DOE/BC/14994-17
OSTI ID: 9330

AN ANALYTICAL MODEL FOR SIMULATING HEAVY-OIL RECOVERY BY CYCLIC STEAM INJECTION USING HORIZONTAL WELLS

SUPRI TR 118

By
Utpal Diwan
Anthony R. Kovscek

July 1999

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

Stanford University
Stanford, California

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

DOE/BC/14994-17
Distribution Category UC-122

An Analytical Model for Simulating Heavy-Oil Recovery by Cyclic Steam Injection Using Horizontal Wells

SUPRI TR 118

By
Utpal Diwan
Anthony R. Kovscek

July 1999

Work Performed Under Contract DE-FG22-96BC14994

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

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

Prepared by
Stanford University
Petroleum Engineering Department
65 Green Earth Sciences Bldg.
Stanford, CA 94305

Table of Contents

List of Tables

List of Figures

Abstract

In this investigation, existing analytical models for cyclic steam injection and oil recovery are reviewed and a new model is proposed that is applicable to horizontal wells. A new flow equation is developed for oil production during cyclic steaming of horizontal wells. The model accounts for the gravity-drainage of oil along the steam-oil interface and through the steam zone. Oil viscosity, effective permeability, geometry of the heated zone, porosity, mobile oil saturation, and thermal diffusivity of the reservoir influence the flow rate of oil in the model. The change in reservoir temperature with time is also modeled, and it results in the expected decline in oil production rate during the production cycle as the reservoir cools. Wherever appropriate, correlations are incorporated to minimize data requirements. A limited comparison to numerical simulation results agrees well, indicating that essential physics are successfully captured.

Cyclic steaming appears to be a systematic method for heating a cold reservoir provided that a relatively uniform distribution of steam is obtained along the horizontal well during injection. A sensitivity analysis shows that the process is robust over the range of expected physical parameters.

Acknowledgements

This work was supported by the Assistant Secretary for Fossil Energy, Office of Oil, Gas and Shale Technologies of the U.S. Department of Energy, under contract No. DEFG22-96BC14994 to Stanford University.

The research was partially supported by the Stanford University Petroleum Research Institute - A (SUPRI-A) thermal recovery affiliates.

Chapter 1

1. Introduction

Shell discovered the process of steam stimulation by accident in Venezuela during production of heavy crude by steamflooding the Mene Grande field near the eastern shore of Lake Maracaibo (Butler *et al.*, 1980). During the flood, a breakthrough of steam to the ground surface occurred and, in order to reduce the steam pressure in the reservoir, the injection well was allowed to flow back. Copious quantities of oil were produced; from this accidental discovery in 1959 came the cyclic steam stimulation process, which also goes by the name of *steam soak* and *huff and puff*. Since then, there have been several mathematical models describing the phenomenon. These range from complex numerical simulators to simple analytical expressions.

This work concerns the application of horizontal wells to thermal oil recovery. It consists of an analytical model developed to calculate oil recovery and reservoir heating during cyclic steam injection. It holds for heavy-oil, pressure-depleted reservoirs where the main driving force for production is gravity. Our objective is to present a relatively simple model taking into account gravity-drainage along the sides of the steam-oil interface, the pressure draw down driving force and the drainage of oil through the steam zone. A brief overview of the analytic and semi-analytic models for response to steam injection follows next.

1.1 Horizontal Wells

Horizontal wells are applied increasingly in steam injection projects for recovering heavy-oil (Basham *et al.*, 1998). In the well-known steam assisted gravity-drainage (SAGD) process, a horizontal injector is located above a horizontal producer (Butler *et al.*, 1980). The producer below collects and drains away the mobilized oil and water (condensed steam). Often the injector contains tubing for delivering steam to the well toe, while the annulus directs the steam to the formation and produces the excess for circulation. Recently, there has been interest in heavy-oil reservoirs in the application of a single dual-stream horizontal well, where the annulus assumes the role of the producer, and the tubing, the injector. Cyclic steaming using a single horizontal well could be considered a variant of SW-SAGD and should be useful for efficient initial heating of the reservoir volume.

The performance of such wells may be predicted from empirical correlations, simple analytical models, or thermal reservoir simulation. Empirical correlations can be extremely useful for correlating data within a field and for predicting performance of new wells in that and similar fields. However, use of such correlations for situations much different from those that led to their development might be subject to large errors. On the other hand, one can use a compositional or black-oil thermal model to predict the performance of cyclic steam operations. Thermal models are based on the fundamental laws of conservation of mass and heat. Fluid flow is related to pressure gradient through

the concept of relative permeability. In addition, a thermal model is sensitive to rock properties, fluid properties and geological features. Much of this information is often unknown and must be estimated from limited data and experience in similar situations. Furthermore, because of the complexity of the SAGD recovery process, the equations of a thermal simulator could be difficult and expensive to solve depending upon the exact scenario. Simply, an analytical model of cyclic steam injection might be useful to expose the first-order mechanisms of reservoir heating and oil production.

1.2 Cyclic Steam Injection

There are two important reasons to study cyclic steam injection with horizontal wells. Firstly, the thermal efficiency of cyclic operation is high, and it is thus attractive. Secondly, cyclic steaming to promote effective initial reservoir heating might precede continuous steam injection during a SW-SAGD recovery process (Elliott and Kovscek, 1999). The steam is usually injected at a fixed rate and known wellhead quality for a short period of time. After some heat loss in the wellbore, the steam enters the reservoir. Uniform steam distribution along the well is an important factor in the success of cyclic operations with a horizontal well (Mendonza, 1998). The bottom-hole quality and pressure may be predicted from a wellbore model of the type discussed by Fontanilla and Aziz (1982). After injection for a specified period of time, the well is shut-in and the steam is allowed to "soak" into the reservoir for another specified period. To complete the cycle, the well is

produced until the oil production rate reaches a specified minimum rate. This cyclic process is repeated until the recovery per cycle drops below an economic limit. Generally, the length of cycles increases as recovery matures. Given the bottom-hole conditions during the production cycle, the wellhead conditions may be predicted.

In principle, the reservoir simulator yields the most accurate answer, but it is only possible to generalize after many different simulations (Aziz and Gontijo, 1984). Further, the reservoir simulator is sensitive to data that are often not known or unreliable. It is natural to develop simpler analytical models (Boberg and Lantz, 1966) that account for the important mechanisms involved in this process, and from which we may draw general conclusions about performance. This indeed has been the cas,e and several models and correlations of varying degree of complexity are available in the literature for cyclic operation of vertical wells and continuous steam injection in dual horizontal wells (Butler *et al.*, 1980).

1.3 Existing Models

Briefly, let us touch upon the models that have been created so far. Most models apply to continuous steam injection, but the principles are identical. Marx and Langenheim (1959) describe a method for estimating thermal invasion rates, cumulative heated area, and theoretical economic limits for sustained hot-fluid injection at a constant rate into an idealized reservoir. Full allowance is made for non-productive reservoir heat losses. In all cases, the heat conduction losses to the overburden and the underburden impose an economic limit upon the size of the area

that can be swept from any around one injection. These depend on the reservoir conditions and heat injection rate.

Jones (1977) presented a simple cyclic steam model for heavy-oil, pressure-depleted, gravity-drainage reservoirs. The Boberg and Lantz (1966) procedure was used as the basis for the reservoir shape and temperature calculations versus time. Here, the only driving force assumed is gravity, and hence, the model tends to calculate lower initial oil rates than observed in the field. For matching, certain empirical parameters are employed.

Van Lookeren (1977) has presented calculation methods for linear and radial steam flow in reservoirs. He assumed immediate gravity overlay of the steam zone and presented analytical expressions to describe interface locus. The steam zone shape is governed by factors A_{LD} and A_{RD} which are dimensionless parameters that characterize the degree of steam override for linear and radial flow, respectively. A simplistic formulation is given to calculate the average steam zone thickness.

Myhill and Stegemeier (1978) presented a model for steam drive correlation and prediction. Assuming a piston like displacement, they modified Mandl and Volek's (1969) method to calculate the steam zone volume. It identifies a critical time beyond which the zone downstream of the advancing front is heated by the water moving through the condensation front. Also, a thermal efficiency of the steam zone is calculated as a function of the dimensionless time and the ratio of the latent heat to the total energy injected.

Butler *et al.* (1979) presented theoretical equations for gravity-drainage of heavy-oils during in-situ steam heating. The method described consists of an expanding steam zone as a result of steam injection and production of oil via the mechanism of gravity-drainage along the steam/oil interface of the steam chamber. The oil is produced through a horizontal well located at the bottom of the steam chamber. Oil flow rate is derived starting from Darcy's law. Heat transfer takes into account the thermal diffusivity of the reservoir and it is proportional to the square root of the driving force. In the case of an infinite reservoir, an analytical dimensionless expression is derived that describes the position of the interface. When the outer boundary of the reservoir is considered, the position of the interface and the oil rate are calculated numerically. Oil production scales with the square root of the height of the steam. An equation describing the growth of the steam chamber is also presented. The method is limited to gravity-drainage and linear flow of heavy-oil from horizontal wells.

Jones (1981) presented a steam drive model that is basically a combination of Van Lookeren's (1977) and Myhill and Stegemeier's (1978) methods. It is limited to continuous steam drive and it uses empirical factors to match calculated rates with measured values.

Vogel (1982) considered heat calculations for steam floods. Similar to Van Lookeren (1977), this model works on the basic assumption of instantaneous steam rise to the top of the reservoir. After this happens, the steam chamber grows downward at a very low velocity. Heat losses to the adjacent formations are

calculated by solving the heat conduction problem from an infinite plane. The model characterizes two main driving forces affecting oil production: gravity-drainage and steam drag. In his conclusions, Vogel says that, above a certain limit, injection rates have little influence on oil production.

Finally, Aziz and Gontigo (1984) presented a model that considers the flow potential to be a combination of pressure drop and gravity forces. The flow equation is derived for oil and water production based on the method illustrated by Butler *et al.* (1979). They solve a combined Darcy flow and a heat conduction problem. The structure of the model is based on Jones' (1981) method.

An important distinction to be made in regard to prior work is that to our knowledge the problem of cyclic steam stimulation of a horizontal well has not been addressed. This is the task of the next section.

Chapter 2

2. Model Development

Cyclic steam injection, commonly referred to as "Huff-n-Puff" involves steam injection into a pressure-depleted reservoir, followed by soaking, and finally the oil production period. As stated earlier, oil production is governed by the gravity driving force. The model developed in the following section incorporates gravity as a prime driving force for oil flow toward the well and thereby predicts the oil production flow rate per unit length of the horizontal well. Heat losses are included to the overburden as well as to the adjacent unheated oil-bearing formation.

2.1 Introduction

Steam introduced near the bottom of the formation through a horizontal well, displaces the oil and rises to the top of the formation where it is trapped if an impermeable cap rock exists. We assume that the steam zone adopts a triangular shape in cross section with dimensions and geometry as shown in Figs. 1 and 2. The steam injection rate around the well remains constant during injection. Therefore, the steam injection pressure generally remains nearly constant if the reservoir is pressure-depleted. Steam heats the colder oil sand near the condensation surface. During production oil drains along the condensation surface by a combination of gravity and pressure difference into the production well as does steam condensate. In addition, oil drains through the steam chamber into the production well. The mechanisms involved in oil production during cyclic steam

injection are diverse and intricate. Reduction of oil viscosity as a result of an increase in the temperature greatly improves the production response. Gravity-drainage and pressure drawdown are the major mechanisms of oil production in the case of cyclic steaming.

The model is divided into three periods: the injection period, the soaking period and the production period. Each will be described in detail. In what follows, correlations and equations are given in field units (°F, psi, ft, BTU, BPD, etc), unless otherwise noted.

2.2 Injection and Soak Periods

During the injection interval, heat losses from the steam zone to the reservoir are considered negligible although heat losses to the overburden must be considered. The oil sand near the wellbore is assumed to be at steam temperature T_s, the saturated steam temperature at the sand face injection pressure. Pressure fall-off away from the well during injection is neglected during this analysis. In the soaking period, heat is lost to the overburden and the reservoir. The enthalpy of the steam zone thus decreases while soaking.

There are certain important variables such as the steam zone volume and the steam-zone horizontal range that need to be addressed. The steam zone volume, V_s, during injection is calculated using the approach of Myhill and Stegemeier (1978)

as

$$V_s = \frac{Q_i E_{h,s}}{M_T \Delta T} \tag{1}$$

where, $E_{h,s}$ is the thermal efficiency, ΔT is the temperature rise of the steam zone above the initial reservoir temperature (assumed to be same as the temperature rise at down-hole condition, ΔT_i), and M_T is the total volumetric heat capacity of the reservoir. Heat capacity is the sum of the rock and fluid contributions and is written

$$M_T = (1-\phi)M_{rr} + \sum_{\beta=w,o,g} \phi S_\beta M_\beta \tag{2}$$

where, ϕ is the porosity, S is the phase saturation, and the subscript β refers to the individual phases. The quantity Q_i is the cumulative heat injected including any remaining heat from the previous cycles. The heat injection rate is given by

$$\dot{Q}_i = w_i (C_w \Delta T + f_{sdh} \Delta H_{vdh}) \tag{3}$$

where, w_i is the mass rate of steam injection in the reservoir, C_w is the average specific heat of water over the temperature range corresponding to ΔT, f_{sdh} and ΔH_{vdh} are the steam quality and the latent heat of vaporization both at down-hole conditions, respectively.

The thermal efficiency is calculated from the function given by Myhill and Stegemeier (1978) as

$$E_{h,s} = \frac{1}{t_D}\left\{ G(t_D) + (1-f_{hv})\frac{U(t_D - t_{cD})}{\sqrt{\pi}}\left[2\sqrt{t_D} - 2(1-f_{hv})\sqrt{t_D - t_{cD}} - \int_0^{t_{cD}} \frac{e^x erfc(\sqrt{x})}{\sqrt{t_D - x}} dx - \sqrt{\pi} G(t_D) \right] \right\} \tag{4a}$$

In the expression above, the heat losses to the overburden as well as the underburden are included. In the model proposed, the triangular cross sectional shape of the steam chamber is such that heat loss occurs only to the overburden. Thus, $E_{h,s}$ needs to be modified to be consistent with our model. Now, $E_{h,s}$ is defined as the ratio of the heat remaining in the zone to the total heat injected. Thus, the quantity $(1-E_{h,s})$ is the ratio of the heat lost to the total heat injected

$$F_l = 1-E_{h,s} \tag{4b}$$

Neglecting loss to the underburden, heat losses are approximately one-half of the value predicted by Eq. 4b. Therefore, the modified thermal efficiency is

$$(E_{h,s})_{mod} = 1- \tfrac{1}{2}\ F_l = \tfrac{1}{2}\,(1.0 + E_{h,s}) \tag{4c}$$

In Eq. 4a, $G(t_D)$ is

$$G(t_D) = 2\sqrt{\frac{t_D}{\pi}} - 1 + e^{t_D}\, erfc\left(\sqrt{t_D}\right) \tag{5}$$

The symbols t_D and t_{cD} represent dimensionless times given by

$$t_D = 4\left(\frac{M_{SH}}{M_T}\right)^2 \frac{\alpha_{SH}}{h_t^2} t \tag{6}$$

where, α_{SH} is the shale thermal diffusivity, t is the time, h_t is the total reservoir thickness and M_{SH} is the shale volumetric heat capacity. The dimensionless critical time (Myhill and Stegemeier, 1978) is defined by

$$e^{t_{cD}}\, erfc(t_{cD}) \equiv 1 - f_{hv} \tag{7}$$

The quantity f_{hv}, the fraction of heat injected in latent form, is given by

$$f_{hv} = \left(1 + \frac{C_w \Delta T}{f_{sdh} \Delta H_{vdh}}\right)^{-1} \tag{8}$$

The step-function U in Eq. 4 is defined as

$$U(x) = 0 \quad for \quad x < 0 \tag{9a}$$

$$U(x) = 1 \quad for \quad x > 0 \tag{9b}$$

We assume that the steam zone shape has a triangular cross section with a *y*-directional length equal to the length of the horizontal well, L (Fig. 1). The volume is

$$V_s = R_h L h_{st} \tag{10a}$$

where R_h is one-half of the base of the triangular heated zone and h_{st} is the steam zone thickness also referred to as "Zone Horizontal Range".

Rearranging Eq. 10a, it follows that,

$$R_h = \frac{V_s}{L h_{st}} \tag{10b}$$

2.3 Heat Remaining in the Reservoir

Boberg and Lantz (1966) give the average temperature of the steam zone, T_{avg}, as

$$T_{avg} = T_R + (T_s - T_R)[f_{HD} f_{VD} (1 - f_{PD}) - f_{PD}] \tag{11}$$

The dimensionless parameters f_{VD}, f_{HD} and f_{PD} are functions of time and represent horizontal loss, vertical loss and energy removed with the produced fluids, respectively. Aziz and Gontijo (1984) define them according to the following expressions.

The horizontal heat losses are expressed as (Aziz and Gontijo, 1984):

$$f_{HD} = \frac{1}{1+5t_{DH}} \tag{12a}$$

where,

$$t_{DH} = \frac{\alpha\left(t - t_{inj}\right)}{R_h^2} \tag{12b}$$

An alternate treatment for the calculation of f_{HD} is presented in Appendix B. The quantity f_{HD} is the average unit solution for the one-dimensional heat conduction problem in the horizontal direction:

$$\alpha\frac{\partial^2 f}{\partial x^2} = \frac{\partial f}{\partial t} \tag{12c}$$

where $f = \left(\frac{T - T_o}{T_s - T_o}\right)$ is the dimensionless temperature.

The averaged solution for this equation after making use of the appropriate boundary conditions consistent with the geometry chosen for our model is:

$$f_{HD} = \frac{1}{2} + \sqrt{\frac{t_{DH}}{\pi}}\left\{-2.0 + \exp\left(-\frac{1}{t_{DH}}\right)\right\} + erf\left(\frac{1}{\sqrt{t_{DH}}}\right)\left\{\left(\frac{2}{\sqrt{\pi}} - 1\right)t_{DH} + \frac{1}{2}\right\} \tag{12d}$$

Similarly, the vertical heat losses are expressed as (Aziz and Gontijo, 1984):

$$f_{VD} = \frac{1}{\sqrt{1+5t_{DV}}} \tag{13a}$$

where,

$$t_{DV} = \frac{4\alpha(t - t_{inj})}{h_t^2} \tag{13b}$$

where t_{inj} denotes the beginning of the injection period. In the equations above, α is the reservoir thermal diffusivity.

Note that for the first cycle, the initial amount of heat in the reservoir is set to zero. For all following cycles, the initial amount of energy is calculated based on steam zone volume and the average temperature at the end of the previous cycle. The average temperature at any time during the cycle is calculated using Eq. 11 (Boberg and Lantz, 1966). Since Boberg and Lantz's equation assumes a cylindrical shape for the heated zone, this equation is just an approximation for the triangular shape being considered. However, we use the approximations for f_{HD} and f_{VD} as employed by Aziz and Gontijo (1984) who assume a conical shape with triangular cross section. This is identical to the cross sectional shape that we have considered. The only difference is that they rotate the triangular shape through π radians whereas our coordinate system is Cartesian.

The term that accounts for the energy removed with produced fluids is given by

$$f_{PD} = \frac{1}{2Q_{MAX}} \int_0^t Q_p dt \tag{14}$$

where,

$$Q_p = 5.615(q_o M_o + q_w M_w)(T_{avg} - T_R) \tag{15}$$

where Q_{MAX} is the maximum heat supplied to the reservoir. It is calculated at the end of the soak period as the amount of heat injected plus the heat remaining in the reservoir from the previous cycle minus the losses to the overburden (Vogel, 1982)

$$Q_{MAX} = Q_{inj} + Q_{last} - 2R_h L K_R (T_s - T_R)\sqrt{\frac{t_{soak}}{\Pi \alpha}} \tag{16a}$$

$$Q_{inj} = 350.376 \overline{Q_i} w_s t_{inj} \tag{16b}$$

$$Q_{last} = V_s M_T (T_{avg} - T_R) \tag{16c}$$

where, L is the length of the horizontal well, K_R is the thermal conductivity of the reservoir, t_{soak} is the soak time, α is the reservoir thermal diffusivity, t_{inj} is the length of the injection cycle, Q_{last} is the heat remaining from the previous cycle, $\overline{Q_i}$ is the amount of heat injected per unit mass of steam, and w_s is the steam injection rate (cold water equivalent).

The steam pressure is calculated from the following approximate relationship (Prats, 1982):

$$p_s = \left[\frac{T_s}{115.95}\right]^{4.4545} \tag{17}$$

The volumetric heat capacities of oil and water are given as (Prats, 1982):

$$\begin{aligned} M_o &= (3.065 + 0.00355T)\sqrt{\rho_o} \\ M_w &= \rho_w C_w \end{aligned} \tag{18}$$

As the integral for f_{PD} is not easy to calculate, it is approximated by the following expression (Aziz and Gontijo, 1984),

$$\Delta f_{PD} = \frac{5.615(q_o M_o + q_w M_w)(T_{avg}^{n-1} - T_R)\Delta t}{2Q_{MAX}} \tag{19}$$

where, Δt is the time step, n is the time step number, q_o and q_w are the oil and water production rates, respectively. The average temperature of the steam zone in the 'n-1'th cycle is T_{avg}^{n-1}.

2.4 Production Period

In this period, the well is opened for flow from the reservoir. Presented in the following section is a model to predict oil and water production rates during the production period.

Butler *et al.* (1980) presented a series of publications related to the gravity-drainage of heavy-oil reservoirs subjected to steam injection in which the theory was directed to linear flow from horizontal wells. In this development, a similar approach is used. However, the steam zone shape has been assumed to be a prism with triangular cross section and the horizontal well lies at the bottom edge. Figure 1 shows the cross sectional shape of the zone.

It is assumed in the derivation of the flow equation that the reservoir is initially saturated with oil and water, and it is pressure-depleted. After steam flooding, the steam chamber occupies a prismatic shape. Steam is distributed uniformly along the well. Heat transfer from the steam chamber to the neighboring oil zone is via conduction. During the injection period, the average temperature of the steam chamber is assumed to be the saturation temperature of the steam. The oil is heavy enough to permit the injected steam to go to the top of the reservoir as suggested by Van Lookeren (1977). The reservoir is assumed to be initially pressure-depleted and is only negligibly re-pressurized when steam is injected. Finally, the potential that causes flow of oil into the same well that acts as a production well is a combination of gravity forces and pressure drive.

2.4.1 Development of Flow Equation

In this approach, we assume that gravity-drainage takes place through the entire steam chamber. Thus, integration is over the entire steam chamber and proceeds angularly with the appropriate approximation for the area. Gravity and

pressure drive are taken as the forces causing production. The steam zone growth is estimated in terms of the decrease in the angle of inclination of the zone with the horizontal, θ. The thickness of the steam zone is assumed to be equal to the reservoir thickness. The heated-zone geometry used to derive the flow equation is given in Fig. 1. The maximum heated dimension in the horizontal direction, perpendicular to the well, is R_h, while the maximum heated dimension in the vertical direction is the reservoir thickness, h.

According to Darcy's law, the flow rate through an incremental area, dA, normal to the direction of flow (Fig. 1b), is written

$$dq_o = -\frac{k_o}{\mu_o}(\rho_o - \rho_s)\nabla\Phi dA \tag{20}$$

where, $\nabla\Phi$ is the gradient of potential in the direction of flow, μ_o is the viscosity of oil, k_o is the effective permeability to oil, and ρ_o - ρ_s is the density difference between the oil and steam. According to Hubbert (1956), the potential is expressed as

$$\Delta\Phi = \frac{p_s - p_{wf}}{\rho_o} + g\sin\theta\ \Delta h \tag{21}$$

Here, g is the acceleration due to gravity. The differential area depends upon distance, r, above the well:

$$dA = 2r\sin\left(\frac{1}{2}d\theta\right) \approx rd\theta \tag{22}$$

The oil-phase effective permeability, k_o is the product of the absolute permeability, k, and relative permeability, k_{ro}. With r equal to $h_{st}\text{cosec}(\theta)$ and applying the approximation suggested in Eq. 22, the differential area reduces to

$$dA = Lh_{st}\,\text{cosec}\,\theta d\theta \tag{23}$$

It follows upon substitution of Eq. 23 into Eq. 20 that the flow rate becomes

$$dq_o = -\frac{k_o}{\mu_o}(\rho_o - \rho_s)\left(\frac{p_s - p_{wf}}{\rho_o} + gh_{st}\sin(\theta)\right)h_{st}L\,\text{cosec}\,\theta d\theta \tag{24}$$

Integration over one-half of the heated area and multiplication by 2 yields

$$\frac{q_o}{L} = -\frac{2k_o}{\mu_o}(\rho_o - \rho_s)\left[144.0\frac{p_s - p_{wf}}{\rho_o}\log\left\{\frac{1}{\tan\left(\frac{\theta}{2}\right)}\right\} + gh_{st}(\pi - 2\theta)\right] \tag{25}$$

2.5 Property Correlations

The steam viscosity, μ_{st}, and density, ρ_{st}, are calculated by correlations given by Farouq Ali (1982).

$$\rho_{st}\left(lb/ft^3\right) = \frac{p_s^{0.9588}}{363.9} \quad , \quad p_s \ in \ psia \tag{26}$$

$$\mu_{st}(cp) = 10^{-4}(0.2T_s + 82) \quad , \quad T_s \ in \ ^oF \tag{27}$$

The average specific heat of water is given by:

$$C_w = \frac{h_w(T_s) - h_w(T_R)}{T_s - T_R} \tag{28}$$

The water enthalpy correlation of Jones (1977) and the steam latent heat correlation of Farouq Ali (1982) are used:

$$h_w = 68\left[\frac{T_s}{100}\right]^{1.24} \quad , \quad T_s \ in \ ^oF \tag{29a}$$

$$\Delta H_{vdh} = 94(705 - T_s)^{0.55} \quad , \quad T_s \ in \ ^oF \tag{29b}$$

where T_s is the steam saturation temperature at the given bottom-hole pressure. The oil and water densities are approximated by (Aziz, 1984):

$$\rho_o = \rho_{o_{STD}} - 0.0214(T - T_{STD}) \tag{30a}$$

$$\rho_w = 82.4 - 11\ln\left[\frac{705 - T_{STD}}{705 - T}\right] \tag{30b}$$

where the subscript STD refers to standard conditions, in °F. The oil density correlation holds for oils with gravity less than 20°API and temperature less than 500°F.

2.5.1 Oil and Water Viscosities

Fluid viscosities as a function of temperature are obtained from relations given by Jones (1977)

$$\mu_o(cp) = 0.001889047 \exp\left(\frac{8956.257}{T + 460}\right) \tag{31a}$$

$$\mu_w(cp) = 0.66\left[\frac{T}{100}\right]^{-1.14} \tag{31b}$$

The correlation for oil viscosity is applicable to oils with specific gravity less than 30° API.

2.5.2 Fluid Saturations and Relative Permeabilities

After the soak period and before the production begins, it is assumed that the only mobile phase around the well is water.

$$\overline{S}_w = 1 - S_{orw} \tag{32}$$

where the subscript *orw* refers to residual oil saturation to water. Once the well is opened for production, it is assumed that the oil saturation around the well increases. Also, the water saturation is given by

$$S_w = \overline{S}_w - (\overline{S}_w - S_{wi})\frac{W_p}{WIP} \tag{33}$$

where W_p is the water production during the cycle and WIP is the amount of mobile water in place at the beginning of the cycle (Aziz and Gontijo, 1984). The normalized water saturation expression is

$$S_w^o = \frac{S_w - S_{wi}}{1 - S_{wi} - S_{orw}} \tag{34}$$

The analytical expressions presented by Farouq Ali (1982) for the relative permeabilities suggested by Gomaa (1980) are as follows:

$$k_{rw} = -0.002167 S_w^o + 0.024167 (S_w^o)^2 \tag{35a}$$

$$k_{ro} = -0.9416 + \frac{1.0808}{S_w^o} - \frac{0.13856}{(S_w^o)^2} \tag{35b}$$

with,

$$k_{ro} = 1.0 \quad if \quad S_w^o \le 0.2 \tag{35c}$$

2.6 Algorithm for Calculation Scheme

These equations and ideas are translated into a seven-step algorithm:

1. Initialize the model by inputting reservoir, fluid, and operational properties.
2. Calculate steam zone geometry (volume, height, and thickness), temperature, and saturations during injection and at the start of the production cycle.
3. Calculate oil and water production flow rates at small time steps within the production interval. Also, calculate the cumulative volume of fluids produced and check against original fluids in place.
4. Calculate the average temperature of the heated zone during production and at the end of the production cycle.
5. If additional steps are needed for cycle completion, go to Stage 3 and repeat, otherwise proceed.
6. Calculate the amount of fluids and heat remaining in the reservoir at the termination of the cycle, and thereby calculate recovery and oil-steam ratio (OSR).
7. If a new cycle is required, repeat steps, otherwise terminate calculations.

2.6.1 Computer Code

A C++ computer program was written to implement the proposed analytical model. The prime objective of this code is to validate the physical applicability of

the model described in Chapter 2. The code outputs the oil production rate versus time. For calculating the oil production, there are many preceding steps that include calculation of thermal properties of the reservoir matrix and fluids, temperature calculations based on heat losses as well as change in rheological behavior of the fluids. The correlations used are simple and easy to apply. The program also calculates the temperature decline of the steam zone with time as well as the water production rate with time. The calculations are made for each cycle that is comprised of three periods: the "injection interval", the "soak interval" and the "production interval". Note that production does not begin until the production interval starts. However, the temperature decline of the heated zone begins from the soaking interval. At the end of each such cycle, the heat remaining in the reservoir is calculated based on the temperature. This heat becomes the initial heat content for the next cycle.

2.6.2 Program Structure

1. Input data
2. Begin cycle calculations
3. Calculate fluid properties
4. Calculate steam zone geometry at the end of injection
5. Calculate average temperature of steam zone at end of soak
6. Calculate thermal properties of reservoir and fluids at each time step as a function of average temperature and saturation distribution

7. Calculate saturations and relative permeabilities of oil and water

8. Calculate pressure gradient acting as driving force

9. Calculate oil and water flow rate until the end of production

10. Calculate cumulative fluids produced

11. Check to see if the end of production or beginning of a new cycle has been reached.

The input data is illustrated in Table 1 on the next page. The complete code is provided in Appendix B.

Table 1: Data for validation of proposed analytical model with results of Elliot *et al.* (1999).

Grid System	
3D Cartesian System	
Total Number of Blocks	5568
X-Dimension (ft)	4592.0
Y-Dimension (ft)	209.92
Z-Dimension (ft)	64.288
Well Length (ft)	2624.0
Reservoir Properties	
Initial Pressure (psia)	385.132
Initial Temperature (°F)	60.8
Initial S_o (%)	85.0
Initial S_w (%)	15.0
Rock Properties	
Porosity (%)	33.0
Operating Conditions	
Injection Rate (B/D)	1.57

Chapter 3

3. Validation of Model, Results and Discussion

The proposed model is appropriate for predicting the oil production from pressure-depleted reservoirs where oil gravity is less than 30° API and steam injection temperatures not more than 450°F. If the input data are not within these limits, the correlations for fluid properties and flow behavior are inaccurate.

The model was validated and results were compared with those presented by Elliot and Kovscek (1999) for cyclic steaming prior to single-well SAGD. The data and the oil viscosity correlation employed were also taken from Elliot and Kovscek (1999), as shown in Table 1. Figure 3 depicts the comparison made. It is a plot of recovery factor versus time. The recovery factor is the ratio of cumulative oil produced to the original oil in place. As observed, the recovery factor obtained using theproposed analytical model matches quite closely with the recovery factor data presented by Elliot and Kovsce*k* at early times. At later times, it is slightly greater, but within reasonable limits.

Runs were made for an example base case and typical output is illustrated are the temperature, steam zone volume, angle of inclination of the steam zone with respect to the vertical, and the cumulative oil production versus time. A sensitivity analysis is performed in order to understand better where cyclic steaming with a horizontal well works best and to verify the limitations of the model.

3.1 Sensitivity Analysis

The sensitivity of the model was tested for the following parameters:

1. down-hole steam quality
2. formation thickness
3. steam injection rate
4. down-hole pressure
5. soaking interval
6. production interval

3.1.1 Base Case

This case was run using the properties and input data shown in Table 2. Briefly, the oil gravity is 14 °API, reservoir permeability is 1.5 D, and the pay thickness is 80 ft. The curves for the oil and water production versus time, average temperature of the steam zone versus time and the steam zone volume versus time are given in Figs. 4a to 4d.

Figure 4a shows the cumulative oil production per foot of the well in STB/ft versus time in days. All production and injection information is per foot of the horizontal well as are steam zone volumes. The three regions correspond to three cycles. Each cycle has distinct flat zones, which are the injection and soak intervals. The soak interval in a given cycle is followed by the production interval. During steam injection and soak there is no production. The instantaneous oil rate

Table 2: Input Data

Variable	Value
Reservoir permeability (Darcy)	1.5
Reservoir porosity	0.2
Initial water saturation	0.25
Connate water saturation	0.1
Well radius (ft)	0.31
Residual oil saturation to steam	0.05
Residual oil saturation to water	0.25
Initial reservoir temperature (°F)	110
Saturated steam temperature (°F)	365
Reservoir thermal conductivity (Btu/ft D. °F)	24.0
Reservoir thermal diffusivity (ft^2/D)	0.48
Time step size (days)	1.0
Injected steam quality	0.67
Steam injection rate (B/D)	1.0
API gravity of oil (°API)	14.0
Injection pressure (psi)	150.0
Pay thickness (ft)	80.0

increases rapidly at early times in the production interval, as witnessed by steep gradients in cumulative production (Fig. 4a) because the oil is hot and its viscosity is low. At later times, oil production rate decreases as the temperature declines. The production at late times is primarily due to gravity-drainage of cool oil.

A similar explanation applies to Fig. 4b for cumulative water production. Here, during steam injection and soak there is no production. The instantaneous water rate is high at early times in the production interval, accounting for condensed steam around the well. But at later times water production rate declines. The production at late times is primarily due to gravity-drainage of condensed water.

Figure 4c depicts the average heated zone temperature history. The temperature is at the steam saturation temperature during the injection interval. During the soaking period it decreases due to conduction heat losses to the adjacent reservoir and overburden. Since the rate of heat loss is proportional to the temperature gradient, the temperature decline is greatest at the beginning of the soak period. At the beginning of the production period, temperature decline increases rapidly because hot fluids are produced. At later times, the rate of decline decreases.

Figure 4d shows the volume of the heated zone versus time. The volume of the heated zone increases continuously during the injection interval. As expected, this is because the amount of injected heat increases. This trend is in agreement with Eq. 16, which gives the volume of the heated zone. In the soak and production

intervals, it remains constant at the value it reached at the end of the injection period. At the end of the production interval the volume of the heated zone collapses to a finite value, Eq. 16c, that accounts for the heat remaining in the hot zone. Therefore, the volume of the heated zone starts at this finite value at the beginning of the next cycle.

Figure 4e shows the cumulative oil-steam ratio (OSR) versus time. During the first injection and soaking intervals, there no production and the ratio is zero. In the first production interval, the OSR increases as oil is produced. At the beginning of the second injection interval, steam is injected and therefore the OSR decreases as no oil is produced. In the ensuing soaking interval, there is no steam injection nor any oil production. Hence, the OSR remains constant. In the production interval, as there is oil production, OSR increases and reaches a peak at the end of the production interval. As k_{rw} is very small, the water flow rate, which is directly proportional to the water relative permeability is also small.

Next, the various sensitivity runs are described.

3.1.2 Sensitivity to Steam Quality

This case was run using four different values, 50%, 70%, 85% and 95%, of down-hole steam quality at an identical mass injection rate. The results are depicted in Figs. 5a to 5d. Arrows are drawn to indicate the trends with increasing steam quality.

In Fig. 5a, as the steam quality increases from 50% to 95%, the cumulative oil production increases. As steam quality increases at constant mass flow rate, the total heat carried by the vapor increases. Hence, there is a greater volume heated as the quality increases, thereby increasing the quantity of heated oil flowing towards the horizontal production well. An increase in the steam quality from 50%- 85% almost doubles the cumulative oil produced at the end of three cycles.

Cumulative water production is not affected greatly by the increase in steam quality as seen in Fig. 5b. This is because the total mass of water injected is the same for all four cases.

The temperature of heated zone shown in Fig. 5c reaches a uniform steam saturation and decreases with similar trends during the soak and production intervals for all the steam quality values.

The volumes of the heated zone shown in Fig. 5d increases for increasing steam quality. As steam quality increases, the heat carried by the vapor increases, thereby causing an increased heated volume.

3.1.3 Sensitivity to Formation Thickness

This case was run using three different formation thicknesses, 80 ft, 200 ft and 300 ft, of the formation thickness holding all other parameters and variables constant. The results are depicted in Figs. 6a to 6d. Again, arrows indicate increasing pay thickness.

As the formation thickness increases, the cumulative oil and water production decreases as the volume of the reservoir contacted by the steam relative to the total volume decreases. Equation 24 predicts that gravity drainage from within the heated zone increases as the base of the heated zone, R_h, increases and θ, likewise, decreases. Hence, for identical steam volume, a short, squat heated zone produces more oil than a taller, narrower heated zone. This is shown in Fig. 6a and Fig. 6b, respectively.

Figure 6c demonstrates that the temperature of the heated zone decreases more rapidly for smaller formation thickness. Essentially, the lesser the formation thicknesses, the larger are the heat losses to the overburden. The greater the losses, the steeper are the temperature gradients. Figure 6d is self-explanatory. The larger the formation thickness, the larger the volume of the heated zone due to lesser heat losses.

3.1.4 Sensitivity to Steam Injection Rate

This case was run using three different values, 1 B/D-ft, 5 B/D-ft, and 10 B/D-ft, of the steam injection rate per foot of well. The results are depicted in Figs. 7a to 7d. Arrows indicate increasing injection rate.

Figures 7a and. 7b depict the cumulative oil and water production versus time, respectively. As the steam injection rate is increased, the oil and water production increases significantly, as greater amount of heat and water is injected and a greater volume of the reservoir is contacted, as shown in Fig. 7d.

Figure 7c shows the temperature of the heated zone. With increasing steam injection rates the net temperature decreases over a cycle is comparatively less. This behavior is attributed to the fact that higher injection rates lead to greater thermal energy in the reservoir volume, and this energy is slow to dissipate. Although the corresponding heated zone volumes for higher injection rates are larger, thermal energy dissipation rates are lower. The surface area for heat transfer does not increase as rapidly as the heated volume.

3.1.5 Sensitivity to Down-hole Steam Pressure

This case was run using three different values of downhole steam pressure, 150 psia, 200 psia, and 300 psia. Hence, the injection temperature increased while

the mass injection rates and qualities are constant. The results are depicted in Figs. 8a to 8d. Arrows indicate increasing pressure.

A slight increase is observed in the cumulative oil and water production as we increase down-hole steam pressure. The effects are small because of the following reasons. The steam quality was kept constant at 0.7. As the down-hole pressure is increased, the latent heat of vaporization decreases, thereby causing more heat to be injected as sensible heat. This fact causes the amount of heat injected to be lower. The heat loss functions are more sensitive to the temperature than to the amount of heat injected. Since the temperature increases with injection pressure, a large fraction of the injected heat is lost quickly to the adjacent layers. Another important reason could be that higher temperatures tend to cause higher initial fluid production, thus more heat is removed at the beginning of the cycle.

As shown in Fig. 8c, the temperature of the heated zone drops rapidly during soak and production for higher injection pressures. Larger initial fluid production results due to larger initial pressures, and larger heat loss at the beginning of each cycle occurs also. Heat losses are large because the temperature gradient is large.

3.1.6 Sensitivity to Steam Soak Interval

This case was run using four different values, 15 days, 25 days, 30 days and 50 days, of the soaking interval. Arrows indicate the lengthening soak interval. The results are depicted in Figs. 9a to 9b.

As shown in Fig. 9a, as the soaking interval is increased, the cumulative oil production decreases. Increase in the soaking interval implies a greater time for energy dissipation. In our case, the cumulative heat losses increase due to larger soaking periods. Therefore, the temperature of the heated zone at the end of the soaking interval is predictably lesser. As a consequence the oil viscosity is higher and mobility of oil is reduced. Hence, cumulative oil production decreases with increased soaking time.

A similar explanation can be put forth to explain the behavior of the cumulative water production depicted in Fig. 9b. However, as the viscosity of oil is not a strong function of temperature, the change in water production observed is small. Quite obviously, the production is delayed as we have longer soaking time.

3.2 Discussion

As observed, an increase in steam quality results in higher oil production, as there is more energy per unit mass of the reservoir matrix due to a greater fraction of energy injected as latent heat of vaporization. A given mass of steam carries more heat due to heat in the latent form, and hence, a greater oil rate is observed. Therefore, a higher steam quality is recommended for greater oil production.

For thicker pay zones, the energy density in the reservoir rock matrix is lowered. This is because of the greater volume of the reservoir rock to be contacted by steam. Also, the reservoir fluid acts as a heat sink. Therefore, too large a pay zone might lead to very low oil rates. On the other hand, if the pay thickness is too low, then the energy density within the zone is extremely large, resulting in large losses to the surrounding matrix.

Higher steam injection rates enhance heat delivery, thereby increasing oil production. Excessively high injection rates cause overheating of the reservoir matrix that results in larger heat losses, and thus causes a decrease in the thermal efficiency of the process. The optimum steam injection rate is the one that results in minimal heat losses and maximum heated volume.

Increasing down-hole steam pressure has a negligible effect on the oil production. The latent heat of vaporization decreases as pressure increases. The total energy carried by steam does not increase greatly as injection pressure is raised.

Chapter 4

4. Conclusions

A simple model has been developed for the cyclic steam process with a horizontal well. It incorporates a new flow equation for gravity drainage of heavy oil. Important factors such as pressure drive, gravity drive, and shape of the steam zone are incorporated into the model. Limited comparisons of the model to numerical simulation results indicate that the model captures the essential physics of the recovery process. Cyclic steaming appears to be effective in heating the near-well area and the reservoir volume. A sensitivity analysis shows that the process is relatively robust over the range of expected physical parameters.

5. Nomenclature

A	= Cross sectional area	ft^2 (m^2)
A_L	= Dimensionless factor for linear flow	
C_w	= Specific heat of water	Btu/lb.°F (kJ/kg°K)
f_{HD}	= Dimensionless factor that accounts for horizontal heat losses	
f_{VD}	= Dimensionless factor that accounts for vertical heat losses	
f_{PD}	= Dimensionless factor that accounts for heat lost by produced fluids	
f_{sdh}	= Steam quality at down-hole conditions (mass fraction)	
F_l	= Ratio of heat lost to heat injected	
g	= Gravitational acceleration	32.17 ft/s^2 or 9.81 m/s^2
h	= Formation thickness	ft (m)
h_{st}	= Steam zone thickness	ft (m)
$h_w(T)$	= Enthalpy of liquid water	Btu/lb (kJ/kg)
H_{inj}	= Amount of heat injected during a cycle	Btu (kJ)
H_{lost}	= Amount of heat remaining in the reservoir at the end of a cycle	Btu (kJ)

K_R	=	Reservoir thermal conductivity	Btu/ft.D.°F (kJ/m.d.°K)
k_o	=	Effective permeability of oil	Darcy
k_{st}	=	Effective permeability of steam	Darcy
k_w	=	Effective permeability of water	Darcy
k_{ro}	=	Relative permeability to oil	fraction
k_{rw}	=	Relative permeability to water	fraction
ΔH_{lvdh}	=	Latent heat of vaporization at down-hole conditions	Btu/lb (kJ/kg)
L	=	Length of horizontal well	ft (m)
M_o	=	Volumetric oil heat capacity	Btu/ft^3.°F (kJ/m^3.°K)
M_w	=	Volumetric water heat capacity	Btu/ft^3.°F (kJ/m^3.°K)
M_{SH}	=	Volumetric shale (overburden / underburden) heat capacity	Btu/ft^3.°F (kJ/m^3.°K)
M_T	=	Total volumetric heat capacity	Btu/ft^3.°F (kJ/m^3.°K)
M_σ	=	Volumetric rock heat capacity (M_{rr})	Btu/ft^3.°F (kJ/m^3.°K)
N_p	=	Cumulative oil production	bbl (m^3)
OIP	=	Volume of oil in place	bbl (m^3)

p_s	= Down-hole steam pressure	psia, (kPa)
p_{wf}	= Bottom hole flowing pressure	psia, (kPa)
q_o	= Oil production rate	BPD (m^3/d)
q_w	= Water production rate	BPD (m^3/d)
$\overline{Q_i}$	= Amount of heat injected per unit mass of steam	Btu/lb (kJ/kg)
Q_i	= Cumulative heat injected	Btu/lb (kJ/kg)
Q_{MAX}	= Maximum heat entering in the reservoir in one cycle	Btu (kJ)
Q_l	= Heat loss to reservoir	Btu (kJ)
w_s	= Steam injection rate (cold water volume)	BPD (m^3/d)
Q_p	= Rate of heat withdrawal from reservoir with produced fluids	Btu/D (kJ/d)
r_w	= Well radius	ft (m)
R_h	= Heated zone horizontal range	ft (m)
R_x	= Distance along hot oil zone interface	ft (m)
S_{ors}	= Residual oil saturation in the presence of steam	
S_{orw}	= Residual oil saturation in the presence of water	
S_{wi}	= Initial water saturation	
$\overline{S}_w$	= Average water saturation	
t_{DH}	= Dimensionless time for horizontal heat loss	
t_{DV}	= Dimensionless time for vertical heat loss	

t_{inj}	= Injection period	days
t_{prod}	= Production time	days
t_{soak}	= Soak time	days
T_{avg}	= Average temperature of heated zone	°F (°K)
T_R	= Original reservoir temperature	°F (°K)
T_S	= Down-hole steam temperature	°F (°K)
T_{SOAK}	= Temperature at the end of soaking period	°F (°K)
T_{STD}	= Temperature at standard conditions	°F (°K)
u	= Velocity of the steam-oil interface growth downwards	ft/sec (cm/sec)
v	= Darcy velocity	ft/sec (cm/sec)
V_s	= Steam zone volume	ft^3 (m^3)
w_i	= Total injection rate	BPD (m^3/d)
WIP	= Volume of water in place	bbl (m^3)
W_p	= Cumulative water production	BPD (m^3/d)
α	= Reservoir thermal diffusivity	ft^2/D (m^2/d)
α_{SH}	= Shale thermal diffusivity	ft^2/D (m^2/d)
Δp	= Pressure drawdown	psi (kPa)
Δt	= Time step size	days
φ	= Porosity	
ϕ	= Potential	ft^2/sec^2 (m^2/sec^2)

μ_o	= Oil viscosity	cp
μ_{st}	= Steam viscosity	cp
μ_w	= Water viscosity	cp
ν	= Kinematic oil viscosity	cs
ν_{avg}	= Kinematic oil viscosity at average temperature	cs
ν_s	= Kinematic viscosity of oil at steam temperature	cs
ν_{ws}	= Kinematic viscosity of water at steam temperature	cs
ρ_o	= Oil density	lb/ft^3 (gm/ml)
ρ_{oSTD}	= Oil density at standard conditions	lb/ft^3 (gm/ml)
ρ_{st}	= Steam density	lb/ft^3 (gm/ml)
ρ_w	= Water density	lb/ft^3 (gm/ml)
θ	= Angle between steam-oil interface and reservoir bed	degrees (radians)
ξ	= Hot oil zone thickness	cm

6. References

1. Aziz, K. and Gontijo J.E.: "A Simple Analytical Model for Simulating Heavy-oil Recovery by Cyclic Steam in Pressure-Depleted Reservoirs", paper SPE 13037 presented at the 59th Annual Technical Conference and Exhibition, Houston (September 16-19, 1984).
2. Basham, M., Fong, W.S. and Kumar, M.: "Recent Experience in Design and Modeling of Thermal Horizontal Wells", paper 119, presented at the 9th UNITAR International Conference in Heavy Crude and Tar Sands, Beijing, China (October 27-30, 1998).
3. Boberg, T.C. and Lantz, R.B.: "Calculation of the Production Rate of a Thermally Stimulated Well", J.Pet.Tech. (December 1966).
4. Butler, R.M. and Stephens, D.J.,: "The Gravity-drainage of Steam-Heated Heavy-Oil to Parallel Horizontal Wells", paper presented at the 31st Annual Technical Meeting of The Petroleum Society of CIM in Calgary (May 25-28, 1980).
5. Carlslaw, H.S. and Jaeger, J.C.,: "Conduction of Heat in Solids", 2nd ed., Oxford Science Publications, 1992.

6. Elliot, K.E. and Kovscek, A.R.: "Simulation of Early-Time Response of Single-Well Steam Assisted gravity-drainage (SW-SAGD)", SPE 54618, presented at the Western Regional Meeting of the SPE, Anchorage, Alaska (May 26-28, 199).

7. Farouq Ali, S.M.: "Steam Injection Theories – A Unified Approach", paper SPE 10746 presented at the California Regional Meeting of the SPE, San Francisco (March 24-26, 1982).

8. Fontanilla, J.P. and Aziz, K.: "Prediction of Bottom-Hole Conditions for Wet Steam Injection Wells", J.Can.Pet.Tech. (March-April 1982).

9. Gomaa, E.E.: "Correlations for Predicting Oil Recovery by Steamflood", J.Pet.Tech. (February 1980) 325-332.

10. Hubbert, M.K. "Darcy's Law and the Field Equations of the Flow of Underground Fluids", Shell Development Cp., J.Pet.Tech. (Sept.12, 1956).

11. Jones, J.: "Cyclic steam Reservoir Model for Viscous Oil, Pressure-depleted, Gravity-drainage Reservoirs", SPE 6544, 47th annual California Regional Meeting of the SPE of AIME, Bakersfield (April 13-15, 1977).

12. Mandl, G. and Volek, C.W.: "Heat and Mass Transport in Steam-Drive Processes", *Soc. Pet. Eng. J.* (March 1969) 59-79; *Trans.*, AIME, **246**.

13. Marx, J.W. and Langenheim, R.H.: "Reservoir Heating by Hot-Fluid Injection", Trans. AIME (1959) 216, 312-315.

14. Mendonza, H.: "Horizontal Well Steam Stimulation: A Pilot Test in Western Venezuela", paper 129, presented at the 9th UNITAR International Conference in Heavy Crude and Tar Sands, Beijing, China (October 27-30, 1998).

15. Myhill, N.A. and Stegemeier, G.L.: "Steam Drive Correlation and Prediction", J. Pet. Tech. (February 1978) 173-182.

16. Prats, M.: "Thermal Recovery", Monograph – Volume 7, SPE of AIME, Henry L. Doherty Memorial Fund of AIME, 1982.

17. Van Lookeren: "Calculation Methods for Linear and Radial Steam Flow in Oil Reservoirs", J.Pet.Tech. (June 1983) 427-439.

18. Vogel, V.: "Simplified Heat Calculation of Steamfloods", paper SPE 11219 presented at the 57th SPE Annual Fall Technical Conference and Exhibition, New Orleans, LA (September 26-29, 1982).

Figures

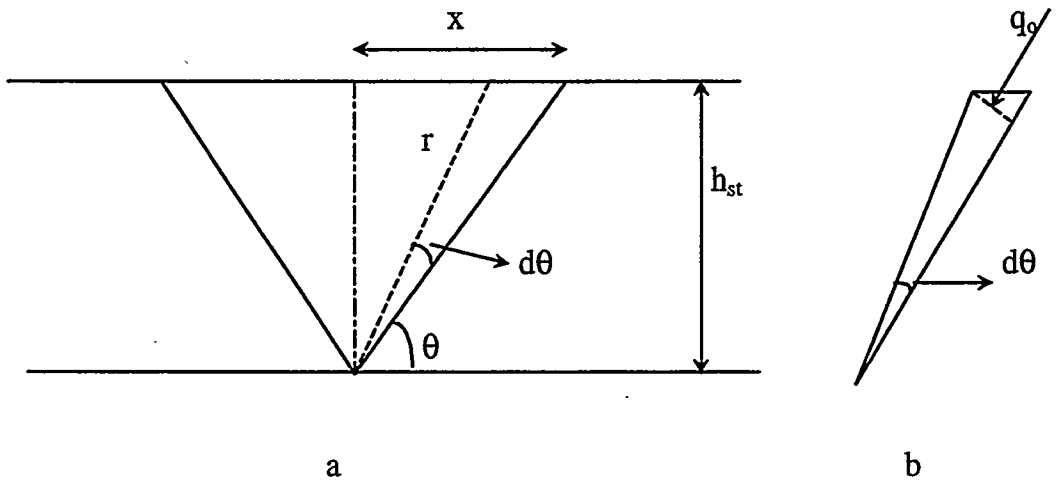


Fig. 1a. Schematic diagram of heated area geometry; b differential element of the heated area.

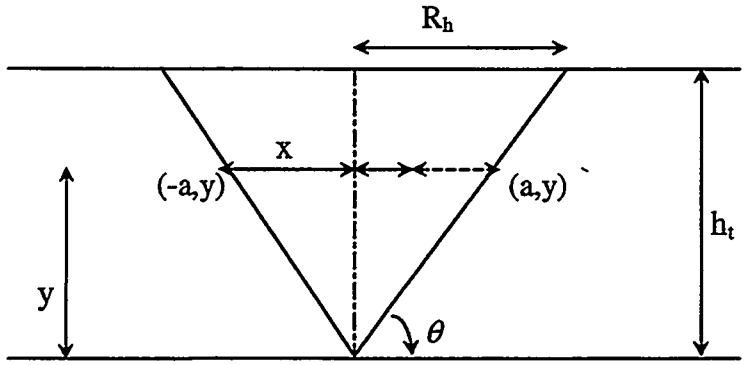


Fig. 2. Schematic diagram of area of cross section of heated area.

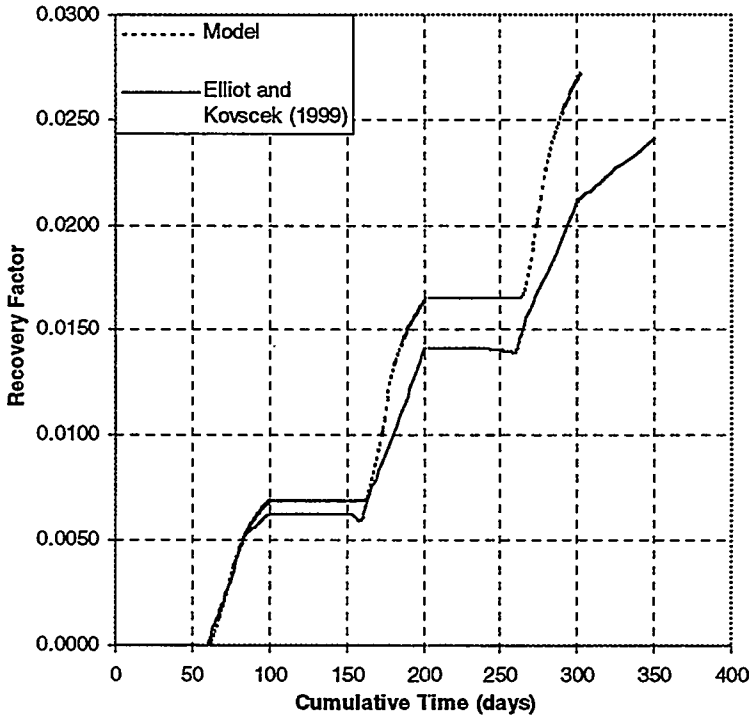
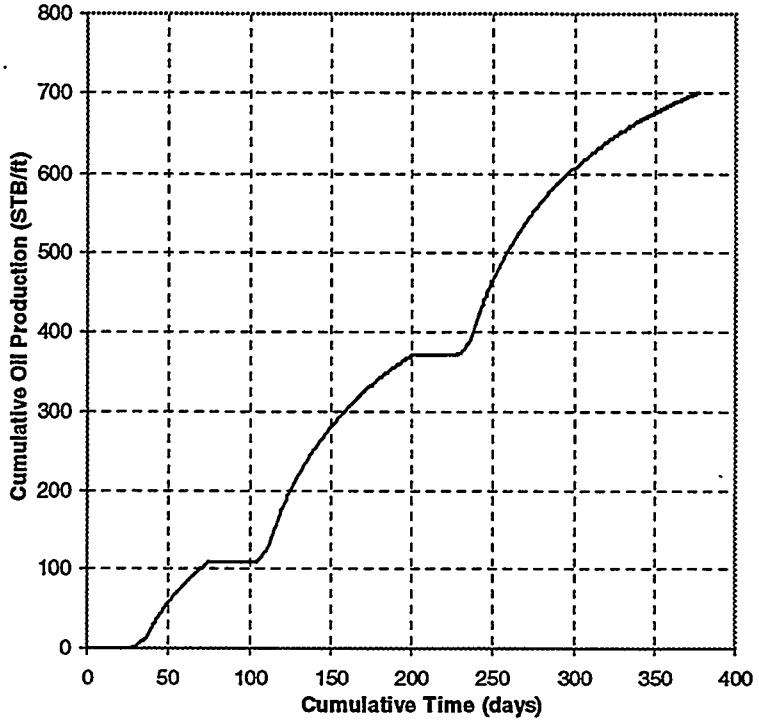
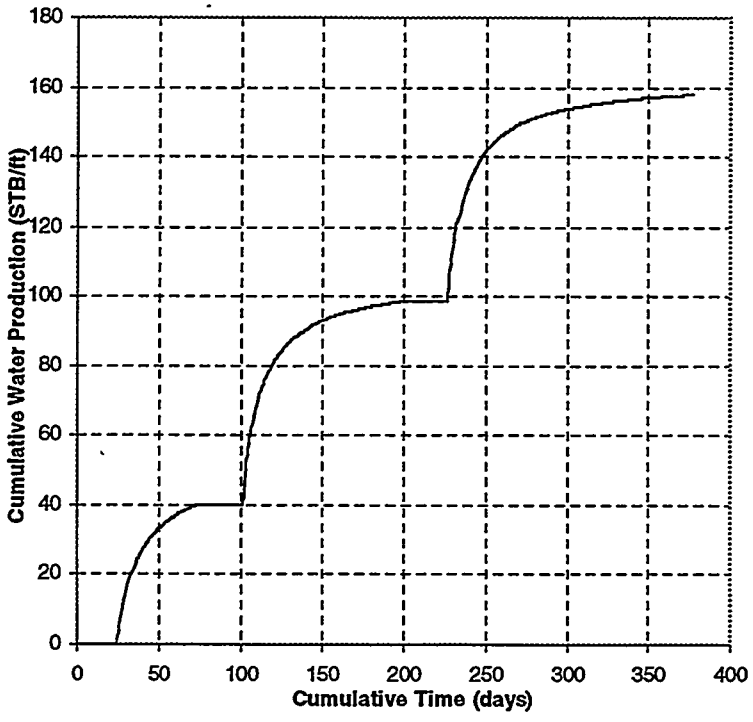


Fig. 3. Recovery factor versus cumulative time (Comparison with results Elliot and Kovscek, 1999)

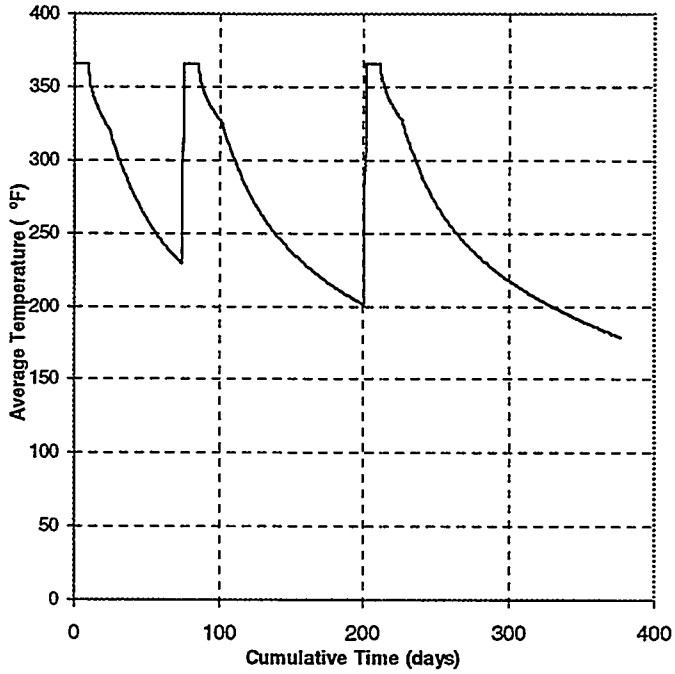
4 a


Cumulative Oil Production (STB/ft)
800
700
600
500
400
300
200
100
0
0
50
100
150
200
250
300
350
400
Cumulative Time (days)

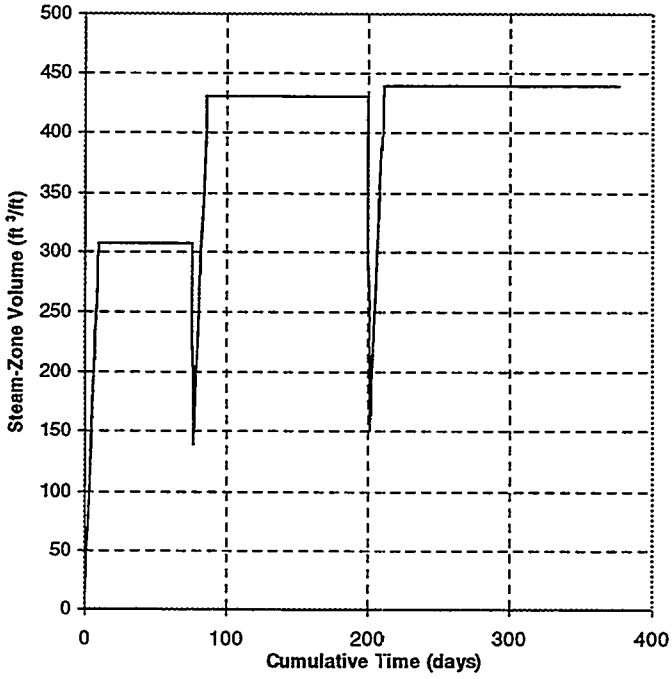
4 b


Cumulative Water Production (STB/ft)
180
160
140
120
100
80
60
40
20
0
0
50
100
150
200
250
300
350
400
Cumulative Time (days)

4 c


Average Temperature (°F)
400
350
300
250
200
150
100
50
0
0
100
200
300
400
Cumulative Time (days)

4d


Steam-Zone Volume (ft ³/ft)
500
450
400
350
300
250
200
150
100
50
0
0
100
200
300
400
Cumulative Time (days)

4e

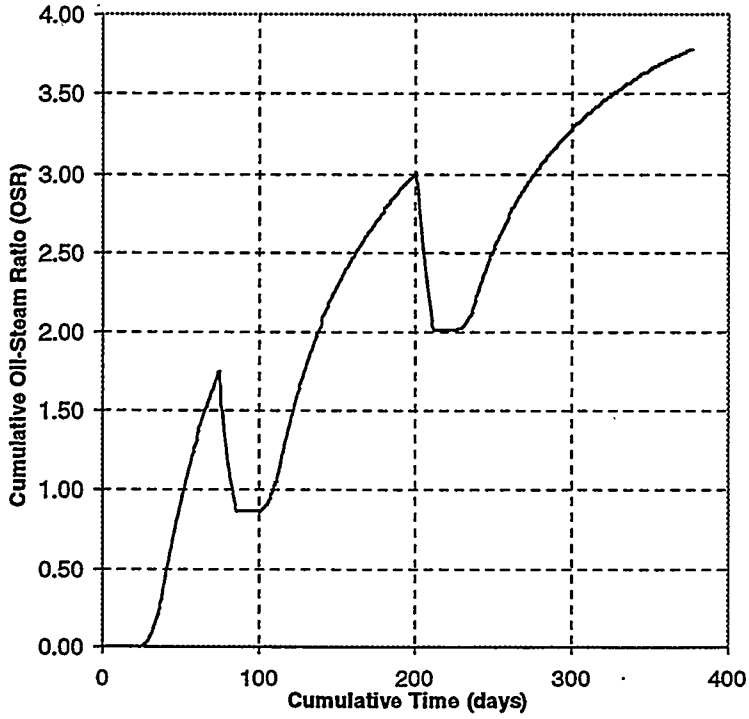
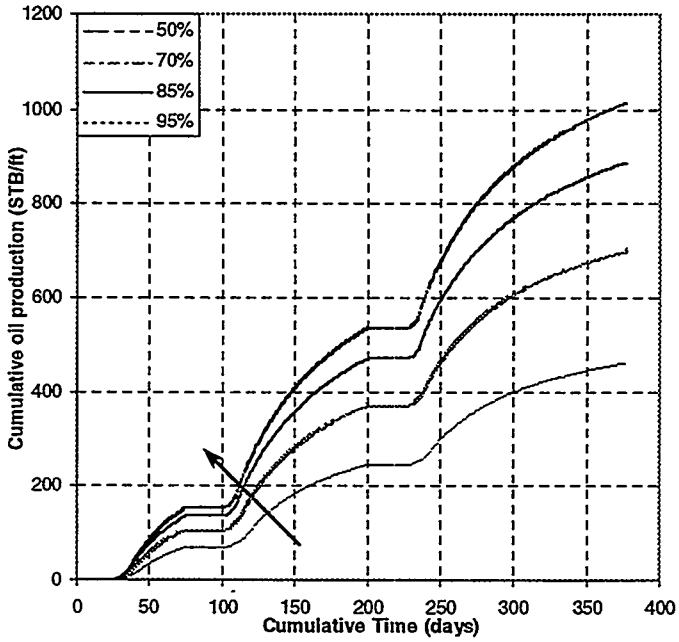
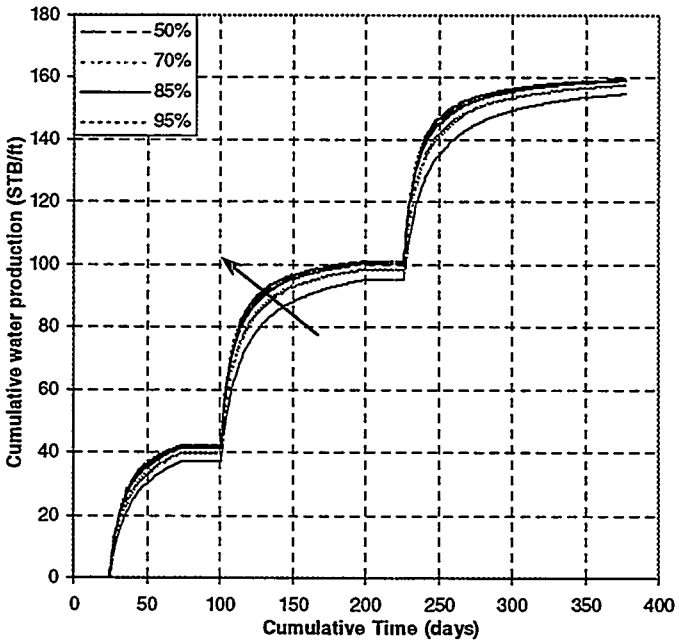


Fig. 4 Base case: a cumulative oil production versus cumulative time,
b cumulative water production versus cumulative time,
c average steam zone temperature versus cumulative time,
d steam zone volume versus cumulative time,
e oil-steam ratio versus cumulative time.

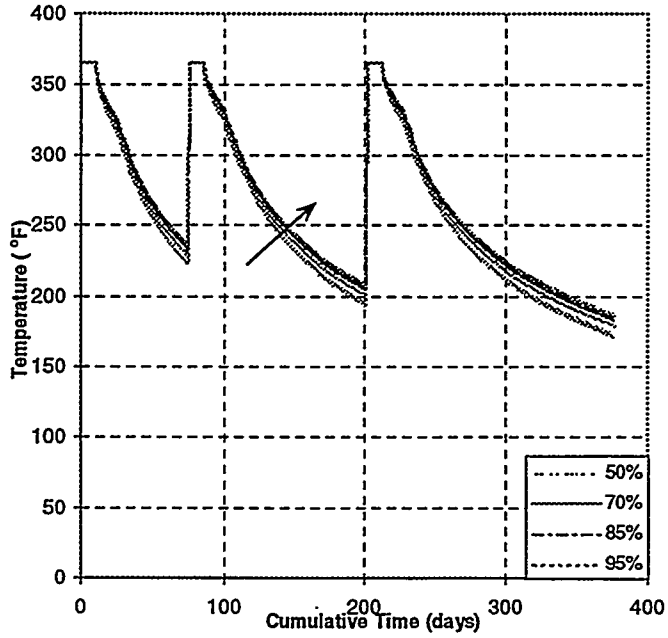
5a


50%
70%
85%
95%
Cumulative oil production (STB/ft)
Cumulative Time (days)
0
200
400
600
800
1000
1200
0
50
100
150
200
250
300
350
400

5b


50%
70%
85%
95%
Cumulative water production (STB/ft)
Cumulative Time (days)
0
20
40
60
80
100
120
140
160
180
0
50
100
150
200
250
300
350
400

5c



5d

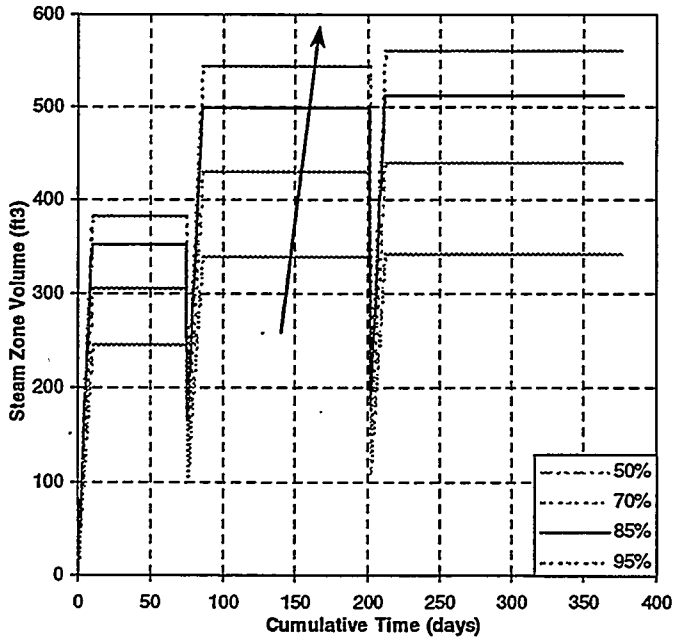
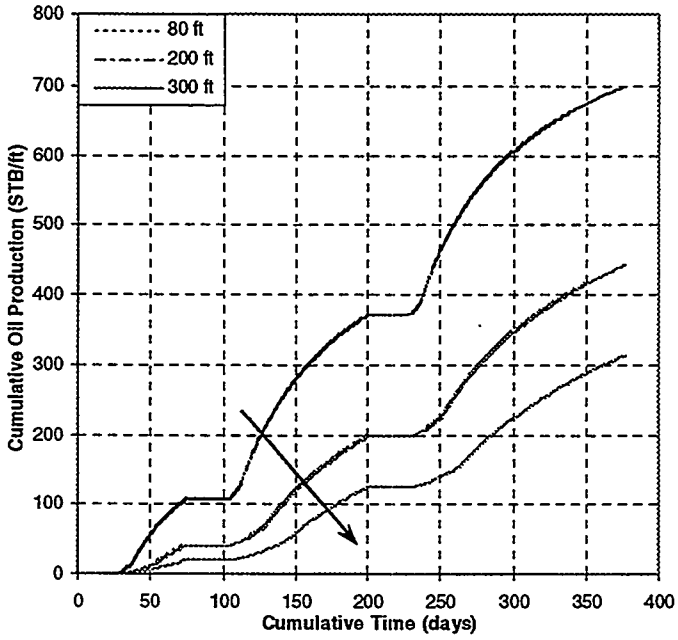


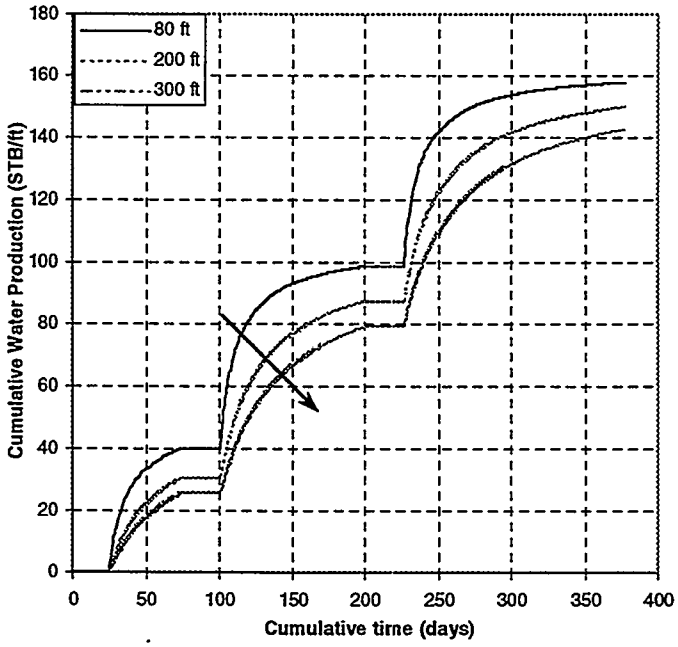
Fig. 5. A sensitivity to steam quality:

a cumulative oil production versus cumulative time,
b cumulative water production versus cumulative time,
c average steam zone temperature versus cumulative time,
d steam zone volume versus cumulative time.

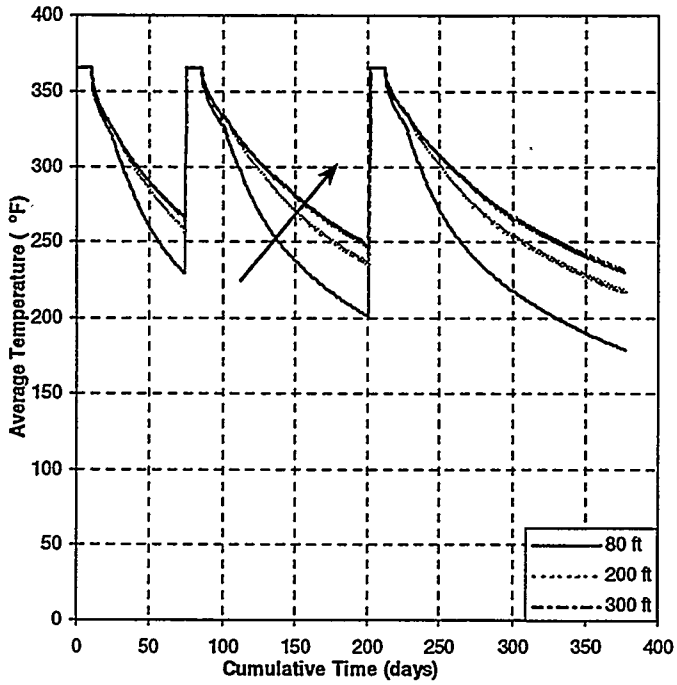
6a


80 ft
200 ft
300 ft
Cumulative Oil Production (STB/ft)
Cumulative Time (days)
0
100
200
300
400
500
600
700
800
0
50
100
150
200
250
300
350
400

6b


80 ft
200 ft
300 ft
Cumulative Water Production (STB/ft)
Cumulative time (days)
0
20
40
60
80
100
120
140
160
180
0
50
100
150
200
250
300
350
400

6c



6d

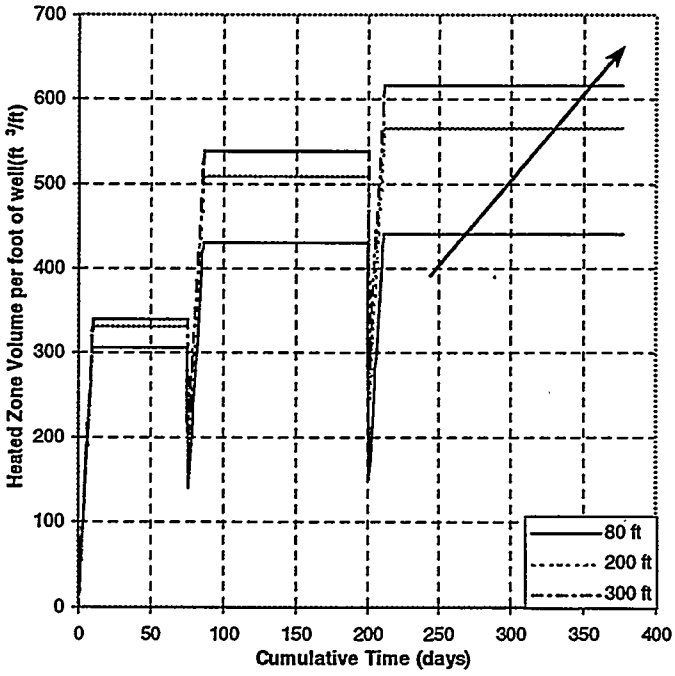
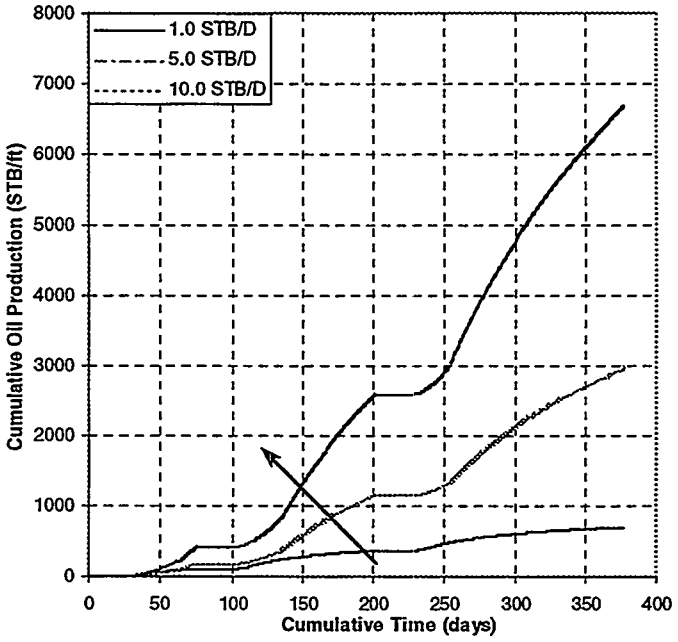
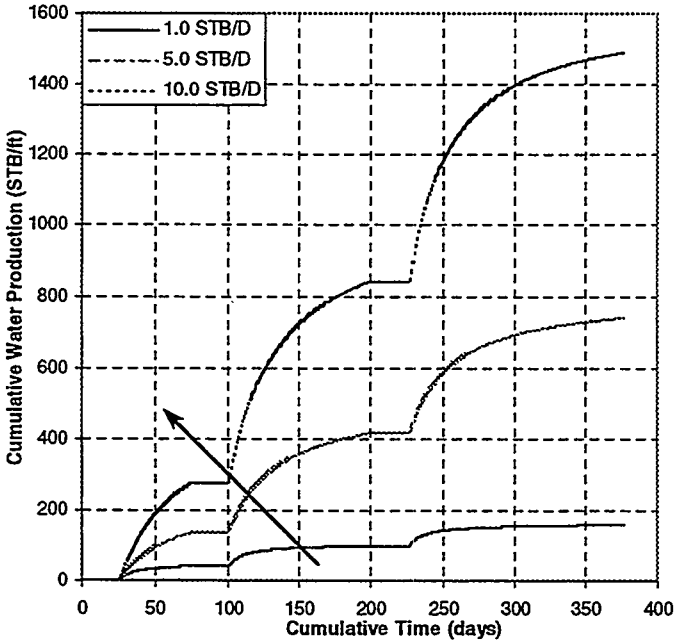


Fig. 6 Sensitivity formation thickness:
- a cumulative oil production versus cumulative time,
- b cumulative water production versus cumulative time,
- c average steam zone temperature versus cumulative time,
- d steam zone volume versus cumulative time.

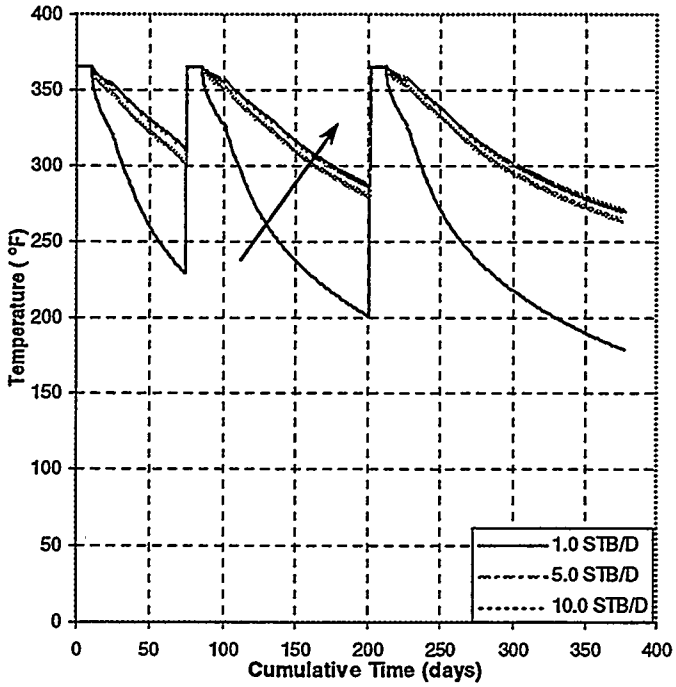
7a


1.0 STB/D
5.0 STB/D
10.0 STB/D
Cumulative Oil Production (STB/ft)
Cumulative Time (days)
0
1000
2000
3000
4000
5000
6000
7000
8000
0
50
100
150
200
250
300
350
400

7b


1.0 STB/D
5.0 STB/D
10.0 STB/D
Cumulative Water Production (STB/ft)
Cumulative Time (days)
0
200
400
600
800
1000
1200
1400
1600
0
50
100
150
200
250
300
350
400

7c



7d

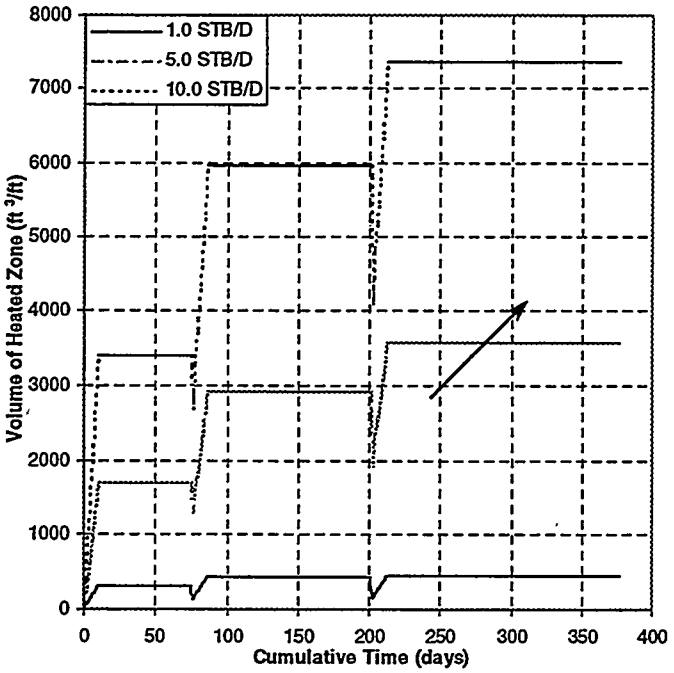
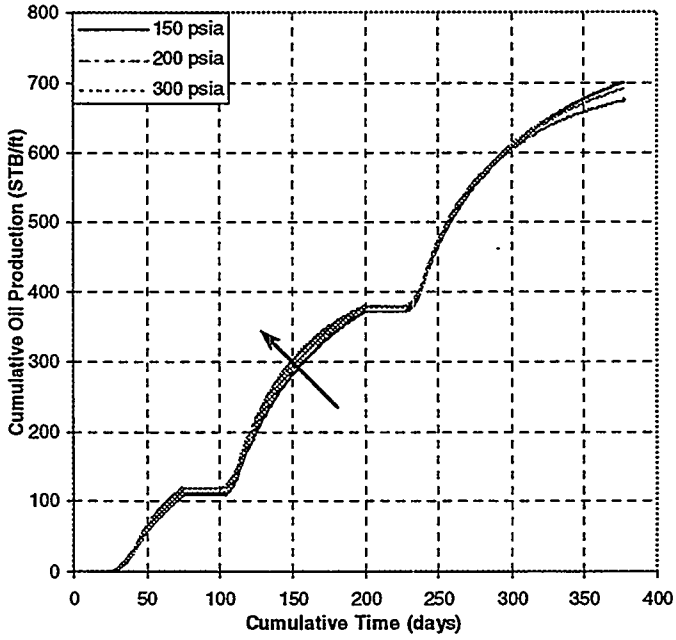


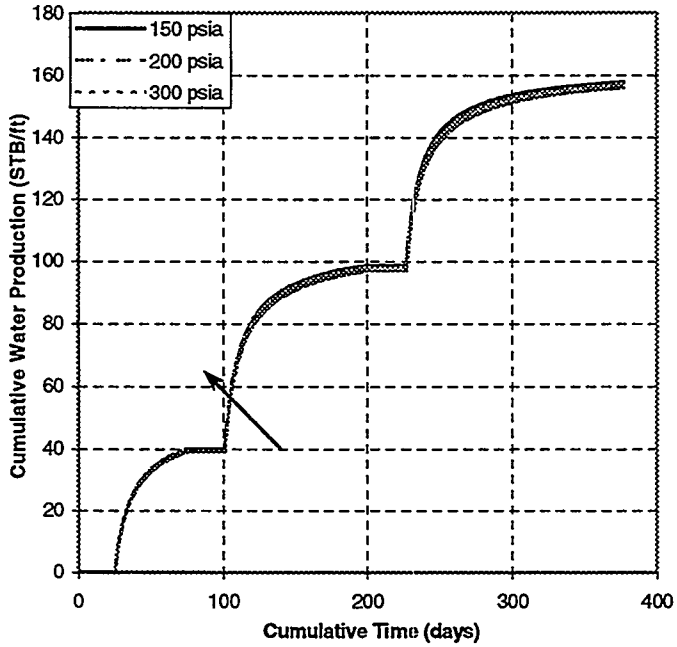
Fig. 7 Sensitivity to steam injection rate:

a cumulative oil production versus cumulative time,
b cumulative water production versus cumulative time,
c average steam zone temperature versus cumulative time,
d steam zone volume versus cumulative time

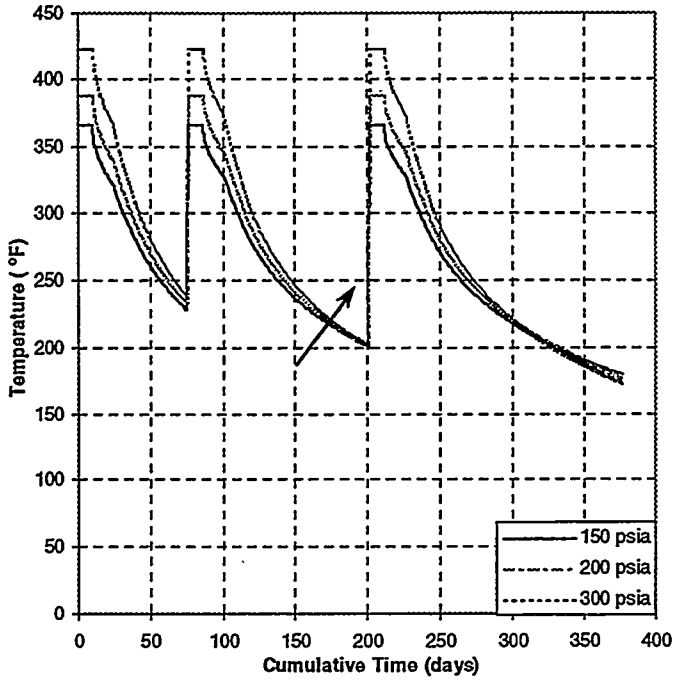
8a


150 psia
200 psia
300 psia
Cumulative Oil Production (STB/ft)
Cumulative Time (days)
0
100
200
300
400
500
600
700
800
0
50
100
150
200
250
300
350
400

8b


150 psia
200 psia
300 psia
Cumulative Water Production (STB/ft)
Cumulative Time (days)
0
20
40
60
80
100
120
140
160
180
0
100
200
300
400

8c



8d

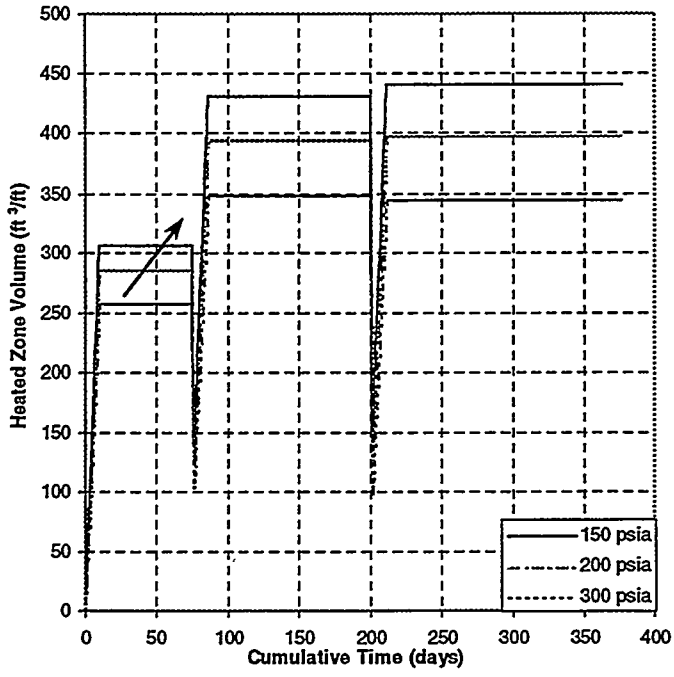
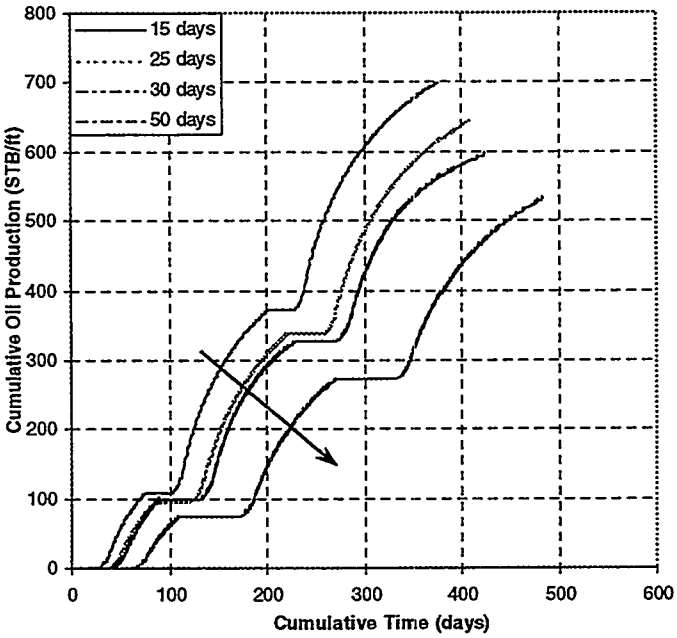


Fig. 8 Sensitivity to bottom-hole pressure:
- a cumulative oil production versus cumulative time,
- b cumulative water production versus cumulative time,
- c average steam zone temperature versus cumulative time,
- d steam zone volume versus cumulative time

9a



9b

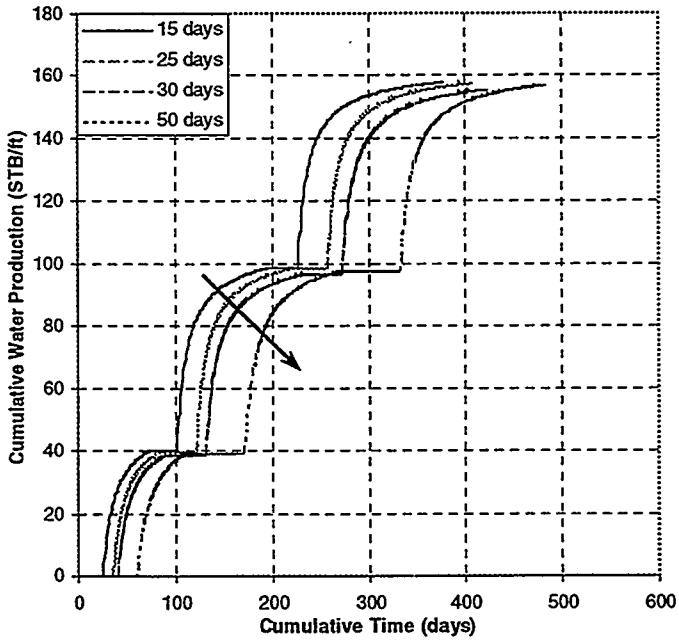


Fig. 9 Sensitivity to soaking interval:
a cumulative oil production versus cumulative time
b cumulative water production versus cumulative time

Appendix A

Derivation for calculating f_{HD}

The heat transfer model for conduction cooling of the heated zone can be solved for the horizontal and vertical conduction mechanisms. We have solved the horizontal heat transfer mechanism problem. When conduction occurs as described, the temperature at any point within the heated geometry can be expressed as the product:

$$v = V v_x v_z$$
$$V = T_s - T_R \tag{A-1}$$

where, v_x and v_z are unit solutions of component conduction problems in the x and z directions, respectively. Similarly, an integrated average temperature for the heated regions may be computed as:

$$\bar{v} = V \bar{v}_x \bar{v}_z \tag{A-2}$$

The average unit solution for $\bar{v}_x$ is obtained by solving the one-dimensional heat conduction problem in the horizontal direction:

$$\alpha \frac{\partial^2 v_x}{\partial x^2} = \frac{\partial v_x}{\partial t} \tag{A-3}$$

Initial and Boundary conditions:

1. $v_x = 1 \; ; \; t = t_{init} \; ; \; 0 < x < a$
2. $v_x = 0 \; ; \; t = t_{init} \; ; \; |x| > a$
3. $v_x = 0 \; ; \; t \geq t_{init} \; ; \; x \rightarrow \infty$

Carlslaw and Jaeger have given the solution for the problem at hand with the boundary condition specified as:

$$v_x = \frac{1}{2}\left\{ erfc\left(\frac{a-x}{2\sqrt{\alpha t}}\right) + erfc\left(\frac{a+x}{2\sqrt{\alpha t}}\right)\right\} \tag{A-4}$$

This solution holds at any vertical distance ' y ' upwards from the horizontal well as shown in the figure. To find the average $\bar{v}_x$ over the entire triangular heated area, we have,

$$\bar{v}_x = \frac{\int_0^A v_x dA}{\int_0^A dA} \tag{A-5}$$

If we take an elementary strip parallel to the x-axis, we will be integrating the given function with respect to x. The ends of this strip are bounded by the lines x = $(b/h_t).y$ and x = $-(b/h_t).y$; so that these are the limits of integration with respect to x. Next, we integrate with respect to y from y = 0 to y = h_t. This, therefore, covers the whole area of the triangle.

Thus, the integral reduces to:

$$\bar{v}_x = \frac{\int_{-\frac{b}{h_t}y}^{\frac{b}{h_t}y}\int_0^{h_t} v_x dxdy}{\int_{-\frac{b}{h_t}y}^{\frac{b}{h_t}y}\int_0^{h_t} dxdy} \tag{A-6}$$

The final integration result is:

$$\bar{V}_x = \frac{1}{2} + \sqrt{\frac{t_{DH}}{\pi}}\left\{-2.0 + \exp\left(-\frac{1}{t_{DH}}\right)\right\} + erf\left(\frac{1}{\sqrt{t_{DH}}}\right)\left\{\left(\frac{2}{\sqrt{\pi}} - 1\right)t_{DH} + \frac{1}{2}\right\} \quad \text{(A-7)}$$

where,

$$t_{DH} = \frac{\alpha\left(t - t_{inj}\right)}{R_h^2} \quad \text{(A-8)}$$

R_h is the horizontal range of the extent of the steam zone and is given by,

$$R_h = h_t \cot(\theta) \quad \text{(A-9)}$$

Appendix B

Code

```
#include <iostream.h>
#include <math.h>
#include <stdio.h>
#include <fstream.h>
#include <stdlib.h>
#include <cstdlib>
#include <iomanip.h>


#define     pi              3.142
#define     g               32.20
#define     length          1.0             // Dimension in the y -
direction is unity.
#define     m               3.5
#define     convFactor      0.001386
#define     numCycles       3
#define     maxNumCycles    numCycles+1

typedef struct {
      double API, alpha, Mdryrock, phi, phif, Ps, rinj[maxNumCycles],
rl;
      double Sor, Soi, Sorw, Sors, Swrs, Sw, Swi, Sw_o, Cw;
      double Tavg, Ts, Ti, Tsoak, Tprod, Xinj, wellRadius;
      double absolutePerm, steamRelperm, krw, kro, pwf;
// pwf = downhole flowing pressure
      double injTime[maxNumCycles], soakTime[maxNumCycles],
            prodTime[maxNumCycles], totalTime, alpha_s, alpha_r, Mr, Ms,
            Mo, Mw;
      double potentialGradient, heightDifferential, angleOfInclination;
      double tD, tcD, fhv, timeStep, time;
      double qo, qw, qocum, qwcum;
      double Swc, Vreservoir;
      double cumTime, WIP[maxNumCycles];
      double cum_oil[maxNumCycles], cum_wat[maxNumCycles];
      double cum_oil_tot, cum_wat_tot, cum_steam_tot;
      double Vx, Vy, OSR, WOR, tDvx;
      double voil, vwat;
      double gradP, Pwf;
} dataRec;

typedef struct {
      double steamZoneVolume, steamZoneThickness, steamZoneArea,
      RateofAreaChange;
      double steamZoneRadius, skin, externalReservoirRadius,
      reservoirThickness, steamFlowRate[maxNumCycles];
      double Q[maxNumCycles], Q_old[maxNumCycles], Qi[maxNumCycles],
      Qi_old[maxNumCycles], Ql[maxNumCycles], Ql_oil[maxNumCycles],
      QiRate[maxNumCycles], QiRate_old[maxNumCycles];
      double Ehs, G, f[3];
      double prevSteamZoneVolume, Area, OIP;
} zoneRec;
```

```
void RunProgram(void);
void GetData(dataRec &data, zoneRec &zone);
void GetSteamZoneParameters(dataRec &data, zoneRec &zone, int cycle);
void SAGD(dataRec &data, zoneRec &zone, int cycle);
void GetfVDfHD(dataRec &data, zoneRec &zone, int cycle);
void GetfPD(dataRec &data, zoneRec &zone, int cycle);
void GetTemperatures(dataRec &data, zoneRec &zone, int cycle);
void GetVolumetricHeatCapacities(dataRec &data, zoneRec &zone);
void GetEhs(dataRec &data, zoneRec &zone);
void GetRelpermOil(dataRec &data, zoneRec &zone, int cycle);
void GetRelpermWat(dataRec &data, zoneRec &zone, int cycle);
void GetSaturation(dataRec &data, zoneRec &zone, int cycle);
double factorial(int n);
double GetInstantaneousWaterRate(dataRec &data, zoneRec &zone, int
cycle);
double GetInstantaneousOilRate(dataRec &data, zoneRec &zone, int cycle);
double GetCumulativeOilProd(dataRec &data, zoneRec &zone);
double GetCumulativeWaterProd(dataRec &data, zoneRec &zone);
double FindtcD(dataRec &data, zoneRec &zone);
double erfc(double x);
double derfc(double x);
double Function(double hd, double ax);
double Derivative(double hd, double ax);
double GetIntegralTerm(double tcD, double tD);
double h_wat(double Ts);
double h_steam(double Ts);
double density_oil(double API, double Ts);
double density_wat(double Ts);
double density_steam(double Ts);
double heatcap_oil(double API, double Ts);
double latent_heat_vapor(double Ts);
double viscosity_oil(double Ts);
double viscosity_wat(double Ts);
double viscosity_steam(double Ts);
double visckin_oil(double ro, double v_oil);
double GetWIP(dataRec &data, zoneRec &zone, int cycle);


void main()
{

      RunProgram();

      // Output files are stored in m:\myhill-stegemeir

      system("cd m:\\myhill-stegemeir");
      system("gps curr.gps curr.gs");
      system("s:\\gstools\\gsview\\gsview32 m:\\myhill-
stegemeir\\curr.gs");

}


void RunProgram(void)
{
      int cycle;
```

```
	dataRec data; zoneRec zone;

	// Invoking input data function

	GetData(data, zone);

	//Storing data in output file

	ofstream outfile("curr.dat", ios::app);

	outfile << "Time" << setw(10) << "CumTime " << setw(15) << "Vs" <<
setw(20) << "Ehs" << setw(20) << "Tavg F" << setw(20) << "angle" <<
setw(20) << "Cum Oil" << setw(20) << "Cum Water" << setw(20) <<
"qoil(cuft/d)" << setw(20) << "qwater (cuft/d)" << setw(20) << "cumoil
(cyc)" << setw(20) << "cumwat (cyc)" << endl;
	for (cycle=1; cycle<=numCycles; cycle++) {
		SAGD(data, zone, cycle);
		zone.prevSteamZoneVolume = zone.steamZoneVolume;
		zone.Q[cycle] = data.Mr * zone.steamZoneVolume * (data.Tavg
		- data.Ti);
	}
}


// Input data function

void GetData(dataRec &data, zoneRec &zone)
{
	int k;
	for(k=1; k<=numCycles; k++) {
		zone.Q[k] = zone.Qi[k]  = zone.Ql[k] = 0.0;
	}
	data.qocum = data.qwcum = data.cum_oil_tot = data.cum_wat_tot =
	0.0;
	data.cum_steam_tot	= 0.0;
	data.cumTime		= 0.0 ;
	data.qo		= data.qw= 0.0 ;
	data.timeStep		= 1.  ;	//day
	data.absolutePerm	= 1.5 ;
	data.phi		= 0.32;	//rock porosity
	data.Xinj		= 0.70;	//X{inj steam quality}
	data.Sor		= 0.35  ;
	data.Mdryrock		= 40.0;
	//dry rock volumetric heat capacity
	data.alpha		= 0.48;
	//thermal diffusivity of rock (sq.ft/D)
	data.alpha_r		= data.alpha_s = data.alpha;
	data.rl			= 24.0;
	//thermal conductivity of over and underburden rock (Btu/ft.D.F)
	data.phif		= 0.25;
	//porosity of adjacent formations
	data.Ms			= data.rl/data.alpha;
	data.Ps			= 150. ;
	//Bottomhole injection pressure in psia
	data.Ts			= 115.95*pow((data.Ps+14.7),.225);
	data.Ti			= 110 ;
	//Reservoir temperature before first cycle began
```

```
//Ti is the temperature of the reservoir at the end of the previous
//cycle
      data.API                        = 14.0;     //API gravity of the oil
      data.wellRadius                 = 0.31;     //feet
      data.Sors                       = 0.05;
      data.Sorw                       = 0.25;
      data.Swi                        = 0.25;
      data.Swc                        = 0.1 ;
      data.Soi                        = 1.0 - data.Swi;
      data.steamRelperm               = 0.25;     //Darcy
      zone.reservoirThickness         = 80.0;
      zone.Area                       = 0.588*43560.0 ; //0.588 acres
      data.Vreservoir                 = zone.reservoirThickness*zone.Area;
      zone.OIP                        =
      zone.Area*zone.reservoirThickness*data.phi*data.Soi;
      zone.prevSteamZoneVolume= 0.0 ;
      data.Pwf                        = 14.7;

      for(k=1; k<=numCycles; k++) {
            data.injTime[k]           = 10.       ;
            data.soakTime[k]          = 15.       ;
            data.prodTime[k]          = k*50.0    ;
            data.rinj[k]              = 1.0       ;
//Steam injection ratein B/D/ft of equivalent water into reservoir
            zone.steamFlowRate[k]     = data.Xinj*data.rinj[k];
            data.WIP[k]               = 0.0;
      }
}

double FindtcD(dataRec &data, zoneRec &zone)
{
      double eps  = 1.0 , xold, xnew ;
      xold        = 1.5;
      while(eps>0.00001) {
            if(Derivative(data.fhv, xold)==0.0) {
                  cout << "System breaking out as derivative in Newton's
method is zero"; break;
            }
            xnew  = xold - Function(data.fhv, xold)/Derivative(data.fhv,
            xold);
            eps         = fabs(xnew-xold);
            xold  = xnew;

      }
      return xnew;
}


//    Solving the SAGD problem

void SAGD(dataRec &data, zoneRec &zone, int cycle)
{
      data.cum_oil[cycle] = data.cum_wat[cycle] = 0.0;
      ofstream outfile("curr.dat",ios::app);
      cout << "Time (days)" << setw(10) << "Sw" << setw(10) <<
            "WIP" << setw(10) << "Swo" << setw(10) << "Krw" << endl;
```

```
      if(cycle>1) {
                  zone.Qi[cycle] += zone.Q[cycle-1];
                  }

// Start cycle calculations

      for (data.time=0.0; data.time <=
      data.injTime[cycle]+data.soakTime[cycle]+data.prodTime[cycle];

      data.time += data.timeStep) {
            if(data.time <= data.injTime[cycle]) {
                  data.cum_steam_tot +=
                  5.615*data.rinj[cycle]*data.timeStep;
                  data.Tavg = data.Ts;
                  }
            if(data.time == 0.0) {
                  GetVolumetricHeatCapacities(data, zone);
                  data.tcD = FindtcD(data, zone);
            }
            GetVolumetricHeatCapacities(data, zone);
            GetSteamZoneParameters(data, zone, cycle);
            GetTemperatures(data, zone, cycle);

            if(data.time>data.injTime[cycle]+data.soakTime[cycle]) {
                  data.qo     = GetInstantaneousOilRate(data, zone,
                  cycle);
                  data.qw     = GetInstantaneousWaterRate(data, zone,
                  cycle);
                  data.cum_oil_tot += data.qo;
                  data.cum_wat_tot += data.qw;
                  data.cum_oil[cycle] += data.qo;
                  data.cum_wat[cycle] += data.qw;
            }
            if(data.time <= data.injTime[cycle]) {
                  data.Tavg = data.Ts;
            }
            data.WOR = data.qw/data.qo;
            data.OSR = data.cum_oil_tot/data.cum_steam_tot;
            outfile << data.time << setw(5) << data.cumTime << setw(15)
            << zone.steamZoneVolume << setw(15) << zone.Ehs <<
            setw(20) << data.Tavg << setw(20) << data.angleOfInclination
            << setw(20) << data.cum_oil_tot << setw(20) <<
            data.cum_wat_tot << setw(20) << data.qo << setw(20) <<
            data.qw << setw(20) << data.cum_oil[cycle] << setw(20) <<
            data.cum_wat[cycle] << endl;
            ofstream ofile("ratios.dat", ios::app);
            ofile << data.cumTime << setw(20) << data.WOR << setw(20) <<
            data.OSR << endl;
            ofstream sat("sat.dat", ios::app);
            sat << data.cumTime << setw(20) << data.Sw_o << setw(20) <<
            data.krw << setw(20) << data.kro << setw(20) <<
            data.angleOfInclination << endl;
            ofstream vx("vxtx.dat", ios::app);
            vx << data.time << setw(20) << data.tDvx << setw(20) <<
            data.Vx << setw(20) << data.OSR << endl;
            data.cumTime+=1.0;
            }
```

```
}


// Function to calculate the volumetric heat capacities and the
dimensionless
// time.

void GetVolumetricHeatCapacities(dataRec &data, zoneRec &zone)
{
      data.Cw = (h_wat(data.Ts)-h_wat(data.Ti))/(data.Ts-data.Ti);
      data.fhv= pow(1.0+data.Cw*(data.Tavg-
      data.Ti)/(latent_heat_vapor(data.Tavg)*data.Xinj),-1.0);
      data.Mw = data.Cw*density_wat(data.Tavg);
      data.Mo =
      heatcap_oil(data.API,data.Tavg)*density_oil(data.API,data.Tavg);
      double s=   h_wat(data.Tavg)*density_steam(data.Tavg);
      data.Mr = 32.5 + (4.6*pow(data.phi, 0.32) - 2.0) * (10.0*data.Swi
      - 1.5);
      data.tD =
4*pow((data.Ms/data.Mr),2.)*data.alpha*data.time*30.4/(pow(zone.reservoi
rThickness,2.));
}


// Function to calculate steam zone volume, steam zone horizontal range,
// angle of inclination with the horizontal as well as the steam zone
area

void GetSteamZoneParameters(dataRec &data, zoneRec &zone, int cycle)
{
      GetEhs(data, zone);
      double wt    =     5.615*data.rinj[cycle]*62.4;
      double ws    =     data.Xinj*wt;
      if( data.time != 0.0 && data.time <= data.injTime[cycle]) {
            zone.QiRate[cycle]= (wt*(h_wat(data.Tavg)-
            h_wat(data.Ti))+ws*latent_heat_vapor(data.Tavg));
            zone.Qi[cycle]     +=(wt*(h_wat(data.Tavg)
            h_wat(data.Ti))+ws*latent_heat_vapor(data.Tavg))
            *data.timeStep;
            zone.Ql[cycle]     += zone.QiRate[cycle]*(1.0-
            exp(data.tD)*erfc(fabs(sqrt(data.tD))))*data.timeStep;
      }
      if (data.time <= data.injTime[cycle]) {
            zone.steamZoneVolume =
            zone.Ehs*zone.Qi[cycle]/(data.Mr*(data.Ts-data.Ti));
            zone.steamZoneArea =
            zone.steamZoneVolume/zone.reservoirThickness;
            zone.steamZoneRadius = zone.steamZoneArea/(length);
            data.angleOfInclination =
            (180.0/pi)*atan(zone.steamZoneRadius/
            zone.reservoirThickness);
      }

}

// Function to calculate the thermal efficiency
```

```
void GetEhs(dataRec &data, zoneRec &zone)
{
      double U;
      double y = sqrt(data.tD); double a =exp(data.tD); double c=
      erfc(y);
      zone.G                  =     2.*fabs(sqrt(data.tD/pi))-1.+a*c;
      if(data.tD <= data.tcD) U     =     0.0;
      if(data.tD > data.tcD)  U     =     1.0;
      if(U != 0.0) {
      double IntegralTerm = GetIntegralTerm(data.tcD,data.tD);
      zone.Ehs = (zone.G+((1-data.fhv)*U/sqrt(pi))*(2.*sqrt(data.tD)-
      2.*(1-data.fhv)*sqrt(fabs(data.tD-data.tcD))
      -IntegralTerm-sqrt(pi)*zone.G))/data.tD;
      zone.Ehs = (1.0+zone.Ehs)/2.0;
      }
      if(data.tD == 0.0 ) zone.Ehs = 1.0;
      if(U == 0.0 && data.tD > 0.0 ) zone.Ehs =
      (1.0+zone.G/data.tD)/2.0;
}

double erfc(double x) //Chebyshev's approximation
{
      double z    = fabs(x);
      double t    = 1.0/(1.0+0.5*z);
      double ans  = t*exp(-z*z-
1.26551223+t*(1.00002368+t*(0.37409196+t*(0.09678418+t*(-0.18628806+t*
                        (0.27886807+t*(-1.13520398+t*(1.488851597+t*(-
0.82215223+t*0.17087277)))))))));
      return ans;
}


double factorial(int n)
{
      if(n==0.0) {return 1;}
      int product = 1;
      for (int i = 1; i <= n; i++) {
            product *= i;
      }
      return (product);
}


double GetIntegralTerm(double tcD, double tD)
{
      double vol              =     0.0;
      if(tD==0) { return 0.0; }
      for (double low=0; low<=tcD; low+=0.001) {
            double up         =     low+0.001;
            vol               +=    (exp(low)*erfc(sqrt(low))/sqrt(tD-
low)+exp(up)*erfc(sqrt(up))/sqrt(tD-up))*(up-low)*0.5;
      }
      return vol;
}

// Function to calculate temperature of the steam zone as a function of
time
```

```
void GetTemperatures(dataRec &data, zoneRec &zone, int cycle)
{
      if(data.time == 0.0) {
            data.Tavg = data.Ts;
      } else if(data.time <= data.injTime[cycle] && data.time >=
      data.timeStep) {
            data.Tavg = data.Ts;


      } else if(data.time>data.injTime[cycle] &&
      data.time<=data.injTime[cycle]+data.soakTime[cycle]) {
            zone.f[2]=0.0;
            GetfVDfHD(data, zone, cycle);
            data.Tavg = data.Ti+(data.Ts-
            data.Ti)*(zone.f[1]*zone.f[0]*(1-zone.f[2])-zone.f[2]);

            if(data.time==data.injTime[cycle]+data.soakTime[cycle])
            {data.Tsoak = data.Tavg; }
      } else {
            GetfPD(data, zone, cycle);
            GetfVDfHD(data, zone, cycle);
            data.Tavg = data.Ti+(data.Ts-data.Ti)*(zone.f[1]*
            zone.f[0]*(1-zone.f[2])-zone.f[2]);

      }
      if(data.Tavg <= data.Ti) data.Tavg = data.Ti;

}

// Function to calculate the unit solutions to the one-dimensional heat
// conduction problem

void GetfVDfHD(dataRec &data, zoneRec &zone, int cycle)
{
      if(data.time == 0.0 || data.time<=data.injTime[cycle]) {
            zone.f[0] = zone.f[1] = 0.0;
            } else {
            double alpha_z =
0.5*data.alpha*(zone.reservoirThickness/sqrt(pow(zone.steamZoneRadius,2)
+pow(zone.reservoirThickness,2))+1.0);
            double tdv  =     4*alpha_z*(data.time-
data.injTime[cycle])/pow(zone.reservoirThickness,2);
            zone.f[1]   =     1.0/sqrt(1+5.*tdv);
            double alpha_x =
data.alpha*zone.steamZoneRadius/sqrt(pow(zone.steamZoneRadius,2)+
                  pow(zone.reservoirThickness,2));
            double tdh  =     alpha_x*(data.time-
data.injTime[cycle])/pow(zone.steamZoneRadius,2);
            zone.f[0]   =     1./(1.+5.*tdh);
            double tDx = data.tDvx = alpha_x*(data.time-
data.injTime[cycle])/pow(zone.steamZoneRadius,2);
            double s1  = sqrt(tDx/pi)*(-2.0+exp(-1.0/tDx));
            double s3  = ((2/sqrt(pi))-1.0)*tDx+0.5;
            double s2  = 1.0-erfc(1/sqrt(tDx));

                  data.Vx = 0.5+s1+s3*s2;
```

```
                zone.f[0] = data.Vx;

            }
}


void GetfPD(dataRec &data, zoneRec &zone, int cycle)
{
      double tempTime         =     data.time;
            double wt               =     5.615*data.rinj[cycle]*62.4;
      //62.4 x 5.615 = 350.376
      double Hinj           = zone.Qi[cycle];
      double Hlast      =     zone.steamZoneVolume*data.Mr*(data.Tavg-
      data.Ti);
      double Qmax             =     Hinj+Hlast-
      2*zone.steamZoneRadius*length*data.rl*(data.Ts-data.Ti)
      *sqrt(data.soakTime[cycle]/(pi*data.alpha));
      double q_o1=GetInstantaneousOilRate(data, zone, cycle);
      double q_w1=GetInstantaneousWaterRate(data, zone, cycle);
      double add = (q_o1*data.Mo+q_w1*data.Mw)*(data.Tavg-data.Ti);
      double delf = 0.5*add*data.timeStep/Qmax;
      double fPD  = zone.f[2];
      if(data.Tavg <= data.Ti) {
            zone.f[2]   +=    0.0;
      } else {
            zone.f[2]   +=    0.5*add*data.timeStep/Qmax;
      }
}


double derfc(double x)
{
      double derf = 0.0;
      double mult = 2./sqrt(pi);
      for(int count=1; count<=10; count++) {
            derf += mult*pow(-1, count+1)*(1+(count-1)*2.)*pow(x,
2*(count-1))/((1+(count-1)*2)*factorial(count-1));
      }
      return -derf;
}



double Function(double hd, double ax)
{
      double f    = exp(ax)*erfc(sqrt(ax))-1.0+hd;
      return f;
}


double Derivative(double hd, double ax)
{
      double deldf= 0.00001;
      double df   = (exp(ax+deldf)*erfc(sqrt(ax+deldf))-
exp(ax)*erfc(sqrt(ax)))/deldf;
      return df;
```

```
}


// The following functions calculate the instantaneous oil and water
rates in the
// production interval

double GetInstantaneousOilRate(dataRec &data, zoneRec &zone, int cycle)
{

      GetRelpermOil(data, zone, cycle);
      double den_o = density_oil(data.API, data.Tavg);
      double den_s = density_steam(data.Tavg);
      data.voil = viscosity_oil(data.Tavg);
      double theta = pi/2-(data.angleOfInclination)*pi/180;
      data.Ps = pow(data.Tavg/115.95, 4.4543);
      data.gradP = 144.0*(data.Ps - data.Pwf)/(den_o-den_s);
      double cot = 1.0/tan(0.5*theta);
      data.potentialGradient = 2.0*data.gradP*log(cot)+
            zone.reservoirThickness*g*2*fabs(pi-2.0*theta);
      data.qo     = convFactor*data.absolutePerm*(den_o-den_s)*data.kro*
            data.potentialGradient/data.voil;
      return data.qo;
}

double GetInstantaneousWaterRate(dataRec &data, zoneRec &zone, int
cycle)
{
      GetRelpermWat(data, zone, cycle);
      double den_w = density_wat(data.Tavg);
      double den_s = density_steam(data.Tavg);
      data.vwat   = viscosity_wat(data.Tavg);
      double theta = pi/2-(data.angleOfInclination)*pi/180;
      data.Ps = pow(data.Tavg/115.95, 4.4543);
      data.gradP = 144.0*(data.Ps - data.Pwf)/(den_w-den_s);
      double cot = 1.0/tan(0.5*theta);
      data.potentialGradient = 2.0*data.gradP*log(cot)+
            zone.reservoirThickness*g*fabs(pi-2.0*theta);
      data.qw     = convFactor*data.absolutePerm*(den_w-den_s)*data.krw*
            data.potentialGradient/data.vwat;
      return data.qw;
}

double GetCumulativeOilProd(dataRec &data, zoneRec &zone)
{
      data.qocum += data.qo*data.timeStep;
      return data.qocum;
}

double GetCumulativeWaterProd(dataRec &data, zoneRec &zone)
{
      data.qwcum += data.qw*data.timeStep;
      return data.qwcum;
}

// Functions to calculate relative permeability of the oil and water
```

```
void GetRelpermWat(dataRec &data, zoneRec &zone, int cycle)
{
      GetSaturation(data, zone, cycle);
      data.krw=-0.002167*data.Sw_o+0.024167*pow(data.Sw_o,2);

}

void GetRelpermOil(dataRec &data, zoneRec &zone, int cycle)
{
      GetSaturation(data, zone, cycle);
      if (data.Sw_o<0.56443) { data.kro = 1.0;
      } else {
      data.kro=-0.9416+(1.0808/data.Sw_o)-0.13856/pow(data.Sw_o,2);
      }
}

// Function to calculate the saturations of oil and water

void GetSaturation(dataRec &data, zoneRec &zone, int cycle)
{
      double Sw_bar=1.-data.Sorw;
      double WIP = GetWIP(data, zone, cycle);
      double cumwaterprod = GetCumulativeWaterProd(data, zone);
      data.Sw=Sw_bar-(Sw_bar-data.Swi)*data.cum_wat[cycle]/WIP;
      data.Sw_o=(data.Sw-data.Swi)/(1-data.Swi-data.Sorw);
}

double GetWIP(dataRec &data, zoneRec &zone, int cycle)
{
      if(cycle == 1) {
            data.WIP[cycle] =
            5.615*data.rinj[cycle]*data.injTime[cycle];
      } else {
            data.WIP[cycle] =
      data.rinj[cycle]*5.615*data.injTime[cycle]+data.WIP[cycle-1]
      -data.cum_wat[cycle-1];
      }
      return data.WIP[cycle];
}
```