172E-25	6.6965E-24
0.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.00	0.	0.	0.	0.	0.	0.	0.	0.	3.8897E-25	1.2063E-23
1.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.4034E-38
2.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.6503E-37
2.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2729E-40
3.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3099E-39
3.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
4.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
4.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
4.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
5.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
5.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
5.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
6.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
6.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
7.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
7.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.2924E-40
7.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.3610E-39
8.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

OIL AND GAS RESERVOIR ENGINEERING CODE

TIME SUMMARY OF PRODUCTION

25

TS	QU	QG	QOG	QT	GUR	DMU	DMS
1.132E+02	3.679E-02	1.373E-07	2.167E-03	5.337E+00	3.356E-01	-2.221E+06	4.241E+06
1.140E+02	3.667E-02	1.394E-07	2.160E-03	5.396E+00	3.391E-01	-2.196E+06	4.198E+06
1.148E+02	3.656E-02	1.414E-07	2.153E-03	5.454E+00	3.425E-01	-2.173E+06	4.155E+06
1.156E+02	3.645E-02	1.435E-07	2.147E-03	5.513E+00	3.460E-01	-2.150E+06	4.114E+06
1.164E+02	3.634E-02	1.455E-07	2.140E-03	5.571E+00	3.494E-01	-2.127E+06	4.074E+06
1.172E+02	3.623E-02	1.474E-07	2.134E-03	5.629E+00	3.528E-01	-2.105E+06	4.034E+06
1.180E+02	3.612E-02	1.494E-07	2.127E-03	5.687E+00	3.562E-01	-2.084E+06	3.995E+06
1.188E+02	3.601E-02	1.513E-07	2.121E-03	5.745E+00	3.596E-01	-2.063E+06	3.958E+06
1.196E+02	3.591E-02	1.533E-07	2.115E-03	5.803E+00	3.630E-01	-2.042E+06	3.921E+06
1.204E+02	3.581E-02	1.551E-07	2.109E-03	5.860E+00	3.664E-01	-2.022E+06	3.885E+06
1.212E+02	3.570E-02	1.570E-07	2.103E-03	5.917E+00	3.698E-01	-2.003E+06	3.849E+06
1.220E+02	3.560E-02	1.589E-07	2.097E-03	5.974E+00	3.731E-01	-1.984E+06	3.814E+06
1.228E+02	3.550E-02	1.607E-07	2.091E-03	6.031E+00	3.765E-01	-1.965E+06	3.778E+06
1.236E+02	3.540E-02	1.625E-07	2.085E-03	6.088E+00	3.799E-01	-1.947E+06	3.747E+06
1.244E+02	3.531E-02	1.643E-07	2.079E-03	6.145E+00	3.832E-01	-1.929E+06	3.715E+06
1.252E+02	3.521E-02	1.661E-07	2.074E-03	6.201E+00	3.865E-01	-1.911E+06	3.683E+06
1.260E+02	3.511E-02	1.678E-07	2.068E-03	6.257E+00	3.898E-01	-1.894E+06	3.651E+06
1.268E+02	3.502E-02	1.696E-07	2.062E-03	6.314E+00	3.931E-01	-1.877E+06	3.621E+06
1.276E+02	3.493E-02	1.713E-07	2.057E-03	6.370E+00	3.964E-01	-1.861E+06	3.590E+06
1.284E+02	3.484E-02	1.730E-07	2.052E-03	6.425E+00	3.997E-01	-1.844E+06	3.561E+06
1.292E+02	3.474E-02	1.747E-07	2.046E-03	6.481E+00	4.030E-01	-1.829E+06	3.532E+06
1.300E+02	3.466E-02	1.765E-07	2.041E-03	6.537E+00	4.063E-01	-1.813E+06	3.505E+06
1.308E+02	3.457E-02	1.780E-07	2.036E-03	6.592E+00	4.095E-01	-1.798E+06	3.478E+06
1.316E+02	3.448E-02	1.796E-07	2.030E-03	6.647E+00	4.128E-01	-1.783E+06	3.448E+06
1.324E+02	3.439E-02	1.812E-07	2.025E-03	6.702E+00	4.160E-01	-1.768E+06	3.417E+06

OIL SATURATIONS AND MASS BALANCE CALCULATIONS

0	2 18	-8.1074453E-05-1.9350552E-07-5.4164130E-06	9.989979E-01	2.000000E-01	1.6020253E+00-2.6776482E-06	9.996757E-01
0	2 20	-8.0774304E-05-1.9347056E-07-5.4248829E-06	9.9899524E-01	2.000000E-01	1.6019271E+00-2.6769944E-06	9.996769E-01
0	4 3343879E+09	3.119304231E+08	1.185154971E+07	-1.423283395E+06	6.572442996E+07	MASS BALANCE CALCULATIONS
0	2 20	-8.0774304E-05-1.9347056E-07-5.4248829E-06	9.9899524E-01	2.000000E-01	1.6019271E+00-2.6769944E-06	9.996769E-01
0	2 2	-8.0449453E-05-1.654679E-07-5.4426242E-06	9.9903298E-01	2.000000E-01	1.6026628E+00-2.6949520E-06	9.996462E-01
0	2 4	-8.7788075E-05-1.9488762E-07-5.4228278E-06	9.9902312E-01	2.000000E-01	1.6027481E+00-2.6902399E-06	9.996489E-01
0	2 6	-8.7182427E-05-1.9434417E-07-5.4071168E-06	9.990061E-01	2.000000E-01	1.6026553E+00-2.689656E-06	9.996513E-01
0	2 8	-8.6627280E-05-1.9386398E-07-5.3928031E-06	9.9900220E-01	2.000000E-01	1.6025242E+00-2.6821612E-06	9.996535E-01
0	2 10	-8.6034380E-05-1.9352768E-07-5.3793574E-06	9.9899409E-01	2.000000E-01	1.6024114E+00-2.6779551E-06	9.996559E-01
0	2 12	-8.542684E-05-1.9277383E-07-5.3621571E-06	9.989854E-01	2.000000E-01	1.6022982E+00-2.6736206E-06	9.996583E-01
0	2 14	-8.4931110E-05-1.9235341E-07-5.3527126E-06	9.9897714E-01	2.000000E-01	1.6021888E+00-2.6702233E-06	9.996605E-01
0	2 16	-8.4411851E-05-1.9191804E-07-5.3412632E-06	9.9897072E-01	2.000000E-01	1.6020786E+00-2.6667282E-06	9.996624E-01
0	2 18	-8.3967199E-05-1.9158314E-07-5.3424287E-06	9.9896305E-01	2.000000E-01	1.6019725E+00-2.6639389E-06	9.996641E-01
0	2 20	-8.3727861E-05-1.9155340E-07-5.3515027E-06	9.9895957E-01	2.000000E-01	1.6018744E+00-2.6633237E-06	9.996651E-01

OGRE

TIME = 2.0000E+01 P0(MIN) = 3.7306E+05
P0(MAX) = 3.8800E+05

OGRE

TIME = 2.0000E+01

OGRE

TIME = 2.0000E+01 PG(MIN) = 3.8423E+05
PG(MAX) = 3.9914E+05

OGRE

TIME = 2.0000E+01 PC(MIN) = 1.1137E+04
PC(MAX) = 1.1156E+04

OGRE

TIME = 2.0000E+01 SO (MIN) = 9.9868E-01
SO (MAX) = 1.0001E+00

OGRE

TIME = 2.0000E+01

OGRE

MASS BALANCE CHECK

OGRE

PRODUCTION RATES

OGRE

GAS PRODUCTION RATES

OGRE

CUMULATIVE PRODUCTION

Appendix E

Fortran Code Listing

C OGRE CODE LISTING

* LIST8
* CARDS COLUMN
* FORTRAN SMART
SUBROUTINE SMART

C
C ASSIGN COMMON AND DIMENSION TO ALL SUBROUTINES
C

```
CLICHE STUFF
COMMON PD,PC,PG,SD,SG,VDU,VOZ,VGU,VGZ,PKD,PKG,VISU,VISG,BD,BG,
1RS,THETA,ZETA,ETA,TAU,C,A,DVOU,DVOZ,DVU,DVGZ,ZLAM,RHO,IUZ,EU,V,
2ICS,DELTA,QT,QO,QG,QOCS,QGCS,QOSTB,GOR,CK,PSI,G,GAMO,GAMG,
3SOI,AK,PHI,POI,DBO,DBG,DRS,B,PI,OU,DZ,Z,U,PCN,SON,F,SOM,SGH,
4PGM,ICT,IQPT,NP,NT,JZ,JJ,JK,PERM,POR,JB,FPERM,FPOR,JC,PW,MO,MM,
5E1,E2,E3,E4,E5,E6,E7,E8,E9,ICOUNT,ALPHA,R,ICON,W
6,VMAX,AVOU,AVOZ,AVGU,AVGZ,POHIN,POMAX,PCM IN,PCMAX,SOMIN,SUMAX,
7PGMIN,PGMAX,PN,C4,IT,AT,BT,CT,VR,ASON
BANK/(1)
COMMON/1267/
1RW,DT,TIME,TERM,QOG,JO,JP,JL,JI,RES,X,Y,D,AA,BB,CC,DD,EE,FF,
2JM,JN,JT,JS,QOS,QGS,QOQS,GORS,QTS,TS,QOCUM,QGCUM,E,H,E10,E11,
3OM,GM,OMT,GMT,OMI,GMI,RJ,RJJ,OMBA,GMBA,OMBB,GMBB,OMBC,GMBC,
4OMBD,GMBD,OMCA,OMCB,GMCA,GMCB,OMCC,GMCC,OMCD,GMCD,DELMO,DELMG,
5DEV,DEVG,IG,IZ,J,K,L,KL,E12,E13,E14,E15,E16,LA,LB,N,GA,GB,
6GC,GD,GE,GF,GG,GH,QX,QY,QZ,QOSC,QGSC,PCO,DVCON,DVCOEF,GVCON,GVCOEF
7,DPRO,DPRG,TK1,TK2,ORS,OMUS,OMGS
DIMENSION PO(33,21),PON(33,21),SO(33,21),SON(33,21),VOU(33,21),
*VOZ(33,21),VGU(33,21),VGZ(33,21),R(33),Z(33),PG(33,21),PC(33,21),
*SG(33,21),PKD(33,21),PKG(33,21),BD(33,21),BG(33,21),RS(33,21),
*VISO(33,21),VISG(33,21),AI(33,3),BI(33,5),XI(33),YI(33),RESI(33),DI(33),
*QO(21),QG(21),QOG(21),PERM(33,21),PUR(33,21),QOS(500),QGS(500),
*QOQS(500),GORS(500),QTS(500),TS(500),U(33),DVOU(33,21),
*DVOZ(33,21),DVGU(33,21),DVGZ(33,21),E1(33,21),E2(33,21),
*E3(33,21),E4(33,21),E5(33,21),E6(33,21),E7(33,21),E8(33,21),
*ETA(33,21),TAU(33,21),E(33,21),H(33),C4(33,21),OMOS(500),ORS(21)
*OMGS(500)
DIMENSION
*E9(33,21),E10(33,21),E11(33,21),E12(33,21),E13(33,21),E14(33,21),
*E15(33,21),E16(33,21),C(33,21),ZETA(33,21),THETA(33,21),W(33)
*,AVOU(33,21),AVOZ(33,21),AVGU(33,21),AVGZ(33,21),PN(33,21)
*,PCO(33,21),OPRO(7,21),DPRG(7,21),TK1(33,21),TK2(33,21)
BANK COEFZ (1)
END CLICHE
USE STUFF
END
```

* LIST8
* CARDS COLUMN
* FORTRAN OGRE

OIL AND GAS RESERVOIR ENGINEERING CODE

DEFINITION OF SYMBOLS

C

C	A	2A	MATRIX COEFFICIENT ARRAY
C	AK		SATURATED MEDIUM PERMEABILITY
C	ALPHA		ANGLE IN RADIAN FROM THE HORIZONTAL
C	ASON		SATURATION - 1.0
C	AVGU	2A	ABSOLUTE VALUE OF VGU
C	AVGZ	2A	ABSOLUTE VALUE OF VGZ
C	AVOU	2A	ABSOLUTE VALUE OF VOJ
C	AVOZ	2A	ABSOLUTE VALUE OF VOZ
C	B	2A	MATRIX VARIABLE
C	BG	2A	GAS FORMATION FACTOR
C	BO	2A	OIL FORMATION FACTOR
C	BOI		INITIAL OIL FORMATION FACTOR
C	BP		OIL BUBBLE POINT
C	C		COEFFICIENT OF OIL PRESSURE EQUATION
C	CK		KOZENY CONSTANT
C	COEFU		U-DIRECTION COEFFICIENT SUBROUTINE
C	COEFZ		Z-DIRECTION COEFFICIENT ENTRY POINT
C	CS		=ICS
C	D	1A	MATRIX VARIABLE
C	DELMG		DEVIATION IN GAS CONSERVATION
C	DELMO		DEVIATION IN OIL CONSERVATION
C	DELTA		CONTROL NUMBER FOR GEOMETRY
C	DEVG		FRACTIONAL ERROR IN GAS CONSERVATION
C	DEVO		FRACTIONAL ERROR IN OIL CONSERVATION
C	DBG		GRADIENT OF GAS FORMATION FACTOR
C	DBO		GRADIENT OF OIL FORMATION FACTOR
C	DP		DELTA PO
C	DPRG		ANALYTICAL DPG/DU
C	DPRO		ANALYTICAL DPO/DU
C	DQT		DELTA QO
C	DRS		GRADIENT OF GAS SOLUTION FACTOR
C	DT		TIME INCREMENT
C	DU		INCREMENT IN HORIZONTAL DIRECTION
C	DVGU	2A	SPACE DERIVATIVE OF RADIAL GAS VELOCITY
C	DVGZ	2A	SPACE DERIVATIVE OF VERTICAL GAS VELOCITY
C	DVOU	2A	SPACE DERIVATIVE OF RADIAL OIL VELOCITY
C	DVOZ	2A	SPACE DERIVATIVE OF VERTICAL OIL VELOCITY
C	DZ		INCREMENT IN VERTICAL DIRECTION
C	E	2A	TEMPORARY COEFFICIENT ARRAY
C	ETA		PRESSURE EQUATION COEFFICIENT
C	EU		U-DIRECTION SCALING PARAMETER
C	F		COEFFICIENT OF DZ
C	FPERM		PERMEABILITY FACTOR
C	FPOR		POROSITY FACTOR
C	G		GRAVITATIONAL CONSTANT
C	GAMG		GAS DENSITY AT STP
C	GAMO		OIL DENSITY AT STP
C	GM		INITIAL MASS OF GAS
C	GMBA		PARAMETER FOR CHOOSING DT
C	GMBB		NUMBER OF STEPS TO LOWER WELL PRESSURE
C	GMBC		INPUT VALUE OF EPSIL
C	GMBD		INPUT VALUE OF BP
C	GMCA		INPUT VALUE OF BCI
C	GMI		=GMT
C	GMT		MASS OF GAS AT T=N
C	GOR		GAS-OIL RATIO
C	GORS	1A	TIME SUMMARY OF GOR
C	GVCOEF		GAS VISCOSITY COEFFICIENT
C	GVCON		GAS VISCOSITY CONSTANT
C	H	2A	TEMPORARY COEFFICIENT ARRAY
C	ICON		NUMBER OF CONTOURS TO BE PLOTTED
C	ICOUNT		CURRENT SIZE OF MATRIX ARRAY

C	ICS		TEST FOR U-DIRECTION SCALING
C	ICT		CARD OR TAPE INPUT
C	IG		TEST FOR ZONE GEOMETRY
C	IOPT		BOUNDARY CONDITION OPTION
C	IT		NUMBER OF CYCLES BETWEEN NEW DT
C	ITER		MATRIX ITERATION COUNTER
C	IUZ		TEST FOR ENTRY POINT
C	J		MATRIX INDEX U-DIRECTION
C	JB		BOUNDARY INDEX FOR SPACE ZONING
C	JC		JB+1
C	JL		JK/2
C	JJ		TOTAL NO. OF GRID POINTS ON U AXIS
C	JK		TOTAL NO. OF GRID POINTS ON Z AXIS
C	JL		JJ/2
C	JM		JJ-1
C	JN		JK-1
C	JO		JJ-2
C	JP		JK-2
C	JS		TIME SUMMARY INDEX
C	JT		TEST FOR TIME SUMMARY
C	JZ		TEST FOR SPACE ZONING
C	K		MATRIX INDEX Z-DIRECTION
C	KL		CYCLE COUNTER
C	L		PRINTOUT CYCLE COUNTER
C	LA		MC-2
C	LB		MM+2
C	LSPIN		TEST ON CONSTANT QD ITERATION
C	LT		ITERATION INDEX FOR DT INCREASE
C	MATRIX		VON HOLT SUBROUTINE
C	MM		UPPER WELL BOUNDARY
C	MO		LOWER WELL BOUNDARY
C	N		OUTPUT CYCLE INDEX
C	NP		TEST FOR PRINT CYCLE NUMBER
C	NT		TEST FOR TIME CYCLE NUMBER
C	ODT		=DT
C	QGBAL		MASS CHECK SUBROUTINE
C	OM		INITIAL MASS OF OIL
C	OMBA		LOCATION OF INPUT DELTA
C	OMBB		TEST FOR MATRIX COEFFICIENT OUTPUT
C	OMBC		INPUT VAE OF DP
C	OMCC		INPUT VAE OF SBO
C	OMT		MASS OF OIL AT T=N
C	ORS		=RS
C	OUTPUT		MAIN OUTPUT SUBROUTINE
C	OVCDEF		OIL VISCOSITY CCEFFICIENT
C	OVCON		OIL VISCOSITY CONSTANT
C	PC	2A	CAPILLARY PRESSURE
C	PCMAX		MAXIMUM VALUE OF PC ARRAY
C	PCMIN		MINIMUM VALUE OF PC ARRAY
C	PCO		=PC
C	POIF		POI + OR - FACTOR
C	PERM	2A	SPACE DEPENDENT MEDIUM PERMEABILITY
C	PG	2A	GAS PRESSURE
C	PGM		INITIAL GAS PRESSURE
C	PGMAX		MAXIMUM VALUE OF PG ARRAY
C	PGMIN		MINIMUM VALUE OF PG ARRAY
C	PHI		MEDIUM POROSITY
C	PI		3.1415926
C	PKG	2A	EFFECTIVE GAS PERMEABILITY
C	PKO	2A	EFFECTIVE OIL PERMEABILITY
C	PN	2A	TIME INTERPOLATION OF PO AND PON
C	PO	2A	OIL PRESSURE
C	POI		INITIAL OIL PRESSURE

C	POR	2A	SPACE DEPENDENT MEDIUM POROSITY
C	POMAX		MAXIMUM VALUE OF PO ARRAY
C	POMIN		MINIMUM VALUE OF PO ARRAY
C	PCN	2A	NEW OIL PRESSURE
C	PQCON		SUBROUTINE FOR VELOCITIES, ETC.
C	PSI		OIL SURFACE TENSION
C	PW		PRESCRIBED WELL PRESSURE
C	QD		$=QT/2$
C	QG	1A	GAS MASS FLOW RATE
C	QGCUM		CUMULATIVE GAS PRODUCTION
C	QGS	1A	TIME SUMMARY OF QG
C	QGSC		GAS VOLUME FLOW RATE
C	QO	1A	OIL MASS FLOW RATE
C	QOCUM		CUMULATIVE OIL PRODUCTION
C	QOG	1A	DISSOLVED GAS MASS FLOW RATE
C	QOGS	1A	TIME SUMMARY OF QOG
C	QOS	1A	TIME SUMMARY OF QO
C	QOSC		OIL VOLUME FLOW RATE
C	QOSTB		OIL VOLUME FLOW RATE IN STOCK TANK BBLs.
C	QP		$=QT*TIME/GMBB$
C	QQ		TEST VALUE OF QO
C	QT		TOTAL MASS FLOW RATE
C	QTS	1A	TIME SUMMARY OF QT
C	QX		SUMMATION OF OIL MASS FLOW RATE
C	QY		SUMMATION OF DISSOLVED GAS FLOW RATE
C	QZ		SUMMATION OF GAS MASS FLOW RATE
C	RES	1A	MATRIX RESIDUAL
C	RHD		U-DIRECTION PARAMETER
C	RJ		DIFFERENCE OF THE SQUARES OF THE RADII
C	RS	2A	GAS SOLUBILITY FACTOR
C	RSI		INITIAL GAS SATURATION
C	RW		WELL RADIUS
C	SG	2A	GAS SATURATION
C	SGM		INITIAL GAS SATURATION
C	SO	2A	OIL SATURATION
C	SOI		IRREDUCIBLE OIL SATURATION
C	SOM		INITIAL OIL SATURATION
C	SOMAX		MAXIMUM VALUE OF SO ARRAY
C	SGMIN		MINIMUM VALUE OF SO ARRAY
C	SON	2A	NEW OIL SATURATION
C	SRS		SINGLE PHASE GAS SATURATION
C	TAU		PRESSURE EQUATION COEFFICIENT
C	TERM		TERM FROM PKO AND PKG EQUATIONS
C	THETA		PRESSURE EQUATION COEFFICIENT
C	TIME		ELAPSED REAL TIME
C	TOLER		$=EPSIL*TIME/GMBB$
C	TS	1A	TIME SUMMARY OF TIME
C	U	1A	RADIAL OR HORIZONTAL COORDINATE
C	VGU	2A	RADIAL GAS VELOCITY
C	VGZ	2A	VERTICAL GAS VELOCITY
C	VISG	2A	GAS VISCOSITY
C	VISO	2A	OIL VISCOSITY
C	VMAX		MAXIMUM VELOCITY
C	VOU	2A	RADIAL OIL VELOCITY
C	VOZ	2A	VERTICAL OIL VELOCITY
C	W	1A	ARRAY OF CONTOUR LEVELS
C	X	1A	MATRIX VALUE OF OIL PRESSURE
C	Y	1A	RIGHT-HAND SIDE OF MATRIX EQUATION
C	Z	1A	VERTICAL COORDINATE
C	ZD		$=JK/2$
C	ZETA		PRESSURE EQUATION COEFFICIENT
C	ZLAM		Z-DIRECTION PARAMETER

C
C
C
C
C

OGRE MAIN CODE

USE STUFF

C
C
C

READ CONTROL PARAMETERS

```

READ INPUT TAPE 2,1200,IOPT,ICT,NP,NT,JZ,JT,IG,IUZ,ICS,IT,GMBB,
*DMBC,GMBC,OMBD,
*RHO,ZLAM,DELTA,ICON,GMBA,OMBB,PDIF
IF (ICT) 12,11,12
12 CONTINUE
REWIND 6
READ TAPE 6,CK,PHI,AK,GAMO,GAMG,POI,SDI,B,DU,DZ,DBD,DBG,DRS,QT,
*G,PI,PSI,RW,SOM,SGM,PGM,FPERM,FPOR,PH,JB,MO,MM,JJ,JK,
*TIME,QOCUM,QGCUM,ALPHA,GMBD,GMCA,OMCA,OMCC,OMBA,OVCEN,GVCEN,
*OVCOEF,GVCOEF,KL,CM,GM
GO TO 13
11 CONTINUE
READ INPUT TAPE 2,1201,CK,PHI,AK,GAMO,GAMG,POI,SDI,DU,DZ,DBD,
*DBG,DRS,QT,G,PI,PSI,RW,SOM,SGM,PGM,FPERM,FPOR,PH,JB,MO,MM,JJ,JK,
*TIME,ALPHA,GMBD,OMCA,GMCA,OMCC,OVCEN,GVCEN,OVCOEF,GVCOEF

```

C
C
C

ESTABLISH GRID AND SET INITIAL INDEXES

```

BP=GMBD
RSI=OMCA
BOI=GMCA
SRS=0.0
DEVO=0.0
DEVG=0.0
DELMG=0.0
DELMG=0.0
F=0.0
VB=0.0
ASGN=0.0
JM=JJ-1
JN=JK-1
JQ=JJ-2
JP=JK-2
IF (ICS) 42,43,42
42 CONTINUE
U(1)=0.0
GO TO 44
43 CONTINUE
U(1)=1.0
44 CONTINUE
Z(1)=0.0
DO 26 J=2,JJ
U(J)=U(J-1)+DU
26 CONTINUE
DO 27 K=2,JK
Z(K)=Z(K-1)+DZ
27 CONTINUE
QOCUM=0.0
QGCUM=0.0
BP=GMBD
RSI=OMCA
BOI=GMCA
GA=DBD*BP+1.04
SRS=0.0

```

```

JM=JJ-1
JN=JK-1
JO=JJ-2
JP=JK-2
DO 18 K=1,JK
DO 18 J=1,JJ
PERM(J,K)=AK
POR(J,K)=PHI
18 CONTINUE
C
C      TEST FOR ZONING OF KA AND PHI
C
IF(JZ)16,17,16
17 CONTINUE
GO TO 19
16 CONTINUE
GO TO(33,34,41,101),IG
33 CONTINUE
DO 20 K=1,JK
DO 21 J=1,JB
PERM(J,K)=AK*FPERM
POR(J,K)=PHI*FPOR
21 CONTINUE
20 CONTINUE
JC=JB+1
DO 22 K=1,JK
DO 23 J=JC,JJ
PERM(J,K)=AK
POR(J,K)=PHI
23 CONTINUE
22 CONTINUE
GO TO 19
101 CONTINUE
DO 120 K=1,JK,1
DO 119 J=1,JJ,1
IF(ICS)102,103,102
102 CONTINUE
RPRW=RW*EXP(U(J))
GO TO 104
103 CONTINUE
RPRW=RW*U(J)
104 CONTINUE
DKA=SQRT(U(JB)**2+RPRW**2)
IF(DKA-425.0)105,105,119
105 CONTINUE
PERM(J,K)=4.84E-12*EXP(-2.65E-02*DKA)
119 CONTINUE
120 CONTINUE
GO TO 19
34 CONTINUE
DO 36 J=1,JB
ZA=SQRT(U(JB)**2-U(J)**2)
IZ=ZA/DZ+1.5
DO 35 K=1,IZ
PERM(J,K)=AK*FPERM
POR(J,K)=PHI*FPOR
35 CONTINUE
36 CONTINUE
GO TO 19
41 CONTINUE
READ INPUT TAPE 2,1204,((PERM(J,K),POR(J,K),J=1,JJ),K=1,JK)
19 CONTINUE

```

C
C
C
C

GENERATE INITIAL CONDITIONS
DETERMINE ORIENTATION

```

IF(SINF(ALPHA))201,202,201
202 CONTINUE
DO 32 J=1,JJ
IF(J-J8)74,75,75
74 CONTINUE
PO(J,1)=POI
GO TO 76
75 CONTINUE
PO(J,1)=POI+PDIF
76 CONTINUE
IF(POI-BP)46,46,47
47 CONTINUE
RS(J,1)=RSI
BO(J,1)=BOI
GO TO 60
46 CONTINUE
BO(J,1)=DBO*POI+1.04
RS(J,1)=DRS*POI
60 CONTINUE
PC(J,1)=PSI*SQRTF(PHI/(CK*AK))*0.01
PG(J,1)=PO(J,1)+PC(J,1)
SO(J,1)=SOM
BG(J,1)=DBG*PG(J,1)+1.00
VQU(J,1)=0.0
VQZ(J,1)=0.0
VQU(J,1)=0.0
VGZ(J,1)=0.0
DVQU(J,1)=0.0
DVQU(J,1)=0.0
DVQZ(J,1)=0.0
DVQZ(J,1)=0.0
PON(J,1)=PO(J,1)
TERM=(SO(J,1)-SOI)/(1.0-SOI)
PKO(J,1)=PERM(J,1)*SO(J,1)**3*SQRTF(TERM)
PKG(J,1)=PERM(J,1)*(1.0-TERM)*(1.0-TERM**.25*SQRTF(SO(J,1)))**2
32 CONTINUE
GO TO 203
201 CONTINUE
DO 204 K=1,JK,1
PO(1,K)=POI
IF(POI-BP)208,208,209
209 CONTINUE
RS(1,K)=RSI
BO(1,K)=BOI
GO TO 210
208 CONTINUE
RS(1,K)=DRS*POI
BO(1,K)=DBO*POI+1.04
210 CONTINUE
PC(1,K)=PSI*SQRTF(PHI/(CK*AK))*0.01
PG(1,K)=PO(1,K)+PC(1,K)
SO(1,K)=SOM
BG(1,K)=DBG*PG(1,K)+1.0
VQU(1,K)=0.0
VQU(1,K)=0.0
VQZ(1,K)=0.0
VGZ(1,K)=0.0
DVQU(1,K)=0.0
DVQU(1,K)=0.0

```



```

DVOZ(1,K)=0.0
DVGZ(1,K)=0.0
PON(1,K)=PO(1,K)
TERM=(SO(1,K)-SOI)/(1.0-SOI)
PKO(1,K)=PERM(1,K)*SO(1,K)**3*SQRTF(TERM)
PKG(1,K)=PERM(1,K)*(1.0-TERM)*(1.0-TERM**25*SQRTF(SO(1,K)))*2
204 CONTINUE
203 CONTINUE
IF(SINF(ALPHA))212,211,212
211 CONTINUE
DO 14 K=2,JK
DO 15 J=1,JJ
PC(J,K)=PSI*SQRTF(PHI/(CK*AK))*0.01
VQU(J,K)=0.0
VQZ(J,K)=0.0
VQU(J,K)=0.0
VGZ(J,K)=0.0
DVCU(J,K)=0.0
DVGU(J,K)=0.0
DVOZ(J,K)=0.0
DVGZ(J,K)=0.0
PO(J,K)=PO(J,K-1)-(GAMO+GAMG*RS(J,K-1))*G*DZ/BO(J,K-1)
DO 38 LN=1,30
LN=LN
IF(PO(J,K)-BP)48,48,49
49 CONTINUE
RS(J,K)=RSI
IF(GAMG)71,70,71
70 CONTINUE
BG(J,K)=DBG*PO(J,K)+1.0
BO(J,K)=1.0/BG(J,K)
GO TO 72
71 CONTINUE
BO(J,K)=GA +OMCC*(PO(J,K)-BP)
72 CONTINUE
GO TO 61
48 CONTINUE
RS(J,K)=DRS*PO(J,K)
BO(J,K)=DBO*PO(J,K)+1.04
61 CONTINUE
PO(J,K)=PO(J,K-1)-(GAMQ+GAMG*RS(J,K))*G*DZ/BO(J,K)
38 CONTINUE
PON(J,K)=PO(J,K)
15 CONTINUE
14 CONTINUE
GO TO 213
212 CONTINUE
DO 205 J=2,JJ,1
DO 206 K=1,JK,1
PC(J,K)=PSI*SQRTF(PHI/(CK*AK))*0.01
VQU(J,K)=0.0
VQZ(J,K)=0.0
VQU(J,K)=0.0
VGZ(J,K)=0.0
DVCU(J,K)=0.0
DVGU(J,K)=0.0
DVOZ(J,K)=0.0
DVGZ(J,K)=0.0
IF(ICS)229,230,229
229 CONTINUE
DR=RW*(EXP(U(J))-EXP(U(J-1)))
GO TO 231

```

```

230 CONTINUE
   DR=RW*DU
231 CONTINUE
   IF(SINF(ALPHA))225,207,207
225 CONTINUE
   PO(J,K)=PO(J-1,K)+(GAMO+GAMG*RS(J-1,K))*G*DR/BO(J-1,K)
   GO TO 221
207 CONTINUE
   PO(J,K)=PO(J-1,K)-(GAMO+GAMG*RS(J-1,K))*G*DR/BO(J-1,K)
221 CONTINUE
   DO 214 LN=1,30
     LN=LN
     IF(PO(J,K)-BP)215,215,216
216 CONTINUE
     RS(J,K)=RSI
     IF(GAMG)217,218,217
218 CONTINUE
     BG(J,K)=DBG*PO(J,K)+1.0
     BO(J,K)=1.0/BG(J,K)
     GO TO 219
217 CONTINUE
     BO(J,K)=GA +OMCC*(PO(J,K)-BP)
219 CONTINUE
     GO TO 220
215 CONTINUE
     RS(J,K)=DRS*PO(J,K)
     BO(J,K)=DBD*PO(J,K)+1.04
220 CONTINUE
     IF(SINF(ALPHA))222,223,223
222 CONTINUE
     PO(J,K)=PO(J-1,K)+(GAMO+GAMG*RS(J,K))*G*DR/BO(J,K)
     GO TO 224
223 CONTINUE
     PO(J,K)=PO(J-1,K)-(GAMO+GAMG*RS(J,K))*G*DR/BO(J,K)
224 CONTINUE
214 CONTINUE
     PON(J,K)=PO(J,K)
206 CONTINUE
205 CONTINUE
213 CONTINUE

```

```

C
C      CHOOSE INITIAL SATURATION DISTRIBUTION
C

```

```

   IF(SINF(ALPHA))226,227,226
226 CONTINUE
   INJ=2
   INK=1
   GO TO 228
227 CONTINUE
   INJ=1
   INK=2
228 CONTINUE
   DO 53 K=INK,JK,1
     DO 52 J=INJ,JJ,1
       IF(PON(J,K)-BP)50,50,51
51 CONTINUE
       PC(J,K)=PSI*SQRTF(PHI/(CK*AK))*0.01
       PG(J,K)=PO(J,K)+PC(J,K)
       SO(J,K)=SOM
       BG(J,K)=DBG*PG(J,K)+1.0
       TERM=(SO(J,K)-SOI)/(1.0-SOI)
       PKO(J,K)=PERM(J,K)*SO(J,K)**3*SQRTF(TERM)
       PKG(J,K)=PERM(J,K)*(1.0-TERM)**.25*SQRTF(SO(J,K))**2

```

```

GO TO 54
50 CONTINUE
SO(J,K)=SO(J,K-1)-(RS(J,K-1)-RS(J,K))/(POR(J,K)*BG(J,K-1))
TERM=(SO(J,K)-SOI)/(1.0-SOI)
DO 45 LN=1,30
LN=LN
PKO(J,K)=PERM(J,K)*SO(J,K)**3*SQRTF(TERM)
PC(J,K)=PSI*SQRTF(POR(J,K)*SO(J,K)/(C*PKO(J,K)))*0.01
PG(J,K)=PO(J,K)+PC(J,K)
BG(J,K)=DBG*PG(J,K)+1.0
SO(J,K)=SO(J,K-1)-(RS(J,K-1)-RS(J,K))/(POR(J,K)*BG(J,K))
45 CONTINUE
PKG(J,K)=PERM(J,K)*(1.0-TERM)*(1.0-TERM**.25*SQRTF(SO(J,K))**.2
54 CONTINUE
52 CONTINUE
53 CONTINUE
OMBA=DELTA
DO 25 K=1,JK
DO 24 J=1,JJ
VISO(J,K)=OVCON+OVCOEF*PON(J,K)
VISG(J,K)=GVCOEF *PG(J,K)+GVCON
24 CONTINUE
25 CONTINUE
POI=PO(1,MO)
13 CONTINUE
CALL ERRORS
CALL CRTID(2HJK,1,0)
CALL PLOTA(1)
WRITE OUTPUT TAPE 3,1203
WRITE OUTPUT TAPE 3,1202,IOPT,ICT,NP,NT,JZ,JT,IG,IUZ,ICS,IT,GMBB,
*OMBC,GMBC,OMBD,RHO,ZLAM,DELTA,GMBA,ICOV,OMBB,OMCC,CK,PHI,AK,GAMO,
*GAMG,POI,SOI, DU,DZ,DBO,DBG,DRS,QT,G,P,I,PSI,RW,SDM,SGN,PGM,FPERM,
*FPOR,PW,BOI
WRITE OUTPUT TAPE 3,1290,JB,MO,MM,JJ,J<,TIME,ALPHA,QDCUM,QGCUM,
1BP,RSI
28 CONTINUE
CALL PQCON
29 CONTINUE
30 CONTINUE
CALL PLOTE
CALL PDUMP1
CALL DUMP
1200 FORMAT(7I10/3I10,E10.3,3E10.2/3F10.1,1I0,2E10.3,E12.3)
1201 FORMAT(6E12.3/6E12.3/5E12.3/6E12.3/5I12/6E12.3/4E12.3)
1202 FORMAT(1H0/42X,10HINPUT DATA///9X,4HIOPT,9X,3HICT,9X,2HNP,10X,
12HNT,10X,2HJZ,10X,2HJT,10X,2HIG//7I12///10X,3HIUZ,9X,3HICS,
*10X,2HIT,6X,4HGMBB, 8X,2HDP,7X,7HEPSILON,6X,4HOMBD//3I12,F12.1,3E
212.2///9X,3HRHO,8X,4HZLAM,7X,5HDELTA,7X,4HGMBA,10X,4HICON,
*7X,4HOMBB,7X,4HOMCC//4F12.2,112,F12.2,E12.3///
3 6X,2HCK,10X,3HPI,9X,2HAK,9X,4HGAMO,8X,4HGAMG,9X,
43HPOI//6E12.3///6X,3HSOI,9X,2HDO,10X,2HDZ,9X,3HBO,9X,3HDBG,10X,
53HRS//6E12.3///6X,2HQT,10X,1HG,11X,2HI,9X,3HPSI,10X,2HRW,10X,
63HSDM//6E12.3///6X,3HSGM,9X,3HPGM,7X,5HFERM,8X,4HFPOR,9X,2HPW,
*9X,3HBOI//6E12.3//)
1290 FORMAT(1H0, 10X,2HJB,10X,2HMO,10X,2HMM,10X,2HJJ,10X,2HJK//
85I12///5X,4HTIME,7X,5HALPHA,7X,5HQDCUM,8X,5HOGCUM,8X,2HBP,7X,
93HRSI//6E12.3)
1203 FORMAT(1H1//35X,38H0IL AND GAS RESERVOIR ENGINEERING CODE///)
1204 FORMAT(6E12.3)
END

```

```

*      LIST8
*      CARDS COLUMN
*      FORTRAN          PQCON
C
C      OIL AND GAS FLOW AT CONSTANT WELL PRESSURE OR FLOW RATE
C
      SUBROUTINE PQCON
      USE STUFF
      LA=MO-2
      LB=MM+2
C
C      READ INPLT DATA
C
      IF(ICT)101,102,101
101 CONTINUE
      READ TAPE 6,((SO(J,K),SG(J,K),PG(J,K),PO(J,K),PC(J,K),BU(J,K),
      *BG(J,K),RS(J,K),VOU(J,K),VOZ(J,K),VGU(J,K),VGZ(J,K),PERM(J,K),
      *POR(J,K),PKQ(J,K),PKG(J,K),VISO(J,K),VISG(J,K),PON(J,K),
      *J=1,JJ),K=1,JK)
      UNLOAD 6
102 CONTINUE
      READ INPUT TAPE 2,1216,IO,LT,DQT
      WRITE OUTPUT TAPE 3,1218
      WRITE OUTPUT TAPE 3,1217,IO,LT,DQT,QVCON,GVCON,QVCOEF,GVCOEF
      WRITE OUTPUT TAPE 3,1215,((PO(J,K),J=1,JJ),K=1,JK)
      WRITE OUTPUT TAPE 3,1215,((SO(J,K),J=1,JJ),K=1,JK)
C
C      SET INDECES AND DEFINITIONS
C
      KK=0
      KM=0
      KKK=0
      JS=0
      L=0
      KL=0
      KKL=0
      K=0
      N=0
      OM=1.0
      GM=1.0
      CS=ICS
      ZD=JK/2
      QD=QT/ZD
      LX=0
      MI=2
      QGSC=0.0
      DT=GMBA
      ODT=DT
      DP=CMBC
      EPSIL=GMDC
      BP=GMBD
      RSI=OMCA
      BDI=GMCA
      SRS=0.0
C
C      SAVE GAS SOLUBILITIES ON WELL-BORE
C
      DO168 K=MO,PM,1
      ORS(K)=RS(1,K)
168 CONTINUE
      DO 401 K=1,JK,1
      DO 402 J=1,JJ,1

```

```

      PCO(J,K)=PC(J,K)
402 CONTINUE
401 CONTINUE
      DO 196 K=1,JK,1
      DO 195 J=1,JJ,1
      SON(J,K)=SO(J,K)
195 CONTINUE
196 CONTINUE
      CALL OGBAL
      OM=OMT
      GM=GMT
C
C      START OF MAIN LOOP
C
199 CONTINUE
      LSPIN=0
      TIME=TIME+3.0*DT
      KL=KL+1
C
C      REPLACE SO
C
      DO 166 K=1,JK,1
      DO 165 J=1,JJ,1
      SO(J,K)=SCN(J,K)
165 CONTINUE
166 CONTINUE
      IF(IUZ)140,244,140
C
C      COMPUTE WELL PRESSURE
C
244 CONTINUE
      IF(PCN(1,MO)-PW)140,140,104
104 CONTINUE
C
C      CHOOSE WELL-BORE BOUNDARY CONDITION
C
      GO TO(215,214,275,274),IOPT
C
C      OPTION 1 PW=F(TIME)
C
215 CONTINUE
      PON(1,MO)=PCI-(POI-PW)*TIME/GMB6
      DO 105 K=MO+1,MM
      PON(1,K)=PO(1,K)-(PO(1,MO)-PON(1,MO))
105 CONTINUE
      GO TO 220
C
C      OPTION 2 PW=F(QT(1))
C
214 CONTINUE
      LSPIN=LSPIN+1
      IF(LSPIN-100)266,266,241
266 CONTINUE
      DO 216 K=MO,MM,1
      PCN(1,K)=PO(1,K)-DP
216 CONTINUE
      GO TO 220
C
C      OPTION 3 PW=F(QT(2))
C
275 CONTINUE
      IF(ICS)277,278,277
277 CONTINUE

```

```

      EU=RW*(EXP(U(2))-EXP(U(1)))
      GO TO 279
278  CONTINUE
      EU=1.0/RW
279  CONTINUE
      DO 276 K=MO+1,MM-1,2
      PTA=DBG
      PTB=1.0-DBG*PCN(2,K)+OVCOEF*QD*EU      / (2.0*PI*GAMO*PKO(1,K)*G*
1DZ*2.0*RW)
      PTC=QD*OVCON*EU      / (2.0*PI*GAMC*G*PCD(1,K)+DZ*2.0*RW)-PCN(2,K)
      PON(1,K)=(-PTB+SQRT(PTB**2-4.0*PTA*PTC))/(PTA*2.0)
276  CONTINUE
      DO 280 K=MO+2,MM-2,2
      PON(1,K)=(PCN(1,K+1)+PCN(1,K-1))*0.5
280  CONTINUE
      PON(1,MC)=2.0*PON(1,MO+1)-PON(1,MO+2)
      PON(1,MP)=2.0*PON(1,MM-1)-PON(1,MM-2)
      GO TO 220

C
C      OPTION 4
C
274  CONTINUE
      IF(KL-1)288,288,289
288  CONTINUE
      DO 290 K=MO,MM,1
      PCN(1,K)=72000.0
290  CONTINUE
      GO TO 291
289  CONTINUE
      EU=RW*(EXP(U(2))-EXP(U(1)))
      DO 286 K=MO+1,MM-1,2
      PCN(1,K)=PO(1,K)+4.0*DZ*GAMC*G**2*PKO(2,K)*DT*(PO(2,K)-
1PO(1,K))/(RW*POR(1,K)*VISO(2,K)*BO(1,K)*EU)
286  CONTINUE
      DO 287 K=MO+2,MM-2,2
      PON(1,K)=(PCN(1,K+1)+PCN(1,K-1))*0.5
287  CONTINUE
      PON(1,MC)=2.0*PON(1,MO+1)-PON(1,MO+2)
      PON(1,MP)=2.0*PON(1,MM-1)-PCN(1,MP-2)
291  CONTINUE
220  CONTINUE

C
C      COMPUTE SATURATION ON WELL-BORE
C
      IF(10)227,228,227
228  CONTINUE
      IF(SON(1,JK)-1.0)229,227,227
229  CONTINUE
      QT=QT/DQT
      EPSIL=EPSIL/DQT
      IO=1
227  CONTINUE

C
C      QUADRATIC INTERPOLATION
C
      CALL QUAD

C
C      COMPUTE FORMATION FACTORS AND FLUID VISCOSITIES
C
      DO 108 K=1,JK,1
      DO 107 J=1,JJ,1
      BG(J,K)=DBG*PG(J,K)+1.0
      IF(PON(J,K)-BP)234,234,235

```

```

235 CONTINUE
  RS(J,K)=RS(J,K)
  IF(GAMG)254,253,254
253 CONTINUE
  BG(J,K)=DBG*PCN(J,K)+1.0
  BG(J,K)=1.0/BG(J,K)
  GO TO 255
254 CONTINUE
  BO(J,K)=GA+CMCC*(PCN(J,K)-BP)
255 CONTINUE
  GO TO 231
234 CONTINUE
  RS(J,K)=DRS*PON(J,K)
  BO(J,K)=DBO*PCN(J,K)+1.04
231 CONTINUE
  VISC(J,K)=OVCON+OVCCOF*PON(J,K)
  VISG(J,K)=GVCCOF *PG(J,K)+GVCON
107 CONTINUE
108 CONTINUE
  GO TO(217,218,217,217),IOPT

```

```

C
C      TEST FOR PROPER PRODUCTION RATE
C

```

```

218 CONTINUE
  QQ=0.0
  IF(TIME-GMBB)292,292,293
292 CONTINUE
  QP=QT*TIME/GMBB
  TOLER=EPSIL*TIME/GMBB
293 CONTINUE
  DO 209 K=MO+1,MM-1,2
    IF(ICS)251,245,251
251 CONTINUE
    EU=EXP(U(2))
    GC TO 252
245 CONTINUE
    EU=U(2)
252 CONTINUE
    QQ=QQ+PI*4.0*EU*G*GAMO*DZ*DPRO(2,K)*PKJ(2,K)/(BO(2,K)*VISO(2,K))
209 CONTINUE
    IF(DELTA)299,298,299
298 CONTINUE
    CQ=CQ*2.0/PI
299 CONTINUE
    WRITE OUTPUT TAPE 3,1211,DP,QQ,PON(1,MO+1),SON(1,MO+1),BO(2,MO+1),
    1PON(3,MO+1),VISO(2,MO+1),PKJ(2,MO+1),QT,LSPIN,DPRO(2,MO+1),QP
    IF(ABS(QQ-CQ)-TOLER)210,210,211
211 CONTINUE
    IF(SENSE SWITCH 1)241,240
240 CONTINUE
    IF(QQ-QP)212,210,213
212 CONTINUE
    DP=DP*1.07
    GO TO 214
213 CONTINUE
    DP=DP/1.05
    GO TO 214
210 CONTINUE
217 CONTINUE
140 CONTINUE
    IF(IUZ)294,295,294
294 CONTINUE

```

```

C
C      QUADRATIC INTERPOLATION
C
C      CALL QUAD
C
C      COMPUTE FORMATION FACTORS AND FLUID VISCOSITIES
C

```

```

      DO 222 K=1,JK,1
      DO 221 J=1,JJ,1
        BG(J,K)=DBG*PG(J,K)+1.0
        IF(PON(J,K)-BP)2,8,238,239
239      CONTINUE
        RS(J,K)=RS(J,K)
        IF(GAMG)257,256,257
256      CONTINUE
        BG(J,K)=DBG*PCN(J,K)+1.0
        BO(J,K)=1.0/BG(J,K)
        GO TO 258
257      CONTINUE
        BO(J,K)=GA+CMCC*(PON(J,K)-BP)
258      CONTINUE
        GO TO 242
238      CONTINUE
        RS(J,K)=DRS*PCN(J,K)
        BO(J,K)=DBO*PON(J,K)+1.04
242      CONTINUE
        VISO(J,K)=OVCCGN+OVCCF*PON(J,K)
        VISG(J,K)=GVCOEF *PG(J,K)+GVCON
221      CONTINUE
222      CONTINUE
295      CONTINUE

```

```

C
C      COMPUTE PRODUCTION RATES
C

```

```

      IF(IUZ)284,285,284
285      CONTINUE
      QX=0.0
      QY=0.0
      QZ=0.0
      IF(ICS)113,188,113
113      CONTINUE
      EU=EXPF(U(2))
      GO TO 155
188      CONTINUE
      EU=U(2)
155      CONTINUE
      DO 141 K=MO+1,MM-1,2
        QD(K)=4.0*PI*EU*GAMG*G*DZ*PKD(2,K)/(BO(2,K)*VISO(2,K))*
        1(DPRD(2,K)+ SINF(ALPHA)*(GAMG+RS(2,K)*GAMG)*G/BO(2,K))
        QX=QX+QD(K)
        QDG(K)=PI*EU*4.0*DZ*PKD(2,K)*GAMG*RS(2,K)*G/(BO(2,K)*VISO(2,K))*
        1(DPRD(2,K)+ SINF(ALPHA)*(GAMG+RS(2,K)*GAMG)*G/BO(2,K))
        QY=QY+QDG(K)
        QG(K)=PI*EU*4.0*DZ*G*GAMG*BG(2,K)*PKG(2,K)/VISG(2,K)*(DPRG(2,K)+
        1GAMG*G*BG(2,K)*SINF(ALPHA))
        QZ=QZ+QG(K)
141      CONTINUE
      IF(DELTA)301,300,301
300      CONTINUE
      QX=QX*2.0/PI
      QY=QY*2.0/PI

```



```

      QZ=QZ*2.0/PI
301  CONTINUE
      QCSC=QX/(GAMG*G)
      IF (GAMG)263,262,263
263  CONTINUE
      QGSC=(QY+QZ)/(GAMG*G)
262  CONTINUE
      IF (QX)297,297,296
296  CONTINUE
      GOR=QGSC*5.615/QCSC
297  CONTINUE
      QOCUM=QOCUM+QX*DT*4.0
      QGCUM=QGCUM+(QY+QZ)*DT*4.0

C
C      COMPUTE CURRENT MASS IN SYSTEM
C
      CALL QGBAL
284  CONTINUE

C
C      REPLACE SO ON WELL-BORE
C
      DO 219 K=MO,MM,1
      SO(1,K)=SCN(1,K)
219  CONTINUE

C
C      SAVE GAS SOLUBILITIES ON WELL-BORE
C
      DO 189 K=MO,MM,1
      ORS(K)=RS(1,K)
189  CONTINUE

C
C      REPLACE PO
C
      DO 152 K=1,JK
      DO 151 J=1,JJ
      PO(J,K)=PON(J,K)
151  CONTINUE
152  CONTINUE

C
C      COMPUTE NEW PRESSURES
C
      IF (IUZ)130,131,130
130  CONTINUE
      CALL COEFZ
      GO TO 132
131  CONTINUE
      CALL COEFU
132  CONTINUE

C
C      SPACE INTERPOLATION OF OIL PRESSURES
C
C
C
C      LINEAR INTERPOLATION
C
      DO 197 K=2,JN,2
      DO 198 J=6,JM,2
      IF (ICS)268,269,268
268  CONTINUE
      RATIO=(EXP(U(J))-EXP(U(J-1)))/(EXP(U(J+1))-EXP(U(J-1)))
      PON(J,K+1)=PON(J-1,K+1)+(PON(J+1,K+1)-PON(J-1,K+1))*RATIO
      PON(J,K-1)=PON(J-1,K-1)+(PON(J+1,K-1)-PON(J-1,K-1))*RATIO
      PON(J,K)=(PON(J,K+1)+PON(J,K-1))*0.5

```

```

GO TO 270
269 CONTINUE
PON(J,K)=(PON(J+1,K+1)+PON(J+1,K-1)+PON(J-1,K+1)+PON(J-1,K-1))
1=0.25
PON(J,K+1)=(PON(J+1,K+1)+PON(J-1,K+1))*0.5
PON(J,K-1)=(PON(J+1,K-1)+PON(J-1,K-1))*0.5
270 CONTINUE
PON(J+1,K)=(PON(J+1,K+1)+PON(J+1,K-1))*0.5
PON(J-1,K)=(PON(J-1,K+1)+PON(J-1,K-1))*0.5
198 CONTINUE
197 CONTINUE

C
C      QUADRATIC INTERPOLATION
C
C      CALL QUAD
C
C      STEP BACK IN TIME
C
C      TIME=TIME-DT
C
C      TIME INTERPOLATION OF OIL PRESSURES
C
C      DO 192 K=1,JK
C      DO 191 J=1,JJ
C      PN(J,K)=(PON(J,K)+PO(J,K))*0.5
191 CONTINUE
192 CONTINUE

C
C      COMPUTE FORMATION FACTORS AND VISCOSITIES
C
C
144 CONTINUE
DO 194 K=1,JK,1
DO 193 J=1,JJ,1
BG(J,K)=DBG*PG(J,K)+1.0
IF(PN(J,K)-BP)236,236,237
237 CONTINUE
RS(J,K)=RS(J,K)
IF(GAMG)260,259,260
259 CONTINUE
BG(J,K)=DBG*P N(J,K)+1.0
BO(J,K)=1.0/BG(J,K)
GO TO 261
260 CONTINUE
BU(J,K)=GA +DMCC*(PN(J,K)-BP)
261 CONTINUE
GO TO 243
236 CONTINUE
RS(J,K)=DRS*PN(J,K)
BO(J,K)=DBO*PN(J,K)+1.04
243 CONTINUE
PG(J,K)=PN(J,K)+PC(J,K)
VISO(J,K)=OVCON+OVCOEF*P N(J,K)
VISC(J,K)=GYCOEF *PG(J,K)+GVCON
193 CONTINUE
194 CONTINUE

C
C      COMPUTE INTERNAL VELOCITIES
C
C
DO 119 K=2,JN,2
DO 118 J=2,JM,2
IF(ICS)134,135,134
134 CONTINUE

```

```

EU=EXP(-U(J))/RW
GO TO 136
135 CONTINUE
EU=1.0/RW
136 CONTINUE
VQU(J,K)=-PKO(J,K)/VISO(J,K)*(EU*(PN(J+1,K)-PN(J-1,K)))/(DU*2.0)
*+(GAMO+RS(J,K)*GAMG)*G*SINF(ALPHA)/BO(J,K))
VQZ(J,K)=-PKO(J,K)/VISG(J,K)*((P N(J,K+1)-P N(J,K-1))/(DZ*2.0)
*+(GAMO+RS(J,K)*GAMG)*G*COSF(ALPHA)/BO(J,K))
ADDRESS JOHN
JOHN CONTINUE
VGU(J,K)=-PKG(J,K)/VISG(J,K)*(EU*(PG(J+1,K)-PG(J-1,K)))/(DU*2.0)
*+GAMG*BG(J,K)*G*SINF(ALPHA))
VGZ(J,K)=-PKG(J,K)/VISG(J,K)*((PG(J,K+1)-PG(J,K-1))/(DZ*2.0)
*+GAMG*BG(J,K)*G*COSF(ALPHA))
118 CONTINUE
119 CONTINUE

```

C
C COMPUTE SPACE DERIVATIVES CF INTERNAL VELOCITIES
C

```

DO 123 K=2,JN,2
DO 122 J=2,JM,2
IF(ICS)137,138,137
137 CONTINUE
EU=EXP(-U(J))/RW
GO TO 139
138 CONTINUE
EU=1.0/RW
139 CONTINUE
IF(GAMG)265,264,265
264 CONTINUE
DBO=-DBG/(BG(J,K)**2)
265 CONTINUE
DVQU(J,K)=-PKO(J,K)/VISO(J,K)*(EU*(P N(J+1,K)-P N(J,K)*2.0+
1P N(J-1,K))/(DU*2.0-(GAMC+RS(J,K)*GAMG)*G*DBQ*(P N(J+1,K)-
2P N(J-1,K))/(DU*2.0*BO(J,K)**2)*SINF(ALPHA)+GAMG*G* DRS*
3(P N(J+1,K)-P N(J-1,K))*SINF(ALPHA)/(DJ*2.0*BO(J,K)))
*-(EU*(P N(J+1,K)-P N(J-1,K))/(DU*2.0)+(GAMO+RS(J,K)*GAMG)*G*
4SINF(ALPHA)/BO(J,K))*
5((PKO(J+1,K)-PKO(J-1,K))/(DU*2.0*VISO(J,K))-PKO(J,K)*
6(VISO(J+1,K)-VISO(J-1,K))/(DU*2.0*VISO(J,K)**2)))
7+CS*(PKO(J,K)*EU*(PN(J+1,K)-PN(J-1,K))/(VISO(J,K)*DU*2.0))
DVGU(J,K)=-PKG(J,K)/VISG(J,K)*(EU*(PG(J+1,K)-PG(J-1,K)*2.0+PG
1(J-1,K))/(DU*2.0+ GAMG*G*DBG*(PG(J+1,K)-PG(J-1,K))*
2SINF(ALPHA)/(DU*2.0))-(EU*(PG(J+1,K)-PG(J-1,K))/(DU*2.0)+GAMG*
2BG(J,K)*G*SINF(ALPHA))*((PKG(J+1,K)-PKG(J-1,K))/(DU*2.0*
3VISG(J,K))-PKG(J,K)*(VISG(J+1,K)-VISG(J-1,K))/(DU*2.0*VISG(J,K)
5**2)))
6+CS*(PKG(J,K)*EU*(PG(J+1,K)-PG(J-1,K))/(VISG(J,K)*DU*2.0))
DVQZ(J,K)=-PKO(J,K)/VISO(J,K)*((P N(J,K+1)-P N(J,K)*2.0+P N(J,K-1)
1)/DZ**2-(GAMO+RS(J,K)*GAMG)*G*DBD*(P N(J,K+1)-P N(J,K-1))*
2COSF(ALPHA)/(DZ*2.0*BO(J,K)**2)+G*GAMG*DRS*(P N(J,K+1)-P N(J,K-1))
3*COSF(ALPHA)/(DZ*2.0*BO(J,K)))-((P N(J,K+1)-P N(J,K-1))/(DZ*2.0)+
*(GAMO+RS(J,K)*GAMG)*G*COSF(ALPHA)/BO(J,K))*((PKO(J,K+1)-PKO(J,K-1)
*)/(2.0*DZ*VISO(J,K))-PKO(J,K)*(VISO(J,K+1)-VISO(J,K-1))/(2.0*DZ*
*VISG(J,K)**2))
DVGZ(J,K)=-PKG(J,K)/VISG(J,K)*((PG(J,K+1)-PG(J,K)*2.0+PG(J,K-1))/
*DZ**2+GAMG*G*DBG*(PG(J,K+1)-PG(J,K-1))*COSF(ALPHA)/(2.0*DZ))
*-(PG(J,K+1)-PG(J,K-1))/(2.0*DZ)+GAMG*G*BG(J,K)*COSF(ALPHA))*
*((PKG(J,K+1)-PKG(J,K-1))/(2.0*DZ*VISG(J,K))-PKG(J,K)*(VISG(J,K+1)-
*VISG(J,K-1))/(2.0*DZ*VISG(J,K)**2))
122 CONTINUE
123 CONTINUE

```

C
C
C

COMPUTE NEW SATURATIONS

```

DO 110 K=2,JN,2
DO 109 J=MI,JM,2
IF (PON(J,K)-BP)225,225,226
226 CONTINUE
SON(J,K)=SO(J,K)
GO TO 109
225 CONTINUE
IF (ICS)111,112,111
111 CONTINUE
EU=EXP(-U(J))/RW
GO TO 133
112 CONTINUE
IF (CMBA)177,178,177
177 CONTINUE
DELTA =CMBA/U(J)
178 CONTINUE
EU=1.0/RW
133 CONTINUE
AT=DBO*(PON(J,K)-PO(J,K))/(BO(J,K)*DT*2.0)
BT=DBO*(EU*VOU(J,K)*(PN(J+1,K)-PN(J-1,K))/(DU*2.0)+VOZ(J,K)*
1(PN(J,K+1)-PN(J,K-1))/(DZ*2.0))/(BO(J,K)*POR(J,K))
CT=-(DELTA*EU*VOU(J,K)+DVOU(J,K)+DVOZ(J,K))/POR(J,K)
SON(J,K)=SO(J,K)*EXP(2.0*AT*DT)+(BT+CT)*2.0*DT
IF (J-2)501,501,502
501 CONTINUE
XATDT=EXP(2.0*AT*DT)
WRITE OUTPUT TAPE 3,1212,J,K,AT,BT,CT,SON(J,K),DT,BO(J,K),VOU(J,K)
*,XATDT
502 CONTINUE
IF (SON(J,K)-SCI)103,106,106
103 CONTINUE
SON(J,K)=SOI
106 CONTINUE

```

C
C
C

TEST FOR NEW SATURATIONS TOO LARGE

```

ASON=SON(J,K)-1.000
IF (ASON)174,174,172
172 CONTINUE
IF (ASON-1.0E-01)173,173,174
173 CONTINUE
SON(J,K)=1.0000
174 CONTINUE
IF (SON(J,K)-1.0)175,175,176
176 CONTINUE
GO TO 241
175 CONTINUE
109 CONTINUE
110 CONTINUE
163 CONTINUE

```

C
C
C

SPACE INTERPOLATION OF NEW SATURATIONS

```

DO 121 K=3,JP,2
DO 120 J=3,JQ,2
IF (ICS)271,272,271
271 CONTINUE
RATIO=(EXP(U(J))-EXP(U(J-1)))/(EXP(U(J+1))-EXP(U(J-1)))
SON(J,K+1)=SON(J-1,K+1)+(SON(J+1,K+1)-SON(J-1,K+1))*RATIO
SON(J,K-1)=SON(J-1,K-1)+(SON(J+1,K-1)-SON(J-1,K-1))*RATIO

```

```

      SON(J,K)=(SCN(J,K+1)+SCN(J,K-1))*0.5
      GO TO 273
272  CONTINUE
      SON(J,K)=(SCN(J+1,K+1)+SON(J+1,K-1)+SON(J-1,K+1)+SON(J-1,K-1))*0.25
      SCN(J,K+1)=(SCN(J+1,K+1)+SON(J-1,K+1))*0.5
      SCN(J,K-1)=(SCN(J+1,K-1)+SCN(J-1,K-1))*0.5
273  CONTINUE
      SON(J+1,K)=(SON(J+1,K+1)+SON(J+1,K-1))*0.5
      SCN(J-1,K)=(SON(J-1,K+1)+SON(J-1,K-1))*0.5
120  CONTINUE
121  CONTINUE
      DO 124 J=2,JM,1
      SON(J,1)=SON(J,2)*2.0-SON(J,3)
      SCN(J,JK)=SCN(J,JN)*2.0-SON(J,JP)
      IF (SON(J,1)-1.0)124,124,179
179  CONTINUE
      SON(J,1)=1.0
124  CONTINUE
      DO 125 K=2,JN,1
      SON(1,K)=SON(2,K)*2.0-SON(3,K)
      SCN(JJ,K)=SCN(JM,K)*2.0-SON(JO,K)
125  CONTINUE
      SON(1,1)=SON(2,2)*2.0-SON(3,3)
      SON(1,JK)=SCN(2,JN)*2.0-SON(3,JP)
      SON(JJ,1)=SCN(JM,2)*2.0-SON(JO,3)
      SON(JJ,JK)=SON(JM,JN)*2.0-SON(JO,JP)
      CALL SOQUAD
C
C      COMPUTE EFFECTIVE PERMEABILITIES
C
      DO 154 K=1,JK,1
      DO 153 J=1,JJ,1
      TERM=(SCN(J,K)-SOI)/(1.0-SOI)
      PKO(J,K)=PERM(J,K)*SCN(J,K)**3*SQRTF(TERM)
      PKG(J,K)=PERM(J,K)*(1.0-TERM)*(1.0-TERM**0.25*SQRTF(SON(J,K)))**2
153  CONTINUE
154  CONTINUE
C
C      SAVE OLD CAPILLARY PRESSURE
C
      DO 249 K=1,JK,1
      DO 248 J=1,JJ,1
      PCO(J,K)=PC(J,K)
248  CONTINUE
249  CONTINUE
C
C      COMPUTE CAPILLARY PRESSURE
C
      DO 184 K=1,JK,1
      DO 182 J=1,JJ,1
      PC(J,K)=PSI*SQRTF(PQR(J,K)*SON(J,K)/(CK*PKO(J,K)))*0.01
182  CONTINUE
184  CONTINUE
C
C      COMPUTE NEW GAS PRESSURE
C
      DO 129 K=1,JK,1
      DO 128 J=1,JJ,1
      PG(J,K)=PON(J,K)+PC(J,K)
128  CONTINUE
129  CONTINUE
C

```

C SELECT TIME INCREMENT
C

 KKL=KKL+1
 IF (IUZ)158,400,158
400 CONTINUE
 ODT=OT
 IF (KKL-IT)158,126,126
126 CONTINUE
 KKL=0
 IF (JZ)398,399,398
399 CONTINUE
 LX=LX+1
 IF (LX-LT)250,250,158
250 CONTINUE
 OT=OT*SGM
 GO TO 158
398 CONTINUE
 CALL KTIME
158 CONTINUE

C
C OUTPUT ROUTINES
C

 N=N+1
 IF (KL-5)207,207,208
208 CONTINUE
 IF (N-NP)170,171,171
171 CONTINUE
 N=0
207 CONTINUE
241 CONTINUE
 L=L+1
 CALL OUTPUT

C
C CALL CRT ROUTINES
C

 CALL POUZ
 CALL VOUZ
 IF (PON(1,PM-1)-BP)246,246,247
246 CONTINUE
 CALL PGUZ
 CALL PCUZ
 CALL SOUZ
 CALL VGUZ
247 CONTINUE
170 CONTINUE
 KM=KM+1
 IF (KM-NT)180,181,181
181 CONTINUE

C
C STORE TIME SUMMARIES
C

 KM=0
 IF (JS-JT)142,143,143
143 CONTINUE
 WRITE OUTPUT TAPE 3,1203
 WRITE OUTPUT TAPE 3,1214,JS,(TS(I),QOS(I),QGS(I),QOGS(I),
 *QTS(I),GORS(I),DMOS(I),DMGS(I),I=1,JS)
 CALL MOBAL
 CALL QOGXT
 CALL QGXT
 CALL QOGCUM
 JS=0

```

142 CONTINUE
   IF (IUZ) 403,404,403
403 CONTINUE
   JS=JS+1
   TS(JS)=TIME
   QQS(JS)=QX
   QQGS(JS)=QY
   QGS(JS)=QZ
   QTS(JS)=QOCUM
   GORS(JS)=QGCUM
   DMOS(JS)=DEVO
   DMGS(JS)=DEVG
404 CONTINUE
180 CONTINUE
   IF (ASON-1.0) 302,302,156
302 CONTINUE
   IF (LSPIN-100) 267,267,156
267 CONTINUE
   IF (SENSE SWITCH 1) 156,157
156 CONTINUE
   WRITE OUTPUT TAPE 3,1203
   WRITE OUTPUT TAPE 3,1214,JS,(TS(I),COS(I),QGS(I),QQGS(I),
* QTS(I),GORS(I),DMOS(I),DMGS(I),I=1,JS)
   CALL MOBAL
   CALL QOGXT
   CALL QGXT
   CALL QOGCUM
   REWIND 5
   WRITE TAPE 5,CK,PHI,AK,GAMO,GAMG,PCI,SOI,B,CU,DZ,DBO,DBG,DRS,QT,
* G,PI,PSI,RW,SOM,SGM,PGM,FPERM,FPOR,PW,JB,MO,MM,JJ,JK,
* TIME,QOCUM,QGCUM,ALPHA,GMBD,GMCA,GMCA,DMCC,OMBA,OVCON,GVCUN,
* OVCOEF,GVCOEF,KL,OM,GM
   WRITE TAPE 5,((SO(J,K),SG(J,K),PG(J,K),PO(J,K),PC(J,K),BO(J,K),
* BG(J,K),RS(J,K),VOU(J,K),VOZ(J,K),VGU(J,K),VGZ(J,K),PERM(J,K),
* POR(J,K),PKG(J,K),PKG(J,K),VISO(J,K),VISG(J,K),PON(J,K),
* J=1,JJ),K=1,JK)
   UNLOAD 5
   GO TO 150
157 CONTINUE
   IF (KL-6000) 185,186,186
186 CONTINUE
   KL=0
   REWIND 5
   WRITE TAPE 5,CK,PHI,AK,GAMO,GAMG,PCI,SOI,B,CU,DZ,DBO,DBG,DRS,QT,
* G,PI,PSI,RW,SOM,SGM,PGM,FPERM,FPOR,PW,JB,MO,MM,JJ,JK,
* TIME,QOCUM,QGCUM,ALPHA,GMBD,GMCA,GMCA,DMCC,OMBA,OVCON,GVCUN,
* OVCOEF,GVCOEF,KL,OM,GM
   WRITE TAPE 5,((SO(J,K),SG(J,K),PG(J,K),PO(J,K),PC(J,K),BO(J,K),
* BG(J,K),RS(J,K),VOU(J,K),VOZ(J,K),VGU(J,K),VGZ(J,K),PERM(J,K),
* POR(J,K),PKG(J,K),PKG(J,K),VISO(J,K),VISG(J,K),PON(J,K),
* J=1,JJ),K=1,JK)
185 CONTINUE
   GO TO 199
150 CONTINUE
100 CONTINUE
   RETURN
1203 FORMAT(1H1//35X,38H OIL AND GAS RESERVOIR ENGINEERING CODE////)
1211 FORMAT(1H0/4E16.7/5E16.7/116,2E16.7)
1212 FORMAT(1H0/214/8E14.7)
1213 FORMAT(1H0/5E15.8)
1214 FORMAT(1H0/20X,26H TIME SUMMARY OF PRODUCTION//10X,110//
* 11X,2HTS,10X,2HQD,11X,2HOG,11X,3HQG,10X,2HQT,11X,3HGOR,10X,3HMD
* ,11X,3HDMG//15X,8E13.3))

```

```

1215 FORMAT(1H1/(8E14.5))
1216 FORMAT(2I10,F10.2)
1217 FORMAT(1H0/9X,2H10,8X,2H1T,7X,3H0GT,7X,5H0VCON,6X,5HGVCON,
      *6X,6H0VCOEF,6X,6HGVCOEF//2I10,F10.2,4E12.3)
1218 FORMAT(1H0)
      END

```

```

*      LIST8
*      CARDS COLUMN
*      FORTRAN      COEFU
      SUBROUTINE COEFU

```

```

C
C      MATRIX COEFFICIENT SUBROUTINE - U-CIRCULATION
C

```

```

      USE STUFF
      BP=GMBD
      MZ=0
      SBD=GMBCC
      SRS=0.0
      CS=ICS
      JL=JJ/2
      JI=JK/2
      ICOUNT=JL+1

```

```

C
C      ZERO OUT COEFFICIENT ARRAYS
C

```

```

      DO 596 K=1,JK
      DO 595 J=1,JJ
      SG(J,K)=0.0
      ZETA(J,K)=0.0
      ETA(J,K)=0.0
      THETA(J,K)=0.0
      TAU(J,K)=0.0
      C(J,K)=0.0
      E(J,K)=0.0
      E1(J,K)=0.0
      E2(J,K)=0.0
      E3(J,K)=0.0
      E4(J,K)=0.0
      E5(J,K)=0.0
      E6(J,K)=0.0
      E7(J,K)=0.0
      E8(J,K)=0.0
      E9(J,K)=0.0
      E10(J,K)=0.0
      E11(J,K)=0.0
      E12(J,K)=0.0
      E13(J,K)=0.0
      E14(J,K)=0.0
      E15(J,K)=0.0
      E16(J,K)=0.0

```

```

595 CONTINUE

```

```

596 CONTINUE
      DO 575 K=1,3
      DO 574 J=1,JJ
      A(J,K)=0.0

```

```

574 CONTINUE

```

```

575 CONTINUE
      DO 576 J=1,JJ
      Y(J)=0.0
      H(J)=0.0

```

```

576 CONTINUE

```

```

C

```


C COMPUTE E TERMS AND ETA, THETA, ZETA, AND TAU
C

```

DO 530 K=1,JK,2
DO 529 J=1,JJ,2
IF (ICS) 531,532,531
531 CONTINUE
EU=EXP(-U(J))/RW
GO TO 533
532 CONTINUE
EU=1.0/(RW*L(J))
533 CONTINUE
IF (GAMG) 101,102,101
102 CONTINUE
ZETA(J,K)=0.0
SBO=-DBG/(BG(J,K)**2)
GO TO 103
101 CONTINUE
ZETA(J,K)=BG(J,K)*BG(J,K)/(RS(J,K)-BO(J,K)*BG(J,K))
103 CONTINUE
THETA(J,K)=ZETA(J,K)*DBG/BG(J,K)
ETA(J,K)=1.0-RS(J,K)/(RS(J,K)-BG(J,K)*BG(J,K))
E5(J,K)=GAMG*BG(J,K)*G*SINF(ALPHA)
E6(J,K)=GAMG*BG(J,K)*G*COSF(ALPHA)
E7(J,K)=(GAMG*RS(J,K)*GAMG)*G*SINF(ALPHA)/BO(J,K)
E8(J,K)=(GAMG*RS(J,K)*GAMG)*G*COSF(ALPHA)/BG(J,K)
E15(J,K)=PKG(J,K)/VISQ(J,K)
E16(J,K)=PKG(J,K)/VISG(J,K)
IF (PO(J,K)-BP) 801,801,800
801 CONTINUE
TAU(J,K)=ETA(J,K)*DBO/BO(J,K)+ORS/(RS(J,K)-BO(J,K)*BG(J,K))
C(J,K)=SO(J,K)*TAU(J,K)+THETA(J,K)*(1.0-SO(J,K))
E11(J,K)=EU*E7(J,K)*DBG/BO(J,K)
E12(J,K)=G*GAMG*EU*DRS*SINF(ALPHA)/BO(J,K)
E13(J,K)=E8(J,K)*DBO/BO(J,K)
E14(J,K)=G*GAMG*DRS*COSF(ALPHA)/BG(J,K)
GO TO 803
800 CONTINUE
TAU(J,K)=ETA(J,K)*SBO/BG(J,K)+SRS/(RS(J,K)-BO(J,K)*BG(J,K))
C(J,K)=SO(J,K)*TAU(J,K)+THETA(J,K)*(1.0-SO(J,K))
E11(J,K)=EU*E7(J,K)*SBO/BO(J,K)
E12(J,K)=G*GAMG*EU*SRS*SINF(ALPHA)/BO(J,K)
E13(J,K)=E8(J,K)*SBO/BO(J,K)
E14(J,K)=G*GAMG*SRS*COSF(ALPHA)/BG(J,K)
803 CONTINUE
529 CONTINUE
530 CONTINUE
DO 591 K=1,JK,2
DO 590 J=3,J0,2
IF (ICS) 536,537,536
536 CONTINUE
EU=EXP(-U(J))/RW
GO TO 538
537 CONTINUE
EU=1.0/RW
538 CONTINUE
E1(J,K)=EU*((PKG(J+1,K)-PKG(J-1,K))/VISG(J,K)-PKG(J,K)*
1 (VISG(J+1,K)-VISG(J-1,K))/VISG(J,K)**2)/(2.0*DU)
E2(J,K)=EU*((PKG(J+1,K)-PKG(J-1,K))/VISQ(J,K)-PKQ(J,K)*
1 (VISQ(J+1,K)-VISQ(J-1,K))/VISQ(J,K)**2)/(2.0*DU)
E9(J,K)=EU*GAMG*G*DBG*SINF(ALPHA)*(PG(J+1,K)-PG(J-1,K))/(2.0*DU)
590 CONTINUE
591 CONTINUE
DO 535 K=3,JP,2

```

```

DO 584 J=1,JJ,2
IF (ICS)592,593,592
592 CONTINUE
EU=EXP(-U(J))/RW
GO TO 594
593 CONTINUE
EU=1.0/RW
594 CONTINUE
E3(J,K)=((PKG(J,K+1)-PKG(J,K-1))/VISG(J,K)-PKG(J,K)*
1(VISG(J,K+1)-VISG(J,K-1))/VISG(J,K)**2)/(2.0*DZ)
E4(J,K)=((PKO(J,K+1)-PKO(J,K-1))/VISO(J,K)-PKO(J,K)*
1(VISO(J,K+1)-VISO(J,K-1))/VISO(J,K)**2)/(2.0*DZ)
E10(J,K)=GAMG*G*DBG*CSF(ALPHA)*(PG(J,K+1)-PG(J,K-1))/(2.0*DZ)
584 CONTINUE
535 CONTINUE
DO 727 J=1,JJ,2
E3(J,1)=((PKG(J,3)-PKG(J,1))/VISG(J,1)-PKG(J,1)*
1(VISG(J,3)-VISG(J,1))/VISG(J,1)**2)/(2.0*DZ)
E3(J,JK)=((PKG(J,JK)-PKG(J,JP))/VISG(J,JK)-PKG(J,JK)*
1(VISG(J,JK)-VISG(J,JP))/VISG(J,JK)**2)/(2.0*DZ)
E4(J,1)=((PKO(J,3)-PKO(J,1))/VISO(J,1)-PKO(J,1)*
1(VISO(J,3)-VISO(J,1))/VISO(J,1)**2)/(2.0*CZ)
E4(J,JK)=((PKO(J,JK)-PKO(J,JP))/VISO(J,JK)-PKO(J,JK)*
1(VISO(J,JK)-VISO(J,JP))/VISO(J,JK)**2)/(2.0*DZ)
E10(J,1)=GAMG*G*DBG*CSF(ALPHA)*(PG(J,3)-PG(J,1))/(DZ*2.0)
E10(J,JK)=GAMG*G*DBG*CSF(ALPHA)*(PG(J,JK)-PG(J,JP))/(DZ*2.0)
727 CONTINUE

```

C
C
C

COMPUTE U-DIRECTION PRESSURES - INTERNAL

```

DO 501 K=3,JP,2
DO 500 J=3,JD,2
IF (ICS)502,503,502
502 CONTINUE
EU=EXP(-U(J))/RW
GO TO 504
503 CONTINUE
IF (OMBA)601,602,601
601 CONTINUE
DELTA=OMBA/U(J)
602 CONTINUE
EU=1.0/RW
504 CONTINUE
AA=RHO*(ZETA(J,K)/(4.0*DU)*(DELTA+E16(J,K)*EU**2-E16(J,K)*
1EU**2/DU+E1(J,K)*EU)-ETA(J,K)/(4.0*DU)*(DELTA+E15(J,K)*EU**2-E15
2(J,K)*EU**2/DU-E11(J,K)*E15(J,K)+E12(J,K)*E15(J,K)+E2(J,K)*EU)+
3THETA(J,K)/(4.0*DU)*(E16(J,K)*EU**2-(PO(J+2,K)-PO(J-2,K))+2.0*
4(PC(J+2,K)-PC(J-2,K)))/(4.0*DU)+E5(J,K)*E16(J,K)*EU)+TAU(J,K)/
5(4.0*DU)*(E15(J,K)*EU**2-(PG(J+2,K)-PG(J-2,K))/(4.0*DU)+E7(J,K)*
6E15(J,K)*EU))+CS*
7(RHO*EU**2*(ZETA(J,K)*E15(J,K)-ZETA(J,K)*E16(J,K))/(DU*4.0))
BB=C(J,K)*PCR(J,K)/(CT*2.0)+RHO*(ZETA(J,K)*E16(J,K)*EU**2/
1(2.0*DU**2)-ETA(J,K)*E15(J,K)*EU**2/(2.0*DU**2))
CC=RHO*(-ZETA(J,K)/(4.0*DU)*(DELTA+E16(J,K)*EU**2+E16(J,K)*EU**2/
1DU+E1(J,K)*EU)+ETA(J,K)/(4.0*DU)*(DELTA+E15(J,K)*EU**2+E15(J,K)*
2EU**2/DU+E12(J,K)*E15(J,K)-E11(J,K)*E15(J,K)+E2(J,K)*EU)-THETA(J,
3K)/(4.0*DU)*E16(J,K)*EU**2-(PO(J+2,K)-PG(J-2,K))+PC(J+2,K)-PC
4(J-2,K))*2.0)/(4.0*DU)+E5(J,K)*E16(J,K)*EU)-TAU(J,K)/(4.0*DU)*
5(E15(J,K)*EU**2/(4.0*DU)*(PO(J+2,K)-PG(J-2,K))+E7(J,K)*E15(J,K)*
6EU))+CS*
7(RHO*EU**2*(ZETA(J,K)*E16(J,K)-ETA(J,K)*E15(J,K))/(DU*4.0))
DD=ZETA(J,K)*(RHO*(DELTA+E5(J,K)*E16(J,K)*EU+DELTA+E16(J,K)*EU**2*
1(PC(J+2,K)-PC(J-2,K))/(4.0*DU)+E16(J,K)*EU**2/(4.0*DU**2)*

```

```

2(PC(J+2,K)-PC(J,K)*2.0+PC(J-2,K))+E1(J,K)*EU*(PC(J+2,K)-PC(J-2,K))
3/(4.0*DU)*E1(J,K)*E5(J,K)+E9(J,K)*E16(J,K))+ZLAM*(E16(J,K)/
4(4.0*DZ**2))*((PO(J,K+2)-PO(J,K)*2.0+PO(J,K-2)))+(PC(J,K+2)-PC(J,K)*
52.0+PC(J,K-2)))+E10(J,K)*E16(J,K)+E3(J,K)/(4.0*DZ)*((PO(J,K+2)-
6PC(J,K-2)))+(PC(J,K+2)-PC(J,K-2))+E3(J,K)*E6(J,K))
EE=-ETA(J,K)*(RHO*(DELTA*E7(J,K)*E15(J,K)*EU+E2(J,K)*E7(J,K))+
1ZLAM*(E15(J,K)*(PO(J,K+2)-PO(J,K)*2.0+PO(J,K-2)))/(4.0*DZ**2)+
2(E14(J,K)*E15(J,K)-E13(J,K)*E15(J,K)+E4(J,K))*((PO(J,K+2)-PO(J,K-2)
*)/(4.0*DZ))+E4(J,K)*E8(J,K))+THETA(J,K)*ZLAM*(E16(J,K)*((PO(J,K+2)
*-PO(J,K-2))*2+2.0*(PO(J,K+2)-PO(J,K-2))*(PC(J,K+2)-PC(J,K-2))+
3(PC(J,K+2)-PC(J,K-2))*2)/(4.0*DZ)**2+E6(J,K)*E16(J,K)*((PO(J,K+2)
4-PO(J,K-2)))+(PC(J,K+2)-PC(J,K-2))
5)/(4.0*DZ))+TAU(J,K)*ZLAM*(E15(J,K)*(PO(J,K+2)-PO(J,K-2))*2/(4.0*
6DZ)**2+E8(J,K)*E15(J,K)*(PO(J,K+2)-PO(J,K-2))/(4.0*DZ))+C(J,K)*
7POR(J,K)*PO(J,K)/(DT*2.0)
FF=THETA(J,K)*RHO*(E16(J,K)*EU**2*(PC(J+2,K)-PC(J-2,K))*2/
1(4.0*DU)**2+E16(J,K)*E5(J,K)*EU*(PC(J+2,K)-PC(J-2,K))/(4.0*DU))+
2POR(J,K)*THETA(J,K)*(1.0-SO(J,K))*(PC(J,K)-PCO(J,K))/(BT*2.0)-CS*
3(ZETA(J,K)*RHO*E16(J,K)*EU**2*(PC(J+2,K)-PC(J-2,K))/(DU*4.0)
E(J,1)=AA
E(J,2)=BB
E(J,3)=CC
H(J)=DD+EE+FF

```

500 CONTINUE

C
C
C COMPUTE OUTER EDGE U-DIRECTION

```

IF(ICS)505,506,505
505 CONTINUE
EU=EXP(-U(JJ))/RW
GO TO 507
506 CONTINUE
IF(CMBA)603,604,603
603 CONTINUE
DELTA=OMBA/U(JJ)
604 CONTINUE
EU=1.0/RW
507 CONTINUE
AA=RHO*(ETA(JJ,K)*E15(JJ,K)*EU**2/(2.0*DU**2)-ZETA(JJ,K)*EU**2*
E16(JJ,K)/(2.0*DU**2))
BB=C(JJ,K)*POR(JJ,K)/(2.0*DT)+RHO*(ZETA(JJ,K)*E15(JJ,K)*EU**2/
1(2.0*DU**2)-ETA(JJ,K)*E15(JJ,K)*EU**2/(2.0*DU**2))
CC=C(JJ,K)*POR(JJ,K)*PO(JJ,K)/(DT*2.0)+ZETA(JJ,K)*(RHO*(DELTA*
E5(JJ,K)*E16(JJ,K)*EU+E16(JJ,K)*EU**2*(PC(JJ,K)-PC(JJ,K-2))/
2(2.0*DU**2))+ZLAM*(E16(JJ,K)/(4.0*DZ**2))*((PO(JJ,K+2)-PO(JJ,K)*
32.0+PO(JJ,K-2)))+(PC(JJ,K+2)-PC(JJ,K)*2.0+PC(JJ,K-2))+E16(JJ,K)*
4E16(JJ,K)+E3(JJ,K)/(4.0*DZ)*((PO(JJ,K+2)-PO(JJ,K-2)))+(PC(JJ,K+2)-
5PC(JJ,K-2))+E3(JJ,K)*E6(JJ,K))
DD=-ETA(JJ,K)*((DELTA*E7(JJ,K)*E15(JJ,K)*EU)*RHO+ZLAM*(E15(JJ,K)*
1(PO(JJ,K+2)-PO(JJ,K)*2.0+PO(JJ,K-2))/(4.0*DZ**2)+(E14(JJ,K)*
2E15(JJ,K)-E13(JJ,K)*E15(JJ,K)+E4(JJ,K))*((PO(JJ,K+2)-PO(JJ,K-2))/
3(4.0*DZ))+E4(JJ,K)*E8(JJ,K))+THETA(JJ,K)*ZLAM*(E16(JJ,K)/(4.0*DZ)
4**2*((PO(JJ,K+2)-PO(JJ,K-2))*2+(PC(JJ,K+2)-PC(JJ,K-2))*
5(PC(JJ,K+2)-PC(JJ,K-2)))+E6(JJ,K)*E16(JJ,K)*(PO(JJ,K+2)-PO(JJ,K-2)
6)/(4.0*DZ))+TAU(JJ,K)*ZLAM*(E15(JJ,K)*(PO(JJ,K+2)-PO(JJ,K-2))*2/
7(4.0*DZ)**2+E8(JJ,K)*E15(JJ,K)*(PC(JJ,K+2)-PO(JJ,K-2))/(4.0*DZ))
EE=THETA(JJ,K)*ZLAM*(E16(JJ,K)*(PC(JJ,K+2)-PC(JJ,K-2))*2/(4.0*
1DZ)**2+E6(JJ,K)*E16(JJ,K)*(PC(JJ,K+2)-PC(JJ,K-2))/(4.0*DZ))
2*THETA(JJ,K)*POR(JJ,K)*(1.0-SO(JJ,K))*(PC(JJ,K)-PCO(JJ,K))/
3(2.0*DT)
E(JJ,1)=AA
E(JJ,2)=BB
H(JJ)=CC+DD+EE

```

```

C
C      COMPUTE INNER EDGE  U-DIRECTION
C
      IF(ICS)508,509,508
508 CONTINUE
      EU=EXP(-U(1))/RW
      GO TO 510
509 CONTINUE
      IF(CMBA)605,606,605
605 CONTINUE
      DELTA=OMBA/U(1)
606 CONTINUE
      EU=1.0/RW
510 CONTINUE
      IF(K-MO)511,513,512
512 CONTINUE
      IF(K-MM)513,513,511
511 CONTINUE
      AA=C(1,K)*PCR(1,K)/(DT*2.0)+RHO*(ZETA(1,K)*E16(1,K)*EU**2/
1(2.0*DU**2)-ETA(1,K)*E15(1,K)*EU**2/(2.0*DU**2))
      BB=RHO*(ETA(1,K)*EU**2+E15(1,K)/(2.0*DU**2)-ZETA(1,K)*E16(1,K)*
1EU**2/(2.0*DU**2))
      CC=C(1,K)*PCR(1,K)*PO(1,K)/(DT*2.0)+ZETA(1,K)*(RHO*(DELTA*E5(1,K)*
1E16(1,K)*EU+E16(1,K)*EU**2*(PC(3,K)-PC(1,K))/(2.0*DU**2))+ZLAM*
2(E16(1,K)*((PC(1,K+2)-PO(1,K)*2.0+PO(1,K-2))+PC(1,K+2)-PC(1,K)*
32.0+PC(1,K-2))/(4.0*DZ**2)+E10(1,K)*E16(1,K)+E3(1,K)*((PO(1,K+2)-
4PO(1,K-2))+PC(1,K+2)-PC(1,K-2))/(4.0*DZ**2)+E3(1,K)*E6(1,K)))
      DD=-ETA(1,K)*(RHO*DELTA*E15(1,K)*EU+ZLAM*(E15(1,K)*(PO(1,K+2)-
1PO(1,K)*2.0+PO(1,K-2))/(4.0*DZ**2)+(E14(1,K)*E15(1,K)-E13(1,K)*
2E15(1,K)+E4(1,K))*((PO(1,K+2)-PO(1,K-2))/(4.0*DZ**2)+E4(1,K)*E8(1,K)))
3+THETA(1,K)*ZLAM/(4.0*DZ)*(E16(1,K)*((PO(1,K+2)-PO(1,K-2))*2+
4PO(1,K+2)-PO(1,K-2))*(PC(1,K+2)-PC(1,K-2))/(4.0*DZ**2)+E6(1,K)*
5E16(1,K)*((PC(1,K+2)-PO(1,K-2))+TAU(1,K)*ZLAM/(4.0*DZ)*(E15(1,K)*
6PO(1,K+2)-PO(1,K-2))*2/(4.0*DZ)+E8(1,K)*E15(1,K)*(PO(1,K+2)-
7PC(1,K-2)))
      EE=THETA(1,K)*ZLAM*(E16(1,K)*(PC(1,K+2)-PC(1,K-2))*2/(4.0*
1DZ**2)+E6(1,K)*E16(1,K)*(PC(1,K+2)-PC(1,K-2))/(4.0*DZ))
      Z=THETA(1,K)*POR(1,K)*(1.0-SO(1,K))*(PC(1,K)-PCO(1,K))/
3(2.0*DT)
      E(1,2)=AA
      E(1,3)=BB
      H(1)=CC+DD+EE
      GO TO 514
513 CONTINUE
      E(1,2)=E(3,1)*2.0
      E(1,3)=0.0
      H(1)=PO(1,K)*E(3,1)*2.0
514 CONTINUE
      IF(DELTA)632,633,632
632 CONTINUE
      DO 641 NN=1,3,1
      DO 600 MN=1,JJ,2
      SG(MN,NN)=ABS(F(MN,NN))
600 CONTINUE
641 CONTINUE
      EMAX=AMAX(F(SG,1,99,1,MZ))
      DO 643 MN=1,JJ,2
      H(MN)=H(MN)/EMAX
      DO 642 NN=1,3,1
      E(MN,NN)=E(MN,NN)/EMAX
642 CONTINUE
643 CONTINUE
633 CONTINUE

```

C
C
C

COMPRES. MATRIX COEFFICIENTS

```

DO 568 I=2,JL,1
A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)
568 CONTINUE
A(1,2)=E(1,2)
A(1,3)=E(1,3)
Y(1)=H(1)
A(JL+1,1)=E(JJ,1)
A(JL+1,2)=E(JJ,2)
Y(JL+1)=H(JJ)
CALL MATRIX
DO 515 I=1,JJ,2
PON(I,K)=X(I/2+1)
515 CONTINUE
501 CONTINUE

```

C
C
C

COMPUTE LOWER EDGE U-DIRECTION

```

DO 516 J=3,J0,2
IF(ICS)517,518,519
517 CONTINUE
EU=EXP(-U(J))/RW
GO TO 519
518 CONTINUE
IF(CMBA)607,608,609
607 CONTINUE
DELTA=CMBA/U(J)
608 CONTINUE
EU=1.0/RW
519 CONTINUE
AA=RHO*(ZETA(J,1)/(4.0*DU)*(E16(J,1)*EU**2*DELTA-E16(J,1)*
1EU**2/DU+
2E1(J,1)*EU)-ETA(J,1)/(4.0*DU)*(DELTA*E15(J,1)*EU**2-E15(J,1)*
3EU**2/DU-E11(J,1)*E15(J,1)+E12(J,1)*E15(J,1)+E2(J,1)*EU)+THETA
4(J,1)/(4.0*DU)*(E16(J,1)*EU**2*(PC(J+2,1)-PO(J-2,1))+(PC(J+2,1)-
5PC(J-2,1)))/(4.0*DU)+E5(J,1)*E16(J,1)*EU)+TAU(J,1)/(4.0*DU)*
6(E15(J,1)*EU**2*(PO(J+2,1)-PO(J-2,1))/(4.0*DU)+E7(J,1)*E15(J,1)*
7EU))+CS*
8(RHO*EU**2*(ETA(J,1)*E15(J,1)-ZETA(J,1)*E16(J,1))/(DU*4.0))
BB=C(J,1)*PCR(J,1)/(DT*2.0)+RHO*(ZETA(J,1)*E16(J,1)*EU**2/
1(2.0*DU**2)-ETA(J,1)*E15(J,1)*EU**2/(2.0*DU**2))
CC=RHO*(-ZETA(J,1)/(4.0*DU)*(DELTA*E16(J,1)*EU**2+E16(J,1)*EU**2/
1DU+E1(J,1)*EU)+ETA(J,1)/(4.0*DU)*(DELTA*E15(J,1)*EU**2+E15(J,1)*
2EU**2/DU+
3E12(J,1)*E15(J,1)-E11(J,1)*E15(J,1)+E2(J,1)*EU)-THETA(J,1)/
4(4.0*DU)*(E16(J,1)*EU**2*(PC(J+2,1)-PO(J-2,1))+(PC(J+2,1)-PC
5(J-2,1)))/(4.0*DU)+E5(J,1)*E16(J,1)*EU)-TAU(J,1)/(4.0*DU)*
6(E15(J,1)*EU**2/(4.0*DU)*(PO(J+2,1)-PO(J-2,1))+E7(J,1)*E15(J,1)*
7EU))+CS*
8(RHO*EU**2*(ZETA(J,1)*E16(J,1)-ETA(J,1)*E15(J,1))/(DU*4.0))
DD=C(J,1)*PCR(J,1)*PO(J,1)/(DT*2.0)+ZETA(J,1)*(RHO*(DELTA*(E5(J,1)
**
1E16(J,1)*EU+E16(J,1)*EU**2*(PC(J+2,1)-PC(J-2,1))/(4.0*DU))+E9(J,1)
**
2E16(J,1)+E16(J,1)*EU**2*(PC(J+2,1)-PC(J,1)*2.0+PC(J-2,1))/(4.0*
3DU**2)+E1(J,1)*EU*(PC(J+2,1)-PC(J-2,1))/(4.0*DU)+E1(J,1)*E5(J,1))
7-ETA(J,1)*(RHO*(DELTA*E7(J,1)*E15(J,1)*EU+E2(J,1)*E7(J,1)))

```

```

EE=THETA(J,1)*RHO*E16(J,1)*(EU**2*(PC(J+2,1)-PC(J-2,1))/
1(4.0*DU)**2+E5(J,1)*EU*(PC(J+2,1)-PC(J-2,1))/(4.0*DU))+
2THETA(J,1)*POR(J,1)*(1.0-SO(J,1))*(PC(J,1)-PCO(J,1))/
3(DT*2.0)-CS*
4(ZETA(J,1)*RHO*E16(J,1)*EU**2*(PC(J+2,1)-PC(J-2,1))/(DU*4.0))
E(J,1)=AA
E(J,2)=BB
E(J,3)=CC
H(J)=DD+EE
516 CONTINUE
C
C      COMPUTE LOWER CORNERS   U-DIRECTION
C
IF(ICS)520,521,520
520 CONTINUE
EU=EXP(-U(1))/RW
GO TO 522
521 CONTINUE
IF(CMBA)609,610,609
609 CONTINUE
DELTA=OMBA/U(1)
610 CONTINUE
EU=1.0/RW
522 CONTINUE
IF(MO-1)474,475,474
475 CONTINUE
E(1,2)=E(3,1)*2.0
E(1,3)=0.0
H(1)=PO(1,1)*E(3,1)*2.0
GO TO 476
474 CONTINUE
AA=C(1,1)*PCR(1,1)/(DT*2.0)+RHO*EU**2/(2.0*DU**2)*(ZETA(1,1)*
1E16(1,1)-ETA(1,1)*E15(1,1))
BB=RHO*EU**2/(2.0*DU**2)*(ETA(1,1)*E15(1,1)-ZETA(1,1)*E16(1,1))
CC=C(1,1)*PCR(1,1)*PO(1,1)/(DT*2.0)+ZETA(1,1)*(RHO*(DELTA*E5(1,1)*
1E16(1,1)+EU*E16(1,1)*EU**2*(PC(3,1)-PC(1,1))/(2.0*DU**2)))
2-
3ETA(1,1)*(RHO*DELTA*E7(1,1)*E15(1,1)*EU)
E(1,2)=AA
E(1,3)=BB
H(1)=CC
476 CONTINUE
IF(ICS)523,524,523
523 CONTINUE
EU=EXP(-U(JJ))/RW
GO TO 525
524 CONTINUE
IF(OMBA)611,612,611
611 CONTINUE
DELTA=OMBA/U(JJ)
612 CONTINUE
EU=1.0/RW
525 CONTINUE
BB=C(JJ,1)*POR(JJ,1)/(DT*2.0)+RHO*EU**2/(2.0*DU**2)*(ZETA(JJ,1)*
1E16(JJ,1)-ETA(JJ,1)*E15(JJ,1))
AA=RHO*EU**2/(2.0*DU**2)*(ETA(JJ,1)*E15(JJ,1)-ZETA(JJ,1)*
1E16(JJ,1))
CC=C(JJ,1)*POR(JJ,1)*PC(JJ,1)/(DT*2.0)+ZETA(JJ,1)*(RHO*(DELTA*
1E5(JJ,1)*E16(JJ,1)*EU*E16(JJ,1)+EU**2*(PC(JJ,1)-PC(JJ,1))/(2.0*
2DU**2)))
3      -ETA(JJ,1)*(RHO*DELTA*E7(JJ,1)*E15(JJ,1)*EU)
E(JJ,1)=AA
E(JJ,2)=BB

```

```

H(JJ)=CC
IF (DELTA) 634,635,634
634 CONTINUE
DO 645 NN=1,3,1
DO 644 MN=1,JJ,2
SG(MN,NN)=ABSF(E(MN,NN))
644 CONTINUE
645 CONTINUE
EMAX=AMAXAF(SG,1,99,1,MZ)
DO 647 MN=1,JJ,2
H(MN)=H(MN)/EMAX
DO 646 NN=1,3,1
E(MN,NN)=E(MN,NN)/EMAX
646 CONTINUE
647 CONTINUE
635 CONTINUE

```

```

C
C      COMPRESS MATRIX COEFFICIENTS
C

```

```

DO 569 I=2,JL,1
A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)
569 CONTINUE
A(1,2)=E(1,2)
A(1,3)=E(1,3)
Y(1)=H(1)
A(JL+1,1)=E(JJ,1)
A(JL+1,2)=E(JJ,2)
Y(JL+1)=H(JJ)
CALL MATRIX
DO 526 I=1,JJ,2
PON(I,1)=X(I/2+1)
526 CONTINUE

```

```

C
C      COMPUTE UPPER EDGE U-DIRECTION
C

```

```

DO 527 J=3,J0,2
IF (ICS) 528,540,528
528 CONTINUE
EU=EXPF(-U(J))/RW
GO TO 541
540 CONTINUE
IF (OMBA) 613,614,613
613 CONTINUE
DELTA=OMBA/L(J)
614 CONTINUE
EU=1.0/RW
541 CONTINUE
AA=RHO*(ZETA(J,JK)/(4.0*DU)*(E16(J,JK)*EU**2-DELTA-E16(J,JK)*
1EU**2/DU+
2E1(J,JK)*EU)-ETA(J,JK)/(4.0*DU)*(DELTA+E15(J,JK)*EU**2-E15(J,JK)*
3EU**2/DU-E11(J,JK)*E15(J,JK)+E12(J,JK)*E15(J,JK)+E2(J,JK)*EU)+
4THETA(J,JK)/(4.0*DU)*(E16(J,JK)*EU**2*(PC(J+2,JK)-PO(J-2,JK))+
5(PC(J+2,JK)-PC(J-2,JK))/(4.0*DU)+E5(J,JK)*E16(J,JK)*EU)+
6TAU(J,JK)/(4.0*DU)*(E15(J,JK)*EU**2*(PJ(J+2,JK)-PO(J-2,JK))/
7(4.0*DU)+E7(J,JK)*E15(J,JK)*EU))+CS*
8(RHO*EU**2*(ETA(J,JK)*E15(J,JK)-ZETA(J,JK)*E16(J,JK))/(DU*4.0))
BB=C(J,JK)*POR(J,JK)/(DT*2.0)+RHO*(ZETA(J,JK)*E16(J,JK)*EU**2/
1(2.0*DU**2)-ETA(J,JK)*E15(J,JK)*EU**2/(2.0*DU**2))
CC=RHO*(-ZETA(J,JK)/(4.0*DU)*(DELTA+E16(J,JK)*EU**2+E16(J,JK)*

```

```

1EU**2/DU+E1(J,JK)*EU+ETA(J,JK)/(4.0*DU)*(DELTA=E15(J,JK)*EU**2+
2E15(J,JK)*EU**2/DU+E12(J,JK)*E15(J,JK)-E11(J,JK)*E15(J,JK)+
3E2(J,JK)*EU)-THETA(J,JK)/(4.0*DU)*(E16(J,JK)*EU**2+1*PO(J+2,JK)-
4PO(J-2,JK))+PC(J+2,JK)-PC(J-2,JK))/(4.0*DU)+E5(J,JK)*E16(J,JK)*
5EU)-TAU(J,JK)/(4.0*DU)*(E15(J,JK)*EU**2/(4.0*DU)*(PO(J+2,JK)-
6PO(J-2,JK))+E7(J,JK)*E15(J,JK)*EU))+CS*
7(RHO*EU**2*(ZETA(J,JK)*E16(J,JK)-ETA(J,JK)*E15(J,JK))/(DU*4.0))
OD=C(J,JK)*POR(J,JK)*PC(J,JK)/(DT*2.0)+ZETA(J,JK)*(RHO*(DELTA*(
1E5(J,JK)*E16(J,JK)*EU+E16(J,JK)*EU**2*(PC(J+2,JK)-PC(J-2,JK))/
2(4.0*DU))+E9(J,JK)*E16(J,JK)+E16(J,JK)*EU**2*(PC(J+2,JK)-PC(J,JK)*
32.0+PC(J-2,JK))/(4.0*DU**2)+E1(J,JK)*EU*(PC(J+2,JK)-PC(J-2,JK))/
4(4.0*DU)+E1(J,JK)*E5(J,JK)))
7-ETA(J,JK)*(RHO*(DELTA*E7(J,JK)*E15(J,JK)*EU+E2(J,JK)*E7(J,JK)))
EE=THETA(J,JK)*RHO*E16(J,JK)*(EU**2*(PC(J+2,JK)-PC(J-2,JK)))/
1(4.0*DU)**2+E5(J,JK)*EU*(PC(J+2,JK)-PC(J-2,JK))/(4.0*DU))+
2THETA(J,JK)*PCR(J,JK)*(1.0-SQ(J,JK))*(PC(J,JK)-PCO(J,JK))/
3(DT*2.0)-CS*
4(ZETA(J,JK)*RHO*E16(J,JK)*EU**2*(PC(J+2,JK)-PC(J-2,JK))/(DU*4.0))
E(J,1)=AA
E(J,2)=BB
E(J,3)=CC
H(J)=OD+EE

```

527 CONTINUE

C
C
C

COMPUTE UPPER CORNERS U-DIRECTION

```

IF(ICS)542,543,542
542 CONTINUE
EU=EXP(-U(1))/RW
GO TO 544
543 CONTINUE
IF(CMBA)615,616,615
615 CONTINUE
DELTA=OMBA/U(1)
616 CONTINUE
EU=1.0/RW
544 CONTINUE
IF(MM-JK)477,478,477
478 CONTINUE
E(1,2)=E(3,1)*2.0
E(1,3)=0.0
H(1)=PO(1,JK)*E(3,1)*2.0
GO TO 479
477 CONTINUE
AA=C(1,JK)*POR(1,JK)/(2.0*DT)+RHO*EU**2/(2.0*DU**2)*(ZETA(1,JK)*
1E16(1,JK)-ETA(1,JK)*E15(1,JK))
BB=RHO*EU**2/(2.0*DU**2)*(ETA(1,JK)*E15(1,JK)-ZETA(1,JK)*
1E16(1,JK))
CC=C(1,JK)*POR(1,JK)*PO(1,JK)/(DT*2.0)+ZETA(1,JK)*(RHO*(DELTA*
1E5(1,JK)*EU+E16(1,JK)+E16(1,JK)*EU**2*(PC(3,JK)-PC(1,JK))/(2.0*
2DU**2)))
3 -ETA(1,JK)*(RHO*DELTA*E7(1,JK)*E15(1,JK)*EU)
E(1,2)=AA
E(1,3)=BB
H(1)=CC
479 CONTINUE
IF(ICS)545,546,545
545 CONTINUE
EU=EXP(-U(JJ))/RW
GO TO 547
546 CONTINUE
IF(CMBA)617,618,617

```



```

617 CONTINUE
    DELTA=OMBA/L(JJ)
618 CONTINUE
    EU=1.0/RW
547 CONTINUE
    BB=C(JJ,JK)*POR(JJ,JK)/(DT*2.0)+RHC*EU**2/(2.0*DU**2)*(ZETA(JJ,JK)
    1+E16(JJ,JK)-ETA(JJ,JK)*E15(JJ,JK))
    AA=RHO*EU**2/(2.0*DU**2)*(ETA(JJ,JK)*E15(JJ,JK)-ZETA(JJ,JK)*
    1E16(JJ,JK))
    CC=C(JJ,JK)*POR(JJ,JK)*PO(JJ,JK)/(DT*2.0)+ZETA(JJ,JK)*(RHO*(DELTA*
    1E5(JJ,JK)*E16(JJ,JK)+EU+E16(JJ,JK)*EU**2*(PC(JJ,JK)-PC(JJ,JK)))/
    2(2.0*DU**2)))
    3                                     -ETA(JJ,JK)*(RHO*DELTA+E7(JJ,JK)*C15
    4(JJ,JK)*EU)
    E(JJ,1)=AA
    E(JJ,2)=BB
    H(JJ)=CC
    IF (DELTA)636,637,636
636 CONTINUE
    DO 649 NN=1,3,1
    DO 648 MN=1,JJ,2
    SG(MN,NN)=ABSF(E(MN,NN))
648 CONTINUE
649 CONTINUE
    EMAX=AMAXAF(SG,1,99,1,MZ)
    DO 651 MN=1,JJ,2
    H(MN)=H(MN)/EMAX
    DO 650 NN=1,3,1
    E(MN,NN)=E(MN,NN)/EMAX
650 CONTINUE
651 CONTINUE
637 CONTINUE

```

```

C
C      COMPRESS MATRIX COEFFICIENTS
C

```

```

    DO 570 I=2,JL,1
    A(I,1)=E(2*I-1,1)
    A(I,2)=E(2*I-1,2)
    A(I,3)=E(2*I-1,3)
    Y(I)=H(2*I-1)
570 CONTINUE
    A(1,2)=E(1,2)
    A(1,3)=E(1,3)
    Y(1)=H(1)
    A(JL+1,1)=E(JJ,1)
    A(JL+1,2)=E(JJ,2)
    Y(JL+1)=H(JJ)
    CALL MATRIX
    DO 548 I=1,JJ,2
    PON(I,JK)=X(I/2+1)
548 CONTINUE
    IJZ=1
    RETURN
    END

```

```

•   LIST8
•   CARDS COLUMN      COEFZ
•   FORTRAN
SUBROUTINE COEFZ
C      MATRIX COEFFICIENT SUBROUTINE - Z-DIRECTION
USE STUFF
SBO=OMCC
SRS=0.0
BP=GMBD

```

```

CS=ICS
JL=JJ/2
JI=JK/2
ICOUNT=JI+1

C
C      ZERO OUT COEFFICIENT ARRAYS
C
DO 596 K=1,JK
DO 595 J=1,JJ
ZETA(J,K)=0.0
ETA(J,K)=0.0
THETA(J,K)=0.0
TAU(J,K)=0.0
C(J,K)=0.0
E(J,K)=0.0
E1(J,K)=0.0
E2(J,K)=0.0
E3(J,K)=0.0
E4(J,K)=0.0
E5(J,K)=0.0
E6(J,K)=0.0
E7(J,K)=0.0
E8(J,K)=0.0
E9(J,K)=0.0
E10(J,K)=0.0
E11(J,K)=0.0
E12(J,K)=0.0
E13(J,K)=0.0
E14(J,K)=0.0
E15(J,K)=0.0
E16(J,K)=0.0
595 CONTINUE
596 CONTINUE
DO 575 K=1,3
DO 574 J=1,JJ
A(J,K)=0.0
574 CONTINUE
575 CONTINUE
DO 576 J=1,JJ
Y(J)=0.0
H(J)=0.0
576 CONTINUE

C
C      COMPUTE E TERMS AND ETA, THETA, ZETA, AND TAU
C
DO 580 K=1,JK,2
DO 579 J=1,JJ,2
IF(ICS)581,582,581
581 CONTINUE
EU=EXP(-U(J))/RW
GO TO 583
582 CONTINUE
EU=1.0/(RW*U(J))
583 CONTINUE
IF(GAMG)101,102,101
102 CONTINUE
ZETA(J,K)=0.0
S80=-DBG/(BG(J,K)**2)
GO TO 103
101 CONTINUE
103 CONTINUE
THETA(J,K)=ZETA(J,K)*DBG/BG(J,K)
ZETA(J,K)=BG(J,K)*BG(J,K)/(RS(J,K)-BG(J,K)*BG(J,K))

```

```

ETA(J,K)=1.0-RS(J,K)/(RS(J,K)-BO(J,K)*BG(J,K))
E5(J,K)=GAMG*BG(J,K)*G*SINF(ALPHA)
E6(J,K)=GAMG*BG(J,K)*G*COSF(ALPHA)
E7(J,K)=(GAMG+RS(J,K)*GAMG)*G*SINF(ALPHA)/BO(J,K)
E8(J,K)=(GAMG+RS(J,K)*GAMG)*G*COSF(ALPHA)/BC(J,K)
E15(J,K)=PKG(J,K)/VISO(J,K)
E16(J,K)=PKG(J,K)/VISG(J,K)
IF (PO(J,K)-BP) 801,801,800
801 CONTINUE
TAU(J,K)=ETA(J,K)*DBO/BO(J,K)+DRS/(RS(J,K)-BO(J,K)*BG(J,K))
C(J,K)=SO(J,K)*TAU(J,K)+THETA(J,K)*(1.0-SO(J,K))
E11(J,K)=EU*E7(J,K)*DBO/BO(J,K)
E12(J,K)=G*GAMG*EU*DRS*SINF(ALPHA)/BO(J,K)
E13(J,K)=E8(J,K)*DBO/BO(J,K)
E14(J,K)=G*GAMG*DRS*COSF(ALPHA)/BO(J,K)
GO TO 803
800 CONTINUE
TAU(J,K)=ETA(J,K)*SBO/BO(J,K)+SRS/(RS(J,K)-BO(J,K)*BG(J,K))
C(J,K)=SO(J,K)*TAU(J,K)+THETA(J,K)*(1.0-SO(J,K))
E11(J,K)=EU*E7(J,K)*SBO/BO(J,K)
E12(J,K)=G*GAMG*EU*SRS*SINF(ALPHA)/BO(J,K)
E13(J,K)=E8(J,K)*SBO/BO(J,K)
E14(J,K)=G*GAMG*SRS*COSF(ALPHA)/BC(J,K)
803 CONTINUE
579 CONTINUE
580 CONTINUE
DO 591 K=1,JK,2
DO 590 J=3,J0,2
IF (ICS) 586,587,586
586 CONTINUE
EU=EXP(-U(J))/RW
GO TO 588
587 CONTINUE
EU=1.0/RW
588 CONTINUE
E1(J,K)=EU*((PKG(J+1,K)-PKG(J-1,K))/VISG(J,K)-PKG(J,K)*
1(VISG(J+1,K)-VISG(J-1,K))/VISG(J,K)**2)/(2.0*DU)
E2(J,K)=EU*((PKO(J+1,K)-PKO(J-1,K))/VISO(J,K)-PKO(J,K)*
1(VISO(J+1,K)-VISO(J-1,K))/VISO(J,K)**2)/(2.0*DU)
E9(J,K)=EU*GAMG*G*DBG*SINF(ALPHA)*(PG(J+1,K)-PG(J-1,K))/(2.0*DU)
590 CONTINUE
591 CONTINUE
DO 585 K=3,JP,2
DO 584 J=1,JJ,2
IF (ICS) 592,593,592
592 CONTINUE
EU=EXP(-U(J))/RW
GO TO 594
593 CONTINUE
EU=1.0/RW
594 CONTINUE
E3(J,K)=((PKG(J,K+1)-PKG(J,K-1))/VISG(J,K)-PKG(J,K)*
1(VISG(J,K+1)-VISG(J,K-1))/VISG(J,K)**2)/(2.0*OZ)
E4(J,K)=((PKO(J,K+1)-PKO(J,K-1))/VISO(J,K)-PKO(J,K)*
1(VISO(J,K+1)-VISO(J,K-1))/VISO(J,K)**2)/(2.0*OZ)
E10(J,K)=GAMG*G*DBG*COSF(ALPHA)*(PG(J,K+1)-PG(J,K-1))/(2.0*OZ)
584 CONTINUE
585 CONTINUE
DO 727 J=1,JJ,2
E3(J,1)=((PKG(J,3)-PKG(J,1))/VISG(J,1)-PKG(J,1)*
1(VISG(J,3)-VISG(J,1))/VISG(J,1)**2)/(2.0*OZ)
E3(J,JK)=((PKG(J,JK)-PKG(J,JP))/VISG(J,JK)-PKG(J,JK)*
1(VISG(J,JK)-VISG(J,JP))/VISG(J,JK)**2)/(2.0*OZ)

```

```

E4(J,1)=1(PK0(J,3)-PK0(J,1))/VISO(J,1)-PK0(J,1)*
1(VISO(J,3)-VISO(J,1))/VISO(J,1)**2/(2.0*OZ)
E4(J,JK)=(PKC(J,JK)-PKC(J,JP))/VISO(J,JK)-PK0(J,JK)*
1(VISO(J,JK)-VISO(J,JP))/VISO(J,JK)**2/(2.0*OZ)
E10(J,1)=GAMG*G*DBG*CSF(ALPHA)*PG(J,3)-PG(J,1)/(OZ*2.0)
E10(J,JK)=GAMG*G*DBG*CSF(ALPHA)*PG(J,JK)-PG(J,JP)/(OZ*2.0)
727 CONTINUE

```

C
C
C

COMPUTE Z-DIRECTION OIL PRESSURE COEFFICIENTS - INTERNAL

```

DO 550 J=3,JO,2
IF (CS1551,552,551)
551 CONTINUE
EU=EXP(-U(J))/RW
GO TO 553
552 CONTINUE
IF (CMBA,619,620,619)
619 CONTINUE
DELTA=OMBA/U(J)
620 CONTINUE
EU=1.0/RW
553 CONTINUE
DO 549 K=3,JP,2
AA=ZLAM*(ZETA(J,K)*(E3(J,K)/(4.0*OZ)-E10(J,K)/(4.0*OZ**2))+ETA
1(J,K)*(E15(J,K)/(4.0*OZ**2)+(E13(J,K)*E15(J,K)-E14(J,K)*E15(J,K)-
2E4(J,K))/(4.0*OZ))+THETA(J,K)*(E16(J,K)*((PO(J,K+2)-PO(J,K-2))+2.0
3(PC(J,K+2)-PC(J,K-2)))/(4.0*OZ)**2+E6(J,K)*E16(J,K)/(4.0*OZ))+
4TAU(J,K)*(E15(J,K)*(PO(J,K+2)-PO(J,K-2))/(4.0*OZ)**2+E8(J,K)*
5E15(J,K)/(4.0*OZ)))
BB=C(J,K)*PCR(J,K)/(DT*2.0)+ZLAM/(2.0*OZ**2)*(ZETA(J,K)*E16(J,K)-
1ETA(J,K)*E15(J,K))
CC=ZLAM*(-ZETA(J,K)*(E16(J,K)/(4.0*OZ**2)+E3(J,K)/(4.0*OZ))+ETA
1(J,K)*(E15(J,K)/(4.0*OZ**2)+(E14(J,K)*E15(J,K)-E13(J,K)*E15(J,K)+
2E4(J,K))/(4.0*OZ))-THETA(J,K)*(E16(J,K)*((PO(J,K+2)-PO(J,K-2))+2.0
3(PC(J,K+2)-PC(J,K-2)))/(4.0*OZ)**2+E6(J,K)*E16(J,K)/(4.0*OZ))-
4TAU(J,K)*(E15(J,K)*(PO(J,K+2)-PO(J,K-2))/(4.0*OZ)**2+E8(J,K)*
5E15(J,K)/(4.0*OZ)))
DD=C(J,K)*PCR(J,K)*PO(J,K)/(DT*2.0)+ZETA(J,K)*(RHO*(DELTA*E16(J,K)
1*EU**2*((PO(J+2,K)-PO(J-2,K))+PC(J+2,K)-PC(J-2,K)))/(4.0*OU)+
2E5(J,K)*E16(J,K)*DELTA*EU+E16(J,K)*EU**2*((PO(J+2,K)-PO(J,K)-2.0+
3PO(J-2,K))+PC(J+2,K)-PC(J,K)*2.0+PC(J-2,K)))/(4.0*OU**2)+L9(J,K)*
4E16(J,K)+
5E1(J,K)*EU*((PO(J+2,K)-PO(J-2,K))+PC(J+2,K)-PC(J-2,K))/(4.0*OU)+
6E1(J,K)*E5(J,K)+ZLAM*(E10(J,K)*E16(J,K)+E16(J,K)*PC(J,K+2)-
7PC(J,K)*2.0+PC(J,K-2))/(4.0*OZ**2)+E3(J,K)*PC(J,K+2)-PC(J,K-2))/
8(4.0*OZ)+E3(J,K)*E6(J,K))
EE=-ETA(J,K)*(RHO*(DELTA*E15(J,K)*EU**2-E11(J,K)*E15(J,K)+E12
1(J,K)*E15(J,K)+E2(J,K)*EU*(PC(J+2,K)-PO(J-2,K))/(4.0*OU)+DELTA*
2E7(J,K)*E15(J,K)*EU*E15(J,K)*EU**2*(PO(J+2,K)-PO(J,K)*2.0+PC(J-2,
3K))/(4.0*OU**2)+E2(J,K)*E7(J,K)+E4(J,K)*E6(J,K))+CS
4(ETA(J,K)*RHO*E15(J,K)*EU**2*(PC(J+2,K)-PC(J-2,K))/(OU*4.0)-
5ZETA(J,K)*RHO*E16(J,K)*EU**2*(PO(J+2,K)-PC(J-2,K)*PC(J+2,K)-
6PC(J-2,K))/(OU*4.0))
FF=RHO*(THETA(J,K)*(E16(J,K)*EU**2*((PO(J+2,K)-PO(J-2,K))**2+2.0
1(PC(J+2,K)-PO(J-2,K))*PC(J+2,K)-PC(J-2,K)))/(4.0*OU)**2+E5(J,K)*
2E16(J,K)*EU*(PO(J+2,K)-PO(J-2,K))/(4.0*OU))+RHO*TAU(J,K)*E15
3(J,K)*EU**2*(PO(J+2,K)-PO(J-2,K))**2/(4.0*OU)**2+E7(J,K)*E15(J,K)+
4EU*(PO(J+2,K)-PO(J-2,K))/(4.0*OU))+THETA(J,K)*RHO*(E16(J,K)*EU**2*
5(PC(J+2,K)-PC(J-2,K))**2/(4.0*OU)**2+E16(J,K)*E5(J,K)*EU
6(PC(J+2,K)-PC(J-2,K))/(4.0*OU))+THETA(J,K)*ZLAM*(E16(J,K)*
7(PC(J,K+2)-PC(J,K-2))**2/(4.0*OZ)**2+E10(J,K)*E6(J,K)+
8(PC(J,K+2)-PC(J,K-2))/(4.0*OZ))+THETA(J,K)*POR(J,K)*(1.0-SU(J,K))+
9(PC(J,K)-PCC(J,K))/(DT*2.0)

```

```

E(K,1)=AA
E(K,2)=BB
E(K,3)=CC
H(K)=DD+EE+FF

```

549 CONTINUE

C
C
C

COMPUTE LOWER EDGE Z-DIRECTION

```

AA=C(J,1)*PCR(J,1)/(DT*2.0)
BB=C.G
CC=C(J,1)*PCR(J,1)*PO(J,1)/(DT*2.0)*ZETA(J,1)*(RHO*(DELTA*E16(J,1)
1*EU**2*(PO(J+2,1)-PC(J-2,1))/(4.0*DU)+DELTA*E16(J,1)*EU**2*
2(PC(J+2,1)-PC(J-2,1))/(4.0*DU)+DELTA*E5(J,1)*E16(J,1)*EU*E16(J,1)+
3EU**2*((PO(J+2,1)-PC(J,1)+2.0*PO(J-2,1))*(PC(J+2,1)-PC(J,1)+2.0+
4PC(J-2,1)))/(4.0*DU**2)+E9(J,1)*E16(J,1)+E1(J,1)*EU*(PO(J+2,1)-
5PO(J-2,1))*(PC(J+2,1)-PC(J-2,1))/(4.0*DU)+E1(J,1)*E5(J,1)))
DD=-ETA(J,1)*(RHO*(DELTA*E7(J,1)*E15(J,1)*EU+DELTA*E15(J,1)*EU**2*
1(PO(J+2,1)-PO(J-2,1))/(4.0*DU)+E15(J,1)*EU**2*(PO(J+2,1)-PO(J,1)+
22.0*PO(J-2,1))/(4.0*DU**2)+E12(J,1)*E15(J,1)-E11(J,1)*E15(J,1)+
3E2(J,1)*EU*(PO(J+2,1)-PO(J-2,1))/(4.0*DU)+E2(J,1)*E7(J,1)))
EE=THE TA(J,1)*RHO*(E16(J,1)*EU**2*((PO(J+2,1)-PO(J-2,1))*2+2.0*
1(PO(J+2,1)-PO(J-2,1))*(PC(J+2,1)-PC(J-2,1))/(4.0*DU))*2+E5(J,1)*
2E16(J,1)*EU*(PO(J+2,1)-PO(J-2,1))/(4.0*DU)+TAU(J,1)*RHO*(E15(J,1)
3*EU**2*(PO(J+2,1)-PO(J-2,1))*2/(4.0*DU)**2+E7(J,1)*E15(J,1)*EU*
4(PO(J+2,1)-PO(J-2,1))/(4.0*DU))+CS
5(ETA(J,1)*RHO*E15(J,1)*EU**2*(PC(J+2,1)-PC(J-2,1))/(DU*4.0)-
6ZETA(J,1)*RHO*E16(J,1)*EU**2*(PO(J+2,1)-PO(J-2,1)*PC(J+2,1)-
7PC(J-2,1))/(DU*4.0))
FF=THE TA(J,1)*RHO*E16(J,1)*EU**2*(PC(J+2,1)-PC(J-2,1))*2/
1(4.0*DU)**2+E5(J,1)*EU*(PC(J+2,1)-PC(J-2,1))/(4.0*DU))+
2THE TA(J,1)*PCR(J,1)*(1.0-SU(J,1))*(PC(J,1)-PC(J,1))/
3(DT*2.0)
E(1,2)=AA
E(1,3)=BB
H(1)=CC+DU+EE+FF

```

C
C
C

COMPUTE UPPER EDGE Z-DIRECTION

```

AA=0.0
BB=C(J,JK)*PCR(J,JK)/(DT*2.0)-AA
CC=C(J,JK)*PCR(J,JK)*PC(J,JK)/(DT*2.0)*ZETA(J,JK)*(RHO*(DELTA*
1E16(J,JK)*EU**2*(PO(J+2,JK)-PC(J-2,JK))/(4.0*DU)+DELTA*E16(J,JK)*
2EU**2*(PC(J+2,JK)-PC(J-2,JK))/(4.0*DU)+DELTA*E5(J,JK)*E16(J,JK)+
3EU*E16(J,JK)*EU**2*((PO(J+2,JK)-PC(J,JK)+2.0*PO(J-2,JK))*(PC(J+2,
4JK)-PC(J,JK)+2.0*PC(J-2,JK)))/(4.0*DU**2)+E9(J,JK)*E16(J,JK)+
5E1(J,JK)*EU*(PO(J+2,JK)-PO(J-2,JK))*(PC(J+2,JK)-PC(J-2,JK)))/
6(4.0*DU)+E1(J,JK)*E5(J,JK)))
DD=-ETA(J,JK)*(RHO*(DELTA*E7(J,JK)*E15(J,JK)*EU+DELTA*E15(J,JK)*
1EU**2*(PO(J+2,JK)-PC(J-2,JK))/(4.0*DU)+E15(J,JK)*EU**2*(PO(J+2,JK)
2-PO(J,JK)+2.0*PO(J-2,JK))/(4.0*DU**2)+E12(J,JK)*E15(J,JK)-E11
3(J,JK)*E15(J,JK)+E2(J,JK)*EU*(PO(J+2,JK)-PO(J-2,JK))/(4.0*DU)+
4E2(J,JK)*E7(J,JK)))
EE=THE TA(J,JK)*RHO*(E16(J,JK)*EU**2*((PO(J+2,JK)-PO(J-2,JK))*2+
*2.0*
1(PO(J+2,JK)-PO(J-2,JK))*(PC(J+2,JK)-PC(J-2,JK)))/(4.0*DU))*2+
2E5(J,JK)*E16(J,JK)*EU*(PC(J+2,JK)-PC(J-2,JK))/(4.0*DU)+TAU(J,JK)*
3RHO*(E15(J,JK)*EU**2*(PO(J+2,JK)-PC(J-2,JK))*2/(4.0*DU)**2+
4E7(J,JK)*E15(J,JK)*EU*(PO(J+2,JK)-PO(J-2,JK))/(4.0*DU))+CS
5(ETA(J,JK)*RHO*E15(J,JK)*EU**2*(PC(J+2,JK)-PO(J-2,JK))/(DU*4.0)-
6ZETA(J,JK)*RHO*E16(J,JK)*EU**2*(PC(J+2,JK)-PO(J-2,JK)*PC(J+2,JK)-
7PC(J-2,JK))/(DU*4.0))

```

```

FF=THETA(J,JK)*RHO*E16(J,JK)*(EU**2*(PC(J+2,JK)-PC(J-2,JK))*2/
1(4.0*DU)**2+E5(J,JK)*EU*(PC(J+2,JK)-PC(J-2,JK))/(4.0*DU))+
2THETA(J,JK)*POR(J,JK)*(1.0-SO(J,JK))*(PC(J,JK)-PCO(J,JK))/
3(DT*2.0)
E(JK,1)=AA
E(JK,2)=BB
H(JK)=CC+DD*EE+FF

```

C
C
C COMPRESS MATRIX COEFFICIENTS

```

DO 571 I=2,JI,1
A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)
571 CONTINUE
A(1,2)=E(1,2)
A(1,3)=E(1,3)
Y(1)=H(1)
A(JI+1,1)=E(JK,1)
A(JI+1,2)=E(JK,2)
Y(JI+1)=H(JK)
CALL MATRIX
DO 554 I=1,JK,2
PCN(J,I)=X(I/2+1)
554 CONTINUE
550 CONTINUE

```

C
C
C COMPUTE OUTER EDGE Z-DIRECTION

```

IF(ICS)555,556,555
555 CONTINUE
EU=EXP(-U(JJ))/RW
GO TO 557
556 CONTINUE
IF(CMBA)621,622,621
621 CONTINUE
DELTA=OMBA/U(JJ)
622 CONTINUE
EU=1.0/RW
557 CONTINUE
DO 558 K=3,JP,2
AA=ZLAM*(ZETA(JJ,K)*(E3(JJ,K)/(4.0*OZ)-E16(JJ,K)/(4.0*OZ**2))+
1ETA(JJ,K)*(E15(JJ,K)/(4.0*OZ**2)+(E13(JJ,K)*E15(JJ,K)-E14(JJ,K)*
2E15(JJ,K)-E4(JJ,K))/(4.0*OZ))+THETA(JJ,K)*(E16(JJ,K)*(PO(JJ,K+2)-
3PO(JJ,K-2)))+
* 2.0*(PC(JJ,K+2)-PC(JJ,K-2))/(4.0*OZ)**2+E6(JJ,K)*E16
4(JJ,K)/(4.0*OZ))+TAU(JJ,K)*(E15(JJ,K)*(PO(JJ,K+2)-PO(JJ,K-2))/
5(4.0*OZ)**2+E8(JJ,K)*E15(JJ,K)/(4.0*OZ)))
RB=C(JJ,K)*POR(JJ,K)/(DT*2.0)+ZLAM/(2.0*OZ**2)*(ZETA(JJ,K)*
1E16(JJ,K)-ETA(JJ,K)*E15(JJ,K))
CC=ZLAM*(-ZETA(JJ,K)*(E16(JJ,K)/(4.0*OZ**2)+E3(JJ,K)/(4.0*OZ))+
1ETA(JJ,K)*(E15(JJ,K)/(4.0*OZ**2)+(E14(JJ,K)*E15(JJ,K)-E13(JJ,K)*
2E15(JJ,K))+
3E4(JJ,K)/(4.0*OZ))-THETA(JJ,K)*(E16(JJ,K)*(PO(JJ,K+2)-PO(JJ,K-2)
*))+2.0*
4 (PC(JJ,K+2)-PC(JJ,K-2))/(4.0*OZ)**2+E6(JJ,K)*E16(JJ,K)/(4.0*
5OZ))-TAU(JJ,K)*(E15(JJ,K)*(PO(JJ,K+2)-PO(JJ,K-2))/(4.0*OZ)**2+
6E8(JJ,K)*E15(JJ,K)/(4.0*OZ)))
DD=C(JJ,K)*POR(JJ,K)*PO(JJ,K)/(DT*2.0)+ZETA(JJ,K)*(RHO*(DELTA*
1E5(JJ,K)*E16(JJ,K)*EU+E16(JJ,K)*EU**2*(PO(JJ,K)-PC(JJ,K))+
2(PC(JJ,K)-PCO(JJ,K)))/(2.0*DU**2))+ZLAM*(E16(JJ,K)*(PC(JJ,K+2)-

```

```

3PC(JJ,K)*2.0+PC(JJ,K-2))/(4.0*DZ**2)+E15(JJ,K)*E16(JJ,K)+
*E3(JJ,K)*(PC(JJ,K+2)-PC(JJ,K-2))/(DZ*4.0)+E3(JJ,K)*
4E6(JJ,K))-ETA(JJ,K)*(RHO*(DELTA*E7(JJ,K)*E15(JJ,K)*EU+E15(JJ,K)*
5EU**2*(PO(JJ,K)-PO(JJ,K))/(2.0*DU**2))+ZLAM*E4(JJ,K)*E8(JJ,K))
EE=THETA(JJ,K)*RHO*(E16(JJ,K)*(PC(JJ,K+2)-PC(JJ,K-2))*2/
1(4.0*DZ)**2+E16(JJ,K)*E6(JJ,K)*(PC(JJ,K+2)-PC(JJ,K-2))/(4.0*DZ))+
2THETA(JJ,K)*POR(JJ,K)*(1.0-SQ(JJ,K))*(PC(JJ,K)-PC0(JJ,K))/
3(DT*2.0)
E(K,1)=AA
E(K,2)=BB
E(K,3)=CC
H(K)=DD+EE
558 CONTINUE
C
C      COMPUTE OUTER CORNERS   Z-DIRECTION
C
AA=C(JJ,1)*POR(JJ,1)/(DT*2.0)
BB=0.0
CC=C(JJ,1)*POR(JJ,1)*PC(JJ,1)/(DT*2.0)+ZETA(JJ,1)*(RHO*(DELTA*
1E5(JJ,1)*E16(JJ,1)*EU+E16(JJ,1)*EU**2*(PC(JJ,1)-PO(JJ,1))*(PC
2(JJ,1)-PC(JJ,1)))/(2.0*DU**2)))
3      -ETA(JJ,1)*RHO*(DELTA*E7(JJ,1)*E15(JJ,1)*EU+E15
4(JJ,1)*EU**2*(PO(JJ,1)-PO(JJ,1))/(2.0*DU**2))
E(1,2)=AA
E(1,3)=BB
H(1)=CC
AA=0.0
BB=C(JJ,JK)*POR(JJ,JK)/(DT*2.0)
CC=C(JJ,JK)*POR(JJ,JK)*PO(JJ,JK)/(DT*2.0)+ZETA(JJ,JK)*(RHO*(DELTA*
1E5(JJ,JK)*E16(JJ,JK)*EU+E16(JJ,JK)*EU**2*(PC(JJ,JK)-PO(JJ,JK))*
2(PC(JJ,JK)-PC(JJ,JK)))/(2.0*DU**2)))
3      -ETA(JJ,JK)*RHO*(DELTA*E7(JJ,JK)*E15
4(JJ,JK)*EU+E15(JJ,JK)*EU**2*(PO(JJ,JK)-PO(JJ,JK))/(2.0*DU**2))
E(JK,1)=AA
E(JK,2)=BB
H(JK)=CC
C
C      COMPRESS MATRIX COEFFICIENTS
C
DO 572 I=2,JI,1
A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)
572 CONTINUE
A(1,2)=E(1,2)
A(1,3)=E(1,3)
Y(1)=H(1)
A(JI+1,1)=E(JK,1)
A(JI+1,2)=E(JK,2)
Y(JI+1)=H(JK)
CALL MATRIX
DO 559 I=1,JK,2
PON(JJ,I)=X(I/2+1)
559 CONTINUE
C
C      COMPUTE INNER EDGE   Z-DIRECTION
C
IF(ICS)560,561,560
560 CONTINUE
EU=EXP(-U(1))/RW
GO TO 562
561 CONTINUE

```

```

        IF (OMBA) 623, 624, 623
623 CONTINUE
        DELTA=OMBA/U(1)
624 CONTINUE
        EU=1.0/RW
562 CONTINUE
        DO 563 K=3, JP, 2
            IF (K-MO) 564, 566, 565
565 CONTINUE
            IF (K-MM) 566, 566, 564
564 CONTINUE
            AA=ZLAM*(ZETA(1,K)*(E3(1,K)/(4.0*DZ)-E16(1,K)/(4.0*DZ**2))+
            IETA(1,K)*(E15(1,K)/(4.0*DZ**2)+(E13(1,K)*E15(1,K)-E14(1,K)*
            2E15(1,K)-E4(1,K))/(4.0*DZ))+THETA(1,K)*(E16(1,K)*((PO(1,K+2)-
            3PO(1,K-2))+2.0*
            * (PC(1,K+2)-PC(1,K-2)))/(4.0*DZ)**2+E6(1,K)*E15(1,K)/
            4(4.0*DZ))+TAU(1,K)*(E15(1,K)*(PO(1,K+2)-PC(1,K-2))/(4.0*DZ)**2+
            5E4(1,K)*E15(1,K)/(4.0*DZ)))
            BB=C(1,K)*PCR(1,K)/(DT*2.0)+ZLAM/(2.0*DZ**2)*(ZETA(1,K)*E16(1,K)-
            IETA(1,K)*E15(1,K))
            CC=ZLAM*(-ZETA(1,K)*(E16(1,K)/(4.0*DZ**2)+E3(1,K)/(4.0*DZ))+
            IETA(1,K)*(E15(1,K)/(4.0*DZ**2)+(E14(1,K)*E15(1,K)-E13(1,K)*
            2E15(1,K)+E4(1,K))/(4.0*DZ))-THETA(1,K)*(E16(1,K)*((PO(1,K+2)-
            3PC(1,K-2))+2.0*
            * (PC(1,K+2)-PC(1,K-2)))/(4.0*DZ)**2+E6(1,K)*E16(1,K)/
            4(4.0*DZ))-TAU(1,K)*(E15(1,K)*(PO(1,K+2)-PC(1,K-2))/(4.0*DZ)**2+
            5E4(1,K)*E15(1,K)/(4.0*DZ)))
            DD=C(1,K)*POR(1,K)*PO(1,K)/(DT*2.0)+ZETA(1,K)*(RHO*(DELTA*
            IET5(1,K)*E16(1,K)*EU+E16(1,K)*EU**2*((PO(3,K)-PO(1,K))+(PC(3,K)-
            2PC(1,K)))/(2.0*DU**2))+ZLAM*(E16(1,K)*(PC(1,K+2)-PC(1,K)*2.0+
            3PC(1,K-2))/(4.0*DZ**2)+E10(1,K)*E16(1,K)+E3(1,K)*E6(1,K)
            *+E3(1,K)*(PC(1,K+2)-PC(1,K-2))/(DZ*4.0)))-
            4ETA(1,K)*(RHO*(DELTA*E7(1,K)*E15(1,K)*EU+E15(1,K)*EU**2*(PO(3,K)-
            5PO(1,K))/(2.0*DU**2))+ZLAM*E4(1,K)*E8(1,K))
            EE=THETA(1,K)*RHO*(E16(1,K)*PC(1,K+2)-PC(1,K-2))*2/
            1(4.0*DZ)**2+E16(1,K)*E6(1,K)*(PC(1,K+2)-PC(1,K-2))/(4.0*DZ))+
            2THETA(1,K)*PCR(1,K)*(1.0-SO(1,K))*(PC(1,K)-PCQ(1,K))/
            3(DT*2.0)
            E(K,1)=AA
            E(K,2)=BB
            E(K,3)=CC
            H(K)=DD+EE
            GO TO 563
566 CONTINUE
            E(K,1)=0.0
            E(K,2)=1.0
            E(K,3)=0.0
            H(K)=PO(1,K)
563 CONTINUE

```

```

C
C      COMPUTE INNER CORNERS   Z-DIRECTION
C

```

```

        IF (MO-1) 474, 475, 474
475 CONTINUE
        E(1,2)=1.0
        E(1,3)=0.0
        H(1)=PO(1,1)
        GO TO 476
474 CONTINUE
        AA=C(1,1)*PCR(1,1)/(DT*2.0)
        BB=0.0
        CC=C(1,1)*PCR(1,1)*PO(1,1)/(DT*2.0)+ZETA(1,1)*(RHO*(DELTA*E5(1,1)*
        IET5(1,1)*EU+E16(1,1)*EU**2*((PO(3,1)-PO(1,1))+(PC(3,1)-PC(1,1)))/

```



```

2(2.0*DU**2)))-
3ETA(1,1)*RHO*(DELTA*E7(1,1)*E15(1,1)*EJ+E15(1,1)*EU**2*(PO(3,1)-
4PO(1,1))/(2.0*DU**2))
E(1,2)=AA
E(1,3)=BB
H(1)=CC
476 CONTINUE
IF (MM-JK)477,478,477
478 CONTINUE
E(JK,1)=0.0
E(JK,2)=1.0
H(JK)=PC(1,JK)
GO TO 479
477 CONTINUE
BB=C(1,JK)*POR(1,JK)/(DT*2.0)
AA=0.0
CC=C(1,JK)*POR(1,JK)*PC(1,JK)/(DT*2.0)+ZETA(1,JK)*(RHO*(DELTA*
1E5(1,JK)*E16(1,JK)*EU+E16(1,JK)*EU**2*((PC(3,JK)-PO(1,JK))+(PC
2(3,JK)-PC(1,JK)))/(2.0*DU**2)))
3      -ETA(1,JK)*RHO*(DELTA*E7(1,JK)*E15(1,JK)*EU+
4E15(1,JK)*EU**2*(PO(3,JK)-PO(1,JK))/(2.0*DU**2))
E(JK,1)=AA
E(JK,2)=BB
H(JK)=CC
479 CONTINUE
C
C      COMPRESS MATRIX COEFFICIENTS
C
DO 573 I=2,JI,1
A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)
573 CONTINUE
A(1,2)=E(1,2)
A(1,3)=E(1,3)
Y(1)=H(1)
A(JI+1,1)=E(JK,1)
A(JI+1,2)=E(JK,2)
Y(JI+1)=H(JK)
CALL MATRIX
DO 567 I=1,JK,2
PON(1,I)=X(I/2+1)
567 CONTINUE
IUZ=0
RETURN
END
*
*      LISTB
*      CARDS COLUMN
*      FORTRAN          MATRIX
SUBROUTINE MATRIX
C
C      VON HOLT TRI-DIAGONAL MATRIX SOLVER
C
USE STUFF
ITER=0
M=ICOUNT
DO 1 I=1,M
X(I)=0.0
DO 1 JA=1,3
B(I,JA+1)=A(I,JA)
1 CONTINUE

```

```

      DO 4 I=1,M-1
      IF (ABSF(B(I,3))-ABSF(B(I+1,2)))3,2,2
2  CONTINUE
      B(I,2)=1.0/B(I,3)
      B(I,1)=-B(I+1,2)*B(I,2)
      B(I+1,3)=B(I+1,3)+B(I,1)*B(I,4)
      B(I,3)=-B(I,4)
      B(I,4)=0.0
      B(I,5)=0.0
      GO TO 4
3  CONTINUE
      B(I,2)=1.0/B(I+1,2)
      B(I,1)=-B(I,3)*B(I,2)
      B(I,3)=-B(I+1,3)
      B(I+1,3)=B(I,4)+B(I,1)*B(I+1,3)
      B(I,4)=-B(I+1,4)
      B(I+1,4)=B(I+1,4)*B(I,1)
      B(I,5)=7.0
4  CONTINUE
      B(M,2)=1.0/B(M,3)
      S=0.0
      DO 5 I=1,M
      RES(I)=Y(I)
      S=S+Y(I)*.2
5  CONTINUE
      IF(S)6,15,6
6  CONTINUE
      DO 9 I=1,M-1
      IF(B(I,5))8,7,8
7  CONTINUE
      RES(I+1)=RES(I+1)+B(I,1)*RES(I)
      GO TO 9
8  CONTINUE
      TEMP=RES(I+1)
      RES(I+1)=RES(I)+B(I,1)*TEMP
      RES(I)=TEMP
9  CONTINUE
      D(M)=B(M,2)*RES(M)
      D(M-1)=B(M-1,2)*(RES(M-1)+B(M-1,3)*D(M))
      DO 10 I=2,M-1
      D(M-1)=B(M-1,2)*(RES(M-1)+B(M-1,3)*C(M-I+1)+B(M-1,4)*C(M-1+2))
10 CONTINUE
      DO 11 I=1,M
      X(I)=X(I)+D(I)
11 CONTINUE
      RES(I)=Y(I)
      DO 12 I=1,M-1
      RES(I)=RES(I)-A(I,2)*X(I)-A(I,3)*X(I+1)
      RES(I+1)=Y(I+1)-A(I+1,1)*X(I)
12 CONTINUE
      RES(M)=RES(M)-A(M,2)*X(M)
      TEMP=0
      DO 13 I=1,M
      TEMP=TEMP+RES(I)*RES(I)
13 CONTINUE
      ITER=ITER+1
      IF(S-TEMP)15,15,14
14 CONTINUE
      S=TEMP
      GO TO 6
15 CONTINUE
      DO 18 I=1,M,1
      IF(X(I))19,19,20

```

```

19 CONTINUE
   QARB=1.0
   GO TO 21
20 CONTINUE
18 CONTINUE
21 CONTINUE
   IF (CMBB) 16,17,16
16 CONTINUE
   WRITE OUTPUT TAPE 3,1920,(X(1),Y(1),RES(1),(A(1,1)),11=1,3),
   11=1,ICOUNT)
   WRITE OUTPUT TAPE 3,1922,ITER
   WRITE OUTPUT TAPE 3,1921,((C(NO,NC),NC=1,JJ,2),NQ=1,JK,2)
17 CONTINUE
   RETURN
1919 FORMAT(1H0/(6E19.8))
1920 FORMAT(1H1/(6E19.8))
1921 FORMAT(1H0/(6E19.8))
1922 FORMAT(1H0/12HITERATIONS =110)
   END
*   LISTB
*   CARDS CGLUMN
*   FCRTAN      CUAD
   SUBROUTINE QUAD
C
C   QUADRATIC INTERPOLATION SUBROUTINE
C
   USE STUFF
   XAX=1.0
   IF (CMBB) 26,27,26
26 CONTINUE
   WRITE OUTPUT TAPE 3,1399
1399 FORMAT(1H1/20X,23HQUADRATIC INTERPOLATION/)
27 CONTINUE
C
C   COMPUTE INTERPOLATED PRESSURES AND DERIVATIVES ALONG Z
C
   DO 16 J=3,5,2
   DO 15 K=3,JP,4
   EFA=PON(J,K-2)/((Z(K-2)-Z(K))*(Z(K-2)-Z(K+2)))
   EFH= PG(J,K-2)/((Z(K-2)-Z(K))*(Z(K-2)-Z(K+2)))
   EFB=PON(J,K)/((Z(K)-Z(K+2))*(Z(K)-Z(K-2)))
   EFN= PG(J,K)/((Z(K)-Z(K+2))*(Z(K)-Z(K-2)))
   EFC=PON(J,K+2)/((Z(K+2)-Z(K-2))*(Z(K+2)-Z(K)))
   EFG= PG(J,K+2)/((Z(K+2)-Z(K-2))*(Z(K+2)-Z(K)))
   EFD=EFA*(Z(K)+Z(K+2))
   EFP=EFH*(Z(K)+Z(K+2))
   EFE=EFB*(Z(K+2)+Z(K-2))
   EFO=EFN*(Z(K+2)+Z(K-2))
   EFF=EFC*(Z(K-2)+Z(K))
   EFR=EFO*(Z(K-2)+Z(K))
   EFG=EFA*(Z(K)+Z(K+2))
   EFS=EFH*(Z(K)+Z(K+2))
   EFH=EFB*(Z(K+2)+Z(K-2))
   EFT=EFN*(Z(K+2)+Z(K-2))
   EFI=EFC*(Z(K-2)+Z(K))
   EFU=EFO*(Z(K-2)+Z(K))
   COAO=EFA+EFB+EFC
   COAG=EFH+EFN+EFO
   COBC=EFD+EFE+EFF
   COBG=EFP+EFQ+EFR
   COCD=EFG+EFH+EFI
   COCG=EFS+EFT+EFU
   PON(J,K+1)=COAO*Z(K+1)**2-COBC*Z(K+1)+COCC

```

```

      PG(J,K+1)=COAG*Z(K+1)*2-CO8G*Z(K+1)+COCG
      DPRO(J,K+1)=COAO*Z(K+1)*2.0-CO8O
      DPRG(J,K+1)=COAG*Z(K+1)*2.0-CO8G
      PON(J,K-1)=COAO*Z(K-1)*2-CO8C*Z(K-1)+CUCO
      PG(J,K-1)=COAG*Z(K-1)*2-CO8G*Z(K-1)+CUCG
      DPRO(J,K-1)=COAO*Z(K-1)*2.0-CO8O
      DPRG(J,K-1)=COAG*Z(K-1)*2.0-CO8G
      IF(ICMBB)24,25,24
24  CONTINUE
      WRITE OUTPUT TAPE 3,1400,COAO,CO8O,COCO,PON(J,K-1),UPRO(J,K-1),
      1PON(J,K+1),DPRO(J,K+1)
1400 FORMAT(1H0,7E15.5)
25  CONTINUE
15  CONTINUE
16  CONTINUE

```

C
C
C

COMPUTE INTERPOLATED PRESSURES AND DERIVATIVES ALONG U

```

      DO 18 J=2,4,2
      DC 17 K=1,JK,1
      IF (ICS)21,22,21
21  CONTINUE
      EFA=PON(J-1,K)/((EXPF(U(J-1))-EXPF(U(J+1)))*(EXPF(U(J-1))-
      1EXPF(U(J+3))))
      EFM= PG(J-1,K)/((EXPF(U(J-1))-EXPF(U(J+1)))*(EXPF(U(J-1))-
      1EXPF(U(J+3))))
      EFB=PON(J+1,K)/((EXPF(U(J+1))-EXPF(U(J+3)))*(EXPF(U(J+1))-
      1EXPF(U(J-1))))
      EFN= PG(J+1,K)/((EXPF(U(J+1))-EXPF(U(J+3)))*(EXPF(U(J+1))-
      1EXPF(U(J-1))))
      EFC=PON(J+3,K)/((EXPF(U(J+3))-EXPF(U(J-1)))*(EXPF(U(J+3))-
      1EXPF(U(J+1))))
      EFO= PG(J+3,K)/((EXPF(U(J+3))-EXPF(U(J-1)))*(EXPF(U(J+3))-
      1EXPF(U(J+1))))
      EFD=EFA*(EXPF(U(J+1))+EXPF(U(J+3)))
      EFP=EFM*(EXPF(U(J+1))+EXPF(U(J+3)))
      EFE=EFB*(EXPF(U(J+3))+EXPF(U(J-1)))
      EFQ=EFN*(EXPF(U(J+3))+EXPF(U(J-1)))
      EFF=EFC*(EXPF(U(J-1))+EXPF(U(J+1)))
      EFR=EFO*(EXPF(U(J-1))+EXPF(U(J+1)))
      EFG=EFA*EXPF(U(J+1))*EXPF(U(J+3))
      EFS=EFM*EXPF(U(J+1))*EXPF(U(J+3))
      EFH=EFB*EXPF(U(J+3))*EXPF(U(J-1))
      EFT=EFN*EXPF(U(J+3))*EXPF(U(J-1))
      EFI=EFC*EXPF(U(J-1))*EXPF(U(J+1))
      EFU=EFO*EXPF(U(J-1))*EXPF(U(J+1))
      COAO=EFA+EFB+EFC
      COAG=EFM+EFN+EFO
      CO8C=EFD+EFE+EFF
      CO8G=EFP+EFQ+EFR
      COCO=EFG+EFH+EFI
      COCG=EFS+EFT+EFU
      PON(J,K)=COAO*EXPF(U(J)*2.0)-CO8C*EXPF(U(J))+COCO
      PG(J,K)=COAG*EXPF(U(J)*2.0)-CO8G*EXPF(U(J))+COCG
      DPRO(J,K)=CCAO*EXPF(U(J))*2.0-CO8O
      DPRG(J,K)=CCAG*EXPF(U(J))*2.0-CO8G
      GO TO 23
22  CONTINUE
      EFA=PON(J-1,K)/((U(J-1)-U(J+1))*(U(J-1)-U(J+3)))
      EFM= PG(J-1,K)/((U(J-1)-U(J+1))*(U(J-1)-U(J+3)))
      EFB=PON(J+1,K)/((U(J+1)-U(J+3))*(U(J+1)-U(J-1)))
      EFN= PG(J+1,K)/((U(J+1)-U(J+3))*(U(J+1)-U(J-1)))
      EFC=PON(J+3,K)/((U(J+3)-U(J-1))*(U(J+3)-U(J+1)))

```

```

EFO= PG(J+3,K)/((U(J+3)-U(J-1))*(U(J+3)-U(J+1)))
EFO=EFA*(U(J+1)+U(J+3))
EFP=EFM*(U(J+1)+U(J+3))
EFE=EFB*(U(J+3)+U(J-1))
EFO=EFN*(U(J+3)+U(J-1))
EFF=EFC*(U(J-1)+U(J+1))
EFR=EFO*(U(J-1)+U(J+1))
EFG=EFA*U(J+1)+U(J+3)
EFS=EFM*U(J+1)+U(J+3)
EFH=EFB*U(J+3)+U(J-1)
EFT=EFN*U(J+3)+U(J-1)
EFI=EFC*U(J-1)+U(J+1)
EFU=EFO*U(J-1)+U(J+1)
COAO=EFA+EFU+EFC
COAG=EFM+EFN+EFO
COBC=EFO+EFE+EFF
COBG=EFP+EFG+EFR
COCO=FG+EFH+EFI
COCS=EFS+EFT+EFU
PON(J,K)=COAO*U(J)**2-COBN*U(J)+CCCC
PG(J,K)=COAG*U(J)**2-COBG*U(J)+COCG
DPRO(J,K)=CAAO*U(J)*2.0-COBO
DPRG(J,K)=COAG*U(J)*2.0-COBG
23 CONTINUE
IF (OMBB) 19,20,19
19 CONTINUE
WRITE OUTPUT TAPE 3,1401,COAO,COBO,COCO,PON(J,K),DPRO(J,K)
1401 FORMAT(1H0,5E15.5)
20 CONTINUE
17 CONTINUE
18 CONTINUE
RETURN
END
SUBROUTINE SOCUAD
USE STUFF
XAX=1.0

```

```

C
C      COMPUTE EXTRAPOLATED SATURATIONS ON WELLBORE
C

```

```

DO 17 K=1,JK,1
IF (ICS) 21,22,21
21 CONTINUE
EFA=SON(2 ,K)/((EXP(U(2 ))-EXP(U(4 )))*(EXP(U(2 ))-
EXP(U(6 ))))
EFB=SON(4 ,K)/((EXP(U(4 ))-EXP(U(6 )))*(EXP(U(4 ))-
EXP(U(2 ))))
EFC=SON(6 ,K)/((EXP(U(6 ))-EXP(U(2 )))*(EXP(U(6 ))-
EXP(U(4 ))))
EFO=EFA*(EXP(U(4 ))+EXP(U(6 )))
EFE=EFB*(EXP(U(6 ))+EXP(U(2 )))
EFF=EFC*(EXP(U(2 ))+EXP(U(4 )))
EFG=EFA*EXP(U(4 ))+EXP(U(6 ))
EFH=EFB*EXP(U(6 ))+EXP(U(2 ))
EFI=EFC*EXP(U(2 ))+EXP(U(4 ))
CCAS=EFA+EFB+EFC
CCBS=EFO+EFE+EFF
COCs=EFG+EFH+EFI
SON(1 ,K)=COAS*EXP(U(1 ))*2.0)-CCBS*EXP(U(1 ))+COCO
GO TO 23
22 CONTINUE
EFA=SON(2 ,K)/(U(2 )-U(4 ))*(U(2 )-U(6 ))
EFB=SON(4 ,K)/(U(4 )-U(6 ))*(U(4 )-U(2 ))
EFC=SON(6 ,K)/(U(6 )-U(2 ))*(U(6 )-U(4 ))

```

```

EFD=EFA*(U(4 )+U(6 ))
EFE=EFB*(U(6 )+U(2 ))
EFF=EFC*(U(2 )+U(4 ))
EFG=EFA*U(4 )*U(6 )
EFH=EFB*U(6 )*U(2 )
EFI=EFC*U(2 )*U(4 )
COAS=EFA+EFB+EFC
COBS=EFD+EFE+EFF
COCS=EFG+EFH+EFI
SON(1 ,K)=COAS*U(1 )**2-COBS*U(1 )+COCS
23 CONTINUE
17 CONTINUE
RETURN
END

* LIST8
* CARDS COLUMN
* FORTRAN KTIME
C
C HETEROGENEOUS RESERVOIR - DELTA T SELECTION
C
SUBROUTINE KTIME
USE STUFF
IF(IT-1)450,450,451
451 CONTINUE
JZ=0
GO TO 509
450 CONTINUE
DO 500 K=2,JN,2
DO 499 J=2,JM,2
IF(ICS)497,498,497
497 CONTINUE
EU=EXPF(-U(J))/RW
GO TO 496
498 CONTINUE
IF(CMBA)501,502,501
501 CONTINUE
DELTA=OMBA/U(J)
502 CONTINUE
EU=1.0/RW
496 CONTINUE
BUT=-BO(J,K)*BG(J,K)**2/DBG
TK1(J,K)=ABSF(POR(J,K)*VISO(J,K)*CU**2/(EU**2*BUT*PKO(J,K)*2.0))
BIT=VOU(J,K)
BAT=BUT*PKO(J,K)*EU/VISO(J,K)
BOT=BUT*(PKO(J+1,K)-PKO(J-1,K))*EU/(VISO(J,K)*DU*2.0)
TK2(J,K)=ABSF(DU*POR(J,K)/(EU*1.414*(BIT+BAT+BOT)))
499 CONTINUE
500 CONTINUE
DT=TK1(2,2)
LJ=2
LK=2
DO 504 K=2,JN,2
DO 503 J=2,JM,2
IF(TK1(J,K)-DT)505,506,506
505 CONTINUE
DT=TK1(J,K)
LK=K
LJ=J
506 CONTINUE
IF(TK2(J,K)-DT)507,508,508
507 CONTINUE
DT=TK2(J,K)

```

```

      LK=K
      LJ=J
508 CONTINUE
503 CCNTINUE
504 CONTINUE
      DT=DT*SGM
      WRITE OUTPUT TAPE 3,22,LK,LJ,DT,SGM
      MX=MX+1
      IF (MX-50)509,509,510
510 CONTINUE
      MX=0
      WRITE OUTPUT TAPE 3,23,((TK1(J,K),J=2,JM,2),K=2,JN,2)
      WRITE OUTPUT TAPE 3,23,((TK2(J,K),J=2,JM,2),K=2,JN,2)
      SGM=SGM*1.1
509 CONTINUE
      22 FORMAT(1H0//(2I10,2E16.5))
      23 FORMAT(1H0//(6E16.5))
      RETURN
      END
*      LISTB
*      CARDS COLUMN
*      FORTRAN          CGBAL
C      CONSERVATION OF MASS CHECK ROUTINE
      SUBROUTINE CGBAL
      USE STUFF
      GMI=0.0
      GMI=0.0
      BP=GMBD
      BOI=GMCA
      RSI=DMCA

C
C      COMPUTE FORMATION FACTORS AND FLUID VISCOSITIES
C
      DO 108 K=1,JK,1
      DO 107 J=1,JJ,1
      BG(J,K)=DBG*PG(J,K)+1.0
      IF (PON(J,K)-BP)234,234,235
235 CONTINUE
      RS(J,K)=RS(J,K)
      IF (GAMG)254,253,254
253 CONTINUE
      BG(J,K)=DBG*PON(J,K)+1.0
      BO(J,K)=1.0/BG(J,K)
      GO TO 255
254 CONTINUE
      BO(J,K)= GA+OHCC*(PON(J,K)-BP)
255 CONTINUE
      GO TO 231
234 CONTINUE
      RS(J,K)=DRS*PON(J,K)
      BO(J,K)=DBO*PON(J,K)+1.04
231 CCNTINUE
107 CONTINUE
108 CONTINUE

C
C      COMPUTE MASSES IN INTERNAL CELLS
C
      IF (DELTA)306,307,306
306 CONTINUE
      GGPZ=GAMG*PI*DZ*2.0*G
      GOPZ=GAMO*PI*DZ*2.0*G
      GO TO 308

```

```

307 CONTINUE
GGPZ=GAMG*DZ*2.0*G
GOPZ=GAMG*DZ*2.0*G
RJ=RW*DU*2.0
308 CONTINUE
OMI=0.0
GMI=0.0
DO 302 K=2,JN,2
DO 301 J=3,J0,2
IF (DELTA) 309,310,309
309 CONTINUE
IF (ICS) 303,304,303
303 CONTINUE
RJ=RW**2*(EXP(2.0*U(J+1))-EXP(2.0*U(J-1)))
BOA=((BO(J-1,K+1)+BO(J-1,K-1))*(EXP(2.0*U(J)))-EXP(2.0*
1U(J-1)))+(BO(J+1,K+1)+BO(J+1,K-1))*(EXP(2.0*U(J+1))-
2EXP(2.0*U(J)))*RW**2/(RJ*2.0)
BGA=((BG(J-1,K+1)+BG(J-1,K-1))*(EXP(2.0*U(J)))-EXP(2.0*
1U(J-1)))+(BG(J+1,K+1)+BG(J+1,K-1))*(EXP(2.0*U(J+1))-
2EXP(2.0*U(J)))*RW**2/(RJ*2.0)
GO TO 305
304 CONTINUE
RJ=RW**2*(U(J+1)**2-U(J-1)**2)
310 CONTINUE
BCA=((BC(J-1,K+1)+BC(J-1,K-1))*(U(J)**2-U(J-1)**2)+
1(BO(J+1,K+1)+BO(J+1,K-1))*(U(J+1)**2-U(J)**2))/(RJ*2.0)
BGA=((BG(J-1,K+1)+BG(J-1,K-1))*(U(J)**2-U(J-1)**2)+
1(BG(J+1,K+1)+BG(J+1,K-1))*(U(J+1)**2-U(J)**2))/(RJ*2.0)
305 CONTINUE
OMI=GMI+GOPZ*POR(J,K)*SON(J,K)*RJ/BOA
GMI=GMI+GGPZ*RJ*POR(J,K)*((1.0-SON(J,K))*BGA +SON(J,K)*RS(J,K)/
1BOA )
301 CONTINUE
302 CONTINUE
C
C      SUM CIL AND GAS PASSES
C
OMT=OMI
GMT=GMI
OMR=OM-OMI
GMR=GM-GMI
C
C      COMPUTE PASS DIFFERENCES
C
DELMO=OM-(OMI+QOCUM)
DELMG=GM-(GMI+QGCUM)
C
C      COMPUTE PASS DEVIATIONS
C
IF (KL) 312,312,701
701 CONTINUE
IF (POI-PH) 600,312,600
600 CONTINUE
DEVO=1.0-OMR/QOCUM
IF (GAMG) 311,312,311
311 CONTINUE
DEVG=1.0-GMR/QGCUM
312 CONTINUE
WRITE OUTPUT TAPE 3,1216,OMI,GMI,OMR,GMR,DELMO,DELMG,DEVO,DEVG,
*TIME
RETURN
1216 FORMAT(1H0/4E20.9/4E20.9/E20.9)
END

```


* LISTA
 * CARDS COLUMN
 * FORTRAN COUTPUT
 * SUBROUTINE COUTPUT

C
 C
 C

MAIN OUTPUT ROUTINE UGRÉ CCCC

USE STUFF

WRITE OUTPUT TAPE 3,1203,KL
 WRITE OUTPUT TAPE 3,1210,PM,L,CT,COSC,OF,CCR,TIME
 WRITE OUTPUT TAPE 3,1217,UM,GM,OGCUM,OGCUM,CELMO,DELMG,DEVO,DEVG
 WRITE OUTPUT TAPE 3,1211,(Z(K),QO(K),QOJ(K),OG(K),K=MU+1,PM-1,2)
 WRITE OUTPUT TAPE 3,1218
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1212,(U(J),(PCN(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1222
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(VCU(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1224
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(VCZ(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1226
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(PKO(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1228
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1216,(U(J),(BC(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1229
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1216,(U(J),(BG(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1231
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(LVOU(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1233
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(LVCZ(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1235
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1216,(U(J),(VISN(J,K),K=1,JK,2),J=1,JJ,1)
 IF (GANG)501,502,501

501 CONTINUE

WRITE OUTPUT TAPE 3,1219
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(SCN(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1220
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1212,(U(J),(PG(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1221
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1212,(U(J),(PCI(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1223
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(VGU(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1225
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(VGZ(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1227
 WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
 WRITE OUTPUT TAPE 3,1214,(U(J),(PKG(J,K),K=2,JN,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1230
 WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
 WRITE OUTPUT TAPE 3,1216,(U(J),(RS(J,K),K=1,JK,2),J=1,JJ,1)
 WRITE OUTPUT TAPE 3,1232

```

WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
WRITE OUTPUT TAPE 3,1214,(U(J),(DVGV(J,K),K=2,JN,2),J=1,JJ,1)
WRITE OUTPUT TAPE 3,1234
WRITE OUTPUT TAPE 3,1215,(Z(K),K=2,JN,2)
WRITE OUTPUT TAPE 3,1214,(U(J),(DVGV(J,K),K=2,JN,2),J=1,JJ,1)
WRITE OUTPUT TAPE 3,1236
WRITE OUTPUT TAPE 3,1213,(Z(K),K=1,JK,2)
WRITE OUTPUT TAPE 3,1216,(U(J),(VISG(J,K),K=1,JK,2),J=1,JJ,1)
502 CONTINUE
RETURN
1203 FORMAT(1H1//35X,38HCIL AND GAS RESERVOIR ENGINEERING CODE//45X,
*4HKL =16//)
1210 FORMAT(1H0/40X,15HPW = CONSTANT =E10.3//6X,4H L =16,5X,4HDT =E10.2
*,5X,6HQDSC =E10.3,5X,4HQT =E10.3,5X,5HGQR =E10.3,5X,6HTIME =E10.3)
1211 FORMAT(1H0//35X,1HZ,18X,2HQO,17X,3HQCG,18X,2HQG//(20X,4E20.4))
1212 FORMAT(1H0/(F8.2,11F10.2))
1213 FORMAT(1H0/4X,11F10.2)
1214 FORMAT(1H0/(F8.2,10E11.4))
1215 FORMAT(1H0/4X,10F11.2)
1216 FORMAT(1H0/(F8.2,11E10.3))
1217 FORMAT(1H0/5X,13HINITIAL GAS =E12.3,
*,2X,14HOIL PRODUCED =E12.3,2X,14HGAS PRODUCED =E12.3//5X,
*,15HOIL MASS DIF. =E12.3,2X,15HGAS MASS DIF. =E12.3,2X,
*,15HOIL MASS DEV. =E12.3,2X,15HGAS MASS DEV. =E12.3)
1218 FORMAT(1H1/30X,18HCIL PRESSURE ARRAY/)
1219 FORMAT(1H1/30X,20HCIL SATURATION ARRAY/)
1220 FORMAT(1H1/30X,18HGAS PRESSURE ARRAY/)
1221 FORMAT(1H1/30X,24HCAPILLARY PRESSURE ARRAY/)
1222 FORMAT(1H1/30X,19HRADIAL OIL VELOCITY/)
1223 FORMAT(1H1/30X,19HRADIAL GAS VELOCITY/)
1224 FORMAT(1H1/30X,21HVERTICAL OIL VELOCITY/)
1225 FORMAT(1H1/30X,21HVERTICAL GAS VELOCITY/)
1226 FORMAT(1H1/30X,22HOIL PERMEABILITY ARRAY/)
1227 FORMAT(1H1/30X,22HGAS PERMEABILITY ARRAY/)
1228 FORMAT(1H1/30X,21HOIL FORMATION FACTORS/)
1229 FORMAT(1H1/30X,21HGAS FORMATION FACTORS/)
1230 FORMAT(1H1/30X,20HGAS SOLUBILITY ARRAY/)
1231 FORMAT(1H1/30X,31HRADIAL OIL VELOCITY DERIVATIVES/)
1232 FORMAT(1H1/30X,31HRADIAL GAS VELOCITY DERIVATIVES/)
1233 FORMAT(1H1/30X,33HVERTICAL OIL VELOCITY DERIVATIVES/)
1234 FORMAT(1H1/30X,33HVERTICAL GAS VELOCITY DERIVATIVES/)
1235 FORMAT(1H1/30X,13HCIL VISCOSITY/)
1236 FORMAT(1H1/30X,13HGAS VISCOSITY/)
END
* LIST8
* CARDS COLUMN
* FCRTAN MCBAL
* SUBROUTINE MORAL
C
C MASS BALANCE PLOTTER
C
USE STUFF
DIMENSION YDA(4)
MF=0
MG=0
P=0.41*42.0
Q=0.94*21.0
CALL SETCH(P,Q,1,0,3,0)
CALL CRTCD(4HQGRE)
P=0.32*64.0
Q=0.84*32.0
CALL SETCH(P,Q,1,0,2,0)

```

CALL CRTBCD(18HMASS BALANCE CHECK)

P=0.01*64.0

Q=0.50*32.0

CALL SETCH(P,Q,1,0,2,0)

CALL CRTBCD(2HMB)

P=0.95*64.0

Q=0.08*32.0

CALL SETCH(P,Q,1,0,2,0)

CALL CRTBCD(1HT)

XTO=TS(1)

XTM=TS(JS)

YDA(1)=AMINAF(DMOS,1,JS,MF)

YDA(2)=AMAXAF(DMOS,1,JS,MG)

YDA(3)=AMINAF(DMGS,1,JS,MF)

YDA(4)=AMAXAF(DMGS,1,JS,MG)

YDMN=AMINAF(YDA,1,4,MF)

YDMX=AMAXAF(YDA,1,4,MG)

STEST=LOGF(XTM/XTO)

IF(STEST-2.30258)12,13,13

12 CONTINUE

CALL MAPG(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.80)

GO TO 14

13 CONTINUE

CALL MAPGLS(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.80)

14 CONTINUE

JPP=JS-1

DO 10 J=1,JPP,1

AXS=TS(J)

AXF=TS(J+1)

AYS=DMOS(J)

AYF=DMOS(J+1)

CALL LINE(AXS,AYS,AXF,AYF)

10 CONTINUE

DO 11 J=1,JPP,1

AXS=TS(J)

AXF=TS(J+1)

AYS=DMGS(J)

AYF=DMGS(J+1)

CALL LINE(AXS,AYS,AXF,AYF,3)

11 CONTINUE

CALL FRAME(1)

RETURN

END

*

LISTB

*

CARDS COLUMN

*

FORTRAN

CGXT

SUBROUTINE CGXT

C

GAS RATES VERSUS TIME

C

C

USE STUFF

MF=0

MG=0

P=0.41*42.0

Q=0.94*21.0

CALL SETCH(P,Q,1,0,3,0)

CALL CRTBCD(4HOGRE)

P=0.32*64.0

Q=0.84*32.0

CALL SETCH(P,Q,1,0,2,0)

CALL CRTBCD(2CHGAS PRODUCTION RATES)

P=0.01*64.0

```

      Q=C.50*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(2HGP)
      P=C.95*64.0
      Q=C.08*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(1HT)
      XTC=TS(1)
      XTM=TS(JS)
      YDMN=AMINAF(QGS,1,JS,MF)
      YDMX=AMAXAF(QGS,1,JS,MG)
      STEST=LOGF(XTM/XTC)
      IF(STEST-2.30258)12,13,13
12  CONTINUE
      CALL MAPG(XTO,XTM,YDMN,YDMX,C.36,0.90,C.08,C.80)
      GO TO 14
13  CONTINUE
      CALL MAPGLS(XTO,XTM,YDMN,YDMX,C.08,C.50,C.08,C.80)
14  CONTINUE
      JPP=JS-1
      DO 10 J=1,JPP,1
      AXS=TS(J)
      AXF=TS(J+1)
      AYS=QGS(J)
      AYF=QGS(J+1)
      CALL LINE(AXS,AYS,AXF,AYF)
10  CONTINUE
      CALL FRAME(1)
      RETURN
      END
•      LISTB
•      CARDS COLUMN
•      FORTRAN          OOGXT
      SUBROUTINE OOGXT

```

```

C
C      OIL AND GAS RATES VERSUS TIME
C

```

```

      USE STUFF
      DIMENSION YDA(4)
      MF=C
      MG=C
      P=C.41*42.0
      Q=C.94*21.0
      CALL SETCH(P,C,1,0,3,0)
      CALL CRTBCD(4HOGRE)
      P=C.32*64.0
      Q=C.84*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(16HPRODUCTION RATES)
      P=C.31*64.0
      Q=C.50*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(2HOG)
      P=C.95*64.0
      Q=C.08*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(1HT)
      XTC=TS(1)
      XTM=TS(JS)
      YDA(1)=AMINAF(QGS,1,JS,MF)
      YDA(2)=AMAXAF(QGS,1,JS,MG)
      YDA(3)=AMINAF(QGS,1,JS,MF)
      YDA(4)=AMAXAF(QGS,1,JS,MG)

```

```

YDMN=AMINAF(YDA,1,4,MF)
YDMX=AMAXAF(YDA,1,4,MG)
STEST=LOGF(XTM/XTO)
IF (STEST-2.30258)12,13,13
12 CONTINUE
CALL MAPG(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.60)
GO TO 14
13 CONTINUE
CALL MAPGLS(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.80)
14 CONTINUE
JPP=JS-1
DO 10 J=1,JPP,1
AXS=TS(J)
AXF=TS(J+1)
AYS=QOS(J)
AYF=QOS(J+1)
CALL LINE(AXS,AYS,AXF,AYF)
10 CONTINUE
DO 11 J=1,JPP,1
AXS=TS(J)
AXF=TS(J+1)
AYS=QOS(J)
AYF=QOS(J+1)
CALL LINE(AXS,AYS,AXF,AYF,3)
11 CONTINUE
CALL FRAME(1)
RETURN
END
* LIST8
* CARDS COLUMN
* FORTRAN QDCUM
SUBROUTINE QDCUM
C
C CUMULATIVE OIL AND GAS PRODUCTION
C
USE STUFF
DIMENSION YDA(4)
MF=0
MG=0
P=0.41*42.0
Q=0.94*21.0
CALL SETCH(P,C,1,C,3,0)
CALL CRTBCD(4HOGRE)
P=0.32*64.0
Q=0.84*32.0
CALL SETCH(P,Q,1,Q,2,0)
CALL CRTBCD(21HCUMULATIVE PRODUCTION)
P=0.01*64.0
Q=0.50*32.0
CALL SETCH(P,Q,1,Q,2,0)
CALL CRTBCD(2HCU)
P=0.95*64.0
Q=0.08*32.0
CALL SETCH(P,C,1,Q,2,0)
CALL CRTBCD(1HT)
XTO=TS(1)
XTM=TS(JS)
YDA(1)=AMINAF(QTS,1,JS,MF)
YDA(2)=AMAXAF(QTS,1,JS,MG)
YDA(3)=AMINAF(GORS,1,JS,MF)
YDA(4)=AMAXAF(GORS,1,JS,MG)
YDMN=AMINAF(YDA,1,4,MF)
YDMX=AMAXAF(YDA,1,4,MG)

```

```

      STEST=LOGF(XTM/XTO)
      IF (STEEST-2.30258)12,13,13
12  CCNTINUE
      CALL MAPG(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.80)
      GO TO 14
13  CONTINUE
      CALL MAPGLS(XTO,XTM,YDMN,YDMX,0.08,0.90,0.08,0.80)
14  CONTINUE
      JPP=JS-1
      DO 10 J=1,JPP,1
      AXS=TS(J)
      AXF=TS(J+1)
      AYS=QTS(J)
      AYF=QTS(J+1)
      CALL LINE(AXS,AYS,AXF,AYF)
10  CONTINUE
      DO 11 J=1,JPP,1
      AXS=TS(J)
      AXF=TS(J+1)
      AYS=GORS(J)
      AYF=GORS(J+1)
      CALL LINEP(AXS,AYS,AXF,AYF,3)
11  CONTINUE
      CALL FRAME(1)
      RETURN
      END
*      LISTB
*      CARDS COLUMN
*      FORTRAN          POUZ
      SUBROUTINE POUZ
      USE STUFF
C
C      ESTABLISH MAPPING
C
      UM=U(JJ)
      ZM=Z(JK)
      UMA=U(1)
      ZMA=Z(1)
      CALL MAPS(UMA,UM,ZMA,ZM,C,1,0.9,0.2,0.8)
C
C      PLOT HEADINGS
C
      P=C,41*42,0
      Q=C,94*21,0
      CALL SETCH(P,C,1,0,3,0)
      CALL CRTBCD(14HOGRE)
      P=C,32*64,0
      Q=C,84*32,0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(25H011 PRESSURE DISTRIBUTION)
      P=C,1*64,0
      Q=C,84*32,0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(1H1)
      P=C,95*64,0
      Q=C,2*32,0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(1HU)
      P=C,1*64,0
      Q=C,09*21,0
      CALL SETCH(P,C,1,0,3,0)
      CALL CRTBCD(6HNTIME #)
      P=C,30*42,0

```

```

U=0.09*21.0
CALL SETCH(P,C,1,0,3,0)
WRITE OUTPUT TAPE 100,1000,TIME

```

```

C
C      PLOT OIL PRESSURE DISTRIBUTION
C

```

```

      K2=ICON+1
      JKK=JJ*JK
      POMIN=PON(1,1)
      PUMAX=PON(1,1)
      DO 305 K=1,JK,2
      DO 304 J=1,JJ,2
      IF (PON(J,K)-PUMAX) 304,304,306
306  CONTINUE
      PUMAX=PON(J,K)
304  CONTINUE
305  CONTINUE
      DO 308 K=1,JK,2
      DO 307 J=1,JJ,2
      IF (PON(J,K)-POMIN) 309,307,307
309  CONTINUE
      POMIN=PON(J,K)
307  CONTINUE
308  CONTINUE
      W(1)=POMIN
      W(2)=PUMAX
      P=0.75*64.0
      Q=0.12*32.0
      CALL SETCH(P,Q,1,0,1,0)
      CALL CRTBCD(9HPO(MIN)= )
      P=0.90*64.0
      Q=0.12*32.0
      CALL SETCH(P,C,1,0,1,0)
      WRITE OUTPUT TAPE 100,1001,W(1)
      P=0.80*64.0
      Q=0.04*32.0
      CALL SETCH(P,Q,1,0,1,0)
      CALL CRTBCD(9HPO(MAX)= )
      P=C.76*64.0
      Q=C.04*32.0
      CALL SETCH(P,Q,1,C,1,0)
      WRITE OUTPUT TAPE 100,1002,W(2)
      CALL ACCTR1(ICON,W,K2,PO,JJ,U,1,JJ,2,2,1,JK,2)
      CALL FRAME(1)
      RETURN
1000 FORMAT(11.4)
1001 FORMAT(11.4)
1002 FORMAT(11.4)
END

```

```

•   LISTB
•   CARDS COLUMN
•   FORTRAN          PGUZ
      SUMMOUTIME PGUZ
      USE STUFF

```

```

C
C      ESTABLISH MAPPING
C

```

```

      UP=U(JJ)
      ZM=Z(JK)
      UPA=U(1)
      ZMA=Z(1)
      CALL MAPSTUPA,UM,ZMA,ZM,0.1,0.9,0.2,0.8)

```

C
C
C

PLOT HEADINGS

```
P=0.41*42.0
Q=0.94*21.0
CALL SETCH(P,0,1,0,3,0)
CALL CRTBCD(4HOGRE)
P=0.32*64.0
Q=0.84*32.0
CALL SETCH(P,0,1,0,2,0)
CALL CRTBCD(25HGAS PRESSURE DISTRIBUTION)
P=0.1*64.0
Q=0.84*32.0
CALL SETCH(P,0,1,0,2,0)
CALL CRTBCD(1HZ)
P=0.95*64.0
Q=0.2*32.0
CALL SETCH(P,0,1,0,2,0)
CALL CRTBCD(1HU)
P=0.10*42.0
Q=0.09*21.0
CALL SETCH(P,0,1,0,3,0)
CALL CRTBCD(6HTIME =)
P=0.30*42.0
Q=0.09*21.0
CALL SETCH(P,0,1,0,3,0)
WRITE OUTPUT TAPE 100,1000,TIME
```

C
C
C

PLOT GAS PRESSURE DISTRIBUTION

```
KZ=ICON+1
JXK=JJ*JK
PGM(N)=PG(1,1)
PGMAX=PG(1,1)
DO 305 K=1,JK,2
DO 304 J=1,JJ,2
IF (PG(J,K)-PGMAX)304,304,306
306 CONTINUE
PGMAX=PG(J,K)
304 CONTINUE
305 CONTINUE
DO 308 K=1,JK,2
DO 307 J=1,JJ,2
IF (PG(J,K)-PGMIN)309,307,307
309 CONTINUE
PGMIN=PG(J,K)
307 CONTINUE
308 CONTINUE
W(1)=PGMIN
W(2)=PGMAX
P=0.75*64.0
Q=0.12*32.0
CALL SETCH(P,0,1,0,1,0)
CALL CRTBCD(9HPGMIN(N)=)
P=0.40*64.0
Q=0.12*32.0
CALL SETCH(P,0,1,0,1,0)
WRITE OUTPUT TAPE 100,1001,W(1)
P=0.6*64.0
Q=0.6*32.0
CALL SETCH(P,0,1,0,1,0)
CALL CRTBCD(9HPGMAX(N)=)
P=0.76*64.0
```



```

C=0.04*32.0
CALL SETCH(P,Q,1,0,1,0)
WRITE OUTPUT TAPE 100,1002,W(2)
CALL RCCNTR(ICON,W,K2,PG,JJ,U,1,JJ,2,Z,1,JK,2)
CALL FRAME(1)
RLTURN
1000 FORMAT(E11.4)
1001 FORMAT(E11.4)
1002 FORMAT(E11.4)
END

• LIST8
• CARDS COLUMN
• FORTRAN PCUZ
  SUBROUTINE PCUZ
  USE STUFF

C
C   ESTABLISH MAPPING
C
  UM=U(JJ)
  Z=Z(JK)
  UMA=U(1)
  ZMA=Z(1)
  CALL MAPS(UMA,UM,ZMA,ZM,0.1,0.9,0.2,0.8)

C
C   PLOT HEADINGS
C
  P=0.41*42.0
  Q=0.94*21.0
  CALL SETCH(P,Q,1,0,3,0)
  CALL CRTBCD(4HGGRE)
  P=0.32*64.0
  Q=0.84*32.0
  CALL SETCH(P,Q,1,0,2,0)
  CALL CRTBCD(28HCAPIL. PRESSURE DISTRIBUTION)
  P=0.1*64.0
  Q=0.84*32.0
  CALL SETCH(P,Q,1,0,2,0)
  CALL CRTBCD(1HZ)
  P=0.95*64.0
  Q=0.2*32.0
  CALL SETCH(P,Q,1,0,2,0)
  CALL CRTBCD(1HU)
  P=0.10*42.0
  Q=0.09*21.0
  CALL SETCH(P,Q,1,0,3,0)
  CALL CRTBCD(6HTIME =)
  P=0.30*42.0
  Q=0.09*21.0
  CALL SETCH(P,Q,1,0,3,0)
  WRITE OUTPUT TAPE 100,1000,TIME

C
C   PLOT CAPILLARY PRESSURE DISTRIBUTION
C
  K2=ICON+1
  JJK=JJ=JK
  PCMIN=PC(1,1)
  PCMAX=PC(1,1)
  DO 305 K=1,JK,2
  DO 304 J=1,JJ,2
    IF(PC(J,K)-PCMAX)304,304,306
306 CONTINUE

```

```

      PCMAX=PC(J,K)
304 CONTINUE
305 CONTINUE
      DO 308 K=1,JK,2
      DO 307 J=1,JJ,2
      IF(PC(J,K)-PCMIN)309,307,307
309 CONTINUE
      PCMIN=PC(J,K)
307 CONTINUE
308 CONTINUE
      W(1)=PCMIN-0.0001
      W(2)=PCMAX
      P=0.75*64.0
      Q=0.12*32.0
      CALL SETCH(P,C,1,0,1,0)
      CALL CRTBCD(9HPC(MIN)= )
      P=0.90*64.0
      Q=0.12*32.0
      CALL SETCH(P,Q,1,0,1,0)
      WRITE OUTPUT TAPE 100,1001,W(1)
      P=0.6*64.0
      Q=0.04*32.0
      CALL SETCH(P,Q,1,0,1,0)
      CALL CRTBCD(9HPC(MAX)= )
      P=0.76*64.0
      Q=0.04*32.0
      CALL SETCH(P,Q,1,0,1,0)
      WRITE OUTPUT TAPE 100,1002,W(2)
      CALL RCCNTR(ICON,W,K2,PC,JJ,U,1,JJ,2,Z,1,JK,2)
      CALL FRAME(1)
      RETURN
1000 FORMAT(E11.4)
1001 FORMAT(E11.4)
1002 FORMAT(E11.4)
      END
*      LISTR
*      CARDS COLUMN
*      FORTRAN          SOUZ
      SUBROUTINE SOUZ
      USE STUFF
C
C      ESTABLISH MAPPING
C
      UM=U(JJ)
      ZM=Z(JK)
      UMA=U(1)
      ZMA=Z(1)
      CALL MAPS(UMA,UM,ZMA,ZM,C-1,0.9,0.2,0.8)
C
C      PLOT HEADINGS
C
      P=0.41*42.0
      Q=0.94*21.0
      CALL SETCH(P,Q,1,0,3,0)
      CALL CRTBCD(4HOGRE)
      P=0.32*64.0
      Q=0.84*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(27HOIL SATURAT ION DISTRIBUT ION)
      P=0.1*64.0
      Q=0.84*32.0
      CALL SETCH(P,C,1,0,2,0)
      CALL CRTBCD(1HZ)

```

```

P=0.95*64.0
Q=0.2*32.0
CALL SETCH(P,Q,1,0,2,0)
CALL CRTBCD(1HU)
P=0.10*42.0
Q=0.09*21.0
CALL SETCH(P,Q,1,0,3,0)
CALL CRTBCD(6HTIME =)
P=0.30*42.0
Q=0.09*21.0
CALL SETCH(P,Q,1,0,3,0)
WRITE OUTPUT TAPE 100,1000,TIME

```

```

C
C      PLOT OIL SATURATION DISTRIBUTION
C

```

```

      K2=ICON+1
      JXK=JJ*JK
      SOMIN=SON(1,1)
      SOMAX=SON(1,1)
      DO 305 K=1,JK,1
      DO 304 J=1,JJ,1
      IF(SO(J,K)-SOMAX)304,304,306
306 CONTINUE
      SOMAX=SO(J,K)
304 CONTINUE
305 CONTINUE
      DO 308 K=1,JK,1
      DO 307 J=1,JJ,1
      IF(SO(J,K)-SOMIN)309,307,307
309 CONTINUE
      SOMIN=SO(J,K)
307 CONTINUE
308 CONTINUE
      W(1)=SOMIN
      W(2)=SOMAX+0.0001
      P=0.75*64.0
      Q=0.12*32.0
      CALL SETCH(P,Q,1,0,1,0)
      CALL CRTBCD(9HSO(MIN)= )
      P=0.90*64.0
      Q=0.12*32.0
      CALL SETCH(P,Q,1,0,1,0)
      WRITE OUTPUT TAPE 100,1001,W(1)
      P=0.6*64.0
      Q=0.04*32.0
      CALL SETCH(P,Q,1,0,1,0)
      CALL CRTBCD(9HSO(MAX)= )
      P=0.76*64.0
      Q=0.04*32.0
      CALL SETCH(P,Q,1,0,1,0)
      WRITE OUTPUT TAPE 100,1002,W(2)
      CALL RCONTR(ICON,W,K2,SO,JJ,U,1,JJ,2,2,1,JK,2)
      CALL FRAME(1)
      RETURN
1000 FORMAT(E11.4)
1001 FORMAT(E11.4)
1002 FORMAT(E11.4)
      END
*      LIST8
*      CARDS COLUMN
*      FORTRAN          VCUZ
*      SUBROUTINE VOUZ

```

```

C
C      ESTABLISH MAPPING
C
      USE STUFF
      UM=U(JJ)
      ZM=Z(JK)
      UMA=U(1)
      ZMA=Z(1)
      CALL MAPS(UMA,UM,ZMA,ZM,C.1,0,9,0.2,0.8)

```

```

C
C      PLOT HEADINGS
      P=0.41*42.0
      Q=0.94*21.0
      CALL SETCH(P,Q,1,0,3,0)
      CALL CRTBCD(4HOGRE)
      P=0.32*64.0
      Q=0.84*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(18HOIL VELOCITY FIELD)
      P=0.1*64.0
      Q=0.84*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(1HZ)
      P=0.95*64.0
      Q=0.2*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(1HU)
      P=0.10*42.0
      Q=0.09*21.0
      CALL SETCH(P,Q,1,0,3,0)
      CALL CRTBCD(6HTIME =)
      P=0.30*42.0
      Q=0.09*21.0
      CALL SETCH(P,Q,1,0,3,0)
      WRITE OUTPUT TAPE 100,1000,TIME

```

```

C
C      PLOT OIL VELOCITY FIELD
C
      DO 304 K=2,JN,2
      DO 305 J=2,JN,2
      AVOU(J,K)=ABSF(VOU(J,K))
      AVOZ(J,K)=ABSF(VOZ(J,K))
305 CONTINUE
304 CONTINUE
      VMAX=AVCU(2,2)
      DO 307 K=2,JN,2
      DO 306 J=2,JN,2
      IF(AVOU(J,K)-VMAX)308,309,309
309 CONTINUE
      VMAX=AVOU(J,K)
308 CONTINUE
      IF(AVOZ(J,K)-VMAX)306,306,310
310 CONTINUE
      VMAX=AVOZ(J,K)
306 CONTINUE
307 CONTINUE
      IF(VMAX)316,317,316
316 CONTINUE
      IF(DU-DZ)311,311,312
311 CONTINUE
      XI=DU
      GO TO 313
312 CONTINUE

```

```

      XI=02
313 CONTINUE
      DO 315 K=2,JN,2
      DO 314 J=2,JM,2
      XB=U(J)
      YB=Z(K)
      XE=XB+XI*VOU(J,K)/VMAX
      YE=YB+XI*VOZ(J,K)/VMAX
      IF (XE-XB)320,321,320
321 CONTINUE
      IF (YE-YB)320,322,320
320 CONTINUE
      CALL PLOTV(XB,YB,XE,YE,U,G08)
322 CONTINUE
314 CONTINUE
315 CONTINUE
317 CONTINUE
      CALL FRAME(1)
      RETURN
1000 FORMAT(E11.4)
      END
•      LISTB
•      CARDS CCLUMX
•      FCRTAN          VGUZ
      SLBROUTINE VGUZ
      USE STUFF
C
C      ESTABLISH MAPPING
C
      UP=U(JJ)
      ZM=Z(JK)
      UPA=U(1)
      ZMA=Z(1)
      CALL MAPS(UPA,UM,ZMA,ZM,0.1,0.9,C.2,0.8)
C
C      PLOT HEADINGS
C
      P=C.41*42.0
      Q=C.94*21.0
      CALL SETCH(P,C,1,0,3,0)
      CALL CRTBCD(4HOGRE)
      P=C.32*64.0
      Q=C.84*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(18HGAS VELOCITY FIELD)
      P=C.01*64.0
      Q=C.84*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(1HZI)
      P=C.95*64.0
      Q=C.20*32.0
      CALL SETCH(P,Q,1,0,2,0)
      CALL CRTBCD(1HUI)
      P=C.10*42.0
      Q=C.09*21.0
      CALL SETCH(P,Q,1,0,3,0)
      CALL CRTBCD(6HTIME =)
      P=C.30*42.0
      Q=C.09*21.0
      CALL SETCH(P,Q,1,0,3,0)
      WRITE OUTPUT TAPE 100,1000,TIME
C
C      PLOT GAS VELOCITY FIELD

```

```

C
DO 304 K=2,JN,2
DC 305 J=2,JM,2
AVGU(J,K)=ABSF(VGU(J,K))
AVGZ(J,K)=ABSF(VGZ(J,K))
305 CCNTINUE
304 CONTINUE
VMAX=AVGU(2,2)
DO 307 K=2,JN,2
DO 306 J=2,JM,2
IF (AVGU(J,K)-VMAX) 308,309,309
309 CONTINUE
VMAX=AVGU(J,K)
308 CCNTINUE
IF (AVGZ(J,K)-VMAX) 306,306,310
310 CONTINUE
VMAX=AVGZ(J,K)
306 CCNTINUE
307 CONTINUE
IF (VMAX) 316,317,316
316 CONTINUE
IF (DU-DZ) 311,311,312
311 CONTINUE
XI=DU
GO TO 313
312 CONTINUE
XI=DZ
313 CONTINUE
DO 315 K=2,JN,2
DO 314 J=2,JM,2
XB=U(J)
YB=Z(K)
XE=XB+XI*VGU(J,K)/VMAX
YE=YB+XI*VGZ(J,K)/VMAX
IF (XE-XB) 320,321,320
321 CONTINUE
IF (YE-YB) 320,322,320
320 CONTINUE
CALL PLCTV(XB,YB,XE,YE,0.008)
322 CCNTINUE
314 CONTINUE
315 CCNTINUE
317 CCNTINUE
CALL FRAME(1)
RETURN
1000 FORMAT(E11.4)
END
*
* L1578
* CARDS COLUMN
* FORTRAN OILSAT
SUBROUTINE OILSAT
USE STUFF
C
C ESTABLISH MAPPING
C
UP=L(JJ)
ZM=Z(JK)
UMA=U(1)
ZMA=Z(1)
CALL MAPS(UPA,UM,ZMA,ZM,G.1,C.9,C.2,D.8)
C
C PLOT HEADINGS

```

C

```

P=0.41*32.0
Q=0.94*16.0
CALL SETCH(P,C,1,2,0)
CALL CRTBCD(14HORE)
P=0.32*64.0
Q=0.84*32.0
CALL SETCH(P,C,1,0,1,0)
CALL CRTBCD(12HORE SATURATION DENSITY)
P=0.1*64.0
Q=0.84*32.0
CALL SETCH(P,C,1,0,1,0)
CALL CRTBCD(1H2)
P=0.95*64.0
Q=0.2*32.0
CALL SETCH(P,C,1,0,1,0)
CALL CRTBCD(1HU)
P=0.10*32.0
Q=0.09*16.0
CALL SETCH(P,C,1,0,2,0)
CALL CRTBCD(6HTIME =)
P=0.30*32.0
Q=0.09*16.0
CALL SETCH(P,C,1,0,2,0)
WRITE OUTPUT TAPE 100,1000,TIME

```

C

PLOT OIL SATURATION IN U-Z SPACE

C

```

DO 100 K=2,JN,2
DO 99 J=2,JN,2
IF (SO(J,K))80,94,80
80 CONTINUE
X1=U(J-1)
Y1=Z(K-1)
X2=U(J+1)
Y2=Z(K-1)
X3=U(J+1)
Y3=Z(K+1)
X4=U(J-1)
Y4=Z(K+1)
IF (SO(J,K)-0.85)87,87,89
87 CONTINUE
KR=6
GO TO 89
88 CONTINUE
IF (SO(J,K)-0.90)85,85,86
85 CONTINUE
KR=5
GO TO 89
86 CONTINUE
IF (SO(J,K)-0.95)83,83,84
83 CONTINUE
KR=4
GO TO 89
84 CONTINUE
KR=3
89 CONTINUE
CALL SHADE(X1,Y1,X4,Y4,X3,Y3,X2,Y2,KR)
99 CONTINUE
100 CONTINUE
CALL FRAME(1)
1000 FORMAT(11,4)
RETURN
END

```

Appendix F

Code Flow Diagrams

PAGE 1

OIL AND GAS RESERVOIR ENGINEERING			
SYMBOL	DEFINITION OF SYMBOLS		
NT ARRAY	2A	MATRIX COEFFICIENT	
PERMEABILITY	AK	SATURATED MEDIUM	
FROM THE HORIZONTAL	ALPHA	ANGLE IN RADIANS	
ASGN	AVGU	SATURATION - 1.0	
VGU	AVGZ	2A	ABSOLUTE VALUE 0
VGZ	AVOU	2A	ABSOLUTE VALUE 0
VOU	AVOZ	2A	ABSOLUTE VALUE 0
VOZ	B	2A	MATRIX VARIABLE
CTOR	BC	2A	GAS FORMATION FA
CTOR	BO	2A	OIL FORMATION FA
ATION FACTOR	BOI		INITIAL OIL FORM
BP			OIL BUBBLE POINT
C			COEFFICIENT OF 0
IL PRESSURE EQUATION	CK		KOZENY CONSTANT
FICIENT SUBROUTINE	COEFU		U-DIRECTION COEF
FICIENT ENTRY POINT	COEFZ		Z-DIRECTION COEF
CS			-ICS
D	1A		MATRIX VARIABLE
CONSERVATION	DELNG		DEVIATION IN GAS
CONSERVATION	DELMO		DEVIATION IN OIL
CONSERVATION	DELTA		CONTROL NUMBER F
OR GEOMETRY	DEVG		FRACTIONAL ERROR
IN GAS CONSERVATION	DEVO		FRACTIONAL ERROR
IN OIL CONSERVATION	DEG		GRADIENT OF GAS
FORMATION FACTOR	DPO		GRADIENT OF OIL
FORMATION FACTOR	DP		DELTA PD
C	DPRG		ANALYTICAL DPG/D
C	DPRO		ANALYTICAL DPO/D
C	DOT		DELTA 00
C	DRE		GRADIENT OF GAS
SOLUTION FACTOR	DT		TIME INCREMENT

DU		INCREMENT IN X OR
ZONTAL DIRECTION	2A	SPACE DERIVATIVE
OF RADIAL GAS VELOCITY	DVGU	2A
OF VERTICAL GAS VELOCITY	DVGZ	2A
OF RADIAL OIL VELOCITY	DVOU	2A
OF VERTICAL OIL VELOCITY	DVOZ	2A
ICAL DIRECTION	DZ	INCREMENT IN VER
IENT ARRAY	E	2A
N COEFFICIENT	ETA	PRESSURE EQUATIO
ING PARAMETER	EU	U-DIRECTION SCAL
OR	F	COEFFICIENT OF D
OR	FPERM	PERMEABILITY FAC
OR	FPOR	POROSITY FACTOR
OR	G	GRAVITATIONAL CO
INSTANT	GANG	GAS DENSITY AT S
TP	GAMO	OIL DENSITY AT S
TP	GH	INITIAL MASS OF
GAS	GMA	PARAMETER FOR CH
DOUSING DT	GMB	NUMBER OF STEPS
TO LOWER WELL PRESSURE	GMBB	
PSIL	GMBG	INPUT VALUE OF E
C	GMBD	INPUT VALUE OF B
C	GMC	INPUT VALUE OF B
C	GMI	-GMI
C	GNT	MASS OF GAS AT T
C	GOR	GAS-OIL RATIO
C	GORS	1A
GOR	GVCOEF	GAS VISCOSITY CO
EFFICIENT	GVCON	GAS VISCOSITY CO
INSTANT	H	2A
IENT ARRAY	ICON	NUMBER OF CONTJU
RS TO BE PLOTTED	ICOUNT	CURRENT SIZE OF
MATRIX ARRAY	ICS	TEST FOR U-DIREC
ION SCALING		

	KEY	CARD OR TAPE TAP
C	UT	TEST FOR ZONE GE
C	OMETRY	BOUNDARY CONDITI
C	ON OPTION	NUMBER OF CYCLES
C	IT	MATRIX ITERATION
C	BETWEEN NEW DT	TEST FOR ENTRY P
C	ITER	MATRIX INDEX U
C	COUNTER	BOUNDARY INDEX F
C	IUZ	JB+1
C	DINT	JK/2
C	J	TOTAL NO. OF GRI
C	DIRECTION	TOTAL NO. OF GRI
C	OR SPACE ZONING	JJ/2
C	JB	JJ-1
C	JK	JJ-1
C	JJ	JJ-2
C	POINTS ON U AXIS	JK-2
C	JK	TIME SUMMARY IND
C	POINTS ON Z AXIS	JT
C	JL	TEST FOR SPACE Z
C	JH	MATRIX INDEX Z
C	JN	CYCLE COUNTER
C	JO	PRINTOUT CYCLE C
C	JP	MO-2
C	JS	PM-2
C	EX	TEST ON CONSTANT
C	JT	ITERATION INDEX
C	PRIMARY	VON HOLT SUBROUT
C	JZ	UPPER WELL BOUND
C	ZONING	LOWER WELL BOUND
C	K	OUTPUT CYCLE IND
C	DIRECTION	TEST FOR PRINT C
C	KL	TEST FOR TIME CY
C	L	DT
C	COUNTER	MASS CHECK SUBRO
C	LA	
C	LB	
C	LSPIN	
C	GO ITERATION	
C	LT	
C	FOR DT INCREASE	
C	MATRIX	
C	TIME	
C	MM	
C	ARY	
C	MO	
C	ARY	
C	N	
C	EX	
C	NP	
C	CYCLE NUMBER	
C	NT	
C	CLE NUMBER	
C	ODT	
C	OGBAL	
C	OUTINE	

	KEY	INITIAL MASS OF
C	OIL	LOCATION OF INPU
C	DELTA	TEST FOR MATRIX
C	QHB	INPUT VALE OF DP
C	EFFICIENT OUTPUT	INPUT VALE OF SB
C	QHC	
C	QHC	
C	QNT	MASS OF OIL AT T
C	QNS	RS
C	OUTPUT	MAIN OUTPUT SUBR
C	OUTINE	OIL VISCOSITY CO
C	QVCOEF	OIL VISCOSITY CO
C	EFFICIENT	CAPILLARY PRESSU
C	QVCON	MAXIMUM VALUE OF
C	INSTANT	MINIMUM VALUE OF
C	PC	PC
C	PC	POI + OR - FACTO
C	PCMAX	SPACE DEPENDENT
C	PCMAX	GAS PRESSURE
C	PCMIN	INITIAL GAS PRES
C	PCMIN	MAXIMUM VALUE OF
C	PCO	MINIMUM VALUE OF
C	PDI	TIME INTERPOLATI
C	PDI	OIL PRESSURE
C	PERM	INITIAL OIL PRES
C	MEDIUM PERMEABILITY	SPACE DEPENDENT
C	PG	GAS PRESSURE
C	PG	INITIAL GAS PRES
C	PG	MAXIMUM VALUE OF
C	PGMAX	MINIMUM VALUE OF
C	PGMIN	MEDIUM POROSITY
C	PHI	3.1415926
C	PI	EFFECTIVE GAS PE
C	PKG	EFFECTIVE OIL PE
C	PERMEABILITY	TIME INTERPOLATI
C	PKO	OIL PRESSURE
C	PH	INITIAL OIL PRES
C	ON OF PO AND PON	SPACE DEPENDENT
C	PO	MAXIMUM VALUE OF
C	POI	MINIMUM VALUE OF
C	SURE	NEW OIL PRESSURE
C	POR	SUBROUTINE FOR
C	MEDIUM POROSITY	OIL SURFACE TENS
C	POMAX	
C	PO	
C	POMIN	
C	PO	
C	PON	
C	POCON	
C	LOCITIES, ETC.	
C	PST	
C	IGN	

PRESCRIBED WELL			
C	PA		
C	PRESSURE		
C	QD		*QT/2
C	QG	1A	GAS MASS FLOW RA
C	TE		
C	QGCUM		CUMULATIVE GAS P
C	RODUCTION		
C	QGS	1A	TIME SUMMARY OF
C	QG		
C	QGSC		GAS VOLUME FLOW
C	RATE		
C	QO	1A	OIL MASS FLOW RA
C	TE		
C	QOCUM		CUMULATIVE OIL P
C	RODUCTION		
C	QOG	1A	DISSOLVED GAS MA
C	SS FLOW RATE		
C	QOGS	1A	TIME SUMMARY OF
C	QOG		
C	QOS	1A	TIME SUMMARY OF
C	QO		
C	QOSC		OIL VOLUME FLOW
C	RATE		
C	QOSTB		OIL VOLUME FLOW
C	RATE IN STOCK TANK BBLs		
C	Q		*Q*TIME/GMBB
C	QO		TEST VALUE OF QO
C	QT		TOTAL MASS FLOW
C	RATE		
C	QTS	1A	TIME SUMMARY OF
C	QT		
C	QX		SUMMATION OF OIL
C	MASS FLOW RATE		
C	QY		SUMMATION OF DIS
C	SOLVED GAS FLOW RATE		
C	QZ		SUMMATION OF GAS
C	MASS FLOW RATE		
C	RES	1A	MATRIX RESIDUAL
C	RHO		U-DIRECTION PARA
C	METER		
C	RJ		DIFFERENCE OF TH
C	E SQUARES OF THE RADII		
C	RS	2A	GAS SOLUBILITY F
C	ACTOR		
C	RSI		INITIAL GAS SATU
C	RATION		
C	RW		WELL RADIUS
C	SG	2A	GAS SATURATION
C	SGH		INITIAL GAS SATU
C	RATION		
C	SO	2A	OIL SATURATION
C	SQI		IRREDUCIBLE OIL
C	SATURATION		
C	SOM		INITIAL OIL SATU
C	RATION		
C	SOMAX		MAXIMUM VALUE OF
C	SO ARRAY		
C	SOMIN		MINIMUM VALUE OF
C	SO ARRAY		

C	SON	2A	NEW OIL SATURATI
C	SON		
C	SRS		SINGLE PHASE GAS
C	SATURATION		
C	TAU		PRESSURE EQUATIO
C	N COEFFICIENT		
C	TERM		TERM FROM PKO AN
C	D PKG EQUATIONS		
C	THETA		PRESSURE EQUATIO
C	N COEFFICIENT		
C	TIME		ELAPSED REAL TIM
C	TIME		
C	TOLER		*EPSILON*TIME/GMBB
C	TS	1A	TIME SUMMARY OF
C	TIME		
C	U	1A	RADIAL OR HORIZO
C	NTAL COORDINATE		
C	VGU	2A	RADIAL GAS VELOC
C	ITY		
C	VGZ	2A	VERTICAL GAS VEL
C	OCITY		
C	VISG	2A	GAS VISCOSITY
C	VISO	2A	OIL VISCOSITY
C	VMAX		MAXIMUM VELOCITY
C	VGU	2A	RADIAL OIL VELOC
C	ITY		
C	VOZ	2A	VERTICAL OIL VEL
C	OCITY		
C	H	1A	ARRAY OF CONTOUR
C	LEVELS		
C	X	1A	MATRIX VALUE OF
C	OIL PRESSURE		
C	Y	1A	RIGHT-HAND SIDE
C	OF MATRIX EQUATION		
C	Z	1A	VERTICAL COORDIN
C	ATE		
C	ZD		*JK/2
C	ZETA		PRESSURE EQUATIO
C	N COEFFICIENT		
C	ZLAM		Z-DIRECTION PARA
C	METER		
C	OGRE		MAIN CODE

USE
STUFF

READ CONTROL PARAMETERS

PAGE 0

PAGE :0

SUBROUTINE
COEFVC MATRIX COEFFICIENT SUBROUTINE -
U-DIRECTIONUSE
STUFF

```

BP=CMBO
M2=0
SBO=PMCC
SR5=0.0
CS=1CS
N=J/2
J1=J/2
ICOUNT=N+1

```

C ZERO OUT COEFFICIENT ARRAYS

DO 596
K=1,JKDO 595
J=1,JJ

```

SG(J,K)=0.0
ZETA(J,K)=0.0
ETA(J,K)=0.0
THETA(J,K)=0.0
TAU(J,K)=0.0
CI(J,K)=0.0
EI(J,K)=0.0
E1(J,K)=0.0
E2(J,K)=0.0
E3(J,K)=0.0

```

```

E4(J,K)=0.0
E5(J,K)=0.0
E6(J,K)=0.0
E7(J,K)=0.0
E8(J,K)=0.0
E9(J,K)=0.0
E10(J,K)=0.0
E11(J,K)=0.0
E12(J,K)=0.0
E13(J,K)=0.0
E14(J,K)=0.0
E15(J,K)=0.0
E16(J,K)=0.0

```

595

596

DO 575
K=1,3DO 574
J=1,JJ

E(J,K)=0.0

574

575

DO 576
J=1,JJY(J)=0.0
W(J)=0.0

576

C COMPUTE E TERMS AND ETA, THETA,
ZETA, AND TAU

```

AA=0;B=1;ZETA(J,K)=1/14
    .00DU1=(DELTA*E16(J,K)+EU
    aa2/DU+E11(J,K)
    BB=C(J,K)+POR(J,K)/(
    DT*2.0)+AA*12ZETA(J,
    K)+E16(J,K)+DU*2/12
    .00DU*2=CY
    CC=0;D=1-ZETA(J,K)/(
    4.00DU1)=(DELTA*E16(J,
    K)+EU*aa2+E16(J,K)+E
    U*2/DU+E11(J,K)
    DD=ZETA(J,K)+DU*0.1D
    ELTA=E11(J,K)+E16(J,K)
    J*EU+DELTA*E16(J,K)+
    DU*20/PC(J,K)
    EE=-E11(J,K)+DU*0.1D
    ELTA+E11(J,K)+E16(J,K)
    I*EU+E21(J,K)+E71(J,K)
    I=ZLAP*E151
    FF=THETA(J,K)+AA*0.1E
    16(J,K)+EU*20/PC(J,K)
    2,K)-PC(J,K-2,K)+0.2/(
    4.00DU1)*2*E
    E1(J,1)=AA
    E1(J,2)=BB
    E1(J,3)=CC
    H1(J)=DD+EE+FF
    
```

508

FC COMPUTE OUTER EDGE U-DIRECTION

FC COMPUTE INNER EDGE U-DIRECTION

SUBROUTINE
COEF2

HAIRIX COEFFICIENT SUBROUTINE -
2-DIRECTION

USE
STUFF

SBO=CMCC
SRS=0.0
BP=CMBD
CS=ICS
JL=JJ/2
JT=JK/2
ICOUNT=JL+1

ZERO OUT COEFFICIENT ARRAYS

DO 596
K=1,JK

DO 595
J=1,JJ

ZETA(J,K)=0.0
ETA(J,K)=0.0
THETA(J,K)=0.0
TAU(J,K)=0.0
C(J,K)=0.0
E1(J,K)=0.0
E2(J,K)=0.0
E3(J,K)=0.0
E4(J,K)=0.0
E5(J,K)=0.0
E6(J,K)=0.0
E7(J,K)=0.0
E8(J,K)=0.0
E9(J,K)=0.0

E10(J,K)=0.0
E11(J,K)=0.0
E12(J,K)=0.0
E13(J,K)=0.0
E14(J,K)=0.0
E15(J,K)=0.0
E16(J,K)=0.0

595

596

DO 575
K=1,JK

DO 574
J=1,JJ

A(J,K)=0.0

574

575

DO 576
J=1,JJ

V(J)=0.0
H(J)=0.0

576

COMPUTE E TERMS AND ETA, THETA,
ZETA, AND TAU

DO 580
K=1,JK

EU=1,0/RW

594

$$E3(I,J,K) = ((PKG(I,J,K+1) - PKG(I,J,K-1)) / VISG(I,J,K) - PKG(I,J,K)) * (VISG(I,J,K+1) - VISG(I,J,K-1)) / VISG(I,J,K)$$

$$E4(I,J,K) = ((PKO(I,J,K+1) - PKO(I,J,K-1)) / VISO(I,J,K) - PKO(I,J,K)) * (VISO(I,J,K+1) - VISO(I,J,K-1)) / VISO(I,J,K)$$

$$E10(I,J,K) = GAMMA * DBGA * COSF(ALPHA) * (PG(I,J,K+1) - PG(I,J,K-1)) / (2 * DZ)$$

504

505

DO 727
J=1,JJ
$$E3(I,J,1) = ((PKG(I,J,3) - PKG(I,J,1)) / VISG(I,J,1) - PKG(I,J,1)) * (VISG(I,J,3) - VISG(I,J,1)) / VISG(I,J,1)$$

$$E3(I,J,K1) = ((PKG(I,J,K1) - PKG(I,J,JP1)) / VISG(I,J,K1) - PKG(I,J,K1)) * (VISG(I,J,K1) - VISG(I,J,JP1)) / VISG(I,J,K1)$$

$$E4(I,J,1) = ((PKO(I,J,3) - PKO(I,J,1)) / VISO(I,J,1) - PKO(I,J,1)) * (VISO(I,J,3) - VISO(I,J,1)) / VISO(I,J,1)$$

$$E4(I,J,K1) = ((PKO(I,J,K1) - PKO(I,J,JP1)) / VISO(I,J,K1) - PKO(I,J,K1)) * (VISO(I,J,K1) - VISO(I,J,JP1)) / VISO(I,J,K1)$$

$$E10(I,J,1) = GAMMA * DBGA * COSF(ALPHA) * (PG(I,J,3) - PG(I,J,1)) / (DZ * 2.0)$$

$$E10(I,J,K1) = GAMMA * DBGA * COSF(ALPHA) * (PG(I,J,K1) - PG(I,J,JP1)) / (DZ * 2.0)$$

727

COMPUTE Z-DIRECTION DIL PRESSURE
COEFFICIENTS - INTERNAL

DO 550
J=3,J0

552

IF(ICS)

551

EU=EXP(-U(J)/RW)

553

552

IF(CMBA)

620

619

DELTA=CMBA/U(J)

620

EU=1,0/RW

553

DO 549

K=3,JP

```

AA=ZLAP*(ZETA(J,K)+1
E5(J,K)/(4.0*DDZ)-E16
(J,K)/(4.0*DDZ**2)+E
TA(J,K)*E15
BB=C(J,K)*POR(J,K)/(
DT*2.0)+ZLAP/(2.0*DDZ
**2)*(ZETA(J,K)+E*6(
J,K)-ETA(J,K
CC=ZLAP*(ZETA(J,K)+
(E16(J,K)/(4.0*DDZ**2
)+E5(J,K)/(4.0*DDZ))+
ETA(J,K)*E15
DD=C(J,K)*POR(J,K)*P
0(J,K)/(DT*2.0)+ZETA
(J,K)*(RHO*(DELTA*E1
6(J,K)+EU**2
EE=-ETA(J,K)*(RHO*(D
ELTA*E15(J,K)+EU**2
-E11(J,K)+E15(J,K)+E
12(J,K)+E15(
FF=RHO*(THETA(J,K)+1
E16(J,K)+EU**2*(PO(
J+2,K)-PO(J-2,K))**2
+2.0*(PO(J+2
E(K,1)=AA
E(K,2)=BB
E(K,3)=CC
H(K)=DD+EE+FF

```

549

C COMPUTE LOWER EDGE Z-DIRECTION

```

AA=C(J,1)*POR(J,1)/(
DT*2.0)
BB=0.0
CC=C(J,1)*POR(J,1)*P
0(J,1)/(DT*2.0)+ZETA
(J,1)*(RHO*(DELTA*E1
6(J,1)+EU**2
DD=-ETA(J,1)*(RHO*(D
ELTA*E7(J,1)+E15(J,1
)+E*4*DELTA*E15(J,1)+
EU**2*(PO(J+
EE=THETA(J,1)*RHO*(E
16(J,1)+EU**2*(PO(J
+2,1)-PO(J-2,1))**2+
2.0*(PO(J+2,
FF=THETA(J,1)*RHO*(E
16(J,1)+EU**2*(PC(J+
2,1)-PC(J-2,1))**2/(
4.0*DU)**2+E
E(1,2)=AA
E(1,3)=BB
H(1,1)=CC+DD+EE+FF

```

C COMPUTE UPPER EDGE Z-DIRECTION

```

AA=0.0
BB=C(J,K)*POR(J,K)
/(DT*2.0)-AA
CC=C(J,K)*POR(J,K)
*PO(J,K)/(DT*2.0)+Z
ETA(J,K)*(RHO*(DELTA
*E16(J,K)+
DD=-ETA(J,K)*(RHO*(
DELTA*E7(J,K)+E15(J
,K)+EU*DELTA*E15(J,
K)+EU**2*(P
EE=THETA(J,K)*RHO*(
E16(J,K)+EU**2*(PO
(J+2,K)-PO(J-2,K))
**2+2.0*(PO(
FF=THETA(J,K)*RHO*(E
16(J,K)+EU**2*(PC(
J+2,K)-PC(J-2,K))
**2/(4.0*DU)**2
E(K,1)=AA
E(K,2)=BB
H(K)=CC+DD+EE+FF

```

C COMPRESS MATRIX COEFFICIENTS

DO 571
I=2,JI

```

A(I,1)=E(2*I-1,1)
A(I,2)=E(2*I-1,2)
A(I,3)=E(2*I-1,3)
Y(I)=H(2*I-1)

```

571

```

A(I,2)=E(I,2)
A(I,3)=E(I,3)
Y(I)=H(I)
A(J+1,1)=E(K,1)
A(J+1,2)=E(K,2)
Y(J+1)=H(K)

```

PAGE 3

[illegible]

SUBROUTINE
SQUAD

$EFG = EFA * EXP(U(4)) * EXP(U(6))$
 $EFH = EFB * EXP(U(6)) * EXP(U(2))$
 $EFI = EFC * EXP(U(2)) * EXP(U(4))$
 $COAS = EFA + EFB + EFC$
 $COBS = EFD + EFE + EFF$
 $COCS = EFG + EFH + EFI$
 $SON(1, K) = COAS * EXP(U(1)) * 2.0 - COBS * EXP(U(1)) * COCS$

23

22

$EFA = SON(2, K) / ((U(2) - U(4)) * (U(2) - U(6)))$
 $EFB = SON(4, K) / ((U(4) - U(6)) * (U(4) - U(2)))$
 $EFC = SON(6, K) / ((U(6) - U(2)) * (U(6) - U(4)))$
 $EFD = EFA * (U(4) + U(6))$
 $EFE = EFB * (U(6) + U(2))$
 $EFI = EFC * (U(2) + U(4))$
 $EFG = EFA * U(4) * U(6)$
 $EFH = EFB * U(6) * U(2)$
 $EFI = EFC * U(2) * U(4)$
 $COAS = EFA + EFB + EFC$
 $COBS = EFD + EFE + EFF$
 $COCS = EFG + EFH + EFI$
 $SON(1, K) = COAS * U(1) * 2 - COBS * U(1) * COCS$

23

17

RETURN

**SUBROUTINE
GCRAL**

CH1=0.0
CH2=0.0
WP=CH20
DO1=CHCA
RE1=CHCA

COMPUTE FORMATION FACTORS AND FLUID VISCOSITIES

Diagram of a hexagonal cross-section of a bolt. The hexagon has a width of 20 mm and a height of 16 mm. The central hole has a diameter of 8 mm. The bolt is labeled with $K=1, K_1$.

DO 107
J-1, J-2

$$B(J,K) = B(J,K) + 1$$

PAGE 3

SUBROUTINE
MODAL

IF MASS BALANCE PLOTTER

USE
STUFF

P=0
M=0
P=0.41*42.0
Q=0.94*21.0

CALL
SETCH

CALL
CRTBCD

P=0.32*64.0
Q=0.84*32.0

CALL
SETCH

CALL
CRTBCD

P=0.01*64.0
Q=0.50*32.0

CALL
SETCH

CALL
CRTBCD

P=0.95*64.0
Q=0.08*32.0

CALL
SETCH

CALL
CRTBCD

Y0=YST11
XTM=TS*(JS
YDA(1)=AMINAF(DMOS,1
,JS,MF)
YDA(2)=AMAXAF(DMOS,1
,JS,MG)
YDA(3)=AMINAF(DMOS,1
,JS,MF)
YDA(4)=AMAXAF(DMOS,1
,JS,MG)
YDMN=AMINAF(YDA,1,4,
MF)
YDMX=AMAXAF(YDA,1,4,
MG)
STEST=LOGF(XTM/XT0)

SUBROUTINE
QGXT

GAS RATES VERSUS TIME

PAGE 2

ROUTINE
COGIT

Oil and Gas Rates Versus Time

USE
STUFF

MP=0
MG=0
P=0.41*42.0
Q=0.94*21.0

CALL
SETEH

CALL
CRTRCD

P=0.32*64.0
Q=0.84*32.0

CALL
SETEH

CALL
CRTRCD

P=0.01*64.0
Q=0.50*32.0

CALL
SETEH

CALL
CRTRCD

P=0.95*64.0
Q=0.08*32.0

CALL
SETEH

CALL
CRTRCD

XYO=TS(1)
XTH=TS(1,JS)
YDA(1)=AMINAF(QOS,1,
JS,MF)
YDA(2)=AMAXAF(QOS,1,
JS,MG)
YDA(3)=AMINAF(QOS,1,
JS,PF)
YDA(4)=AMAXAF(QOS,1,
JS,MG)
YDM=AMINAF(YDA,1,4,
MF)
YDM=AMAXAF(YDA,1,4,
MG)
STEST=LOGF(XTH/XTO)

PAGE 2

SUBROUTINE
OCCUM

CUMULATIVE OIL AND GAS PRODUCTION

USE
STUFF

```

PF=0
MC=0
P=0.41042.0
Q=0.94021.0

```

CALL
SETCHCALL
CRIBCD

```

P=0.32060.0
Q=0.84032.0

```

CALL
SETCHCALL
CRIBCD

```

P=0.01064.0
Q=0.50032.0

```

CALL
SETCHCALL
CRIBCD

```

P=0.95064.0
Q=0.08032.0

```

CALL
SETCHCALL
CRIBCD

```

X*O=TS(1)
XTH=TS(JS)
YDA(1)=AMINAF(QTS,1,
JS,PF)
YDA(2)=AMAXAF(QTS,1,
JS,MC)
YDA(3)=AMINAF(QORS,1
JS,PF)
YDA(4)=AMAXAF(QORS,1
JS,MC)
YDMX=AMINAF(YDA,1,4,
PF)
YDMX=AMAXAF(YDA,1,4,
MC)
STEST=LOGF(XTH/XTD)

```


PAGE 2

SUBROUTINE
POUZ

PAGE 3

PAGE 3

SUBROUTINE
PCUZ

PAGE 3

SUBROUTINE
S002

PAGE 3

SUBROUTINE
VOUZ

FC ESTABLISH MAPPING

USE
STUFF

UH=U(JJ)
ZH=Z(JK)
UHA=U(1)
ZHA=Z(1)

CALL
MAPS

FC PLOT HEADINGS

P=0.41842.0
Q=0.9421.0

CALL
SETCH

CALL
CRTBCD

P=0.3264.0
Q=0.8432.0

CALL
SETCH

CALL
CRTBCD

P=0.1864.0
Q=0.8432.0

CALL
SETCH

CALL
CRTBCD

P=0.9564.0
Q=0.2032.0

CALL
SETCH

CALL
CRTBCD

P=0.1842.0
Q=0.0921.0

CALL
SETCH

PAGE 3

SUBROUTINE
VBUZ

PAGE 3

PAGE 3

LEGAL NOTICE

This report was prepared as an instrument of Government sponsored work within the United States, and the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, in the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

Printed in USA. Available from:

Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards, U. S. Department of Commerce
Springfield, Virginia 22151

Price: Printed Copy \$3.00; Microfiche \$0.65.