TECHNO-ECONOMIC AND RISK EVALUATION OF A THERMAL
RECOVERY PROJECT

SUPRI TR 102

By
Sameer Joshi
William E. Brigham
Louis M. Castanier

RECEIVED
JUL 14 1997
OSTI

July 1997

Performed Under Contract No. DE-FG22-96BC14994

Stanford University
Department of Petroleum Engineering
Stanford, California

National Petroleum Technology Office
U. S. DEPARTMENT OF ENERGY
Tulsa, Oklahoma

MASTER

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government.

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831; prices available from (615) 576-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield VA 22161

Techno-Economic And Risk Evaluation Of A Thermal Recovery Project

SUPRI TR 102

By
Sameer Joshi
William E. Brigham
Louis M. Castanier

July 1997

Work Performed Under Contract No. DE-FG22-96BC14994

Prepared for
U.S. Department of Energy
Assistant Secretary for Fossil Energy

Thomas B. Reid, Project Manager
National Petroleum Technology Office
P.O. Box 3628
Tulsa, OK 74101

Prepared by:
Stanford University
Department of Petroleum Engineering
Stanford, California 94305-2220

Acknowledgments

I was able to carry out this work due to the funding provided me by the Department of Petroleum Engineering and SUPRI-A, which I thankfully acknowledge. I would also like to acknowledge the support provided by the US Department of Energy [DE-FG22-96BC14994] and the SUPRI-A Industrial Affiliates for this research work.

DISCLAIMER

**Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.**

Abstract

The Wilmington Steamflood of Union Pacific Resources Co. (UPRC) at Long Beach, CA was initiated in 1989, in a previously waterflooded reservoir. Average initial reservoir oil saturation, at the start of the steamflood, was 35%.

Field production data were studied, to derive an overall energy balance for the steamflood, to calculate the steamflood capture efficiency and predict future steamflood performance. Heat-losses due to produced fluids were also calculated. Predicted production schedules from the model were history-matched with field production data

The reservoir parameters (porosity, ϕ , net thickness, h_n , initial oil saturation, S_{oi} , and residual oil saturation, S_{or}) were evaluated statistically using both Gaussian and triangular distributions. These resulted in distributed recovery predictions. The Gaussian distributions behaved as predicted; but of great importance, the skewed triangular distributions also behaved in much the same manner. The results fit closely with predictions using logical formulas to predict expected values, peak values and standard variations of recoveries. This result is important, for it indicates that complete Monte-Carlo simulations may not be necessary.

All steamflood calculations were carried out using a PC-based spreadsheet program. The major results were as follows:

- The capture efficiency of the Wilmington steamflood was calculated at 60%. This is an acceptable value, taking into account the reservoir geometry and history

- The calculated heat balance showed high heat-loss to adjacent formations and through produced fluids. Of the cumulative heat injected at the time of the study, 21% had been lost to vertical conduction and 21% through produced fluids.
- Predicted production schedules indicated that up to 43% of the oil in place (at steamflood initiation) could be recovered by the steamflood.

Contents

Acknowledgments	ii
Abstract	iii
List of Figures	vii
List of Tables	x
1 Introduction and Background	1
2 Wilmington Steam Flood	3
2.1 Calculation of Heat Losses	7
2.1.1 Surface Steam Distribution Network	8
2.1.2 Injection Wellbores	11
2.1.3 Produced Fluids	12
2.1.4 Heat Lost to Adjacent Formations	15
2.2 Calculation of Production Rates	18
2.3 Summary of Results	24
3 Analysis by Gadjica Method	26
4 Review of Statistics and Risk Analysis	30
5 Monte-Carlo Simulation and the Ramey Method	38
5.1 Simulation Using @RISK	38
5.2 Application to Ramey Steam Flood Model	42
5.3 Distributions Used	46
5.4 Results Obtained	53

5.5	Statistical Analysis of the Results	63
6	Economic and Risk Analysis	69
6.1	Discounted Cash Flow Model	70
6.2	Steam Flood Project Costs	71
6.3	Statistical Analysis of Results	77
7	Summary and Conclusions	87
	Bibliography	90
	Appendix	93
A.1	FORTRAN source code for Gadjica method	93
A.1.1	File tot.f	93
A.1.2	Include file d1.com	135
A.1.3	Main data file main.dat	135

List of Figures

2.1	Wilmington Steam Flood Pattern (after Lim <i>et al.</i> [4])	4
2.2	Wilmington Field Location Map and Cross-Section (after Lim <i>et al.</i> [4])	5
2.3	Pipeline Heat Loss Rates	10
2.4	Total Heat Loss For Different Pipe Sizes	10
2.5	Heat Loss Rate vs Time for an Average Well	12
2.6	Surface & Bottomhole Temperatures vs Time	14
2.7	Produced Fluid Heat Loss Rate vs Time	14
2.8	Produced Fluid Heat Loss Rate as Percentage of Cumulative Injection	15
2.9	Percentage of Cumulative Injected Heat Lost to Adjacent Function	18
2.10	Distribution of Cumulative Injected Heat	20
2.11	History Match with Production Rates (Capture Efficiency, 60%)	21
2.12	Predicted Oil Production Rates	23
3.1	History Match Using Modified Gadjica Method	28
5.1	Sample Histogram—Cumulative Oil Production at Breakthrough	45
5.2	Sample Production Rate Summary Graph	46
5.3	Porosity Histogram for Normal Distribution	48
5.4	Porosity Histogram for Triangular Distribution	48
5.5	Porosity Histogram for Triangular (10/90) Distribution	49
5.6	S_{or} Histogram for Normal Distribution	49
5.7	S_{or} Histogram for Triangular Distribution	50

5.8	S_{or} Histogram for Triangular (10/90) Distribution	50
5.9	Net Pay Histogram for Normal Distribution	51
5.10	Net Pay Histogram for Triangular Distribution	51
5.11	Net Pay Histogram for Triangular (10/90) Distribution	52
5.12	Summary Cumulative Oil Production—Normal	54
5.13	Summary Cumulative Oil Production—Triangular	54
5.14	Summary Cumulative Oil Production—Triangular (10/90)	55
5.15	Summary Production Rate Graph—Normal	55
5.16	Summary Production Rate Graph—Triangular	56
5.17	Summary Production Rate Graph—Triangular (10/90)	56
5.18	Summary Forecasted Cumulative Production—Normal	58
5.19	Summary Forecasted Cumulative Production—Triangular	58
5.20	Summary Forecasted Cumulative Production—Triangular (10/90)	59
5.21	Summary Forecasted Production Rates—Normal	59
5.22	Summary Forecasted Production Rates—Triangular	60
5.23	Summary Forecasted Production Rates—Triangular (10/90)	60
5.24	Cumulative Production Histogram at Breakthrough—Normal	61
5.25	Cumulative Production Histogram at Breakthrough—Triangular	61
5.26	Cumulative Production Histogram at Breakthrough—Triangular (10/90)	62
6.1	Oil Price Histogram (Year 9)	80
6.2	Steam Cost Histogram (Year 9)	80
6.3	O&M Cost Histogram (Year 9)	81
6.4	Summary Graph for Future Oil Prices	82
6.5	Summary Graph for Future Steam Costs	82

6.6	Summary Graph for Future O&M Costs.....	83
6.7	Cash Flow Histogram (Year 9)	83
6.8	Summary Graph for Cash Flow	84
6.9	Histogram for NPV (20% Discount Rate)	85
6.10	Summary Graph for NPV	85
6.11	Histogram for IRR	86

List of Tables

2.1	Wilmington Cumulative Heat Loss Calculations.....	19
2.2	Capture Efficiencies for Steam Flood Projects	22
2.3	Summary of Predicted Results	23
5.1	Sample @RISK Spreadsheet.....	43
5.2	Probability Distributions Used.....	44
5.3	Expected Cumulative Production Using Different Distributions	62
5.4	Calculation of Standard Deviations for Gaussian Distribution.....	65
5.5	Calculation of Recovery for Non-Gaussian Distributions	66
5.6	Calculation of Standard Deviations for Non-Gaussian Distributions	68
6.1	Breakdown of Initial Capital Investment for 130 acre Steam flood Project	74
6.2	Expectations and Standard Deviations of Economic Variables	76
6.3	Economic Analysis Spreadsheet	78
6.4	Statistical Comparison of Results	79

Chapter 1

Introduction and Background

This study is a revision, as well as extension, of a continuing SUPRI-A research effort concerning the techno-economic analysis of thermal recovery projects.

Williams et al. [1] in 1980 developed a model to evaluate the comparative economics of steam versus in-situ combustion projects. Their work was further advanced in 1987 by Ramage et al. [2].

Both of the above research groups focused their efforts on development of general engineering-economic models comparing the relative merits of steam injection and in-situ combustion applied to the same oilfield. This was partly because accurate operating data was unavailable for most projects, as mentioned by Ramage in his report. However this work differs from earlier efforts in that detailed data on production operations was available for the Wilmington Field, by courtesy of Union Pacific Resources Company (UPRC). Thus it became possible to carry out a specific engineering analysis of the Wilmington steam flood and attempt a history match of the production data. This history matching is important because all the economic calculations and predictions would depend strongly upon a correct production rate schedule.

Hence this work applies Ramey's generalization of the Marx and Langenheim method for production performance evaluation and prediction, after calculating an overall heat balance for the project. An important feature of this model is the calculation of heat loss due to produced fluids which, it is shown, is a significant fraction of the total heat loss.

A risk analysis is carried out along with the economic analysis, by using Monte-Carlo simulation. The risk analysis work of Ramage et al. is extended to Monte-Carlo simulation on the reservoir parameters, in addition to the economic parameters. The software used allows both Monte-Carlo and Latin Hypercube sampling methods. Any number of iterations can be made for each simulation, and a multiple number of simulations can be done in a single analysis. All software used will run easily on available PC/Macintosh computers.

A part of this work carried out for this thesis has been presented earlier [3], at three different professional conferences.

Chapter 2

Wilmington Steam Flood

The Wilmington Oilfield, Los Angeles County, California is the third largest in the United States, after Prudhoe Bay and the East Texas fields, on the basis of cumulative oil production, with a total of 2.4 billion barrels produced. A good description of the initial pilot steam flood carried out by UPRC in this field from 1982-1989 is given by Lim et al., 1993 [4]. A map of the original steam flood pattern is attached in Fig. 2.1. Figure 2.2 is taken from Lim et al. and shows the location of the initial steam pilot as well as a cross-section of Wilmington Field reservoirs. The pilot 20 acre steam flood is now part of the main 130 acre Wilmington steam flood. The portion of the field studied had initially been water flooded to a low oil saturation of 35% before the main steam flood was initiated in 1989.

The steam flooded reservoir is unconsolidated sandstone in the Tar Zone (T, D1 and D3 members) with average gross and net thicknesses of 170 feet and 128 feet, respectively. Average reservoir pressure was estimated at 350 psig in 1993. The steam flood project has a surface pattern area of 129.73 acres divided into approximately 17 individual seven-spot

Figure 2.1 Wilmington Steam Flood Pattern (after Lim et al.[4])

Figure 2.2 Wilmington Field Location Map and Cross-Section (after Lim et al.[4])

patterns, each of 7.5 acres. Average well depth is 2500 feet. The oil in place at steam flood initiation was calculated at 18 million barrels with a movable oil volume of 12.9 million barrels, assuming an S_{oi} of 35% (at steam flood initiation), a final S_{or} of 10% and an average porosity of 40%.

It can be calculated from the preceding reservoir parameters that the product of porosity and oil saturation for this field is 0.14. This agrees well with a criterion that is often used to determine whether thermal recovery will be feasible for an EOR project, that the product ϕS_{oi} should be greater than 0.10.

Volumetric calculations were carried out using the EarthVision® and Interactive Surface Modeling (ISM)® programs on a Silicon Graphics® workstation in connection with the present research work. All other calculations were carried out using PC based spreadsheet programs. Reservoir and production data were obtained from UPRC records and computer databases. Production data was available from the time of steam flood initiation in May 1989 up to July 1993.

The method of analysis followed in this study was essentially as follows:

1. Determine a heat balance for the steam flood project by calculating individual heat losses from the following individual portions of the total system:
 - Surface steam distribution network
 - Injection wellbores
 - Produced fluids

- Vertical conduction to adjacent formations
2. After accounting for the above heat losses, calculate a saturated steamed area and, thus, a steam zone volume.
 3. From the steam zone volume, calculate displaced oil versus time and attempt to match the history of the recorded production rates. This step also determines the capture efficiency of the steam flood.
 4. Use the history matched model to predict future oil production under different assumed operating scenarios.

It was determined that the largest amount of heat loss out of the steam flood area was to the adjacent formations and by way of produced fluids, each being 21% of the cumulative injected heat up to July 1993. As of July 1993, 4.79 million barrels of oil had been produced by the steam flood. It is predicted in the present research work that 7.8 million barrels of oil will be recovered by the time the pattern area is fully saturated with steam. The capture efficiency of the Wilmington steam flood was calculated to be at 60%.

The following sections will describe the models used and the individual results obtained to achieve the above overall results.

2.1 Calculation of Heat Losses

The first step of the analysis consisted of calculating the magnitude of heat losses in different subsystems of the steam flood system. Heat losses were calculated, as mentioned

earlier, for the following subsystems:

- Surface steam distribution network
- Injection wellbores
- Produced fluids
- Vertical conduction to adjacent formations

2.1.1 Surface Steam Distribution Network

The distribution network consisted of a total length of about 18,900 feet of different sizes of surface pipelines which conveyed steam from the generator to the wellheads. Pipes varied in size from 1.5 inches to 14 inches with insulation thicknesses ranging from one inch to four inches. The basic equation used to calculate heat losses per unit length, $\dot{Q}_{ls}$, was taken from Prats [5] and is given by Prats' Eq. 10.1 (page 125),

$$\dot{Q}_{ls} = \frac{T_b - T_a}{R_h} \quad (2.1)$$

where T_b is the bulk temperature of the fluid in the pipe, T_a is the ambient temperature and R_h is the specific thermal resistance (thermal resistance per unit length) of pipe or wellbore, given in units of (BTU/ft-Day-°F)⁻¹. A steam temperature of 580°F, ambient temperature at 60°F and an average wind speed of 20 mph normal to the surface pipelines was assumed, with steady state conditions.

For a pipe covered with insulation, R_h is given by Prats' Eq. 10.2 (page 126). The effect of scale deposits, contact thermal resistances and the film coefficient of heat transfer between steam and pipe can be neglected, and then the equation reduces to

$$R_h = \left[\frac{1}{\lambda_{ins}} \ln \frac{r_{ins}}{r_o} + \frac{1}{h_{fc} r_{ins}} \right] \quad (2.2)$$

where λ_{ins} is the thermal conductivity of the insulation, r_o is the outer radius of the pipe, r_{ins} is the outer radius of the insulation and h_{fc} is the heat transfer coefficient due to forced convection on the outer surface of the insulation.

The heat transfer coefficient, h_{fc} , can be calculated using Prats' Eq. 10.4 (page 127)

$$h_{fc} r_{ins} = 18 v_w^{0.6} r_{ins}^{0.6} \quad (2.3)$$

where v_w is the wind speed in miles per hour normal to the pipeline. The calculated heat loss rates for different insulated pipe sizes used in the field are shown in Fig. 2.3. The insulation thicknesses are 1 inch on the 1.5 inch pipelines, 2 inches on the 3 inch pipelines, 3 inches on the 4 inch through 10 inch pipelines and 4 inches on the 12 inch and 14 inch pipelines.

The total heat loss for the different insulated pipe sizes used in the field, is shown in Fig. 2.4. The highest heat loss contributions are from the 3 inch and 14 inch pipelines which account for a loss of 28.8 MMBTU/D (57.5% of the total pipeline heat loss). The total loss due to all pipe was calculated as being 50 MMBTU/D, which is only 0.5% of the co-generator output.

Figure 2.3 Pipeline Heat Loss Rates

Figure 2.4 Total Heat Losses For Different Pipe Sizes

2.1.2 Injection Wellbores

The specific thermal resistance for a well is given by Prats' Eq. 10.6 (page 129). After neglecting insignificant terms it reduces to

$$R_h = \frac{1}{2\pi} \left[\frac{1}{\lambda_{ins}} \ln \frac{r_{ins}}{r_o} + \frac{1}{h_{rc,an} r_{ins}} + \frac{1}{\lambda_{cem}} \ln \frac{r_w}{r_{co}} + \frac{f(t_d)}{\lambda_E} \right] \quad (2.4)$$

Here r_w is the well radius, λ_{cem} and λ_E are the thermal conductivities of cement and earth, $h_{rc,an}$ is the radiation and convection coefficient of heat transfer for the annulus and $f(t_d)$ is the time function that reflects the time variation in the thermal resistance of the earth. The term $f(t_d)$ is a function of the dimensionless time, t_d , given by

$$t_d = \frac{\alpha_E t}{r_w^2} \quad (2.5)$$

where α_E is the thermal diffusivity of the earth in square feet per day and t is the time in days. An iterative procedure is used to find both R_h and $h_{rc,an}$. An example calculation is shown in Prats' Example 10.2 (page 130). Calculations for an average injection well in Wilmington indicate that the heat loss rate decreases with time as shown in Fig. 2.5. An average Wilmington well is 2500 feet deep, with a well radius of 10 inches, casing diameter of 7 inches and tubing diameter of 3.5 inches.

Using an average value of the heat loss rate over 10 years, the total heat loss due to all injection wells is calculated at 130 MMBTU/D, which is about 1.5% of the co-generator capacity. Thus the total heat loss due to surface pipelines and injection wells is calculated to be only 2% of the total co-generator output. This is a low value and indicates that the insulation methods adopted for the steam injection distribution system are adequate.

Figure 2.5 Heat Loss Rate vs Time for an Average Well

2.1.3 Produced Fluids

Heat loss through produced fluids is the heat transported out of the reservoir by the produced oil, water and steam. The factors determining this loss are the oil and water flow rates, the bottomhole flowing temperatures and the fluid specific heats and enthalpies. The oil and water production rates are known. The bottomhole flowing temperatures were determined from the average surface flowline temperatures using an adaptation of Ramey's equations, given by Prats' Eq. 10.13 (page 133). Assumptions made while deriving the equation were:

- The system is at steady state, except for conduction heat losses into the earth
- Friction pressure drop is negligible

- Vertical heat transfer is by convection only
- Heat losses to the formation are radial

In this calculation, an average value of the specific heat was used over the given temperature range. The temperature over an individual monthly injection period was taken as constant. A more rigorous procedure would be to integrate $C_p dt$ over the temperature rise in an injection period. However, since all calculations were being done using a spreadsheet, it was decided to use the method described above. The specific heats of the fluids were calculated using Gambill's [6] correlations, given, for liquid hydrocarbons and oil, by:

$$C_o = \frac{(0.388 + 0.00045 T)}{\sqrt{\gamma_o}} \quad (2.6)$$

where C_o is the specific heat in BTU/lb-°F, γ_o is the specific gravity, and T the temperature in °F. For saturated water, the corresponding relation is

$$C_w = 1.0504 - 6.05 \times 10^{-4} T + 1.79 \times 10^{-6} T^2 \quad (2.7)$$

where C_w is the specific heat in BTU/lb-°F, and T is the temperature in °F. Figure 2.6 plots the measured flowline temperatures and corresponding produced BHT versus time, using loss calculation procedures.using Ramey's heat loss calculation procedures.

The total heat loss rate due to produced fluids, and that due to water alone, are plotted in Fig. 2.7. The difference between the two rates is very small, as expected, since at an oil cut of 10% the majority of fluid production is water, which also has a higher specific heat than oil.

Figure 2.6 Surface & Bottomhole Temperatures vs Time

Figure 2.7 Produced Fluid Heat Loss Rate vs Time

Figure 2.8 Produced Fluid Heat Loss as a Percentage of Cumulative Injection

The heat loss due to produced fluids as a percentage of the cumulative heat injection since the start of the steam flood (May 1989) is shown in Fig. 2.8.

It can be seen that a large fraction of the injected heat (21% to 7/93) has been lost through produced fluids. This is almost half (47%) of the total heat lost due to different mechanisms in the field up to this point of time, as is shown later in Fig. 2.10. At present, this produced heat is not being effectively utilized. This magnitude of heat loss suggests that it may be feasible to recover some of this heat by heat exchange at the surface.

2.1.4 Heat Lost to Adjacent Formations

The Marx and Langenheim method was used to calculate heat losses to adjacent formations. Since the varying average monthly heat injection rates were known, it was decided to use

Ramey's generalization of this method [7, 8] for variable rates of heat injection. The main assumptions of this model are:

- The pressure drop due to the flow is small so that the steam zone temperature throughout the reservoir is nearly constant and equal to the temperature of the injected steam.
- Heat losses from the reservoir are only by vertical conduction to adjacent formations.
- There is a piston-like vertical displacement by the steam front.

Myhill and Stegemeier (1978) [9] have pointed out that for the Marx and Langenheim model (and its extensions) to be applicable, it is not necessary that the heat front be vertical. It is only necessary that the total volume of the steam zone can be represented by the expression, $Ah/2$. In this system, A represents the sum of the upper and lower surface areas where the steam zone contacts the adjacent layers, i.e. a sloped but straight front that is advancing linearly; and h is the vertical distance between the surfaces. The results obtained using the Marx and Langenheim method are quite close to those obtained from numerical simulators. A comparison by Satter and Parrish in 1971 [10], of the heat losses obtained with the Marx and Langenheim model and those calculated from a numerical simulator, has shown that the difference between the results varies from 0.5% to 3.5%, depending upon the injection rates and pressures.

The major variables involved in the Marx and Langenheim model are

- The heat injection rate history
- Gross formation thickness
- Formation and adjacent formation thermal parameters (thermal conductivity and heat capacity)
- Elapsed time

The rate of advance of the steam front depends on the heat loss to the adjacent strata. If the heat injection rates can be expressed by constant monthly average values, then the heat remaining in the reservoir at any time is given by Prats' Eq. 5.20 (page 47). This equation was used to calculate the percentage of the net injected heat that remained in the reservoir at any time. Net injected heat is defined as the difference between the total rate at which heat is generated in or injected into the reservoir through any number of wells (located anywhere) and the total rate at which heat is withdrawn from the reservoir through production of hot fluids from any number of wells (also located anywhere). Figure 2.9 shows the percentage of the cumulative net injected heat since May 1989 that is lost to the adjacent formations.

It can be seen that the rate of heat loss was high initially, but then gradually decreased, as one would expect. By the end of May 1993, 27% of the net injected heat had been lost to vertical conduction. The cumulative heat loss data is summarized in Table 2.1.

An overall balance of the cumulative injected heat, as of 7/93, is shown in Fig. 2.10, p. 20.

Figure 2.9 Percentage of Cumulative Injected Heat Loss to Adjacent Formations

2.2 Calculation of Production Rates

A simple analytical steam flood model based on Ramey's generalization [7, 8] of Marx and Langenheim's method was used to predict oil production rates. A history match was obtained using the model and was further used to predict future oil production rates for three different cases. After accounting for adjacent formation heat losses, the model calculates a saturated steamed area and thus a steam zone volume. The model assumes that all the movable oil in the steam zone is displaced. From this a displaced oil rate can be calculated. However, this calculated rate will differ from the actual oil production rate due to the following reasons:

- The amount of oil displaced out of the steam zone will depend upon the sweep efficiency.
- All of the displaced oil will not be produced.

To account for these effects a capture efficiency is defined, which is the fraction of the displaced oil that is actually produced. The capture efficiency for the Wilmington steam flood was determined by trying to match the measured production rates with the calculated production rates. It was found that the calculated rates had to be multiplied by a factor of 0.60 to obtain a good history match. This value also provided a good match with the measured cumulative production. The unadjusted rates matched well during later times but were too high initially. Thus it was concluded that the capture efficiency for the Wilmington steam flood is 60%. This value of capture efficiency implies that the ultimate oil recovery

Table 2.1 Wilmington Cumulative Heat Loss Calculations

Date	Cumulative Heat Injection (MMMBTU)	Percentage Heat Losses To			Cumulative Not Injection (MMMBTU)	Cumulative Loss To Adjacent Layers	% Heat Retained
		Pipe	Oil Prodn.	Water Prodn.			
Dec-89	2,711.27	2.00%	0.25%	5.65%	2,497.00	12.39%	79.71%
Dec-90	6,372.65	2.00%	0.62%	12.30%	5,421.52	16.43%	68.65%
Dec-91	9,198.24	2.00%	0.87%	15.97%	7,482.17	18.90%	62.45%
Dec-92	13,007.84	2.00%	1.05%	19.63%	10,056.81	20.02%	57.29%
Jul-93	15,215.28	2.00%	1.06%	19.95%	11,714.19	21.26%	55.73%

Figure 2.10 Distribution of Cumulative Injected Heat

from the field will be 43% of the OIP. This is calculated using Eq. 2.8

$$\text{Recovery} = \text{Capture Efficiency} \times \left(\frac{S_{oi} - S_{or}}{S_{oi}} \right) \quad (2.8)$$

Using a comparison by Myhill and Stegemeier [6] of results from different steam drive field projects it can be determined that the capture efficiencies for these projects vary from 0.66 to 1.1 (a value of 1.1 indicates that oil is being produced from outside the pattern area by factors such as gravity drainage). This comparison is shown in Table 2.2.

The calculated versus actual production rates are shown in Fig. 2.11 after reducing the calculated rates by the 60% capture efficiency factor.

Figure 2.11 History Match of Production Rates (Capture Efficiency, 60%)

Future production rates were predicted for three different cases:

- [Case 1] Constant steam injection rate of 25,000 bbl/D.
- [Case 2] Constant steam injection rate of 21,000 bbl/D, which corresponds to the scenario of full injection into Phase 1C and Phase 2 while maintaining the current steam flood.
- [Case 3] Steam rate decreasing by 3,000 bbl/D every year, which corresponds to the scenario of drilling a new pattern every year outside the current pattern area, with each pattern requiring an injection rate of 3,000 bbl/D.

The predicted future rates are shown in Fig. 2.12, and summarized in Table 2.3.

Table 2.2 Capture Efficiencies for Steam Flood Projects

Field	Ec
Brea ("B" sand)	1.08
Coalinga (section 27, zone 1)	1.13
El Dorado (N-W pattern)	0.40
Inglewood	0.68
Kern River	0.81
Schoonebeek	0.81
Slocum (Phase 1)	0.62
Smackover	0.78
Tatums (Hefner steam drive)	0.77
Tia Juana	0.63
Yorba Linda ("F" sand)	1.06
Wilmington (as calculated in this study)	0.60

Figure 2.12 Predicted Oil Production Rates

	Case 1	Case 2	Case 3
Fill up Date	Dec. 1996	Mar. 1998	
Decline Rate	246 BPD/yr	193 BPD/yr	
<u>Dates</u>	<u>Oil Rate</u> <u>(BPD)</u>	<u>Oil Rate</u> <u>(BPD)</u>	<u>Oil Rate</u> <u>(BPD)</u>
12/93	2849	2137	2309
12/94	2579	1944	1559
12/95	2329	1752	885
12/96	2117	1560	269
12/97		1392	218

Table 2.3 Summary of Predicted Results

The time required for the steam zone to fill up the entire pattern area of 129.73 acres was obtained for each case studied. Total cumulative oil production to this time was calculated. Cases 1 and 2 indicate that the recovery from the steam flood, at the time saturated steam fills the area, will be 7.8 million barrels of oil, which is 43% of the calculated OIP (18 million barrels). This matches exactly with the recovery value calculated earlier using Eq. 2.8.

Although the total predicted recovery is the same for both cases, the fill up time is different as the steam injection rates are different. The fill up date for Case 1 is December 1996, and that for Case 2 is March 1998. Since the model assumes a vertical displacement front with no gravity override, steam breakthrough is quite likely to occur earlier than the predicted steam fill-up date, with correspondingly lower production rates and recovery thereafter.

2.3 Summary of Results

From the above analysis, we can deduce the following main conclusions:

1. The pipeline and tubing insulation are adequate in controlling heat losses.
2. A large fraction of the injected heat is lost through produced fluids and vertical conduction. By July 1993, 21% of the injected heat had been lost through produced fluids, which is almost half (47%) of the total heat lost. Also 21% is lost to adjacent layers.
3. Based on a simple model, the capture efficiency for the Wilmington steam flood was calculated at 60%, as compared to values of between 66% and 110% for other steam floods.

4. Over 7.3 million barrels of the OOIP (43%) will be recovered when saturated steam has filled the steam flood area.
5. The current steam flood (Case 2) is predicted to decline at 193 BOPD/yr, based upon the predicted production rates.

Chapter 3

Analysis by Gadjica Method

A semi-analytical model (SAM) was developed by Gadjica et al. [11] for steam recovery calculations. The model includes formation dip; compressible formation, water and oil; and thermal expansion of the formation, water and oil. The model is restricted to one-dimensional and two-dimensional linear cross-sectional systems. Only two wells are allowed, one injector and one producer, at each end of the reservoir. Wet steam is injected at a constant rate and enthalpy, while the production well produces at a constant flowing bottomhole pressure. Oil and water production rates are calculated by material balance.

The Gadjica Model requires specification of a rectangular reservoir with one injector and one producer. The Wilmington Steam flood consists of 7-spots with one injector and six producers. Therefore a method was needed to convert the 7-spot into an equivalent rectangular pattern with one injector and one producer.

The procedure followed was to use a number of rectangular patterns equal to the number of injectors. The area of the pattern was taken as the total area divided by the number of injectors. The injector-producer distance for the equivalent pattern was kept equal to the

average injector-producer distance for the 7-spots. The pattern width was then adjusted such that the product of length and width was equal to the average pattern area.

The average injection rate for a pattern was taken equal to the total field injection rate divided by the number of injectors.

The other data required for this model, which are not needed in the Ramey method, are

- PVT data,
- Viscosity-temperature data,
- Relative permeabilities as a function of saturation.

Using this method, a history match was attempted with the recorded production rates. The match obtained for oil production rate is shown in Fig. 3.1, on the following page.

It can be seen that the model matches the average trend of production rate fairly well. However this model predicts a much earlier water breakthrough time than the Ramey model.

This result may be due to the following factors:

- Actual steam injection rates for the steam flood varied from month to month, whereas the SAM assumes a constant injection rate.
- The Ramey method assumes a linear steam front, whereas the Gadjica method calculates the shape of the front.

Figure 3.1 History Match Using Modified Gadjica Method

- The SAM was developed for a two-dimensional linear system with only one injector and producer. However, the Wilmington steam flood has been developed in a 7-spot pattern having one injector and six producers. Thus the flow geometry is quite different.

The SAM was found to be quite a fast and convenient method, and it may be possible to extend this model to different pattern geometries with further work. It may be possible to calculate an areal sweep for the steam flood system with an extension of the model.

The FORTRAN computer code available for the model was for PC compatible systems. During this study the code was modified, and transported to a UNIX platform to make it more widely accessible, and has performed efficiently after the modification. The FORTRAN source code and associated files are included in the appendix.

Chapter 4

Review of Statistics and Risk Analysis

Uncertainty and risk are two terms which we commonly come across when analyzing any exploration prospect or oil field investment project. Risk arises because there is uncertainty about the course of future events. Risk analysis tries to quantify uncertainty by treating uncertain input parameters of a problem as random variables, which are distributed according to quantifiable statistical distribution functions. Different outcomes can then be simulated by letting a computer recalculate the model many times, using different randomly selected sets of values for the input parameters. This is like trying all valid combinations of values of the input parameters to simulate all possible outcomes. The net result of this process would be to specify the model output as a statistical distribution of probable values. This process is also sometimes called Monte-Carlo simulation.

In this chapter we review a few commonly used statistical techniques and terms and their application to Monte-Carlo simulation.

If a die is tossed once and the result recorded, we have an observation or measurement. A statistical experiment is any process of obtaining or generating an observation. The *sample space* of the experiment is the collection of all its possible *simple events*, where a simple event is an outcome that cannot be decomposed further. In this example, the sample space is the collection of numbers from 1 to 6. Specific collections of simple events are called events or random events. For example, getting an even number on tossing a die can arise from any one of three simple events—getting a 2, 4, or 6.

The *probability* of a simple event is a number that measures the likelihood that the event will occur when the experiment is performed. The probability of occurrence of a simple event A is denoted by $P(A)$. The two most important rules that govern probability are as follows. If $A_1, A_2, \dots, A_n$ are the simple events in a sample space, then:

- All probabilities for simple events must lie between 0 and 1.
For all $i = 1, 2, \dots, n$:

$$0 \leq P(A_i) \leq 1 \quad (4.1)$$

- The sum of the probabilities of all the simple events within a sample space must equal 1:

$$\sum_{i=1}^n P(A_i) = 1 \quad (4.2)$$

- The probability of an event E , is equal to the sum of the probabilities of the simple events in event E .

We write $P(A + B + \dots)$ for the probability that at least one of the events $A, B, \dots$ occurs; $P(AB\dots)$ for the probability that all the events $A, B, \dots$ occur; and $P(A | B)$ for the probability that the event A occurs when it is known that the event B has occurred. $P(A | B)$ is called the *conditional probability* of A given B . From these definitions, we have:

$$P(A + B + K) \leq P(A) + P(B) + K \quad (4.3)$$

and

$$P(AB) = P(A|B)P(B) \quad (4.4)$$

If only one of the events $A, B, \dots$ can occur, they are called *exclusive* and the equality holds in Eq. 4.3. If at least one of the events $A, B, \dots$ must occur, they are called *exhaustive* and the left hand side of Eq. 4.3 is 1. If $P(A | B) = P(A)$ we say that A and B are independent. The chance of A occurring does not depend on the occurrence of B .

A *random variable*, η , is a numerically valued function defined over a sample space. Each event in the sample space corresponds to a unique value of η . A probability distribution for a random variable, η , is a formula, table or graph that gives the probability associated with each possible value of η . Random variables may be *discrete* or *continuous*. A discrete random variable is one that can assume only a countable number of values. There are many random variables that are not discrete because the number of values they can assume is not countable. These are called continuous variables. The porosity or water saturation in a reservoir are examples of continuous variables; whereas the number of wells needed to drain a reservoir, or the number of dry holes, are examples of discrete variables.

Every random variable, η , is associated with a *cumulative distribution function* (CDF), $F(y)$, defined as the probability that the event which occurs has a value of η not exceeding a value y . This is written

$$F(y) = P(\eta \leq y) \quad (4.5)$$

We can see that $F(-\infty) = 0$ and $F(+\infty) = 1$. The CDF, $F(y)$, is a monotonically non-decreasing function of y . If $g(\eta)$ is a function of η then the *mean* or *expectation* of $g(\eta)$, μ , is defined as

$$\mu = E[g(\eta)] = \int g(y) dF(y) \quad (4.6)$$

Equation 4.6 can be interpreted in two ways:

- If $F(y)$ has a derivative $f(y)$, then it can be written as

$$E[g(\eta)] = \int g(y) f(y) dy \quad (4.7)$$

- If $F(y)$ is not continuous, but is a step function with steps of height f_i at the points y_i , then

$$E[g(\eta)] = \sum_i g(y_i) f_i \quad (4.8)$$

The expectation of $g(\eta)$ is effectively the weighted average of $g(\eta)$, with the weights being the respective probabilities of different possible values of η . The quantities $f(y)$ and f_i are called the frequency functions of the random variable, η . Usually $f(y)$ is called the probability density function (PDF).

The quantity $E(\eta^r)$ is called the r th moment of η . Also, the quantity $\mu_r = E[(\eta - \mu)^r]$, where $\mu = E(\eta)$ is known as the r th central moment of η . The most important moments are μ and μ_2 , known as the mean, μ , and variance, μ_2 , of η . For a random variable, the mean is a measure of location and the variance is a measure of dispersion about that mean. The *standard deviation* is defined as $\sigma = \sqrt{\mu_2}$. The *coefficient of variation* is σ/μ .

If η and θ are two random variables with means μ and ν respectively, then the covariance of η and θ , $\text{cov}(\eta, \theta)$, is defined as

$$\text{cov}(\eta, \theta) = E[(\eta - \mu)(\theta - \nu)] \quad (4.9)$$

The *correlation coefficient*, ρ , between η and θ is defined as

$$\rho = \frac{\text{cov}(\eta, \theta)}{\sqrt{\text{var } \eta \times \text{var } \theta}} \quad (4.10)$$

where $\text{var } \eta$ is the variance of η , and $\text{var } \theta$ is the variance of θ . The correlation coefficient always lies between ± 1 . If $\rho = 0$, then η and θ are uncorrelated, or independent random variables: they are positively correlated if $\rho > 0$, and negatively correlated if $\rho < 0$. In petroleum engineering situations, porosity and water saturation may be negatively correlated, whereas area and net pay could be positively correlated. Thus this dependency should be taken into account when carrying out simulations. Otherwise they may produce physically nonsensical scenarios—e.g. high permeability and low porosity in a reservoir.

If a linear function of n random variables, $X_1, \dots, X_n$ exists, then the expectation of the function is given by

$$E(a_1 X_1 + K + a_n X_n) = \sum_{i=1}^n a_i E(X_i) \quad (4.11)$$

and its variance by

$$Var(a_1 X_1 + K + a_n X_n) = \sum_{i=1}^n a_i^2 Var(X_i) + 2 \sum_{i=1}^n \sum_{j=i+1}^n a_i a_j Cov(X_i, X_j) \quad (4.12)$$

If $X_1, \dots, X_n$ are independent, then

$$Var(a_1 X_1 + K + a_n X_n) = \sum_{i=1}^n a_i^2 Var(X_i) \quad (4.13)$$

For a product of functions $g_1, \dots, g_n$, of n independent random variables $X_1, \dots, X_n$, we have

$$E(g_1(X_1) \cdot g_n(X_n)) = E(g_1(X_1)) \cdot E(g_n(X_n)) \quad (4.14)$$

Some well known distributions are the binomial distribution,

$$f(y) = \binom{n}{y} p^y q^{n-y} \quad (4.15)$$

where the variables are

p = probability of a success on a single trial

$q = 1-p$

n = the number of trials

y = the number of successes in n trials

the uniform distribution,

$$f(y) = \begin{cases} \frac{1}{b-a} & \text{if } a \leq y \leq b \\ 0 & \text{elsewhere} \end{cases} \quad (4.16)$$

the normal distribution,

$$f(y) = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-(y-\mu)^2}{2\sigma^2} \quad -\infty < y < \infty \quad (4.17)$$

the lognormal distribution,

$$f(y) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp \frac{-(\ln x - \mu)^2}{2\sigma^2} \quad (4.18)$$

and the exponential distribution,

$$f(y) = \frac{e^{-y/b}}{\beta} \quad 0 \leq y \leq \infty \quad (4.19)$$

All the above distributions have a common feature: each distribution function has a specified mathematical form which depends upon some unspecified constants called the parameters of the distribution (e.g. n , p , a , b , μ and σ). Williams et al. [1] describe Murphy's modification [12] of a method originally presented by Davidson and Cooper [13] for parameter estimation. This is done by achieving a range of uncertainty for each variable. High, low and most likely values are picked, such that there is only a 10% chance to be beyond the high or low values of the range. Thus there is an 80% probability that the value of the parameter lies within this range. This is known as the 80% *confidence interval range*. This information is used in estimation of parameters like mean and variance of variables. A full description of the process is given in Williams et al. [1].

The *Central Limit Theorem* is one of the most important theorems in statistics. It states that if random samples of n observations are drawn from a population with finite mean μ and standard deviation σ , then, when n is sufficiently large, the sampling distribution of the sample mean can be approximated by a normal PDF with mean μ and standard deviation $\sigma/\sqrt{n}$. This concept implies that we can use the normal distribution to approximate the sampling distribution of the sample mean, as long as the population has a finite mean and variance, and the number of measurements in the sample is sufficiently large. In fact, in many cases, $n=10$ is reasonably large, and $n=25$ is effectively infinite. See, for instance, Example 7.4 (page 232) of Mendenhall and Sincich [14]. This calculates and plots the means of computer generated random samples from different statistical distributions, of different sample sizes (n ranging from 5 to 100). As the sample size increases, the shape of the sampling distribution of the sample mean tends towards the shape of the normal distribution (symmetric and mound shaped), regardless of the shape of the probability distribution from which the sample was selected.

It can also be shown that the sampling distribution of any linear function of normally distributed variables, even those that are correlated and have different means and variances, is also a normal distribution. Similarly, it can be shown that the normal distribution can be used to approximate the binomial distribution when the number of trials, n , is large. Thus in many cases we can assume a normal distribution for some input parameters, when the actual distribution cannot be determined.

Chapter 5

Monte-Carlo Simulation and the Ramey Method

In this chapter we will describe the @RISK analysis package [15] and how it was used in Monte-Carlo simulation of the Ramey steam flood model [7, 8].

5.1 Simulation using @RISK

The @RISK package is an add-in for the Microsoft EXCEL spreadsheet. A standard spreadsheet analysis combines single-point estimates of a model's parameters to produce a single final result. Estimates of model parameters have to be made, because the values which actually occur are not known with certainty, as they are stochastic in nature. The @RISK package allows the user to specify uncertain parameters as random variables with a specified probability distribution. This procedure enables specification of the range of values that the variable can take, as well as the likelihood of occurrence of each value. The @RISK package has 28 built-in probability distributions including many common distributions such as: uniform, triangular, normal and log-normal. The distribution

functions in @RISK can themselves be used like any other EXCEL function. They may be included in a spreadsheet cell formula and can have arguments that refer to other worksheet cells.

Once the cells containing the input parameters and their distributions have been specified, the next step is to specify the output cell or range of output cells for which simulation results are desired. For example, one simple simulation may be to calculate volumetric reserves using Eq. 5.1

$$N = A h R \quad (5.1)$$

where N is volume of oil, A is the area, h the net pay and R is the recovery factor. Here the input cells would contain area, net pay and recovery factor along with a specification of their probability distributions. The output cell would contain the formula for oil volume. This output cell would be the one for which simulation results are obtained. The simulation result, in this case, would be a probability distribution of the possible values of oil reserves, given the specified input parameter distributions.

A @RISK simulation works by repetitively recalculating a worksheet, selecting a new set of values of the input parameters at each recalculation. The selection of values from the input probability distributions is called *sampling*, and each recalculation of the EXCEL worksheet is called an *iteration*. The net result of this process is a *simulation output*. In the example given above, the simulation output is a probability distribution of the oil volume produced. This output distribution would be generated by consolidating single-valued results from all the iterations.

The @RISK package allows the user to specify the number of iterations to be carried out for one simulation. The simulations can be repeated many times, with differing input parameters for each different trial. There is also a facility to continue running a simulation for a larger number of iterations than originally specified. This process is useful to check whether the simulation has converged, and that further iterations will not substantially change the output distribution obtained.

This simulation process is called Monte-Carlo simulation. However, a distinction must be made here between Monte-Carlo simulation and Monte-Carlo sampling. The @RISK package offers a choice of two different sampling methods, that is, the process by which values are selected from the input probability distributions. These two methods are Monte-Carlo sampling and Latin Hypercube sampling. Monte-Carlo sampling is entirely random. Any given sample may fall anywhere within the range of the input distribution. With enough iterations, Monte-Carlo sampling will re-create the input distribution through sampling. However, if enough iterations are not performed, then a problem of clustering may arise. This is because samples are more likely to be drawn from areas of the distribution which have higher probabilities of occurrence. Thus, in some cases, Monte-Carlo sampling may not include the effect of low probability outcomes, which may have a major impact on some results.

This problem led to the development of stratified sampling techniques like Latin Hypercube sampling. Latin Hypercube sampling is designed to accurately re-create an input distribution in fewer iterations, as compared to Monte-Carlo sampling. In stratified sampling, the cumulative distribution function (CDF) curve is divided into equal intervals on the cumulative probability scale. A sample is then randomly taken from each interval or

CHAPTER 5 MONTE-CARLO SIMULATION AND THE RAMEY METHOD

stratification of the input distribution. In this way the sampling is forced to represent values in each interval, and thus, more accurately re-creates the input distribution in fewer iterations than needed for Monte-Carlo sampling.

There are a large number of display options for viewing the simulation results, both in tabular and graphical forms. A summary report on simulation results is generated automatically at the end of the simulation. A probability distribution can be divided into equal probability increments called percentiles. The summary report gives five statistical parameters for the distribution of values in each output cell—its minimum possible value, 10th percentile, mean, 90th percentile and maximum value.

There are many options for graphically displaying the output cell data. Output data for a cell can be graphed either as a histogram or as a CDF. A summary graph for a range of output cells can be obtained, e.g. a summary graph of expected monthly production rates. Such graphs would summarize the different distributions generated for each month's production in the form of a single output graph. This process is explained, along with an example in Section 5.2.

The @RISK package also generates statistics tables for each output distribution from a simulation. This table gives results such as the minimum, maximum, mean, standard deviation, skewness and kurtosis for an output cell. The table also calculates every fifth percentile for a distribution. It can also be used to automatically calculate the probability of occurrence of different target output values for a cell. If a target value is entered for a cell then @RISK calculates its probability of occurrence. If a probability value is entered for a cell, then @RISK calculates the corresponding output target value.

5.2 Application to Ramey Steam Flood Model

As described in Chapter 2 the Ramey method was used to calculate the area of the steam zone at the end of any steam injection period. These calculated areas are used to calculate steam zone volumes and, thus, the volumes of displaced oil at the end of each period. The formula used is

$$N_d = V_s \phi (S_{oi} - S_{or}) h_n / h_t \quad (5.2)$$

where N_d is the cumulative volume of oil displaced at a particular time step, V_s is the steam zone volume at that time, ϕ is the porosity, S_{oi} and S_{or} are the initial and residual oil saturations and h_n and h_t are the net and gross pay thicknesses.

Equation 5.2 contains parameters like oil saturations, porosity and net pay. The values we have for these parameters are estimates and not exact values. Thus they were defined as random variables in the spreadsheet, with specified probability distributions. The other two variables, steam zone volume and gross pay, were assumed to be known. Monte-Carlo simulation was carried out in the manner described next.

A part of the spreadsheet model for this calculation is shown in Table 5.1. The spreadsheet has columns for time (in days), steamed area, steam zone volume, cumulative oil production and oil production rate. The input parameters mentioned in Eq. 5.2 are also specified as input cells for the spreadsheet. Using @RISK, these cells are defined as random variables having prescribed probability distributions.

Table 5.1 Sample @RISK Spreadsheet

Phi	Soi	Sor	Net Pay (ft)	Gross Pay (ft)
0.35	0.35	0.11	118.33	170.00

Month	Days per month	Cum. days	Area (acres)	V(s) (ac. ft.)	Nd (Bbl)	Qod (Bbl/D)
May-89	31	31	0.26	44.20	2.0049E+04	646.75
Jun-89	30	61	0.75	127.50	5.7834E+04	1,259.50
Jul-89	31	92	7.08	1,203.60	5.4596E+05	15,745.86
Aug-89	31	123	10.09	1,715.30	7.7806E+05	7,487.37
Sep-89	30	153	12.91	2,194.70	9.9552E+05	7,248.57
Oct-89	31	184	15.93	2,708.10	1.2284E+06	7,512.25
Nov-89	30	214	18.08	3,073.60	1.3942E+06	5,526.39
Dec-89	31	245	20.80	3,536.00	1.6039E+06	6,766.00
Jan-90	31	276	20.41	3,469.70	1.5739E+06	-970.12
Feb-90	28	304	22.89	3,891.30	1.7651E+06	6,829.96
Mar-90	31	335	25.40	4,318.00	1.9587E+06	6,243.62
Apr-90	30	365	27.85	4,734.50	2.1476E+06	6,297.52
May-90	31	396	29.58	5,028.60	2.2810E+06	4,303.37
Jun-90	30	426	31.55	5,363.50	2.4329E+06	5,063.72
Jul-90	31	457	32.43	5,513.10	2.5008E+06	2,189.00
Aug-90	31	488	34.35	5,839.50	2.6488E+06	4,776.00
Sep-90	30	518	35.90	6,103.00	2.7683E+06	3,984.14
Oct-90	31	549	37.28	6,337.60	2.8748E+06	3,432.75
Nov-90	30	579	38.78	6,592.60	2.9904E+06	3,855.62
Dec-90	31	610	40.15	6,825.50	3.0961E+06	3,407.87

The values used for the parameters are shown in Table 5.2. The output cells are defined as the spreadsheet rows containing cumulative oil produced and oil production rate. When the simulation is executed, the program will select a set of values of the input parameters by taking samples from their probability distributions. It calculates the output values—cumulative production and production rate—for each output cell specified. This process is called one iteration of the simulation.

Table 5.2 Probability Distributions Used

<u>Distribution Type</u>	<u>Ø</u>	<u>Soi</u>	<u>Sor</u>	<u>Net Pay</u>
Normal (μ, σ) EV	(0.35, 0.035) EV=0.35	(0.35, 0.035) EV=0.35	(0.1, 0.01) EV=0.100	(125, 12.5) EV= 125.00
Triangular (min, most-likely, max) EV	(0.3, 0.35, 0.4) EV=0.35	(0.3, 0.35, 0.4) EV=0.35	(0.08, 0.1, 0.15) EV=0.110	(100, 125, 130) EV= 118.33
Triangular (10%, most-likely, 90%) EV	(0.3, 0.35, 0.4) EV=0.35	(0.3, 0.35, 0.4) EV=0.35	(0.08, 0.1, 0.15) EV=0.1129	(100, 125, 130) EV=116.33

The process is then repeated with fresh sets of values of the input parameters up to the number of iterations specified. The program saves the values obtained at each iteration for each output cell and consolidates them in the form of an output probability distribution for the cell. Thus the net result of the simulation process is that the output oil production at any time is obtained as a probability distribution, rather than a fixed deterministic value. This probability distribution can be graphed either as a histogram or a CDF curve. A sample histogram for cumulative oil production at breakthrough is shown in Fig. 5.1. This was

calculated by using the normal distributions for porosity, oil saturations and net pay indicated in Table 5.2.

Additionally, all the output distributions for a range of output cells, e.g. oil production from beginning to end, can be further consolidated into one summary graph. A sample summary graph for oil production rate from start to end of injection is shown in Fig. 5.2. To do this, each output cell is plotted as one point on the summary graph. This point is the mean value of the output probability distribution for that cell. Further, each point also has an error bar showing the 10% and 90% values for that cell. The tops of light bars are the 90% values, while the bottoms of the heavy bars are the 10% values. The expected values are at the intersections of the light and heavy bars.

Figure 5.1 Sample Histogram—Cumulative Oil Production at Breakthrough

Figure 5.2 Sample Production Rate Summary Graph

5.3 Distributions Used

It was decided to try three different types of probability distributions for the input parameters, as seen in Table 5.2.

The *normal* or *Gaussian* distribution is the standard bell-shape applicable to many data sets. The parameters required to be specified are the mean and standard deviation. The values used for the different parameters are given in Table 5.2. The standard deviation is taken as 10% of the mean value for all parameters. The peak values are equal to the expected values (means) for all four parameters, ϕ , S_{oi} , S_{or} and net pay.

The *triangular distribution* is specified by three points—a minimum, most likely and a maximum. The direction of the *skew* of the triangular distribution is set by the position of the most likely value relative to the minimum and the maximum. It is important to note that

the probabilities of occurrence of the minimum and maximum values are zero. This is because these are the points where the distribution intersects the x-axis, at zero probability. From Table 5.2 it can be seen that ϕ and S_{oi} are unskewed, while S_{or} is negatively skewed and net pay is positively skewed. The most likely values for all four parameters are the same as for the Gaussian case. The expected value for S_{or} is higher, and net pay is lower, than for the Gaussian case.

The *triangular (10/90)*, or truncated triangular distribution specifies a triangular distribution with three points—one at the most likely value and one each at the specified bottom and top percentiles. The bottom and top percentiles are values greater than 0 and less than 100. Each percentile value gives the percentage of the total area under the triangle that falls to the left of the entered point. For this simulation the percentiles taken were at 10% and 90%. In this case too, S_{or} is negatively skewed while net pay is positively skewed, while the most likely values are the same as before. Again, as compared to the expected values of the other two distributions, S_{or} is the highest and net pay is the lowest, for this distribution.

Three simulations were carried out on the most likely case (Case 2) of the Wilmington steam flood described in Chapter 2. For the first simulation, the input parameters were taken as normal distributions, for the second as triangular, and for the third as triangular (10/90). Figures 5.3 through 5.11 show the histograms which were obtained for input parameters using the three different distributions. The graphs for ϕ and S_{oi} were similar in all cases, as they have the same numerical values, so the graphs for S_{oi} are not shown.

Figure 5.3 Porosity Histogram for Normal Distribution

Figure 5.4 Porosity Histogram for Triangular Distribution

Figure 5.5 Porosity Histogram for Triangular (10/90) Distribution

Figure 5.6 S_{or} Histogram for Normal Distribution

Figure 5.7 S_{or} Histogram for Triangular Distribution

Figure 5.8 S_{or} Histogram for Triangular (10/90) Distribution

Figure 5.9 Net Pay Histogram for Normal Distribution

Figure 5.10 Net Pay Histogram for Triangular Distribution

Figure 5.11 Net Pay Histogram for Triangular (10/90) Distribution

All simulations were repeated for 500, 1000 and 1500 Latin Hypercube iterations. It was found that there was some difference between the results of 500 and 1000 iterations. However there was very little difference between the results of 1000 and 1500 iterations. The comparisons were made by independently calculating both sides of Eq. 5.5, as described in Section 5.5. Equation 5.5 was suggested by Prof. W.E. Brigham for analyzing the Monte-Carlo statistical outputs, but it also turned out to be a sensitive equation for calculating the optimum number of iterations for the simulation. Thus it was decided to run all the simulations for 1500 iterations although the @RISK manual states that results and statistics for most models become stable after 300 to 500 Latin Hypercube iterations.

5.4 Results Obtained

Figure 5.12 shows the summary graph obtained for cumulative oil production from May 1989 to July 1993 using the normal distribution. As described earlier, each point on the graph represents the mean value of the probability distribution for that month's production. Each point also has an error bar. The tops of the error bars represent the 90% values for the distribution, while the bottoms represent the 10% values. Thus, at July 1993, the mean value of cumulative production is about 7 million barrels of oil whereas the 10% value is about 5 MMBO and the 90% value is about 8.5 MMBO. This illustrates how a small uncertainty in input parameters can lead to a large range of expected cumulative oil output.

Figure 5.13 shows the corresponding summary graph for the triangular case. In this case, the mean value of cumulative production at July 1993 is about 6 MMBO, whereas the 10% and 90% values are about 5 MMBO and 7 MMBO respectively.

Figure 5.14 shows the summary graph for the triangular (10/90) case. Here the mean value of cumulative production at July 1993 is again about 6 MMBO, whereas the 10% and 90% values are about 4 MMBO and 8 MMBO respectively. Thus, using this distribution has given about the same mean value as the triangular distribution, but has resulted in a greater spread of possible values.

Figure 5.15 shows the summary distribution of calculated oil production rates over the same period, using the normal distribution. Figures 5.16 and 5.17 show the corresponding results using the triangular and truncated triangular (10/90) distributions.

Figure 5.12 Summary Cumulative Oil Production—Normal

Figure 5.13 Summary Cumulative Oil Production—Triangular

Figure 5.14 Summary Cumulative Oil Production—Triangular (10/90)

Figure 5.15 Summary Production Rate Graph—Normal

Figure 5.16 Summary Production Rate Graph—Triangular

Figure 5.17 Summary Production Rate Graph—Triangular (10/90)

Figure 5.18 extends the calculations into the future and shows the summary graphs for forecasted cumulative oil production, using the normal distribution. Figures 5.19 and 5.20 show the corresponding forecast for the two triangular distributions. Figures 5.21 through 5.23 show the forecasted production rates using the three distributions.

In addition to the summary graphs, @RISK can also plot histograms of the probability distribution for any output cell. Figure 5.24 shows the probability histogram for the forecasted cumulative production from the field at steam breakthrough, using the normal distribution. The mean value is 11.044 MMBO.

Figures 5.25 and 5.26 show the histograms of cumulative production at steam breakthrough using the two triangular distributions.

From Figure 5.24 it can be seen that the output distribution is slightly skewed to the right. Figures 5.25 and 5.26 are not noticeably skewed. These results are as expected and explained later in Section 5.5 in connection with Eq. 5.5.

The expected value of final cumulative production for the regular triangular distribution is 10.035 MMBO. The corresponding value for the triangular (10/90) distribution is 9.742 MMBO.

The results obtained from the three different cases are compared in Table 5.3. The highest predicted value for cumulative production came from using the normal distributions of variables, the next highest from the triangular distributions and the lowest from using the truncated triangular (10/90) distributions. However, this result is as should be expected, in view of the ranges of parameters in Table 5.2. As we move from the normal to the

truncated triangular (10/90) distribution, the residual oil saturation keeps increasing in value while net pay keeps decreasing.

Figure 5.18 Summary Forecasted Cumulative Production—Normal

Figure 5.19 Summary Forecasted Cumulative Production—Triangular

Figure 5.20 Summary Forecasted Cumulative Production—Triangular (10/90)

Figure 5.21 Summary Forecasted Production Rates—Normal

Figure 5.22 Summary Forecasted Production Rates—Triangular

Figure 5.23 Summary Forecasted Production Rates—Triangular (10/90)

Figure 5.24 Cumulative Production Histogram at Steam Breakthrough—Normal

Figure 5.25 Cumulative Production Histogram at Breakthrough—Triangular

Figure 5.26 Cumulative Production Histogram at Breakthrough—Triangular (10/90)

Table 5.3 Expected Cumulative Production Using Different Distributions

<u>Distribution Used</u>	<u>Expected Cumulative Production (MMBO)</u>
Normal	11.044
Triangular	10.035
Triangular (10%/90%)	9.742

5.5 Statistical Analysis of the Results

The volumes of oil displaced at the end of each steam injection period were calculated using Eq. 5.2, which is

$$N_d = V_s \phi (S_{oi} - S_{or}) h_n / h_i \quad (5.2)$$

This can be simplified to

$$N_d = C \phi (S_{oi} - S_{or}) h_n \quad (5.3)$$

where $C = V_s / h_i$. Here, V_s is the steam zone volume at the end of each steam injection period and h_i is the gross formation thickness, both of which are assumed to be known values in this analysis.

Considering the case where the remaining parameters are considered to be normally distributed and assuming them to be independent random variables, then, using Eq. 4.14 we can write the following equation for the mean value of oil recovery,

$$\overline{N_d} = C \overline{\phi} (\overline{S_{oi}} - \overline{S_{or}}) \overline{h_n} \quad (5.4)$$

where the bars above the variables imply their mean values.

The mean value that was obtained for cumulative oil production, $\overline{N_d}$, in Section 5.4 after Monte-Carlo simulation was 11.044 MMBO. If we now substitute the steam zone volume and the assumed mean values of the normally distributed input variables ϕ , S_{oi} , S_{or} and h_n into Eq. 5.4, we obtain a value of 11.042 MMBO for the calculated mean cumulative oil production, $(\overline{N_d})_{\text{calc}}$. These two values are nearly equal. It may be noted one value was obtained after going through 1500 iterations of the Monte-Carlo simulation process while

the other was obtained by simply substituting values into Eq. 5.4. This implies that the input parameters actually are independent random variables. This also confirms that one can get an accurate estimate of the mean oil recovery for a steam flood by simply using Eq. 5.4. The usual calculations for steam zone volumes would, of course, have to be carried out.

We can also write an approximate equation which relates the standard deviation, σ_{N_d} , and mean value of the oil displacement, $\bar{N}_d$, to the standard deviations and mean values of the four input variables ϕ , S_{oi} , S_{or} and h_n . This is Eq. 5.5, which contains the first terms of an infinite series.

$$\left(\frac{\sigma_{N_d}}{\bar{N}_d} \right)^2 = \left(\frac{\sigma_\phi}{\bar{\phi}} \right)^2 + \left(\frac{\sigma_{h_n}}{\bar{h}_n} \right)^2 + \frac{\sigma_{S_{oi}}^2 + \sigma_{S_{or}}^2}{(\bar{S}_{oi} - \bar{S}_{or})^2} \quad (5.5)$$

The left- and right-hand sides of Eq. 5.5 were calculated. The left-hand side was calculated by substituting the values of mean cumulative oil production and its standard deviation, which were obtained from Monte-Carlo simulation. The right hand side is calculated by substituting the known mean values and standard deviations of the assumed normally distributed variables. The right hand side is thus calculated immediately to be 0.0412. The left hand side was calculated three times, after 500, 1000 and 1500 iterations respectively, of the simulation, to check whether the simulation had converged. The results of these calculations are shown in Table 5.4.

Two conclusions can be drawn from the calculations shown in Table 5.4. One is that Eq. 5.5 is valid when normal distributions are used for the random variables in the simulation. The other is that the agreement between the two sides of the equation improves

Table 5.4 Calculation of Standard Deviations for Gaussian Distribution

Eq. 5.5 Evaluation			
<u>Iterations</u>	<u>Left hand side</u>	<u>Right hand side</u>	<u>% Difference</u>
500	0.04287	0.04120	4.05
1000	0.04162	0.04120	1.02
1500	0.04154	0.04120	0.83

as the number of iterations of the Monte-Carlo simulation increases. The optimum number of iterations for which the simulation should be run can also be determined from this table. It is seen that while there is an appreciable difference between the results of 500 and 1000 iterations, the difference is much smaller between 1000 and 1500 iterations. Thus Eq. 5.5 also provides a sensitive method for checking the convergence of the simulation.

As mentioned earlier, Eq. 5.5 is only the first terms in an infinite series defining the difference in the variables. The net result of such an equation is that the output distribution tends to be log-normal rather than normally distributed. With this in mind we can now look at Figs. 5.24 through 5.26 in detail. As expected, Fig. 5.24 is slightly skewed to the right and therefore log-normally distributed. The amount of skew is not very large, and this is an important point. A rather large range of variables were used for the unknown reservoir parameters, but still, an almost Gaussian output was obtained. Figures 5.25 and 5.26 are not noticeably skewed, but this is due to the input values used. The skews in S_{Or} and h_n

were such that the output would tend to extend the results towards lower values of cumulative production. Thus they tended to balance the natural skew to the right caused by the approximation inherent in Eq. 5.5, and resulted in an approximately symmetrical output distribution.

A distributional analysis was also carried out for the other two probability distributions used—triangular and truncated triangular (10%/90%). It was decided to test these two equations for these non-Gaussian probability distributions. The results obtained on calculating mean values of recovery, Eq. 5.4, for these distributions are shown in Table 5.5. The simulation was carried out to 1500 iterations for both distributions. It can be seen that the equation also fits these two non-Gaussian distributions extremely well.

Table 5.5 Calculation of Recovery for non-Gaussian Distributions

Eq. 5.4 Evaluation			
Distribution	$\overline{N_d}$	$(\overline{N_d})_{\text{calc}}$	% difference
Triangular	10.035	10.035	0.00
Triangular (10%/90%)	9.742	9.745	0.03

Table 5.6 shows the distribution results obtained when Eq. 5.5 was calculated for the non-Gaussian distributions. As there is no closed form expression for calculating the standard deviation of these distributions, they were generated by using the statistical tables that @RISK creates for specified cells, after the simulation is over, as described in Section 5.1. Again, the equations fit the results extremely well.

The Monte-Carlo simulation process has shown how future oil production rates and cumulative production can be calculated as probability distributions. The probability of occurrence of any particular value of oil production can be calculated by using the distributions. The results also show that an extremely good estimate of mean oil production can be obtained by merely carrying out a calculation of Eq. 5.4 for the system, rather than going through the entire Monte-Carlo simulation process. Also, a very good estimate of the

Table 5.6 Calculation of Standard Deviations for non-Gaussian Distributions

Eq. 5.5 Evaluation			
<u>Distribution</u>	<u>Left hand</u> <u>side</u>	<u>Right hand</u> <u>side</u>	<u>%</u> <u>Difference</u>
Triangular	0.01728	0.01750	-1.26
Triangular (10%/90%)	0.06004	0.05681	+5.69

CHAPTER 5 MONTE-CARLO SIMULATION AND THE RAMEY METHOD

distribution of these results can be made by using Eq. 5.5. As seen in Figs. 5.25 and 5.26, approximately symmetric distributions were obtained, due to the fact that the negative skew in input parameters tended to offset the natural tendency for a positive skew due to the approximations in Eq. 5.5. This type of negative skew in input parameters would be quite normal in oil reservoir calculations. Therefore we might also expect this kind of result in other field calculations.

Chapter 6

Economic and Risk Analysis

The results of the previous chapters provided a risk-analyzed oil production rate schedule. Economic analysis is especially important in a steam flood project, since most such projects have high investment and operating costs and are low profit operations. Workover costs are usually high for the unconsolidated sand reservoirs studied in this work. Also, heavy crude fetches a lower price than lighter crudes. To evaluate these factors, the production rate schedule was translated into an economic result by means of a standard discounted cash flow analysis. However, since some of the input economic parameters were uncertain, they were treated as normal, or Gaussian, random variables for a risk analysis evaluation. The procedure followed is described in the following sections. The following analysis was carried out for the most likely case (Case 2) of the Wilmington steam flood described in Chapter 2.

6.1 Discounted Cash Flow Model

The first basic principle of finance is that a dollar received today is worth more than a dollar received in the future, because the dollar today can be invested to earn interest and thus will be worth more than a dollar received in the future. This principle is fundamental in analyzing the economic feasibility of any oilfield project, since a future oil production rate schedule must be translated into future cash flows, which in turn must be related to an investment decision in the present. Therefore a method is needed to convert a delayed payoff into a value today, a *present value* (PV). This is done by multiplying the delayed payoff by a *discount factor*, which is less than 1.0. If there are a series of delayed cash flows C_i at times t_i then their PV is given by

$$PV = \sum_{i=1}^n DF_i \times C_i \quad (6.1)$$

where DF_i is the discount factor at time t_i . The discount factor is given by

$$DF_i = \frac{1}{(1+r_i)^{t_i}} \quad (6.2)$$

In Eq. 6.2, r_i is the discount rate. It is often the rate of return that would be offered by other comparable investments (at time t_i). This rate of return is also called by various names; the *discount rate*, *hurdle rate* or *opportunity cost of capital*. The term opportunity cost arises because it is the return that is foregone by not investing in safe securities.

The net present value (NPV) is obtained by adding the initial cash flow C_0 for the project (usually a negative number, since it is a cash outflow) to the PV equation.

$$NPV = C_0 + PV \quad (6.3)$$

One important point to be noted about the NPV rule is that it is stated in terms of cash flows. Cash flows are, simply stated, just the difference between dollars received and dollars paid out. Cash flows must not be confused with accounting profits. They must, however, be estimated on an after-tax basis, as tax payments are real cash out-flows. Therefore, for most analyses, we can estimate cash flows using the general equation

$$CF = NOI - TAX \quad (6.4)$$

where CF is cash flow, NOI is net operating income and TAX is taxes paid. Here, NOI equals gross income minus expenses.

The main criterion to be used, when evaluating any project by the NPV method, is that the NPV must be greater than zero. This simply implies that the present value of any future payoffs from the project is greater than the initial and future (discounted) cash out-flows required for the project.

Another related number, which is sometimes used, is the internal rate of return (IRR). The IRR is simply the discount rate, r_i , which makes the NPV equal to zero. Thus the IRR is the rate of return at which the PV of future payoffs is equal to the initial cash outflow for the project. However, the IRR is less useful than the NPV because often a project may not have a unique IRR, and in some rare cases an IRR may not even exist.

6.2 Steam Flood Project Costs

Using Eq. 6.4, the net cash flow for a steam project, for any year, is

$$CF = (N_p \times P_o - ORR - FC - O\&M - INV) - TAX \quad (6.5)$$

where the terms within parentheses on the right-hand side are the terms that make up the *NOI*. In this part of the equation, N_p is the yearly oil production, P_o is the oil price and thus $N_p \times P_o$ is the gross income. The remaining terms within the parentheses are the expenses, which consist of *ORR*, the overriding royalty payment, *FC*, the generator fuel costs, *O&M*, the yearly operating and maintenance (including workover) costs and *INV* which is the annual capital investment required (if any). The last term, *TAX* consists of the combined federal, state, and local taxes. All these expenses, for the Wilmington project, are discussed further below.

Since detailed financial data for the Wilmington project were not available, standard costs for Southern California thermal projects were used. The actual Wilmington project uses a co-generator plant for steam generation. Therefore, since the plant sells electricity in addition to generating steam, it has different, and more profitable, economics than usual steam floods. In this analysis it is assumed that normal steam generators are used rather than a co-generator system.

The costs for a steam injection project can be divided into two main groups—costs related to the development of the project and costs related to operations.

Development costs are mainly capital costs and include expenditures for drilling of wells, installation of injection and production systems, steam generation and water-treatment equipment, and all other equipment and facilities required for operating the project. The development costs were assumed to be fixed, determinable costs. Data used in the analysis was taken from Ramage et al. [3] and Sarathi and Olsen [19].

Operating costs include fuel and water treatment costs, operating and maintenance costs for wells, and other associated costs related to fuel, labor and supplies. These were assumed to be normal random variables, having specified means and standard deviations. as discussed later.

The initial capital investment consists of the development costs, which were all calculated in 1991 US dollars, are listed below:

- The cost of drilling injection and production wells. This was taken as \$125,000 for a standard Wilmington type well. A total of 93 wells needed to be drilled to cover all the 7-spot patterns.
- The cost of installing steam generators. This worked out to be \$750,000 per generator (capacity 50 MMBTU/hr). A total of eight such generators would be needed to maintain the required steam injection rate for the field.
- Cost of installing steam lines. This was taken as \$11 per foot, for a total value of \$200,000.
- Facilities costs. These costs include items like the free water knockout plant (FWKO), heater treater, two automatic well testing (AWT) units, one lease area custody transfer (LACT) unit, and tank batteries.

It was estimated that the total initial capital investment required to set up a project similar to Wilmington would be approximately \$18 million (in 1991 US dollars). The breakdown of estimated costs is shown in Table 6.1.

Table 6.1 Breakdown of Initial Capital Investment for a 130 acre Steamflood Project

<u>S. No.</u>	<u>Description</u>	<u>Unit Cost (1991 \$'s)</u>	<u>Extended Cost (1991 \$'s)</u>
1.	Drilling of wells (93 nos.)	125, 000	11, 625, 000
2.	Steam generators (8 nos.)	750, 000	6, 000, 000
3.	Steam lines (18,000 ft. approx.)	11	200, 000
4.	Water treatment	250, 000	250, 000
5.	Free water knock out unit	30, 000	30, 000
6.	Well testing units (2 nos.)	35, 000	70, 000
7.	LACT unit (1 no.)	40, 000	40, 000
8.	Tank battery	75, 000	75, 000

The operating expenses can now be calculated. Equation 6.5 can be further expanded in the following manner, taking the combined federal, state, and local tax rate to be 50%:

$$\begin{aligned}
 CF &= NOI - Tax \\
 &= NOI - 0.5 (Taxable Income) \\
 &= NOI - 0.5 [NOI - Depreciation]
 \end{aligned}$$

which simplifies considerably, since we are assuming a 50% tax rate,

$$CF = 0.5 [NOI + Depreciation] \quad (6.6)$$

Now we can define net operating income as follows,

$$NOI = \text{Gross Revenues} - \text{Royalties} - FC - O\&M$$

or,

$$NOI = N_p (P_o) - 0.125(N_p) (P_o) - (\text{Steam Used}) (\text{Steam Cost}) - O\&M \quad (6.7)$$

taking the royalty rate to be 12.5%, and assuming no recurring yearly capital investment, other than the initial capital investment required to set up the project. Now, substituting Eq. 6.7 into Eq. 6.6, we get,

$$CF = 0.5 [0.875 (N_p) (P_o) - (\text{Steam Used}) (\text{Steam Cost}) - O\&M + Depreciation] \quad (6.8)$$

Equation 6.8 is the final expression used to determine the yearly cash flow. From this expression we can see clearly that three main cash items determine the yearly cash flow. These are the oil price (\$/bbl), the steam cost (\$/MMBTU) and the operating and maintenance cost (\$/yr). These were all assumed to be normal random variables with means and standard deviations as given in Table 6.2.

Table 6.2 Expectations and Standard Deviations of Economic Variables

Year	Oil Price		Steam Cost		Operating and Maintenance	
	\$/bbl	σ_o	\$/MMBTU	σ_s	E(O&M) \$/yr	$\sigma_{O\&M}$
1	15.86	1.586	0.65	0.03250	2,950,000.00	147,500.00
2	20.03	2.003	0.68	0.03413	3,097,500.00	154,875.00
3	16.50	1.650	0.72	0.03583	3,252,375.00	162,618.75
4	17.40	1.740	0.75	0.03762	3,414,993.75	170,749.69
5	18.27	1.827	0.79	0.03950	3,585,743.44	179,287.17
6	19.18	1.918	0.83	0.04148	3,765,030.61	188,251.53
7	20.14	2.014	0.87	0.04355	3,953,282.14	197,664.11
8	21.15	2.115	0.91	0.04573	4,150,946.25	207,547.31
9	22.21	2.221	0.96	0.04802	4,358,493.56	217,924.68

Future values for these three main costs were forecast in the manner discussed below:

The mean oil prices from Years 1 through 4 were taken as the average US wellhead prices, for that type of oil, for that year. From Years 5 through 9 the mean oil price was escalated by 5% of the previous year's mean price. For all years, the standard deviation of oil prices was taken as 10% of the mean oil price for that year.

The mean steam cost for Year 1 was taken as 65 cents/MMBTU. For succeeding years, the mean cost was escalated by 5% of the previous year's mean steam cost. For all years, the standard deviation of steam cost was taken as 5% of the mean steam cost for that year.

Operating and maintenance costs consisted mainly of labor, workover and water treatment costs. For Year 1 these were taken to be \$2.95 million, which consisted of \$2,000,000 as labor costs, \$450,000 for water treatment (@\$0.04/bbl treated water) and \$500,000 for well workover. For succeeding years, this mean value was escalated by 5% per year. Standard deviation was taken as 5% of the mean cost.

Equation 6.8 was then used to calculate the cash flow for each year. The cash flows thus obtained were discounted at various rates using Eq. 6.1 to get present values and using Eq. 6.3 to calculate the NPV for the project. The final spreadsheet for the economic calculation with risk analysis is shown in Table 6.2.

It can be seen from Table 6.3 that the project has a high NPV of \$2.7 million even at a discount rate of 20%. The IRR is calculated to be 26.3%.

6.3 Statistical Analysis of Results

Since Eq. 6.8 is a linear function of random variables, we can put it into expected value form using Eq. 4.11. This leads to Eq. 6.9.

$$E(CF) = 0.5[0.875(N_p)E(P_o) - (\text{Steam Used})E(\text{Steam Cost}) - E(O\&M) + \text{Depreciation}] \quad (6.9)$$

Also, assuming the random variables to be independent and using Eq. 4.13, we get Eq. 6.10, for the variance of cash flow

$$\begin{aligned} \text{var}(CF) = & (0.5 \times 0.875 \times N_p)^2 \text{var}(P_o) \\ & - (0.5 \times \text{Steam Used})^2 \text{var}(\text{Steam Cost}) - (0.5)^2 \text{var}(O\&M \text{ Cost}) \end{aligned} \quad (6.10)$$

Table 6.3 Economic Analysis Spreadsheet

Year	Production (Bbl/yr)	Price \$/Bbl	Gross Revenue	Royalties (12.5%)	Steam Cost	Fuel Used MMBTU/yr	Fuel Costs (\$/yr)
0							
1	8.43E+05	15.86	1.34E+07	1.67E+06	0.65	3.65E+06	2.37E+06
2	9.85E+05	20.03	1.97E+07	2.47E+06	0.68	3.65E+06	2.49E+06
3	1.35E+06	16.50	2.23E+07	2.79E+06	0.72	3.65E+06	2.62E+06
4	1.31E+06	17.40	2.27E+07	2.84E+06	0.75	3.65E+06	2.75E+06
5	8.55E+05	18.27	1.56E+07	1.95E+06	0.79	3.65E+06	2.88E+06
6	7.05E+05	19.18	1.35E+07	1.69E+06	0.83	3.65E+06	3.03E+06
7	6.33E+05	20.14	1.28E+07	1.59E+06	0.87	3.65E+06	3.18E+06
8	5.67E+05	21.15	1.20E+07	1.50E+06	0.91	3.65E+06	3.34E+06
9	4.67E+05	22.21	1.04E+07	1.30E+06	0.96	3.65E+06	3.51E+06

Year	Operating Costs (\$/yr)	NOI (\$/yr)	Investment (\$/yr)	Depreciation (\$/yr)	Taxable Income (\$/yr)	Combined Tax (\$/yr)	Net Cashflow (\$/yr)
0			-1.80E+07				-1.80E+07
1	2.95E+06	6.37E+06		2.00E+06	4.37E+06	2.19E+06	4.19E+06
2	3.10E+06	1.17E+07		2.00E+06	9.67E+06	4.83E+06	6.83E+06
3	3.25E+06	1.37E+07		2.00E+06	1.17E+07	5.84E+06	7.84E+06
4	3.41E+06	1.37E+07		2.00E+06	1.17E+07	5.86E+06	7.86E+06
5	3.59E+06	7.19E+06		2.00E+06	5.19E+06	2.60E+06	4.60E+06
6	3.77E+06	5.04E+06		2.00E+06	3.04E+06	1.52E+06	3.52E+06
7	3.95E+06	4.03E+06		2.00E+06	2.03E+06	1.01E+06	3.01E+06
8	4.15E+06	3.01E+06		2.00E+06	1.01E+06	5.06E+05	2.51E+06
9	4.36E+06	1.22E+06		2.00E+06	0.00E+00	0.00E+00	1.22E+06
IRR (%)	NPV (@10%)(%)	NPV (@12%)(%)	NPV (@14%)(%)	NPV (@16%)(%)	NPV (@18%)(%)	NPV (@20%)(%)	
26.3	9.80E+06	8.00E+06	6.42E+06	5.03E+06	3.80E+06	2.70E+06	

In both Eq. 6.9 and Eq. 6.10, the right hand sides can be evaluated immediately by putting in values from Table 6.2. The left hand sides were obtained from the Monte-Carlo simulations and a comparison of the results obtained is shown in Table 6.4. There is an excellent fit, as can be seen. This is important, for it again shows that simple equations like Eqs. 6.9 and 6.10 can be used to give approximate results quickly without resorting to Monte-Carlo simulation.

Figures 6.1 through 6.3 show the different histograms obtained for oil price, steam cost and O&M costs in Year 9. They all show the typical bell shape of the normal distribution which they are assumed to follow.

Table 6.4 Statistical Comparisons of Results

E(CF) LHS Eq. 6.9	E(CF) RHS Eq. 6.9	% diff	Var (CF) LHS Eq. 6.10	Var (CF) RHS Eq. 6.10	% diff
4,186,197.80	4,185,597.48	-0.014342	3.51497E+11	3.50813E+11	-0.194853
6,833,399.03	6,833,317.06	-0.001200	7.55084E+11	7.54235E+11	-0.112515
7,837,960.26	7,838,253.63	0.003743	9.62057E+11	9.65862E+11	0.393943
7,861,229.45	7,861,393.39	0.002085	1.00699E+12	1.00046E+12	-0.652928
4,595,328.44	4,595,415.26	0.001889	4.83064E+11	4.79747E+11	-0.691326
3,518,186.85	3,518,132.03	-0.001558	3.61318E+11	3.64419E+11	0.850951
3,013,057.35	3,013,016.46	-0.001357	3.28031E+11	3.27377E+11	-0.199784
2,506,435.87	2,506,297.65	-0.005515	2.99761E+11	2.93458E+11	-2.147811
1,215,784.68	1,215,247.98	-0.044164	8.85359E+11	9.02492E+11	1.898343

Figure 6.1 Oil Price Histogram (Year 9)

Figure 6.2 Steam Cost Histogram (Year 9)

Figure 6.3 O&M Cost Histogram (Year 9)

Figure 6.4 is a summary graph of future oil prices. They have been taken at actual values for Years 1 through 4 and then escalate at 5% per year for succeeding years. As earlier, the middle of each vertical bar represents the mean value for that year and the tops and bottoms the 10% and 90% values.

Figures 6.5 and 6.6 show a summary of the future steam and O&M costs. They are assumed to escalate by 5% every year.

Figure 6.7 is a histogram of the cash flow in Year 9. As expected, the shape of the distribution is log-normal, i.e. approximately normal, with a small positive skew. Figure 6.8 is a summary graph of the cash flows from Year 1 to Year 9.

Figure 6.4 Summary Graph for Future Oil Prices

Figure 6.5 Summary Graph for Future Steam Costs

Figure 6.6 Summary Graph for Future O&M Costs

Figure 6.7 Cash Flow Histogram (Year 9)

Figure 6.8 Summary Graph for Cash Flow

Figure 6.9 is a histogram of the NPV at a 20% discount rate. The distribution shape is, again, approximately normal with perhaps a small positive skew.

Figure 6.10 is a summary graph for NPV and Fig. 6.11 is a histogram for IRR. The histogram is again, approximately normal in shape.

Figure 6.9 Histogram for NPV (20% Discount Rate)

Figure 6.10 Summary Graph for NPV

Figure 6.11 Histogram for IRR

Chapter 7

Summary and Conclusions

This study revises and extends a continuing SUPRI-A research effort in the technoeconomic and risk analysis of thermal recovery projects. Previous efforts had focused on development of general engineering-economic models comparing different methods of thermal recovery. In this study, a specific engineering and risk analysis was done on the Wilmington steam flood project carried out by Union Pacific Resources Company (UPRC) in Los Angeles County, California.

The procedures adopted for the analysis and the results were as follows:

1. A heat balance for the steam flood project was calculated by determining heat losses due to the different mechanisms operating in the field. An important heat loss mechanism considered was that due to produced fluids from the wellbores. It was calculated that, four years after the steam flood was initiated, almost half (47%) of the total heat lost from the steam flood was due to produced fluids.
2. Ramey's generalization of the Marx & Langenheim method was used to calculate reservoir heat losses and oil production rates for the steam flood. This was used to match the production history of the project, and thus determine a capture efficiency of 60% for the steam flood. Future production rates from the field were also predicted by this method. It was calculated that up to 43% of the total oil in place (at

steam flood initiation) would be recovered by the steam flood. Another analysis method due to Gadjica et al. was also tried, and suggestions were made for further work to make this method more applicable.

3. Since most of the reservoir parameters used in the steam flood analysis cannot be exactly determined, it was decided to treat them as stochastic variables obeying statistical distribution functions. Monte-Carlo simulations were then carried out on the Wilmington steam flood so that the predicted oil production at any time could be determined as a probability distribution, rather than a fixed deterministic value. Probability distributions used for the input variables were the Gaussian (Normal), as well as two types of triangular distributions. Results obtained indicate that it may be possible to obtain a good estimate of the mean oil recovery and standard deviation of that recovery by simple calculations involving the expected values and standard deviations of the input parameters, without going through the complete Monte-Carlo simulation process.
4. Finally, an economic analysis and Monte-Carlo simulation of the steam flood future was carried out, by treating some of the input parameters (oil price, fuel cost, and operating cost as normal random variables. Thus, future oil prices, cashflows, net present value (NPV) and internal rate of return (IRR) for the project were calculated as probability distributions, rather than fixed deterministic values. The project was estimated to have an IRR of 26% and an NPV of \$2.7 million at a discount rate of 20%.

By following the above procedure a complete technical, economic and risk analysis was carried out for an operating steam flood.

Further improvements and advancements can be made to the above procedure.

- Other analytical steam flood methods can be tried to see if a better match is obtained with the production history.

- In this analysis, the Monte-Carlo simulations for the steam flood calculations and the economic calculations were carried out separately. The two calculations can be simulated simultaneously, to see if more useful results can be obtained.
- Other different types of probability distributions can be tried out for the input parameters to see if they give better results, and to know whether they obey Eqs. 5.4 and 5.5, the simple way to approximate the effect of parameter variations.
- The method used here can be used to analyze other steam flood projects to check its applicability.

Bibliography

- [1] Williams, R. L., Brown, S. L and Ramey, H. J., Jr.: *A Method to Evaluate the Economics of Thermal Recovery Projects*, Engineer's Thesis, Stanford U., Stanford, CA (June 1980).
- [2] Ramage, W. E. and Ramey, H. J., Jr.: *The Comparative Economics of Thermal Recovery Projects*, Engineer's Thesis, Stanford U., Stanford, CA. (1986).
- [3] Joshi, S., Wood, M.C., Castanier, L.M., and Brigham, W.E.: *Steamflooding a Waterflooded Reservoir—Performance Evaluation and Prediction*, IEA Conference, Japan (Oct. 15-18, 1995); also Sixth UNITAR Conference, Houston (Feb. 12-17, 1995); also paper SPE 29669 presented at the 1995 Western Regional Meeting, Bakersfield (Mar. 8-10, 1995).
- [4] Lim, F. H., Saner, W. B., Stillwell, W. H. and Patton, J. T.: *Steamflood Pilot Test in Waterflooded, 2500 ft. Tar Zone Reservoir, Fault Block II Unit, Wilmington Field, California*, paper SPE 26615 (1993).
- [5] Prats, M.: *Thermal Recovery*, SPE Monograph Volume 7 (1982).
- [6] Gambill, W.R.: *You Can Predict Heat Capacities*, Chem. Engr. (June 1957).
- [7] Ramey, H. J., Jr.: Discussion of Marx and Langenheim Paper, Petroleum Transactions, Vol. 216, p. 364 (1959).

BIBLIOGRAPHY

- [8] Ramey, H. J., Jr.: Correction of equation given in [7], JPT, p. 658 (July 1961).
- [9] Myhill, N. A. and Stegemeier, G. L.: *Steam Drive Correlation and Prediction*, JPT, p. 1227-1232 (Nov. 1972).
- [10] Satter, A. and Parrish, D. R.: *A Two Dimensional Analysis of Reservoir Heating by Steam Injection*, SPEJ, Vol. 11, p. 185-197 (1971).
- [11] Gadjica, R. J., Brigham, W. E. and Aziz, K. A.: *A Semi-Analytical Model for Steam Drive Prediction*.
- [12] Murphy, R. J.: *An Engineering and Economic Analysis of the Soldotna Creek Unit, Swanson River Oil Field, Alaska*, Engineer's Thesis, Stanford U. Stanford, CA (June 1977).
- [13] Davidson, L. B. and Cooper, D. O.: *A Simple Way of Developing a Probability Distribution of Present Values*, JPT, p. 1069-1078 (Sept. 1976).
- [14] Mendenhall, W. and Sincich, C.: *Statistics for the Engineering and the Sciences (3rd Ed.)*, Dellen Publishing Company (a division of Macmillan Publishing Company), San Francisco, CA (1992).
- [15] @RISK User's Guide (Release 1.11), Risk Analysis and Simulation Add-In for Microsoft Excel, Palisade Corporation, Newfield, NY, USA (July 1993).
- [16] Rubinstein, R. Y.: *Simulation and the Monte-Carlo Method*, John Wiley and Sons, New York, NY (1981).
- [17] Newendorp, P. D.: *Decision Analysis for Petroleum Exploration*, Petroleum Publishing Company, Tulsa, OK (1975).
- [18] Megill, R. E.: *An Introduction to Risk Analysis*, PenWell Books, Tulsa, OK (1985).

BIBLIOGRAPHY

- [19] Sarathi, P. S. and Olsen, D. K.: *Practical Aspects of Steam Injection Processes—A Handbook for Independent Operators*, National Institute for Petroleum and Energy Research (NIPER), Bartlesville, OK (October 1992).

Appendix

A.1 FORTRAN source code for Gadjica method

The main FORTRAN source file is called *tot.f*. Other associated files are a common file called *dl.com* and the main data file, called *main.dat*.. These are attached below.

A.1.1 File *tot.f*:

c Modification Comment

c Program originally written for PC's. Modified and ported onto UNIX machine (pangea)
c by Sameer Joshi -----June 1994

c*****

c

TWO-DIMENSIONAL STEAM INJECTION MODEL

c

BY RONALD J. GAJDICA

c

MAIN PROGRAM

c

c*****

program main
implicit real*8 (a-h,k-z)
include 'dl.com'

c-----

c

OPEN FILES

c-----

call open

c-----

c

READ DATA

c-----

call read(xangle)

c-----

c

INITIALIZE VARIABLES

c-----

pi=3.14159
ifbt=0
rcqo=0.
rcqw=0.
cqoold=0.
cqwold=0.

APPENDIX

```

cqhsi=0.
cqlsi=0.
timold=0.
p0=300.
pb1=pres
tb2=tem
porb2=por
deng1=1.0
denw2=denrw
deno2=denro
sob1=1.-swd(2)
sob2=soi
sob3=soi
swb1=swd(2)
swb2=swi
swb3=swi
sgb2=0.
sgb3=0.
krb1=1.
krw1=1.
kro2=1.
krw2=1.
inum=0
nj=0.
lv=1000.

```

```

c-----
c      CALCULATE INITIAL OIL & WATER IN PLACE
c-----

```

```

call dens(psurf,tsurf,prsr,temr,cw,cw2,cw3,denrw,denwst)
call dens(psurf,tsurf,prsr,temr,co,co2,co3,denro,denost)
pinit=(pres+presb)/2.
tinit=(tem+temb)/2.
call dens(pinit,tinit,prsr,temr,co,co2,co3,denro,denoi)
dp=ht/2.*denoi/144.*(1.0-s**2)**.5
pinit=pinit+dp
call dens(pinit,tinit,prsr,temr,co,co2,co3,denro,denoi)
call dens(pinit,tinit,prsr,temr,cw,cw2,cw3,denrw,denwi)
call poros(pinit,tinit,porp,temr,cf,cf3,por,pori)
boi=denost/denoi
bwi=denwst/denwi
ooip=leng*wid*ht*pori*soi/(5.6146*boi)
owip=leng*wid*ht*pori*swi/(5.6146*bwi)
write(2,*) 'owip (STB)='owip
write(2,*) 'ooip (STB)='ooip

```

```

c-----c
c      CALL FRACTIONAL FLOW CURVES
c-----c
call frac(m)

```

c

APPENDIX

```

call fracg(vgrs,vgrw)
pv=leng*wid*ht*pori
pvipd=qwmax*5.6146/pv
call steadj(s,m,pv,pvipd,stefac)
c-----c
c           Water Sweep Efficiency           c
c-----c
c=cos(xangle/57.29578)
tang=s/c
rat=127758*qwmax*uo2/(akx*wid*ht*(denwi-deno))
ratio=(rat/c + tang)*leng/ht
call ronnie(ratio,m,afac,bfac)
b=0.0
c=0.0
if(m.le.1.0) then
  b=0.01
else
  b=0.01+0.02*(m-1.0)/.441
  if(m.gt.1.441) b=0.03+0.04*(m-1.441)/1.808
  if(pvipd.lt.0.0001872) c=(0.0001872-pvipd)/0.0000936*0.01
  if(b.gt.0.07) b=0.07
  if(c.gt.0.01) c=0.01
endif
d=1.5+1.5*(3.249-m)/1.808
if(d.lt.1.5) d=1.5
if(d.gt.2.8) d=2.8
60 swb2 = swbft - b - c
sob2=1.-swb2
c-----c
c           TIME LOOP           c
c-----c
do 70 icnt = 0,9000
inum = inum + 1
if(icnt.eq.0) tim = 0.001
if(icnt.ge.1) tim = deltim*icnt
pviold=pvi
pvi = (5.6146*qwmax*tim + volste)/pv
lsteol=lste
p0old=p0ol
p0ol=p0
if(icnt.ge.3) p0=p0ol+(p0ol-p0old)
qste=0.3*qwmax*denwst/deng1
pb2=pres
p4old=0.
call fracg(vgrs,vgrw)
iter1=0
iter2=0
p0min=0

```


APPENDIX

```

p0maxx=p0max
x=cw*deltim/(qual*lv)
hf=1./(1.+nj)
ev=afac+hf*(bfac-afac)
pvibt=ev*(swbfbt-swi)
10 iter2=iter2+1
20 iter1=iter1+1
  call yort(volste,stefac,nj)
  call length(lwt,vgrs,volste,pvibt,frahot)
  call inj
  call ste(qste)
  call wat(lwt,ev,frahot)
  call oil(lwt,ev)
  call pro
  b1=abs(p4-p4old)
  if(b1.lt..0001) goto 30
  if(iter1.eq.6) then
    p4=(p4+p4old)/2.
    goto 30
  endif
  p4old=p4
  goto 20
30 continue
  b1=(p4-delp5)-p5-dp
  if(b1.lt.0) then
    if(p0.gt.p0min) p0min=p0
    p0=p0-b1+.000002
    if(p0.lt.p0min) p0=(p0min+p0maxx)/2.
    if(iter2.ge.10) then
      if(p0maxx.eq.p0max) p0=p0min
      if(p0maxx.ne.p0max) p0=(p0min+p0maxx)/2.
    endif
  else
    if(p0.lt.p0maxx) p0maxx=p0
    pold=p0
    p0=p5+delp1+dpot2+dpot3+dpot4+delp5+dp
    if((p0.gt.p0maxx).and.(p0min.gt.0)) p0=(p0min+p0maxx)/2.
    if((iter2.ge.10).and.(p0min.gt.0)) p0=(p0min+p0maxx)/2.
    b1=abs(pold-p0)
    if(b1.lt..00001) goto 40
  endif
  iter1=0
  if(iter2.eq.10) goto 40
  goto 10
40 ltot=ls(1)+lw(1)
  if((ifbt.eq.0).and.(ltot.ge.leng)) then
    ifbt=1
    lstebt=lsteol+(lste-lsteol)*(pvibt-pviold)/(pvi-pviold)

```

APPENDIX

```

      inum=1
    endif
    if(ls(1).ge.(leng-0.1)) then
      if(ifbt.ne.21) inum=1
      ifbt=21
    endif
    write(6,50) tim
50 format(t2,'time=',f8.2)
    if(p0.gt.p0max) write(6,*) 'Pressure exceeds constraint'
c
c  CALCULATE WATER SATURATION
c
    if(ifbt.eq.1) then
      if(m.le.1.0) then
        c=(lste-lstebt)/(leng-lstebt)*0.7
        swb2 = swbfbt*(1-c) + swd(2)*c
      else
        swb2 = swb2 + d*(swd(2)-swbfbt)*deltim*5.6146*qwmax/pv
      endif
    elseif(ifbt.eq.21) then
      swb2 = swb2 + (swd(2)-swb2)*(2.*deltim*5.6146*qwmax/pv)
    endif
    if(swb2.gt.swd(2)) swb2=swd(2)
    sob2 = 1.0 - swb2
    call go(roilol,rwatol,inum,m)
    ii=mod(icnt,10)
    if(ii.eq.0) call out
c   if(i.eq.1) call out
    lwtold=lwt
    if(tim.ge.timmax) goto 80
70 continue
80 jj=1
    end
c*****
c
c          SUBROUTINE OPEN
c
c          OPENS FILES
c
c*****
      subroutine open
      implicit real*8(a-h,k-z)
      open(unit=1,file='main.dat',status='old')
      open(unit=2,file='res.out')
      open(unit=3,file='go.out')
      open(unit=4,file='qw.out')
      open(unit=7,file='cgo.out')
      open(unit=9,file='cqw.out')

```

APPENDIX

```

rewind(1)
rewind(2)
rewind(3)
rewind(4)
rewind(7)
rewind(9)
write(3,*) '1 TIME OIL RATE'
write(4,*) '1 TIME WATER RATE'
write(7,*) '1 CUMULATIVE OIL'
write(9,*) '1 CUMULATIVE WATER'
return
end
c*****
c
c          READ IN DATA
c
c*****
c      subroutine read(xangle)
c      implicit real*8 (a-h,k-z)
c      include 'dl.com'
c      write(2,*) ' TWO-DIMENSIONAL SEMI-ANALYTICAL MODEL RESULTS'
c      write(2,*) ' -----'
c-----
c
c          GENERAL CONTROL
c
c-----
c      do 10 i=1,9
c          read(1,*) buf1
c10 continue
c          read(1,*) deltim
c          read(1,*) timmax
c          write(2,*) 'GENERAL CONTROL'
c          write(2,*) '-----'
c          write(2,20) deltim
c20 format('DELTIM times step size',t62,f11.6)
c          write(2,30) timmax
c30 format('TIMMAX maximum time',t62,f11.6)
c-----
c
c          RESERVOIR DESCRIPTION
c
c-----
c      do 40 i=1,5
c          read(1,*) buf1
c40 continue
c          read(1,*) xangle
c          read(1,*) leng

```

APPENDIX

```

read(1,*) wid
read(1,*) igbk
ht=0.
do 50 i=1,igbk
  read(1,*) blsizk(i)
  ht=ht+blsizk(i)
50 continue
read(1,*) blsizi
read(1,*) por
sum=0
do 60 i=1,igbk
  read(1,*) kx(i)
  sum=sum+kx(i)*blsizk(i)
60 continue
akx=sum/ht
read(1,*) akz
s = sin(xangle/57.29578)
write(2,*) ''
write(2,*) 'RESERVOIR DESCRIPTION'
write(2,*) '-----'
write(2,70) xangle
70 format('XANGLE formation dip, degrees',t62,f11.6)
write(2,80) leng
80 format('LENG  reservoir length, ft',t62,f11.6)
write(2,90) wid
90 format('WID  reservoir width, ft',t62,f11.6)
write(2,100) igbk
100 format('IGBK  number of layers in system',t62,i11)
do 110 i=1,igbk
  write(2,120) blsizk(i)
110 continue
120 format('BLSIZK size of block in k-direction, ft',t62,f11.6)
write(2,130) blsizi
130 format('BLSIZI size of block in i-direction, ft',t62,f11.6)
write(2,140) por
140 format('POR  porosity, fraction',t62,f11.6)
do 150 i=1,igbk
  write(2,160) kx(i)
150 continue
160 format('KX  permeability in x-direction, md',t57,f16.6)
write(2,170) akx
170 format('AKX  average x-direction permeability, md',t57,f16.6)
write(2,180) akz
180 format('AKZ  permeability in z-direction, md',t57,f16.6)
c-----
c
c      INITIAL CONDITIONS
c

```

APPENDIX

```

c-----
  do 190 i=1,5
    read(1,*) buf1
190 continue
    read(1,*) pres
    read(1,*) presb
    read(1,*) swi
    read(1,*) soi
    read(1,*) tem
    read(1,*) temb
    write(2,*) ''
    write(2,*) 'INITIAL CONDITIONS'
    write(2,*) '-----'
    write(2,200) pres
200 format('PRES  pressure at top of reservoir, psia',t62,f11.6)
    write(2,210) presb
210 format('PRESB  pressure at bottom of reservoir, psia',t62,f11.6)
    write(2,220) swi
220 format('SWI   initial water saturation, fraction',t62,f11.6)
    write(2,230) soi
230 format('SOI   initial oil saturation, fraction',t62,f11.6)
    write(2,240) tem
240 format('TEM   temperature at top of reservoir, F',t62,f11.6)
    write(2,250) temb
250 format('TEMB  temperature at bottom of reservoir, F',t62,f11.6)

```

```

c-----
c
c           PVT DATA
c
c-----

```

```

  do 260 i=1,5
    read(1,*) buf1
260 continue
    read(1,*) prorp
    read(1,*) prsr
    read(1,*) psurf
    read(1,*) temr
    read(1,*) tsurf
    read(1,*) denrw
    read(1,*) cw
    read(1,*) cw2
    read(1,*) cw3
    read(1,*) denro
    read(1,*) co
    read(1,*) co2
    read(1,*) co3
    read(1,*) avg
    read(1,*) bvg

```

APPENDIX

```

write(2,*) ''
write(2,*) 'PVT DATA'
write(2,*) '-----'
write(2,270) prorp
270 format('PRORP reference pressure for porosity, psia',t62,f11.6)
write(2,280) prsr
280 format('PRSR reference pressure for densities, psia',t62,f11.6)
write(2,290) psurf
290 format('PSURF surface pressure, psia',t62,f11.6)
write(2,300) temr
300 format('TEMR reference temperature for porosity and density, deg
& F',t62,f11.6)
write(2,310) tsurf
310 format('TSURF surface temperature, deg F',t62,f11.6)
write(2,320) denrw
320 format('DENRW reference water density, lbm/cu.ft',t62,f11.6)
write(2,330) cw
330 format('CW water compressibility, 1/psi',t62,f11.6)
write(2,340) cw2
340 format('CW2 water thermal expansion coefficient one, 1/F',
&t62,f11.6)
write(2,350) cw3
350 format('CW3 water thermal expansion coefficient two, 1/F',
&t62,f11.6)
write(2,360) denro
360 format('DENRO reference oil density, lbm/cu.ft',t62,f11.6)
write(2,370) co
370 format('CO oil compressibility, 1/psi',t62,f11.6)
write(2,380) co2
380 format('CO2 oil thermal expansion coefficient one, 1/F',
&t62,f11.6)
write(2,390) co3
390 format('CO3 oil thermal expansion coefficient two, 1/F',
&t62,f11.6)
write(2,400) avg
400 format('AVG steam viscosity coefficient',t62,f11.6)
write(2,410) bvg
410 format('BVG steam viscosity exponent',t62,f11.6)
write(2,*) ''

```

```

c-----
c
c      OIL & WATER VISCOSITY VS TEMPERATURE
c
c-----

```

```

read(1,*) iv
do 420 i=1,8
  read(1,*) buf1
420 continue

```

APPENDIX

```
write(2,*) 'OIL AND WATER VISCOSITY DATA'
write(2,*) '-----'
write(2,430) iv
430 format(TV    number of viscosity entries',t62,i3)
write(2,*) 'TEMP(F)  UW(CP)  UO(CP)'
do 440 i=1,iv
  read(1,*) vist(i),visw(i),viso(i)
  write(2,450) vist(i),visw(i),viso(i)
440 continue
450 format(t3,f5.1,t12,f8.3,t24,f8.3)
c-----
c
c      OIL-WATER RELATIVE PERMEABILITY
c
c-----
  do 460 i=1,9
    read(1,*) buf1
460 continue
    read(1,*) swd(1),krwd(1),krowd(1)
    read(1,*) swd(2),krwd(2),krowd(2)
    read(1,*) noil
    read(1,*) nwat
    write(2,*) ''
    write(2,*) 'OIL-WATER RELATIVE PERMEABILITY DATA'
    write(2,*) '-----'
    write(2,470) swd(1)
470 format('SWD(1) irreducible water saturation, fraction',t62,f11.6)
    write(2,480) swd(2)
480 format('SWD(2) maximum water saturation, fraction',t62,f11.6)
    write(2,490) krwd(1)
490 format('KRWD(1) water relative perm at SWD(1)',t62,f11.6)
    write(2,500) krwd(2)
500 format('KRWD(2) water relative perm at SWD(2)',t62,f11.6)
    write(2,510) krowd(1)
510 format('KROWD(1) oil relative perm at SWD(1)',t62,f11.6)
    write(2,520) krowd(2)
520 format('KROWD(2) oil relative perm at SWD(2)',t62,f11.6)
    write(2,530) noil
530 format('NOIL  Corey exponent for oil',t62,f11.6)
    write(2,540) nwat
540 format('NWAT  Corey exponent for water',t62,f11.6)
    write(2,*) ''
c-----
c
c      LIQUID-GAS RELATIVE PERMEABILITY
c
c-----
  do 550 i=1,9
```

APPENDIX

```

    read(1,*) buf1
550 continue
    read(1,*) sld(1),krgd(1),krogd(1)
    read(1,*) sld(2),krgd(2),krogd(2)
    read(1,*) ngas
    read(1,*) nliq
    write(2,*) 'GAS-LIQUID RELATIVE PERMEABILITY DATA'
    write(2,*) '-----'
    write(2,560) sld(1)
560 format('SLD(1) irreducible liquid saturation, fraction',t62,f11.6)
    write(2,570) sld(2)
570 format('SLD(2) maximum liquid saturation, fraction',t62,f11.6)
    write(2,580) krgd(1)
580 format('KRGD(1) gas relative perm at SLD(1)',t62,f11.6)
    write(2,590) krgd(2)
590 format('KRGD(2) gas relative perm at SLD(2)',t62,f11.6)
    write(2,600) krogd(1)
600 format('KROGD(1) liquid relative perm at SLD(1)',t62,f11.6)
    write(2,610) krogd(2)
610 format('KROGD(2) liquid relative perm at SLD(2)',t62,f11.6)
    write(2,620) ngas
620 format('NGAS Corey exponent for gas',t62,f11.6)
    write(2,630) nliq
630 format('NLIQ Corey exponent for liquid',t62,f11.6)
c-----
c
c          THERMAL DATA
c
c-----
    do 640 i=1,5
        read(1,*) buf1
640 continue
        read(1,*) cf
        read(1,*) cf3
        read(1,*) denr
        read(1,*) shr
        read(1,*) lamob
        read(1,*) alfob
        write(2,*) ''
        write(2,*) 'THERMAL DATA'
        write(2,*) '-----'
        write(2,650) cf
650 format('CF    formation compressibility, 1/psi',t62,f11.6)
        write(2,660) cf3
660 format('CF3    formation thermal expansion coefficient, 1/F',
            &t62,f11.6)
        write(2,670) denr
670 format('DENR    density of reservoir rock, lbm/cu.ft',t62,f11.6)

```


APPENDIX

```

write(2,680) shr
680 format('SHR  specific heat of reservoir rock, btu/lbm-F',
&t62,f11.6)
write(2,690) lamob
690 format('LAMOB  thermal conductivity of overburden, btu/ft-day-F',
&t62,f11.6)
write(2,700) alfob
700 format('ALFOB  thermal diffusivity of overburden, sq.ft/day',
&t62,f11.6)

```

```

c-----
c
c          INJECTION WELL
c
c-----

```

```

do 710 i=1,5
  read(1,*) buf1
710 continue
  read(1,*) p0max
  read(1,*) qwmax
  read(1,*) qual
  read(1,*) tinjw
  read(1,*) wi
  write(2,*) ''
  write(2,*) 'INJECTION WELL'
  write(2,*) '-----'
  write(2,720) p0max
720 format('P0MAX  maximum pressure at injection well, psia',
&t62,f11.6)
  write(2,730) qwmax
730 format('QWMAX  maximum water rate at injection well, bbl/day',
&t62,f11.6)
  write(2,740) qual
740 format('QUAL  steam quality at injection sandface, fraction',
&t62,f11.6)
  write(2,750) tinjw
750 format('TINJW  temperature of injected fluid at sandface, deg F',
&t62,f11.6)
  write(2,760) wi
760 format('WI  injectivity index, bbl/day/psi',t62,f11.5)

```

```

c-----
c
c          PRODUCTION WELL
c
c-----

```

```

do 770 i=1,5
  read(1,*) buf1
770 continue
  read(1,*) qtmax

```

APPENDIX

```

      read(1,*) p5
      read(1,*) rw
      read(1,*) cc
      read(1,*) ss
      write(2,*) ''
      write(2,*) 'PRODUCTION WELL'
      write(2,*) '-----'
      write(2,780) qtmx
780 format('QTMAX maximum liquid rate at producer, bbl/day',
      &t62,f11.6)
      write(2,790) p5
790 format('P5 constant fbhp at producer, psia',t62,f11.6)
      write(2,800) rw
800 format('RW wellbore radius of producer, ft',t62,f11.6)
      write(2,810) cc
810 format('CC shape factor at producer',t62,f11.6)
      write(2,820) ss
820 format('SS skin factor at producer',t62,f11.6,/)
      return
      end
c*****
c
c          SUBROUTINE FRAC
c
c          FRACTIONAL FLOW CURVE FOR WATER
c
c OUT: swf - water saturation at front
c      swbf - average water saturation behind front
c      fwbt - fractional flow of water at breakthrough
c      slmax - slope at tangent point
c
c*****
c
c      subroutine frac(m)
c      implicit real*8 (a-h,k-z)
c      include 'd1.com'
c      tt=(tem+temb)/2.
c      call viscos(tt,ug2,uo2,uw2)
c      slmax = 0.0
c      slold=-100.0
c      fwmax = 1.0
c      b1 = swd(2) - 0.00001
c      do 10 j=1,99999
c          swf = swd(1) + j*.0001
c          call fweval(swf,y)
c          if(y.gt.fwmax) fwmax = y
c          slmax = y/(swf - swd(1))
c          if(slmax.lt.slold) goto 20
c          if(swf.ge.b1) goto 20

```

APPENDIX

```

    slold=slmax
10 continue
20 swbfbt = swd(1) + 1.0/slmax
    so=1.-swbfbt
    call rel3(so,swbfbt,dum,krw,krg)
    call rel3(soi,swi,kro,dum,krg)
    m = krw*uo2/(kro*uw2)
    return
end
c*****
c
c          SUBROUTINE FWEVAL
c
c          EVALUATE fw AT sw
c
c  IN: swirr - irreducible water saturation
c      swmax - maximum water saturation
c      krwend - water relative permeability endpoint
c      uw - water viscosity
c      uo - oil viscosity
c      sw - water saturation
c
c  OUT: fw - fractional flow of water, fraction
c
c*****
c      subroutine fweval(sw,fw)
c      implicit real*8 (a-h,k-z)
c      include 'dl.com'
c      swstar = (sw - swd(1))/(swd(2) - swd(1))
c      if(swstar.lt.0) swstar = 0.0
c      if(swstar.gt.1) swstar = 1.0
c      krw = (swstar**nwat)*krwd(2)
c      kro = (1.0 - swstar)**noil
c      if(krw.eq.0) then
c          fw = 0.0
c      else
c          b1 = 7.8264e-06*akx*kro*wid*ht*(denwi-denoil)*s/(uo2*qwmax)
c          fw=(1.0 + b1)/(1.0 + uw2*kro/(uo2*krw))
c      endif
c      return
c      end
c*****
c
c          SUBROUTINE DFWEVA
c
c          EVALUATE dfw/dsw AT sw
c
c  IN: swirr - irreducible water saturation

```

APPENDIX

```

c   swmax - maximum water saturation
c   krwend - water relative permeability endpoint
c   uw   - water viscosity
c   uo   - oil viscosity
c   sw   - water saturation
c
c OUT: dfw - slope of tangent at sw
c
c *****
c   subroutine dfweva(sw,dfw)
c   implicit real*8 (a-h,k-z)
c   swm = sw - 0.00000001
c   swp = sw + 0.00000001
c   call fweval(swm,fwm)
c   call fweval(swp,fwp)
c   dfw = (fwp - fwm)/(swp - swm)
c   return
c   end
c *****
c
c   SUBROUTINE FRACG
c
c   FRACTIONAL FLOW CURVE FOR GAS
c
c OUT: sgf - gas saturation at front
c   sgbf - average gas saturation behind front
c   fgbt - fractional flow of gas at breakthrough
c   slmaxg - slope at tangent point
c
c *****
c   subroutine fracg(vgrs,vgrw)
c   implicit real*8 (a-h,k-z)
c   include 'dl.com'
c   call viscos(tinjaw,ugl,uol,uwl)
c   call stepp(pbl,tbl,vwin,vsin,hwin,hsin,lvin)
c   dengi = 1.0/vsin
c   qtot = 0.1*qwmax*((1.-qual)+qual*denwi*vsin)
c   slmaxg = 0.0
c   slold=-100.0
c   fgmax = 1.0
c   b1 = 1.0 - sld(1) - 0.00001
c   do 10 j=1,99999
c     sgf = 1.0 - sld(2) + j*.001
c     call fgeval(qtot,dengi,sgf,fgbt)
c     if(fgbt.gt.fgmax) fgmax = fgbt
c     slmaxg = fgbt/(sgf - 1.0 + sld(2))
c     if((slmaxg.lt.slold).and.(slmaxg.gt.0)) goto 20
c     if(sgf.ge.b1) goto 20

```

APPENDIX

```

    slold=slmaxg
10 continue
20 sgbf = sgf + (1.0 - fgbt)/slmaxg
   if(lste.eq.0) vgrs = 1.0
   if((lste.ne.0).and.(ifbt.lt.21)) then
      vgrs=144.0*akx*delp2/((1.-s**2)*akz*lste*(denwi-dengi))
   endif
   if(icnt.le.4) then
      so = 1.0 - swbfbt
      call rel3(so,swbfbt,kro2,krw2,kr2)
      mobil = kro2/uo2 + krw2/uw2
      dpdx = qwmax/(0.001127*akx*wid*ht*mobil)
      if(lwat.eq.0) vgrw=1.
      if(lwat.ne.0) vgrw=144.0*akx*dpdx/(akz*(denwi-deno))
   endif
   return
end
c*****
c
c          SUBROUTINE FGEVAL
c
c  CALCULATES GAS FRACTIONAL FLOW VALUE FOR A GIVEN GAS
SATURATION
c
c  IN: sg   - gas saturation, fraction
c
c  OUT: fg   - fractional flow of gas, fraction
c
c*****
c  subroutine fgeval(qtot,dengi,sg,fg)
c  implicit real*8 (a-h,k-z)
c  include 'd1.com'
c  oil-water system kr
c  swstar = (swd(2) - swd(1) - sg)/(swd(2) - swd(1))
c  if(swstar.lt.0) swstar = 0.0
c  krw = (swstar**nwat)*krwd(2)
c  krow = (1.0 - swstar)**noil
c  liquid-gas system kr
c  slstar = (1.0 - sg - sld(1))/(sld(2) - sld(1))
c  if(slstar.lt.0) slstar = 0.0
c  krog = slstar**nliq
c  krg = krgd(1)*(1.0 - slstar)**ngas
c  if(krg.eq.0) then
c     fg = 0.0
c  else
c     b1 = 7.8264e-06*akx*krw*wid*ht*(denwi-dengi)*s/(uw1*qtot)
c     fg = (1.0 - b1)/(1.0 + ug1*krog/(uw1*krg))
c  endif

```

APPENDIX

```

      return
      end
c*****
c
c          SUBROUTINE RONNIE
c
c          CALCULATES VOLUMETRIC SWEEP EFFICIENCY
c          AT WATER BREAKTHROUGH
c*****
      subroutine ronnie(ratio,m,afac,bfac)
      implicit real*8 (a-h,k-z)
      dimension row(5),col(36),a(5,36),b(5,36)
      call ron(row,col,a,b)
      x = 0.4343*log(ratio)
      y = m
      if(x.lt.col(1)) then
        i=2
        c=0.0
        goto 24
      endif
      if(x.gt.col(36)) then
        i=36
        c=1.0
        goto 24
      endif
      do 10 i=1,36
        if(col(i).gt.x) goto 20
      10 continue
      20 c=(x-col(i-1))/(col(i)-col(i-1))
      24 if(y.lt.row(1)) then
        j=2
        r=0.0
        goto 44
      endif
      if(y.gt.row(5)) then
        j=5
        r=1.0
        goto 44
      endif
      do 30 j=1,5
        if(row(j).gt.y) goto 40
      30 continue
      40 r=(y-row(j-1))/(row(j)-row(j-1))
      44 xx1=a(i-1,j-1) + c*(a(i,j-1)-a(i-1,j-1))
        xx2=a(i-1,j) + c*(a(i,j)-a(i-1,j))
        afac=xx1+r*(xx2-xx1)
        xx1=b(i-1,j-1) + c*(b(i,j-1)-b(i-1,j-1))

```

APPENDIX

```
xx2=b(i-1,j) + c*(b(i,j)-b(i-1,j))
bfac=xx1+r*(xx2-xx1)
return
end
```

c
c

```
subroutine ron(row,col,a,b)
implicit real*8 (a-h,k-z)
dimension row(5),col(36),a(5,36),b(5,36)
col(1)=-3.0
col(2)=-2.8
col(3)=-2.6
col(4)=-2.4
col(5)=-2.2
col(6)=-2.0
col(7)=-1.8
col(8)=-1.6
col(9)=-1.4
col(10)=-1.2
col(11)=-1.0
col(12)=-0.8
col(13)=-0.6
col(14)=-0.4
col(15)=-0.2
col(16)= 0.0
col(17)= 0.2
col(18)= 0.4
col(19)= 0.6
col(20)= 0.8
col(21)= 1.0
col(22)= 1.2
col(23)= 1.4
col(24)= 1.6
col(25)= 1.8
col(26)= 2.0
col(27)= 2.2
col(28)= 2.4
col(29)= 2.6
col(30)= 2.8
col(31)= 3.0
col(32)= 3.2
col(33)= 3.4
col(34)= 3.6
col(35)= 3.8
col(36)= 4.0
row(1) = 0.039
row(2) = 0.319
row(3) = 1.441
```

APPENDIX

row(4) = 3.249
row(5) = 4.801
a(1,1) = 0.20
a(1,2) = 0.225
a(1,3) = 0.250
a(1,4) = 0.277
a(1,5) = 0.306
a(1,6) = 0.34
a(1,7) = 0.379
a(1,8) = 0.424
a(1,9) = 0.474
a(1,10) = 0.529
a(1,11) = 0.59
a(1,12) = 0.655
a(1,13) = 0.721
a(1,14) = 0.783
a(1,15) = 0.838
a(1,16) = 0.88
a(1,17) = 0.907
a(1,18) = 0.922
a(1,19) = 0.928
a(1,20) = 0.930
a(1,21) = 0.930
a(1,22) = 0.932
a(1,23) = 0.936
a(1,24) = 0.941
a(1,25) = 0.946
a(1,26) = 0.95
a(1,27) = 0.95
a(1,28) = 0.95
a(1,29) = 0.95
a(1,30) = 0.95
a(1,31) = 0.95
a(1,32) = 0.95
a(1,33) = 0.95
a(1,34) = 0.95
a(1,35) = 0.95
a(1,36) = 0.95
a(2,1) = 0.20
a(2,2) = 0.20
a(2,3) = 0.20
a(2,4) = 0.20
a(2,5) = 0.20
a(2,6) = 0.20
a(2,7) = 0.219
a(2,8) = 0.236
a(2,9) = 0.255
a(2,10) = 0.280

APPENDIX

$a(2,11) = 0.315$
 $a(2,12) = 0.363$
 $a(2,13) = 0.422$
 $a(2,14) = 0.488$
 $a(2,15) = 0.559$
 $a(2,16) = 0.63$
 $a(2,17) = 0.699$
 $a(2,18) = 0.763$
 $a(2,19) = 0.820$
 $a(2,20) = 0.868$
 $a(2,21) = 0.905$
 $a(2,22) = 0.929$
 $a(2,23) = 0.943$
 $a(2,24) = 0.949$
 $a(2,25) = 0.95$
 $a(2,26) = 0.95$
 $a(2,27) = 0.95$
 $a(2,28) = 0.95$
 $a(2,29) = 0.95$
 $a(2,30) = 0.95$
 $a(2,31) = 0.95$
 $a(2,32) = 0.95$
 $a(2,33) = 0.95$
 $a(2,34) = 0.95$
 $a(2,35) = 0.95$
 $a(2,36) = 0.95$
 $a(3,1) = 0.23$
 $a(3,2) = 0.23$
 $a(3,3) = 0.23$
 $a(3,4) = 0.23$
 $a(3,5) = 0.23$
 $a(3,6) = 0.23$
 $a(3,7) = 0.23$
 $a(3,8) = 0.23$
 $a(3,9) = 0.23$
 $a(3,10) = 0.23$
 $a(3,11) = 0.23$
 $a(3,12) = 0.23$
 $a(3,13) = 0.23$
 $a(3,14) = 0.23$
 $a(3,15) = 0.290$
 $a(3,16) = 0.35$
 $a(3,17) = 0.406$
 $a(3,18) = 0.459$
 $a(3,19) = 0.507$
 $a(3,20) = 0.553$
 $a(3,21) = 0.595$
 $a(3,22) = 0.635$

APPENDIX

$a(3,23) = 0.671$
 $a(3,24) = 0.704$
 $a(3,25) = 0.734$
 $a(3,26) = 0.76$
 $a(3,27) = 0.782$
 $a(3,28) = 0.800$
 $a(3,29) = 0.815$
 $a(3,30) = 0.827$
 $a(3,31) = 0.835$
 $a(3,32) = 0.841$
 $a(3,33) = 0.844$
 $a(3,34) = 0.846$
 $a(3,35) = 0.848$
 $a(3,36) = 0.85$
 $a(4,1) = 0.20$
 $a(4,2) = 0.20$
 $a(4,3) = 0.20$
 $a(4,4) = 0.20$
 $a(4,5) = 0.20$
 $a(4,6) = 0.20$
 $a(4,7) = 0.20$
 $a(4,8) = 0.20$
 $a(4,9) = 0.20$
 $a(4,10) = 0.20$
 $a(4,11) = 0.20$
 $a(4,12) = 0.20$
 $a(4,13) = 0.20$
 $a(4,14) = 0.20$
 $a(4,15) = 0.217$
 $a(4,16) = 0.25$
 $a(4,17) = 0.281$
 $a(4,18) = 0.310$
 $a(4,19) = 0.336$
 $a(4,20) = 0.361$
 $a(4,21) = 0.385$
 $a(4,22) = 0.408$
 $a(4,23) = 0.430$
 $a(4,24) = 0.453$
 $a(4,25) = 0.476$
 $a(4,26) = 0.50$
 $a(4,27) = 0.525$
 $a(4,28) = 0.550$
 $a(4,29) = 0.575$
 $a(4,30) = 0.599$
 $a(4,31) = 0.62$
 $a(4,32) = 0.638$
 $a(4,33) = 0.655$
 $a(4,34) = 0.670$

APPENDIX

$a(4,35) = 0.685$
 $a(4,36) = 0.70$
 $a(5,1) = 0.175$
 $a(5,2) = 0.175$
 $a(5,3) = 0.175$
 $a(5,4) = 0.175$
 $a(5,5) = 0.175$
 $a(5,6) = 0.175$
 $a(5,7) = 0.175$
 $a(5,8) = 0.175$
 $a(5,9) = 0.175$
 $a(5,10) = 0.175$
 $a(5,11) = 0.175$
 $a(5,12) = 0.175$
 $a(5,13) = 0.175$
 $a(5,14) = 0.175$
 $a(5,15) = 0.186$
 $a(5,16) = 0.21$
 $a(5,17) = 0.234$
 $a(5,18) = 0.259$
 $a(5,19) = 0.283$
 $a(5,20) = 0.307$
 $a(5,21) = 0.33$
 $a(5,22) = 0.352$
 $a(5,23) = 0.374$
 $a(5,24) = 0.396$
 $a(5,25) = 0.420$
 $a(5,26) = 0.445$
 $a(5,27) = 0.469$
 $a(5,28) = 0.492$
 $a(5,29) = 0.513$
 $a(5,30) = 0.533$
 $a(5,31) = 0.55$
 $a(5,32) = 0.565$
 $a(5,33) = 0.578$
 $a(5,34) = 0.590$
 $a(5,35) = 0.600$
 $a(5,36) = 0.61$
 $b(1,1) = 0.40$
 $b(1,2) = 0.430$
 $b(1,3) = 0.458$
 $b(1,4) = 0.488$
 $b(1,5) = 0.526$
 $b(1,6) = 0.58$
 $b(1,7) = 0.652$
 $b(1,8) = 0.736$
 $b(1,9) = 0.823$
 $b(1,10) = 0.906$

APPENDIX

$b(1,11) = 0.975$
 $b(1,12) = 1.024$
 $b(1,13) = 1.057$
 $b(1,14) = 1.077$
 $b(1,15) = 1.090$
 $b(1,16) = 1.10$
 $b(1,17) = 1.112$
 $b(1,18) = 1.125$
 $b(1,19) = 1.138$
 $b(1,20) = 1.150$
 $b(1,21) = 1.16$
 $b(1,22) = 1.167$
 $b(1,23) = 1.171$
 $b(1,24) = 1.174$
 $b(1,25) = 1.176$
 $b(1,26) = 1.18$
 $b(1,27) = 1.18$
 $b(1,28) = 1.18$
 $b(1,29) = 1.18$
 $b(1,30) = 1.18$
 $b(1,31) = 1.18$
 $b(1,32) = 1.18$
 $b(1,33) = 1.18$
 $b(1,34) = 1.18$
 $b(1,35) = 1.18$
 $b(1,36) = 1.18$
 $b(2,1) = 0.37$
 $b(2,2) = 0.37$
 $b(2,3) = 0.37$
 $b(2,4) = 0.37$
 $b(2,5) = 0.37$
 $b(2,6) = 0.37$
 $b(2,7) = 0.392$
 $b(2,8) = 0.416$
 $b(2,9) = 0.448$
 $b(2,10) = 0.491$
 $b(2,11) = 0.55$
 $b(2,12) = 0.627$
 $b(2,13) = 0.714$
 $b(2,14) = 0.805$
 $b(2,15) = 0.890$
 $b(2,16) = 0.96$
 $b(2,17) = 1.010$
 $b(2,18) = 1.042$
 $b(2,19) = 1.061$
 $b(2,20) = 1.072$
 $b(2,21) = 1.08$
 $b(2,22) = 1.089$

APPENDIX

$b(2,23) = 1.100$
 $b(2,24) = 1.111$
 $b(2,25) = 1.121$
 $b(2,26) = 1.13$
 $b(2,27) = 1.137$
 $b(2,28) = 1.141$
 $b(2,29) = 1.145$
 $b(2,30) = 1.147$
 $b(2,31) = 1.15$
 $b(2,32) = 1.15$
 $b(2,33) = 1.15$
 $b(2,34) = 1.15$
 $b(2,35) = 1.15$
 $b(2,36) = 1.15$
 $b(3,1) = 0.33$
 $b(3,2) = 0.33$
 $b(3,3) = 0.33$
 $b(3,4) = 0.33$
 $b(3,5) = 0.33$
 $b(3,6) = 0.33$
 $b(3,7) = 0.33$
 $b(3,8) = 0.33$
 $b(3,9) = 0.33$
 $b(3,10) = 0.33$
 $b(3,11) = 0.33$
 $b(3,12) = 0.398$
 $b(3,13) = 0.467$
 $b(3,14) = 0.535$
 $b(3,15) = 0.603$
 $b(3,16) = 0.67$
 $b(3,17) = 0.735$
 $b(3,18) = 0.797$
 $b(3,19) = 0.854$
 $b(3,20) = 0.906$
 $b(3,21) = 0.95$
 $b(3,22) = 0.986$
 $b(3,23) = 1.014$
 $b(3,24) = 1.036$
 $b(3,25) = 1.055$
 $b(3,26) = 1.07$
 $b(3,27) = 1.084$
 $b(3,28) = 1.097$
 $b(3,29) = 1.108$
 $b(3,30) = 1.117$
 $b(3,31) = 1.125$
 $b(3,32) = 1.131$
 $b(3,33) = 1.136$
 $b(3,34) = 1.140$

APPENDIX

$b(3,35) = 1.145$
 $b(3,36) = 1.15$
 $b(4,1) = 0.462$
 $b(4,2) = 0.462$
 $b(4,3) = 0.462$
 $b(4,4) = 0.462$
 $b(4,5) = 0.462$
 $b(4,6) = 0.462$
 $b(4,7) = 0.462$
 $b(4,8) = 0.462$
 $b(4,9) = 0.462$
 $b(4,10) = 0.462$
 $b(4,11) = 0.462$
 $b(4,12) = 0.462$
 $b(4,13) = 0.462$
 $b(4,14) = 0.462$
 $b(4,15) = 0.462$
 $b(4,16) = 0.525$
 $b(4,17) = 0.586$
 $b(4,18) = 0.645$
 $b(4,19) = 0.701$
 $b(4,20) = 0.752$
 $b(4,21) = 0.80$
 $b(4,22) = 0.843$
 $b(4,23) = 0.881$
 $b(4,24) = 0.914$
 $b(4,25) = 0.944$
 $b(4,26) = 0.97$
 $b(4,27) = 0.993$
 $b(4,28) = 1.013$
 $b(4,29) = 1.030$
 $b(4,30) = 1.046$
 $b(4,31) = 1.06$
 $b(4,32) = 1.073$
 $b(4,33) = 1.085$
 $b(4,34) = 1.097$
 $b(4,35) = 1.108$
 $b(4,36) = 1.12$
 $b(5,1) = 0.418$
 $b(5,2) = 0.418$
 $b(5,3) = 0.418$
 $b(5,4) = 0.418$
 $b(5,5) = 0.418$
 $b(5,6) = 0.418$
 $b(5,7) = 0.418$
 $b(5,8) = 0.418$
 $b(5,9) = 0.418$
 $b(5,10) = 0.418$

APPENDIX

```
b(5,11) = 0.418
b(5,12) = 0.418
b(5,13) = 0.418
b(5,14) = 0.418
b(5,15) = 0.418
b(5,16) = 0.46
b(5,17) = 0.504
b(5,18) = 0.547
b(5,19) = 0.591
b(5,20) = 0.632
b(5,21) = 0.67
b(5,22) = 0.704
b(5,23) = 0.735
b(5,24) = 0.763
b(5,25) = 0.788
b(5,26) = 0.81
b(5,27) = 0.831
b(5,28) = 0.849
b(5,29) = 0.865
b(5,30) = 0.879
b(5,31) = 0.89
b(5,32) = 0.898
b(5,33) = 0.905
b(5,34) = 0.910
b(5,35) = 0.915
b(5,36) = 0.92
return
end
c
subroutine steadj(s,m,pv,pvipd,stefac)
implicit real*8(a-h,k-z)
dimension a(5),b(2),c(3),d(5,2,3)
ad=m
bd=pv
cd=pvipd
c a = mobility
c b = pore volume
c c = pvi/day
a(1) = 0.0
a(2) = 1.0
a(3) = 1.441
a(4) = 3.249
a(5) = 999.0
b(1) = 3.0e+07
b(2) = 6.0e+07
c(1) = 0.94e-04
c(2) = 1.87e-04
c(3) = 3.74e-04
```

APPENDIX

```

d(1,1,1)=1.04
d(1,1,2)=1.04
d(1,1,3)=1.09
d(1,2,1)=0.91
d(1,2,2)=0.91
d(1,2,3)=0.96
d(2,1,1)=1.04
d(2,1,2)=1.04
d(2,1,3)=1.09
d(2,2,1)=0.91
d(2,2,2)=0.91
d(2,2,3)=0.96
d(3,1,1)=1.23
d(3,1,2)=1.27
d(3,1,3)=1.55
d(3,2,1)=1.04
d(3,2,2)=1.08
d(3,2,3)=1.36
d(4,1,1)=1.31
d(4,1,2)=1.60
d(4,1,3)=2.15
d(4,2,1)=1.11
d(4,2,2)=1.40
d(4,2,3)=1.95
d(5,1,1)=1.31
d(5,1,2)=1.60
d(5,1,3)=2.15
d(5,2,1)=1.11
d(5,2,2)=1.40
d(5,2,3)=1.95
if(ad.lt.a(1)) ad=a(1)*1.0001
if(ad.gt.a(5)) ad=a(5)*0.9999
if(bd.lt.b(1)) bd=b(1)*1.0001
if(bd.gt.b(2)) bd=b(2)*0.9999
if(cd.lt.c(1)) cd=c(1)*1.0001
if(cd.gt.c(3)) cd=c(3)*0.9999
do 10 i=1,5
  if(ad.lt.a(i)) goto 20
10 continue
20 do 30 j=1,2
  if(bd.lt.b(j)) goto 40
30 continue
40 do 50 jj=1,3
  if(cd.lt.c(jj)) goto 60
50 continue
60 afrac=(ad-a(i-1))/(a(i)-a(i-1))
  a1=d(i-1,j-1,jj-1)+(d(i,j-1,jj-1)-d(i-1,j-1,jj-1))*afrac
  a2=d(i-1,j,jj-1)+(d(i,j,jj-1)-d(i-1,j,jj-1))*afrac

```


APPENDIX

```

a3=d(i-1,j,jj)+(d(i,j,jj)-d(i-1,j,jj))*afrac
a4=d(i-1,j-1,jj)+(d(i,j-1,jj)-d(i-1,j-1,jj))*afrac
bfrac=(bd-b(j-1))/(b(j)-b(j-1))
b1=a1+(a2-a1)*bfrac
b2=a4+(a3-a4)*bfrac
cfrac=(cd-c(jj-1))/(c(jj)-c(jj-1))
stefac=b1+(b2-b1)*cfrac
b1=(0.866-s)/0.866
if(b1.gt.1.) b1=1.
if(b1.lt.0.) b1=0.
stefac=stefac*b1+0.7*(1.-b1)
return
end
c*****
c
c          SUBROUTINE YORT
c
c CALCULATES STEAM ZONE LENGTH USING YORTSOS & GAVALAS METHOD
c
c*****
c      subroutine yort(volste,stefac,nj)
c      implicit real*8 (a-h,k-z)
c      include 'dl.com'
c      b1=lste/leng/2.
c      tinit=tem*(1.-b1)+temb*b1
c      CALCULATE FLUID VALUES AT INITIAL TEMPERATURE
c      call stept(tinit,pinit,vwi,vsi,hwi,hsi,lvi)
c      CALCULATE FLUID VALUES AT INJECTION CONDITIONS
c      call stept(tinjw,pinjw,vwin,vsin,hwin,hsin,lvin)
c      MASS INJECTION RATE OF TOTAL FLUID
c      qlsmi = 5.6146*qwmax*denwst
c      MASS INJECTION RATE OF STEAM
c      qssmi = qlsmi*qual
c      LATENT HEAT INJECTION RATE
c      qlvi = qssmi*lvin
c      TOTAL HEAT INJECTION RATE
c      qhsi = qlsmi*(hwin - hwi) + qlvi
c      f = qlvi/qhsi
c      call solve(f,tda)
c      CALCULATE FLUID VALUES AT AVERAGE PRESSURE
c      if(pb1.lt.0) write(6,*) 'pb1<0! = ',pb1
c      call stepp(pb1,tb1,vws,vss,hws,hss,lvs)
c      deltmp = tb1 - tinit
c      CALCULATE POROSITY AT AVERAGE PRESSURE
c      call poros(pb1,tb1,prorp,temr,cf,cf3,por,porb1)
c      CALCULATE OIL DENSITY AT AVERAGE PRESSURE
c      call dens(pb1,tb1,prsr,temr,co,co2,co3,denro,deno1)
c      denw1 = 1.0/vws

```

APPENDIX

```

deng1 = 1.0/vss
sho1 = (.388+.00045*tinit)/(denoi/62.4)**.5
sho2 = (.388+.00045*tb1)/(deno1/62.4)**.5
shw = (hws - hwi)/deltmp
swb1 = 1.0 - sob1 - sgbf
nj=shw*deltmp/(qual*lvin)
b1 = porb1*sob1*deno1*(sho1 + sho2)/2.0
b2 = porb1*swb1*denw1*shw
b3 = porb1*sgbf*deng1*(shw+lv/deltmp)
b4 = (1.0 - porb1)*denr*shr
c  VOLUMETRIC HEAT CAPACITY OF FLUID FILLED RESERVOIR
m1 = b1 + b2 + b3 + b4
c  CUMULATIVE HEAT LOSS TO OVERBURDEN
qloss = cqhsi + qhsi*(tim - timold) - volste*deltmp*m1
c = 2*lamob*deltmp/(ht*(pi*alfob)**.5)
m2 = porb1*lv*deng1*sgbf/deltmp
b = m2/m1
td = tim*4.0*(c/(m1*deltmp))**2
sqrid = td**.5
c
c  CALCULATE STEAM LENGTH
c
if(td.le.tda) then
  lsd = sqrid - 1.0 + exp( - sqrid )
else
  lsd = f*(sqrid - b + b*exp( - sqrid/b ))
endif
lste = lsd*qhsi*m1*deltmp/(2*ht*wid*c**2)
lste = lste*stefac
c
c  CALCULATE GAS SATURATION
c
b = lste/leng
if(b.gt.1.0) b = 1.0
sgb1 = 0.7*sgbf + 0.3*(1.0 - sld(1)) + b*0.08
if(sgb1.gt.(1.-sld(1))) sgb1=1.-sld(1)
return
end
c
c
c
subroutine solve(f,tda)
implicit real*8 (a-h,k-z)
lhs = 1 - f
tdamin = 0.0
tdamax = 100.0
tda = 1.0
10 srtda = tda**.5

```

APPENDIX

```

rhs = (1.0 - exp(-srtda))/srtda
if(abs(lhs-rhs).lt..0000001) goto 20
if(rhs.gt.lhs) then
  tdamin = tda
else
  tdamax = tda
endif
tda = (tdamax + tdamin)/2.0
goto 10
20 return
end

```

c

c

```

subroutine viscos(t,ug,uo,uw)
implicit real*8 (a-h,k-z)
include 'dl.com'
if(t.ge.vist(iv)) then
  uw = visw(iv)
  uo = viso(iv)
  goto 30
endif
if(t.le.vist(1)) then
  uw = visw(1)
  uo = viso(1)
endif
do 10 i = 1,iv
  if(t.lt.vist(i)) goto 20
  if(i.ne.iv) then
    tia = vist(i)
    viswa = visw(i)
    visoa = viso(i)
  endif
10 continue
20 c1 = log(visoa)
   c2 = log(viso(i))
   c3 = log(vist(i))
   c4 = log(tia)
   c5 = log(t)
   c6 = log(viswa)
   c7 = log(visw(i))
   uo = c1 + (c2 - c1)/(c3 - c4)*(c5 - c4)
   uo = exp(uo)
   uw = c6 + (c7 - c6)/(c3 - c4)*(c5 - c4)
   uw = exp(uw)
30 ug = avg*((t + 459.6)**bvg)
return
end

```

c

APPENDIX

```

c
c SPE 17094
c Saturated Steam Property Functional Correlations for
c Fully Implicit Thermal Reservoir Simulation
c
c W. S. Totri
c S. M. Farouq Ali
c
c subroutine stept(ti,pr,vw,vs,hw,hs,lv)
c implicit real*8 (a-h,k-z)
c tk = (ti - 32.0)*5.0/9.0 + 273.15
c vw = 16.018464/(3786.31 - 37.2487*tk + 0.196246*tk**2
c & - 5.04708e-04*tk**3 + 6.29368e-07*tk**4 - 3.08480e-10*tk**5)
c hw = (23665.2 - 366.232*tk + 2.26952*tk**2 - 0.00730365*tk**3
c & + 1.30241e-05*tk**4 - 1.22103e-08*tk**5
c & + 4.70878e-12*tk**6)/2.326
c vs = -93.7072 + 0.833941*tk - 0.00320809*tk**2
c & + 6.57652e-06*tk**3 - 6.93747e-09*tk**4 + 2.97203e-12*tk**5
c vs = exp(vs)
c vs = 16.018464/vs
c hs = (-22026.9 + 365.317*tk - 2.25837*tk**2 + 0.00737420*tk**3
c & - 1.33437e-05*tk**4 + 1.26913e-08*tk**5
c & - 4.96880e-12*tk**6)/2.326
c lv = 7184500 + 11048.6*tk - 88.4050*tk**2 + 0.162561*tk**3
c & - 1.21377e-04*tk**4
c lv = (lv**5)/2.326
c pr = 0.1450382*(-175.776 + 2.29272*tk - 0.0113953*tk**2
c & + 2.62780e-05*tk**3 - 2.73726e-08*tk**4
c & + 1.13816e-11*tk**5)**2
c return
c end
c
c subroutine stepp(pin,tf,vw,vs,hw,hs,lv)
c implicit real*8(a-h,k-z)
c p = log(6894.7353*pin)
c tk = 325.442 - 44.6171*p + 8.93074*p**2 - 0.626842*p**3
c & + 0.0190017*p**4
c tf = (tk - 273.15)*1.8 + 32.0
c call stept(tf,p,vw,vs,hw,hs,lv)
c return
c end
c*****
c
c SUBROUTINE LENGTH
c
c CALCULATES LENGTHS TO FRONT LOCATIONS
c
c IN: leng - length of system

```

APPENDIX

```

c   wid  - width of system
c   ht   - height of system
c   por   - porosity
c   qwmax - water injection rate
c   tim   - time
c   sw    - initial water saturation
c   swb2  - average water saturation behind front
c
c OUT: lwat - length of water zone, ft
c   loil - length of oil zone, ft
c   lw    - length of water system, ft
c
c *****
c   subroutine length(lwt,vgrs,volste,pvibt,fracot)
c   implicit real*8 (a-h,k-z)
c   include 'dl.com'
c   porb = (lste*porb1 + lwat*porb2)/(lste + lwat)
c   b2 = qwmax*(denwt/denw2)*tim*5.6146
c   vol = (volste*(1.0 - deng1/denw1) + b2)/(leng*wid*ht*porb)
c   lwt = vol*leng/(swbfbt - swi)
c   lwat = lwt - lste
c   loil = leng - lwt
c   if(lwt.ge.leng) then
c     lwat = leng - lste
c     loil = 0.0
c   endif
c   if(lste.ge.leng) then
c     lwat = 0.0
c     loil = 0.0
c   endif
c   call shape(vgrs,pvibt,fracot)
c   volste = 0.0
c   do 50 i=1,igbk
c     volste = volste + ls(i)*wid*blsizk(i)*porb1*sgb1
c 50 continue
c   return
c   end
c *****
c
c   SUBROUTINE SHAPE
c
c   CALCULATES SHAPE OF STEAM FRONT
c
c   OUT:
c   h(i) = dimensionless height of each zone
c   la(i) = length adjustment for each layer water front
c   ls(i) = length of steam zone for layer, feet
c   lw(i) = length of water zone for layer, feet

```

APPENDIX

```

c  lo(i) = length of oil zone for layer, feet
c
c*****
subroutine shape(vgrs,pvibt,fracht)
implicit real*8 (a-h,k-z)
include 'd1.com'
dimension h(20)
x = vgrs**0.5
vg = (x + 1.0)/x
c --- calculate fractional area for each layer
do 10 i=1,igbk
  h(i)=blsizk(i)/ht
10 continue
c
c --- calculate steam lengths ---
c
  old = 0.0
  do 20 i = igbk,1,-1
    y2 = h(i) + old
    ls(i) = lste*(y2**vg - old**vg)/h(i)
    old = old + h(i)
  20 continue
  do 30 i=igbk,1,-1
    ls(i) = ls(i)*lste/ls(1)
    if(ls(i).gt.leng) ls(i) = leng
  30 continue
  tm=(ht-bliszk(1))/(lste-ls(igbk))
  tb=-tm*ls(igbk)
c
c --- calculate water lengths ---
c
  lwtb = leng*pvi/pvibt
  if(lwtb.gt.leng) lwtb=leng
  arewat=0.
  do 40 i=1,igbk
    lw(i) = lwtb - ls(i)
    if(lw(i).lt.0.) lw(i) = 0.0
    arewat=arewat+blsizk(i)*lw(i)
  40 continue
  do 50 i=1,igbk
    lo(i) = leng - lw(i) - ls(i)
    if(lo(i).lt.0.0) lo(i) = 0.0
  50 continue
c
c --- calculate hot water zone ---
c
  th=0.0
  arehot=0.

```

APPENDIX

```

do 60 i=igbk,1,-1
  th=th+blsizk(i)/2.
  tx=(th-tb)/tm
  if(tx.ge.leng) tx=leng
  txr=tx-ls(i)
  if(txr.lt.0.0) txr=0.0
  arehot=arehot+blsizk(i)*txr
  th=th+blsizk(i)/2.
60 continue
  if(arewat.eq.0) then
    frahot=1.0
  else
    frahot=arehot/arewat
  endif
  return
end
c*****
c
c          SUBROUTINE INJ
c
c          CALCULATES PRESSURE DROP IN INJECTION WELL
c
c  IN: qwmax - water injection rate
c      wi    - water injectivity index
c      p0    - pressure in injection well
c
c  OUT: delp1 - pressure drop in injection well
c      p1     - pressure just outside of injection well
c
c*****
c  subroutine inj
c  implicit real*8 (a-h,k-z)
c  include 'dl.com'
c  call dens(p0,tem,prsr,temr,cw,cw2,cw3,denrw,denw)
c  delp1 = qwmax*denwst/(denw*wi)
c  p1 = p0 - delp1
c  return
c  end
c*****
c
c          SUBROUTINE POROS
c
c          CALCULATES POROSITY AS A FUNCTION OF TEMPERATURE AND
c          PRESSURE
c
c  IN: p    - pressure, psia
c      t    - temperature, deg F
c      pr   - reference pressure for porosity, psia

```

APPENDIX

```

c   tr - reference temperature for porosity, deg F
c   cf - formation compressibility, 1/psi
c   cf3 - formation thermal expansion coefficient, 1/deg F
c   por - porosity, reference, fraction
c
c OUT: phi - porosity, fraction
c
c*****
c   subroutine poros(p,t,pr,tr,cf,cf3,por,phi)
c   implicit real*8 (a-h,k-z)
c   phi = por*(1.0 + cf*(p - pr) - cf3*(t - tr))
c   return
c   end
c*****
c
c   SUBROUTINE DENS
c
c   CALCULATES DENSITY AS A FUNCTION OF TEMPERATURE AND
c   PRESSURE
c
c   IN: p - pressure, psia
c   t - temperature, deg F
c   pr - reference pressure, psia
c   tr - reference temperature, deg F
c   c1 - compressibility, 1/psi
c   c2 - thermal coefficient 1, 1/deg F
c   c3 - thermal coefficient 2, 1/deg F/deg F
c   denref - reference density, lbm/cu.ft
c
c   OUT: den - density at p and t, lbm/cu.ft.
c
c*****
c   subroutine dens(p,t,pr,tr,c1,c2,c3,denref,den)
c   implicit real*8 (a-h,k-z)
c   b1 = 1.0 - c1*(p - pr)
c   b2 = 1.0 + c2*(t - tr) + (c3*(t - tr)**2)/2.0
c   den = denref/(b1*b2)
c   return
c   end
c*****
c
c   SUBROUTINE STE
c
c   CALCULATES PRESSURE DROP THROUGH STEAM ZONE
c
c*****
c   subroutine ste(qste)
c   implicit real*8 (a-h,k-z)

```


APPENDIX

```

include 'd1.com'
call viscos(tb1,ug1,u01,uw1)
call viscos(tem,ug,u0,uw)
sob1 = 1.d0 - swd(2)
swb1=1.-sob1-sgb1
if(swb1.lt.swd(1)) swb1=swd(1)
sob1 = 1.0 - swb1 - sgb1
call rel3(sob1,swb1,kro1,krw1,krp1)
bw=denwst/denw1
denav1=deno1*sob1+denw1*swb1+deng1*sgb1
g2=lste*s*denav1/144.
mobil=krw1/uw1+krp1/ug1
delp2=qste*bw*ls(1)/(.001127*akx*wid*0.2*ht*mobil)
dpot2=delp2-g2
p2=p1-dpot2
pb1=(p2+p1)/2.
return
end
c*****
c
c  CALCULATE PRESSURE DROP IN WATER ZONE
c
c  COMMON INPUT:
c    qwmax - fluid flow rate
c    lwat  - length of water zone
c    akx   - permeability in x-direction
c    wid   - width of system
c    ht    - height of system
c    swbf  - average water saturation in water zone
c    p1    - pressure at near end of water zone
c
c  COMMON OUTPUT:
c    p3    - pressure at far end of water zone
c    delp3 - pressure drop in water zone
c
c*****
c  subroutine wat(lwt,ev,frahot)
c  implicit real*8 (a-h,k-z)
c  include 'd1.com'
c  tavg=(tb1+tem)/2.
c  tb2=frahot*tavg+(1.-frahot)*tem
c  call poros(pb2,tb2,prorp,temr,cf,cf3,por,porb2)
c  call dens(pb2,tb2,prsr,temr,co,co2,co3,denro,deno2)
c  call dens(pb2,tb2,prsr,temr,cw,cw2,cw3,denrw,denw2)
c  xdw=lwt/leng
c  if(xdw.gt.1.) xdw=1.
c  xds=lste/leng
c  if(xds.gt.1.) xds=1.

```

APPENDIX

```

plw=pres*(1.-xdw)+presb*xdw
pls=pres*(1.-xds)+presb*xds
pp=(plw+pls)/2.
tlw=tem*(1.-xdw)+temb*xdw
tls=tem*(1.-xds)+temb*xds
tt=(tlw+tls)/2.
call viscos(tb2,ug2,uo2,uw2)
call rel3(sob2,swb2,kro2,krw2,kr2)
mo=kro2/uo2
mw=krw2/uw2
bo=denos/deno2
bw=denwst/denw2
fw=mw/(mo+mw)
bt=fw*bw+(1.-fw)*bo
mobil=kro2/uo2+krw2/uw2
lenwat=lw(igbk) - ls(1) + ls(igbk)
if(ifbt.eq.21) lenwat=0.0
hgt = ht*(ev + .1*(1.0-ev))
delp3=qwmax*bt*lenwat/(.001127*akx*wid*hgt*mobil)
if(delp3.lt.0.0) delp3 = 0.0
denav2=deno2*sob2+denw2*swb2
g3=denav2*s*lenwat/144.
dpoi3=delp3-g3
p3=p2-dpoi3
pb2=(p3+p2)/2
return
end

```

c*****

c

c

SUBROUTINE OIL

c

CALCULATES PRESSURE DROP IN OIL ZONE

c

c IN: p3 - pressure at near end of oil zone

c

c OUT: pb3 - average pressure in oil zone

c

c p4 - pressure at far end of oil zone

c

c delp4 - pressure drop through oil zone

c

c*****

```

subroutine oil(lwt,ev)
implicit real*8 (a-h,k-z)
include 'dl.com'
call poros(pb3,tb3,pror,temr,cf,cf3,por,porb3)
call dens(pb3,tb3,prsr,temr,cw,cw2,cw3,denrw,denw3)
call dens(pb3,tb3,prsr,temr,co,co2,co3,denro,deno3)
swb3=swi
sob3=1.-swb3

```

APPENDIX

```

xdw=lwt/leng
if(xdw.gt.1.) xdw=1.
tb3=((tem*(1-xdw)+temb*xdw)+temb)/2.
call viscos(tb3,ug3,u03,uw3)
call rel3(sob3,swb3,kro3,krw3,kr3)
hgt = ht*(ev + .1*(1.0-ev))
if(ifbt.ge.1) lo(igbk)=0.0
delp4=qwmax*(denost/deno3)*(lo(igbk))/
& (.001127*akx*wid*hgt*(kro3/u03+krw3/uw3))
if(delp4.lt.0.0) delp4 = 0.0
denav3=deno3*sob3+denw3*swb3
g4=denav3*s*lo(igbk)/144.
dpoi4=delp4-g4
p4=p3-dpoi4
pb3=(p4+p3)/2.
return
end
c*****
c
c CALCULATE PRESSURE DROP THROUGH PRODUCTION WELL
c
c COMMON INPUT:
c   cc - shape factor
c   rw - wellbore radius
c   blsizi - size of grid block in i-direction
c   blsizj - size of grid block in j-direction
c   pi - 3.14159
c   ss - skin factor on production well
c   qwmax - fluid flow rate
c
c COMMON OUTPUT:
c   delp5 - pressure drop in production well
c
c*****
subroutine pro
implicit real*8 (a-h,k-z)
include 'd1.com'
call dens(p5,temb,prsr,temr,cw,cw2,cw3,denrw,denw5)
call dens(p5,temb,prsr,temr,co,co2,co3,denro,deno5)
b1 = cc/rw*((blsizi**2 + wid**2)/pi)**.5
b2 = log(b1) + ss
sw = swi
sg = 0.0
if(ifbt.ge.1) sw = swb2
if(ifbt.eq.21) then
  sw = swb1
  sg = sgb1
endif
endif

```

APPENDIX

```

so = 1.0 - sw - sg
call viscos(temb,ug,uo,uw)
call rel3(so,sw,kro,krw,krp)
mo = kro/uo
mw = krw/uw
mg = krg/ug
fw = mw/(mo + mw + mg)
bo = denost/deno5
bw = denwst/denw5
bt = fw*bw + (1.0 - fw)*bo
b3= 7.081e-03*ht*akx*(mo + mw + mg)
delp5=qwmax*bt*b2/b3
return
end
c*****
c
c          SUBROUTINE REL3
c  CALCULATE THREE-PHASE RELATIVE PERMEABILITY USING STONE'S 2ND
c  MODEL
c
c*****
c  subroutine rel3(soil,swat,kro,krw,krp)
c  implicit real*8 (a-h,k-z)
c  include 'd1.com'
c  swstar=(swat-swd(1))/(swd(2)-swd(1))
c  sl=soil+swat
c  slstar=(sl-sld(1))/(sld(2)-sld(1))
c  if(swstar.lt.0) then
c    krw=krwd(1)
c    krow=krowd(1)
c    write(6,*) 'REL3 swstar < 0'
c    write(6,*) 'REL3 swat=',swat
c    goto 30
c  endif
c  if(swstar.gt.1) then
c    krw=krwd(2)
c    krow=krowd(2)
c    write(6,*) 'REL3 swstar > 1',swat
c    goto 30
c  endif
c  krw=(swstar**nwat)*krwd(2)
c  krow=(1.-swstar)**noil
30 if(krow.lt.0) kro=0
c  if(slstar.lt.0) then
c    krg=krgd(1)
c    krog=krogd(1)
c    write(6,*) 'REL3 slstar < 0'
c    goto 40

```

APPENDIX

```

endif
if(slstar.gt.1) then
  krg=krgd(2)
  krog=krogd(2)
  write(6,*) 'REL3 slstar > 1'
  goto 40
endif
krog=slstar**nliq
krg=krgd(1)*(1.-slstar)**ngas
40 kro=(krow+krw)*(krog+krg)-(krw+krg)
if(kro.lt.0) kro=0.
return
end
c*****
c
c  CALCULATE PRODUCTION STATISTICS
c
c  COMMON INPUT:
c    lwat - length of water zone
c    loil - length of oil zone
c    wid  - width of system
c    ht   - height of system
c    por  - porosity
c    sob2 - average oil saturation in water zone
c    swb2 - average water saturation in water zone
c    sob3 - average oil saturation in oil zone
c    swb3 - average water saturation in oil zone
c    qwmax - water injection rate
c    tim  - time
c
c  COMMON OUTPUT:
c    roil - oil rate
c    rwat - water rate
c    cqo  - cumulative oil production
c
c*****
c
c  subroutine qo(roilol,rwatol,inum,m)
c  implicit real*8 (a-h,k-z)
c  include 'dl.com'
c  oip1=0.
c  wip1=0.
c  gip1=0.
c  oip2=0.
c  wip2=0.
c  oip3=0.
c  wip3=0.
c  do 10 i=1,igbk
c    oip1=oip1+ls(i)*wid*blsizk(i)*porb1*sob1*deno1

```

APPENDIX

```

wip1=wip1+ls(i)*wid*blsizk(i)*porb1*swb1*denw1
gip1=gip1+ls(i)*wid*blsizk(i)*porb1*(1.-sob1-swb1)
& *deng1
oip2=oip2+lw(i)*wid*blsizk(i)*porb2*sob2*deno2
wip2=wip2+lw(i)*wid*blsizk(i)*porb2*swb2*denw2
oip3=oip3+lo(i)*wid*blsizk(i)*porb3*sob3*deno3
wip3=wip3+lo(i)*wid*blsizk(i)*porb3*swb3*denw3
10 continue
oip=(oip1+oip2+oip3)/(denost*5.6146)
wip=(wip1+gip1+wip2+wip3)/(denwst*5.6146)
cgo=ooip-oip
cqw=qwmax*tim
cqw=owip-wip+cqw
cqt=cgo+cqw
roil=(cgo-cqoold)/(tim-timold)
rwat=(cqw-cqwold)/(tim-timold)
if(ifbt.eq.0) then
  rwat = 0.0
endif
if(((ifbt.ge.1).and.(ifbt.lt.21)).and.(inum.eq.1)) then
  if(roil.gt.roilol) roil = roilol
  if(rwat.lt.rwatol) rwat = rwatol
endif
if((ifbt.eq.21).and.(inum.eq.1)) then
  if(roil.gt.roilol) roil = roilol
  if(rwat.lt.rwatol) rwat = rwatol
endif
if((ifbt.eq.21).and.(m.gt.1.5)) then
  if(roil.gt.roilol) roil = roilol
  if(rwat.lt.rwatol) rwat = rwatol
endif
if(icnt.eq.0) roil = 0.0
if(roil.lt.0) roil=0.
if(rwat.lt.0) rwat=0.
if(cgo.lt.0) roil=0.
rcgo=rcgo+roil*(tim-timold)
rcqw=rcqw+rwat*(tim-timold)
write(3,100) tim,roil,ifbt,sob2
100 format(t2,f9.2,t12,f12.3,t29,i2,t35,f8.5)
write(4,101) tim,rwat,ifbt,inum
101 format(t2,f9.2,t12,f12.3,t29,i2,t35,i3)
write(7,102) tim,rcgo,cgo
102 format(t2,f9.2,t12,f12.1,t27,f12.1)
write(9,103) tim,rcqw,cqw
103 format(t2,f9.2,t12,f12.1,t27,f12.1)
cqoold=cgo
cqwold=cqw
cqhsi=cqhsi+qhsi*(tim-timold)

```

APPENDIX

```

cqlvi=cqlvi+qlvi*(tim-timold)
timold=tim
roilol=roil
rwatol=rwat
return
end
c*****
c
c          SUBROUTINE OUT
c
c          OUTPUT RESULTS TO A FILE
c
c*****
c
c      subroutine out
c      implicit real*8 (a-h,k-z)
c      include 'dl.com'
c      dum=0.0
c      write(2,*) 'time=',tim
c      write(2,10)
c10 format(t11,'Injectn',t23,'Steam',t41,'Water',
c      &t59,'Oil',t73,'Productn')
c      write(2,20)
c20 format(t11,'Well',t23,'Zone',t41,'Zone',
c      &t59,'Zone',t73,'Well')
c      write(2,*) ' '
c      do 30 i=1,igbk
c          ll=ls(i)+lw(i)
c          write(2,40) dum,dum,ls(i),ll,leng,leng
c30 continue
c40 format('length',t9,f7.1,t17,f7.1,t31,f7.1,t49,f7.1,
c      &t64,f7.1,t72,f7.1)
c      write(2,50) sob1,sob2,sob3
c50 format('sobar',t22,f7.4,t40,f7.4,t58,f7.4)
c      write(2,60) swb1,swb2,swb3
c60 format('swbar',t22,f7.4,t40,f7.4,t58,f7.4)
c      write(2,70) sgb1,dum,dum
c70 format('sgbar',t22,f7.4,t40,f7.4,t58,f7.4)
c      write(2,80) p0,p1,p2,p3,p4,p5+dp
c80 format('pressure',t9,f7.1,t17,f7.1,t31,f7.1,t49,f7.1,
c      &t64,f7.1,t72,f7.1)
c      write(2,90) pb1,pb2,pb3
c90 format('pbar',t22,f7.2,t40,f7.2,t58,f7.2)
c      write(2,100) tb1,tb2,tb3
c100 format('tbar',t22,f7.2,t40,f7.2,t58,f7.2)
c      write(2,*) ' '
c      end

```

APPENDIX

A.1.2 Include file *d1.com*

```
c*****
c
c COMMON FILE FOR ENTIRE PROGRAM
c
c*****
  common akx,akz,avg,alfob,
  & blsizi,blsizk(20),bvg,beta2,
  & cqoold,cqwold,cw,cw2,cw3,co,co2,co3,cf,cf3,cc,cqhsi,cqlvi,
  & denost,denwst,denoi,denwi,deno1,denw1,dengl,deno2,denw2,deno3,
  & denw3,delp1,delp2,delp3,delp4,delp5,dpot2,dpot3,dpot4,denrw,dp,
  & denro,denr,denav1,denav2,denav3,deltim,fwcur,fwbi,fwmax,fsc,
  & fi(20),grad2,grad3,grad4,g2,g3,g4,ht,
  & igbi,igbj,igbk,ifbt,iv,icnt,
  & krwd(2),krowd(2),krgd(2),krogd(2),kx(20),kro1,kro2,
  & kro3,krw1,krw2,krw3,krq1,krq2,krq3,ls(20),lw(20),lo(20),
  & leng,loil,lwat,lste,lamob,noil,nwat,ngas,nliq,nbig,ooip,owip,
  & p0,p1,pb1,p2,pb2,p3,pb3,p4,p5,porb1,porb2,porb3,
  & por,pres,presb,prorp,prsr,psurf,p0max,pi,pvi,
  & qwmax,qtmax,qt,qual,qhsi,qlvi,qloss,rw,rtot,rcqo,rcqw,swmax,
  & swb1,swb2,swb3,
  & sob1,sob2,sob3,sgb1,
  & sgbf,swi,soi,ss,swd(2),sld(2),swf,swbfbt,slmax,s,shr,
  & tim.timmax,tem,temb,temr,tsurf,tinjt,tb1,tb2,tb3,
  & timold,uol,uo2,uwl,uw2,ugl,vist(20),viso(20),visw(20),vstm,
  & wid,wi,xd
```

A.1.3 Main data file *main.dat*

```
1 -----
1
1      DATA FILE FOR TWO-DIMENSIONAL MODEL
1
1 -----
1
1      GENERAL CONTROL
1
1 -----
1      36.5      deltim, time step size (days)
1      1460.0    timmax, maximum time (days)
1 -----
1
1      RESERVOIR DESCRIPTION
1
1 -----
```


APPENDIX

0.00	xangle, formation dip, down is positive (degrees)
355.0	leng, reservoir length (ft)
920.0	wid, reservoir width (ft)
4	igbk, number of layers in system
42.5	blsizk, size of block in k-direction, top layer (ft)
42.5	blsizk, size of block in k-direction (ft)
42.5	blsizk, size of block in k-direction (ft)
42.5	blsizk, size of block in k-direction, bottom layer (ft)
10.0	blsizi, size of production block in i-direction (ft)
0.4	por, porosity (fraction)
1500.0	kx, permeability in x-direction, top layer (md)
1500.0	kx, permeability in x-direction (md)
1500.0	kx, permeability in x-direction (md)
1500.0	kx, permeability in x-direction, bottom layer (md)
1500.0	akv, permeability in z-direction, all layers (md)
1	-----
1	
1	INITIAL CONDITIONS
1	
1	-----
350.0	pres, pressure at top of reservoir (psia)
350.0	presb, pressure at bottom of reservoir (psia)
0.65	swi, water saturation (fraction)
0.35	soi, oil saturation (fraction)
150.0	tem, temperature at top of reservoir (deg F)
150.0	temb, temperature at bottom of reservoir (deg F)
1	-----
1	
1	PVT DATA
1	
1	-----
14.7	prorp, reference pressure for porosity (psia)
14.7	prsr, reference pressure for densities (psia)
14.7	psurf, surface pressure (psia)
60.0	temr, reference temperature for porosity and density (deg F)
60.0	tsurf, surface temperature (deg F)
62.4	denrw, water density at p=prsr, t=temr (lbm/cu.ft)
0.000000	cw, water compressibility (1/psi)
0.0000	cw2, water thermal expansion coefficient one (1/F)
0.0	cw3, water thermal expansion coefficient two (1/F/F)
60.0	denro, oil density at p=prsr, t=temr (lbm/cu.ft)
0.000000	co, oil compressibility (1/psi)
0.0000	co2, oil thermal expansion coefficient one (1/F)
0.0	co3, oil thermal expansion coefficient two (1/F/F)
0.00048	avg, steam viscosity coefficient one
0.593	bvg, steam viscosity coefficient two
20	iv, number of viscosity entries
1	-----

APPENDIX

OIL AND WATER VISCOSITY AS A FUNCTION OF TEMPERATURE

vist(i) temperature (deg F)	visw(i) water viscosity (cp)	viso(i) oil viscosity (cp)
68.0	1.0	10000.0
86.0	1.0	4000.0
104.0	1.0	900.0
122.0	1.0	280.0
140.0	1.0	170.0
176.0	1.0	47.0
194.0	1.0	30.0
212.0	1.0	20.0
230.0	1.0	13.0
266.0	1.0	8.0
284.0	1.0	6.0
302.0	1.0	4.0
320.0	1.0	3.7
356.0	1.0	2.4
374.0	1.0	1.8
392.0	1.0	1.5
410.0	1.0	1.2
446.0	1.0	0.9
464.0	1.0	0.76
482.0	1.0	0.67

WATER-OIL RELATIVE PERMEABILITY AS f(Sw)

swd(i) water saturation	krwd(i) water relative permeability	krowd(i) oil relative permeability
0.15	0.00000	1.00000
0.85	0.0430	0.00000
2.0	noil, Corey exponent for oil	
3.0	nwat, Corey exponent for water	

GAS-LIQUID RELATIVE PERMEABILITY AS f(Sl)

sld(i) liquid saturation	krgd(i) gas relative permeability	krogd(i) oil relative permeability
--------------------------------	---	--

APPENDIX

```

1 -----
      .55      0.52000      0.00000
      1.00      0.00000      1.00000
2.0    ngas, Corey exponent for gas
3.0    nliq, Corey exponent for liquid
1 -----
1
1
1    THERMAL DATA
1
1 -----
0.000000  cf, formation compressibility (1/psi)
0.0      cf3, formation thermal expansion coefficient (1/F)
167.0    denr, density of reservoir rock (lbm/cu.ft)
0.21     shr, specific heat of reservoir rock (Btu/lbm-F)
24.0     lamob, thermal conductivity of overburden (Btu/ft-day-F)
0.74     alfob, thermal diffusivity of overburden (sq.ft/day)
1 -----
1
1
1    INJECTION WELL
1
1 -----
1500.0    p0max, maximum pressure at injection well (psia)
900.0     qwmax, maximum water rate at injection well (bbl/day)
0.8       qual, steam quality at injection sandface (fraction)
450.0     tinjw, temperature of injected fluid at sandface (deg F)
99000.0    wi, injectivity index (bbl/day-psi)
1 -----
1
1
1    PRODUCTION WELL
1
1 -----
10000.0    qmax, maximum liquid rate at producer (bbl/day)
200.0      p5, constant fbhp at producer, top perf (psia)
1.0        rw, wellbore radius of producer (ft)
0.0008     cc, shape factor at producer (dimensionless)
0.0        ss, skin factor at producer (dimensionless)

```