RATE DECLINE ANALYSIS FOR NATURALLY FRACTURED RESERVOIRS

A REPORT
SUBMITTED TO THE DEPARTMENT OF PETROLEUM ENGINEERING OF STANFORD UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

By
Katsunori Fujiwara
June 1989
I certify that I have read this report and that in my opinion it is fully adequate, in scope and in quality, as partial fulfillment of the degree of Master of Science in Petroleum Engineering.

Younes Jalali-Yazdi
(Principal advisor)
Acknowledgement

I would like to thank Dr. Younes Jalali-Yazdi for his essential help and his able guidance as my research advisor during the course of this research work and the preparation of this report.

I also would like to thank the faculty, students, and staff of the Department of Petroleum Engineering for providing me with encouragement and good basis for research work.

I would like to dedicate the present study to my wife, Rinko, for providing me patience and happiness.

Finally, I would like to thank Nippon Mining Co., Ltd. and Japan National Oil Corporation for having provided the financial support for my Master’s program at Stanford University.
Abstract

In this work, transient rate analysis for constant pressure production in a naturally fractured reservoir is presented. The solution for the dimensionless flowrate is based on a model which treats interporosity flow as a function of a continuous matrix block size distribution. Several distributions of matrix block size are considered. This approach is similar to that of Ref. 1, which examined the pressure response.

The flowrate response is investigated for both pseudo-steady state (PSS) and unsteady state (USS) interporosity models, which include slab, cylindrical, and spherical matrix block geometries. It was found that the flowrate decline becomes smooth, specially for the unsteady state model, and approaches the decline behavior of a nonfractured reservoir when matrix block size variability is large, i.e., when fracturing is extremely nonuniform. The difference in flowrate for various geometric models of blocks is not significant, with the spherical geometry yielding the highest and the slab yielding the lowest flowrate.

This work suggests why certain naturally fractured reservoirs do not exhibit a sudden rate decline followed by a period of constant flowrate as predicted by classical double porosity models. Also, the results indicate that reservoir producibility is directly proportional to fracture intensity and inversely proportional to the degree of fracture nonuniformity. Hence, the Warren and Root model which assumes fracturing is perfectly uniform, provides an upper bound of reservoir producibility and cumulative production.
Contents

Acknowledgement iii
Abstract iv
Table of Contents vi
List of Tables vii
List of Figures ix

1 Introduction 1

2 Mathematical Model 4
   2.1 Initial Boundary Value Problem ......................... 4
   2.2 Probability Density Functions (PDF) ..................... 6
   2.3 Solutions ........................................... 8
      2.3.1 General Solution .............................. 8
      2.3.2 Solutions for Slab Geometry ................. 8
      2.3.3 Solutions for Cylindrical Geometry .......... 11
      2.3.4 Solutions for Spherical Geometry ............. 11
      2.3.5 Dimensionless Parameters .................. 12

3 Discussion 15
   3.1 Effect of Matrix Block Geometry ..................... 15
   3.2 Effect of the Mode of Interporosity Flow ........... 26
List of Tables

2.1 Constant $\alpha$ for various matrix geometries .......................... 13
# List of Figures

1.1 Naturally Fractured Reservoir ................................................. 3  
2.1 Uniform Distribution ......................................................... 6  
2.2 Positively Skewed Linear Distribution ................................. 7  
2.3 Negatively Skewed Linear Distribution ................................. 7  
2.4 Matrix Slab Geometry ..................................................... 13  
2.5 Matrix Cylindrical Geometry ............................................. 14  
2.6 Matrix Spherical Geometry .............................................. 14  
3.1 Flowrate profile for slab geometry in **USS** model ............... 17  
3.2 Flowrate profile for cylindrical geometry in USS model ........ 18  
3.3 Flowrate profile for spherical geometry in **USS** model ........ 19  
3.4 Comparison of flowrate response of various matrix block geometries . 20  
3.5 Difference in flowrate of various matrix block geometries ....... 21  
3.6 Pressure and pressure derivative profile for slab geometry in **USS** model ......................................................... 22  
3.7 Pressure and pressure derivative profile for cylindrical geometry in USS model .......................................................... 23  
3.8 Pressure and pressure derivative profile for spherical geometry in **USS** model .......................................................... 24  
3.9 Comparison of pressure derivative profile of various matrix block geometries ................................................................. 25  
3.10 Comparison of **PSS** and USS flowrate response ................. 27  
3.11 Comparison of **PSS** and USS pressure derivative profiles ....... 28
3.12 PSS flowrate profile for various matrix block size distributions .... 30
3.13 PSS pressure and pressure derivative profile for various matrix block size distributions ........................................ 31
3.14 USS flowrate profile for various matrix block size distributions .... 32
3.15 USS pressure and pressure derivative profile for various matrix block size distributions ........................................ 33
3.16 Cumulative production at \( t_D = 10^5 \) for PSS model ............... 35
3.17 Cumulative production at \( t_D = 10^5 \) for USS model ............... 36
3.18 Flowrate profile for various \( \lambda_{g\text{mean}} \) .............................. 37
3.19 Pressure derivative profile for various \( \lambda_{g\text{mean}} \) .................. 38
Section 1

Introduction

Much work has been done on the pressure transient modeling of naturally fractured reservoirs. However, the rate response and producing capacity of these reservoirs have not received adequate attention. This work examines flowrate decline behavior of naturally fractured reservoirs.

Naturally fractured reservoirs are heterogeneous porous media which consist of fractures and matrix blocks. The matrix blocks store most of the fluid, but have low permeability. On the other hand, the fractures do not store much, but have extremely high permeability. Most of the reservoir fluid flows from the matrix blocks into the wellbore through the permeable fractures. Therefore, the producing capacity of a naturally fractured reservoir is governed by matrix-fracture fluid transport capacity, which is called interporosity flow. To describe flow in naturally fractured reservoirs, double porosity model has been widely used. Fig. 1.1 shows the schematic of a naturally fractured reservoir and its double porosity idealization. This concept was first proposed by Barenblatt et al [2,3]. Transient pressure behavior for this model has been studied by many researchers [2-16]. Mavor and Cinco-Ley [10] and Da Prat et al [17] examined the rate response of this model by applying the rate decline concept proposed by Fetkovich [18]. Raghavan and Ohaeri [19], and Sageev [20] also examined the rate decline behavior of naturally fractured reservoirs. These works indicate that the double porosity model predicts an initial high flowrate followed by a sudden rate decline and a period of constant
flowrate (the interporosity flow period). Although many naturally fractured reservoirs exhibit such behavior, many others do not. They exhibit a gradual rate decline throughout the life of the reservoir, similar to that of nonfractured reservoirs. Jalali-Yazdi et al [21] suggested the concept of a distributed interporosity flow strength to explain such behavior.

Here, a similar approach is adopted, and the effect of the variation of matrix block size on flowrate is investigated. This work demonstrates that the gradual rate decline occurs in nonuniformly fractured reservoirs where the variation in matrix block size is large. Also, this work shows that fracture nonuniformity has an adverse effect on the producing capacity of naturally fractured reservoirs.
SECTION 1. INTRODUCTION

Figure 1.1: Naturally Fractured Reservoir (after Ref. 4)
Section 2

Mathematical Model

The partial differential equations and their solutions for pseudo-steady state and three unsteady state cases, which include slab, cylindrical, and spherical matrix block geometries, are presented in this section. Also, the probability density functions (PDF) used in this study to represent various distributions of matrix block size, are presented.

2.1 Initial Boundary Value Problem

The fracture diffusivity equation for a double porosity reservoir with a continuous matrix block size distribution is given by:

\[ \nabla^2 P_f = \frac{\phi_f c_f \mu}{k_f} \frac{\partial P_f}{\partial t} + \frac{\mu}{k_f} \int_{h_{\text{min}}}^{h_{\text{max}}} U(h) P(h) dh, \quad (2.1) \]

and the matrix diffusivity equation is:

\[ \nabla^2 P_m = \frac{\phi_m c_m \mu}{k_m} \frac{\partial P_m}{\partial t}. \quad (2.2) \]

In Eqn. 2.1, \( P(h) \) is the probability density function of matrix block size and is discussed in the next section. \( U(h) \) in Eqn. 2.1 is the flow contribution from a matrix block of size \( h \), and is given by:

\[ U(h) = -\frac{k_m A}{\mu V} \nabla P_m \bigg|_{\text{interface}}, \quad (2.3) \]
SECTION 2. MATHEMATICAL MODEL

where $A/V$ is the specific surface area of the matrix block.

The main assumptions used to develop the equations and solutions are as follows:

- flow is single phase and obeys Darcy’s law,
- reservoir fluid is slightly compressible,
- reservoir is radial and infinite in extent,
- matrix and fracture properties are homogeneous and isotropic, and
- the well is either producing at constant pressure (rate decline) or constant rate (pressure decline).

For a reservoir at constant pressure, the initial condition is:

$$P_f = P_m = P_i \quad ; \quad at \quad t = 0. \quad (2.4)$$

The radial inner boundary condition for wellbore storage and skin are:

$$q = C \left( \frac{\partial P_w}{\partial t} \right) - \frac{2\pi k_i h}{\mu} \left( r \frac{\partial P_f}{\partial r} \right)_{r=r_w}, \quad (2.5)$$

$$P_w = [P_f - S_d \left( r \frac{\partial P_f}{\partial r} \right)_{r=r_w}] \quad . \quad (2.6)$$

The radial outer boundary condition for an infinite acting reservoir is:

$$P_f = P_i \quad ; \quad as \quad r \rightarrow \infty. \quad (2.7)$$

Two boundary conditions are needed for flow in matrix blocks. One specifies that $P_m = P_f$ at the matrix-fracture interface. The other, depending on the geometry, specifies either no-flow boundary or bounded pressure at the center of the matrix block (see Appendix A for details).
2.2 Probability Density Functions (PDF)

Three probability density functions for the matrix block size distribution are considered — uniform, positively skewed linear, and negatively skewed linear distributions. Fig. 2.1 shows the schematic of a uniform distribution. The normalized probability density function for uniform distribution is given by:

\[ P(h_D) = \frac{1}{1 - \frac{h_{\text{min}}}{h_{\text{max}}}}. \]  

Fig. 2.2 depicts a positively skewed linear distribution, where the normalized probability density function is given by:

\[ P(h_D) = \frac{2(h_D - h_{\text{min}})}{(1 - \frac{h_{\text{min}}}{h_{\text{max}}})^2}. \]  

The negatively skewed linear distribution is shown in Fig. 2.3, and the normalized probability density function is given by:

\[ P(h_D) = \frac{2(1 - h_D)}{(1 - \frac{h_{\text{min}}}{h_{\text{max}}})^2}. \]  

Figure 2.1: Uniform Distribution
SECTION 2. MATHEMATICAL MODEL

Figure 2.2: Positively Skewed Linear Distribution

Figure 2.3: Negatively Skewed Linear Distribution
2.3 Solutions

2.3.1 General Solution

The initial boundary value problem is rendered dimensionless and solved using Laplace transforms. The procedure is stated in Appendix A. The constant pressure solution is obtained from the constant rate solution (with \( C_D = 0 \)) using the result presented by van Everdingen and Hurst [22]:

\[ \bar{q}_D = \frac{1}{s^2 \bar{P}_{Dw}}, \quad (2.11) \]

where:

\[ \bar{P}_{Dw} = \frac{K_0(x) + S_D x K_1(x)}{s x K_1(x)}. \quad (2.12) \]

The cumulative production is obtained by integrating the flowrate over time:

\[ Q_D = \int_0^{t_{dp}} q_D dt_D. \quad (2.13) \]

Or, in Laplace space:

\[ \bar{Q}_D = \frac{q_D}{s}. \quad (2.14) \]

The argument \( x \) of modified Bessel functions is:

\[ x = \sqrt{s g(s)}. \quad (2.15) \]

The function \( g(s) \) can be obtained by specifying the mode of interporosity flow (PSS or USS), the geometry of matrix blocks (slab, cylinder, or sphere), and the distribution of matrix block size.

2.3.2 Solutions for Slab Geometry

**Fig.** 2.4 shows the schematic of the slab geometry. In this geometry, matrix blocks and fractures are assumed to be piled up on one another. No flow boundary exists at the center of the matrix slab due to flow symmetry. Pseudo-steady state and unsteady state cases are considered for the three probability density functions discussed in section 2.2.
SECTION 2. MATHEMATICAL MODEL

Pseudo-Steady State Solutions

o Arbitrary Distribution

Let $P(h_D)$ denote any probability density function of matrix block size. From Appendix A, we have:

$$g(s) = \omega_f + \omega_m \int_{\lambda_{\min}}^{\lambda_{\max}} \frac{\lambda}{\omega_s} P(h_D) dh_D.$$  \hspace{1cm} (2.16)

o Uniform Distribution

By combining Eqns. 2.16 and 2.8, one obtains:

$$g(s) = \omega_f + \omega_m \frac{\sqrt{\lambda_{\min}}}{1 - \sqrt{\lambda_{\max}}} \left[ \arctan \left( \frac{\lambda_{\max}}{\omega_s} \right) - \arctan \left( \frac{\lambda_{\min}}{\omega_s} \right) \right].$$  \hspace{1cm} (2.17)

o Positively Skewed Linear Distribution

By combining Eqns. 2.16 and 2.9, one obtains:

$$g(s) = \omega_f + \frac{\omega_m \lambda_{\min}}{(1 - \sqrt{\lambda_{\max}})^2} \left[ \ln \frac{\lambda_{\max} \omega_s + \lambda_{\min}}{\omega_s \lambda_{\min}} \right]$$

$$- \frac{2}{\sqrt{\omega_s} \lambda_{\max}} \left[ \arctan \left( \frac{\lambda_{\max}}{\omega_s} \right) - \arctan \left( \frac{\lambda_{\min}}{\omega_s} \right) \right].$$  \hspace{1cm} (2.18)

o Negatively Skewed Linear Distribution

By combining Eqns. 2.16 and 2.10, one obtains:

$$g(s) = \omega_f + \frac{\omega_m \sqrt{\lambda_{\min}}}{(1 - \sqrt{\lambda_{\max}})^2} \left[ \frac{2}{\sqrt{\omega_s}} \left\{ \arctan \left( \frac{\lambda_{\max}}{\omega_s} \right) - \arctan \left( \frac{\lambda_{\min}}{\omega_s} \right) \right\} \right]$$

$$- \frac{\sqrt{\lambda_{\min}}}{\omega_s} \ln \left( \frac{\lambda_{\max} \omega_s + \lambda_{\min}}{\omega_s \lambda_{\min}} \right).$$  \hspace{1cm} (2.19)
SECTION 2. MATHEMATICAL MODEL

Unsteady State Solutions

- **Arbitrary Distribution**
  Similarly, let $P(h_D)$ denote any probability density function of matrix block size. One obtains:

$$g(s) = \omega_f + \omega_m \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \sqrt{\frac{\lambda}{3\omega_m s}} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) P(h_D) dh_D. \quad (2.20)$$

- **Uniform Distribution**
  Eqns. 2.20 and 2.8 yield:

$$g(s) = \omega_f + \omega_m \frac{\sqrt{\frac{\lambda_{\text{min}}}{3\omega_m s}}}{2 \left( 1 - \sqrt{\frac{\lambda_{\text{min}}}{\lambda_{\text{max}}}} \right)} \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{1}{\lambda} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) d\lambda. \quad (2.21)$$

- **Positively Skewed Linear Distribution**
  Eqns. 2.20 and 2.9 yield:

$$g(s) = \omega_f + \frac{\omega_m}{\left( 1 - \sqrt{\frac{\lambda_{\text{min}}}{\lambda_{\text{max}}}} \right)^2}$$

$$\times \left[ \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{\lambda_{\text{min}}}{\sqrt{3\omega_m s \lambda^2}} \frac{1}{\lambda} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) d\lambda \right. \right.$$ 

$$- \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{\lambda_{\text{min}}}{\sqrt{3\omega_m s \lambda^2 \lambda_{\text{max}}}} \frac{1}{\lambda} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) d\lambda \right]. \quad (2.22)$$

- **Negatively Skewed Linear Distribution**
  Eqns. 2.20 and 2.10 yield:

$$g(s) = \omega_f + \frac{\omega_m}{\left( 1 - \sqrt{\frac{\lambda_{\text{min}}}{\lambda_{\text{max}}}} \right)^2}$$

$$\times \left[ \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{\sqrt{\frac{\lambda_{\text{min}}}{\lambda_{\text{max}}}}}{\sqrt{3\omega_m s \lambda^2}} \frac{1}{\lambda} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) d\lambda \right. \right.$$ 

$$- \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{\lambda_{\text{min}}}{\sqrt{3\omega_m s \lambda^2}} \frac{1}{\lambda_{\text{max}}} \tanh \left( \sqrt{\frac{3\omega_m s}{\lambda}} \right) d\lambda \right]. \quad (2.23)$$
2.3.3 Solutions for Cylindrical Geometry

In this model, the matrix blocks are assumed to be of cylindrical shape and are separated by vertical fractures. The schematic of this geometry is shown in Fig. 2.5.

The PSS solutions are the same as those given for the slab geometry, Eqns. 2.16 through 2.19 (but different definition of $X$ as discussed in section 2.3.5 is used). The USS solutions are given below.

- **Arbitrary Distribution**

  Let $P(r_{mD})$ denote any probability density function of matrix block size. One obtains:

  \[ g(s) = \omega_f + \omega_m \int_{r_{min}}^{r_{max}} \sqrt{\frac{2\lambda}{3\omega_m s}} \frac{I_1 \left( \frac{\sqrt{6\omega_m r}}{\lambda} \right)}{I_0 \left( \frac{\sqrt{6\omega_m r}}{\lambda} \right)} P(r_{mD}) dr_{mD}. \]  

  (2.24)

- **Uniform Distribution**

  Eqns. 2.24 and 2.8 yield:

  \[ g(s) = \omega_f + \omega_m \frac{\sqrt{\frac{r_{min}}{3\omega_m s}}}{2 \left( 1 - \sqrt{\frac{r_{min}}{r_{max}}} \right)} \int_{r_{min}}^{r_{max}} \frac{1}{\lambda} \frac{I_1 \left( \frac{\sqrt{6\omega_m r}}{\lambda} \right)}{I_0 \left( \frac{\sqrt{6\omega_m r}}{\lambda} \right)} d\lambda. \]

  (2.25)

2.3.4 Solutions for Spherical Geometry

This model is shown in Fig. 2.6. Matrix blocks have spherical shape and are surrounded by fractures. The PSS solutions are the same as those given for the slab or the cylindrical geometry, Eqns. 2.16 through 2.19 (but different definition of $X$ as discussed in section 2.3.5 is used). The USS solutions are given below.

- **Arbitrary Distribution**

  Let $P(r_{mD})$ denote any probability density function of matrix block size. One obtains:

  \[ g(s) = \omega_f + \omega_m \int_{r_{min}}^{r_{max}} \sqrt{\frac{\lambda}{\omega_m s}} \coth \left( \sqrt{\frac{9\omega_m s}{\lambda}} \right) P(r_{mD}) dr_{mD} \]

  - \[ -\frac{1}{3s} \int_{r_{min}}^{r_{max}} \lambda P(r_{mD}) dr_{mD}. \]

  (2.26)
SECTION 2. MATHEMATICAL MODEL

- Uniform Distribution

Eqns. 2.26 and 2.8 yield:

\[
g(s) = \omega_f + \omega_m \frac{\sqrt{\frac{\lambda_{\text{min}}}{\omega_m s}}}{2(1 - \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}})} \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{1}{\lambda} \coth \left( \sqrt{\frac{s \omega_m \lambda}{\lambda}} \right) d\lambda
\]

- \[ - \frac{\sqrt{\lambda_{\text{min}}}}{6s(1 - \frac{\lambda_{\text{min}}}{\lambda_{\text{max}}})} \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} \frac{d\lambda}{\sqrt{\lambda}} \]

(2.27)

2.3.5 Dimensionless Parameters

The following dimensionless parameters are used:

\[
P_{Df} = \frac{2\pi k_f h_f (P_i - P_f)}{q \mu B}, \quad (2.28)
\]

\[
P_{Dm} = \frac{2\pi k_f h_f (P_i - P_m)}{q \mu B}, \quad (2.29)
\]

\[
t_D = \frac{k_f t}{(\phi_f c_f + \phi_m c_m) \mu r_w^2}, \quad (2.30)
\]

\[
\tau_D = \frac{r}{r_w}, \quad (2.31)
\]

\[
\omega_m = \frac{\phi_m c_m}{\phi_f c_f + \phi_m c_m}, \quad (2.32)
\]

\[
\omega_f = 1 - \omega_m. \quad (2.33)
\]

For slab geometry:

\[
\lambda = \alpha \frac{k_m r_w^2}{k_f h^2}. \quad (2.34)
\]

\[
z_D = \frac{z}{h}. \quad (2.35)
\]

\[
h_D = \frac{h}{h_{\text{max}}}. \quad (2.36)
\]

For cylindrical and spherical geometry:

\[
\lambda = \alpha \frac{k_m r_w^2}{k_f r_m^2}. \quad (2.37)
\]

\[
\eta_D = \frac{\eta}{r_m}. \quad (2.38)
\]
SECTION 2. MATHEMATICAL MODEL

\[ T_{mD} = \frac{T_m}{T_{m,m}}. \]  \hspace{1cm} (2.39)

The constant \( a \) in the definition of interporosity flow coefficient \( X \) is given in Table 2-1.

Table 2.1: Constant \( a \) for various matrix geometries

<table>
<thead>
<tr>
<th>Geometry</th>
<th>( a )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slab</td>
<td>3</td>
</tr>
<tr>
<td>Cylinder</td>
<td>6</td>
</tr>
<tr>
<td>Sphere</td>
<td>9</td>
</tr>
</tbody>
</table>

Figure 2.4: Matrix Slab Geometry
Figure 2.5: Matrix Cylindrical Geometry

Figure 2.6: Matrix Spherical Geometry
Section 3

Discussion

In this work, flowrate and pressure response are considered and the factors which affect them are investigated. To convert the rate and pressure solutions from Laplace space into real space, Stehfest Algorithm [23] is used. The unsteady state solutions are evaluated using numerical integration.

3.1 Effect of Matrix Block Geometry

Figs. 3.1 through 3.3 show the rate response for the uniform distribution and USS interporosity flow for slab, cylindrical, and spherical matrix block geometries, respectively. Each figure is for one value of \( \omega_m \) and \( \lambda_{max} \), and several values of \( \lambda_{ratio} \). The parameter \( \lambda_{ratio} \) is defined as:

\[
\lambda_{ratio} = \frac{\lambda_{max}}{\lambda_{min}},
\]

which is a measure of the spread (variance) of the matrix block size distribution and determines the duration of the interporosity flow period. In each of the Figs. 3.1 through 3.3, the interporosity flow period for a given \( \lambda_{ratio} \) begins and ends at the same time for the three geometries. This occurs because the duration of the interporosity flow period depends only on \( \lambda_{ratio} \). The effect of block geometry appears only in the interporosity flow period. Fig. 3.4 shows that the spherical model yields a higher flowrate than the Cylindrical model, which in turn yields a
higher flowrate than the slab model. The differences are, however, not significant. Also, as Fig. 3.5 indicates, as \( \lambda_{\text{ratio}} \) increases, the difference in flowrate decreases.

Figs. 3.6 through 3.8 show the pressure and the pressure derivative response for constant rate production for slab, cylindrical, and spherical geometries, respectively. Fig. 3.9 compares the pressure derivative response for the three geometries. The spherical model shows a smoother derivative profile than the cylindrical or the slab models, especially for small \( \lambda_{\text{ratio}} \). Due to the striking similarity of the rate and the pressure response for the three block geometries, only the slab model will be used for the remainder of this study.
Figure 3.1: Flowrate profile for slab geometry in USS model
Figure 3.2: Flowrate profile for cylindrical geometry in USS model
Figure 3.3: Flowrate profile for spherical geometry in USS model
Figure 3.4: Comparison of flowrate response of various matrix block geometries
SECTION 3. DISCUSSION

\[ \omega_m = 0.99 \]
\[ \chi = 10^{-5} \]

Spherical geometry
Cylindrical geometry

Figure 3.5: Difference in flowrate of various matrix block geometries
SECTION 3. DISCUSSION

Figure 3.6: Pressure and pressure derivative profile for slab geometry in USS model
Figure 3.7: Pressure and pressure derivative profile for cylindrical geometry in USS model
Figure 3.8: Pressure and pressure derivative profile for spherical geometry in USS model
SECTION 3. DISCUSSION

Figure 3.9: Comparison of pressure derivative profile of various matrix block geometries
3.2 Effect of the Mode of Interporosity Flow

Fig. 3.10 is a flowrate profile that compares the USS and PSS response for several values of $\lambda_{ratio}$. The interporosity flow period ends at the same time for these two models, however, the beginning time is different. In the USS model, the interporosity flow period begins much earlier than in the PSS model. Thus, the duration of the interporosity flow period of the USS model is much longer than that of the PSS model. Also, the USS model yields a higher flowrate than the PSS model during the interporosity flow period. Both models produce at exactly the same flowrate at the end of this period. Thus, the USS model yields a larger cumulative production. In the PSS model, the flowrate during the interporosity flow period is almost constant for small $\lambda_{ratio}$ values. In the USS model, on the other hand, a period of constant flowrate does not occur and a gradual decline is observed throughout the interporosity flow period. This gradual flowrate decline is particularly pronounced for large $\lambda_{ratio}$ values. Fig. 3.11 shows the comparison of pressure derivative profiles for the PSS and the USS models. The figure shows the sharp character of the PSS profile compared to the USS profile.
Figure 3.10: Comparison of PSS and USS flowrate response


Figure 3.11: Comparison of PSS and USS pressure derivative profiles
3.3 Effect of PDF Type

In this section, the rate and the pressure response for three matrix block size distributions — uniform, positively skewed linear, and negatively skewed linear distributions, are examined. Both PSS and USS modes of interporosity flow are considered. Only the slab matrix geometry is used, since the transient response is not very sensitive to the matrix block geometry (see section 3.1).

Fig. 3.12 shows the PSS flowrate profile for the three probability density functions. The duration of the interporosity flow period is the same for the three models. The probability distribution function does not affect the duration of this period as long as the limits of the distribution ($\lambda_{min}$ and $\lambda$) are the same. The response shows the lowest flowrate for the positively skewed linear distribution and the highest flowrate for the negatively skewed linear distribution. This indicates that the higher frequency of smaller matrix block sizes yields a higher flowrate. The flowrate for the uniform distribution is between that of the positively and the negatively skewed linear distributions, but is closer to the latter. This implies that the adverse effect of large blocks on reservoir producibility outweighs the advantage of small blocks. Fig. 3.13 shows the PSS pressure and the pressure derivative response for the three distributions. The positively skewed distribution, which corresponds to a higher frequency of larger blocks, shows a delayed response.

Figs. 3.14 and 3.15 show the rate profile and the pressure derivative profile for the USS model, respectively. The effect of the probability density function is less significant compared to the PSS model. However, the features which were mentioned with regard to the PSS model are observed here as well.
Figure 3.12: PSS flowrate profile for various matrix block size distributions
Figure 3.13: PSS pressure and pressure derivative profile for various matrix block size distributions

\[ \omega_m = 0.99 \]
\[ \lambda_{\text{max}} = 10^{-5} \]
\[ \lambda_{\text{min}} = 10^{-9} \]
Figure 3.14: USS flowrate profile for various matrix block size distributions
Figure 3.15: USS pressure and pressure derivative profile for **various matrix block size distributions**

\[ \omega_m = 0.99 \]
\[ \chi = 10^{-5} \]
\[ \lambda_{min} = 10^{-9} \]
SECTION 3. DISCUSSION

3.4 Effect of Fracture Intensity and Uniformity

Here, the effect of the mean and the variance of the matrix block size distribution on flowrate and cumulative recovery is considered. This is done for the uniform distribution and slab geometry for both PSS and USS models.

For PSS interporosity flow model, Fig. 3.16 shows the cumulative production at $t_D = 10^5$ versus $\lambda_{gmean}$ with $\lambda_{ratio}$ as a parameter. The parameter $\lambda_{gmean}$ is the geometrical mean of the distributed interporosity flow coefficient and is given by:

$$\lambda_{gmean} = \sqrt{\lambda_{max}\lambda_{min}}.$$  (3.2)

Two trends are apparent. First, cumulative recovery is directly proportional to $\lambda_{gmean}$. That is, as $\lambda_{gmean}$ increases, i.e., as matrix blocks become smaller or fracture intensity becomes larger, then cumulative recovery becomes larger for a given $\lambda_{ratio}$.

Second, for a given $\lambda_{gmean}$, as $\lambda_{ratio}$ increases, cumulative recovery decreases. That is, as matrix block size variability increases or fracturing becomes more nonuniform, then cumulative recovery decreases. For a given $\lambda_{gmean}$, the maximum cumulative recovery is given by a $\lambda_{ratio}$ of unity, which is the smallest possible value $\lambda_{ratio}$ and represents a reservoir with perfectly uniform fracturing, i.e., a Warren and Root type model. The effect of $\lambda_{ratio}$ is more pronounced when $\lambda_{gmean}$ is large.

On the other hand, the effect of $\lambda_{ratio}$ is negligible for small $\lambda_{gmean}$. For the USS interporosity flow model, Fig. 3.17 shows identical results.

Figs. 3.18 and 3.19 show the PSS rate response and the pressure derivative profile, respectively. It is observed that the time at which the interporosity flow period begins is determined by $\lambda_{gmean}$. Also, the time at which this period ends is determined by $\lambda_{min}$. In Fig. 3.19, the shape of the pressure derivative profile is determined by $\lambda_{ratio}$, whereas its temporal position is determined by $\lambda_{gmean}$. Thus, variations in fracture intensity (or $\lambda_{gmean}$) only affect the temporal position of the response, whereas variations in fracture uniformity (or $\lambda_{ratio}$) affect the shape of the transient response.
Figure 3.16: Cumulative production at $t_D = 10^5$ for PSS model
Figure 3.17: Cumulative production at $t_D = 10^5$ for USS model
Figure 3.18: Flow rate profile for various $\lambda_{g\text{mean}}$. 

- $\lambda_{g\text{mean}} = 10^{-5}$
- $\lambda_{g\text{mean}} = 10^{-7}$
- $\lambda_{g\text{mean}} = 10^{-9}$
- $\lambda_{\text{ratio}} = 1$
- $\lambda_{\text{ratio}} = 100$
- $\lambda_{\text{ratio}} = 10^4$
- $\lambda_{\text{ratio}} = 10^6$

$\omega_m = 0.99$

$q_D$

$t_D$
Figure 3.19: Pressure derivative profile for various $\lambda_{g\text{mean}}$. 
Section 4

Conclusion

1. The distributed formulation of interporosity flow predicts a gradual decline of the flowrate unlike the classical double porosity models which predict a sudden rate decline followed by a period of constant flowrate. This gradualness is more pronounced for larger matrix block size variabilities, i.e., cases of extremely nonuniform fracturing. Also, the gradualness is more apparent in the unsteady state model of interporosity flow.

2. Matrix block geometry does not have a significant effect on the rate or the pressure response. Spherical geometry yields slightly higher flowrate than the cylindrical geometry. The latter yields slightly higher flowrate than the slab geometry. Matrix block geometry only enters the formulation for unsteady state interporosity flow and not for pseudo-steady state interporosity flow.

3. The unsteady state formulation of interporosity flow yields higher flowrate (and hence higher cumulative recovery) than the pseudo-steady state formulation.

4. The negatively skewed linear distribution of matrix block size yields higher flowrate than the uniform distribution. The latter, in turn, yields higher flowrate than the positively skewed linear distribution. Thus, from the viewpoint of reservoir producibility, it is more advantageous to have a high frequency of small blocks than large blocks.
5. Reservoir producibility is directly proportional to fracture intensity and inversely proportional to the degree of fracture nonuniformity. Hence, the Warren and Root model which assumes perfectly uniform fracturing, yields an upper bound of reservoir producibility.
Nomenclature

\[ A = \text{area of matrix fracture interface, ft}^2 \]
\[ A_1 = \text{constant} \]
\[ A_2 = \text{constant} \]
\[ A_3 = \text{constant} \]
\[ B = \text{formation volume factor, RB/STB} \]
\[ B_1 = \text{constant} \]
\[ B_2 = \text{constant} \]
\[ B_3 = \text{constant} \]
\[ c_f = \text{fracture total compressibility, psi}'' \]
\[ c_m = \text{matrix total Compressibility, psi}^{-1} \]
\[ C_D = \text{wellbore storage coefficient, dimensionless} \]
\[ C_1 = \text{constant} \]
\[ D_1 = \text{constant} \]
\[ E_1 = \text{constant} \]
\[ E_2 = \text{constant} \]
\[ E_3 = \text{constant} \]
\[ F_1 = \text{constant} \]
\[ G_1 = \text{constant} \]
\[ g(s) = \text{a parameter in the Bessel function argument} \]
\[ h = \text{matrix block size for slab geometry, ft} \]
\[ h_D = \text{matrix block size for slab geometry, dimensionless} \]
**NOMENCLATURE**

$h_{\text{max}}$ = **maximum matrix** block size, ft

$h_{\text{min}}$ = **minimum** matrix block size, ft

$I_0(x)$ = modified Bessel function, first kind, zero order

$I_1(x)$ = modified Bessel function, first kind, first order

$k_f$ = fracture permeability, md

$k_m$ = matrix permeability, md

$K_0(x)$ = modified Bessel function, second kind, zero order

$K_1(x)$ = modified Bessel function, second kind, first order

$P_{Df}$ = fracture pressure, dimensionless

$\tilde{P}_{Df}$ = Laplace transformed fracture pressure

$P_{Dm}$ = matrix pressure, dimensionless

$\tilde{P}_{Dm}$ = Laplace transformed matrix pressure

$P_{Dw}$ = wellbore pressure response, dimensionless

$\tilde{P}_{Dw}$ = Laplace transformed wellbore pressure response

$P_f$ = fracture fluid pressure, psi

$P(h)$ = probability density function of matrix block size distribution

$P(h_D)$ = probability density function of normalized matrix block size distribution in slab geometry

$P(r_{mD})$ = probability density function of normalized matrix block size distribution in cylindrical and spherical geometries

$P_i$ = initial pressure, psi

$P_m$ = matrix fluid pressure, psi

$P_w$ = wellbore pressure response, psi

$q$ = volumetric flow rate, STB/D

$q_D$ = volumetric **flow rate**, dimensionless

$q_D$ = Laplace transformed **flow rate**

$Q_D$ = cumulative production, dimensionless
NOMENCLATURE

\( \dot{Q}_D \) = Laplace transformed cumulative production
\( r \) = radial coordinate, ft
\( r_D \) = radial Coordinate, dimensionless
\( r_m \) = matrix block radius for cylindrical and spherical geometries, ft
\( r_{mD} \) = matrix block radius for cylindrical and spherical geometries, dimensionless
\( r_{m_{\text{max}}} \) = maximum matrix block radius, ft
\( r_{m_{\text{min}}} \) = minimum matrix block radius, ft
\( r_w \) = wellbore radius, ft
\( s \) = Laplace parameter
\( S_D \) = skin factor, dimensionless
\( t \) = time, hour
\( t_D \) = time, dimensionless
\( t_{DP} \) = production time, dimensionless
\( U(h) \) = interporosity flow contribution from matrix size \( h \)
\( U(h_D) \) = interporosity flow contribution from dimensionless matrix size \( h_D \)
\( U(r_{mD}) \) = interporosity flow contribution from dimensionless matrix radius \( r_{mD} \)
\( V \) = volume of matrix block, ft\(^3\)
\( x \) = Bessel function argument
\( z \) = coordinate for slab matrix block
\( z_D \) = dimensionless coordinate for slab matrix block
\( \alpha \) = constant in the definition of interporosity flow coefficient
\( \eta \) = coordinate for matrix block radius
\( \eta_{D} \) = dimensionless coordinate for matrix block radius
\( \lambda \) = interporosity flow coefficient, dimensionless
\( \lambda_{\text{mean}} = \) geometrical mean of interporosity flow coefficient, dimensionless

\( \lambda_{\text{max}} = \) maximum interporosity flow coefficient, dimensionless

\( \lambda_{\text{min}} = \) minimum interporosity flow coefficient, dimensionless

\( \lambda_{\text{ratio}} = \) ratio of \( \lambda_{\text{max}} \) to \( \lambda_{\text{min}} \)

\( \mu = \) viscosity, cp

\( \phi_f = \) fracture porosity, dimensionless

\( \phi_m = \) matrix porosity, dimensionless

\( \omega_f = \) fracture storativity ratio, dimensionless

\( \omega_m = \) matrix storativity ratio, dimensionless

**SI METRIC CONVERSION FACTORS**

\[
\begin{align*}
\text{bbl} & \times 1.589873 \times 10^{-1} = m^3 \\
\text{cp} & \times 1.0 \times 10^{-3} = \text{pa.s} \\
\text{ft} & \times 3.048 \times 10^{-1} = m \\
\text{psi} & \times 6.894757 \times 10^{-1} = \text{kpa} \\
\text{psi}^{-1} & \times 1.450 \times 10^{-1} = \text{kpa}^{-1}
\end{align*}
\]

* Conversion factor is exact.
Bibliography


Appendix A

Derivation of Solution

This appendix contains a derivation of the constant rate and constant pressure solutions.

A.1 Slab Geometry

A.1.1 Unsteady State

In dimensionless form, Eqns. 2.1 through 2.3 become:

\[
\frac{\partial^2 P_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} + \frac{\mu}{k_f} \left( -\frac{2\pi k_f h_f r_D^2}{q \mu B} \right) \int_{\theta_{\text{min}}}^{\theta_{\text{max}}} U(h_D) P(h_D) dh_D, \quad (A.1)
\]

\[
\frac{\partial^2 P_{Dm}}{\partial z_D^2} = \frac{3 \omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D}, \quad (A.2)
\]

\[
U(h_D) = -\frac{k_m}{\mu h^2} \left( -\frac{q \mu B}{2\pi k_f h_f} \right) \frac{\partial P_{Dm}}{\partial z_D} |_{z_D=0}. \quad (A.3)
\]

Combining Eqns. A.1 and A.3 yields:

\[
\frac{\partial^2 P_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} - \int_{\theta_{\text{min}}}^{\theta_{\text{max}}} \frac{\lambda}{3} \frac{\partial P_{Dm}}{\partial z_D} |_{z_D=0} P(h_D) dh_D. \quad (A.4)
\]

In dimensionless form, Eqns. 2.4 through 2.7 are:

\[
P_{Df} = P_{Dm} = 0 \quad \text{at} \quad t_D = 0, \quad (A.5)
\]
APPENDIX A. DERIVATION OF SOLUTION

\[ C_D \frac{\partial P_D}{\partial t_D} - \frac{\partial P_D}{\partial r_D} = 1; \ \text{at} \ r_D = 1, \quad (A.6) \]

\[ P_D = P_D - S_D \frac{\partial P_D}{\partial r_D}; \ \text{at} \ r_D = 1, \quad (A.7) \]

\[ P_D = 0; \ \text{as} \ r_D \to \infty. \quad (A.8) \]

Matrix boundary conditions are:

\[ P_{Dm} = P_{Df}; \ \text{at} \ z_D = 0, \quad (A.9) \]

and

\[ \frac{\partial P_{Dm}}{\partial z_D} = 0; \ \text{at} \ z_D = 1. \quad (A.10) \]

Eqns. A.2 and A.4, subject to the six conditions, Eqns. A.5 through A.10 are to be solved. Applying Laplace transformation to these equations yields:

\[ \frac{\partial^2 \tilde{P}_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial \tilde{P}_{Df}}{\partial r_D} = \omega_s \tilde{P}_{Df} - \int_{h_{\text{min}}}^{1} \frac{\lambda}{3} \frac{\partial \tilde{P}_{Dm}}{\partial z_D} |_{z_D = 0} P(h_D) dh_D, \quad (A.11) \]

\[ \frac{\partial^2 \tilde{P}_{Dm}}{\partial z_D^2} = \frac{3 \omega_m}{\lambda} s \tilde{P}_{Dm}, \quad (A.12) \]

\[ \tilde{P}_{Df} = \tilde{P}_{Dm} = 0; \ \text{at} \ t_D = 0, \quad (A.13) \]

\[ C_D s \tilde{P}_{Dw} - \frac{\partial \tilde{P}_{Df}}{\partial r_D} = \frac{1}{s}; \ \text{at} \ r_D = 1, \quad (A.14) \]

\[ \tilde{P}_{Dw} = \tilde{P}_{Df} - S_D \frac{\partial \tilde{P}_{Df}}{\partial r_D}; \ \text{at} \ r_D = 1, \quad (A.15) \]

\[ \tilde{P}_{Df} = 0; \ \text{as} \ r_D \to \infty, \quad (A.16) \]

\[ \tilde{P}_{Dm} = \tilde{P}_{Df}; \ \text{at} \ z_D = 0, \quad (A.17) \]

\[ \frac{\partial \tilde{P}_{Dm}}{\partial z_D} = \Theta \ \text{at} \ z_D = 1. \quad (A.18) \]

The general solution of Eqn. A.12 is:

\[ \tilde{P}_{Dm} = A_1 \cosh \left( \sqrt{\frac{3 \omega_m}{\lambda}} z_D \right) + B_1 \sinh \left( \sqrt{\frac{3 \omega_m}{\lambda}} z_D \right). \quad (A.19) \]

Applying Eqn. A.17:

\[ A_1 = \tilde{P}_{Dm} = \tilde{P}_{Df}. \quad (A.20) \]
Next, differentiating Eqn. A.19 and using Eqn. A.18:

\[ B_1 = -\tanh \left( \sqrt{\frac{3\omega_m \sigma}{\lambda}} \right) \tilde{P}_{Df}. \]  

(A.21)

Thus:

\[ \frac{\partial \tilde{P}_{D_m}}{\partial z_D} \bigg|_{z_D=0} = -\sqrt{\frac{3\omega_m \sigma}{\lambda}} \tanh \left( \sqrt{\frac{3\omega_m \sigma}{\lambda}} \right) \tilde{P}_{Df}. \]  

(A.22)

Substituting Eqn. A.22 into Eqn. A.11, and rearranging yields:

\[ r_D^2 \frac{\partial^2 \tilde{P}_{Df}}{\partial r_D^2} + r_D \frac{\partial \tilde{P}_{Df}}{\partial r_D} - r_D^2 g(s) \tilde{P}_{Df} = 0, \]  

(A.23)

where:

\[ g(s) = \omega_f + \omega_m \int_{\text{meas}}^{1} \sqrt{\frac{\lambda}{3\omega_m \sigma}} \tanh \left( \sqrt{\frac{3\omega_m \sigma}{\lambda}} \right) P(h_D) dh_D. \]  

(A.24)

The general solution for Eqn. A.23 is:

\[ \tilde{P}_{Df} = C_1 K_0(\sqrt{sg(s)} r_D) + D_1 I_0(\sqrt{sg(s)} r_D). \]  

(A.25)

Applying boundary condition, Eqn. A.16:

\[ \tilde{P}_{Df} = C_1 K_0(\sqrt{sg(s)} r_D). \]  

(A.26)

Differentiating Eqn. A.26:

\[ \frac{\partial \tilde{P}_{Df}}{\partial r_D} = -C_1 \sqrt{sg(s)} K_1(\sqrt{sg(s)} r_D). \]  

(A.27)

From the skin and storage conditions, Eqns. A.14 and A.15:

\[ C_1 = \frac{\ell}{s \left[ C_D s \{ K_0(\sqrt{sg(s)}) + S_D \sqrt{sg(s)} K_1(\sqrt{sg(s)}) \} + \sqrt{sg(s)} K_1(\sqrt{sg(s)}) \} \right]. \]  

(A.28)

Then, from Eqns. A.26 and A.28, \( \tilde{P}_{Df} \) is obtained as:

\[ P_{Df} = \frac{K_0(\sqrt{sg(s)} \ell)}{s \left[ C_D s \{ K_0(x) + S_D x K_1(x) \} + x K_1(x) \right]} ), \]  

(A.29)

where:

\[ x = \sqrt{sg(s)}, \]  

(A.30)
and:

\[ P_{Dw} = \frac{K_0(x) + S_D x K_1(x)}{s [C_D s \{K_0(x) + S_D x K_1(x)\} + x K_1(x)]}. \]  

(A.31)

The relation between flowrate and wellbore pressure in Laplace space is [22]:

\[ \tilde{q}_D = \frac{1}{s^2 P_{Dw}}. \]  

(A.32)

Then, flowrate for no storage case is:

\[ \tilde{q}_D = \frac{x K_1(x)}{s \{K_0(x) + S_D x K_1(x)\}}. \]  

(A.33)

Cumulative production is given as:

\[ \dot{Q}_D = \frac{x K_1(x)}{s^2 \{K_0(x) + S_D x K_1(x)\}}. \]  

(A.34)

**A.1.2 Pseudo-Steady State**

Solution for pseudo-steady state interporosity flow is obtained by eliminating the spacial dependency of time rate of change of pressure. That is, Eqn. A.2 becomes:

\[ \frac{\partial^2 P_{Dm}}{\partial z_D^2} = \frac{3 \omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D} \equiv E_1(t_D), \]  

(A.35)

where \( E_1(t_D) \) is independent of \( z_D \). This partial differential equation is solved as:

\[ \frac{\partial P_{Dm}}{\partial z_D} = E_1(t_D) z_D + F_1, \]  

(A.36)

\[ P_{Dm} = \frac{1}{2} E_1(t_D) z_D^2 + F_1 z_D + G_1. \]  

(A.37)

Applying boundary condition, Eqns. A.9, and A.10:

\[ F_1 = -E_1(t_D), \]  

(A.38)

\[ G_1 = P_{Df}. \]  

(A.39)

Then:

\[ P_{Dm} = \frac{1}{2} E_1(t_D) z_D^2 - E_1(t_D) z_D + P_{Df}. \]  

(A.40)
APPENDIX A. DERIVATION OF SOLUTION

By differentiating Eqn. A.40 and using Eqn. A.35:
\[
\frac{\partial P_{Dm}}{\partial z_D} \bigg|_{z_D=0} = -E_1(t_D) = -\frac{3\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D}. \tag{A.41}
\]

Substituting Eqn. A.41 into Eqn. A.4, the diffusivity equation becomes:
\[
\frac{\partial^2 P_{DF}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{DF}}{\partial r_D} = \omega f \frac{\partial P_{DF}}{\partial t_D} + \omega_m \int_{h_{min}}^{1} \frac{\partial P_{Dm}}{\partial t_D} P(h_D) dh_D. \tag{A.42}
\]

In Eqn. A.40, by averaging \(E_1(t_D)\) across the matrix block thickness, the following equation is obtained:
\[
P_{Dm} - P_{DF} = \left( \frac{1}{2} z_D^2 - z_D \right) E_1(t_D)
= -\frac{1}{3} E_1(t_D)_{\text{avg}}. \tag{A.43}
\]

Then:
\[
E_1(t_D)_{\text{avg}} = 3(P_{DF} - P_{Dm}). \tag{A.44}
\]

Eqns. A.44 and A.35 give:
\[
\frac{\partial P_{Dm}}{\partial t_D} = \frac{\lambda}{\omega_m} (P_{DF} - P_{Dm}). \tag{A.45}
\]

Applying Laplace transformation to Eqns. A.42 and A.45:
\[
\frac{\partial^2 \tilde{P}_{DF}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial \tilde{P}_{DF}}{\partial r_D} = \omega f s \tilde{P}_{DF} + \int_{h_{min}}^{1} s \tilde{P}_{Dm} P(h_D) dh_D, \tag{A.46}
\]

and
\[
s \tilde{P}_{Dm} = \frac{\lambda}{\omega_m} \left( \tilde{P}_{DF} - \tilde{P}_{Dm} \right). \tag{A.47}
\]

From Eqn. A.47:
\[
\tilde{P}_{Dm} = \frac{\lambda}{\omega_m s + \lambda} \tilde{P}_{DF}. \tag{A.48}
\]

Substituting Eqn. A.48 into Eqn. A.46, the following equation is obtained:
\[
r_D^2 \frac{\partial^2 \tilde{P}_{DF}}{\partial r_D^2} + r_D \frac{\partial \tilde{P}_{DF}}{\partial r_D} - r_D^2 s g(s) \tilde{P}_{DF} = 0, \tag{A.49}
\]

where:
\[
g(s) = \omega_f + \omega_m \int_{h_{min}}^{1} \frac{\lambda}{\omega_m s + \lambda} P(h_D) dh_D. \tag{A.50}
\]

Eqn. A.49 is the same as Eqn. A.23 in the unsteady state model, which yields the same general solution. The procedure and solutions of Eqns. A.25 through A.34 are valid for the pseudo-steady state model also.
A.2 Cylindrical Geometry

A.2.1 Unsteady State

In dimensionless form, Eqns. 2.1 through 2.3 become:

\[
\frac{\partial^2 P_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} + \frac{\mu}{k_f} \left( \frac{2\pi k_f h_f r_D^2}{q\mu} \right) \int_{r_{\text{min}} }^{r_{\text{max}}} U(r_{mD}) P(r_{mD}) dr_{mD},
\]  

(A.51)

\[
\frac{\partial^2 P_{Dm}}{\partial \eta_D^2} + \frac{1}{\eta_D} \frac{\partial P_{Dm}}{\partial \eta_D} = \frac{6\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial \eta_D},
\]  

(A.52)

\[
U(r_{mD}) = \frac{2k_m}{\mu r_D^2} \left( \frac{q\mu B}{2\pi k_f h_f} \right) \left. \frac{\partial P_{Dm}}{\partial \eta_D} \right|_{\eta_D=1}.
\]  

(A.53)

Combining Eqn. A.51 and Eqn. A.53 yields:

\[
\frac{\partial^2 \tilde{P}_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial \tilde{P}_{Df}}{\partial r_D} = \omega_f \frac{\partial \tilde{P}_{Df}}{\partial t_D} + \int_{r_{\text{min}} }^{r_{\text{max}}} \lambda \left. \frac{\partial P_{Dm}}{\partial \eta_D} \right|_{\eta_D=1} P(r_{mD}) dr_{mD}.
\]  

(A.54)

Eqns. A.5 through A.8 are used as initial and boundary conditions in cylindrical geometry, too. As for the boundary conditions for the matrix coordinate, the following are used instead of Eqns. A.9 and A.10:

\[
P_{Dm} = P_{Df} \quad \text{at} \quad \eta_D = 1,
\]  

(A.55)

\[
P_{Dm} \text{ is finite at } \eta_D = 0.
\]  

(A.56)

Applying Laplace transform to Eqns. A.54 and A.52:

\[
\frac{\partial^2 \tilde{P}_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial \tilde{P}_{Df}}{\partial r_D} = \omega_f s \tilde{P}_{Df} + \int_{r_{\text{min}} }^{r_{\text{max}}} \frac{\lambda}{3} \left. \frac{\partial \tilde{P}_{Dm}}{\partial \eta_D} \right|_{\eta_D=1} P(r_{mD}) dr_{mD},
\]  

(A.57)

and

\[
\frac{\partial^2 \tilde{P}_{Dm}}{\partial \eta_D^2} + \frac{1}{\eta_D} \frac{\partial \tilde{P}_{Dm}}{\partial \eta_D} = \frac{6\omega_m}{\lambda} s \tilde{P}_{Dm}.
\]  

(A.58)

The boundary conditions in Laplace space are given by Eqns. A.13 through A.16 end the following two equations:

\[
\tilde{P}_{Dm} = \tilde{P}_{Df} \quad \text{at} \quad \eta_D = 1,
\]  

(A.59)

\[
\tilde{P}_{Dm} \text{ is finite at } \eta_D = 0.
\]  

(A.60)
The general solution for Eqn. A.58 is given as:

\[ P_{Dm} = A_2 K_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \eta_D \right) + B_2 I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \eta_D \right). \] (A.61)

From boundary condition, Eqn. A.60, \( P_{Dm} \) is finite at \( \eta_D = 0 \), therefore:

\[ A_2 = 0. \] (A.62)

From the other boundary condition, Eqn. A.59:

\[ B_2 = \frac{P_{Df}}{I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)}. \] (A.63)

Then:

\[ P_{Dm} = \frac{I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \eta_D \right)}{I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)} P_{Df}. \] (A.64)

By differentiating Eqn. A.64 and evaluating at \( \eta_D = 1 \):

\[ \frac{\partial P_{Dm}}{\partial \eta_D} \bigg|_{\eta_D=1} = \frac{\sqrt{\frac{6\omega_m s}{\lambda}} I_1 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)}{I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)} P_{Df}. \] (A.65)

Substituting Eqn. A.65 into Eqn. A.57:

\[ r_D^2 \frac{\partial^2 P_{Df}}{\partial r_D^2} + r_D \frac{\partial P_{Df}}{\partial r_D} - r_D \frac{\partial P_{Df}}{\partial r_D} - r_D g(s) P_{Df} = 0, \] (A.66)

where:

\[ g(s) = \omega_f + \omega_m \int_{r_{mm}}^{1} \sqrt{\frac{2\lambda}{3\omega_m s}} \frac{I_1 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)}{I_0 \left( \sqrt{\frac{6\omega_m s}{\lambda}} \right)} P(r_{mD}) dr_{mD}. \] (A.67)

Eqn. A.65 is the same as Eqn. A.23 in the slab geometry, so solutions shown by Eqns. A.29 through A.34 are valid for this geometry too, while \( g(s) \) is different.

### A.2.2 Pseudo-Steady State

The procedure is exactly the same as the case of slab geometry. Eqn. A.52 is assumed to be independent of space, which yields:

\[ \frac{\partial^2 P_{Dm}}{\partial \eta_D^2} + \frac{1}{\eta_D} \frac{\partial P_{Dm}}{\partial \eta_D} = \frac{6\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D} = E_2(t_D), \] (A.68)
where $E_2(t_D)$ is independent of $\eta_D$. Solving this partial differential equation yields:

$$P_{Dm} = \frac{1}{4} E_2(t_D) (\eta_D^2 - 1) + P_{Df}. \quad (A.69)$$

Differentiating, evaluating at $\eta_D = 1$, and considering Eqn. A.68:

$$\frac{\partial P_{Dm}}{\partial \eta_D} |_{\eta_D=1} = \frac{3\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D}. \quad (A.70)$$

Substituting Eqn. A.70 into Eqn. A.54, the diffusivity equation becomes:

$$\frac{\partial^2 P_{Df}}{\partial t_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} + \omega_m \int_{r_{min}}^{1} \frac{\partial P_{Dm}}{\partial t_D} P(r_m) dr_m. \quad (A.71)$$

In Eqn. A.69, averaging $E_2(t_D)$ across the matrix radius:

$$P_{Dm} - P_{Df} = \frac{1}{4} \left( \eta_D^2 - 1 \right) E_2(t_D)$$

$$= -\frac{1}{6} E_2(t_D)_{avg}. \quad (A.72)$$

Then,

$$E_2(t_D)_{avg} = 6(P_{Df} - P_{Dm}). \quad (A.73)$$

Eqns. A.68 and A.73 give:

$$\frac{\partial P_{Dm}}{\partial t_D} = \frac{\lambda}{\omega_m} (P_{Df} - P_{Dm}). \quad (A.74)$$

This equation is the same as Eqn. A.45 and fracture flow equation, Eqn. A.71, is also the same as Eqn. A.42 for the slab geometry. Then, the procedure to obtain the function $g(s)$ follows that of Eqs. A.46 through A.50. The function $g(s)$ obtained is the same as that of the slab geometry (except $r_m$ should be used in these equations instead of $h$).

### A.3 Spherical Geometry

#### A.3.1 Unsteady State

In this case, the fracture flow equation, Eqn. A.51, is valid. Matrix flow equation, Eqn. 2.2, becomes:

$$\frac{\partial^2 P_{Dm}}{\partial \eta_D^2} + \frac{2}{\eta_D} \frac{\partial P_{Dm}}{\partial \eta_D} = \frac{9\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D}, \quad (A.75)$$
and Eqn. 2.3 becomes:

\[
U(r_m) = \frac{3k_m}{\mu r_m^2} \left( -\frac{q\mu B}{2\pi k_f h_f} \right) \frac{\partial P_{Dm}}{\partial \eta_D} \bigg|_{\eta_D=1}. \tag{A.76}
\]

Combining Eqn. A.51 and Eqn. A.76 yields:

\[
\frac{\partial^2 P_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} + \int_{r_{min}}^{r_{max}} \frac{\lambda}{3} \frac{\partial P_{Dm}}{\partial \eta_D} \bigg|_{\eta_D=1} P(r_m) dr_m. \tag{A.77}
\]

Initial and boundary conditions are the same as the case of cylindrical geometry. Eqns. A.5 through A.8 and Eqns. A.55 and A.56 are used. Applying Laplace transform to Eqns. A.77 and A.75:

\[
\frac{\partial^2 \tilde{P}_{Dm}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial \tilde{P}_{Dm}}{\partial r_D} = \omega_f s \tilde{P}_{Df} + \int_{r_{min}}^{r_{max}} \frac{\lambda}{3} \frac{\partial \tilde{P}_{Dm}}{\partial \eta_D} \bigg|_{\eta_D=1} P(r_m) dr_m. \tag{A.78}
\]

and

\[
\frac{\partial^2 \tilde{P}_{Dm}}{\partial \eta_D^2} + \frac{2}{\eta_D} \frac{\partial \tilde{P}_{Dm}}{\partial \eta_D} = \frac{9\omega_m}{\lambda} s \tilde{P}_{Df}. \tag{A.79}
\]

The conditions in Laplace space are shown in Eqns. A.13 through A.16 and Eqns. A.59 and A.60.

The general solution for Eqn. A.79 is given as:

\[
\tilde{P}_{Dm} = \frac{1}{\eta_D} \left\{ A_3 \cosh \left( \sqrt{\frac{9\omega_m}{\lambda}} \eta_D \right) + B_3 \sinh \left( \sqrt{\frac{9\omega_m}{\lambda}} \eta_D \right) \right\}. \tag{A.80}
\]

From boundary condition, Eqn. A.60, \( \tilde{P}_{Dm} \) is finite at \( \eta_D = 0 \), therefore:

\[
A_3 = 0. \tag{A.81}
\]

From another boundary condition, Eqn. A.59:

\[
B_3 = \frac{\tilde{P}_{Df}}{\sinh \left( \sqrt{\frac{9\omega_m}{\lambda}} \right)}. \tag{A.82}
\]

Then:

\[
\tilde{P}_{Dm} = \frac{1}{\eta_D} \frac{\sinh \left( \sqrt{\frac{9\omega_m}{\lambda}} \eta_D \right)}{\sinh \left( \sqrt{\frac{9\omega_m}{\lambda}} \right)} \tilde{P}_{Df}. \tag{A.83}
\]
APPENDIX A. DERIVATION OF SOLUTION

By differentiating Eqn. A.83 and evaluating at $\eta_D = 1$:

$$\left. \frac{\partial P_{Dm} \right|_{\eta_D=1} = \left\{ \sqrt{\frac{9\omega_m s}{\lambda}} \coth \left( \sqrt{\frac{9\omega_m s}{\lambda}} \right) - 1 \right\} P_{Df}. \quad (A.84)$$

Substituting Eqn. A.84 into Eqn. A.78:

$$r_D^2 \frac{\partial^2 P_Df}{\partial r_D^2} + \frac{\partial P_Df}{\partial \eta_D} - r_D^2 g(s) P_Df = 0, \quad (A.85)$$

where:

$$g(s) = \omega_f \int_{r_{min}}^{1} \frac{\lambda}{\beta s} \left\{ \sqrt{\frac{9\omega_m s}{\lambda}} \coth \left( \sqrt{\frac{9\omega_m s}{\lambda}} \right) - 1 \right\} P(r_{mD})dr_{mD}. \quad (A.86)$$

Eqn. A.85 is the same as Eqn. A.25 for the slab geometry, so solutions obtained by Eqns. A.29 through A.34 are also valid for this geometry, while $g(s)$ is different.

### A.3.2 Pseudo-Steady State

The procedure is exactly the same as the case of slab geometry. Eqn. A.75 is assumed to be independent of space, which yields:

$$\frac{\partial^2 P_{Dm}}{\partial \eta_D^2} + \frac{2}{\eta_D} \frac{\partial P_{Dm}}{\partial \eta_D} = \frac{9\omega_m}{\lambda} \frac{\partial P_{Dm}}{\partial t_D} \equiv E_3(t_D), \quad (A.87)$$

where $E_3(t_D)$ is independent of $\eta_D$. Solving this partial differential equation gives:

$$P_{Dm} = \frac{1}{6} E_3(t_D) (\eta_D^2 - 1) + P_{Df}. \quad (A.88)$$

Differentiating Eqn. A.88, evaluating at $\eta_D = 1$, and using Eqn. A.87:

$$\left. \frac{\partial P_{Dm}}{\partial \eta_D} \right|_{\eta_D=1} = \frac{3\omega_m}{\lambda} P_{Dm} \quad (A.89)$$

Substituting Eqn. A.89 into Eqn. A.77, the diffusivity equation becomes:

$$\frac{\partial^2 P_{Df}}{\partial r_D^2} + \frac{1}{r_D} \frac{\partial P_{Df}}{\partial r_D} = \omega_f \frac{\partial P_{Df}}{\partial t_D} + \omega_m \int_{r_{min}}^{1} \frac{\partial P_{Dm}}{\partial t_D} P(r_{mD})dr_{mD}. \quad (A.90)$$

In Eqn. A.88, averaging $E_3(t_D)$ across the matrix radius:

$$P_{Dm} - P_{Df} = \frac{1}{6} (\eta_D^2 - 1) E_3(t_D)$$

$$= \frac{1}{9} E_3(t_D) \left. \right|_{\text{avg}}. \quad (A.91)$$
Then:

\[ E_3(t_D)|_{avg} = 9(P_{D_f} - P_{D_m}). \]  

(Eqns. A.87 and A.92 give:

\[ \frac{\partial P_{D_m}}{\partial t_D} = \frac{\lambda}{\omega_m} (P_{D_f} - P_{D_m}). \]  

This equation is the same as Eqn. A.45 and the fracture flow equation, Eqn. A.90, is also the same as Eqn. A.42 for the slab and the cylindrical cases. Then, the procedure for obtaining the function \( g(s) \) follows that of Eqns. A.46 through A.50. The function \( g(s) \) is the same as that for slab and cylindrical cases (except \( \tau_m \) should be used in these equations instead of \( h \)).
Appendix B

Computer Programs

This section contains the computer programs which are used in this study. To solve some of the equations, IMSL fortran routines were used.
Calculation of PD, qD, and QD
for Slab, Cylindrical, and Spherical Geometries

P.S.S. & U.S.S. ------

Katsunori Fujiwara Apr. 29, 89

*******************************************************************************
implicit real*8 (a-h, o-z)
common m, /a/sn, cd, xlamda, ylamda, omgm.
dimension td(121), pd(121), xlmd(10), ylmd(10), dpd(120),
       rate(121), cum(121)
open (unit=3, file='pl.dat')
open (unit=4, file='dp1.dat')
open (unit=7, file='ratel.dat')
open (unit=8, file='cum1.dat')
rewind (unit=3)
rewind (unit=4)
rewind (unit=7)
rewind (unit=8)
n=10
m=1
read(5,*) sn
read(5,*) cd
read(5,*) omgm
read(5,*) nxlmd
do 60 i=1, nxlmd
  60 read(5,*) xlmd(i)
read(5,*) nylmd
do 70 i=1, nylmd
  70 read(5,*) ylmd(i)
td(1)=0.1
do 40 j=1, 120
  40 td(j+1)=(10.0**0.1)*td(j)
        continue
do 20 i=1, nxlmd
  xlamda-xlmd(i)
do 30 k=1, nylmd
  ylamda-ylmd(k)
write (3, *) 121
write (4, *) 120
write (7, *) 121
write (8, *) 121
do 50 l=1, 121
  call pwd(td(l), n, pd(l), rate(l), cum(l))
write (3, *) td(l), pd(l)
write (7, *) td(l), rate(l)
write (8, *) td(l), cum(l)
SO continue
     do 55 l=1,120
     dpd(l)=(pd(l+1)-pd(l))/(dlog(td(l+1))-dlog(td(l)))
     write(4,*) td(l),dpd(l)
 55 continue
30 continue
20 continue
C
C
   stop
   end
\section*{Calculation of $PD, qD,$ and $QD$

for Slab, Cylindrical, and Spherical Geometries

---

Katsunori Fujiwara Apr. 29, 89

\begin{verbatim}
function plapl(s)
implicit real*8(a-h,o-z)
double precision k0,kl,gg,xx
common sn/a/sn,cd,xlamda,ylamda,omgm

if (xlamda.eq.ylamda) then
  gg=1.0-omgm+omgm*xlamda/(omgm*s+xlamda)
else
  gg=1.0-omgm+omgm*dsqrt(xlamda/omgm/s)/(1.0-
&  dsqrt(xlamda/ylamda))
& (datan (dsqrt(ylamda/omgm/s)-datan (sqrt(xlamda/omgm/
& s))))
endif

xx=dsqrt(s*gg)
k0=dbsk0(xx) kl=dbskl(xx)
plapl=(k0+sn*xx*kl)/(s*(cd*s*(k0+sn*xx*kl)+xx*kl))
return
end

function plap2(s)
implicit real*8(a-h,o-z)
double precision k0,kl,gg,plap,xx
common m,/a/sn,cd,xlamda,ylamda,omgm

if(xlamda.eq.ylamda) then
  gg=1.0-omgm+omgm*xlamda/(omgm*s+xlamda)
else
  gg=1.0-omgm+omgm*dsqrt(xlamda/omgm/s)/(1.0-
& dsqrt(xlamda/ylamda))
& (datan (dsqrt(ylamda/omgm/s)-datan (sqrt(xlamda/omgm/
& s))))
endif

xx=dsqrt(s*gg)
k0=dbsk0(xx) kl=dbskl(xx)
plap=(k0+sn*xx*kl)/(s*(cd*s*(k0+sn*xx*kl)+xx*kl))
plap2=1./s/s/plap
return
end

function plap3(s)
implicit real*8(a-h,o-z)
double precision k0,kl,gg,plap,xx
\end{verbatim}
common m, a, sn, cd, xlamda, ylamda, omgm

if (xlamda.eq. ylamda) then
  gg=1.0-omgm+omgm*xlamda/(omgm*s+xlamda)
else
  gg=1.0-omgm+omgm*dsqrt(xlamda/omgm/s)/(1.0- &
    dsqrt(xlamda/ylamda))
    & (datan(dsqrt(ylamda/omgm/s))-datan (sqrt(xlamda/omgm/ &
    & s)))
endif

xx=dsqrt(s*gg)
k0=dsk0(xx)
kl=dsk1(xx)
plap=(k0+sn*xx*kl)/(s*(cd*s*(k0+sn*xx*kl)+xx*kl))
plap3=1./s/s/s/plap
return
end
Calculation of \( P_D, q_D, \) and \( Q_D \)

for Slab Geometry ------ U.S.S. ------

Katsunori Pujiwara

---

Function \( \text{plapl}(s, \text{ans}) \)

\[
\text{plapl}(s, \text{ans}) = \frac{k_0 + \text{sn} \times xx \times k_1}{s \times (\text{cd} \times s \times (k_0 + \text{sn} \times xx \times k_1) + xx \times k_1)}
\]

Function \( \text{plap2}(5, \text{ans}) \)

\[
\text{plap2}(s, \text{ans}) = \frac{k_0 + \text{sn} \times xx \times k_1}{s \times (\text{cd} \times s \times (k_0 + \text{sn} \times xx \times k_1) + xx \times k_1)}
\]

Function \( \text{plap3}(s, \text{ans}) \)

\[
\text{plap3}(s, \text{ans}) = \frac{k_0 + \text{sn} \times xx \times k_1}{s \times (\text{cd} \times s \times (k_0 + \text{sn} \times xx \times k_1) + xx \times k_1)}
\]
if (xlamda .ne. ylamda) then
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)/(1.0-
  & dsqrt(xlamda/ylamda))*ans
else
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)*
  & dtanh (dsqrt(3.*omgm*s/xlamda))
endif

ct-dsqrt(s*gg)
k0=dbsk0(xx)
k1=dbsk1(xx)
plap(k0+sn*xx*k1)/((s*(cd*s*(k0+sn*xx*k1)+xx*k1))
plap3=1./s/s/s/plap
return
end

c Definition of function f which is used for integration

function f(x)
  implicit real*8 (a-h,o-z)
  f=dtanh(x)/x
  return
end
function plapl(s, ans)
implicit real*8 (a-h,o-z)
double precision k0, k1, gg, xx
common m, /a/sn, cd, xlamda, ylamda, omgm

if(xlamda .ne.ylamda) then
  gg=1.0-omgm+omgm*dsqrt(2.*xlamda/3./omgm/s)/(1.0-
    & dsqrt(xlamda/ylamda))*ans
else
  gg=1.0-omgm+omgm*dsqrt(2.*xlamda/3./omgm/s)*
    & dbsile(dsqrt(6.*omgm*s/xlamda))/
    & dbsi0e(dsqrt(6.*omgm*s/xlamda))
endif

xx=dsqrt(s*gg)
kk0=dbsk0(xx)
kl=dbskl(xx)
plapl=(k0+sn*xx*kk1)/(s*(cd*s*(k0+sn*xx*kk1)+xx*kl))
return
end

function plap2(s, ans)
implicit real*8 (a-h,o-z)
double precision k0, k1, gg, xx
common m, /a/sn, cd, xlamda, ylamda, omgm

if(xlamda .ne.ylamda) then
  gg=1.0-omgm+omgm*dsqrt(2.*xlamda/3./omgm/s)/(1.0-
    & dsqrt(xlamda/ylamda))*ans
else
  gg=1.0-omgm+omgm*dsqrt(2.*xlamda/3./omgm/s)*
    & dbsile(dsqrt(6.*omgm*s/xlamda))/
    & dbsi0e(dsqrt(6.*omgm*s/xlamda))
endif

xx=dsqrt(s*gg)
kk0=dbsk0(xx)
kl=dbskl(xx)
plap2=(k0+sn*xx*kk1)/(s*(cd*s*(k0+sn*xx*kk1)+xx*kl))
plap2=1./s/s/plap
return
end

function plap3(s, ans)
implicit real*8 (a-h,o-z)
double precision k0, k1, gg, xx
common m, /a/sn, cd, xlamda, ylamda, omgm

if(xlamda .ne.ylamda) then


\[ gg = 1.0 - \text{omgm} + \text{omgm} \times \text{dsqrt} \left( \frac{2. \times \text{ylamda}}{3. \times \text{omgm} / \text{s}} \right) / (1.0 - \text{dsqrt} \left( \frac{\text{ylamda}}{\text{ylamda}} \right)) \times \text{ans} \] 

\text{else} 

\[ gg = 1.0 - \text{omgm} + \text{omgm} \times \text{dsqrt} \left( \frac{2. \times \text{ylamda}}{3. \times \text{omgm} / \text{s}} \right) \times \text{dsbile} \left( \text{dsqrt} \left( \frac{6. \times \text{omgm} \times \text{s}}{\text{ylamda}} \right) \right) / (1.0 - \text{dsbile} \left( \text{dsqrt} \left( \frac{6. \times \text{omgm} \times \text{s}}{\text{ylamda}} \right) \right)) \] 

\text{endif} 

\text{else} 

\[ gg = l.0 - \text{omgm} + \text{omgm} \times \text{dsqrt} \left( \frac{2. \times \text{ylamda}}{3. \times \text{omgm} / \text{s}} \right) \times \text{dsbile} \left( \text{dsqrt} \left( \frac{6. \times \text{omgm} \times \text{s}}{\text{ylamda}} \right) \right) / (1.0 - \text{dsbile} \left( \text{dsqrt} \left( \frac{6. \times \text{omgm} \times \text{s}}{\text{ylamda}} \right) \right)) \] 

\text{endif} 

\[ \text{xx} = \text{dsqrt} \left( s \times \text{gg} \right) \] 

\[ k0 = \text{dbsk0} \left( \text{xx} \right) \] 

\[ k1 = \text{dbsk1} \left( \text{xx} \right) \] 

\[ \text{plap} = \left( k0 + \text{sn} \times \text{xx} \times k1 \right) / \left( \text{s} \times \left( \text{cd} \times \text{s} \times \left( \text{k0} + \text{sn} \times \text{xx} \times k1 \right) + \text{xx} \times k1 \right) \right) \] 

\text{plap} = l.0 / \text{s} / \text{s} / \text{s} / \text{plap} 

\text{return} 

\text{end} 

\text{c} 

\text{Definition of function f which is used for integration} 

\text{function f(x)} 

\text{implicit real*8 (a-h,o-z)} 

\text{double precision i0e,ile} 

\text{c} 

\[ i0e = \text{dbsi0e} \left( \text{x} \right) \] 

\[ ile = \text{dbsile} \left( \text{x} \right) \] 

\[ f = ile / i0e / \text{x} \] 

\text{c} 

\text{return} 

\text{end}
*************** Calculation of \( P_D, q_D, \) and \( Q_D \)

* for Spherical Geometry ----- U.S.S. ----- 

* 

Katsunori Fujiwara

******************************

```
function plapl(s, ans1, ans2)
    implicit real*8 (a-h, o-z)
    double precision k0, k1, gg, xx
    common m, /a/sn, cd, xlamda, ylamda, omgm

    if (xlamda .ne. ylamda) then
        gg = 1.0 - omgm + omgm * dsqrt(xlamda/omgm/s)/
        & (-dsqrt(xlamda)*6./s/(1.0-dsqr(xlamda/ylamda))*ans1
    else
        gg = 1.0 - omgm + omgm * dsqrt(xlamda/omgm/s)/dtanh
        & (dsqrt(9.*omgm*s/xlamda) - xlamda/3./s)
    endif

    xx = dsqrt(s*gg)
    k0 = dbsk0(xx)
    k1 = dbsk1(xx)
    plapl = (k0 + sn*xx*k1)/s*(cd*s*(k0 + sn*xx*k1) + xx*k1))
    return
end

function plap2(s, ans1, ans2)
    implicit real*8 (a-h, o-z)
    double precision k0, k1, gg, xx
    common m, /a/sn, cd, xlamda, ylamda, omgm

    if (xlamda .ne. ylamda) then
        gg = 1.0 - omgm + omgm * dsqrt(xlamda/omgm/s)/
        & (-dsqrt(xlamda)*6./s/(1.0-dsqr(xlamda/ylamda))*ans1
    else
        gg = 1.0 - omgm + omgm * dsqrt(xlamda/omgm/s)/dtanh
        & (dsqrt(9.*omgm*s/xlamda) - xlamda/3./s)
    endif

    xx = dsqrt(s*gg)
    k0 = dbsk0(xx)
    k1 = dbsk1(xx)
    plap = (k0 + sn*xx*k1)/s*(cd*s*(k0 + sn*xx*k1) + xx*k1))
    plap2 = 1./s/s/plap
    return
end

function plap3(s, ans1, ans2)
    implicit real*8 (a-h, o-z)
    double precision k0, k1, gg, xx
    common m, /a/sn, cd, xlamda, ylamda, omgm

    if (xlamda .ne. ylamda) then
```
gpl.0-omgm+omgm*dsqrt(xlambda/omgm/s)/(1.0-
& dsqrt(xlambda/ylambda))*ans1
& -dsqrt(xlambda/ylambda)/6./s/(1.0-dsqrt(xlambda/ylambda))*ans2
else
  gg=1.0-omgm+omgm*dsqrt(xlambda/omgm/s)/dtanh
  {dsqrt(9.*omgm*s/xlambda)}-xlambda/3./s
endif

c  Definition of function f which is used for integration
function f1(x)
 implicit real*8 (a-h,o-z)
  f1=1./dctanh (x)/x
 return
end

function f2(x)
 implicit real*8 (a-h,o-z)
  f2=1./dsqrt(x)
 return
end
function plapl(s)
  implicit real*8(a-h,o-z)
  double precision k0,kl,gg,xx
  common m,/a/sn,cd,xlамда,ylамда,omgm
  common /b/nflag

  if (xlамда .eq. ylамда) then
    gg=1.0-omgm+omgm*xlамда/(omgm*s+xlамда)
  else
    if( nflag .eq. 1) then
      gg=1.0-omgm+omgm*dsqrt(xlамда/omgm/s)/(1.0-
        dsqrt(xlамда/ylамда))
      & (datan(dsqrt(ylамда/omgm/s)) - datan (sqrt(xlамда/omgm/
        s))
    else
      endif
    ratio=xlамда/ylамда
    arg1=dsqrt(xlамда/omgm/s)
    arg2=dsqrt(ylамда/omgm/s)
    arg3=1.0-ratio
    aab=omgm*xlамда/arg3/arg3
    bbb=dlog(ylамда*(omgm*s+xlамда)/xlамда/(omgm*s+
      ylамда))
    ccc=datan(arg2)-datan(arg1)
    if( nflag .eq. 2) then
      gg=1.0-omgm+aaa*(bbb/omgm/s-2.*ccc/dsqrt(omgm*s*ylамда))
    else
      endif
    if( nflag .eq. 3) then
      gg=1.0-omgm+aaa*(2.*ccc/dsqrt(omgm*s*xlамда)-bbb/omgm/s)
    else
      endif
  endif

  xx=dsqrt(s*gg)
  kl=dskkl(xx)
  plapl=(k0+sn*xx*kl)/(s*(cd*s*(k0+sn*xx*kl)+xx*kl))
  return
end

function plap2(s)
  implicit real*8(a-h,o-z)
  double precision k0,k1,gg,plap,xx
  common m,/a/sn,cd,xlамда,ylамда,omgm
  common /b/nflag

  if (xlамда .eq. ylамда) then

function plap3(s)
  implicit real*8 (a-h,o-z)
  double precision k0, kl, gg, plap, xx
  common m, a/sn, cd, xlamda, ylamda, omgm
  common /b/nflag

  if (xlamda .eq. ylamda) then
    gg=1.0-omgm+omgm*xlamda/(omgm*s+xlamda)
  else
    if( nflag .eq. 1) then
      gg=1.0-omgm+omgm*dsqrt(xlamda/omgm/s)/(1.0-
      &    dsqrt(xlamda/ylamda))
    &    (datan(dsqrt(ylamda/omgm/s))-datan(sqrt(xlamda/omgm/)
      &    s)))
    else
      endif
      ratio=xlamda/ylamda
      argl=dsqrt(xlamda/omgm/s)
      arg2=dsqrt(ylamda/omgm/s)
      arg3=1.0-ratio
      aaa=omgm*xlamda/arg3/arg3
     bbb=dlog(ylamda*(omgm*s+xlamda)/xlamda/(omgm*s+
      &    ylamda))
      &    ccc=datan(arg2)-datan(arg1)
      if( nflag .eq. 2) then
        gg=1.0-omgm+aaa*(bbb/omgm/s-2.*ccc/dsqrt(omgm*s*ylamda))
      else
        endif
      if( nflag .eq. 3) then
        gg=1.0-omgm+aaa*(2.*ccc/dsqrt(omgm*s*xlamda)-bbb/omgm/s)
      else
        endif 
      endif

    xx=dsqrt(s*gg)
k0=dbsk0(xx)
k1=dbskl(xx)
plap=-(k0+sn*xx*kl)/(s*(cd*s*(k0+sn*xx*kl)+xx*kl))
plap2=1./s/s/plap 
  return
end

end
if( nflag .eq. 3) then
  gg=1.0-omgm+aaa*(2.*ccc/sqrt(omgm*s*xlamda)-bbb/omgm/s)
else
endif
endif

xx=dsqrt(s*gg)
k0=dbsk0(xx)
k1=dbsk1(xx)
plap=(k0+sn*xx*k1)/(s*(cd*s*(k0+sn*xx*k1)+xx*k1))
plap3=1./s/s/s/plap
return
end
function plapl(s,ans1,ans2)
implicit real*8 (a-h,o-z)
double precision k0,k1,gg,xx
common m,/a/sn,cd,xlamda,ylamda,omgm
common /b/nflag

if (xlamda .eq. ylamda) then
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)*
  & dtanh (dsqrt(3.*omgm*s/xlamda))
else if (nflag .eq. 1) then
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)/((1.0-
  & dsqrt(xlamda/ylamda))*ansl)
else endif
if (nflag .eq. 2) then
  gg=1.0-omgm+omgm/((1.0-dsqrt(xlamda/ylamda))**2.)
  & -2.*xlamda/dsqrt(3.*omgm*s*ylamda)*ans2
else endif
if (nflag .eq. 3) then
  gg=1.0-omgm+omgm/((1.0-dsqrt(xlamda/ylamda))**2.)
  & (2.*dsqrt(xlamda/3./omgm/s)*anal
  & -2.*xlamda/3./omgm/s*ans2)
else endif
endif

xx=dsqrt(s*gg)
k0=dbsk0(xx)
k1=dbsk1(xx)
plapl=(k0+sn*xx*k1)/(s*(cd*s*(k0+sn*xx*k1)+xx*k1))
return
end

function plap2(s,ans1,ans2)
implicit real*8 (a-h,o-z)
double precision k0,k1,gg,plap,xx
common m,/a/sn,cd,xlamda,ylamda,omgm
common /b/nflag

if (xlamda .eq. ylamda) then
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)*
  & dtanh (dsqrt(3.*omgm*s/xlamda))
else if (nflag .eq. 1) then
  gg=1.0-omgm+omgm*dsqrt(xlamda/3./omgm/s)/(1.0-
& \text{dsqrt}(\text{xlamda}/\text{ylamda})^*\text{ans1}
  
else
  endif
  if (nflag .eq. 2) then
    \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)*\text{ans2} 
  & (2.*\text{xlamda}/3./\text{omgm}/s*\text{ylamda})^*\text{ans1}) 
else
  endif
  if (nflag .eq. 3) then
    \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)*\text{ans1} 
  & (2.*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)*\text{ans2} 
  & -2.*\text{xlamda}/3./\text{omgm}/s*\text{ans2}) 
else
  endif
endif

\text{xx}=\text{dsqrt}(s^*\text{gg})
\text{k0}=\text{dbsk0}(\text{xx})
\text{kl}=\text{dbskl}(\text{xx})
\text{plap}=\text{plap}3(\text{k0}+\text{sn}^*\text{xx}^*\text{kl})/(\text{s}*(\text{cd}^*\text{s}*(\text{k0}+\text{sn}^*\text{xx}^*\text{kl})+\text{xx}^*\text{kl}))
\text{plap2}=1./\text{s}^2/\text{plap}
return
end

\text{function plap3}(s,\text{ansl},\text{ans2})
\text{implicit real*8 (a-h,o-z)}
\text{double precision k0,kl,gg,plap,xx}
\text{common m,} /a/sn, cd, xlamda, ylamda, omgm
\text{common /b/nflag}

\text{if} (\text{xlamda} .eq. \text{ylamda}) \text{then}
  \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)^*\text{dthanh}(\text{dsqrt}(3.*\text{omgm}/s/\text{xlamda}))
\else
  \text{if} (nflag .eq. 1) \text{then}
    \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)/(1.0-\text{dsqrt}(\text{xlamda}/\text{ylamda}))^*\text{ans1}
  \else
    \text{endif}
  \if (nflag .eq. 2) then
    \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)^*\text{ans2}
  & (2.*\text{xlamda}/3./\text{omgm}/s*\text{ylamda})^*\text{ans1}) 
else
  endif
  if (nflag .eq. 3) then
    \text{gg}=1.0-\text{omgm}+\text{omgm}^*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)^*\text{ans1} 
  & (2.*\text{dsqrt}(\text{xlamda}/3./\text{omgm}/s)*\text{ans1} 
  & -2.*\text{xlamda}/3./\text{omgm}/s*\text{ans2}) 
else
  endif
endif

\text{xx}=\text{dsqrt}(s^*\text{gg})
\text{k0}=\text{dbsk0}(\text{xx})
\text{kl}=\text{dbskl}(\text{xx})
\text{plap}=\text{plap}3(\text{k0}+\text{sn}^*\text{xx}^*\text{kl})/(\text{s}*(\text{cd}^*\text{s}*(\text{k0}+\text{sn}^*\text{xx}^*\text{kl})+\text{xx}^*\text{kl}))
\text{plap3}=1./\text{s}^2/\text{s}/\text{plap}
return
end
C Definition of function f which is used for integration

function f1(x)
  implicit real*8 (a-h, o-z)
  f1 = dtanh(x)/x
  return
end

function f2(x)
  implicit real*8 (a-h, o-z)
  f2 = dtanh(x)
  return
end
THE STEHFEST ALGORITHM

*********************

SUBROUTINE PWD(TD,N,PD,rate,cum)
THIS FUNCTION COMPUTES NUMERICALLY THE LAPLACE TRANSFORM
INVERSE OF F(S).
IMPLICIT REAL*8 (A-H,0-Z)
DIMENSION G(50),V(50),H(25)
COMMON M,/a/sn,cd,xlamma,ylamda,cmgm

NOW IF THE ARRAY V(I) WAS COMPUTED BEFORE THE PROGRAM
GOES DIRECTLY TO THE END OF THE SUBROUTINE TO CALCULATE
F(S).
IF (N.EQ.M) GO TO 17
M=N
DLOGTW=0.6931471805599
NH=N/2

THE FACTORIALS OF 1 TO N ARE 'CALCULATED INTO ARRAY G.
DO 1 I=2,N
G(I)=G(I-1)*I
CONTINUE

TERMS WITH K ONLY ARE CALCULATED INTO ARRAY H.
DO 6 I=2,NH
FI=I
IF (I-NH) 4,5,6
H(I)=FI**NH*G(2*I)/(G(NH-I)*G(I)*G(I-1))
GO TO 6
CONTINUE

THE TERMS (-1)**NH+1 ARE CALCULATED.
FIRST THE TERM FOR I=1
SN=2*(NH-NH/2)**2-1
THE REST OF THE SN'S ARE CALCUATED IN THE MAIN ROUTINE.

THE ARRAY V(I) IS CALCULATED.
DO 7 I=1,N
FIRST SET V(I)=0
V(I)=0.
THE LIMITS FOR K ARE ESTABLISHED.
THE LOWER LIMIT IS K1=INTEGR((I+1/2))
K1=(I+1)/2
THE UPPER LIMIT IS K2=MIN(I,N/2)
K2=I
IF (K2-NH) 8,8,9
K2=NH
THE SUMMATION TERM IN V(I) IS CALCULATED.
DO 10 K=K1,K2
IF (2*K-I) 12,13,12
IF (I-K) 11,14,11
V(I)=V(I)+H(K)/(G(I-K)*G(2*K-I))
GO TO 10
V(I)=V(I)+H(K)/(G(I-K))
GO TO 10
V(I)=V(I)+H(K)/(2*K-I)
CONTINUE
THE V(I) ARRAY IS FINALLY CALCULATED BY WEIGHTING ACCORDING TO SN.
V(I) = SN*V(I)

THE TERM SN CHANGES ITS SIGN EACH ITERATION.
SN = -SN

CONTINUE

THE NUMERICAL APPROXIMATION IS CALCULATED.

A = DLOGTW/TD
PD = 0
rate = 0.
cum = 0.
DO 15 I = 1, N
   ARG = A*I
   PD = PD + V(I)*PLAP1(ARG)
   rate = rate + V(I)*PLAP2(ARG)
   cum = cum + V(I)*PLAP3(ARG)
15 CONTINUE
PD = PD*A
rate = rate*A
cum = cum*A
RETURN
END